Modeling Underwater Explosion (UNDEX) Shock Effects for Vulnerability Assessment in Early Stage Ship Design

Ajai Kurian Mathew

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

MASTER OF SCIENCE

in

Ocean Engineering

Alan J. Brown, Chair
Robert A. Canfield
Kevin G. Wang

February 12, 2018
Blacksburg, VA

Keywords: Underwater Explosion, Shock, Naval Ship Design

Modeling Underwater Explosion (UNDEX) Shock Effects for Vulnerability Assessment in Early Stage Ship Design

Ajai Kurian Mathew

ABSTRACT

This thesis describes and assesses a simplified tool for modeling underwater explosion shock effects during early naval ship concept design. A simplified fluid model using Taylor flat-plate theory is incorporated directly into the OpenFSI module code in Nastran and used to interface with the structural solver in Nastran to simulate a far-field shockwave impacting the hull. The kick-off velocities and the shock spectra captured in this computationally efficient module is compared to results from a high-fidelity CASE (Cavitating Acoustic Spectral Element) fluid model implemented with the ABAQUS/Nastran structural solver to validate the simplified framework and assess the sufficiency of this very simple but, fast approach for early stage ship design.

KEYWORDS: Underwater Explosion, Shock, Naval Ship Design

Modeling Underwater Explosion (UNDEX) Shock Effects for Vulnerability Assessment in Early Stage Ship Design

Ajai Kurian Mathew

GENERAL AUDIENCE ABSTRACT

This thesis describes and assesses a simplified tool for modeling underwater explosion shock effects during early-stage naval ship design. Far-field explosions have a significant effect in terms of damage to equipment and mission capability of a ship. A simplified fluid-structure interaction model using the concept "Taylor flat-plate theory" is developed to simulate a far-field shockwave impacting the hull. This model is directly incorporated inside 'OpenFSI', a module used to couple an external code with the Nastran structural solver software. The initial peak velocity in the time-history and the shock spectra characteristics captured in this computationally efficient module is compared to results from a high-fidelity "CASE" (Cavitating Acoustic Spectral Element) fluid-structure interaction model. The "CASE" model implemented with the ABAQUS/Nastran structural solver is used to validate the simplified framework and assess the sufficiency of this very simple, but fast approach for early stage ship design.

KEYWORDS: Underwater Explosion, Shock, Naval Ship Design

ACKNOWLEDGEMENTS

I wish to express my sincere and deepest gratitude to Dr. Alan J. Brown for his inspiring guidance and supervision at all stages of my thesis work. I deeply appreciate your patience and understanding.

I would like to thank my committee members, Dr. Robert A. Canfield and Dr. Kevin G. Wang for their advice, support and assistance.

I am extremely grateful to my co-researcher Zhaokuan Lu for his collaboration and support throughout the thesis work. I would like to thank the AOE Department staff, Steve and Jonathan, for their prompt actions for all IT-related problems.

I am also thankful to my other fellow researchers and friends for their support and encouragement. Last, but not the least, I would like to deeply thank my parents and my brother, Vijai for their love, prayer and support.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

1 INTRODUCTION

1.1 Motivation

Major design decisions affecting ship underwater explosion (UNDEX) vulnerability are made very early in naval ship concept exploration when thousands of design alternatives must be assessed quickly. Currently vulnerability is not adequately assessed early resulting in the need to fix things later at great expense or not being able to meet survivability requirements when they are finally evaluated. This is largely due to the fact that the tools currently used in the design process are computationally time prohibitive and a significant amount of time and excessive detail is required to model each explosive case considered for each ship. As such, there is a need for a method and tools for modeling a ship's response to UNDEX phenomenon in early-stage ship design that can provide accurate results in a reasonably short time. This thesis represents an effort to develop a model and tool that fills this niche, specifically for far-field UNDEX shock problems. UNDEX shock effects and their impact on installed ship systems and personnel is critical in a naval ship. Figure 1 shows the destructive power UNDEX can have.

Figure 1: Shock trial of USS Jackson. [1]

1.2 Underwater Explosions – Background

1.2.1 Ship Survivability

An important criterion in the designing of vessels for the Navy and Coast Guard involves ensuring the survivability of a given vessel in hostile conditions. The standards for measuring survivability for the US Navy are given in OPNAVINST 9070.1A where survivability is defined as: "A measure of both the capability of the ship, mission critical systems, and crew to perform assigned warfare missions, and of the protection provided to the crew to prevent serious injury or death. Both of these capabilities are applicable whether in combat or in either combat or noncombat related accidents (e.g., groundings, collisions, fires)." [2]

As such, a Naval ship is expected to be able to survive the rigors of combat, sea states, accidents, exposure to biological agents, exposure to chemical agents, exposure to radiation, and other threats to the wellbeing of the ship and the crew[2]. This includes exposure to UNDEX threats.

Survivability is a probabilistic event and as such can also be defined as:

$$S = 1 - P_h P_{k/h}\left(1 - P_{r/k}\right) \tag{1.1}$$

where S is the probability of survival for a naval ship, P_h is the probability of the ship being hit by a threat or weapon, $P_{k/h}$ is the probability of a kill or loss of mission capability given the ship is hit, and $P_{r/k}$ is the probability of the ship recovering given a specific kill or loss. These probability terms are called susceptibility, vulnerability, and recoverability.

Susceptibility is defined as "the inability of a ship to avoid the sensors, weapons and weapons effects of that man-made hostile environment" [3]. Susceptibility, represented by P_h in the equation above, is assessed in three phases. First, the probability that the threat is active is considered. Second, the probability of enemy detecting, classifying and targeting a ship, is evaluated. Finally, the probability that the weapon will be successfully launched, flown out and hit the ship, is assessed [3]. For lowering susceptibility, active avoidance methods like maneuvering, striking the attacker, electronic warfare techniques like radar jamming, etc. and passive avoidance techniques like limiting ship signatures (radar cross-section, thermal, magnetic) are used [2].

Vulnerability is defined as the "the inability of the ship to withstand damage mechanisms from one or more hits, to its vincibility, and to its liability to serious damage or loss when hit by threat weapons" [4]. Various design features like shock hardening, separation and redundancy of

vital components, the size and compartmentation of the ship and structural detail affect vulnerability [4].

Recoverability is defined as the "ability of a ship and its crew to restore mission essential functions given a hit by one or more threat weapons" [4]. It depends on the damage control features on the ship, the quantity and type of damage control equipment on board and the ability/skill-set of the crew [4].

This thesis focuses on the vulnerability of a surface ship to a shock wave generated by a far-field underwater explosion (UNDEX).

1.2.2 Near-field Versus Far-field UNDEX Problems

In general, UNDEX phenomenon can be characterized as either being near or far field and early-time or late-time as shown in Figure 2. In the near-field case, the explosion typically is large and occurs close to the ship. For this type of problem, the predominant effects are the blast and the gas bubble, and the predominant concerns are plastic deformation and rupturing of the hull of the ship. There are also concerns with respect to loss of buoyancy in a given section of the ship which can cause structural collapse. Rupture or other types of structural collapse can result in the loss of a ship and the lives of the crew.

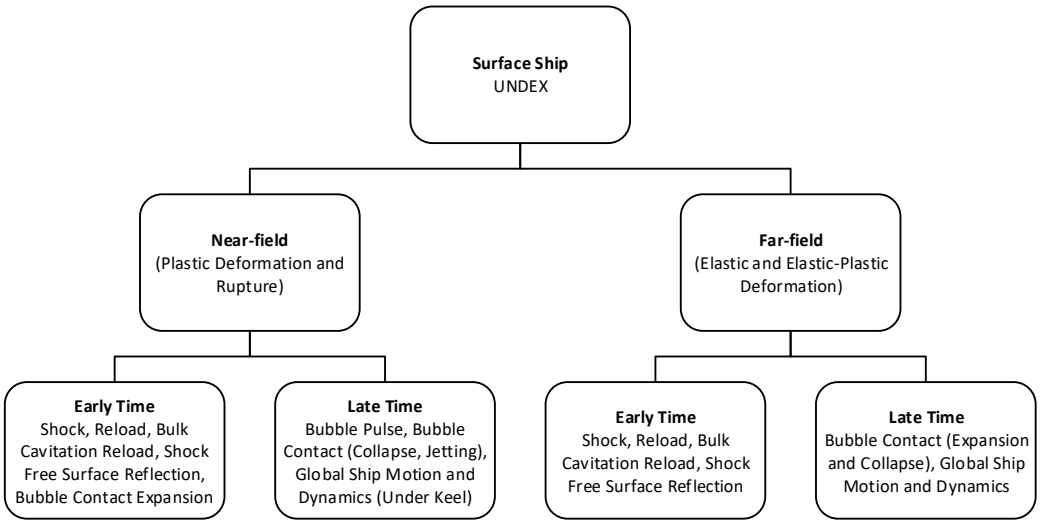

Figure 2: Classification of UNDEX Events. Adapted from [5]

Far field explosions occur at a distance where the gas bubble doesn't directly interact with the structure of the ship and results only in elastic or elastic-plastic deformation [6]. In this case, the ship is impacted primarily by the initial shock wave produced by the explosion and subsequent bubble pulses from the expansion/contraction cycle of the gas bubble. The distance at which these

deformations occur depends on the size and power of the charge and the structural characteristics of the ship. At a significant enough distance, the bubble pulses can be considered negligible. The primary concern for far field explosions is the impact on internal systems and ship personnel. Early time response is the result of the shock wave contacting the ship, while late time response is limited to lesser global ship motion and dynamics. For the purpose of this thesis, a far-field UNDEX event is defined as an UNDEX event that is far enough away from the ship to not cause hull rupture or plastic deformation or produce bubble pulses which influence the late time response of the ship.

Near-field explosion is more critical in terms of structural damage resulting in the failure of the ship although it may cause machinery damage too. Far-field explosions are critical even though it may not cause significant structural deformation as they might cause failure of Vital Components or Machinery and thus cause failure of the ship in terms of its mission capability. So it is important to study about the effects of far-field explosions too.

1.2.3 Detonation process in UNDEX Phenomena

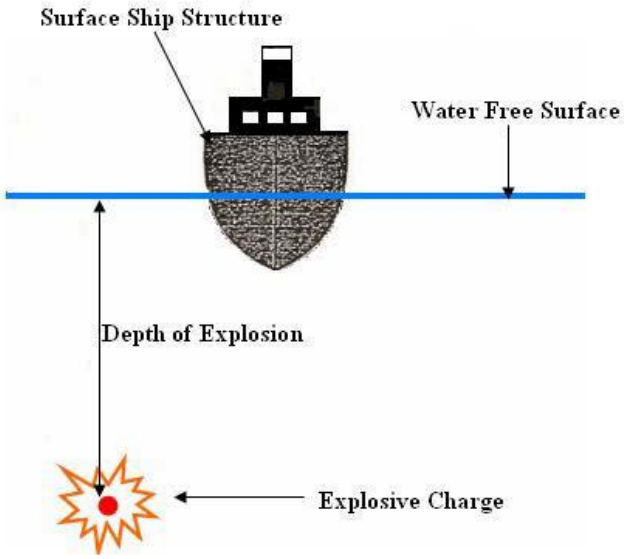

Figure 3: General geometry of UNDEX Problem

Figure 3 shows the general UNDEX problem with a ship or submarine close to an explosive charge of specific type and size and the detonation of the charge. Based on the weight of the charge, the position relative to the target structure (submarine/ship) and the response of the structure, the event maybe classified as either far-field or near-field events in the early or late time frames.

Explosive charges generally consist of two parts: an inherently unstable explosive material serves as the component that supports the reaction and a primer that serves to trigger the reaction in the unstable compound [7]. The primer is composed of a compound that is more sensitive but less energy dense than the compound responsible for the main effects of the explosive. When heat is added to the primer, generally via a firing pin, the primer detonates. The detonation of the primer causes an exothermic chemical reaction that is sufficient to detonate the surrounding explosive compound. A "detonation wave" propagates at supersonic speed through the unexploded high energy compound which almost instantaneously produces a gas bubble with a very high temperature and high pressure [7]. Figure 4 below illustrates the process.

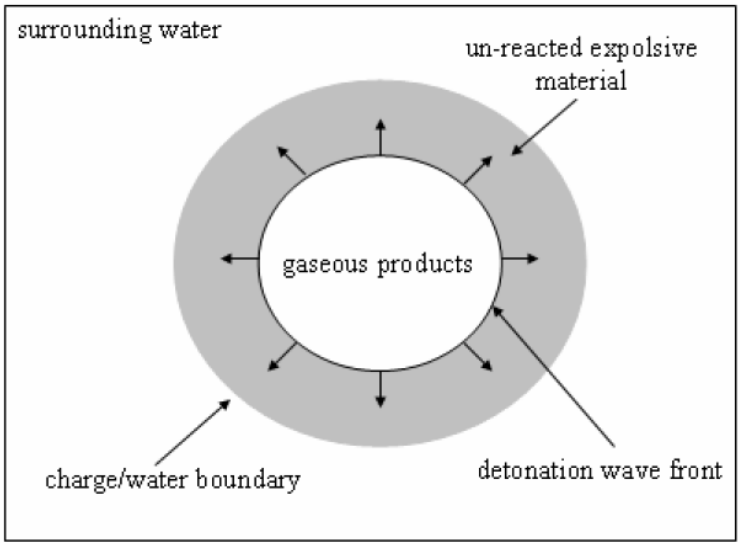

Figure 4: UNDEX Detonation Process Schematic [8]

By the time the detonation wave reaches the explosive material/water boundary, all unreacted material is converted now into a very dense, superheated, spherical gas bubble as shown in Figure 5 completing the UNDEX detonation process. This thesis treats the conclusion of this detonation process as the initial condition for an UNDEX event.

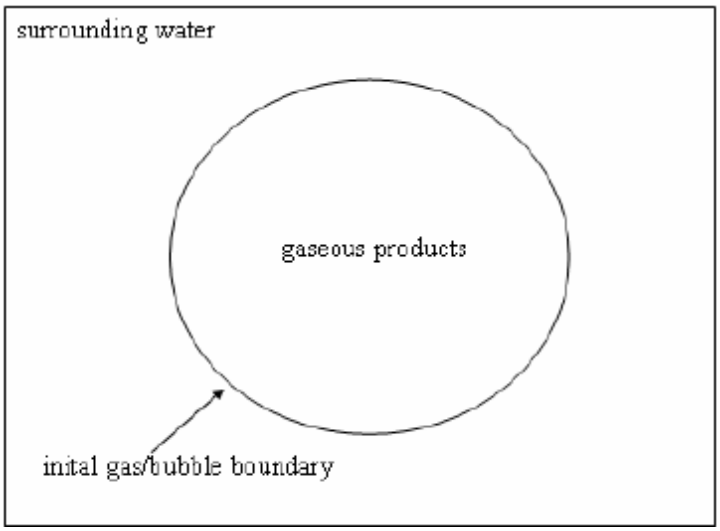

Figure 5: UNDEX Post-detonation instant schematic.[8]

1.2.4 Shock Wave Phenomena During UNDEX

When discussing an UNDEX event, the detonation effects are generally viewed as occurring instantaneously in the sense that the detonation process is assumed to have completed and all explosive material is assumed to have been consumed [7]. The shock wave is a large, nearly discontinuous, compressive pressure wave that is produced by the initial detonation wave reaching the outer surface of the dense gas bubble [7]. The shock wave propagates as a spherical wave that initially travels faster than the speed of sound due to acceleration from the initial explosion. As the wave travels further from the point of initial detonation, the shock wave slows to the speed of sound and its pressure decreases in magnitude. As the distance increases, the shock wave begins to stretch as shown in Figure 6.

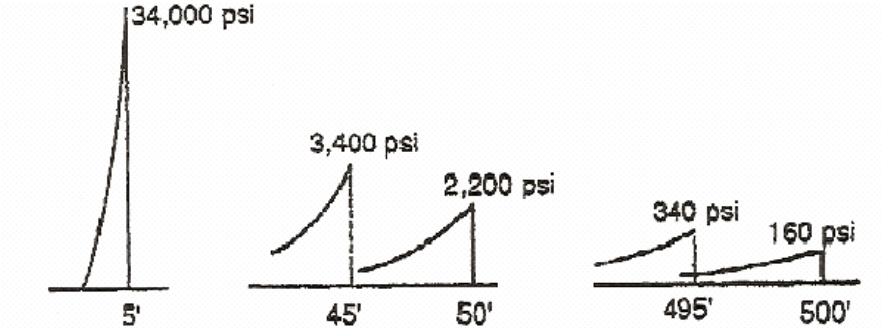

Figure 6: The Shock Wave Profile as a Function of Distance [7]

The shockwave may encounter three physical boundaries as it propagates: the sea floor, the free surface and the fluid-structure interface at the ship/submarine hull. Based on the acoustic

wave reflection properties, the properties of the reflected shock wave vary. If shock wave impacts the free surface, it is reflected as a tensile wave. If the reflection occurs from the sea-floor, some energy gets absorbed and the rest of the energy is reflected as a weaker, compressive wave [6, 7]. In case of an air-backed steel hull, the impinging shockwave undergoes an almost perfect reflection.

When the shockwave hits the hull, the structure is accelerated causing it to move in an inward component direction [6]. When the structure moves faster than the surrounding fluid, this may expose the water/fluid to tension which it cannot sustain and leads to the occurrence of cavitation at the fluid-structure interface. This is called "local" or "hull" cavitation where the hull is not in contact with water. The hull reaches its maximum velocity, called the "kick-off" velocity [9], at about the same time as the onset of local cavitation. This thesis attempts to capture this kick-off velocity by the method of Taylor flat plate theory discussed in Section 2.2. Once the kick-off velocity is reached, the velocity starts to decay and reverse its direction (rebound). Due to the outward motion of the hull and the influence of hydrostatic pressure, the cavitation region closes, causing the water to impact the hull resulting in a re-loading (which is not large as the initial pulse) [10].

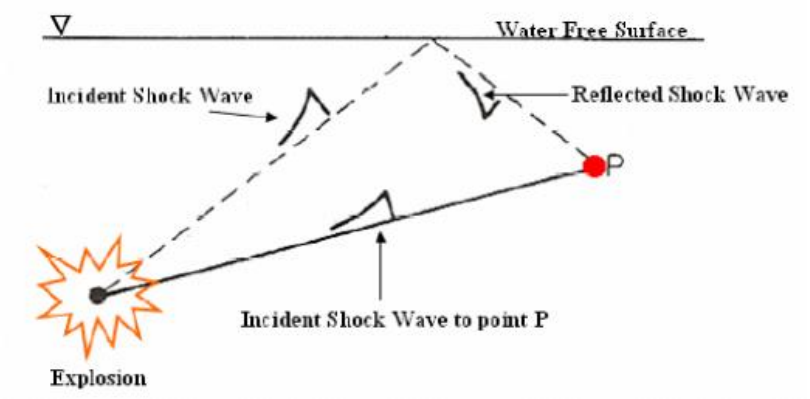

Figure 7: Surface cavitation occurrence at Point P [11]

The third physical boundary encountered by the shock wave is the free surface where it is reflected back into the water as a tensile wave with a magnitude almost exactly the same as the incident compressive shock wave on free surface [6]. The arrival of shockwave at the free surface creates an upheaval of white water, a "spray dome", due to the vertical velocity as the shockwave is reflected. An expanding ring of darkened water, called the "slick", is also created as the ripples

are smoothed by the reflected shockwave pressure resulting in a cavitated water layer that lasts only for a few milliseconds.

When the incident and the reflected shockwaves (from the free surface) meet, the pressures cancel each other allowing the net pressure to go below the vapor pressure of liquid water (See Figure 7). This leads to the creation of a large bulk cavitation region near the surface (See Figure 8) which is symmetric about the vertical axis of the explosion origin and is bounded by an upper and lower boundary [6]. As the pressure wave decays, the reflected wave is no longer large enough to reduce the absolute pressure to vapor pressure. This causes the bulk cavitation region to close like a zipper, causing the water above the upper cavitation boundary to crash down on the water below the lower boundary and thus radiate a cavitation pulse from the region. This pulse can be very destructive depending on the proximity of the hull to the cavitation region [6].

The consequent events involved in the UNDEX phenomena include bubble pulse, bubble migration and associated events like jetting, whipping of the hull etc. Since the focus of this thesis is on the propagating shock wave and its fluid/structure interaction with a ship, these associated phenomena are not discussed further.

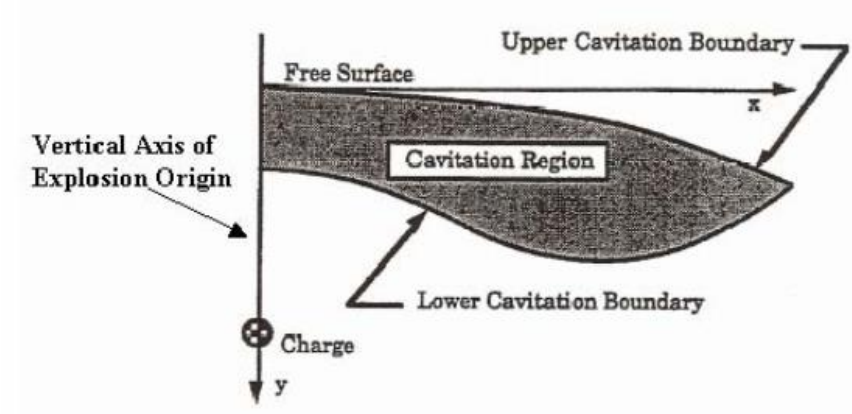

Figure 8: A typical bulk cavitation region.

1.2.5 UNDEX Similitude Equations and Peak Pressure Approximation

The UNDEX similitude equations present a simplified way to calculate the characteristics of an underwater explosion as the analytical models are very complex. These formulations also permit smaller domains for numerical models thus saving computational time and avoiding long range numerical damping issues. The UNDEX similitude equations are parametric equations that

use experimentally determined constants and relate charge size and location to properties like pressure, impulse and energy associated with each of the events in UNDEX phenomena without actually having to model the explosion.

The primary variables in the equations are charge weight and the distance of the point of interest from the charge. The "Principle of Similarity" described by Cole [7] stipulates that if the primary variables of two different explosions acting at different locations are scaled by a constant ratio then the final characteristics evaluated for the two explosions are also scaled by the same ratio. It has to be remembered that similitude equations do not model the physics directly, but provide a good understanding of the UNDEX event behavior.

In this thesis, the far-field early-time phenomenon associated with a shockwave is of interest and hence the focus is on the similitude equations that estimate the properties of shockwave. The pressure time history due to a shock wave is given by the "Peak Pressure approximation", derived in Section 2.1. It can be written as:

$$p(t) = P_{max} e^{-\frac{t}{\theta}} \tag{1.2}$$

The maximum pressure is formulated based on similitude as [7, 12]:

$$P_{max} = K \left(\frac{W^{\frac{1}{3}}}{R} \right)^{\alpha} \tag{1.3}$$

Where R is the distance of the point of interest to the center of the charge and W is the charge weight. The decay constant, θ is defined as the time taken for the pressure to fall to 36.8% of the peak pressure [12] and is also given by similitude as:

$$\theta = W^{1/3} K \left(\frac{W^{\frac{1}{3}}}{R} \right)^{\alpha} \tag{1.4}$$

The constants, K and α are experimentally derived and are specific for a given metric system of units. See Table 1 for the values of constants in SI units incorporated in the thesis. The impulse, I, and energy flux, E, of the shock wave are also given by same form of similitude relations as the decay constant with the values shown in Table 1.

Equation	K	α
Peak Pressure	52.4	1.13
Decay Constant	0.084	-0.23
Impulse	5.75	0.89
Energy Flux Density	84.4	2.04

Table 1: Peak Wave Approximation Constants in SI units. Adapted from [8, 12]

These equations are limited to a range of the primary variables and are restricted in terms of the assumptions used. The key assumptions which are important in this thesis are that there are no boundary surface effects and that it is a "free-water" explosion [12]. Other assumptions include no bubble migration, irrotational fluid and others as listed in Table 2.

Assumption	Impact
No boundary surface effects	No reflection of shock wave or bubble pulses. No cavitation or free surface effects. All similitude equations based on explosions that took place significantly far away from boundary surfaces.
Free-water (infinite fluid) explosion	Bubble reaches its maximum before it reaches the surface.
No gas bubble migration	Shape of bubble pulse unaffected by moving bubble. All similitude equations and constants based on explosions that resulted in minimal bubble migration.
Spherical gas bubble	Allows bubble radius to be found using similitude. Affects volume of bubble used in energy equations
Incompressible fluid/no energy losses in bubble oscillation	Creates error in hydrodynamic bubble energy calculations.
Irrotational fluid	Allows potential flow to be used to develop energy equations.
Radius-time curve symmetrical about bubble maximum time	Ignores any energy lost between bubble minimums.

Table 2: Assumptions used in Similitude Equations. Adapted from [13]

1.3 Response of Ships to Underwater Explosions

The focus of this thesis is the early time effects of far-field explosions. The damage to the hull plating due to this far-field near-time phenomenon is primarily elastic or elastic-plastic deformation [11]. In other scenarios, such as far-field late time or near field explosions, the damage to structure may be more significant and involves phenomenon like hull whipping due to bubble

migration, rupture and plastic deformation such as dishing of shell plating which are not of interest in this thesis.

The severity of underwater explosions due to shockwaves has been widely measured using the shock factor value [6, 11] which is based on the energy density of the shockwave arriving at the hull. The shock factor is a function of the explosive charge weight (W), stand-off distance (R) and the relative attack geometry (determined by the angle of incidence of shock wave, α). The shock velocities aboard a ship can be given by the general formulation [11]:

$$Shock\ Factor, SF = C\frac{W^{\frac{1}{2}}}{R}f(\alpha) \tag{1.5}$$

Where: C =C(ship-type, location aboard the ship, type of installation); and

$f(\alpha) =$ Angle function of the form, $[\eta + (1-\eta)\cos(\alpha)]$ where η is determined from experiments.

This formulation allows us to define the shock velocities aboard a ship as:

$$V = SF \times \overline{V}(t) \tag{1.6}$$

Here $\overline{V}(t)$ is called the 'Shock signature' showing typical variation of velocity in time. There are normalized velocity histories for each typical case such as vertical or horizontal shock velocities. This thesis may enable the development of a parametric equation for Shock Factor using a Design of Experiments for shock at different locations in the ship and a Response Surface Methodology to calculate C as a function of (x, y, z).

A typical profile of vertical velocity in the ship due to shock looks like Figure 9. The peak velocity is called the kick-off velocity and the time taken to reach the kick-off velocity after the shock reaches the hull is called the rise time to maximum velocity. Similarly, there is rise time for maximum initial displacement. The shock severity at different locations is shown for a typical ship in Figure 10.

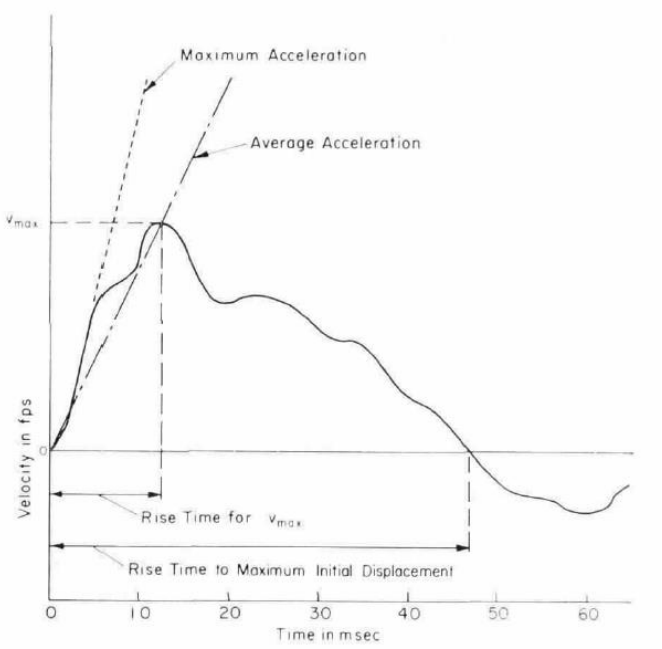

Figure 9: Typical profile for vertical velocity of hull due to shock. [11]

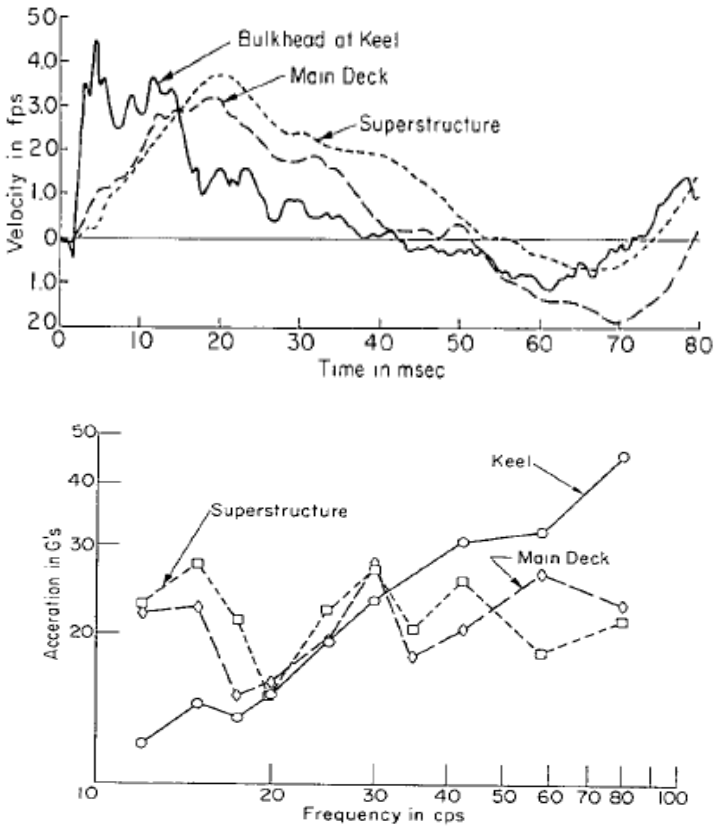

Figure 10: Shock severity at different locations for a typical ship. [11]

In case of UNDEX far-field early time phenomena, although the hull deformations might not be significant, the effect of the velocities and accelerations on the hull plating, foundations, personnel and equipment can be extreme. Shock can cause damage to equipment in terms of machinery failure, brittle fracture of foundations, damage to hold-down bolts, failure of connections between installation, ejection of insufficiently secured equipment and more [11]. Items to be tested are classified in accordance with one of the following grades, as specified [14]:

1. Grade A items are items which are essential to the safety and continued combat capability of the ship.
2. Grade B items are items whose operation is not essential to the safety and combat capability of the ship but which could become a hazard to personnel, to grade A items, or to the ship as a whole as a result of exposure to shock.

Analysis of effects of far-field explosion is crucial as it can cause failure of a ship in terms of its functionality or mission capability. The response to far-field explosions are measured in terms of the shock factor based on the peak velocity at different locations and also the shock response spectra at different locations for a given incident shockwave.

1.4 Early-Stage Ship Design and UNDEX Shock Vulnerability Context

When designing a ship, it is important to take into account the ship's ability to survive UNDEX phenomenon if it is likely that the ship could be exposed to them. Currently, during the early-stage design of a naval ship, survivability is often treated as a secondary concern when compared to meeting mission requirements and other design criteria. This means that the ship structure and system arrangements are not optimized for survivability. Survivability with respect to UNDEX phenomenon is primarily deferred until later in the design process due to the level of detail available and difficulties associated with modeling the fluid/structure interaction.

At Virginia Tech, the Concept and Requirements Exploration (C&RE) process shown in Figure 11 is used for early-stage (exploratory) ship design. A primary goal of this process is to understand the relationship between cost, effectiveness, and risk for the proposed technologies in balanced ship designs. Steps in this process where UNDEX shock vulnerability should be considered are outlined in red.

The C&RE process considers a large design space with a range of inputs for hull-form characteristics, propulsion system options, and combat system options, performs a series of system and subsystem explorations, develops metrics, assembles a synthesis model from surrogates and

data generated in these explorations, and searches the design space for non-dominated designs using a multi-objective genetic optimization[15].

The first step in the C&RE process is to expand the mission description from the Initial Capabilities Document. This mission description is used to develop Measures of Effectiveness (MOEs), Measures of Performance (MOPs) and Operational Effectiveness Models (OEMs). Ship vulnerability is an important component of OEMs.

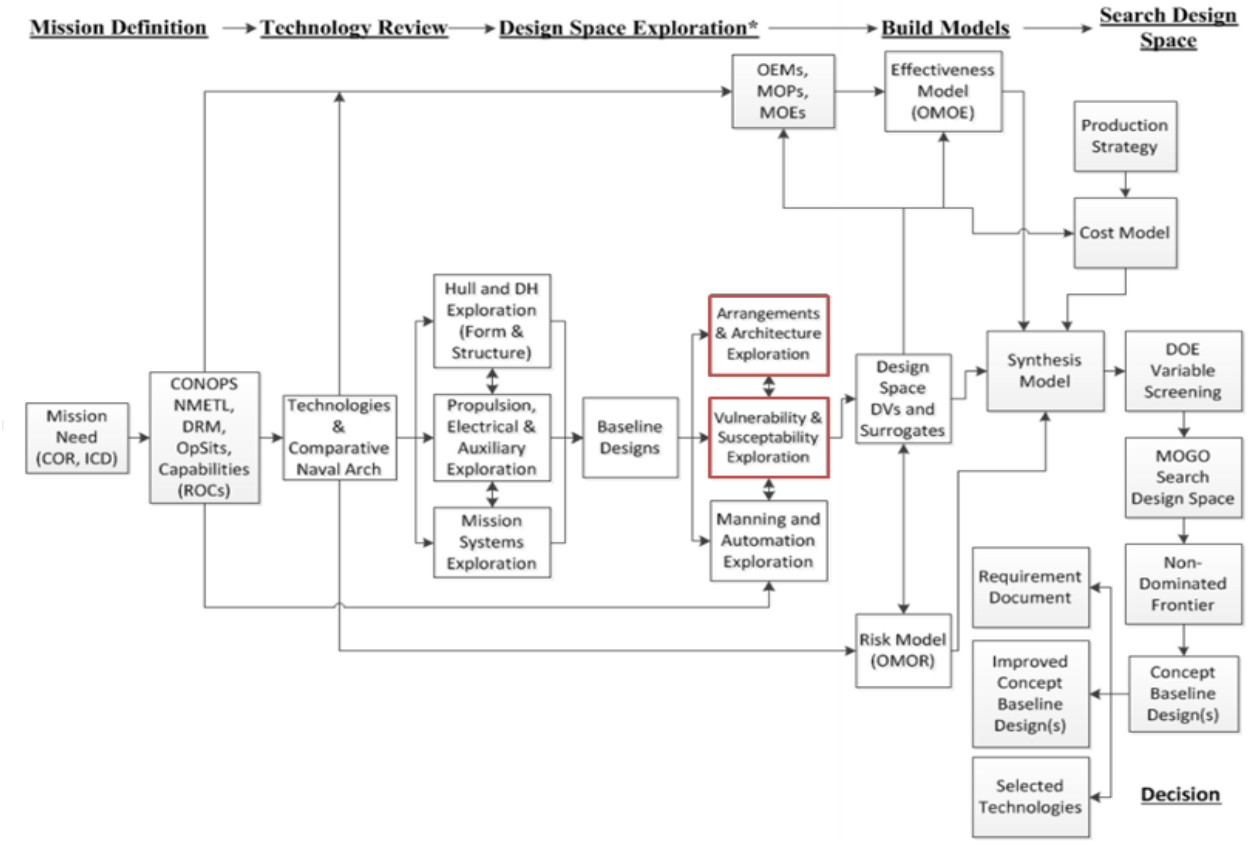

Figure 11: Virginia Tech Concept and Requirements Exploration Module (C&RE)

The C&RE process continues with a collection and review of applicable technologies and a comparative naval architecture study of existing ships with similar missions. These studies begin to define the design space and a preliminary Baseline Design. Next the process expands into concept explorations of six critical categories: hull-form and deckhouse geometry, structures, machinery and propulsion, mission systems, arrangements, manning and automation, and survivability. These explorations result in the collection and analysis of data for each discipline using a design of experiment (DOE) approach, identification of key design variables and parameters, definition and refinement of the design space for each area, and response surface

models (RSMs) that approximate the relationships between input design variables and the response characteristics for use in a design-specific synthesis model. Because generic parametric equations and a generic synthesis model based on limited data from past ship designs are not sufficiently applicable or flexible for thinking outside the box in new designs, a more physics-based, design-specific approach is required. Once these individual explorations are a complete, a set-based integration approach may be used to define an integrated design space for ship synthesis.

Next, an initial set of Design Variables (DV's) and Design Parameters (DP's) are selected and examined including the generation of a baseline design complete with hull-form, propulsion and power, and mission systems. A Design Parameter (DP) is a ship characteristic that is assigned a fixed (constant) value that is shared across all potential ships in the design space. The list of DP's and DV's are used as the input to a ship synthesis model (SSM) to generate complete designs and assess their feasibility. The SSM is then used in a Multi-Objective Genetic Optimization to search the design space based on their associated risk, cost, and effectiveness and to identify a set of non-dominated designs. Examination of the Non-Dominated Frontier enables the selection of preferred designs and tradeoffs between cost, risk, and effectiveness can be considered. Often preferred selections are at "knees in the curve", designs at the top of steep effectiveness slopes.

Vulnerability is the ships ability to withstand damage and is heavily dependent on ship arrangements, vital component (VC) locations, and system architecture. Vital Components or "VC's" are ship equipment vital to ship system capability. A Vital Component can be a ship service generator, switchboard, sprinkler, VLS launcher, or any other equipment that exists within a deactivation diagram for ship systems considered in the Vulnerability Model. Vital Components are assigned to compartments and in turn, the compartments are assigned to subdivision block (SDB) locations. VC's are elements of systems and subsystems with their architecture defined initially using one line diagrams and deactivation diagrams. A critical area of VC vulnerability is UNDEX shock effects which are dependent on the UNDEX location relative to the ship, charge size, the ship structural design and VC locations within the ship. Ultimately, an UNDEX can cause VC failure, system failure, loss of ship capability and reduction in ship effectiveness.

The Preliminary Arrangements and Vulnerability (PA&V) process, *Figure* 12, used in the C&RE considers all combinations of mechanical and electrical and mission system options with their associated weight, volume, and power requirements, system option architectures, VC lists, deactivation diagrams and preliminary hull-form to define and assess vulnerability for preliminary

arrangements of compartments and VCs. The Vulnerability Assessment step in this process will apply UNDEX threats and calculate UNDEX shock effects and resulting shock factors at VC subdivision block locations in the ship to determine the failure of Vital Components due to shock and ultimately an UNDEX probability of kill given hit, $P_{k/h}$. This assessment will be performed for all system combinations, for each threat, and for each of multiple compartment arrangements.

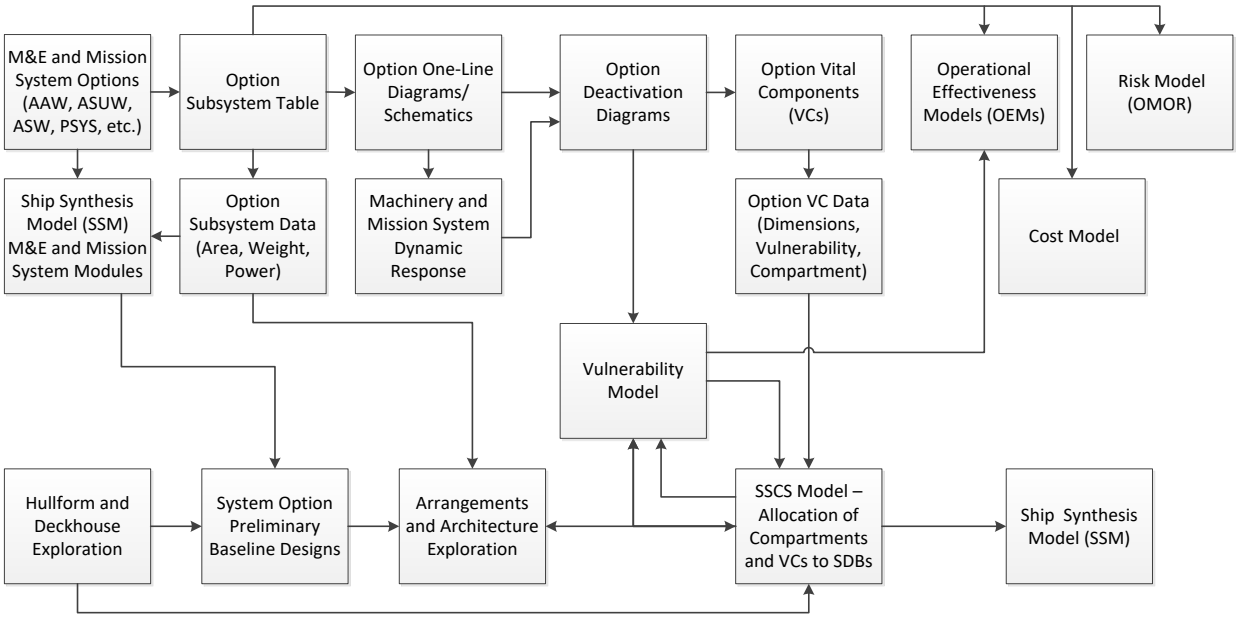

Figure 12: Preliminary Arrangements and Vulnerability Process

The calculation of UNDEX shock effects and shock factors requires a hydrodynamic model of the UNDEX event and a finite element model of the target ship. This involves a significant design effort for this very early stage of design. It is expected that UNDEX shock effects and shock factors will be calculated for a small series of representative baseline structural designs and shock factor response surface models (RSMs) will be developed and applied to the full design space. This approach is similar to that proposed by Keil [11], but uses structural analyses and a design of experiments (DOE)/RSM approach to derive shock factor parametric equations. This thesis describes tool development for this approach.

1.5 Literature Survey

Underwater explosion is a phenomenon with a lot of physics going on in the scale of milliseconds and microseconds. It grew to be more relevant during the time of World War-II i.e., the 1940s and beyond, when not just contact explosions, but even non-contact explosions activated by proximity fuses managed to immobilize full-size ships. Hence, the design of ships against

vulnerability to underwater explosions needed to be more effective and affordable to avoid the large cost and damage possible during shock trials. This is proposed to be accomplished using analytical and numerical models so that their results can be used in the early-stage design of ships.

1.5.1 UNDEX Fluid-Structure Interaction Models

As the initial underwater explosion phenomenon occurs in the fluid region, a primary challenge has been to correctly model the interaction of the fluid on the structure over time. Much research has addressed fluid models that can transfer loads to the structure. Early models used the similitude equations discussed in the works of Cole [7] and Arons [16] and summarized in Section 1.2.5.

1.5.1.1 Analytical Model – Taylor Flat Plate equation

The Taylor plate theory introduced by Taylor[17] and the formulation for the total pressure on a rigid plate due to shockwave impact is discussed well for both air-backed and water-backed plates in Scavuzzo [18].

Fleck and Deshpande[19] investigate the one-dimensional shock response of sandwich plates for the case of identical face sheets separated by a compressible foam core. They use the Taylor plate theory to compute the momentum transmitted to the sandwich plate by treating the outer face of the sandwich plate as a free-standing plate without the support of the core and uses finite element as well as lumped parameter models to explore the accuracy of this assumption. Fleck and Deshpande[20] also analyzes the blast resistance of clamped sandwich beams in different stages and the one-dimensional fluid-structure interaction problem during the blast loading event is solved using the Taylor Flat plate theory. The influence of structural support to the face sheet has been modeled by adding a restraining force term to the analytical equation.

Z. Liu and Y.L. Young[21] extends the Taylor Flat Plate theory for air-backed plates to the case of submerged water-backed plates and provides the formulation for a FSI (fluid structure interaction) parameter applicable for the water-backed plates assuming rigid plates and 1D system behavior. The paper also defines a momentum transmission coefficient as a quantitative measure of momentum transfer (in percent) during shock impact in case of both water-backed and air-backed plates. It also performs a parametric study of the momentum transmission coefficient, peak response time and cavitation inception time against the FSI parameter.

Kambouchev, Radovitzky and Noels[22] uses the FSI parameter given by the Taylor plate theory and extends it to compressible range and uses it to describe the relative impulse

transmitted to the plate. The asymptotic limits of very heavy and very light free standing plates are derived exactly for arbitrary shock intensities. In [23],Kambouchev, Radovitzky and Noels extends the Taylor Plate theory by incorporating the non-linear compressibility effects in the fluid so that it can be used as fluid-structure interaction model for air-blast loading of structures. The non-linear compressibility incorporation shows higher reductions in transmitted impulse compared to ignoring FSI in the acoustic limits and these results help with better blast resistant design.

The fact that it is possible to Taylor plate theory can be extended for non-rigid plates based on the above papers gave the inspiration that it can be used to obtain the initial kick-off velocities and the rise-time at different parts of the ship.

1.5.1.2 Boundary element method (BEM)

This method solves the linear partial differential equations of a domain by formulating them as (boundary) integral equations. Using the boundary conditions, the integral equation is solved and then used to calculate the solution in the interior of the domain.

The most common BEM used for the UNDEX problem is the Doubly Asymptotic Approximation (DAA) [24] which has the advantage that the governing equations are modeled using the wetted surface variables only. This method has the advantage of not having to model the actual fluid at all which eliminates the difficulty of capturing the shock wave accurately and also allows for faster computation [8, 25, 26]. The DAA method is comprised of an early-time approximation and a late-time approximation. The initial model DAA (DAA_1) was not suited for the intermediate frequency range so an improved model (DAA_2) was introduced by Geers in 1978 [27] to overcome this limitation.

The DAA method is incorporated in the USA (Underwater Shock Analysis) code [25]. A far-field early-time underwater explosion case was solved by Fox [28] for a submerged cylinder model using the USA code with a good match to experimental results. However, for capturing cavitation, the fluid region around the structure must be modeled and coupled with the structure [8, 9], so the USA code was later integrated with LS-DYNA. In this case, the LS-DYNA finite element code is used to model the part of the fluid domain which undergoes cavitation and also, as the structural solver. The USA code is used to model the fluid domain as a radiation boundary. Wood [9] and Beiter [29] modeled ship-like box structures subjected to UNDEX shockwaves using the above coupling and concluded that cavitation had a significant effect on structural response and could not ignored for obtaining accurate results.

1.5.1.3 Hydrocodes

Hydrocodes are computational continuum mechanics tools that simulate responses in both solids and fluids to wave propagation (typically shock waves) [24, 30]. Hydrocodes are general-purpose methods and models to solve the general form of the fundamental equations of continuum mechanics [31]. These codes make fewer simplifying assumptions than classical CFD (Computational Fluid Dynamics) or CSM (Computational Structural Mechanics) codes [31]. The major hydrocode types include: Lagrangian, Eulerian, Coupled Eulerian-Lagrangian (CEL), and Arbitrary Lagrangian Eulerian (ALE) [24]. Hydrocodes have the capability to calculate fluid and structural response simultaneously; however, they demand high computational resources.

1.5.1.4 Computational Fluid Dynamics

Computational Fluid Dynamics (CFD) is a branch of fluid mechanics that creates quantitative predictions of the fluid-flow phenomena based on conservation laws governing fluid motion and are defined in terms of the flow geometry, the physical properties of the fluid, and boundary and initial conditions of the flow field [32]. CFD codes can be used to obtain fluid responses coupled to structural solvers [24]. In the case of UNDEX, CFD has been useful in capturing the bubble motion and time histories accurately [33]. However, CFD codes demand significant computational resources and time to perform analyses based on domain refinement and accuracy of results.

1.5.1.5 Cavitation Acoustic Finite Element (CAFÉ) and Cavitation Acoustic Spectral Element (CASE) method

Shock waves are usually modeled assuming the fluid medium is of an acoustic nature i.e. the disturbances in the fluid propagate rapidly and are geometrically linear, even if the constitutive behavior of the fluid is non-linear [34]. This assumption is particularly valid for the far-field problem as shock waves originating far from the structure cause relatively small structural and fluid displacements, particularly as they approach the structure [35]. Fluid-structure interaction modeling for underwater explosions also requires modeling cavitation effects which adds additional complexity.

The governing equation of an acoustic fluid can be modeled in a vector formulation or a scalar formulation. However, the large size of the structure and fluid model for ship shock analysis requires using the scalar formulation to be computationally efficient as it reduces the number of unknowns at a node from three to one [34].

The Cavitating Acoustic Finite Elements (CAFE) approach, developed by Sprague and Geers, was evolved from Newton's cavitation model, which uses the displacement potential formulation of the acoustic wave equation. LS-DYNA, a commercial finite element code, has incorporated this CAFE approach. Shin [36] has done simulations of ship shock trials using LS-DYNA/USA (Underwater Shock Analysis) code. Some deficiencies of the CAFE approach include:

1. Fluid meshes can be highly dispersive due to the use of linear elements.
2. Requires significant fluid mesh refinement to adequately capture cavitation extents

To overcome the deficiencies, Sprague and Geers used higher order Spectral Elements (SE) in what they called Cavitating Acoustic Spectral Elements (CASE) [37, 38]. Spectral elements use high order basis functions derived from orthogonal polynomials to reduce the interpolation error at the element nodes. This feature makes it less dispersive and improves the super-refinement issues. Sprague and Geers also incorporated a non-conformal coupling algorithm for fluid-structure interaction modeling and a scattered field model (described in Section 2.3.1). CASE thus addresses deficiencies of the CAFE approach although it has its own limitations:

1. Smaller critical time-steps i.e. more computationally expensive in time.
2. Tendency to increase spurious oscillations [37, 39].

This latter deficiency is important because of steep gradients/discontinuities present in far-field UNDEX problems. Newton referred to this phenomenon during cavitation as "Frothing" where isolated regions of cavitated fluid are observed within the pressurized fluid [40]. Artificial damping was introduced to reduce frothing in the fluid although it added non-physical energy loss and smeared the wave front and cavitation boundary. Artificial damping is found to be a more valid approach in the case of shock and cavitation loading as the effects of damping are mitigated due to presence of the impulse of propagating waves that drive the structural response.

1.5.2 Ship Structural Solver for UNDEX

To obtain the response of a hull to an underwater explosion, we need to choose a structural solver to model the geometry, discretize and perform the relevant static or dynamic analysis. Finite Element Analysis (FEA) is a standard method used for these solvers and there are various commercial and academic finite element codes available. However, there are not many codes developed specifically for ship structural analysis. The accuracy of the runs depends on the mesh sizing and also the time step in the case of transient or dynamic analysis. As discussed before, to

model the effect of UNDEX on ship structures, one needs to couple the FEA code with the UNDEX fluid model by means of a fluid-structure interface (FSI) code for accurate analysis.

1.5.2.1 MAESTRO Structural design software

MAESTRO (Method for Analysis, Evaluation, and STRuctural Optimization) is a design, analysis and evaluation tool specifically developed for floating structures [41]. The main components of MAESTRO are: rapid coarse-mesh finite element modeling, ship-based loading, finite element analysis, limit state and design evaluation [41]. MAESTRO also has a well-developed module for optimization of plate panels as part of limit state and design evaluation, and an excellent tool for exporting a Nastran model based on the MAESTRO model including static and tank loads.

1.5.2.2 NASTRAN

MSC Nastran is a general purpose finite element analysis computer program [42]. Written primarily in FORTRAN, it can address a wide range of engineering problems, both static and dynamic, whether linear or non-linear. It also enables coupling with CFD solvers and in-house codes for FSI problem implementation with a unique OpenFSI interface. NASTRAN is a structural solver and requires a preprocessor and a post-processor to create the model and visualize/interpret the results. FEMAP and MSC Patran may be used for the pre- and post-processing. MAESTRO, can be used in pre-processing in conjunction with FEMAP or MSC PATRAN. Further discussion of these tools is made in Chapter 3 and Chapter 4 regarding the analysis process used.

1.6 Research Objectives

This thesis describes the development of and assesses a simplified tool for modeling underwater explosion shock effects during early naval ship concept design. The primary objective of this research is to develop a computationally efficient and sufficient design tool which captures the peak velocity and shock spectra characteristics for ship vulnerability assessment in underwater explosions. Shock damage to equipment and personnel injury are the primary weapon effects in far-field underwater explosions and this tool strives to capture these response characteristics by means of a simplified fluid structure interaction model created in a Nastran-OpenFSI code. A second objective is to assess the performance of this very simple model to determine if results are sufficient for early-stage ship design by comparison to benchmark results and more capable CASE model results in an UNDEX applied to a floating shock platform (FSP).

1.7 Thesis Outline

Chapter 1 introduces underwater explosion phenomena and the concepts involved in survivability design. A literature review is also provided on the various UNDEX/shock simulation methodologies researched and validated in the past.

Chapter 2 develops the theory of a plane shock wave followed by a description of Taylor flat plate theory, used for our simplified fluid-structure interaction model. Theory for the more comprehensive Cavitating Acoustic Spectral Element (CASE) simulation which is used for comparison to our OpenFSI Taylor Plate model is also described. Shock spectra concepts are discussed with applications/design parameters, especially in case of underwater explosions.

Chapter 3 describes the development and framework of our OpenFSI Taylor Plate model. A benchmark analytical Bleich-Sandler plate model is used to validate our model. A shell-structure similar to the wetted surface of the floating shock platform (FSP) is also analyzed and OpenFSI Taylor Plate results are compared to CASE results.

Chapter 4 describes the Floating Shock Platform (FSP) structural details and model set-up. OpenFSI Taylor Plate results are compared to FSP CASE results obtained by Zhaokuan Lu [43] to assess the sufficiency of the OpenFSI Taylor Plate method.

Chapter 5 provides concluding remarks and proposed future work which includes integration of the OpenFSI Taylor Plate C++ code with a ship synthesis code in Model Center for application in a Design of Experiments to assess a real design application and model sensitivity.

2 THEORETICAL BACKGROUND

In this chapter, the theory of a plane shock wave is developed, followed by a description of Taylor flat plate theory used for our simplified fluid-structure interaction model. Theory for the more comprehensive Cavitating Acoustic Spectral Element (CASE) simulation which is used for comparison to our OpenFSI Taylor Plate model is also described. Shock spectra concepts are discussed with applications/design parameters, especially in case of underwater explosions.

2.1 Derivation of 1D Planar Shock Wave as it arrives at the Structure

As discussed in Section 1.2.4, the shock wave is one of the key components of the Underwater Explosion Phenomena. An UNDEX shock wave has a nearly discontinuous rise in pressure followed by a brief period of exponential decay, on the order of few milliseconds. The shock wave propagates in water as a spherical wave, initially traveling somewhat faster than the speed of sound, but upon arriving at far-field location has a nearly planar wave form as seen by a single shell structure element. As it moves further away from the explosion origin, the peak pressure reduces and the shock wave speed also falls to the speed of sound in water. The shock wave can be assumed to be in the form of plane wave for the early-time far-field case as long as it's basic characteristics are estimated based on the actual spherical wave. This is accomplished using the similitude equations [7].

The equation of a 1-dimensional shock wave can be derived using the perturbation wave equation in terms of the velocity potential,ϕ assuming irrotational flow [44].

$$\nabla^2\phi = \frac{1}{c^2}\frac{\partial^2\phi}{\partial t^2} \tag{2.1}$$

Where, c is defined as a constant which turns out to be the speed of sound in water. The speed of sound in an acoustic fluid is given as:

$$c = \sqrt{\left(\frac{\partial p'}{\partial \rho'}\right)_T} \tag{2.2}$$

Here p is the fluid pressure and ρ is the fluid density.

The wave perturbation equation in 1-D can be written as (here x is the distance coordinate and t is time):

$$\frac{\partial^2\phi}{\partial x^2} = \frac{1}{c^2}\frac{\partial^2\phi}{\partial t^2} \tag{2.3}$$

The general solution of the above equation is of the form,

$$\phi(x,t) = \phi(x + mt) \tag{2.4}$$

Taking temporal and spatial derivatives of velocity potential, we have:

$$\frac{\partial^2 \phi}{\partial x^2} = \phi''(x + mt) \tag{2.5}$$

And

$$\frac{\partial^2 \phi}{\partial t^2} = m^2\, \phi''(x + mt) \tag{2.6}$$

Substituting the above expressions in (2.3), we get:

$$1 - \frac{m^2}{c^2} = 0 \tag{2.7}$$

Where:

$$m = \pm c \tag{2.8}$$

And:

$$\phi = f(x + ct) + g(x - ct) \tag{2.9}$$

Here $f(x + ct)$ corresponds to a left traveling wave and $g(x + ct)$ to a right traveling wave.

Then D'Alembert's method provides a solution to the initial value problem. For the initial condition, we can assume that there is a given $u(x, 0)$ and $p(x, 0)$.

The velocity potential definition then gives:

$$\left.\frac{\partial \phi}{\partial x}\right|_{t=0} = u_x(x, 0) \tag{2.10}$$

Integrating, we get,

$$\phi(x, 0) = \int u_x(x, 0)\, dx = F(x) \tag{2.11}$$

Based on the general form derived in Eqn. 2.9, and putting t = 0,

$$\phi(x, 0) = f(x) + g(x) = F(x) \tag{2.12}$$

Using the momentum conservation equation, we have:

$$\frac{\partial \phi}{\partial t} + \frac{(\nabla \phi)^2}{2} + \frac{p}{\rho_0} = 0 \tag{2.13}$$

Substituting for t = 0:

$$\left.\frac{\partial \phi}{\partial t}\right|_{t=0} = \frac{-u_x(x,0)^2}{2} - \frac{p(x,0)}{\rho_0} \approx -\frac{p(x,0)}{\rho_0} = G(x) \tag{2.14}$$

Also from the general form of velocity potential at t = 0, we have:

$$\left.\frac{\partial \phi}{\partial t}\right|_{t=0} = c[f'(x) - g'(x)] = G(x) \tag{2.15}$$

Re-arranging:

$$f'(x) = g'(x) + \frac{1}{c}G(x) \tag{2.16}$$

Integrating:

$$f(x) = g(x) + \frac{1}{c}\int_0^x G(\xi)d\xi = F(x) - f(x) + \frac{1}{c}\int_0^x G(\xi)d\xi \tag{2.17}$$

Re-arranging:

$$f(x) = \frac{1}{2}F(x) + \frac{1}{2c}\int_0^x G(\xi)d\xi \tag{2.18}$$

Similarly, we have:

$$g'(x) = f'(x) - \frac{1}{c}G(x) \tag{2.19}$$

Integrating:

$$g(x) = f(x) - \frac{1}{\mathrm{c}}\int_0^{\mathrm{x}} \mathrm{G}(\xi)\mathrm{d}\xi = \mathrm{F(x)} - g(x) - \frac{1}{\mathrm{c}}\int_0^{\mathrm{x}} \mathrm{G}(\xi)\mathrm{d}\xi \tag{2.20}$$

Rearranging:

$$g(x) = \frac{1}{2}F(x) - \frac{1}{2c}\int_0^x G(\xi)d\xi = \frac{1}{2}F(x) + \frac{1}{2c}\int_x^0 G(\xi)d\xi \tag{2.21}$$

Substituting for $f(x)$ and $g(x)$ in Eqn. 2.9:

$$\begin{aligned}\phi(x,t) &= f(x+ct) + g(x-ct) \\ &= \frac{1}{2}[F(x+ct) + F(x-ct)] + \frac{1}{2c}\int_{x-ct}^{x+ct} G(\xi)d\xi\end{aligned} \tag{2.22}$$

Separating the velocity potential into negative-direction and positive-direction propagating wave portions of the velocity potential, we have:

$$\phi(x,t) = \frac{1}{2}\left[F(x+ct) + \frac{1}{c}\int_0^{x+ct} G(\xi)d\xi\right] + \frac{1}{2}\left[F(x-ct) + \frac{1}{c}\int_{x-ct}^{0} G(\xi)d\xi\right] \tag{2.23}$$

So, if we consider a 1-D, arbitrary, positive propagating shock wave, the velocity potential would be:

$$\phi(x,t) = \frac{1}{2}\left[F(x-ct) + \frac{1}{c}\int_{x-ct}^{0} G(\xi)d\xi\right] \tag{2.24}$$

If we assume an initial disturbance of the following form:

$$p(x \geq a, 0) = 0 \tag{2.25}$$

$$p(x \leq a, 0) = P_0 e^{b(x-a)} \tag{2.26}$$

$$u_x(x,0) = 0 \tag{2.27}$$

Then we have from Eqn. 2.11 :

$$F(x) = \int u_x(x,0)\, dx = 0 \tag{2.28}$$

Also, from Eqn. 2.15

$$G(x) = \frac{-u_x(x,0)^2}{2} - \frac{p(x,0)}{\rho_0} = -\frac{p(x,0)}{\rho_0} \tag{2.29}$$

Substituting these into 2.24,

$$\phi(x,t) = \frac{1}{2}\left[F(x-ct) + \frac{1}{c}\int_{x-ct}^{0} G(\xi)d\xi\right] = -\frac{1}{2c}\int_0^{x-ct} G(\xi)d\xi \tag{2.30}$$

For $\xi \leq a$:

$$\phi(x,t) = -\frac{1}{2c}\int_0^{x-ct} \frac{-p(\xi,0)}{\rho_0} = \frac{1}{2c}\int_0^{x-ct} \frac{P_0 e^{b(x-a)}}{\rho_0} \tag{2.31}$$

Expanding the integral,

$$\phi(x,t) = \frac{P_0}{2cb\rho_0}\left[e^{b(x-ct-a)} - e^{-ab}\right] \tag{2.32}$$

The velocity potential is calculated for ξ equal to the upper limit of the integral, as beyond that the initial disturbance is zero, i.e.

$$\xi = x - ct \leq a \tag{2.33}$$

Where:

$$x \leq a + ct \tag{2.34}$$

Hence,

$$\phi(x,t) = \frac{P_0}{2cb\rho_0}\left[e^{b(x-ct-a)} - e^{-ab}\right] \quad for \quad x \leq a + ct \tag{2.35}$$

$$\phi(x,t) = 0 \quad for \; x > a + ct \tag{2.36}$$

Since we are assuming irrotational flow, we can obtain pressure as:

$$p(x,t) = \; -\rho_0 \frac{\partial \phi}{\partial t} \tag{2.37}$$

Thus:

$$p(x,t) = \frac{P_0}{2}\, e^{b(x-ct-a)} \quad for \;\; x \leq a + ct \tag{2.38}$$

$$p(x,t) = 0 \quad for \;\; x > a + ct \tag{2.39}$$

This leads to the standard form for a shock wave, if we substitute values of coefficients as:

$$a = 0; b = \frac{1}{c\theta}; P_{max} = \frac{P_0}{2} \tag{2.40}$$

Thus standard form of a 1-D Planar shock wave can be written as:

$$p(x,t) = P_{max} e^{\frac{x-ct}{c\theta}} \tag{2.41}$$

Here, P_{max} is referred to as the 'Peak Pressure' of the shock wave and θ is referred to as the 'Decay Constant'. The formulation of these constant values can be obtained for different types of explosives. The values are also dependent on location and weight of the explosive charge. This was discussed in Section 1.2.5.

2.2 Taylor Flat Plate Equation and Application.

Taylor Flat Plate theory [17] is a simplified fluid-structure interaction equation which does not explicitly model the fluid. It is used to determine the dynamic response of hull plating to an initial shock wave caused by an underwater explosion.

In this model, a shockwave is incident on an infinite, air-backed plate and is reflected which imparts an additional pressure and velocity to the plate. The theory develops a relationship between the plate velocity and the incident pressure based on Newton's second law of motion. It assumes that the magnitude of the reflected pressure is equal to the incident pressure. This is a good conservative approximation for an air-backed hull plate as it has been determined that 92% of the pressure is reflected for steel plates with normal incidence [18].

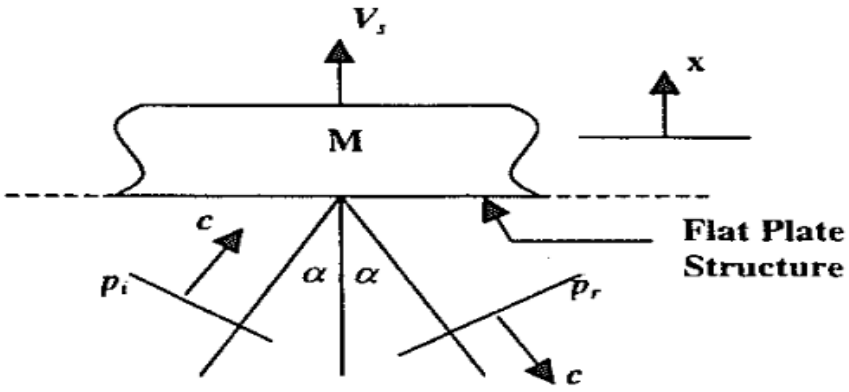

Figure 13: Taylor Flat Plate Fluid-Structure Interaction Model [18]

Here,

p_i: $incident\ pressure$

p_r: $reflected\ pressure$

p_t: $pressure\ at\ fluid\ structure\ interface$

α : $Angle\ between\ the\ normal\ of\ wave\ front\ and\ the\ normal\ to\ the\ flat\ surface$

V_i: $Particle\ velocity\ of\ the\ incident\ wave$

V_r: $Particle\ velocity\ of\ the\ reflected\ wave$

V_s: $Plate(Structure)\ velocity$

The interface boundary continuity relation requires that:

$$V_i cos\alpha\ = V_r cos\alpha + V_s \tag{2.42}$$

The particle velocity of the fluid, V_p can be related to the pressure, p, for a planar shock wave as:

$$p\ = \rho c V_p \tag{2.43}$$

Here, the term ρc is the acoustic impedance. Eqn. 2.42 can then be written as:

$$\frac{p_i \cos\alpha}{\rho c} = \frac{p_r \cos\alpha}{\rho c} + V_s \tag{2.44}$$

And rearranged as:

$$p_r = p_i - \frac{\rho c V_s}{cos\alpha} \tag{2.45}$$

The total pressure at the plate boundary is:

$$p_t = p_i + p_r \tag{2.46}$$

Substituting Eqn. 2.45 in the above equation, we have

$$p_t = 2p_i - \frac{\rho c V_s}{cos\alpha} \tag{2.47}$$

The Eqn. *(2.47),* as developed by G. I. Taylor [17, 18] can then be used to calculate the total pressure at the plate-fluid interface based on the explosion wave characteristics, P_{max} and θ, fluid properties, c and ρ, and the structural dynamic response, V_s in case of air-backed plates.

The above formulation with the following assumptions enables an analytical solution:

1. No stiffness i.e. a rigid unconstrained plate
2. No added mass
3. No damping
4. No cavitation in the original formulation
5. No gravity (body forces)
6. Plate is located at x = 0

In case of a water-backed plate, (e.g. A propeller blade), the water on the backside of the plate provide additional fluid impedance and a pressure equal to $\frac{\rho c V_s}{cos\alpha}$ develops on the opposite side of the plate where the shockwave hits. Thus the expression of total pressure for a water-backed plate can be written as:

$$p_t = 2p_i - \frac{2\rho c V_s}{cos\alpha} \tag{2.48}$$

An analytical solution for the structural response of an air-backed plate i.e. the plate velocity can be obtained based on the incident pressure formulation,

$$p_i(t) = P_0 e^{-t/\theta} \tag{2.49}$$

Let 'M' be the mass of the air-backed plate, 'A' be the plate cross-sectional area and the initial conditions are:

$$x(0) = 0; \left.\frac{dx}{dt}\right|_{x=0} = 0 \tag{2.50}$$

The Newton's second law of motion can be applied as follows:

$$M\frac{dV_s}{dt} = A\left(2p_i - \frac{\rho c V_s}{cos\alpha}\right) \tag{2.51}$$

which can be re-arranged as:

$$m_s\frac{dV_s(t)}{dt} + \frac{\rho c}{\cos\alpha}V_s(t) = 2P_0 e^{-t/\tau} \tag{2.52}$$

Here, $m_s = \frac{M}{A}$ (Mass per unit area). If we define a variable, $\beta = \frac{\rho c\theta}{\mathrm{m_s}\cos\alpha}$, the solution to the above differential equation can be written as:

$$V_s(t) = \frac{2P_0\theta}{m_s(1-\beta)}\left[e^{-\left(\frac{\beta t}{\theta}\right)} - e^{-\left(\frac{t}{\theta}\right)}\right] \tag{2.53}$$

The corresponding total pressure term is:

$$p_t(t) = P_0\left[\frac{2}{1-\beta}e^{-\left(\frac{t}{\theta}\right)} - \frac{2\beta}{1-\beta}e^{-\left(\frac{\beta t}{\theta}\right)}\right] \tag{2.54}$$

The kick-off velocity ($v_{p_{max}}$) and the rise-time to attain that kick-off velocity ($t_{\max}$) can be derived from the equation as:

$$v_{p_{max}} = \frac{2P_0\theta}{m_p}\beta^{\frac{\beta}{1-\beta}} = \quad ; \quad t_{max} = \theta\frac{\ln(\beta)}{\beta - 1} \tag{2.55}$$

The high pressure associated with the shockwave causes the plate to reach high velocities rapidly. However, the inward acceleration of the plate causes tension in the fluid on the surface leading to local fluid cavitation. The Kennard paper [45] proposes a modification to the Taylor Flat Plate theory equation to model the local cavitation with a pressure cut-off as follows:

$$p_t = \begin{Bmatrix} 2p_i - \frac{\rho c V_s}{cos\alpha}\ ;\ p_t > p_c \\ p_t = p_c\ ;\ otherwise \end{Bmatrix} \tag{2.56}$$

Here p_c is the cavitation or cut-off pressure which corresponds to the pressure in the cavitation bubble region and for water, it is taken to be zero.

The Taylor Flat Plate theory equation can also be extended to incorporate the atmospheric pressure and the hydrostatic pressure. The equation then becomes:

$$p_t(t) = \begin{Bmatrix} 2p_i(t) - \frac{\rho c V_s}{cos\alpha} + p_{atm} + \rho_w g h\ ;\ p_t > p_c \\ p_t = p_c \qquad\qquad\qquad\qquad ;\ otherwise \end{Bmatrix} \tag{2.57}$$

where p_{atm} is the atmospheric pressure and $\rho_w gh$ is the hydrostatic pressure. (h is the depth at which the plate is located) .

2.2.1 Analytical Taylor plate equation - Sample case

In this section a simple application of Taylor Flat plate theory is used to discuss the ship structural response to underwater shockwaves [46]. The following are the parameters involved:

Density of water, $\rho_w = 1025\ kg/m^3$; Speed of sound in water, $c = 1514\ m/s$.

Mass of an explosive charge, $W = 6.739\ kg$.

Distance to rigid plate located vertically above the charge on the free surface, $R = 2.127\ m$.

Angle of incidence of the shockwave on the plate, $\alpha = 0°$.

From the similitude relations discussed in Section 1.2.5, the peak pressure and decay constant are calculated as:

$$P_{max} = 52.4\left(\frac{W^{\frac{1}{3}}}{R}\right)^{1.13} = 45.808\ MPa \tag{2.58}$$

$$\theta = 0.084 \times W^{\frac{1}{3}} \times \left(\frac{W^{\frac{1}{3}}}{R}\right)^{-0.23} = 1.63\ ms \tag{2.59}$$

For the rigid plate on the free surface: if $Mass = 10\ kg$ and $Area = 1\ m^2$, then mass per unit area, $m_p = 10\ kg/m^2$ and:

$$\beta = \frac{\rho c \theta}{\mathrm{m_p} \cos \alpha} = 25.265 \tag{2.60}$$

From the analytical solution to the ODE of the Taylor plate equation obtained in Eqn. (*2.53)* and Eqn.*(2.54)*, the velocity profile is calculated as shown in *Figure* 14 and the pressure profile as shown in *Figure 15*. The kick-off velocity ($v_{p_{max}}$) and the rise-time to attain that kick-off velocity ($t_{\max}$) are calculated from Eqn. *(2.55)* as:

$$v_{p_{max}} = \frac{2P_0\theta}{m_p}\beta^{\frac{\beta}{1-\beta}} = 51.67\frac{m}{s} \quad ; \quad t_{max} = \theta\frac{\ln(\beta)}{\beta - 1} = 21.7\ \mu s \tag{2.61}$$

Applying the modification to include pressure cut-off in Taylor Flat Plate theory given by Eqn. *(2.56)*, the total pressure is shown along with the previous result in *Figure 15* and the corresponding velocity profile is shown in Figure *14*.

The pressure plot in Figure 15 shows how the total pressure on the rigid plate varies with and without pressure cut-off. The incident pressure profile exponentially decays in time, and the

total pressure on the plate, given by the Taylor Flat Plate model, has the maximum magnitude of twice the incident pressure at t = 0, when the plate velocity, $V_s = 0$.

If there is no pressure cut-off considered, the initial pressure causes the plate to start moving upwards. As time progresses the velocity reaches a maximum kick-off velocity. During this time the plate velocity reduces the total pressure on the plate by a multiple of acoustic impedance ($-\rho c V_s$) combined with the fact that the incident pressure is also reducing exponentially. Eventually the total pressure becomes negative. This causes the force to be in the re-bound direction and causes a gradual reduction in velocity until the velocity eventually reaches zero.

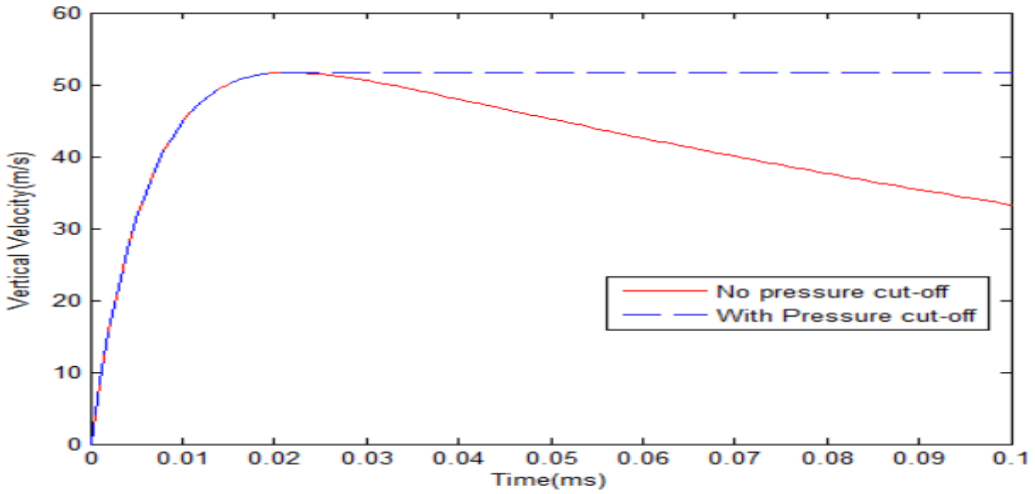

Figure 14:Vertical velocity profile of rigid plate (No pressure cut-off considered)

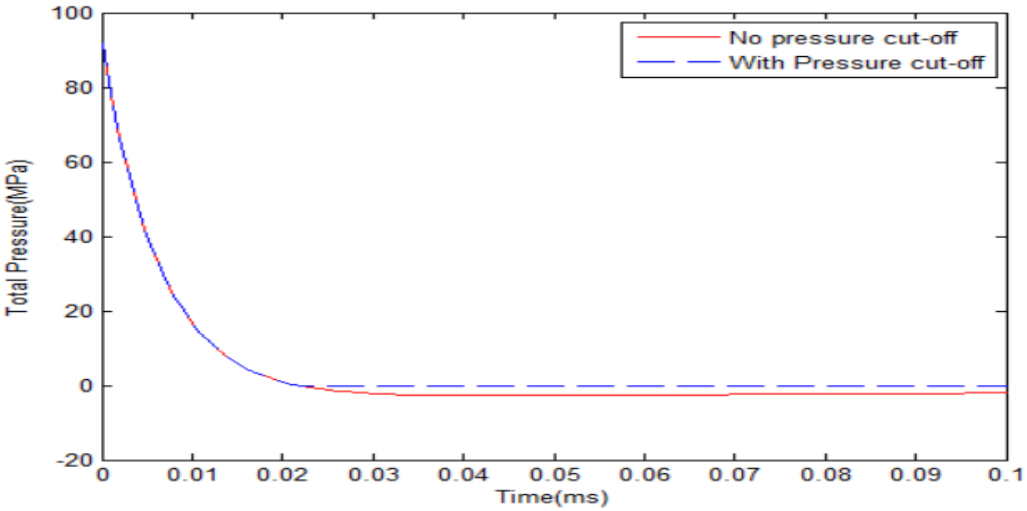

Figure 15: Total pressure plot based on Taylor Flat Plate theory

If a pressure cut-off is implemented, i.e. the total pressure is set to zero (vapor pressure of water) when the total pressure starts going negative, this velocity reduction never happens. In this case, the plate would rebound only if there is stiffness added to force it back as in the next example.

2.2.2 Taylor Flat Plate theory in conjunction with a spring-damper system

It is important to understand how Taylor Flat Plate theory might behave when applied to a real structure. This can be modeled in a very simplistic way using a spring-damper [46]. The overdamped and underdamped cases with no pressure-cut-off are explored. The case setup is as shown in Figure 16. The setup uses the same rigid plate model from the derivation of Taylor Flat Plate theory, however, a stiffness, k, and damping, b, are added to model the surrounding structure. The plate moves in the vertical direction due to the incident shockwave. The atmospheric pressure considered while calculating the total force from the fluid on the structure is also acting from the top and hence can initially be considered balanced. It is important to consider atmospheric pressure explicitly in order to properly predict the onset of cavitation and cutoff on the plate in the fluid.

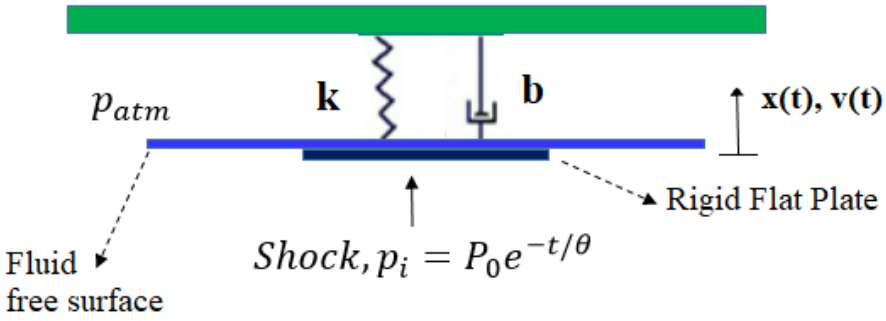

Figure 16: Taylor Plate with Spring and Damper

The hydrostatic pressure in the basic Taylor Flat Plate theory equation given by Eqn. *(2.57)* is modified by separating the hydrostatic term $\rho_w g S_0$ into a buoyancy term, $\rho_w g T$, and a reduction in buoyancy term, $-\rho_w g x$: [46]

$$p_t(t) = 2p_i(t) - \frac{\rho c V_s}{cos\alpha} + p_{atm} + \rho_w g T - \rho_w g x \tag{2.62}$$

After removing the $\rho_w gT\ term$, which is balanced by the weight term, Mg, both of which are also very small values, the equation of motion can be written as:

$$M\frac{d^2x(t)}{dt^2} + b\frac{dx(t)}{dt} + kx(t) = A(p_t - p_{atm}) \\ = A\left(2p_i - \frac{\rho c}{cos\alpha}\frac{dx(t)}{dt} - \rho g x\right) \tag{2.63}$$

Where M is the mass of the plate, b is the damping in the system, k is the stiffness of the system and A is the cross-sectional area. The terms can be re-arranged according to the coefficients of $\frac{dx}{dt}$ and x:

$$M\frac{d^2x(t)}{dt^2} + \left(b + \frac{\rho cA}{cos\alpha}\right)\frac{dx(t)}{dt} + (k + \rho gA)x(t) = 2AP_0e^{-t/\theta} \tag{2.64}$$

The equivalent SDOF equation of motion can be written as:

$$M\frac{d^2x(t)}{dt^2} + B\frac{dx(t)}{dt} + Kx(t) = 2AP_0e^{-\frac{t}{\theta}}; \\ where\ B = \left(b + \frac{\rho cA}{cos\alpha}\right); K = (k + \rho gA) \tag{2.65}$$

The above formulation is valid provided:

1. No added mass
2. No cavitation/pressure cut-off
3. The displacements are very small and plate stays in water ($x < plate\ thickness$)
4. The plate bottom is at x, x positive upwards, plate floating on surface.

Eqn. *(2.65)* is an ordinary differential equation for a forced oscillator with a homogeneous solution, x_h and a particular solution, x_p. It can be written as:

$$x_h = C_1e^{r_1t} + C_2e^{r_2t}\ ; x_p = Xe^{-t/\theta}; x(t) = x_h(t) + x_p(t) \tag{2.66}$$

Where:

$$M\frac{d^2x_h(t)}{dt^2} + B\frac{dx_h(t)}{dt} + Kx_h(t) = 0 \tag{2.67}$$

And r is given by:

$$r_1, r_2\ = -\frac{B \pm \sqrt{B^2 - 4MK}}{2M} \tag{2.68}$$

The solution to the forced oscillator can be written as:

$$x(t) = x_h(t) + x_p(t) \tag{2.69}$$

The initial conditions for the incident shockwave problem are that the initial displacement at velocity at t=0 is zero as written below:

$$x(0) = C_1 + C_2 + X = \ 0\ ;$$
$$x'(0) = r_1 C_1 + r_2 C_2 - \frac{X}{\theta} = \ 0 \tag{2.70}$$

Also substituting the expressions for x and x' at t = 0 into the equation of motion, we get:

$$M\frac{d^2 x_p(0)}{dt^2} = 2AP_0 \tag{2.71}$$

Thus the constants, X, C_1 and C_2 are derived as:

$$X = \frac{2AP_0\theta^2}{M};$$
$$C_1 = \frac{2AP_0\theta}{M} \times \frac{\theta r_2 + 1}{r_1 - r_2}\ ;\ C_2 = -C_1 - X = \ -C_1 - \frac{2AP_0\theta^2}{M} \tag{2.72}$$

If we consider the nature of roots, for the homogeneous solution, r_1 and r_2, if $B^2 > 4MK$, then both r_1 and r_2 are negative, real and unequal, and the system is overdamped. If $B^2 < 4MK$, then both r_1 and r_2 are complex and unequal, the system is underdamped and there will be oscillations.

2.2.2.1 Overdamped Case ($B^2 > 4MK$).

As an example of the overdamped case, the following values for stiffness and damping are used along with the case set-up used in Section 2.2.1 where the mass of plate is 10 kg:

Plate Damping, $b = 7 \times 10^4\ kg/s$

Plate Stiffness, $k = 10^{10}\ kg/s^2$

Then the damping and the stiffness terms in the equation of motion as derived in Section 2.2.2 are:

$$B = \left(b + \frac{\rho cA}{cos\alpha}\right) = 1.622 \times 10^6 \frac{kg}{s};$$
$$K = \ (k + \rho gA) \ = \ 10^{10}\ kg/s^2 \tag{2.73}$$

Now the natural frequencies of the system are calculated as:

$$r_1, r_2 \ = -\frac{B \pm \sqrt{B^2 - 4MK}}{2M};$$
$$r_1 \ = \ -6420\ Hz\ ;\ \ r_2 \ = \ -155765\ Hz\ ; \tag{2.74}$$

The constants are calculated as:

$$C_1 \ = \ -0.243\ m\ ;\ \ C_2 = 4.513 \times 10^{-4}\ m\ ;\ \ X = 0.2434\ m \tag{2.75}$$

The total pressure and its components for the case without spring and damper are plotted in Figure *17*. This can be compared against Figure *18* which is the pressure plot for the overdamped case set-up. Because the spring-damper quickly reduces the plate velocity as shown in Figure *19* the total pressure on the plate does not go negative and there is no cavitation cut-off. The two components in the Taylor Flat Plate equation corresponding to incident shock wave and the plate velocity are plotted along with total pressure in Figure *18*. The Taylor Flat Plate pressure correction term, $\rho c v_s$, is observed to never cross above the $2\ p_i$ term so the total pressure never goes negative.

The vertical velocity time history shown in Figure *19* corresponds to the overdamped case. The stiffness forces the plate to reduce velocity more drastically compared to the basic Taylor Flat Plate model after reaching the kick-off velocity and goes on to have negative velocity (meaning mass rebounds or comes back down). Being an overdamped case, the velocity after going negative quickly goes back to zero. In the no spring-damper case, the negative pressure is the only factor bringing the velocity down from the kick-off velocity.

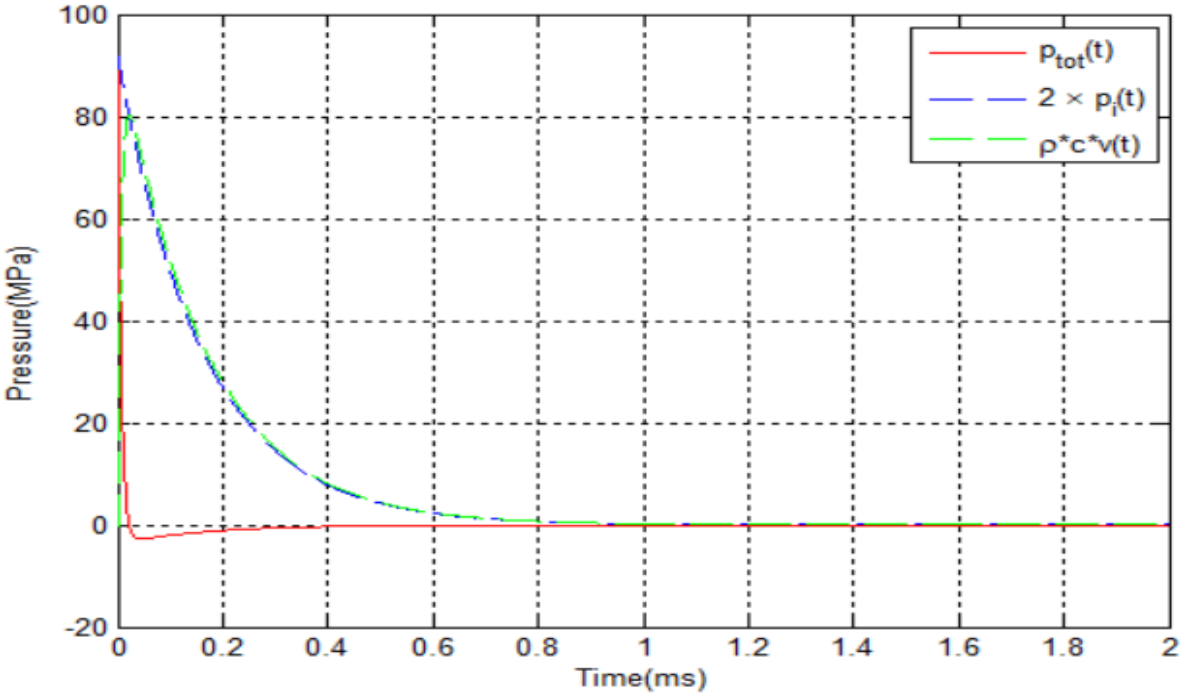

Figure 17:Pressure plot based on analytical solution of Taylor Plate with no spring and damper

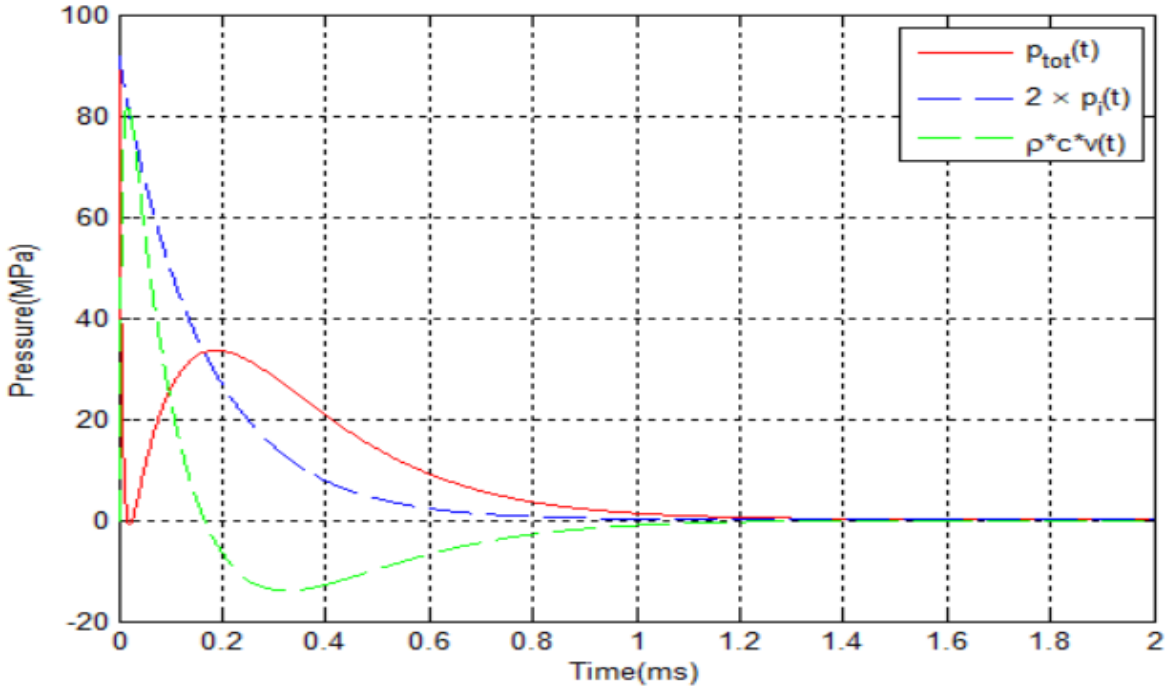

Figure 18: Pressure plot based on analytical solution of Taylor Plate with spring and damper – Overdamped case

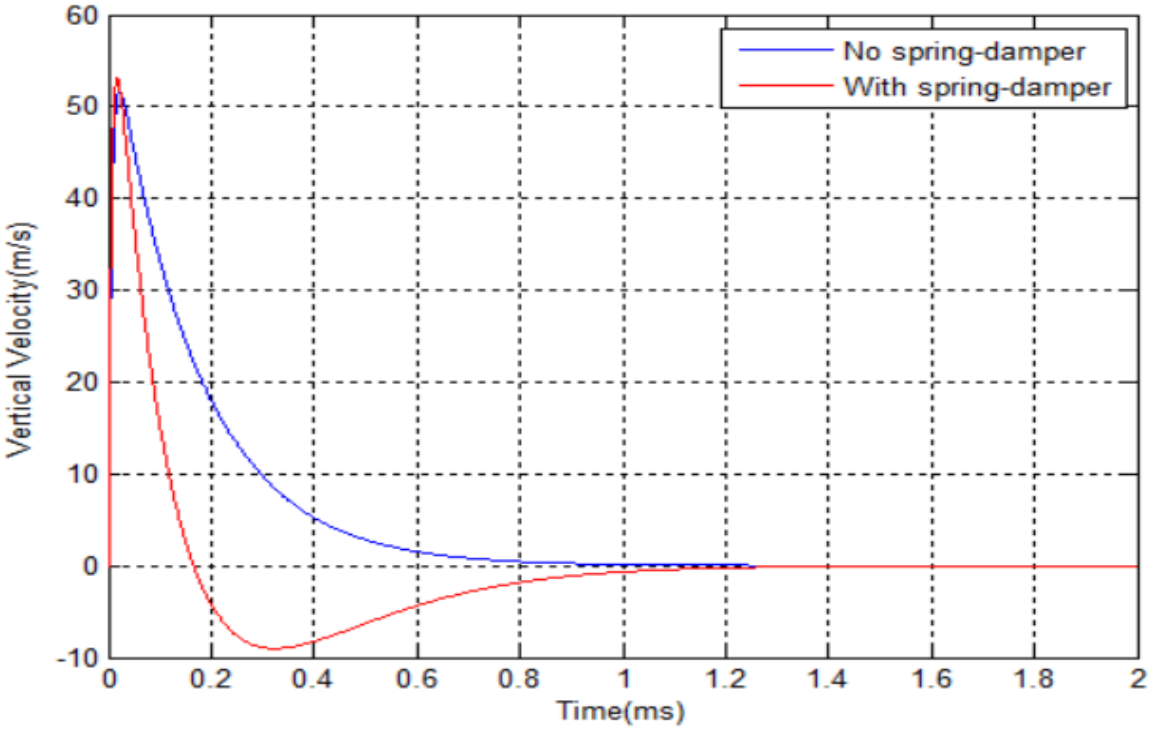

Figure 19: Vertical velocity of Taylor Flat Plate model, with and without spring-damper and no pressure cut-off implementation.- Overdamped case

The velocity profile of the overdamped spring-damper case is more indicative of the actual case where the shell plating of a ship is deformable and has adjacent plates or stiffeners giving it some form of stiffness and damping.

2.2.2.2 Underdamped Case ($B^2 < 4MK$).

As an example of the underdamped case, the same values for stiffness and damping are used as in the overdamped case, but the mass of plate is increased to 1000 kg so that $B^2 < 4MK$.

The damping and the stiffness terms in the equation of motion as derived in Section 2.2.2 become:

$$B = \left(b + \frac{\rho c A}{cos\alpha}\right) = 1.622 \times 10^6 \frac{kg}{s}; \qquad K = (k + \rho g A) = 10^{10}\ kg/s^2 \tag{2.76}$$

Now the r_1 and r_2 are calculated and is observed to be complex values:

$$r_1, r_2 = -\frac{B \pm \sqrt{B^2 - 4MK}}{2M}; \qquad r_1 = -810.925 + 3.056\text{i}\ Hz\,; \quad r_2 = -810.925 - 3.056\text{i}\ Hz\,; \tag{2.77}$$

The constants are calculated as:

$$C_1 = -0.001217 - 0.00212i\ m\,; \qquad C_2 = -0.001217 + 0.00212i\ m \qquad X = 0.2434\ m \tag{2.78}$$

The pressure plot for this underdamped case is shown in Figure *20* and the velocity response is shown in *Figure* 21. Figure *21* shows an oscillatory pattern characteristic of an underdamped case and the complex roots of the homogeneous equation. The kick-off velocity for the plate shown in Figure *21* is much lower in this case due to the higher mass, and the oscillations show that there are multiple rebounds for the plate before the velocity attenuates to zero. The pressure history shows that the total pressure remains above zero except for a small duration where it goes slightly below zero as part of the oscillation but the stiffness and damping raises the pressure to be zero very quickly.

This case is even more indicative of actual scenarios where the plate is bounded by other plates or stiffeners which provides stiffness and damping to the plate and the system is underdamped. Thus the simple Taylor Flat Plate theory model with some adjustments to its application is able to provide a somewhat realistic shock response in terms of the values of kickoff

velocity, rise time and behavior. The curves are not noisy as in an actual ship as there is only a single natural frequency and not multiple different modal frequencies indicative of a multi-degree of freedom system.

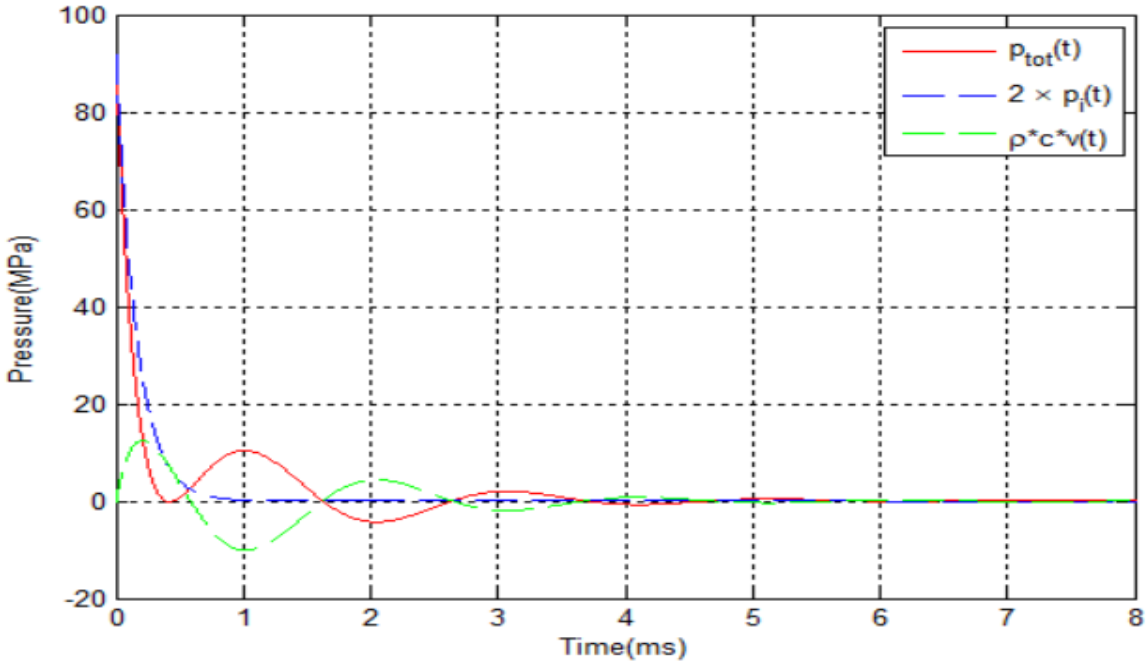

Figure 20: Pressure plot based on analytical solution of Taylor Plate with spring and damper – Underdamped case

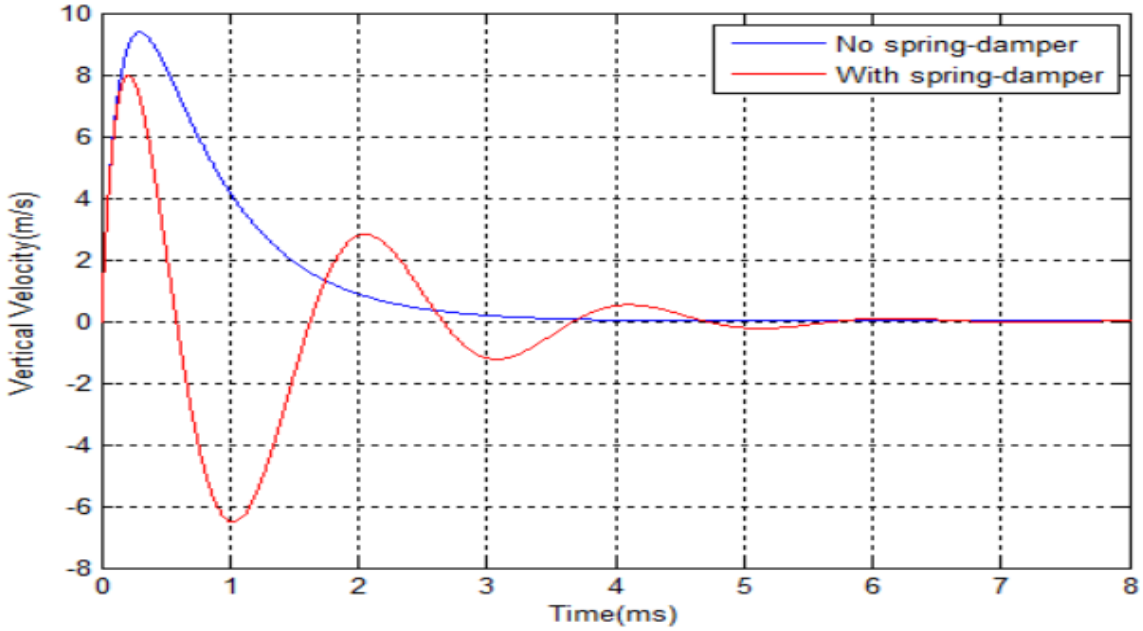

Figure 21: Vertical velocity of Taylor Flat Plate model, with and without spring-damper and no pressure cut-off implementation. – Underdamped Case

2.3 Cavitating Acoustic Elements

The governing equations of the Cavitating Acoustic Spectral Element formulation is briefly discussed below [8]. The validity of the acoustic treatment of the fluid was discussed in Section 1.5.1.5. The results in this thesis use the results obtained using the Spectral Element approach for validation.

We assume that the fluid can be modeled as a Navier-Stokes fluid and thus its motion is governed by the Euler momentum equation and the continuity equation. The total condensation [37, 47] is given by:

$$S = \frac{\rho}{\rho_0} - 1 \tag{2.79}$$

The total displacement,$\vec{\xi}$, of a fluid particle is defined as [48]:

$$\vec{\xi}(\vec{X}, t) = \vec{x}(\vec{X}, t) + \vec{x}_{eq}(\vec{X}) \tag{2.80}$$

Where x is the dynamic displacement of a fluid particle, x_{eq} is the equilibrium displacement of a fluid particle, and $\vec{X}$ is the global Cartesian coordinate vector, and t is time.

We assume that the fluctuations between density, ρ, and the reference density, $\rho_{0,}$ remain small for an acoustic fluid, although it is compressible. The continuity equation, with displacement of the fluid particle measured relative to the equilibrium value is:

$$\frac{\partial \rho}{\partial t} + \nabla.\left(\rho \frac{\partial \vec{x}}{\partial t}\right) = 0 \tag{2.81}$$

The above equation written in terms of total condensation is:

$$\frac{\partial S}{\partial t} + \nabla.\left(\frac{\partial \vec{x}}{\partial t}\right) = 0 \tag{2.82}$$

For $|S| \ll 1$. Similarly, the Euler momentum equation can be simplified by neglecting viscous effects and convective acceleration terms as:

$$\rho \frac{\partial^2 \vec{x}}{\partial t^2} = \rho_0 \vec{f} - \nabla p^t \tag{2.83}$$

Where p^t is the total fluid pressure and $\vec{f}(\vec{X})$ is the body force field.

2.3.1 Displacement Potential formulation

The state of the fluid is assumed to consist of equilibrium, incident and scattered fields. The equilibrium field consists of atmospheric and hydrostatic pressure. The incident field is caused by the shock wave propagating in a homogeneous unbounded fluid. The scattered field is caused

by the presence of both the structure and the free surface. Two models are considered for the fluid while solving the fluid-structure interaction problem. The first, the total field model, has its dependent variables composed of both incident and scattered fields which are unknown, while the equilibrium field is known. The second is the scattered field model which has its dependent variables composed of only the unknown scattered field whereas the incident field and the equilibrium field is known for all time.

The dynamic displacement potential of the fluid is defined as

$$-\nabla\Psi = \vec{x} = \vec{\xi} - \vec{x}_{eq} \tag{2.84}$$

When the displacement potential shown above is differentiated w.r.t time and then substituted into Eqn. 2.82, we get

$$\frac{\partial S}{\partial t} - \nabla^2 \frac{\partial \Psi}{\partial \mathrm{t}} = 0 \tag{2.85}$$

The above equation can be integrated in time to obtain the strain-displacement relation for CAFÉ and CASE.

$$S - \nabla^2\Psi = S_{eq}(\vec{X}) \tag{2.86}$$

Here S_{eq} represents the equilibrium condensation in the fluid. From Eqn. 2.84 and Eqn. 2.86, the dynamic condensation,S_{dyn}, is the measure of volumetric strain or dilation of the fluid.

$$S_{dyn} = S - S_{eq} = \nabla^2\Psi = -\nabla.(\vec{x}) \tag{2.87}$$

Now applying Eqn. 2.84 into Eqn. 2.83, the momentum equation can be written as:

$$-\rho_0 \frac{\partial^2 \nabla\Psi}{\partial t^2} = \rho_0 \vec{f} - \nabla p^t \tag{2.88}$$

For a static equilibrium problem, the body force term is the hydrostatic pressure of the liquid. Thus the above equation becomes

$$-\rho_0 \frac{\partial^2 \nabla\Psi}{\partial t^2} = \nabla(p^H - p^t) \tag{2.89}$$

Spatial integration of the above equation yields,

$$-\rho_0 \frac{\partial^2 \Psi}{\partial t^2} = p^H - p^t + k(t) \tag{2.90}$$

Here $k(t)$ is the constant of integration. As in [37], equilibrium pressure is defined to be, $p_{eq} = p^H + k(t)$. Then the above equation becomes:

$$\rho_0 \frac{\partial^2 \Psi}{\partial t^2} = p \tag{2.91}$$

Where p ($p = p^t - p_{eq}$) is the dynamic pressure.

Since the underwater explosion modeling involves cavitation, a possible simple but adequate model is a bilinear constitutive relation for the fluid. The bilinear equation of state for a Cavitating acoustic fluid is [49]:

$$p^t = p^{cav} + \begin{Bmatrix} B_0 \left(\frac{\rho}{\rho_0} - 1\right); \rho \geq \rho_0 \\ 0 \qquad\qquad ; \rho < \rho_0 \end{Bmatrix} \tag{2.92}$$

Where ρ is the density, and ρ_0 is the density of uncavitated fluid or reference density, B_0 is the bulk modulus of the uncavitated acoustic fluid, p_{cav} is the vapor pressure of the fluid and p_t is the total fluid pressure. The bulk modulus is defined as

$$B_0 = \rho_0 c^2 \tag{2.93}$$

Where c is the speed of sound in the uncavitated acoustic fluid.

Now, the bilinear equation of state (Eqn. 2.92) can be written in terms of total condensation as:

$$p^t = p_{cav} + \begin{Bmatrix} \rho_0 c^2 S \quad ; \ if\ S > 0 \\ 0 \qquad ; otherwise \end{Bmatrix} \tag{2.94}$$

Rewriting the above in terms of dynamic pressure, we have:

$$p = p_{cav} - p_{eq} + \begin{Bmatrix} \rho_0 c^2 S \quad ; \ if\ S > 0 \\ 0 \qquad ; otherwise \end{Bmatrix} \tag{2.95}$$

From [37], we define the densified dynamic displacement potential as:

$$\psi = \rho_0 \Psi \tag{2.96}$$

And densified dynamic condensation as:

$$s = \rho_0 (S - S_{eq}) = -\rho_0 \nabla . (\vec{x}) \tag{2.97}$$

Using Eqns. 2.96 and 2.97, we have the final system of CASE governing equations as:

$$s = \nabla^2 \psi \tag{2.98}$$

$$\frac{\partial^2\psi}{\partial t^2} = \begin{cases} c^2 s & ;\ if\ s > \dfrac{p_{cav} - p_{eq}}{c^2} \\ p_{cav} - p_{eq} & ;\ otherwise \end{cases} \tag{2.99}$$

The scattered field model has only the scattered field as unknown dependent variables. The governing equation for scattered field model is described in [8].

2.3.2 Spatial discretization using CASE

The method used is sub-parametric discretization whereby, geometry is represented using linear basis functions and the dependent variables are represented using higher-order basis functions. The fluid domain is divided into n_{el} non-overlapping hexagonal elements. The dependent field variables can be represented as:

$$s(\xi,\eta,\zeta,t) = \phi^{\mathrm{T}}(\xi,\eta,\zeta)s^{el}(t) = \sum_{i,j,k=0}^{N} \phi_i(\xi)\phi_j(\eta)\phi_k(\zeta)s_{ijk}^{el}(t) \tag{2.100}$$

$$\psi(\xi,\eta,\zeta,t) = \phi^{T}(\xi,\eta,\zeta)\psi^{el}(t) = \sum_{i,j,k=0}^{N} \phi_i(\xi)\phi_j(\eta)\phi_k(\zeta)\psi_{ijk}^{el}(t) \tag{2.101}$$

where s^{el} and ψ^{el} are column vectors of time-dependent nodal values ($s_{ijk}^{el}(t)$ and $\psi_{ijk}^{el}(t)$), ϕ is a column vector consisting of N^{th} order polynomial basis functions $\phi_i(\xi)\phi_j(\eta)\phi_k(\zeta)$ and ξ,η,ζ are the element natural coordinates($-1 \le \xi,\eta,\zeta \le 1$)

Sprague and Geers [37] uses a Gauss-Lobatto-Legendre(GLL) element node distribution for the CASE discretization which are given by the roots of

$$(1-\xi^2)L'_N(\xi) = 0 \tag{2.102}$$

Here L'_N is the first derivative of the N^{th} degree Legendre polynomial w.r.t the argument. The basis function is defined using Lagrangian interpolants given by:

$$\phi_i(\xi) = \frac{(1-\xi^2)L'_N(\xi)}{N(N+1)L_N(\xi_i)(\xi-\xi_i)} \tag{2.103}$$

Where ξ_i is the i^{th}node given by GLL distribution discussed before. The above equation also satisfies the relation,

$$\phi_i(\xi_i) = \delta_{ij} \tag{2.104}$$

Where δ_{ij} is the Kronecker Delta.

The derivatives of the spectral basis functions evaluated at the GLL nodes are given by:

$$\left.\frac{\partial \phi_i}{\partial \xi}\right|_{\xi=\xi_i} = \begin{cases} \dfrac{L_N(\xi_j)}{L_N(\xi_i)}\dfrac{1}{\xi_j-\xi_i}; & i \neq j \\ -\dfrac{N(N+1)}{4}; & i=j=1 \\ \dfrac{N(N+1)}{4}; & i=j=N \\ 0; & otherwise \end{cases} \quad \begin{matrix} i = 1,2,\ldots..N+1 \\ j = 1,2,\ldots..N+1 \end{matrix} \tag{2.105}$$

The governing equations for CASE discretization are solved numerically using standard Galerkin approach. The test function,ϕ, is pre-multiplied to Eqn. 2.98 and then integrated over the fluid element domain,Ω_f:

$$\int_{\Omega_f} [s\phi - \nabla^2\psi\phi] d\Omega \; = 0 \tag{2.106}$$

Now Green's first identity is applied yielding

$$\int_{\Omega_f} s\phi d\Omega \; + \int_{\Omega_f} \nabla\phi \cdot \nabla\psi \; d\Omega = \int_{\Gamma_f} \phi\nabla\psi \cdot \vec{n} \; d\Gamma \tag{2.107}$$

Where Γ_f is the fluid boundary in element domain and $\vec{n}$ is the outward pointing normal vector on Γ_f. Substituting Eqn. 2.100and 2.101 into the weak formulation developed above, we get the matrix form of the CASE strain-displacement equation as follows:

$$\mathbf{Q^{el}s^{el}} + \mathbf{H^{el}}\boldsymbol{\psi^{el}} \; = \mathbf{b^{el}} \tag{2.108}$$

Where

$$\mathbf{Q^{el}} = \int_{\Omega_f} \phi\phi^{\mathrm{T}} \, d\Omega \; ; \quad \mathbf{H^{el}} = \int_{\Omega_f} \nabla\phi \cdot \nabla\phi^{\mathrm{T}} \, d\Omega \, ; \; \mathbf{b^{el}} \; = \int_{\Gamma_f} \phi\nabla\psi \cdot \vec{n} \, d\Gamma \tag{2.109}$$

Here $\mathbf{Q^{el}}$ is called the elemental capacitance matrix, $\mathbf{H^{el}}$ is the element reactance matrix, and $\mathbf{b^{el}}$ is the elemental boundary-interaction vector. The element level equations are assembled to form the global system to obtain the semi-discrete equations:

$$\mathbf{Qs} + \mathbf{H}\boldsymbol{\psi} = \mathbf{b} \tag{2.110}$$

The integrals in the matrix, $\mathbf{Q}$, $\mathbf{H}$ and $\mathbf{b}$ are evaluated based on the spectral basis functions and its derivatives formulated in Eqns. 2.103 and 2.104.This treatment automatically lumps the capacitance matrix, $\mathbf{Q}$ as the element nodes and quadrature points as coincident. Also, the global reactance matrix is evaluated as a product $\mathbf{H}\boldsymbol{\psi}$ element-by-element at each time-step and not explicitly calculated or stored.

2.3.3 Boundary Conditions in CASE

Pressure, symmetry plane, fluid-structure interaction and non-reflecting(NRB) boundaries are the main types of boundaries applied to the matrix formulation given by Eqn. (*2.110*. The essential boundaries are the known pressure boundaries. The boundary values at the known pressure boundaries are enforced in the time-update equations. At the symmetry plane, the normal fluid displacements i.e. $-\rho_0 \nabla\psi . \boldsymbol{n}$ becomes zero, where $\boldsymbol{n}$ is the normal vector to the plane.ψ is left free along the symmetry planes, being a natural boundary condition. The other natural boundary conditions are given by the fluid-structure interaction term and the non-reflecting boundary term, both of which are governed by the fluid force vector, b.

$$\boldsymbol{b} = \boldsymbol{b^s} + \boldsymbol{b^{nrb}} \tag{2.111}$$

In the above equation, $\boldsymbol{b^s}$ is the force of the structure on the fluid and $\boldsymbol{b^{nrb}}$ is the force of the NRB on the fluid. In the total field model, the $\boldsymbol{b^s}$ is defined as:

$$\boldsymbol{b^s} = -\rho_0 \int_{\Gamma_s^e} \phi \bar{x}^e \, d\boldsymbol{\Gamma} \tag{2.112}$$

Here $\bar{x}^e$ is the average normal displacement of the structure(positive being out of the fluid direction) at the centre of the wetted surface area, Γ_s^e is the area separating the wetted structure element from the fluid element and the integral is evaluated in closed form.

$$\boldsymbol{b^{nrb}} = -\rho_0 \int_{\Gamma_{\mathrm{nrb}}^e} \phi\phi^{\mathrm{T}} \left(\boldsymbol{x}^e - \boldsymbol{x}_{eq}^{\boldsymbol{e}}\right) d\boldsymbol{\Gamma} \tag{2.113}$$

The above integral is the second term of the boundary condition set-up corresponding to the area of the non-reflecting boundary and is solved using GLL quadrature. The Plane Wave Approximation (PWA) is applied on the non-reflecting boundary as it is found to give similar results to the Curved Wave Approximation (CWA) and Doubly Asymptotic Approximation (DAA) and save a lot of computational time for the far-field UNDEX problem.

2.3.4 Temporal Integration in CASE method

A second-order Newmark explicit central difference scheme is used for temporal integration for both the structure and the fluid. Time-step sub-cycling is incorporated for both fluid and structure for efficient and accurate integration as the integration is then done with maximum time increments allowed by the respective stability criteria as shown in [37]. For the fluid equations, the upper bound for time increment is,

$$\Delta t_{cr} = \frac{2}{c\sqrt{\lambda_{max}\,(1+2\beta)}} \tag{2.114}$$

Here β is the artificial damping constant and λ_{max} is the maximum eigenvalue of the global generalized eigen problem,

$$(\mathbf{H} - \lambda\mathbf{Q})\mathbf{z} = \mathbf{0} \tag{2.115}$$

The fluid equations for temporal integration is written as:

$$\begin{aligned} \dot{\psi}_{n+\frac{1}{2}} &= \dot{\psi}_{n-\frac{1}{2}} + \Delta t\ddot{\psi}_n \\ \psi_{n+1} &= \psi_n + \Delta t\dot{\psi}_{n+\frac{1}{2}} \\ \ddot{\psi}_{n+1} &= Q^{-1}(-H\psi_{n+1} + b_{n+1}) \\ p_{n+1} &= c^2\ddot{\psi}_{n+1} \end{aligned} \tag{2.116}$$

The structural equations are given as:

$$\begin{aligned} \dot{u}^{i+\frac{1}{2}} &= \dot{u}^{i-\frac{1}{2}} + \Delta t\ddot{u}^i \\ u^{i+1} &= u^i + \Delta t^{i+1}\dot{u}^{i+\frac{1}{2}} \\ \ddot{u}^i &= M^{-1}(F^i - I^i) \end{aligned} \tag{2.117}$$

Where M is the diagonal lumped mass matrix, F is the applied load vector and I is the internal force vector. The implementation of the CASE model discussed above in three-dimensions for a far-field underwater explosion problem has been done by Lu[43]. A small discussion on the same has been done in Section 3.3.

2.4 Shock Response Spectrum (SRS)

The Shock response spectrum is a plot of the maximum response experienced by multiple single degree of freedom (SDOF) spring-mass systems, as a function of their different natural frequency, in response to an applied shock. The response maybe expressed in terms of acceleration, velocity or displacement [50]. Shock Response Spectrum was conceived by Dr. Maurice Biot and described in his Ph.D. thesis published in 1932[51]. It transforms the acceleration-time history of the base from the time domain to the maximum response of a SDOF system in the frequency domain. Shock Spectrum is typically used by engineers in the field of seismic design (against earthquakes), naval shock design (against underwater explosions and pyro-shock design (in aerospace vehicles).

Since naval ships have exposure to both contact and non-contact explosions, the design against shock is an important criterion while constructing new ships. As discussed in Section 1.3, possible damage not only includes rupture or deformation of the structure but damage to equipment and installations too. Ship shock transients lasting milliseconds have its peaks occur in the first few cycles and hence damping is less important. The spectra are usually calculated without damping with the exception of systems like masts where the modal values are affected due to its slenderness.

Initially, the "Static G Method", which determined forces by multiplying mass times acceleration or "G" load, was used to design equipment against shock. However, as this was not correct for most dynamical systems, complex base motion transient (excitation) solved for complex MDOF system was used. Again, this method gives a numerical force result which depends on a single input and do not represent all possibilities the ship the ship is exposed to. This led to the use of shock spectrum in the design of equipment against naval shock. The shock spectrum has to be adjusted for systems with large mass or significant shock absorbing foundation based on the discovery of "spectrum-dip" phenomenon discussed in Section 2.4.1.

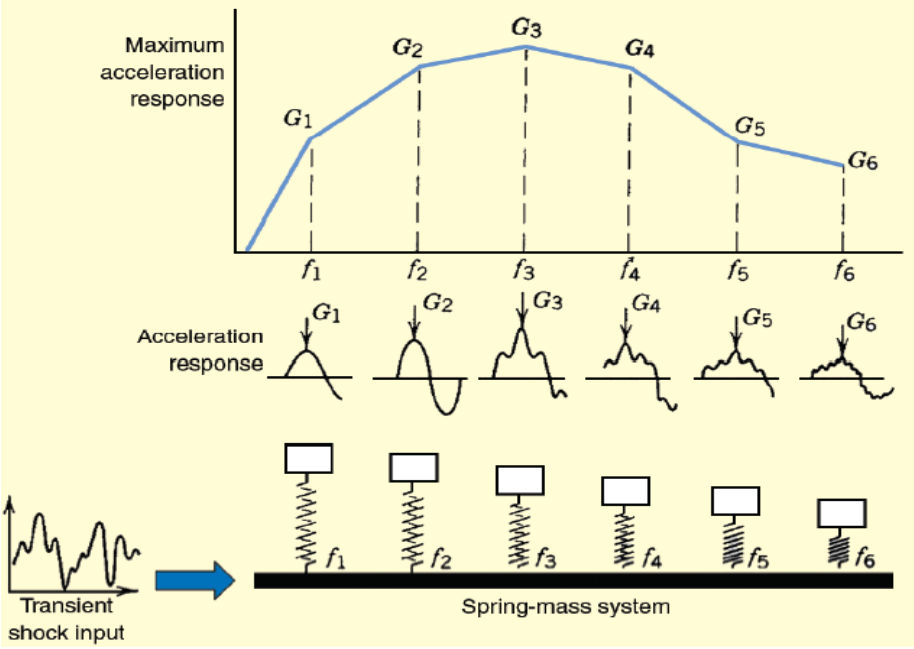

Figure 22*:* Methodology of plotting the shock spectrum [52].

Figure 22 is a visual depiction of how a shock response spectrum plot is constructed. The base of the system is initially assumed to be sufficiently massive such that the motion of the

oscillators does not affect the motion of the base (effect considered in design spectra). The acceleration response of an oscillator of a particular frequency is shown above that oscillator.

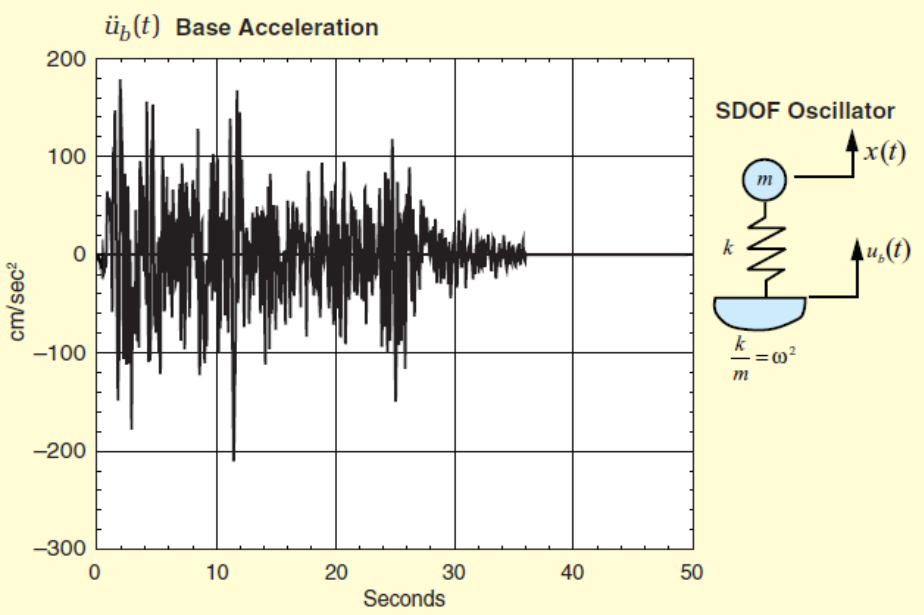

Figure 23: Base acceleration time-history and the case set-up [51]

The shock spectrum plot indicates the frequency content of the transient shock input signal provided at the base. The maximum response typically drops off for systems with higher and lower frequencies, which represent stiffer and softer systems respectively [51]. Figure 23, Figure 24 and the following equations show the mathematical development of the shock response spectrum.

Referring to Figure 23, the equation of motion of the SDOF system based on the balance of spring and inertial forces is:

$$m\ddot{x}(t) + k\big(x(t) - u_b(t)\big) = 0 \tag{2.118}$$

Defining a coordinate $z(t)$ relative to the base motion,

$$z(t) = x(t) - u_b(t) \tag{2.119}$$

Differentiating the above equation twice and substituting into the equation of motion, Eqn. 2.118:

$$\ddot{z}(t) + \omega^2 z(t) = -\ddot{u}_b(t) \tag{2.120}$$

where the absolute acceleration from the definition of relative displacement is:

$$\ddot{x}(t) = -\omega^2 z(t) \tag{2.121}$$

and ω^2 is k/m. The closed form solution to the equation of motion in relative coordinates is obtained using the Duhamel's integral given by:

$$z(t) = -\frac{1}{\omega}\int_0^t \ddot{u}_b(\tau)\sin\omega(t-\tau)\ d\tau \tag{2.122}$$

The above expression is substituted in Eqn. 2.121 to obtain the formulation of absolute acceleration. This is the important acceleration because it determines the force on the mass.

$$\ddot{x}(t) = \omega\int_0^t \ddot{u}_b(\tau)\sin\omega(t-\tau)\ d\tau \tag{2.123}$$

When damping is included the expression of absolute acceleration is:

$$\ddot{x}(t) = \omega\int_0^t \ddot{u}_b(\tau)e^{-\xi\omega(t-\tau)}\ \sin\omega(t-\tau)\ d\tau \tag{2.124}$$

where $\omega = \ \omega_d = \omega_n\sqrt{1-\xi^2}$ and $\xi = damping\ ratio.$

The shock response spectrum, S_A is defined as the maximum $|\ddot{x}(t)|$ for each frequency. Thus

$$S_A = |\ddot{x}(t)|_{max} = \left|\omega\int_0^t \ddot{u}_b(\tau)e^{-\xi\omega(t-\tau)}\ \sin\omega(t-\tau)\ d\tau\right|_{max} \tag{2.125}$$

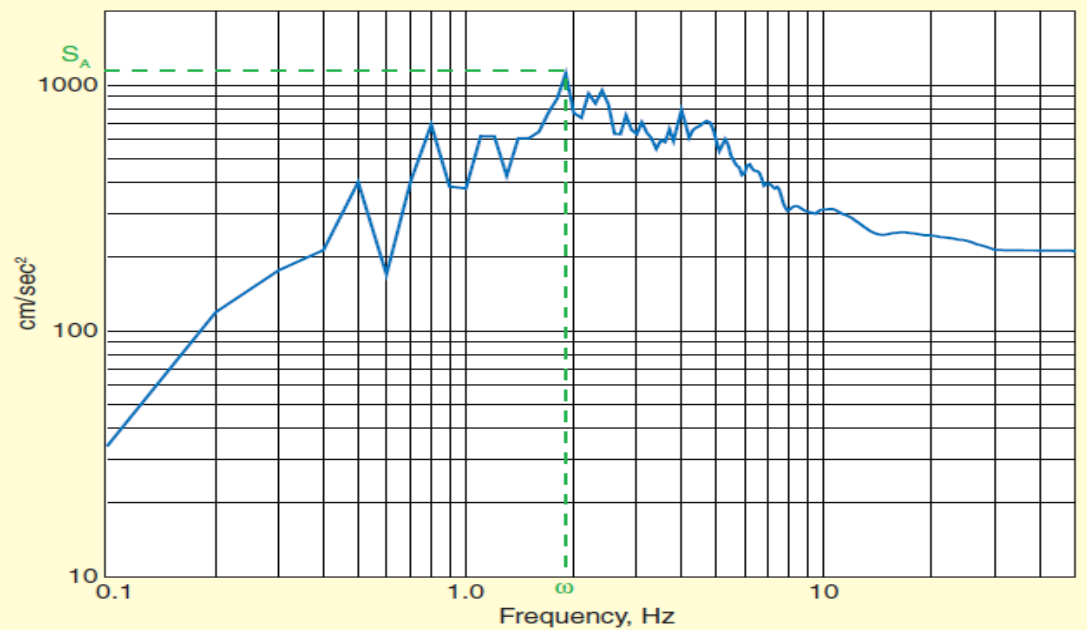

Figure 24:Shock response spectrum obtained from above system[51]

Shock response spectrum plots may be calculated for the time history while the shock is being applied or excited which is called the 'Primary' shock spectrum. The response also can be calculated based on the infinite duration after the shock transient which is called the 'Residual' shock spectrum. Damping is not considered for the residual spectrum. The whole time-history is also used to obtain the 'Overall' shock spectrum. The spectrum obtained by extracting only from

the positive peak responses is called the positive shock response spectrum whereas extracting the spectrum only from negative peak responses is called the negative shock response spectrum. This classification can be used for either the primary or the residual shock spectra.

Shock spectra are often plotted on a four-coordinate, tripartite grid. Using this plotting technique, the spectral displacement, velocity and acceleration can all be shown on the same graph since they are related linearly by ω and ω^2. The grid lines with positive 45° slope are measures of spectral acceleration($\omega^2 D \ or \ \ddot{x}_{max}$), horizontal grid lines are measures of spectral velocity(pseudo velocity, ωD) and grid lines with a negative 45° slope are measures of spectral displacement ($D \ or \ z_{max}$). Since the force on the system mass, m, is equal to $m\,\ddot{x}(t)$, $\ddot{x}(t)$ is the relevant acceleration to use, not $\ddot{z}(t)$. This is important when considering frequency response, particularly at high and low limits.

For design purposes, Biot (founder of the shock spectra concept) recommended an envelope approach for all types of spectra. For example, if we have three different kinds of spectra made available to create a design spectrum, then the maxima envelope is created from the maxima of the three shock spectra. Naval shock design spectra are discussed further in Section 2.4.1. The shock spectra plots in this thesis have been created from the acceleration time histories using a MATLAB code created by Tom Irvine and the Smallwood algorithm as described in [53, 54].

2.4.1 Spectrum-Dip effect in Shock Spectra

If the impedance of the transient motion base is large compared to the system, the reaction forces to the base motion will not alter the input transient motion. In the case of ship-shock, the impedance of the base (ship structure) is lower compared to the earth in earthquake excitation. Hence the dynamic interaction of the system inertial forces with the base structure input motion may be significant and able to alter the resulting spectrum considerably. This Spectrum-Dip effect may reduce the inputs to the system at its fixed base natural frequencies often by almost an order of magnitude. This dip must be reflected in ship design shock spectra in order to avoid significant over-design.

2.4.2 Dynamic Design Analysis Method (DDAM)

DDAM is an analytical procedure developed by the Naval Research Laboratory, U.S.A, for evaluating the design of equipment subject to dynamic loading caused by Underwater Explosions. DDAM models reduce the equipment or structure to an equivalent mass-elastic system. The desired strength levels are specified in terms of design spectral values which are

frequency and mass dependent [55]. By solving the equations of motion, the forces and displacement of each modal mass/structure in the system is determined and compared to specified allowable values.

DDAM involves five phases:

1. Problem formulation phase
2. Mathematical modeling phase
3. Coefficient computation phase
4. Dynamic computation phase
5. Evaluation phase

DDAM uses the design spectrum values for each mode of the system as a function of the frequency and modal mass of each mode (determined by the Eigen vector of mode of interest and the mass matrix of the system). The modal mass, which represents the effective contribution due to the i^{th}mode of the system is given by:

$$\bar{M}_i = \frac{[\{\varphi\}_i^T [M]\{1\}]^2}{\{\varphi\}_i^T [M]\{\varphi\}_i} \tag{2.126}$$

Based on studies conducted by O'Hara [56], it has been observed that as modal mass increases, the velocity and acceleration response decreases. DDAM considers this spectrum-dip effect for calculating the maximum spectra value for each mode.

2.4.3 Naval Shock Design Shock Spectra

Naval shock design is based on shock from a base motion transient, as induced by an underwater explosion. The design spectrum is typically divided into three regions: constant displacement limit, constant velocity limit and constant acceleration limit. For high UNDEX loadings, if the frequency of the system being excited by the base motions is high enough, the acceleration of the excited system is equal to the peak acceleration of the input motion [18]. In other words, as we consider the higher frequencies, the SDOF mass system acts more like a rigid body (large k/m) and the mass moves along with the base with a constant acceleration and the peak value for that frequency is same as the peak of base acceleration ($\ddot{u}_{b_{max}}$). As we move towards the low frequency zones, the SDOF mass system acts with very low rigidity (small k/m i.e. more inertia against the base motion) and hence the relative displacement will be the same as base displacement and thus the response spectra in this range will have the plot corresponding to constant displacement equal to maximum base displacement ($u_{b_{max}}$). In the mid-range of

frequency, the design spectra is bounded by values of constant velocity corresponding to kick-off velocity ($V_{kick-off}$).

Based on the results of many different shock tests on surface ships, design spectra values which vary with ship location and equipment weight have been developed [18]. The design spectra can be described as a constant velocity region plus a constant acceleration region.

For surface ships, the equations for the acceleration design spectra have the following form for the shell- and hull-mounted equipment:

$$A_0 = \frac{c_4(c_1 + W)(c_2 + W)}{(c_3 + W)^2}, \text{g's} \tag{2.127}$$

And for deck-mounted equipment:

$$A_0 = \frac{c_5(c_6 + W)}{c_3 + W}, \text{g's} \tag{2.128}$$

The velocity inputs for both conditions are:

$$V_0 = \frac{c_7(c_2 + W)}{c_3 + W}, \text{ft/s} \tag{2.129}$$

The above three equations give the velocity inputs with bounding acceleration for shell, hull and deck-mounted equipment in surface ships. W is the modal weight of equipment-and-foundation system in *kips* (kilo-pounds). These basic values are modified for different ship locations and shock directions using different c_A and c_v as shown below.

$$V_a = c_v V_0 \tag{2.130}$$

$$A_a = c_A A_0 \tag{2.131}$$

Eqn. (*2.130*) represents limiting velocity in the constant velocity or low to mid frequency portion of the spectra. Eqn. (*2.131* represents the limiting acceleration in the constant acceleration or high frequency portion of the spectra. Displacement limits are also sometimes used for very low frequencies, however, its formulation is still being researched.

A critical frequency at the intersection of the acceleration and velocity limits is defined as:

$$\omega_c = \frac{A_0 g}{V_0} \tag{2.132}$$

For systems with $\omega > \omega_c$, the design spectra are acceleration limited. When $\omega < \omega_c$, the design spectra are velocity limited.

A hypothetical naval shock design input is shown in Figure 25. Here the critical frequency is about 90 Hz when the modal weight, W = 0.

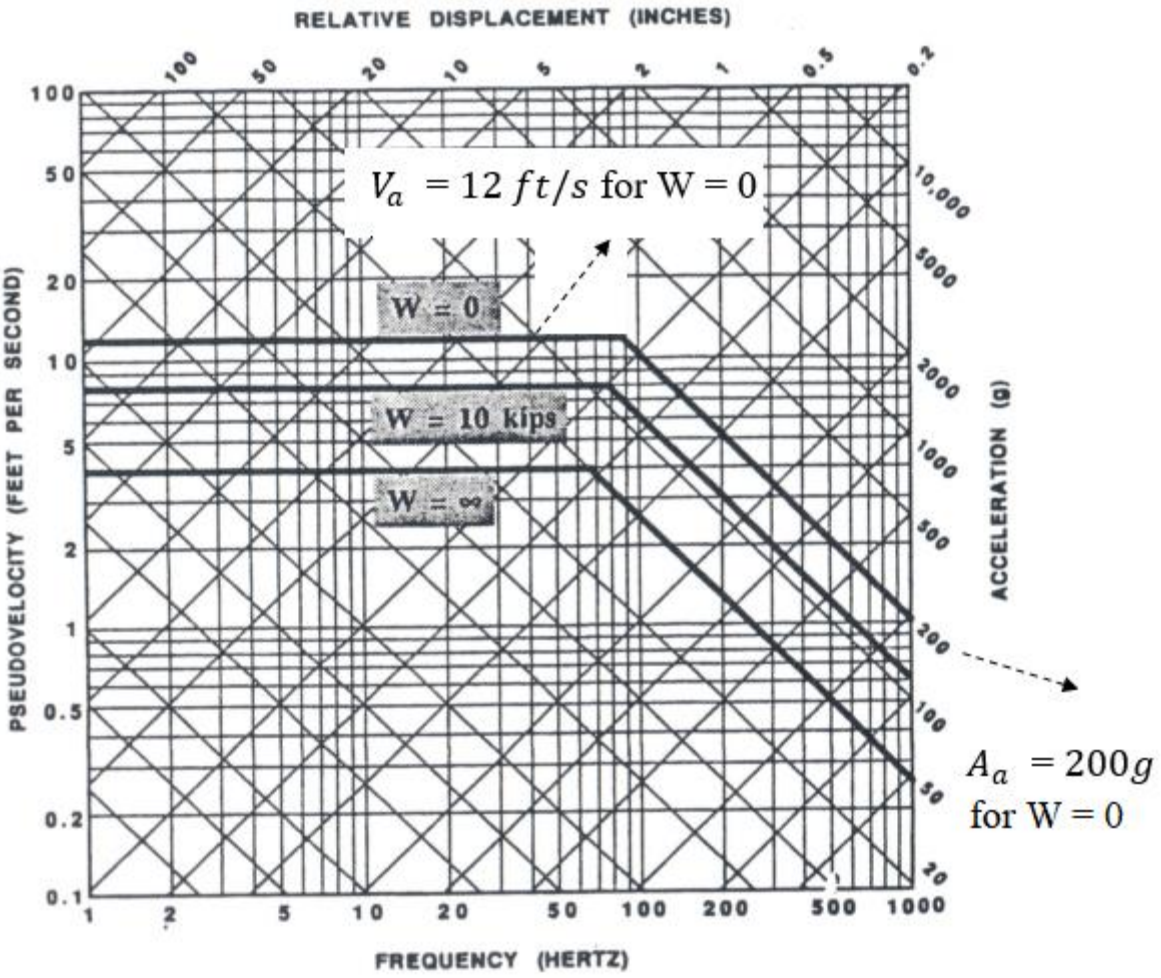

Figure 25:Hypothetical design shock input[18]

3 SIMPLIFIED FLUID STRUCTURE INTERACTION IMPLEMENTATION

This chapter presents a simplified shock model using a planar shock wave equation, Taylor-Plate theory, and Nastran with fluid/structure interaction. The initial Nastran model is developed in MAESTRO and exported to Nastran. The preliminary set-up details are explained with the help of a benchmark case of the Bleich-Sandler Plate Problem.

3.1 NASTRAN-OpenFSI Module

NASTRAN, as introduced in Section 1.5.2.2, is a common structural solver in the academic and industrial fields and selected for this research due to various features available, including the capability to interface with a fluid code. MAESTRO, which is used as a pre-processor is highly compatible with NASTRAN in terms of import and export options.

OpenFSI is a module implemented within the SOL 400 (Non-linear Transient) solution sequence in MSC NASTRAN that permits coupling of an external fluid code with the NASTRAN structural solver. SOL400 in MSC NASTRAN uses a second order implicit time integrator-based solution sequence. In the OpenFSI module, the external program calculates the nodal forces for each time-step and transfers them to NASTRAN (SOL 400) and in turn NASTRAN solves the structural nodal displacements and velocity vectors (translational and rotational) and sends them back for the next time-step [57]. This constitutes an "iteration" during a time-step. The data transfer occurs at the nodes on the "Wetted Surface", defined in the NASTRAN input deck. In OpenFSI terminology, "Wetted Node" refers to a node in the structural mesh that is in contact with the custom loads code, "Wetted Element" refers to an element in the structural mesh made up of nodes that are in contact with the custom loads code, and "Wetted Surface" refers to a set of elements put together forming a surface where the fluid is in contact with the solid structure. It should be noted that there may be dissimilarities between the meshing of the wetted elements defined on the structure and the fluid mesh defined in the external code. This requires mesh mapping to be done so that the information exchange is performed correctly. A spline method is one such mapping method [58]. A fluid mesh and mesh mapping is not required with Taylor Plate theory.

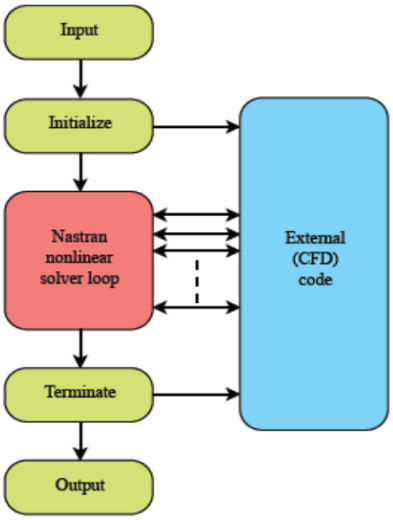

Figure 26: MSC Nastran OpenFSI Simulation basic flow chart [57]

The flowchart above shows the basic implementation of the OpenFSI module. In the SOL400 solution sequence, the solver may use explicit or implicit service. In explicit service, there is only one 'iteration' per time-step whereas in the implicit service there are multiple 'iterations' per time-step inside a Newton-Raphson loop. The application of Taylor Plate theory does not require an external fluid code. The Taylor Plate FSI equations are incorporated directly in the OpenFSI module. Figure 27 and Figure 28 show the flow chart for explicit and implicit service.

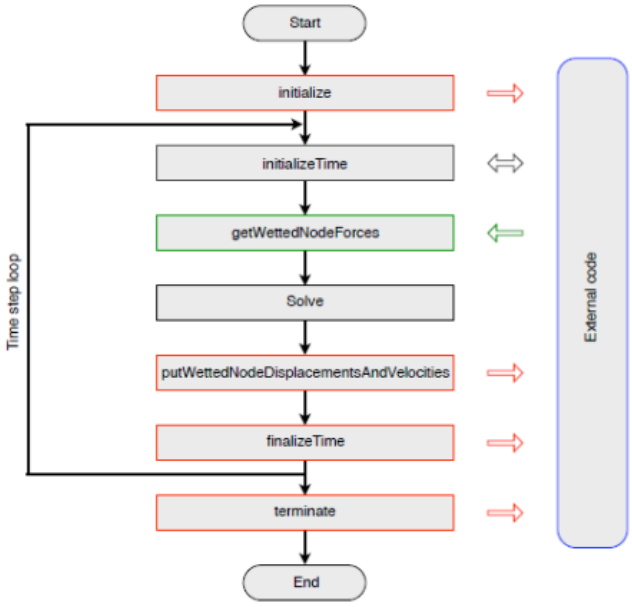

Figure 27: Flowchart of NASTRAN solver for explicit OpenFSI SCA service [57]

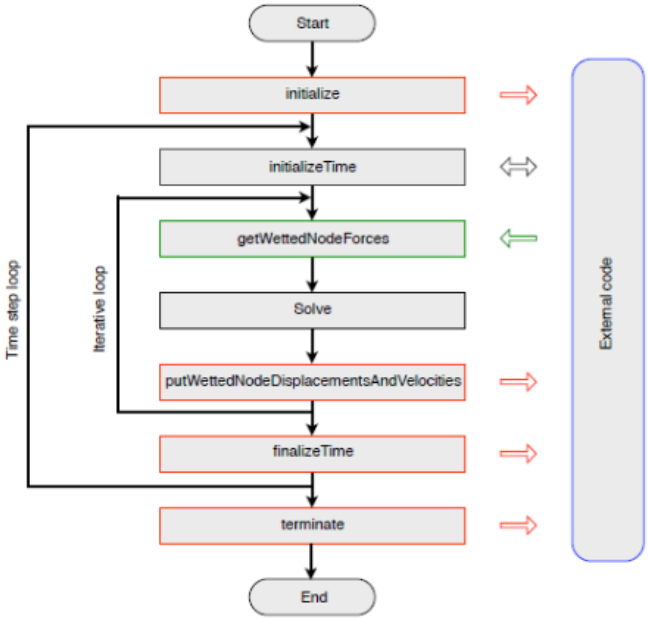

Figure 28: Flowchart of NASTRAN solver for implicit OpenFSI SCA service[57]

In this thesis, we use the OpenFSI explicit service formulation and one-to-one (no mapping). This is because the governing equations are very stable as there is no actual fluid modeled. Hence we follow the flowchart in Figure 27. The names in the flowchart are actually functions in the Openfsi.cpp, a C++ code which helps to link the fluid code with the Nastran structural solver. Further discussion of the C++ code layout and the integration with Nastran is discussed in APPENDIX A: NASTRAN-OPENFSI SET-UP. There is a limitation to the OpenFSI module in that it can handle only triangular or quadrilateral elements as wetted elements. Also, rotational DOF (degree of freedom) is not enabled in OpenFSI. The basic structure of the function call sequence for OpenFSI is as follows [57]:

- Initialization stage; sending wetted mesh data and initial condition:
  - initialize (wettedSurfaceMesh; initialCondition)
- Solver stage; sending and receiving solution data:
  - initializeTimeStep (time; deltaTime)
  - getWettedNodeForces (wettedNodeForces)
  - putWettedNodeDispAndVel (wettedNodeDisp; wettedNodeVel)
  - finalizeTimeStep (time)
- Finalize stage; sending termination signal:
  - Terminate (message)

3.1.1 Modeling consideration in MSC Nastran input file

The input file used for simulation is attached in APPENDIX C: NASTRAN INPUT FILE FOR FLOATING SHOCK PLATFORM with most of node definitions truncated.

File Management	CONNECT SERVICE
Executive Control	**Must be SOL NONLIN**
Case Control	DLOAD
OpenFSI Bulk Data	FSICTRL WETLOAD WETSURF WETELMG WETELME
GeneraL Bulk Data	DLOAD TLOAD1 TABLED1

Figure 29: Input File Entries Related to OpenFSI Simulations [57]

File Management	CONNECT SERVICE SCAFSI'ExternalCodeVendor.OpenFSI'
Executive Control	SOL NONLIN CEND
Case Control	SUBCCASE 1 Analysis = NLTRAN DLOAD - SID . . .
Bulk Data	BEGIN BULK . . . TLOAD1 SID WL1 T1 TABLED1 T1 LINEAR LINEAR 0.0 1.0 1.0e10 1.0 ENDT FSICTRL SCAFSI EXPLICIT 1 WETLOAD WL1 WS1 SCAFSI WETSURF WS1 surfname WE1 WE2 WE3 etc. WETELMG WE1 TYPE G1 G2 G3 G4 G5 G6 G7 G8 WETELME WE2 E1 SIDE . . .

Figure 30: MSC Nastran Input File Structure for Single Wetted Load WL1 and a Single Wetted Surface WS1 [57]

Figure *29* shows the relevant input file entries related to OpenFSI simulations and Figure *30* shows a typical structure of the Nastran input file. In the bulk data section, the entities FSICTRL, WETLOAD, WETSURF, WETELMG, and WETELME, are used for specifying the coupling with OpenFSI.

DLOAD Case Control is used to specify that there is a dynamic load to be applied. FSICTRL is used to specify that an Explicit OpenFSI service is used. TLOAD1 defines a time-dependent dynamic load. TABLED1 defines a tabular function for use in generating frequency-dependent and time-dependent dynamic loads [57]. There are primarily two loads applied on the structure: Gravity and the force transferred from the fluid assigned through two TLOAD1 commands. The forces transferred from the fluid are imposed on the structure by running the OpenFSI service. To specify the time-step, the TSTEPNL command is used. Based on a mesh size of 1ft and speed of sound of 1450m/s, the time step is imposed based on the CFL condition:

$$c\frac{\Delta t}{\Delta x} \leq 1 \quad \text{i.e.}$$
$$\Delta t \leq \frac{\Delta x}{c} = 2 \times 10^{-4} s \tag{3.1}$$

In the case of the FSP model, based on the time-scale under consideration (5ms to 20 ms) and the small time-step required to describe the shock pressure history properly in the fluid code as there is no sub-cycling, a time step of $2 \times 10^{-6}\ s$ is used.

3.2 Implementation of Taylor Flat Plate Equation in OpenFSI

This thesis describes a simplified fluid structure interaction model based on Taylor Flat Plate theory as discussed in Section 2.2. In this case, the code is implemented in the OpenFSI module and returns the forces exerted by the fluid and the forces exerted otherwise in the structural solver itself.

The one-dimensional plane wave equation derived in Section 2.1 is used as the incident pressure on the structure. Also to determine the time when the wave reaches a node in the structure from the stand-off point i.e. the reference point where t= 0, the Heaviside (unit step) function is implemented. The stand-off point in this thesis is the structural node most proximate to the explosive charge. Thus the incident pressure will be incorporated into the OpenFSI code in the following format:

$$P_{in}(x,t) = P_{max} e^{-\frac{f(t)}{\theta}} H(f(t)) \tag{3.2}$$

Here $f(t)$ is a function of time derived in Section 2.1:

$$f(t) = -(R - (fsiTime + abs(ChargeProximity/c)) * c) / c; \quad (3.3)$$

$ChargeProximity$ is the distance from the explosive to the stand-off point. Thus a$bs(ChargeProximity/c)$ is the time for the plane wave to reach the stand-off point. $fsiTime$ is the time series in the code starting at $t = 0$ when the shockwave is at the most proximate node from the charge. We need to do this as the nodal distance is measured from the explosive charge location. The variable, R is the distance from the node to the explosive. Thus the expression $f(t)$ becomes negative if the distance covered by shockwave is less than the distance to the node and will be of the form given by Eqn. (*3.2)* otherwise. Here $H(\,)$ is the Heaviside function and $f(t)$ is used to evaluate whether $H(\,)$ is zero or one.

It is assumed that all nodes in an element are in the same plane. To incorporate the normal vector, the cross-product of the 2 adjacent sides of the quadrilateral or triangular element is found so that the normal vector pointing outward is obtained. To obtain the angle between normal and the vector joining node to the charge, the corresponding dot product is taken to find the cosine.

Since the simulation accounts for the far-field early-time underwater explosion, the regions in the shadow region, behind the ship relative to the explosion, are not modeled to be hit by the shockwave. The non-shadow region condition for the incident shockwave is implemented by means of checking that the angle for applying Taylor flat plate theory is acute (angle$\leq 90°$) i.e. cosine of the angle is ≤ 0. The pressure calculation at a node also incorporates the local pressure cut-off model at the fluid interface by checking if the pressure is below the vapor pressure of water (taken to be equal to zero).

Once the pressure acting at each node is found using Taylor Flat plate theory, an averaging is done so that the forces acting on each node are found and thus transferred back into the MSC Nastran structural solver. Thus the pressure from all the nodes of an element are summed up and added together and then multiplied by area of the element. This force magnitude found is divided by the square of the number of sides of an element so that it is the average force acting on a particular node in the element. Using the connectivity matrix, the 'forces on the node' from each of the adjacent elements is found and added to obtain the actual force on the node at a particular time-step due to the shock wave. The forces due to buoyancy can also be added into this formulation although it affects all the wetted nodes rather than just the non-shadow region elements. The formulation can be written in a basic equation format as discussed below.

For a wetted element in the non-shadow region, which has exposure to both incident shockwave and buoyant forces, we assume that the incident pressure due to the shockwave on the 4 nodes in an element is p_i and the area of an element is A. In order to properly implement the pressure cut-off criteria, atmospheric pressure is added on both the top and bottom of the plate element (assuming that the plate is air-backed). Hence the total pressure at a node on plate (in the non-shadow area) from the fluid would include the total shock wave pressure from the Taylor flat plate theory along with the atmospheric pressure and hydrostatic pressure (given by depth,h) as derived in Eqn.(*2.57)*. It is assumed that the cut-off pressure of water is zero so the total pressure cannot be negative.

$$p_{total\_at\_one\_node} = \begin{cases} 2p_i(t) - \dfrac{\rho c V_s}{cos\alpha} + p_{atm} + \rho_w g h \; ; \; p_t > 0 \\ 0 \qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad ; p_t > 0 \end{cases} \tag{3.4}$$

The total pressure at a node on plate in the shadow area would be just the sum of the atmospheric pressure and the hydrostatic pressure.

$$p_{total\_at\_one\_node} = p_{atm} + \rho_w g h \tag{3.5}$$

Since we are done checking for the cut-off condition, the atmospheric pressure on the opposite side of the plate can be applied as it is not applied as pressure in the structural solver by default. Hence, when obtaining the total force on an element (say, 4-noded), we subtract the atmospheric pressure in the final step, which is the equivalent of applying it from the structural solver.

$$P_{total\_1element} = \sum_{i=1}^{4} P_{total\_at\_one\_node} \tag{3.6}$$

$$Force_{total\_one\_element} = \left(\left(P_{total_{1element}} - 4 \times p_{atm} \right) \times A \right) / 4 \tag{3.7}$$

This average force on an element, an average value based on this sum is assigned to the node.

$$Force_{total\_one\_node} = Force_{total\_one\_element} / 4 \tag{3.8}$$

This force found in Eqn. (*3.8)* is in the normal direction to the element and hence when the forces found from adjacent elements are assembled for a node, there should be a vector addition of forces. This is the basic framework of the calculation of forces in the OpenFSI C++ code.

3.2.1 OpenFSI code implementation

The base OpenFSI code created has a few parameters to be modified on a case-by-case basis such as:

- Stand-off distance of explosive from the origin.
- Depth at which the charge is located.
- Weight of the explosive charge.
- Half-breadth of the hull.
- Draft of the hull considered.
- Location of the pressure node where total pressure or incident pressure is to be plotted.

The OpenFSI code is compiled after updating the parameters and the environment variables are pointed to the location where the code is compiled and executables created.

The input file is created such that a file with ".xdb" extension is created which can be imported into FEMAP/PATRAN for the visualization of results.

3.3 CASE 3D implementation used for comparison.

The CASE method discussed in Section 2.3 was implemented in three-dimensions as an in-house fluid solver, written in C++, by Zhaokuan Lu. A partitioned FSI framework was created coupling the in-house CASE fluid code with ABAQUS or MSC NASTRAN using MpCCI. MpCCI is a multiphysics coupling software which supports most state-of-the-art commercial solvers and in-house codes. It enables data exchange between the meshes of 2 different solvers, thus acting as a multiphysics interface. The results of two of the main cases discussed in this thesis are verified using this CASE 3D implementation as it provides a high fidelity solution for cases involving shockwave incidence, local cavitation and/or bulk cavitation.

The CASE-3D code written in C++ is linked to the MpCCI environment through several specific API functions. This set-up is coupled with any structural software supported by MpCCI such as ABAQUS or NASTRAN. In case of ABAQUS, the coupling will look like Figure 31.

In the CASE-3D method, the structural time-step size is typically smaller than the fluid and hence a monotone cubic spline interpolation is to be done between the fluid and the structure code implemented by MpCCI. The structural time-step is kept the same to isolate the effect of the fluid.

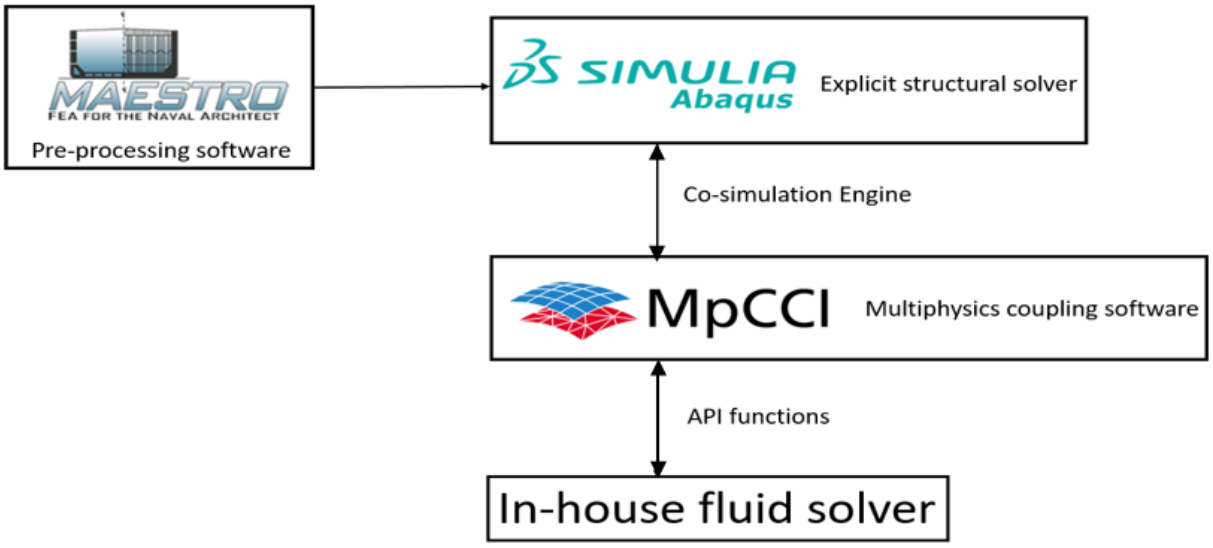

Figure 31: Partitioned FSI framework using ABAQUS and MpCCI

3.4 Bleich- Sandler Benchmark Case

The first case that is used to demonstrate this framework is the benchmark problem given in a paper by Bleich-Sandler [49]. It consists of a mass which can be modeled as a rigid plate placed on the free surface above a fluid column that is infinitely deep. This problem is analytically correct, but is not physically possible. For static equilibrium before application of forces due to the shock wave, we assume that the plate is placed at a draft below the water-level so as to apply a hydrostatic pressure and balance the weight of the plate although the plate is air-backed. The case set-up is shown in Figure *32*. The FEMAP model of the plate is shown in *Figure 33*.

3.4.1 Bleich-Sandler Case Description

The objective of this case is to simulate a moving plate mass and hence the plate is unrestrained and acts like a rigid body. A four-element plate panel is modeled in Nastran and the Young's modulus is set to 10^6 times more than the actual value to model a rigid body motion. The wetted surface includes all the elements of the plate. The node arrangement is done with the normal vector pointing upwards, but the OpenFSI code considers the outward/downward normal vector to calculate the angle in case of oblique incidence. The force of gravity acting on the plate is implemented by means of specifying the density of the material along with the dimensions and gravitational force in the input deck of MSC Nastran.

The mass of the plate in this configuration is 144 kg. The plate dimensions are $1\ m \times 1m$. This gives the draft of the plate as $T = 0.141\ m$ as the water density is assumed to be $\rho_w = 989\ kg/m^3$. The thickness of the plate is based on the mass, the draft and the density of the plate ($7850\ kg/m^3$) and is $t_p = 0.0018\ m$. The loading is considered using an incident shockwave with a peak pressure of $0.712\ MPa$ and a decay constant of $0.999\ ms$. This corresponds to a charge weight, $W = 121.877\ kg$ at a depth $R = 222.53\ m$. In the paper [49], the method of characteristics is used to solve the cavitation problem with an actual fluid model.

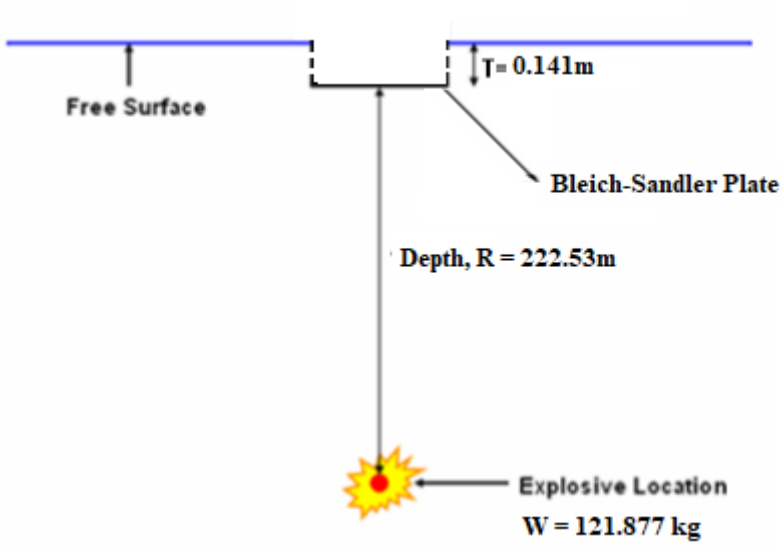

Figure 32: Bleich-Sandler Case Geometry

3.4.2 Results and plots

The original paper [49] considers the effect of shock loading on the plate with and without cavitation in the fluid. However, the Taylor flat plate theory is only able to model the effects of the pressure cut-off if that happens at the interface on the plate. It does not explicitly consider the fluid.

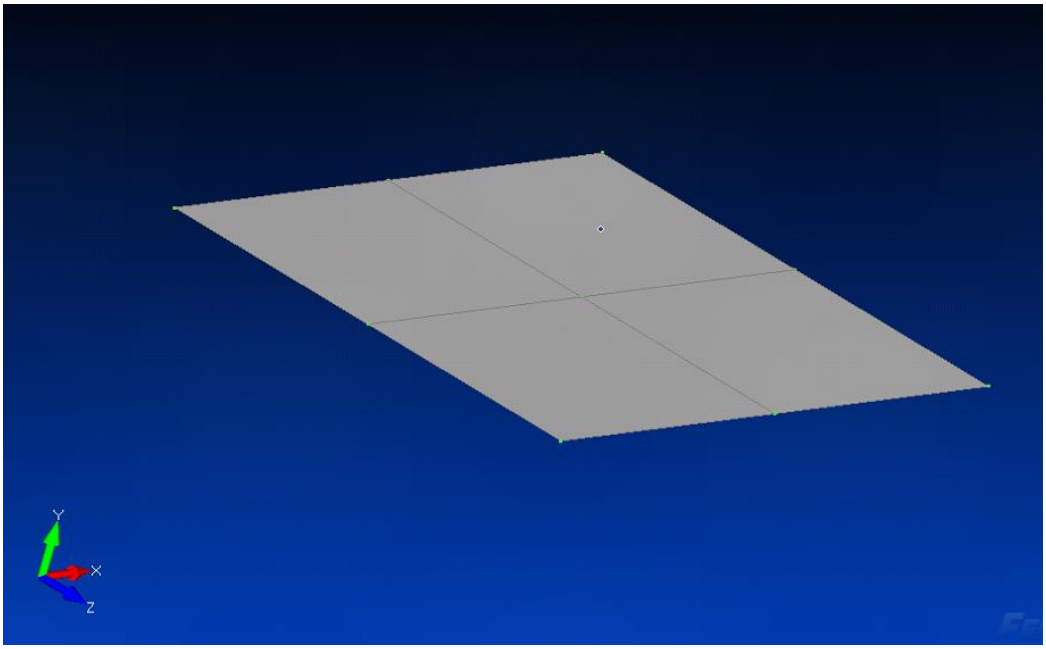

Figure 33: Bleich-Sandler case plate model

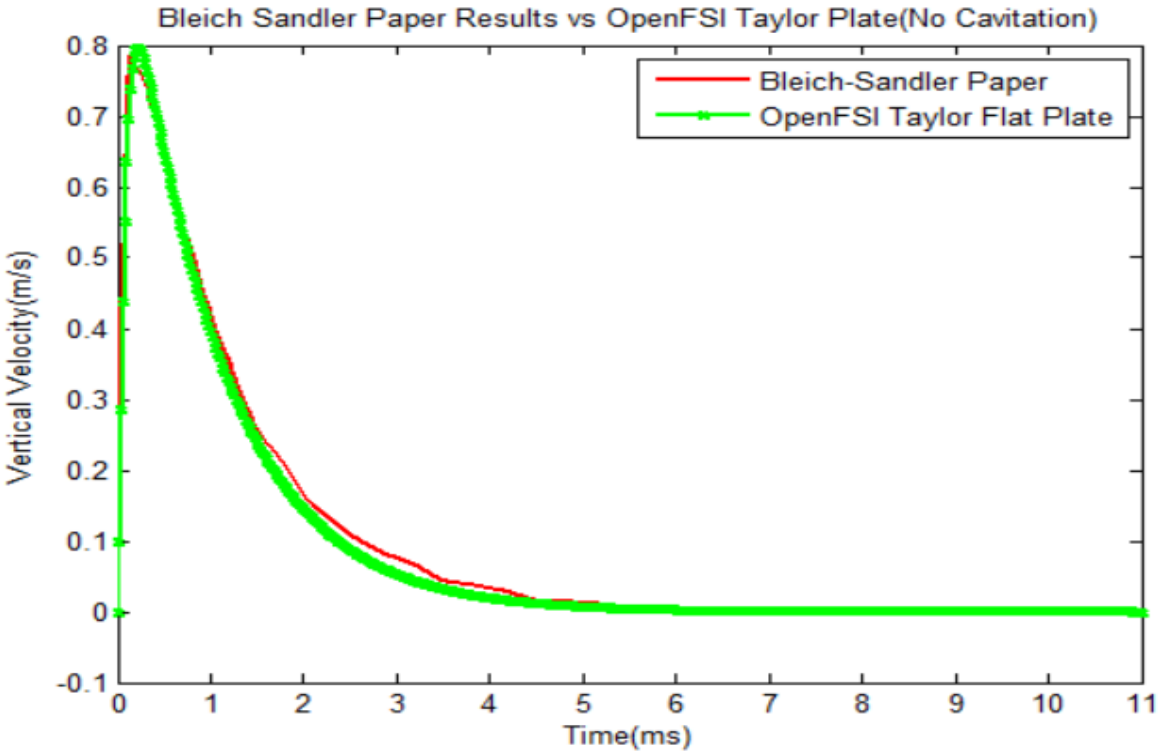

Figure 34:Vertical velocity of plate from Bleich-Sandler paper vs. Taylor flat plate results if no cavitation is considered in either model

Figure 34 shows the vertical velocity of the plate due to the shockwave excitation if cavitation is not considered. The output from OpenFSI Taylor Flat Plate theory matches almost exactly with the result from Bleich Sandler model and CASE-3D model as the pressure is modeled correctly at the interface in both cases. Using the Taylor Flat Plate formulation in Eqn.(*3.4)*, the variation of velocity can be understood. As the shock wave impacts the plate at t=0, the incident pressure is maximum followed by an exponential decay. The upward velocity (positive) of the plate also modifies the total pressure on the plate. Thus, as time passes, the incident shockwave pressure decays and the velocity reaches the peak or the kick-off velocity. This is when the total pressure reaches a minimum but its value is still greater than the vapor pressure of water ($0\ MPa$) and after which the velocity reduces due to the low pressure which causes the force to be in the re-bound direction. This causes total pressure on the plate from the fluid to be raised back to a total of atmospheric pressure and hydrostatic pressure as the incident wave reduces completely to zero. As this happens, the plate velocity also reduces to zero as there is no longer an acoustic impedance due to shock wave and the gravity and the atmospheric pressure on top of the plate is balanced by the hydrostatic pressure and atmospheric pressure on the bottom of the plate. As the plate comes back down, there is a very small scale oscillation in the Taylor plate model as the velocity reduces to zero. This happens due to the fact that there is a balancing between the forces due to hydrostatic pressure and the gravity as the draft is being adjusted but this occurs over a larger time scale compared to shockwave phenomenon considered.

Figure 36 shows the results from the Bleich-Sandler model, CASE-3D model, and Taylor Flat plate theory model with the implementation of a pressure cut-off in the fluid if the total pressure goes below the vapor pressure of water. In this case, the Bleich-Sandler paper results and the CASE-3D results model the fluid domain correctly and the pressure at the interface is modified by a cavitation region formed inside the fluid column whereas the Taylor Flat Plate theory is unable to capture that as it is based on the variation of pressure at fluid-structure interface only. *Figure 35* shows the cavitated region formed in the fluid column below the plate and it can be seen that it never reaches the plate ($x = 0$) during the time scale considered.

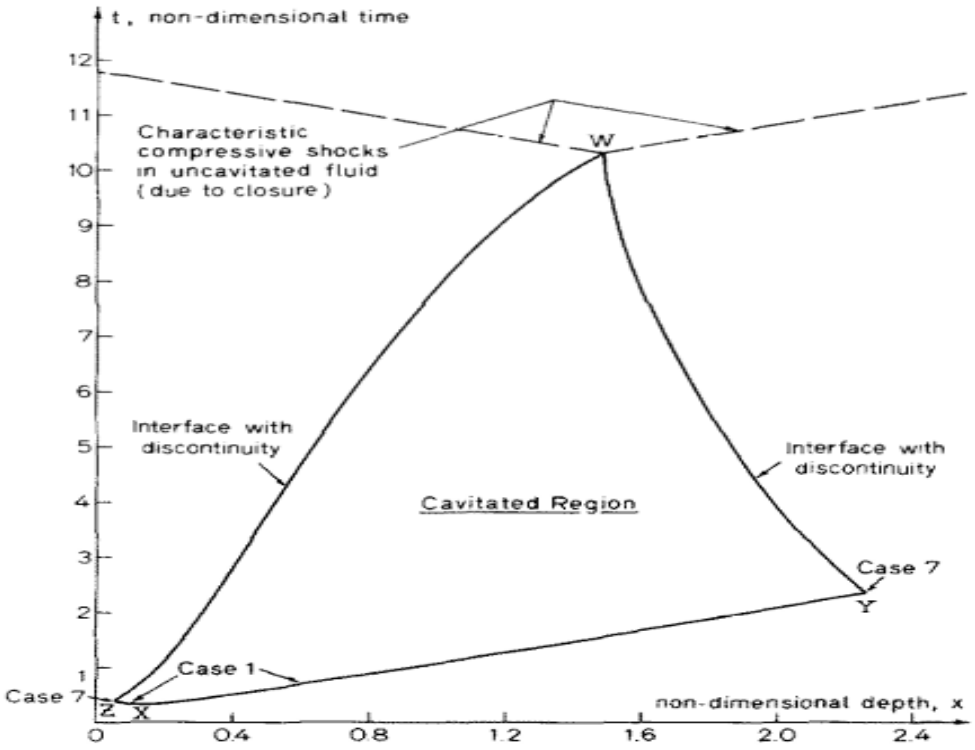

Figure 35:Cavitated region in the fluid as given in the Bleich-Sandler Paper[49]

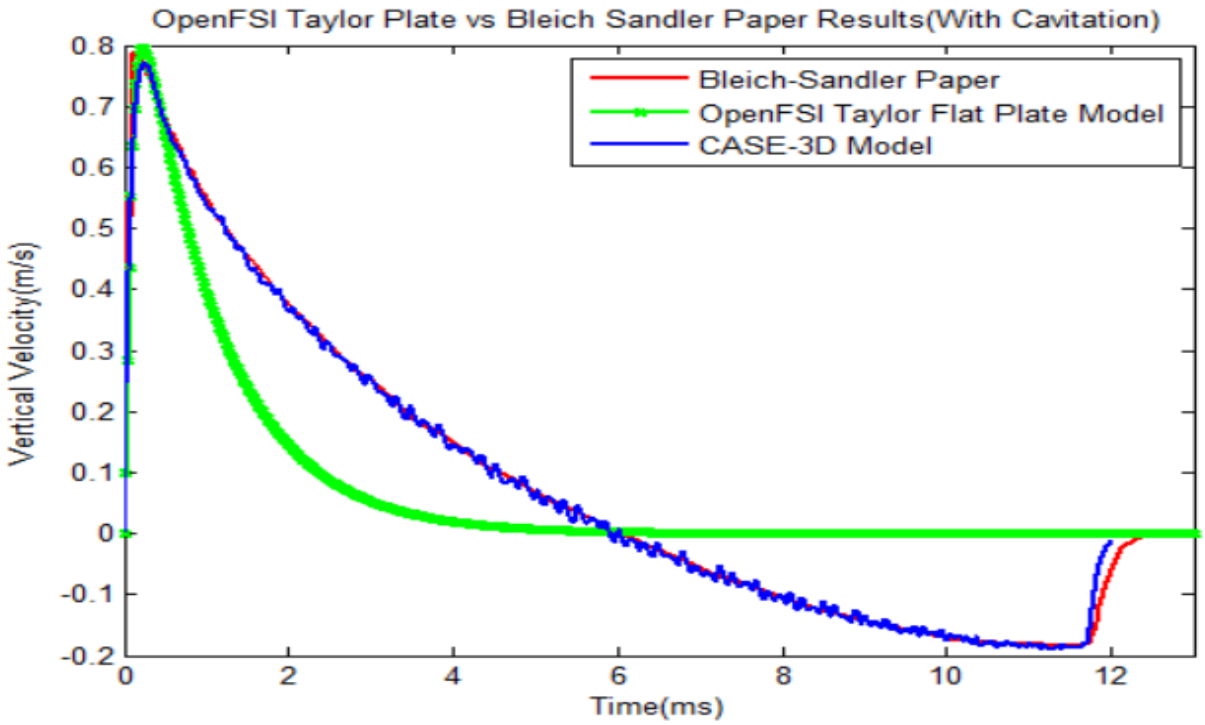

Figure 36: Vertical velocity of plate from Bleich-Sandler paper vs. Taylor flat plate results if cavitation is considered

This explains the difference of the velocity reduction trend in CASE-3D/Bleich-Sandler Method as compared to Taylor Flat Plate model. As time passes (t=12ms), the cavitation region in the fluid closes causing it to send out a closure pulse and a sudden rise in pressure at the interface resulting in the rebound of the plate (increase in velocity) in the upward direction which is captured only by CASE-3D and Bleich-Sandler paper and not by the Taylor Flat plate formulation. The plate returns to its original location. The CASE-3D model, being a numerical model is also affected by very small spurious oscillations compared to Bleich-Sandler results.

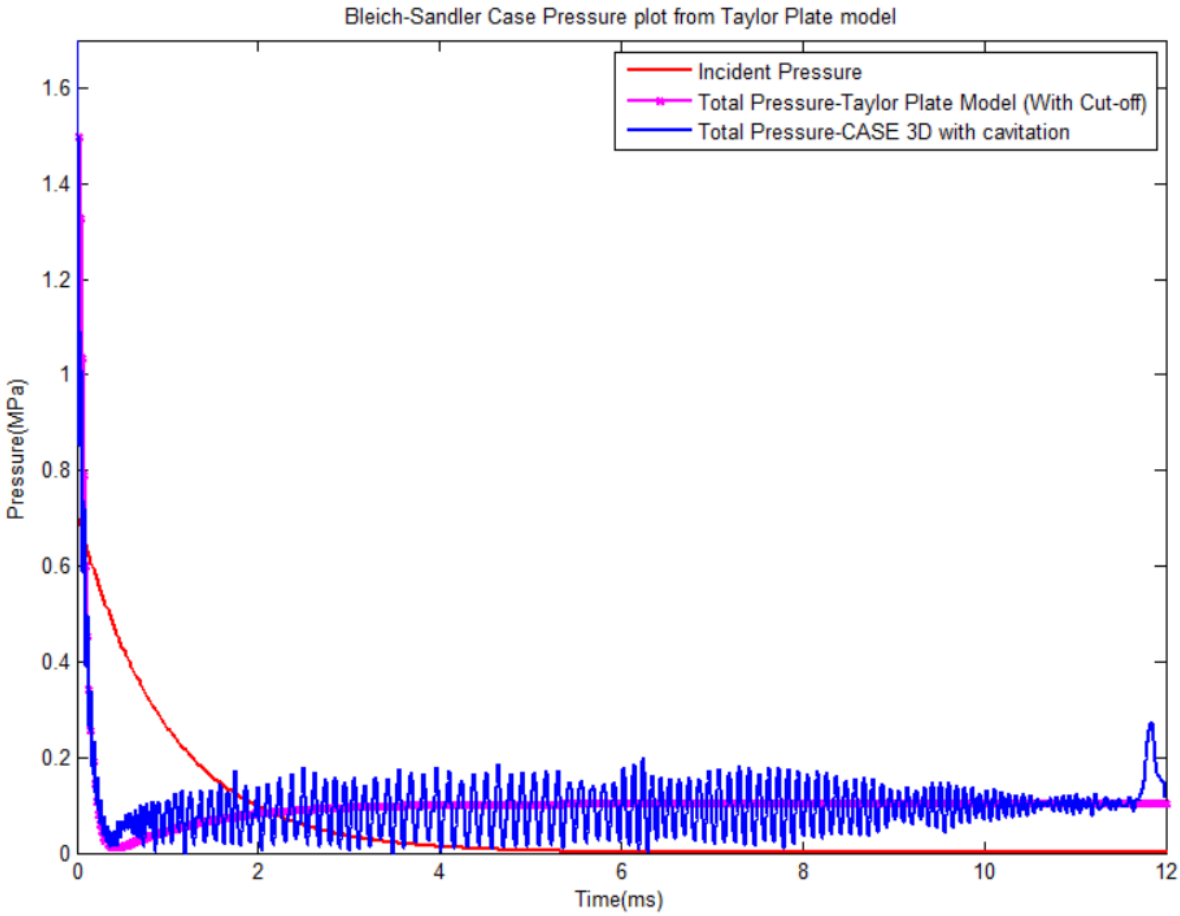

Figure 37: Incident and total pressure for the Bleich-Sandler paper case using Taylor flat plate theory.

The pressure plot in *Figure* 37 shows that the total pressure on the plate predicted by the Taylor plate model does not go negative in the Bleich-Sandler case just as in the CASE-3D model. The sum of shockwave pressure, atmospheric pressure, and hydrostatic pressure remains positive for the duration of interest (12 milliseconds).

To see if the pressure cut-off has the correct effect in the Taylor flat plate theory model, we select a mass with 10 times that of the Bleich Sandler case to insure a negative plate pressure and test the results with and without pressure cut-off. Since the mass has changed, the draft of the plate also changes to $1.429\ m$.

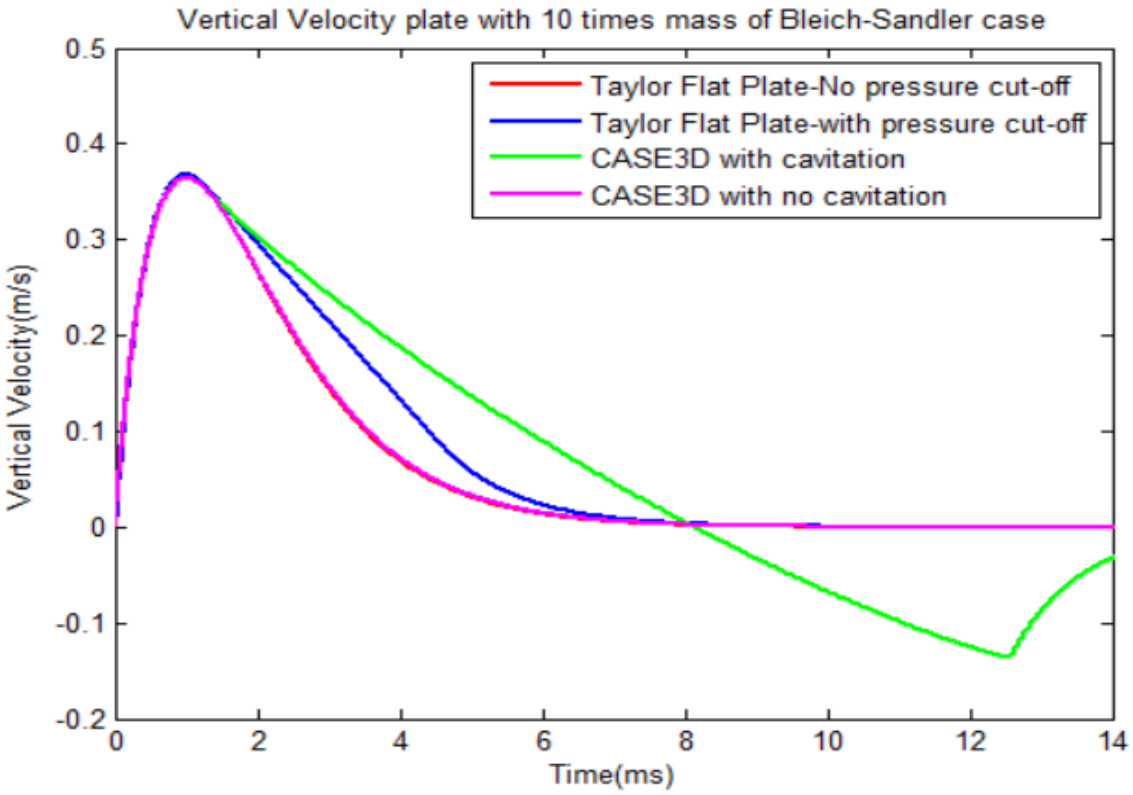

Figure 38: Vertical velocity of a rigid plate (Mass= 1440 kg) using Taylor flat plate theory to check influence of pressure cut-off.

The pressure plot in Figure 39 shows that if pressure cut-off is not enabled, the higher mass configuration will cause a negative pressure at the fluid-structure interface using the Taylor Plate theory formulation. Figure 39 also shows that the pressure variation is the same in both CASE-3D without cavitation and Taylor Flat Plate theory model with no cutoff.

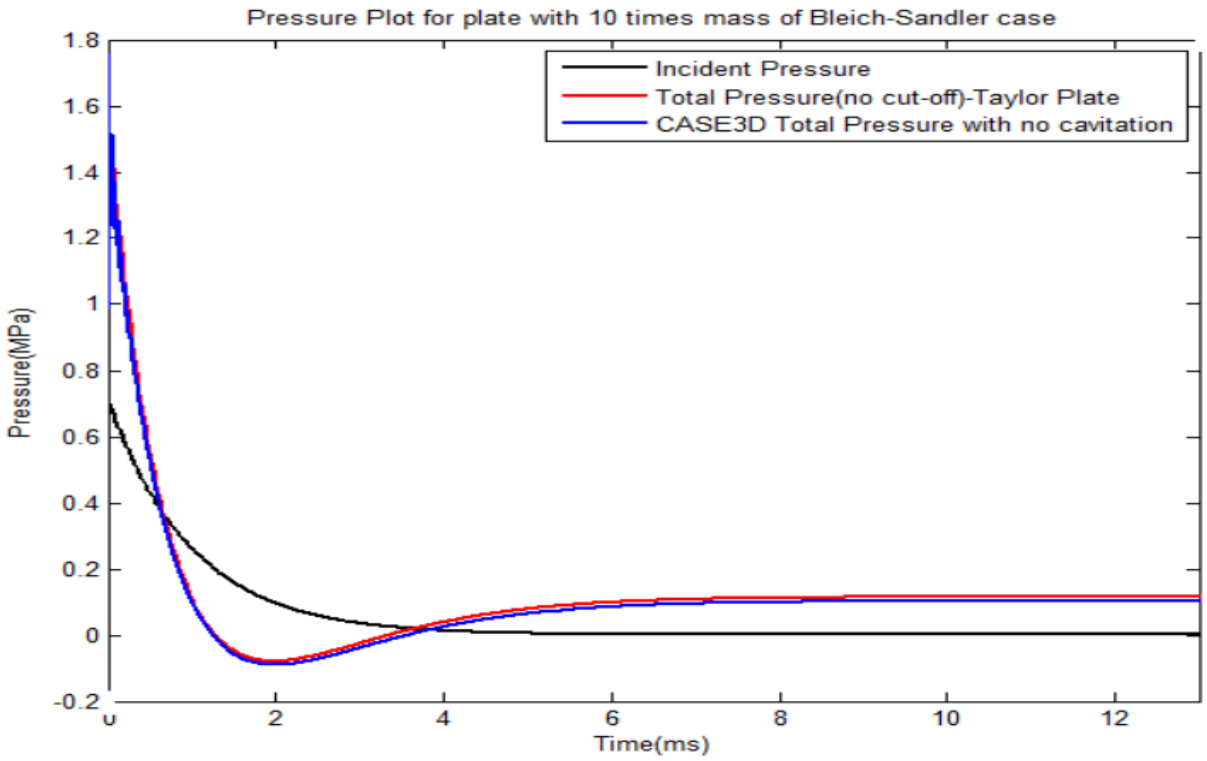

Figure 39: Incident and total pressure for the plate of mass = 1440 kg with no Pressure cut-off cavitation modeled

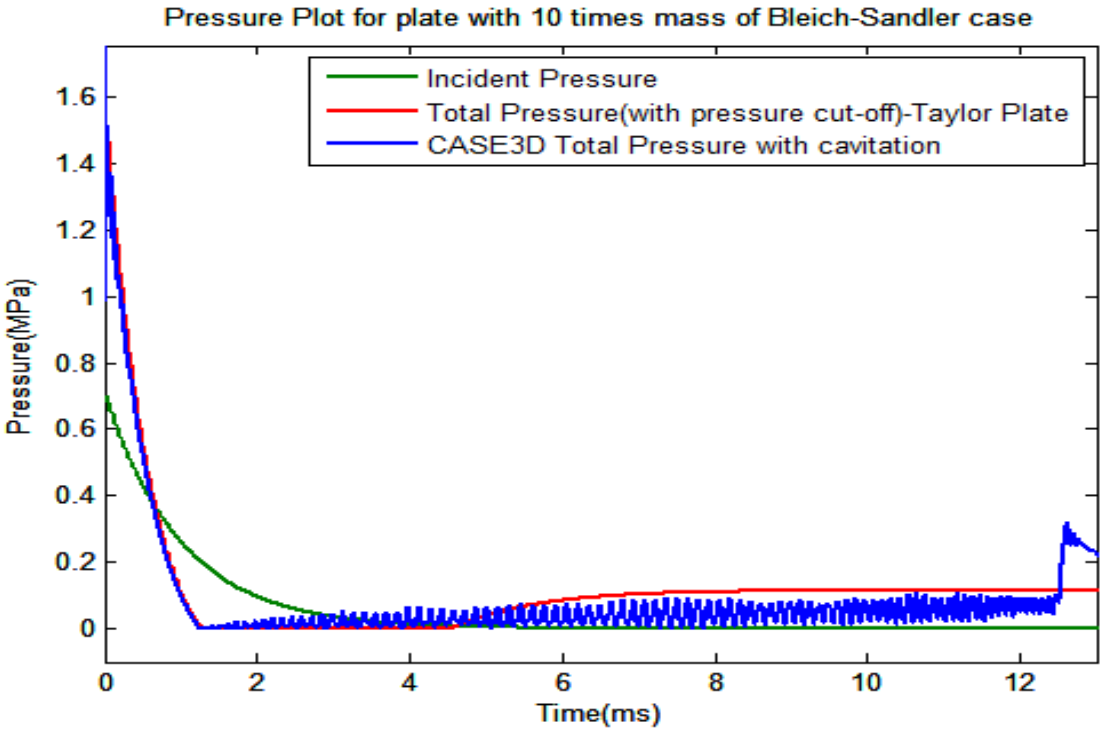

Figure 40: Incident and total pressure for the plate of mass = 1440 kg with Pressure cut-off cavitation modeled

When the pressure cut-off is implemented in the Taylor Plate model and cavitation is implemented in CASE, the Taylor plate model undergoes a reduction in velocity less drastically (lower slope) compared to the set-up without cavitation as shown in Figure 38. However, it is still not able to match the slope of CASE-3D model as it is not modeling the actual fluid where the cavitation region in the fluid also affects the pressure at the interface. Figure 38 and Figure 40 also show that the Taylor plate model is not able to capture the re-loading.

Thus in this case, where the Bleich-Sandler plate configuration with 10 times the original mass is analyzed, the key trends remain the same in terms of the velocity and the pressure plots compared to the Bleich-Sandler case making it an easy problem to verify the configuration for pressure cut-off implementation.

3.5 Shell Case

The model shown in Figure 41 is an extension of the Bleich-Sandler case and a simplified version of the floating shock platform in that it is a dish-like model where equipment can be mounted. This provides a transition between the two cases and a simple verification of the 3D code implementation. The results of this simulation using OpenFSI are compared to CASE-3D simulation results.

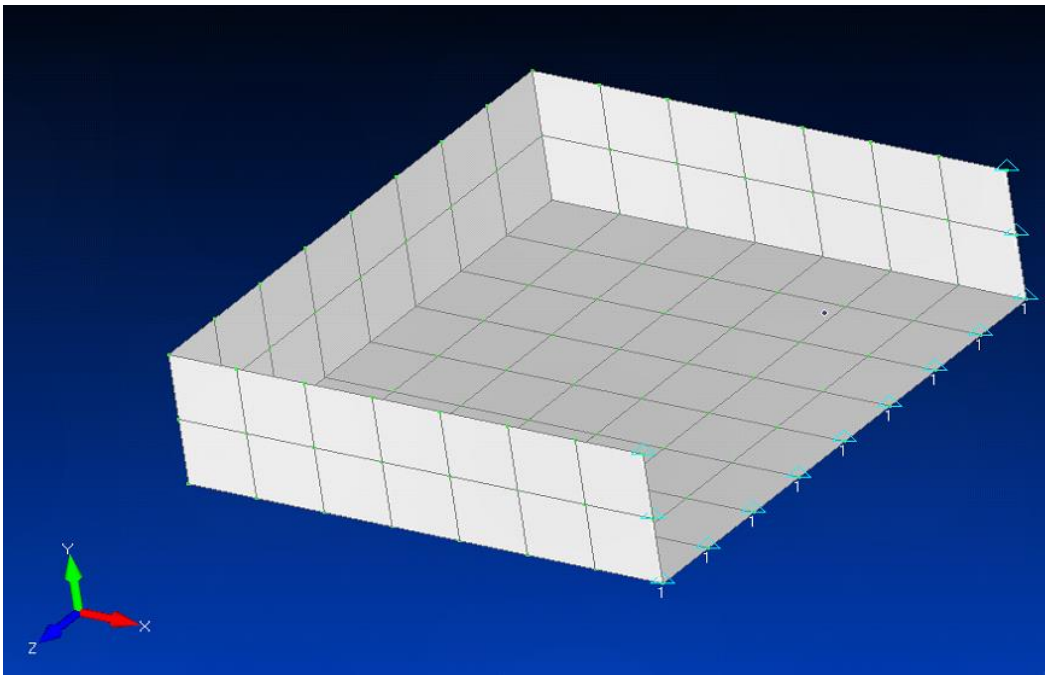

Figure 41:Shell case as modeled in FEMAP/NASTRAN

3.5.1 Case Setup

This case is similar to the Bleich-Sandler model in the previous section in terms of the explosive charge location and weight as shown in Figure 42. The draft of the model is 0.2032m. It should be noted that the model is also non-physical as the bottom is the only wetted surface even though it is assumed to have a specific draft. The structure is a half-model with a symmetry plane implemented by imposing horizontal constraints on the half-section in the longitudinal direction. There are no stiffeners in the model. It consists of just the shell plating of thickness 0.02m (20mm). The dimensions of the shell full model are $1.4224\ m \times 0.8128\ m \times 0.2032\ m$ (which is defined based a ratio of the FSP dimensions.

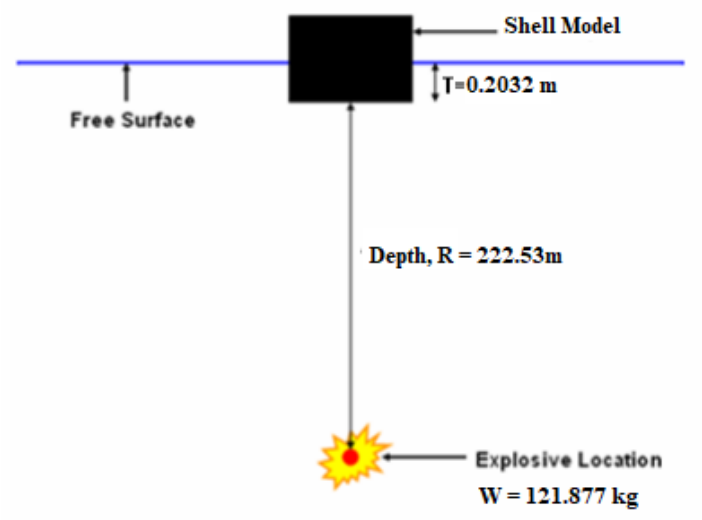

Figure 42:Shell Case set-up

3.5.2 Shell Case Results

The shell is modeled as a rigid body by using an artificially high Young's modulus of $2 \times 10^{20} MPa$. The results are compared to CASE-3D model results as shown in Figure 43 below. Just like the Bleich-Sandler case, the peak or the kick-off velocity and the rise-time are captured well. However, the velocity reduction due to cavitation effects and the rebound are not captured by the Taylor Flat Plate theory model.

Next the shell is modeled as a deformable body with the Young's modulus of steel ($2 \times 10^{11}\ MPa$) and vertical velocity values are compared to the CASE-3D model results as shown in Figure 44. It can be seen that the kick-off velocity and the initial trend corresponding to the first 2 ms is captured well. However, the fluid cavitation effects dominate after the initial peak velocity and the Taylor Plate model no longer matches the results from CASE-3D. Both plots show

oscillation resulting from having a deformable body, and both follow the same general trend, but with more ups and downs in the CASE-3D plot.

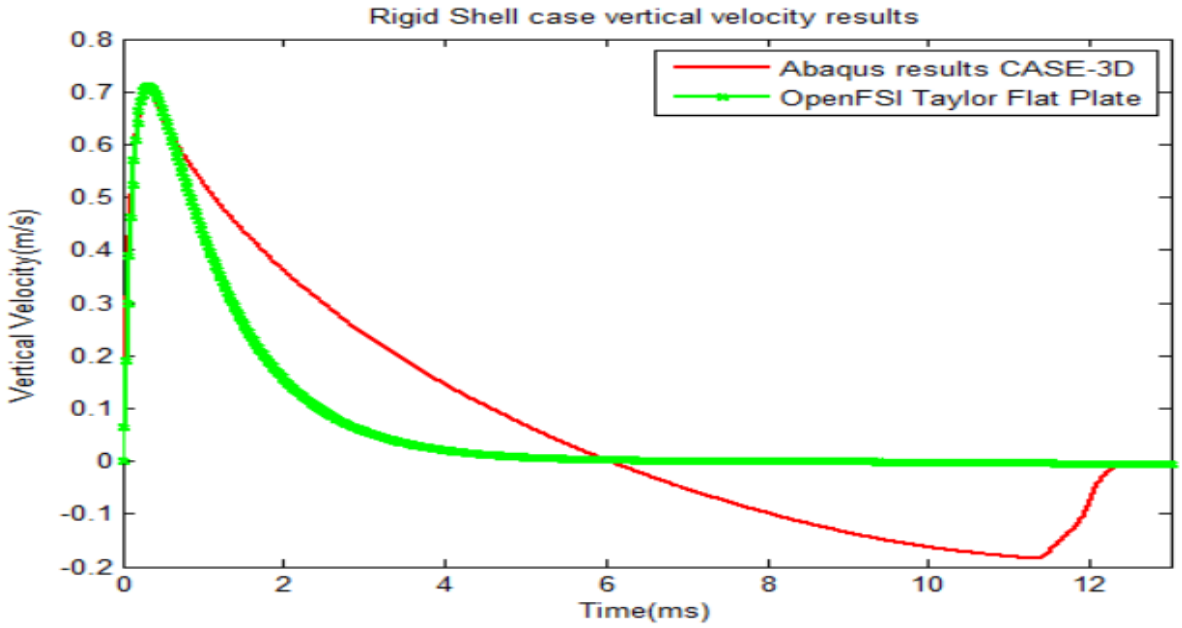

Figure 43:Vertical velocity of node 85 in bottom plating with shell as a rigid body using OpenFSI (Taylor Flat Plate) and CASE-3D

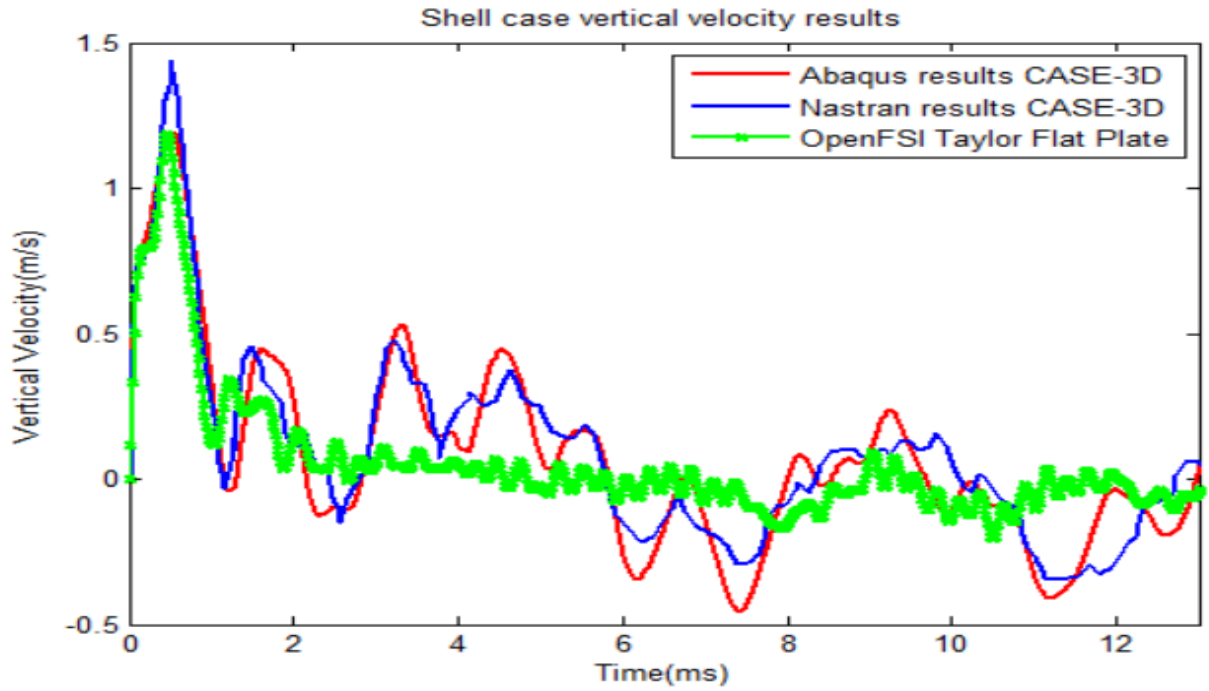

Figure 44: Vertical velocity of node 85 in bottom plating on a deformable shell (Young's Modulus: 2E11 MPa) using OpenFSI(Taylor Flat Plate) and CASE-3D

The pressure plots shown in Figure 45 give insight into the pressure variation in the Taylor Flat Plate theory model compared to the CASE-3D model. The total pressure starts out on a similar trend, however, the CASE-3D model, having modeled the fluid domain, has its pressure curve affected much earlier than Taylor Flat Plate model due to the onset of cavitation compared to the simple Taylor Plate pressure cut-off. This also reflects the variation in their velocity profiles after the initial peak shown in Figure 44.

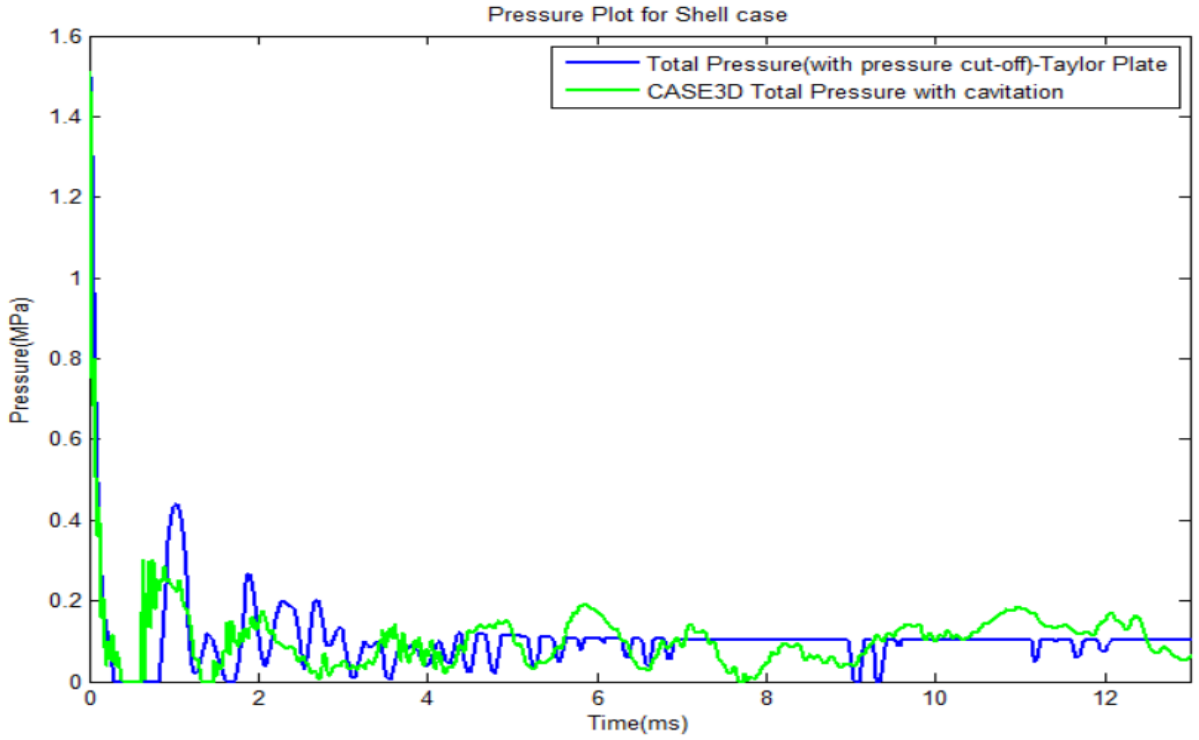

Figure 45:Total pressure plot of the Shell model node 85 from Taylor Flat Plate set-up and CASE-3D set-up

For the design of structures or equipment or for their placement, the shock spectrum is also important because the excitation frequency content will drive the response and hence the spectra obtained from the two models are compared. Figure 46 is the plot of the acceleration shock spectra and Figure 47 is the corresponding pseudo velocity spectra plot generated using [54]. The shock spectrum results from Taylor Flat Plate theory model and CASE-3D model match pretty well suggesting the former may be a sufficient design tool for UNDEX at least in early stage design. In the next chapter, we discuss the final case of the floating shock platform and discuss its results.

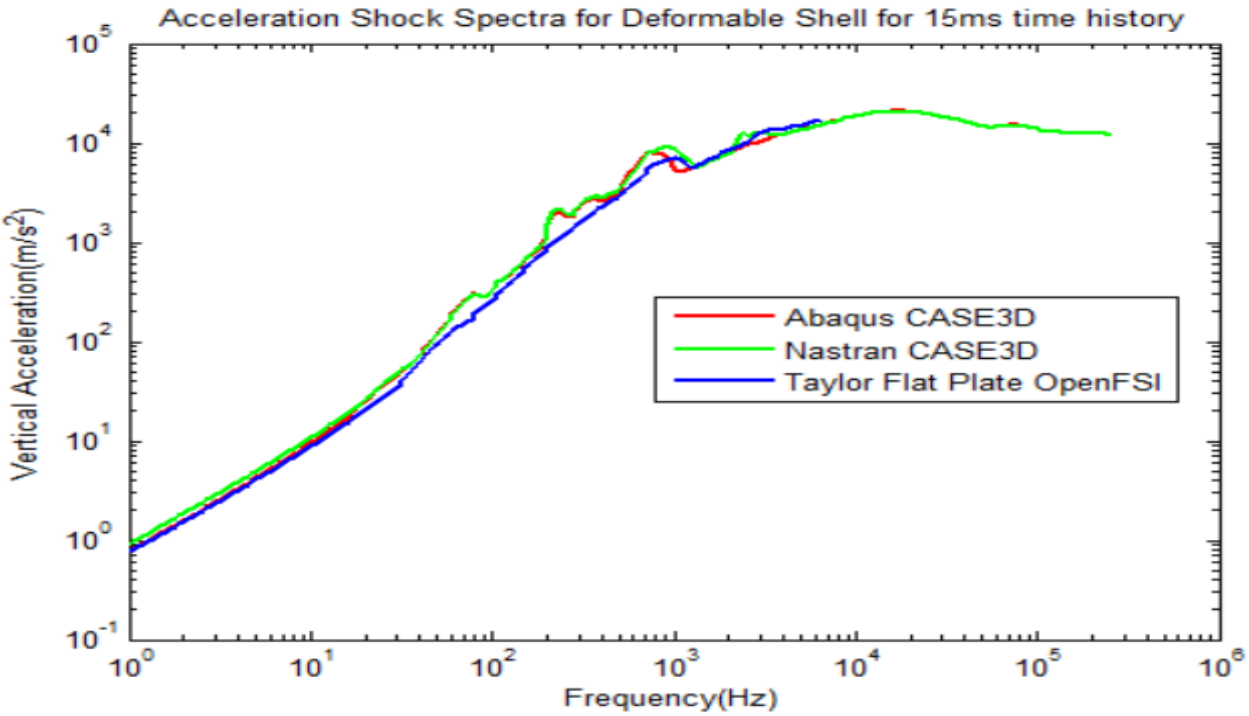

Figure 46:Acceleration Shock spectra obtained from deformable shell case using Taylor Flat Plate theory and CASE-3D

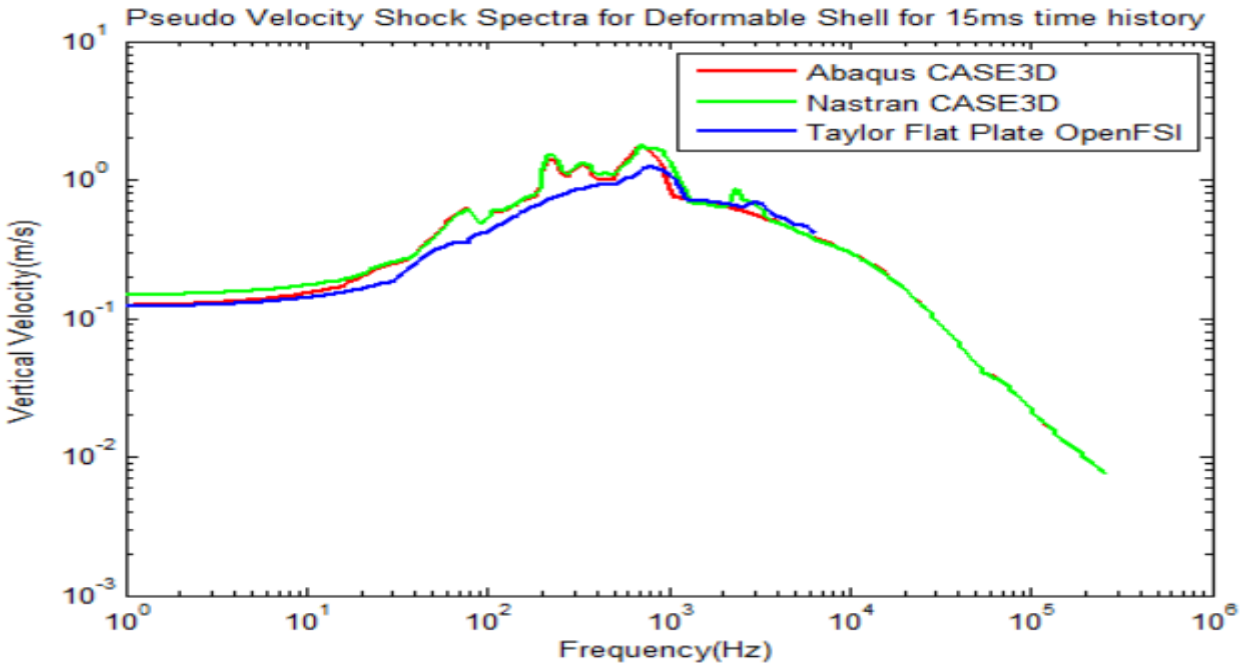

Figure 47: Pseudo velocity Shock spectra obtained from deformable shell case using Taylor flat plate theory and CASE-3D

4 FLOATING SHOCK PLATFORM TEST CASE AND RESULTS

This chapter shows how the OpenFSI code development described in Chapter 3 is extended and applied to a Floating Shock Platform (FSP), an important tool in equipment design for shock during underwater explosions. The structural details of the FSP are discussed followed by loading details. The typical testing locations and their relevance are also discussed.

4.1 Structural details and Case set-up for the FSP

The FSP model is created in MAESTRO and imported to Nastran to incorporate the input parameters relevant for OpenFSI. The model considered here corresponds to the Standard Floating Shock Platform as referred to in [59, 60] and has the following specifications:

- Length: 28 feet.
- Breadth: 16 feet
- Depth (Height) until canopy:11 feet
- Draft: 4 feet

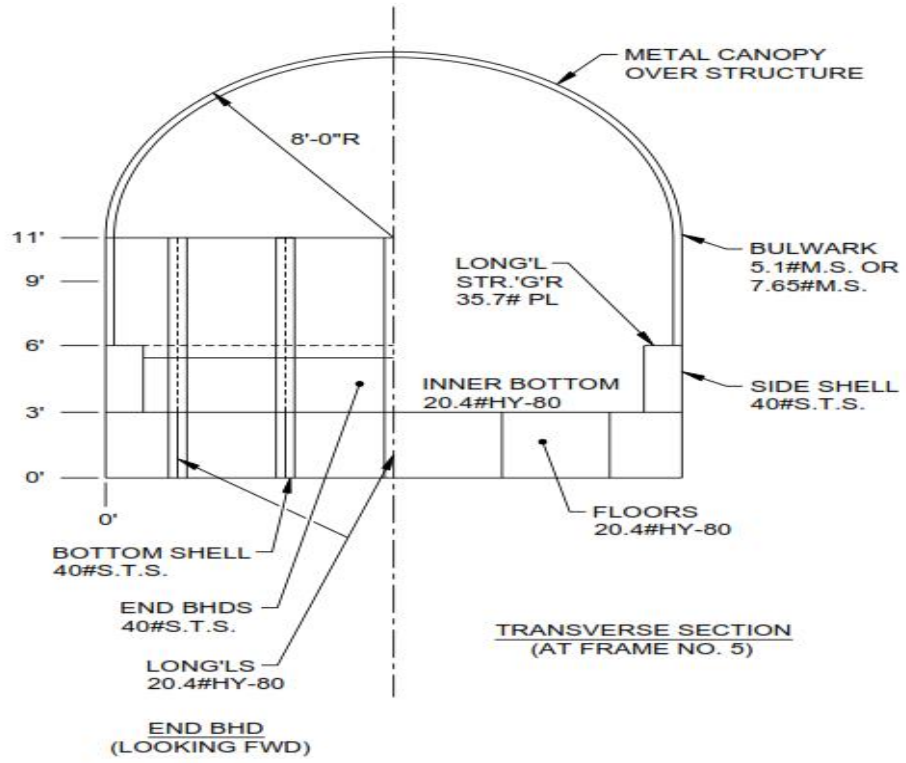

Figure 48: Transverse section view of the Standard Floating Shock Platform

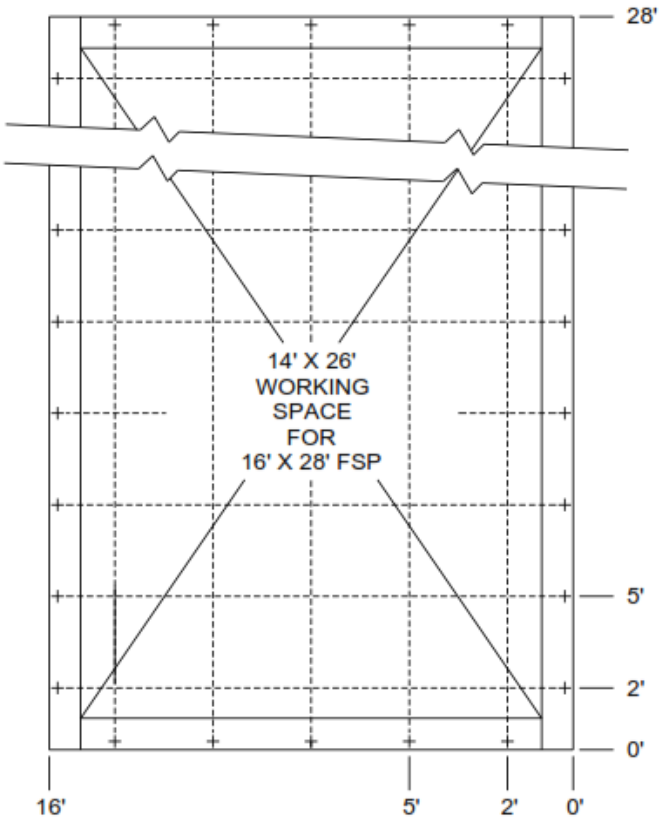

Figure 49:Plan view of the Standard Floating Shock Platform

The FSP plating/stiffener configuration is also shown in *Figure 48* and Figure *49*. The loaded draft is 4 feet which corresponds to a displacement of 114,668lbs for the 28m long FSP. The charge is located at a stand-off distance of 10 feet and at a depth of 30 feet. This is the base configuration for which the tests are done as shown in *Figure 50*. The frame spacing is 3 feet (0.9144 m) except for the spacing of 2 feet (0.6096 m) between the side-shell and the first floor. The preliminary mesh size used in our model is 1 foot. The FEMAP/MAESTRO model is shown in *Figure* 51.

The outputs are measured at 3 important locations.

- Intersection of stiffener/girder in bottom plating.
- Mid-point of plate element in bottom plating.
- Intersection of stiffener/girder in inner-bottom plating.

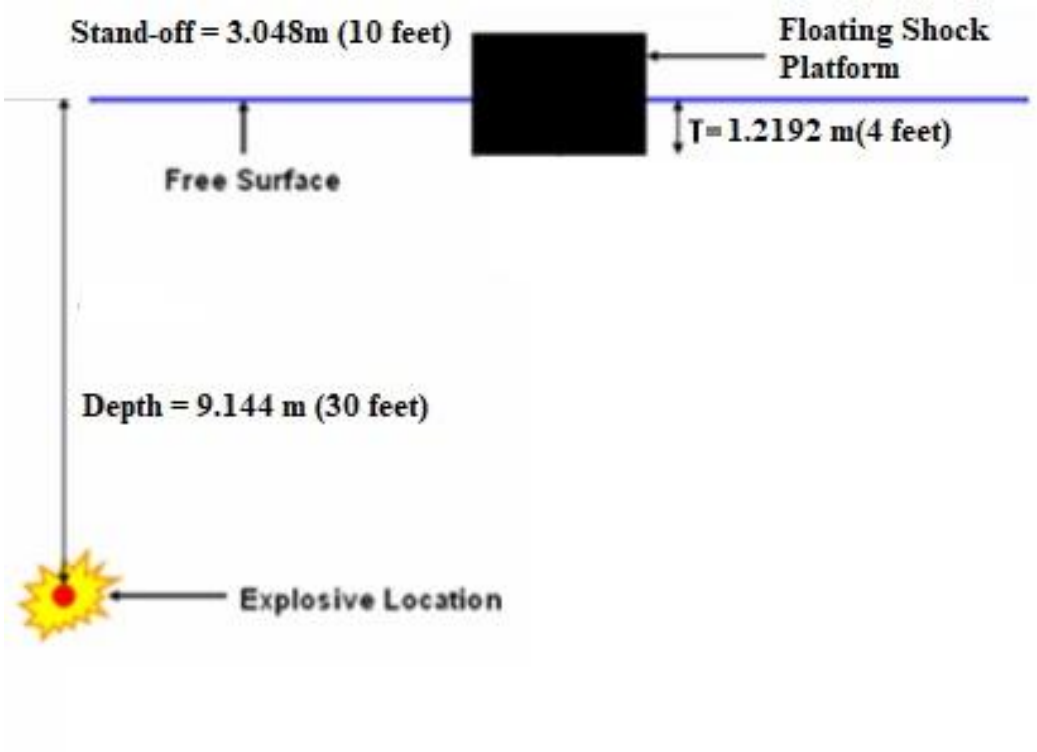

Figure 50:Case set-up for the Floating Shock Platform

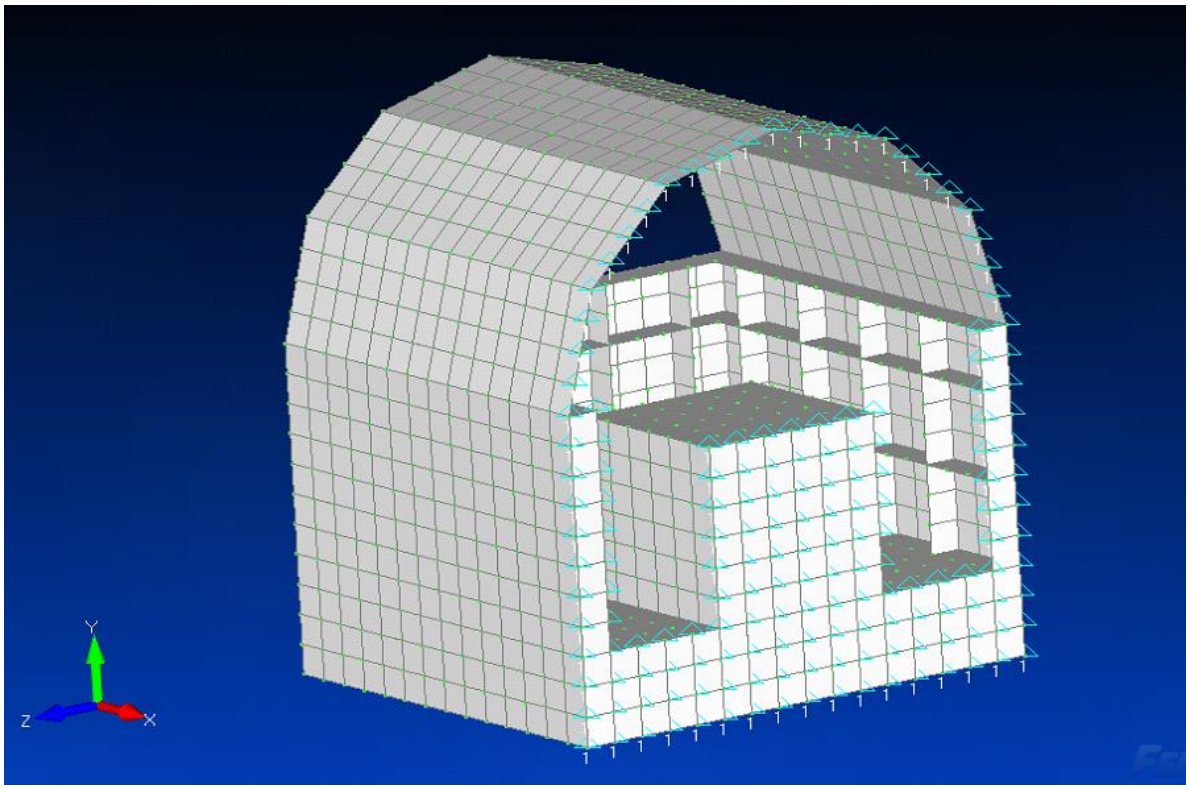

Figure 51: Model of Floating Shock Platform (FSP)

4.1.1 Modeling Considerations in MAESTRO for the FSP

This original FSP model was created in MAESTRO as it allows the easy creation of a barge like structure and has provisions for specification of the draft and the wetted surfaces (including the normal) which are included in its exported Nastran model. The model is a half model along the longitudinal direction considering the location of the charge tested. Hence, the model is restrained in the x-direction along the half-plane. The models used for the CASE-3D simulations and the Taylor Flat Plate theory simulations are identical in terms of the nodes and the elements.

MAESTRO treats all side and bottom elements as automatically wetted, so it is important to de-select the elements that are not wetted when importing to Nastran. Otherwise, the wetted surfaces exported from MAESTRO were formatted as per NASTRAN requirements using a simple C++ file read/write program and added to the NASTRAN input file (Refer to APPENDIX B: NASTRAN INPUT FILE CREATION FROM MAESTRO MODEL). The direction of element normal vectors are also important. In MAESTRO, the wetted elements have the normal based on the side where the force is acting on the plate (pressure side) i.e. inwards where the hydrostatic pressure is acting on the plate. Hence the pressure sides should be checked and flipped wherever necessary, for example, if a tank is present. This is opposite the normal direction for angle calculation in the Taylor flat plate model for an incident shockwave. Hence the normal direction is flipped while doing the relevant calculations inside the OpenFSI code.

First the FSP model is created using the basic dimensions and stiffener scantlings as a coarse mesh model. MAESTRO provides an easy-to-use meshing tool. The mesh size of the MAESTRO model is refined to 1 ft., which is half of the minimum stiffener spacing.

4.2 Results

As a preliminary analysis, the FSP is analyzed as a rigid body to compare vertical velocities by setting a high Young's modulus of $2 \times 10^{20}\ Pa$. The Taylor flat plate results for this case are compared with the CASE-3D results. The CASE-3D FSI model simulations are implemented in ABAQUS as discussed in Section 3.3. The results in the cases considered below, one can observe that the CASE-3D model results are affected by the cavitation region in the fluid domain. Hence, only the initial kick-off velocities and the rise times are captured well by the Taylor Flat Plate theory model when compared to the CASE-3D results.

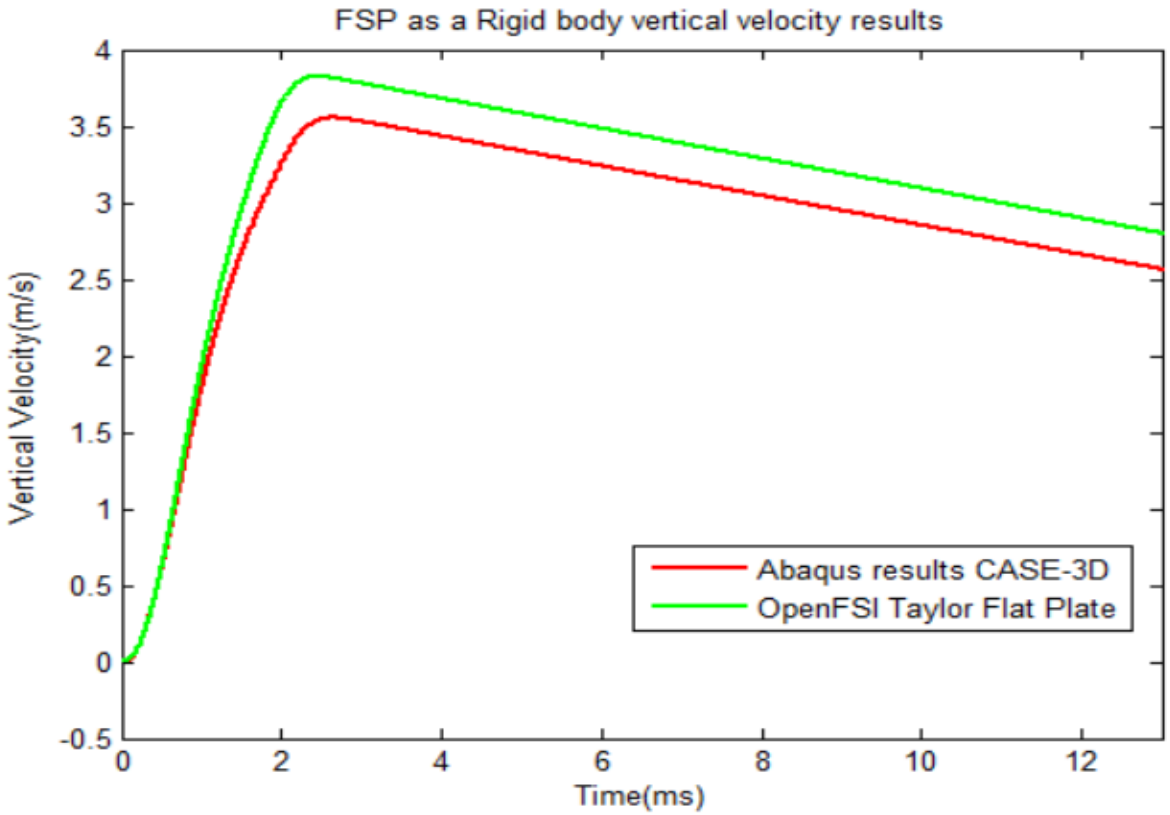

Figure 52: Vertical velocity of FSP as a rigid body.

Next the FSP is modeled with its normal material configuration as a deformable body and analyzed. The results of vertical velocities are obtained at nodes in the inner-bottom plating as well as bottom plating and then compared against the high-fidelity CASE-3D model. Then, the acceleration and the pseudo velocity shock spectra are also calculated and compared against the CASE-3D results. A design range based on [18, 61] is also drawn and compared against the results.

4.2.1 Design spectra preliminaries

Before discussing the deformable FSP results, some more discussion on shock design spectra and specifically FSP shock spectra is helpful. The design spectra formulation discussed in Section 2.4.3 must consider various locations in the ship or FSP in this case. Only design spectra values are available for the FSP and we do not know the precise locations where measurements were taken. The design spectra values are estimated from measured velocity and acceleration responses incorporating the effect of the shock spectrum dip phenomenon. The actual response spectra obtained using the Taylor Flat Plate theory model should be the same order of magnitude

as the design spectra, but we do not expect exact correspondence. Figure 53 shows a design spectra plot for the Floating Shock Platform for different velocity directions.

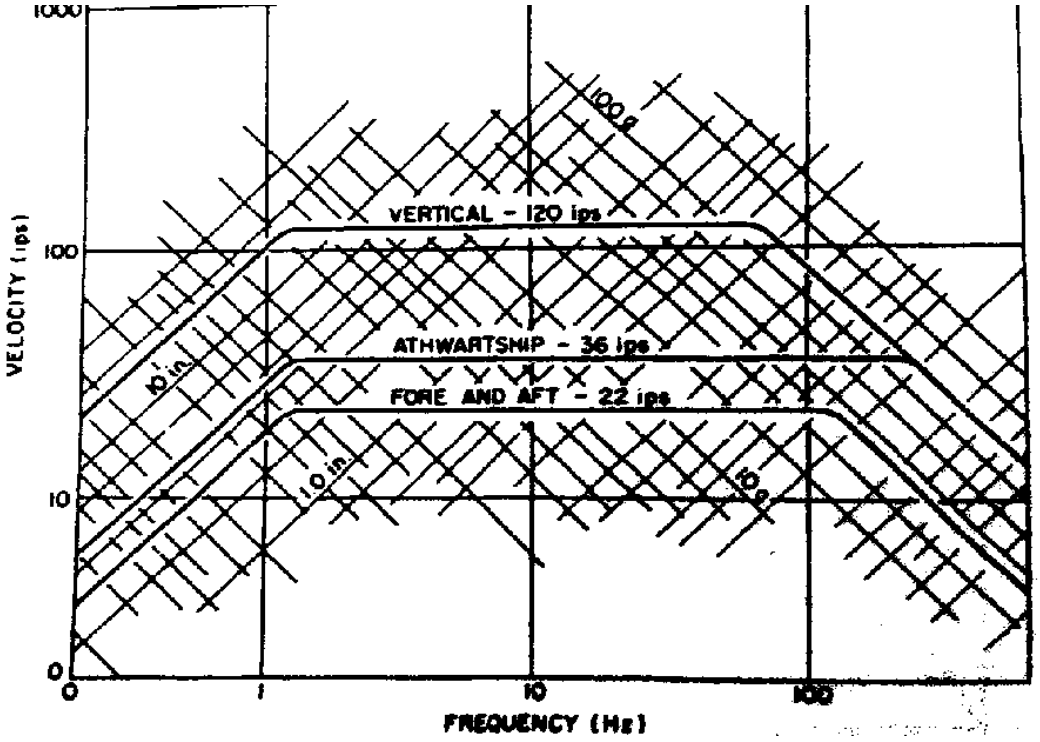

Figure 53:Design spectra for a Floating Shock Platform[18]

Figure 53 shows the bound for an FSP constant displacement spectra as $16\ in$ $(0.406\ m)$, the bound for a constant velocity spectra as $120\ in/s$ $(3.048\ m/s)$ and the bound for a constant acceleration spectra as $150g$ $(1471\ m/s^2)$. The critical frequency where the design spectra becomes velocity-limited based on *Figure 53* is approximately $1.2\ Hz$. The critical frequency where the design spectra becomes velocity-limited based on *Figure 53* is approximately$77\ Hz$. These values are used to assess the spectra obtained from our FSP CASE-3D model and OpenFSI Taylor Plate theory model.

4.2.2 Taylor Plate and CASE FSP Model Results for Various FSP Locations

The response of the floating shock platform in shock is captured in various vertical velocity plots and their corresponding shock spectra plots. All the Taylor flat plate based simulations using the OpenFSI-Nastran module are done with pressure cut-off implementation as discussed in Section 3.2. The plots are compared against the higher-fidelity CASE-3D model results.

4.2.2.1 Node 1646 in the inner bottom of FSP along the intersection of girders

Node 1646 is located at the intersection of a girder and floor in the inner bottom for the 1 ft. size mesh. The intersection of girder and floors in the inner bottom is an important location for equipment installation and where most of the testing is done. An example includes the installation of engine foundations.

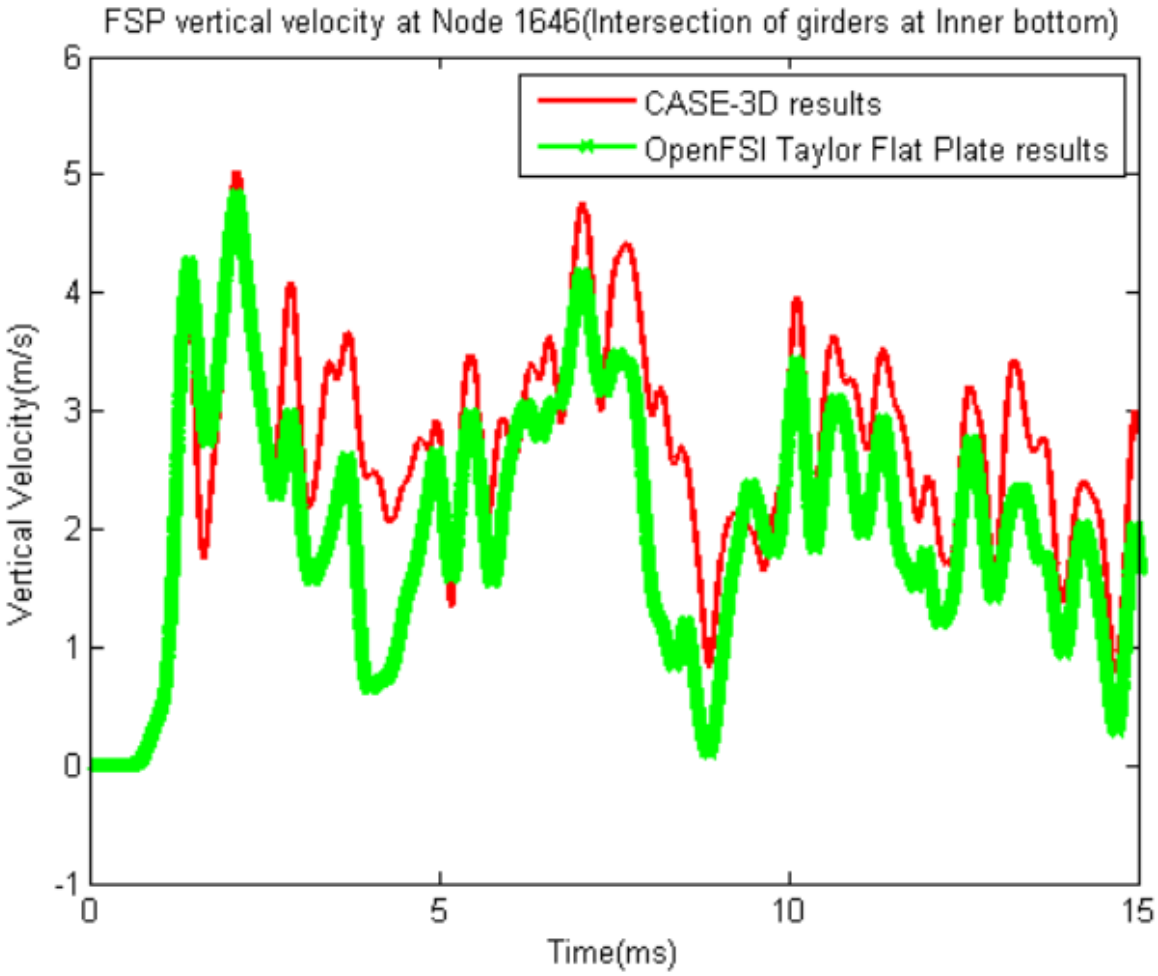

Figure 54:Vertical velocity time history at node 1646 of FSP (inner bottom plating – intersection of girders)

Figure 54 shows the initial two peaks of the vertical velocity and the general trend of the velocity curve given by the OpenFSI Taylor Flat Plate theory model is very similar to that of the high-fidelity CASE-3D model. The maximum velocity in this case is captured by both models as 4.8 m/s. *Figure* 54 also shows that the presence of stiffeners cause the oscillations to be of higher frequency compared to bare hull plating as seen later in *Figure* 57.

The comparison of shock spectra is also important when considering installed equipment response. *Figure* 55 shows that the maximum accelerations for a range of natural frequencies are captured well by the Taylor Flat plate theory compared to the 80ms time-history and 20 ms time-history of the CASE-3D model. The pseudo velocity spectra for node 1646 is shown in *Figure* 56 and in this plot the design spectra curve from *Figure 53* is also plotted for comparison.

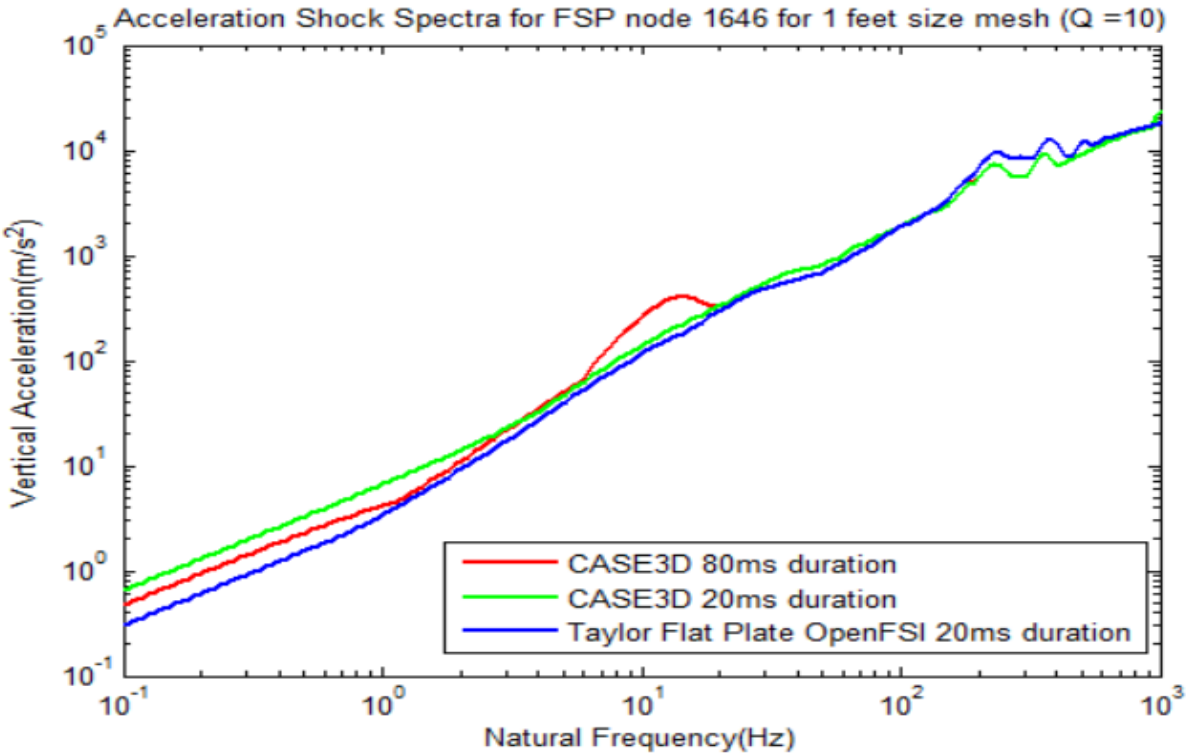

Figure 55:Vertical acceleration shock spectra at node 1646 of FSP (inner bottom plating –intersection of girders) compared with 20 ms as well as 80 ms spectra

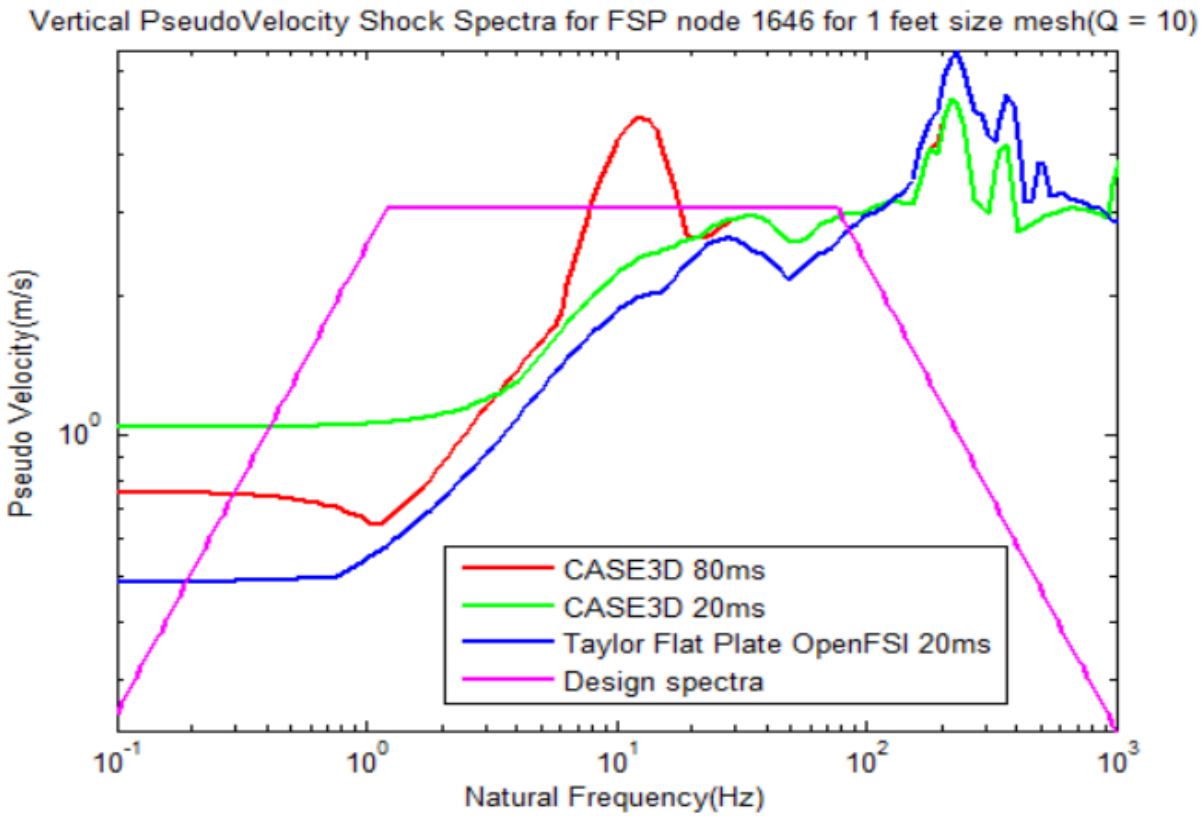

Figure 56:Vertical pseudo velocity shock spectra at node 1646 of FSP(inner bottom plating – intersection of girders) compared with 20 ms as well as 80 ms spectra

The natural frequencies between 0.2 Hz and 100 Hz (roughly) have repsonses that lie within the design limits. This range of natural frequencies where the design spectra bounds the results is also the similar to the range of frequencies for a typical destroyer as shown in Figure *10* [11].

4.2.2.2 Node 2320 in the inner-bottom of the FSP on the bare-plating

Next a node in the inner-bottom shell is selected such that it is on bare plating. The equipment or installations on bare plating require higher strength requirements as this region undergoes higher deformation than stiffened regions. These installations are typically smaller in size compared to the frame spacing of the stiffeners.

Node 2320 is selected on that basis and the vertical velocity of that node is plotted in *Figure* 57. The first two peaks and the trend of the curve is much better in this case. The maximum velocity captured by both the Taylor Flat Plate model and high-fidelity CASE-3D models equal to 8m/s. The plot of the vertical velocity on the bare plating goes to much higher magnitudes than the ones

along intersection of girders due to lower stiffness. As discussed in Section 4.2.2.1, the oscillation of vertical velocity on the bare plating is of much lesser frequency compared to the one along intersection of stiffeners.

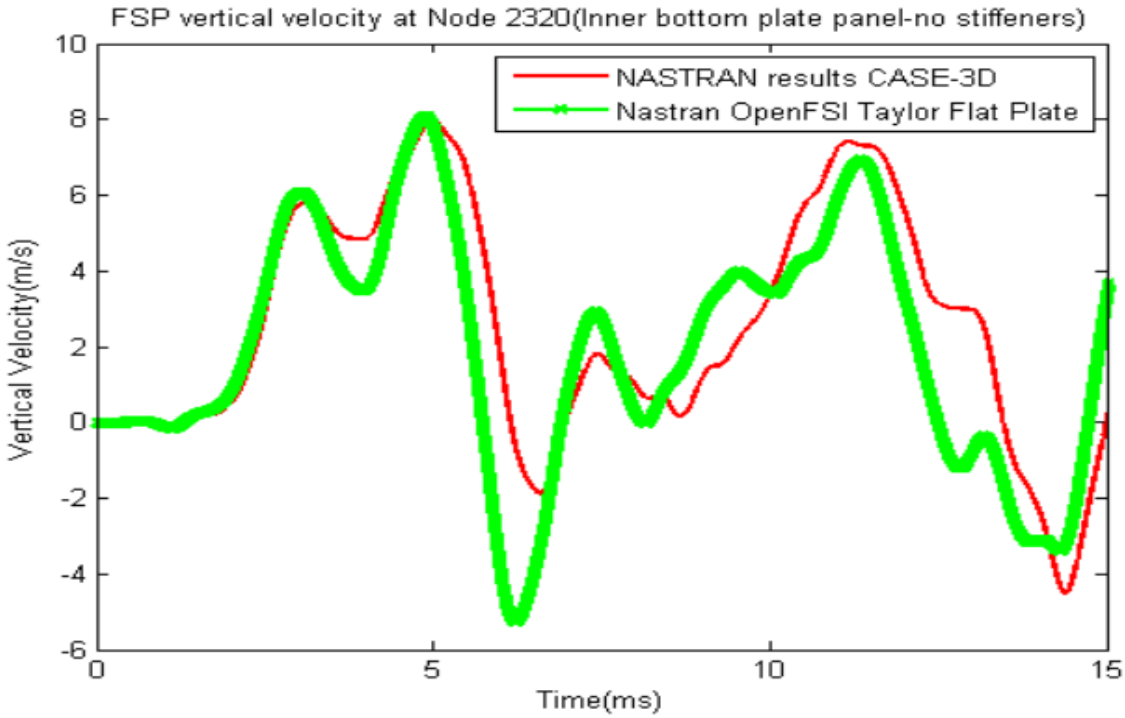

Figure 57:Vertical velocity time history at node 2320 of FSP (inner bottom plating – mid-point of panel)

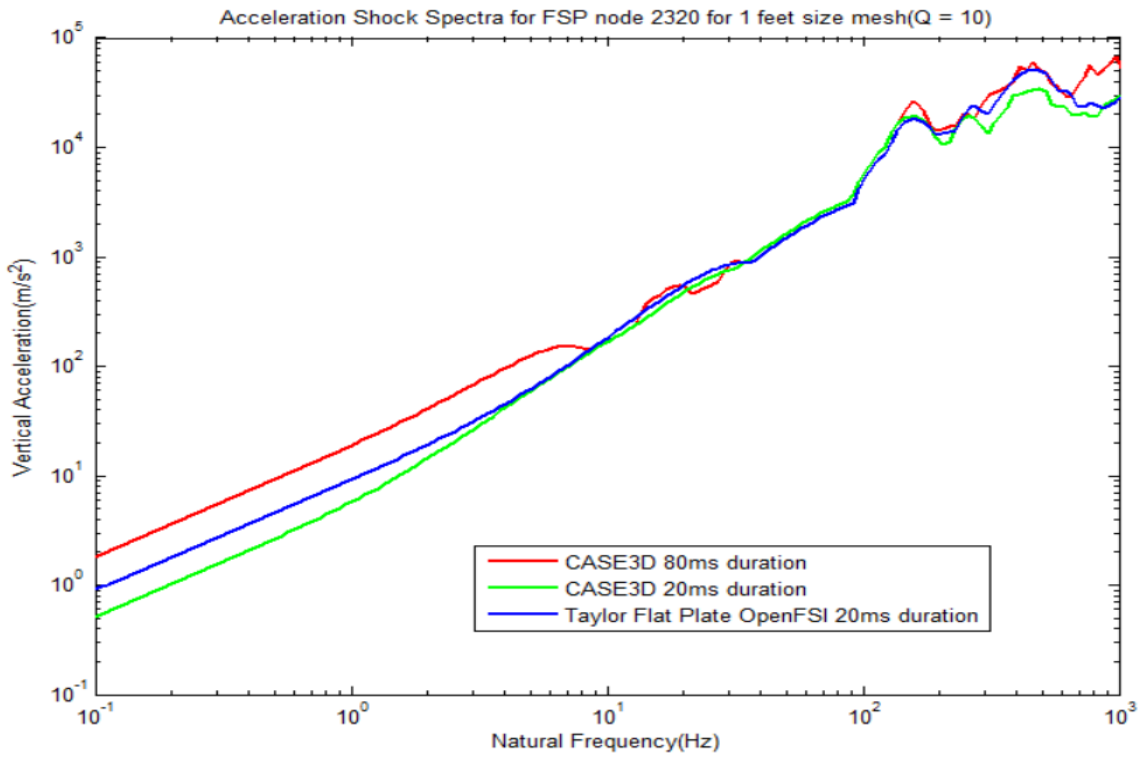

Figure 58:Vertical acceleration shock spectra at node 2320 of FSP (inner bottom plating – midpoint of panel) compared with 20 ms as well as 80 ms of CASE-3D spectra

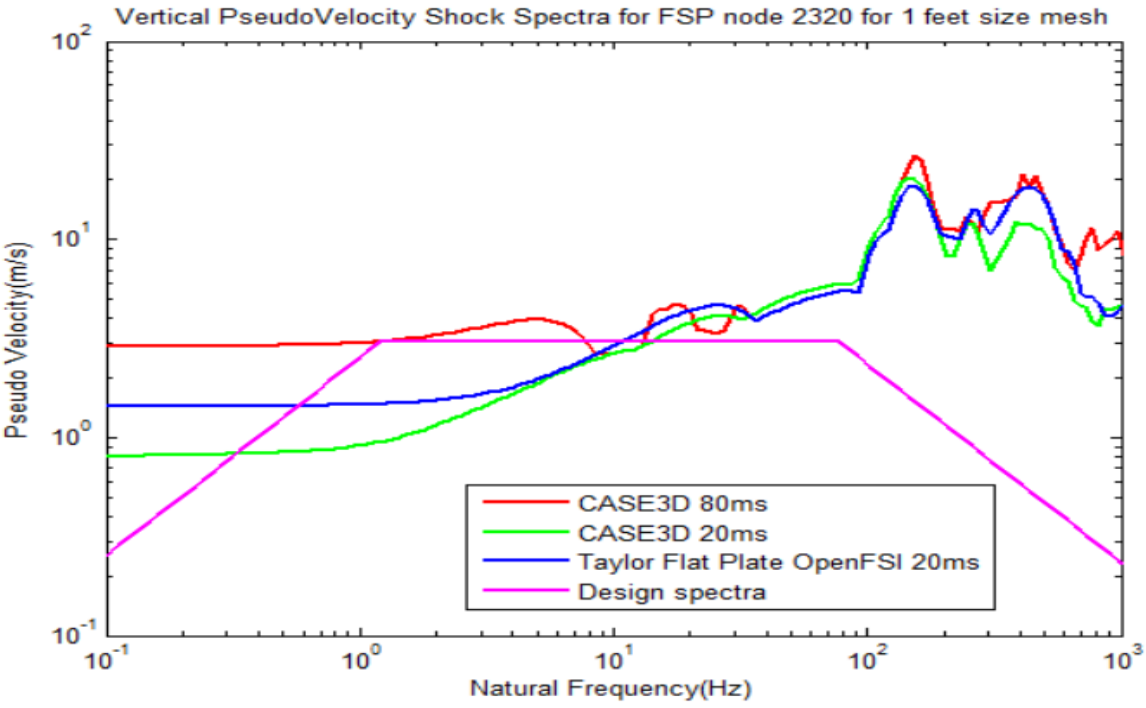

Figure 59:Vertical pseudo velocity shock spectra at node 2320 of FSP(inner bottom plating – midpoint of panel) compared with 20 ms as well as 80 ms CASE-3D spectra

The acceleration shock spectra plots as shown in Figure *58* compares OpenFSI Taylor Flat Plate model against the the 80ms time-history and 20 ms time-history of the CASE-3D model. The plots compare well for the natural frequencies 3 Hz to 250 Hz (approximately). The pseudo velocity spectra plots in Figure *59* has the results from Taylor Flat plate as well as the CASE-3D models. In this case, for the bare plating, the design spectrum does not bound the response spectrum sufficiently, suggesting that it is valid primarily for the nodes along intersection of girders/floors. Hence, we require the use of higher design spectral values of velocities and acceleration for this case.

4.2.2.3 Node 1488 in the bottom shell of the FSP along the intersection of girders/floors

Finally, a node in the bottom shell along the intersection of a girder and floor is considered. This is relevant for the equipment or foundations installed in the bottom plating, as the it could be in the region of direct impact. However, in the case of FSP, the double bottom is not high enough for placement and testing of equipment and hence the results are for reference purposes.

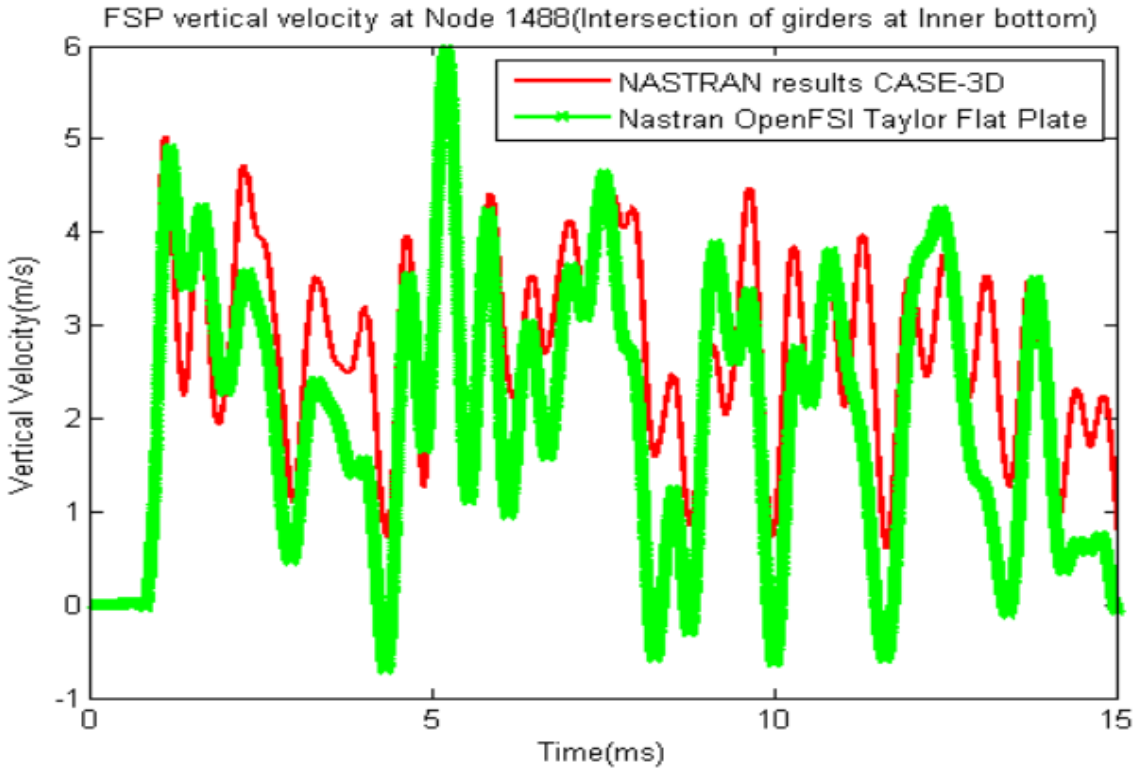

Figure 60:Vertical velocity time history at node 1488 of FSP (bottom plating – intersection of girders)

The initial peak or the kick-off vertical velocity attained is 5 m/s for both Taylor Flat Plate and CASE-3D models and the frequency of oscillation of vertical velocity is higher compared to bare plating seen in the case of node 2320. The Taylor Flat Plate theory achieves fairly accurate initial two peak velocities and the rise time to kick-off correctly.

Further, the acceleration spectra from the Taylor Flat Plate theory model in *Figure* 61 matches well with the CASE-3D model. The pseudo velocity spectra from Taylor Flat Plate model as well as the CASE-3D model lie well within the design spectra bounds in the frequency range of 1 Hz to 100 Hz as shown in *Figure* 62 but goes above the design values beyond 100 Hz.

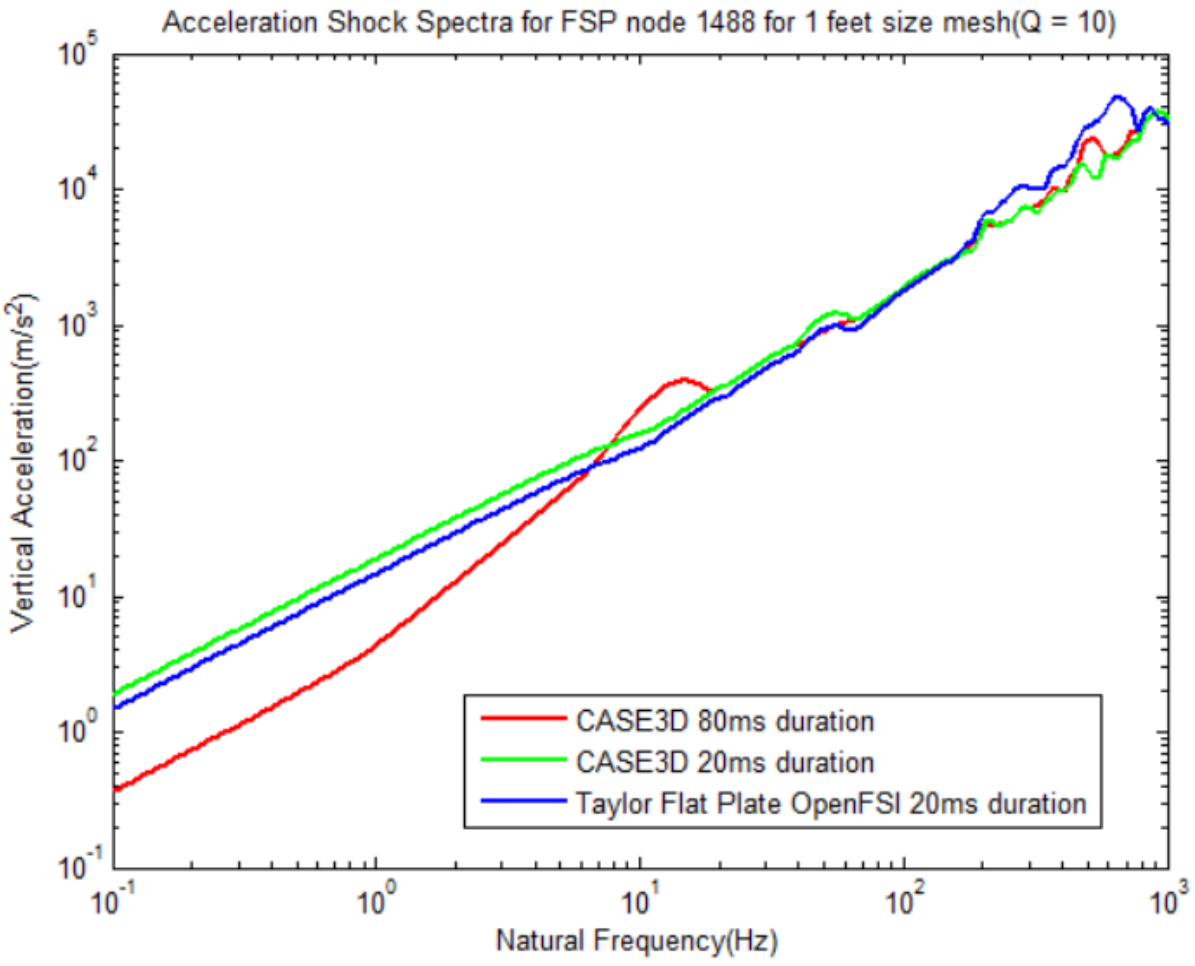

Figure 61:Vertical acceleration shock spectra at node 1488 of FSP (bottom plating – intersection of girders)

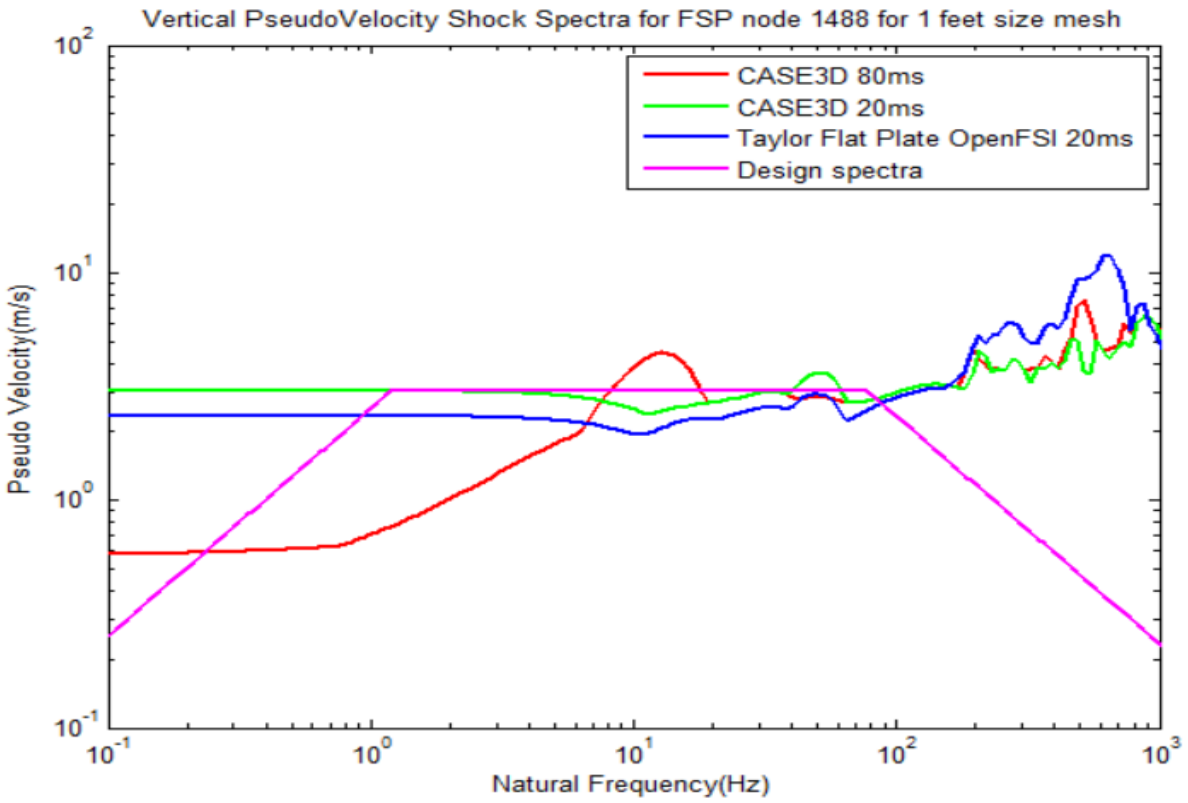

Figure 62:Vertical pseudo velocity shock spectra at node 1488 of FSP(bottom plating – intersection of girders)

4.2.2.4 Summary

In these case, it can be observed that towards the both ends of the natural frequencies, the response values obtained from the models exceed the design spectra values. However, these results are before the consideration of spectrum dip and the final values is expected to be lower than the design spectra curves, thus resulting in a feasible design.

The results obtained by applying the Taylor Flat Plate theory model at different locations in the FSP showed that it is able to capture kick-off velocities and the basic trend of the vertical velocity profile against the higher fidelity CASE-3D model. The acceleration spectra from CASE-3D model for time-history of 20ms and 80ms compare well against the Taylor Flat Plate theory model. The pseudo velocity curve plots obtained from the shock spectrum lies within the range of natural frequencies used for design.

4.2.3 Time-Series Contour Plots of Floating Shock Platform

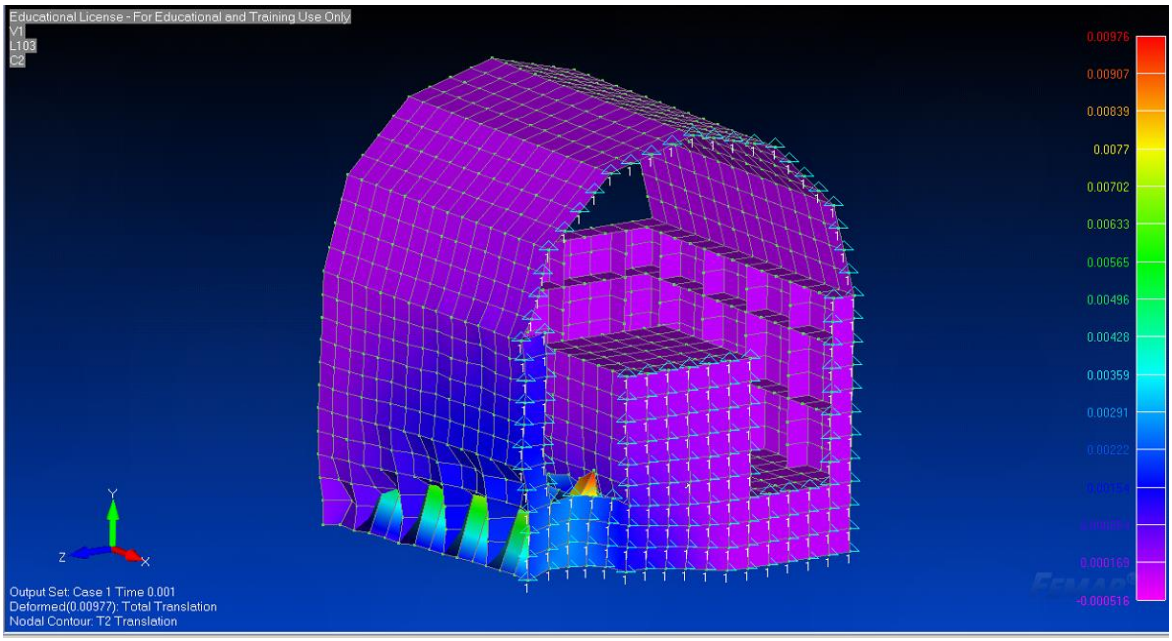

Figure 63:FSP contour plot of vertical translation(m) using FEMAP at t = 1 ms

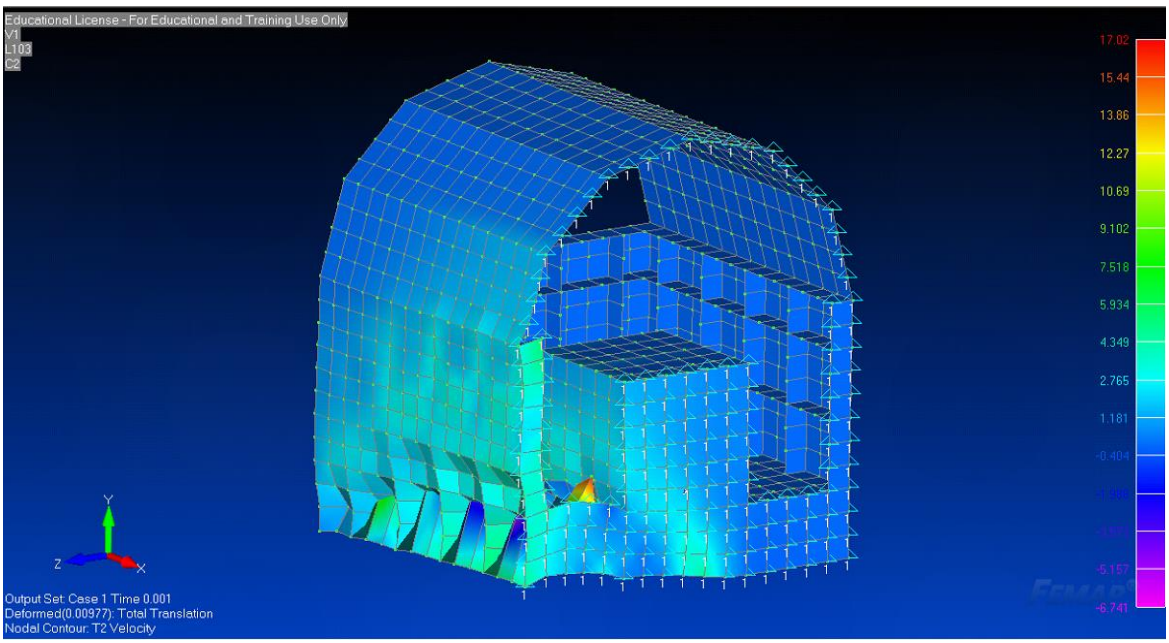

Figure 64: FSP contour plot of vertical velocity(m/s) using FEMAP at t = 1 ms

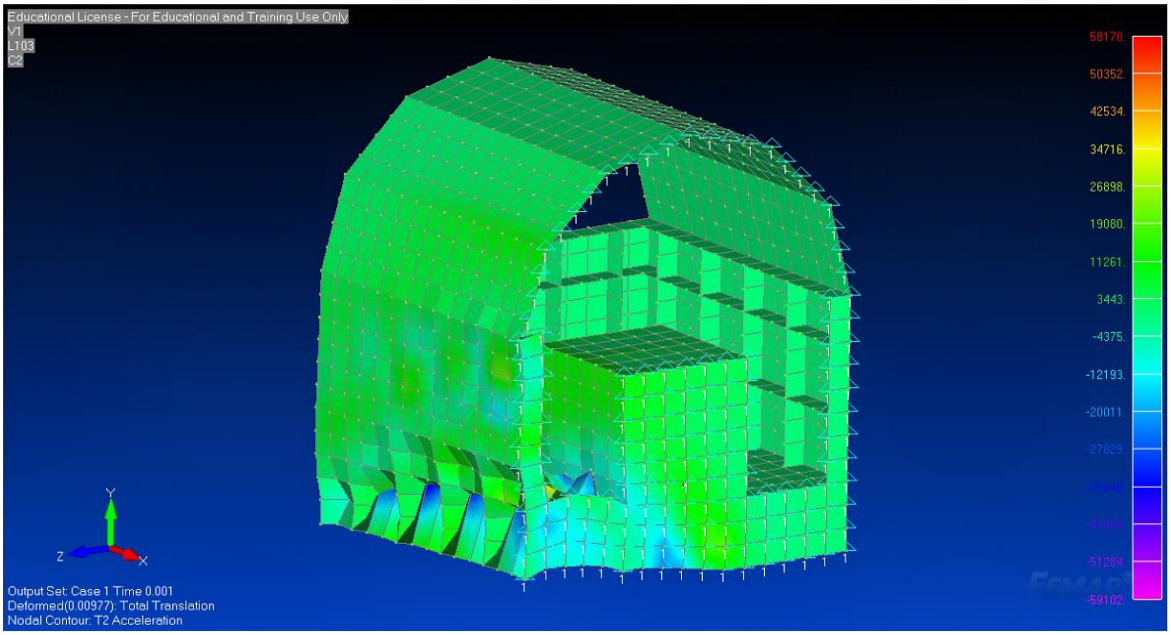

Figure 65: FSP contour plot of vertical acceleration (m/s^2) using FEMAP at t = 1 ms

Contour plots give an overall view of the distribution of velocities or acceleration or displacements on a color scale. Figure 63 shows the vertical translation of the different parts of the FSP as a contour plot with the values shown in 'm'. Figure 64 shows the vertical velocity propagation in the structure whereas Figure 65 shows the vertical acceleration propagation in the structure. The displacement is scaled about maximum 10% of the model size and this exaggerated deformation profile gives a sense of the direction of shock impact propagation.

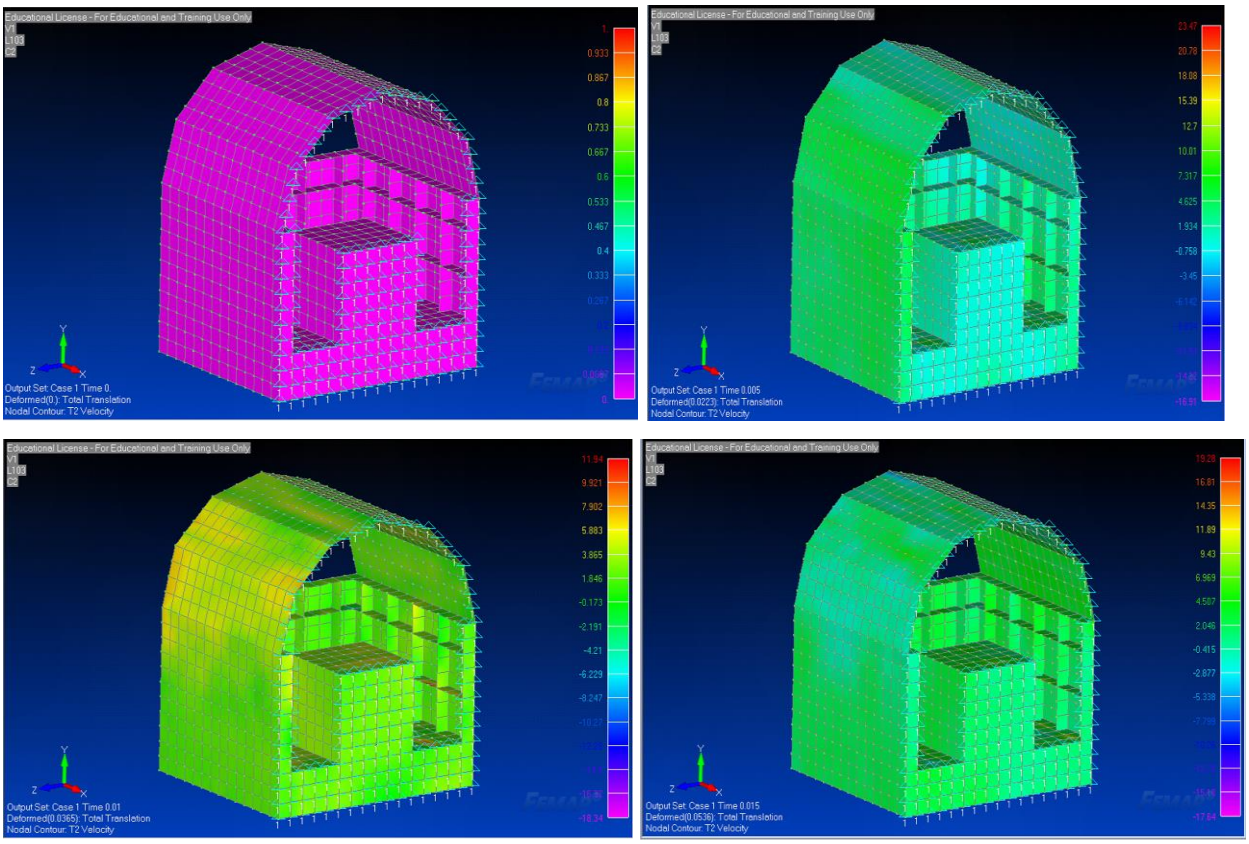

Figure 66:Vertical Velocity progression in FSP from t = 0ms to t=15ms.

Figure 66 shows the velocity progression from rest to kick-off velocity in the bottom areas towards 5ms and then the higher velocities in the deck areas towards 10ms and then the system going back to lower velocities as the incident shockwave pressure decreases.

This reflects how the shock causes high velocities in the bottom area first and the deck and the sides affected as the shock impact is transferred from the bottom shell to other areas and finally reducing its value as the incident shockwave decays exponentially.

5 CONCLUSION AND FUTURE WORK

In this thesis, the use of Taylor Flat Plate theory for the implementation of a simplified fluid-structure interaction tool for the design of ships against far-field early time UNDEX effects was explored. First the theory was developed and Taylor Plate behavior explored. Then code was developed to implement the Taylor Plate theory in conjunction with Nastran and Nastran OpenFSI.

A Bleich-Sandler plate case study was performed using the Taylor Plate and CASE models with further comparison to the published Bleich-Sandler results. It was shown that kick-off velocity and rise-time were captured well although the variations due to cavitation were not captured as the cut-off cavitation behavior did not occur at the Taylor Plate interface for that mass configuration. To compare the phenomenon when cut-off does occur at interface, a plate with 10 times the original mass was modeled and it is found that the Taylor Flat Plate theory model with cut-off undergoes velocity reduction less drastically than without pressure cut-off although not accurately as captured in the CASE-3D model.

A somewhat more complex dish-shell model was developed to transition the Taylor Plate model for application to a real structure. The results of this model also give an accurate kick-off velocity justifying further application to a Floating Shock Platform (FSP) barge. The results obtained applying the FSP model at different locations in the FSP inner bottom and bottom shell showed that it is able to capture kick-off velocities and the basic trend of the vertical velocity profile. The validity of the pseudo velocity curves within the range of natural frequencies used for design justifies the use of the Taylor Flat Plate theory model into early-stage vulnerability design.

The Taylor flat plate theory model behaves well when compared to the numerical results obtained using the CASE-3D model, a higher fidelity model. Also the Taylor flat plate theory is able to capture the basic physics of the phenomenon. It was not possible to compare against experimental data due to classified nature of such information. Taylor flat plate theory is helpful for design comparisons between different cases. The response spectra at different locations generated by the model enables the design decision regarding the shock sensitive areas and the placement of equipment or extent of shock hardening required.

In terms of computational effort, the Taylor Flat Plate Theory model would a better option to incorporate in design optimization as there is no actual fluid modelling involved. A sample run for the Floating Shock Platform shows that Taylor Plate model in Nastran takes 20-30% lesser time than CASE-3D model runs in Abaqus for sampling around 6 nodes. However, there is a need

to explore the model for curved hull surfaces and examine the computational efficiency as compared to CASE-3D simulation. Further, the accuracy is limited to a range of roughly 10-20% of the high fidelity model in terms of the initial kick-off velocity values observed for different nodes. A more detailed quantification in terms of the code implementation as well as the solver/scheme used is required.

5.1 Future Work

It is proposed that the Taylor Flat Plate theory model could be incorporated into the design exploration process for vulnerability assessment and consideration in early stage ship design. Recall Keil's Shock Factor:

$$Shock\ Factor, SF = C\frac{W^{\frac{1}{2}}}{R}f(\alpha) \tag{5.1}$$

Where: $C = C$(ship-type, location aboard the ship, type of installation)

The current C&RE process discussed in Section 1.4, already uses other metrics such as hit probability to guide and assess early design. In order to use metrics like Shock Factor, we need to be able to calculate C for different ship types, sizes structural designs and locations. Even with our simplified Taylor Flat Plate method, this still requires a lot of data regarding:

1. hull/deckhouse
2. subdivision and subdivision blocks (SDBs)
3. structural design and scantlings
4. MAESTRO model
5. Nastran model
6. Taylor Plate solution at 100s of SDB nodes
7. Design of experiments (DOE) to collect data
8. Response surface model (RSM) for calculating C and SF

Once we can calculate SF at different locations for different designs, we can assign vital components to SDBs, route cables and piping between SDBs, improve distributed system design, survivability, reduce cost, reduce space and weight requirements, and improve performance. This the way ahead and there is a lot of future work to do.

REFERENCES

1. *Shock Trial of USS Jackson,U.S. Navy photo, 2016, http://www.navy.mil/view_image.asp?id=220304, Accessed Jan 8,2018.*
2. *OPNAV INSTRUCTION 9070.1A, SURVIVABILITY POLICY AND STANDARDS FOR SURFACE SHIPS AND CRAFT OF THE U.S. NAVY, .* 2012.
3. Ball, R.E. and C.N. Calvano, *Establishing the fundamentals of a surface ship survivability design discipline.* Naval Engineers Journal, 1994. **106**(1): p. 71-74.
4. Said, M.O., *Theory and practice of total ship survivability for ship design.* Naval Engineers Journal, 1995. **107**(4): p. 191-203.
5. Park, J., *A Coupled Runge Kutta Discontinuous Galerkin-Direct Ghost Fluid (RKDG-DGF) Method to near-Field Early-Time Underwater Explosion (UNDEX) Simulations*. 2008, University Libraries, Virginia Polytechnic Institute and State University.
6. Reid, W.D., *The Response of Surface Ships to Underwater Explosions*. 1996, DEFENCE SCIENCE AND TECHNOLOGY ORGANIZATION CANBERRA (AUSTRALIA).
7. Cole, R.H., *Underwater Explosions*. 1948, Princeton University Press.
8. Klenow, B., *Finite and Spectral Element Methods for Modeling Far-Field Underwater Explosion Effects on Ships*. 2009, University Libraries,Virginia Polytechnic Institute and State University.
9. Wood, S.L., *Cavitation Effects on a Ship-Like Box Structure Subjected to an Underwater Explosion*. 1998, NAVAL POSTGRADUATE SCHOOL MONTEREY CA.
10. Mäkinen, K., *Cavitation models for structures excited by a plane shock wave.* Journal of Fluids and Structures, 1998. **12**(1): p. 85-101.
11. Keil, A., *The response of ships to underwater explosions*. 1961, DAVID TAYLOR MODEL BASIN WASHINGTON DC.
12. Swisdak Jr, M.M., *Explosion effects and properties. Part II. Explosion effects in water*. 1978, NAVAL SURFACE WEAPONS CENTER WHITE OAK LAB SILVER SPRING MD.
13. Webster, K.G., *Investigation of close proximity underwater explosion effects on a ship-like structure using the multi-material arbitrary Lagrangian Eulerian finite element method.* 2007.
14. *Shock Tests, High Impact Shipboard Machinery, Equipment and Systems, Requirements for, MIL-DTL-901E*. 2017.
15. Brown, A. and L.C. Kerns. *Multi-objective optimization in naval ship concept design*. in *Marine Systems and Technology (MAST) 2010 Conference, Rome, Italy*. 2010.
16. Arons, A. and D. Yennie, *Energy partition in underwater explosion phenomena.* Reviews of modern Physics, 1948. **20**(3): p. 519.
17. Taylor, G.I., *The pressure and impulse of submarine explosion waves on plates.* The scientific papers of GI Taylor, 1963. **3**: p. 287-303.
18. Scavuzzo, R.J., et al., *Naval Shock Analysis and Design*. 2000: Shock and Vibration Information Analysis Center, Booz-Allen and Hamilton, Incorporated.
19. Deshpande, V. and N. Fleck, *One-dimensional response of sandwich plates to underwater shock loading.* Journal of the Mechanics and Physics of Solids, 2005. **53**(11): p. 2347-2383.
20. Fleck, N. and V. Deshpande, *The resistance of clamped sandwich beams to shock loading.* Journal of Applied Mechanics, 2004. **71**(3): p. 386-401.

21. Liu, Z. and Y.L. Young, *Transient response of submerged plates subject to underwater shock loading: an analytical perspective.* Journal of applied mechanics, 2008. **75**(4): p. 044504.
22. Kambouchev, N., R. Radovitzky, and L. Noels, *Fluid–structure interaction effects in the dynamic response of free-standing plates to uniform shock loading.* Journal of Applied Mechanics, 2007. **74**(5): p. 1042-1045.
23. Kambouchev, N., L. Noels, and R. Radovitzky, *Nonlinear compressibility effects in fluid-structure interaction and their implications on the air-blast loading of structures.* Journal of Applied Physics, 2006. **100**(6): p. 063519.
24. Mair, H.U., *Benchmarks for submerged structure response to underwater explosions.* Shock and Vibration, 1999. **6**(4): p. 169-181.
25. DeRuntz Jr, J., *The Underwater Shock Analysis (USA) Manual.* Unique Software Applications, Colorado Spring, CO, 1996.
26. DeRuntz, J., T. Geers, and C. Felippa, *The Underwater Shock Analysis Code (USA-Version 3): A Reference Manual*. 1980, LOCKHEED MISSILES AND SPACE CO INC PALO ALTO CA.
27. Geers, T.L., *Doubly asymptotic approximations for transient motions of submerged structures.* The Journal of the Acoustical Society of America, 1978. **64**(5): p. 1500-1508.
28. Fox, P., Y.W. Kwon, and Y. Shin, *Nonlinear Response of Cylindrical Shells to Underwater Explosion: Testings and Numerical Prediction Using USA/DYNA3D/June 1, 1991-March 1, 1992*. 1992, Monterey, California: Naval Postgraduate School.
29. Beiter, K.A., *The effect of stiffener smearing in a ship-like box structure subjected to an underwater explosion*. 1998, Monterey, California. Naval Postgraduate School.
30. Mair, H.U., *Hydrocodes for structural response to underwater explosions.* Shock and Vibration, 1999. **6**(2): p. 81-96.
31. Mair, H.U., et al. *Lagrangian hydrocode modeling of underwater explosive/target interaction*. in *61st Shock and Vibration Symposium*. 1990.
32. Kundu, P.K., I.M. Cohen, and D. Dowling, *Fluid Mechanics 4th*. 2008, ELSEVIER, Oxford.
33. Miller, S., et al., *A pressure-based, compressible, two-phase flow finite volume method for underwater explosions.* Computers & Fluids, 2013. **87**: p. 132-143.
34. Felippa, C. and J. DeRuntz, *Acoustic fluid volume modeling by the displacement potential formulation, with emphasis on the wedge element.* Computers & structures, 1991. **41**(4): p. 669-686.
35. Zienkiewicz, O. and P. Bettess, *Fluid-structure dynamic interaction and wave forces. An introduction to numerical treatment.* International Journal for Numerical Methods in Engineering, 1978. **13**(1): p. 1-16.
36. Shin, Y.S. and N.A. Schneider. *Ship shock trial simulation of USS Winston S. Churchill (DDG 81): modeling and simulation strategy and surrounding fluid volume effects*. in *74th Shock and vibration symposium*. 2003.
37. Sprague, M. and T. Geers, *A spectral-element method for modelling cavitation in transient fluid–structure interaction.* International Journal for Numerical Methods in Engineering, 2004. **60**(15): p. 2467-2499.
38. Sprague, M.A. and T.L. Geers, *A spectral-element/finite-element analysis of a ship-like structure subjected to an underwater explosion.* Computer methods in applied mechanics and engineering, 2006. **195**(17): p. 2149-2167.

39. Mulder, W., *Spurious modes in finite-element discretizations of the wave equation may not be all that bad.* Applied Numerical Mathematics, 1999. **30**(4): p. 425-445.
40. Newton, R. *Finite element analysis of shock-induced cavitation*. in *ASCE, Spring Convention*. 1980.
41. *MAESTRO Structural Design Software Help Manual.* DRS Advanced Marine Technology Center, 2015.
42. *Getting Started with MSC/NASTRAN: User's Guide*. 2013, MacNeal-Schwendler Corporation.
43. Lu, Z. and A.J. Brown, *Cavitating Acoustic Spectral Element Method based 3-D Fluid-Structure Interaction model for far-field underwater explosion problem (Private Conversations & Team Collaboration)*. 2018, Virginia Tech, Department of Aerospace and Ocean Engineering.
44. Rogers, P.H., *Weak-shock solution for underwater explosive shock waves.* The Journal of the Acoustical Society of America, 1977. **62**(6): p. 1412-1419.
45. Kennard, E., *Cavitation in an elastic liquid.* Physical Review, 1943. **63**(5-6): p. 172.
46. Brown, A.J., *"Naval and Marine System Engineering –UNDEX Weapon Effects (Shock Fluid-Structure Interaction- Taylor Plate) "* 2018, Virginia Tech.
47. Kinsler, L.E., et al., *Fundamentals of acoustics.* Fundamentals of Acoustics, 4th Edition, by Lawrence E. Kinsler, Austin R. Frey, Alan B. Coppens, James V. Sanders, pp. 560. ISBN 0-471-84789-5. Wiley-VCH, December 1999., 1999: p. 560.
48. Felippa, C. and J. DeRuntz, *Finite element analysis of shock-induced hull cavitation.* Computer Methods in Applied Mechanics and Engineering, 1984. **44**(3): p. 297-337.
49. Bleich, H. and I. Sandler, *Interaction between structures and bilinear fluids.* International Journal of Solids and structures, 1970. **6**(5): p. 617-639.
50. Harris, C.M. and A.G. Piersol, *Harris' shock and vibration handbook*. Vol. 5. 2002: McGraw-Hill New York.
51. Alexander, J.E., *Shock response spectrum-a primer.* Sound & vibration, 2009. **43**(6): p. 6-15.
52. Steinberg, D.S., *Vibration analysis for electronic equipment*. 2000: John Wiley & Sons United States of America.
53. Irvine, T., *An introduction to the shock response spectrum.* Rev P, Vibrationdata, 2002.
54. Irvine, T., *Shock Response Spectrum (https://www.mathworks.com/matlabcentral/fileexchange/7398-shock-response-spectrum), MATLAB Central File Exchange, Retrieved 25 Jan 2018.* 2006.
55. McCarthy, R., *Shock Design Criteria for Surface Ships*. 1995, NAVSEA 0908-LP-000-3010, Naval Sea Systems Command.
56. Cunniff, P. and G. O'Hara, *A procedure for generating shock design values.* Journal of Sound and Vibration, 1989. **134**(1): p. 155-164.
57. *MSC Nastran User Defined Services User's Guide*. 2013, MacNeal-Schwendler Corporation.
58. Valente, C., et al., *Openfsi interface for strongly coupled steady and unsteady aeroelasticity*, in *International Forum on Aeroelasticity and Structural Dynamics, IFASD 2015*. 2016, International Forum on Aeroelasticity and Structural Dynamics (IFASD). p. 591-606.
59. Specification, M., *MIL-S-901D, Shock Tests, HI (High Impact) Shipboard Machinery, Equipment and Systems, Requirement for.* DODSIF Issue, 1989.

60. Specification, M., *MIL-DTL-901E, Requirement for Shock Tests, HI (High Impact) Shipboard Machinery, Equipment and Systems.* DODSIF Issue, 2017.
61. O'Hara, G.J. and R.O. Belsheim, *Interim design values for shock design of shipboard equipment*. 1963: National Technical Information Service.

APPENDIX A: NASTRAN-OPENFSI SET-UP

The explicit OpenFSI service flowchart for linking the fluid code to Nastran is shown in Figure *27* which is used in this thesis. The fluid code in this thesis is programmed inside the OpenFSI module rather than linking an external code to the OpenFSI module. The setting up of the OpenFSI code for compilation and integration to Nastran along with the formatting of the NASTRAN input file is discussed here.

Setting up of OpenFSI

The software requirements for the set-up used in this thesis are:

- Visual Studio 2010 or higher.
- Intel Fortran Compiler 12 or higher.
- MSC Nastran SDK (the same version as MSC Nastran)
- CYGWIN64 Terminal for C++ code compilation.
- MSC Nastran 2013 or higher.

Prior to running any OpenFSI related code, called an "OpenFSI SCA service" in NASTRAN, the 'User' Environment Variables need to be set as shown below.

- MSCNAST_EXE = C:\MSC.Software\MSC_Nastran\20131\bin\nastranw.exe
- SCA_LIBRARY_PATH = Z:\MyOpenFSI\Apps\Apps\WIN8664\lib
- SCA_RESOURCE_DIR = Z:\MyOpenFSI\Apps\Apps\res
- SCA_SERVICE_CATALOG = Z:\MyOpenFSI\Apps\Apps\res\SCAServiceCatalog.xml

The location prior to "\Apps\..." is the folder location where the OpenFSI code is set-up for compilation. The fluid code is constructed from a sample given in the folder found in: "C:\MSC.Software\MSC_Nastran\20130\msc20130\nast\services\Implementations\OpenFSI\". The completed code is placed in the Z:\MyOpenFSI\src folder (or the equivalent on the system). The following are the steps as part of setting up the OpenFSI service:

Step 1

The 'src' folder prior to compilation would have the following files obtained from the aforementioned folder:

- myOpenFSI.cpp
- myOpenFSI.h
- myOpenFSI.cdl

- myOpenFSI.sdl
- Sconscript

"Sconscript", "myOpenFSI.cdl" and "myOpenFSI.sdl" are kept intact. "myOpenFSI.h" is the header files and the relevant global variables used are defined here. "myOpenFSI.cpp" is file where the relevant modification is done.

Step 2

The following files should be copied from "C:\MSC.Software\MSC_Nastran\20130\msc20130\nast\services" and placed in the Z:\MyOpenFSI folder.

- SConopts
- SConscript
- SConstruct

Step 3

The code is compiled using CYGWIN64 in administrator mode. The code is to be recompiled if a modification is made. The following commands are used in the following order.

- alias genskeleton='C:/MSC.Software/SDK/2013/Tools/genskeleton.cmd'
- alias scons='C:/MSC.Software/SDK/2013/Tools/scons.cmd'
- cd /cygdrive/z/MyOpenFSI
- scons

The code implemented in OpenFSI.cpp

The function definition based on the algorithm described in Section 3.2 is provided below. The primary function in this file that is modified and used for this thesis is *getWettedNodeForces( )* function. The values used correspond to the Floating Shock Platform case.

```
SCA::SCAResult
myOpenFSI::getWettedNodeForces(SCA::MDSolver::Util::OpenFSI::wettedNodeSeq&
wettedNodeForces,SCA::SCABool& redoTimeStep)
{
	long int i, j, z, temp, k;		// loop counter. Added j, z, temp, and k
	//Declared Pmax, W, c, theta, R, c0, rho, P0, R2 as doubles.
	double Pmax, W, c, theta, R, c0, rho, P0, R2, tx, ty, tz, Angle, Vel, Vel1, Vel2,
	Vel3, PT, PT1, Area, FT, T1, T2, T3, T4, T5, T6, tPmax, tV, test, yLow, yHigh, h;
```

```
//Declares explosion coordinates (meters)
double draft = 1.2192;//in m
double halfbreadth = 2.438;//in m
double standoff = 3.048; //in m.  Stand-off is in z-direction as we are checking
the distance from the transverse section.
double depth=9.144;//in m
double ChargeProximity = pow((pow((depth-draft),2) + pow(standoff,2)),(double)
0.5);//in m from the closest node
//Draft = 0.141m for Bleich Sandler Case. Draft = 0.2032m for Shell Case. Draft =
0.9144m for FSP Case.
double expCords[3] = { 0,-depth,(standoff+ halfbreadth)}; //in m; Location of
explosive
double V1[3], V2[3], V3[3], V4[3], N[4], R3[3];
double val;
c = 1450; // m/s
rho = 1025; //kg/m^3
W = 27.216; //kg
double Patm = 101325; //P_atm in Pascals
////////////////////////////////////////////////////////////////

SCA::SCAReal64 force_x, force_y, force_z;   // Artificial force.
SCA::MDSolver::Util::OpenFSI::wettedNode nodalForce;
std::ofstream outfile;
outfile.open ("forces.fsi", std::ios::out | std::ios::app);
nWetNodes = wettedNodeForces.size();
std::cout << "       In --- getWettedNodeForces --- service method." << std::endl;
std::cout << "       Force size " << nWetNodes << std::endl;
////////////////////////////////////////////////////////////////
//Ensures previous nodal forces are 0
for (i=0; i< nWetNodes; ++i)
{
       nodalForce.id = wetNodes[i];
       nodalForce.x = 0;
       nodalForce.y = 0;
       nodalForce.z = 0;
       myForce[i][1] = 0;
       myForce[i][2] = 0;
       myForce[i][3] = 0;
       wettedNodeForces[i] = nodalForce;
}//Setting forces to zero before calculation
//End Addition
////////////////////////////////////////////////////////////////
//New loop for calculating force: //JC
for (i=0; i < nWettedElements; ++i)
{
       PT = 0;
       for (j=0; j < mesh[i][1]; j++)
       {
              z = 0;
              temp = 0;
              while (z == 0) {
              //Sets up temporary coordinates for each element and temporary
              velocity for each element
                     if (myCoordinates[temp][0] == mesh[i][j+2])
                     {
                            tempCoordinates[j][0] = myCoordinates[temp][0];
                            tempCoordinates[j][1] =
                                   myCoordinates[temp][1]+myDisplacement[j][1];
```

```
                        tempCoordinates[j][2] =
                                myCoordinates[temp][2]+myDisplacement[j][2];
                        tempCoordinates[j][3] =
                                myCoordinates[temp][3]+myDisplacement[j][3];
                        z = 1;
                }
                temp++;
        }
}//Loop for updating/setting tempCoordinates.
//Calculates a normal vector for each element
tx = tempCoordinates[0][1]-tempCoordinates[1][1];
ty = tempCoordinates[0][2]-tempCoordinates[1][2];
tz = tempCoordinates[0][3]-tempCoordinates[1][3];
V1[0] = tx;
V1[1] = ty;
V1[2] = tz;
tx = tempCoordinates[2][1]-tempCoordinates[1][1];
ty = tempCoordinates[2][2]-tempCoordinates[1][2];
tz = tempCoordinates[2][3]-tempCoordinates[1][3];
V2[0] = tx;
V2[1] = ty;
V2[2] = tz;
tx = V1[2]*V2[1]-V2[2]*V1[1];
ty = -(V1[2]*V2[0]-V1[0]*V2[2]);
tz = V1[1]*V2[0]-V1[0]*V2[1];
//N[3] = (pow(pow(tx,2)+pow(ty,2)+pow(tz,2),.5));
/*MAESTRO models wetted side(pressure side) normals inward i.e. Nastran
input file - nodes are clockwise - i.e in OpenFSI we need to flip to make
normal outward.*/
N[3] = -(pow(pow(tx,2)+pow(ty,2)+pow(tz,2),.5));
N[0] = tx/N[3]; //Components of a unit vector
N[1] = ty/N[3]; //Components of a unit vector
N[2] = tz/N[3]; //Components of a unit vector
//Calculates force at each node for each element
for (k=0; k < mesh[i][1]; k++)
{
        //Calculates R, R2 (range in ft), Pmax, c, theta, and PI: /JC
        R = pow(pow(tempCoordinates[k][1]-
        expCords[0],2)+pow(tempCoordinates[k][2]-
        expCords[1],2)+pow(tempCoordinates[k][3]-expCords[2],2),.5); // m
        //Using Equation in SI units as given in Swisdak paper.
        tPmax = pow(W,(double)1/3)/R;
        double K_P = 52.4*1e6;
        double alpha_P = 1.13;
        double K_tau = 0.084;
        double alpha_tau = -0.23;
        Pmax = K_P*pow(tPmax,alpha_P); //pascal
        theta=K_tau*pow(W,(double)1/3)*pow(pow(W,(double)1/3)/R,alpha_tau)
                /1000; //s
        val = -(R - (fsiTime + abs(ChargeProximity/ c))*c) / c;
        if (val<0)
                P0 = 0;
        else
                P0 = Pmax*exp(-val/ theta); // in Pascal
        tx = expCords[0]-tempCoordinates[k][1];
        ty = expCords[1]-tempCoordinates[k][2];
        tz = expCords[2]-tempCoordinates[k][3];
        R3[0] = tx;
```

```
R3[1] = ty;
R3[2] = tz;
Angle =
(R3[0]*N[0]+R3[1]*N[1]+R3[2]*N[2])/(pow(pow(R3[0],2)+
pow(R3[1],2)+pow(R3[2],2),.5)*pow(pow(N[0],2)+pow(N[1],2)+
pow(N[2],2),(double).5));

//Check if the element is in shadow region or non-shadow region.
if (Angle>=0)
	hydrostaticP[i] = 1;
else
	hydrostaticP[i] = 0;
Angle=acos(Angle);

for (z=0; z < 3;  ++z)
{
	if (z == 0)
	{
		yLow = tempCoordinates[0][2];
		yHigh = tempCoordinates[0][2];
	}
	else
	{
		if (tempCoordinates[z][2] < yLow)
		{
			yLow = tempCoordinates[z][2];
		}
		if (tempCoordinates[z][2] > yHigh)
		{
			yHigh = tempCoordinates[z][2];
		}
	}

}

//Modification for inclusion of atmospheric pressure and if
condition for heaviside.
h = abs((abs(yHigh)+abs(yLow)))/2;


//If the element is in non-shadow region, it is hit by shockwave.
if(hydrostaticP[i] == 1)
{
	z = 0;
	temp = 0;
	while (z == 0)
	{
		if (tempCoordinates[k][0] == myVelocity[temp][0])
		{
			test = myVelocity[temp][0];
			Vel1 = myVelocity[temp][1];
			Vel2 = myVelocity[temp][2];
			Vel3 = myVelocity[temp][3];
			z = 1;
		}
		temp++;
	}
	tx = abs(N[0]*Vel1);
```

```
                ty = abs(N[1]*Vel2);
                tz = abs(N[2]*Vel3);

                if (tx > ty && tx > tz && N[0]*Vel1 < 0)
                {
                        Vel1 = N[0]*Vel1;
                        Vel2= N[1]*Vel2;
                        Vel3 = N[2]*Vel3;
                        Vel = abs(Vel1 + Vel2 + Vel3);
                }
                else if (ty > tx && ty > tz && N[1]*Vel2 < 0)
                {
                        Vel1 = N[0]*Vel1;
                        Vel2= N[1]*Vel2;
                        Vel3 = N[2]*Vel3;
                        Vel = abs(Vel1 + Vel2 + Vel3);
                }
                else if (tz > tx && tz > ty && N[2]*Vel3 < 0)
                {
                        Vel1 = N[0]*Vel1;
                        Vel2= N[1]*Vel2;
                        Vel3 = N[2]*Vel3;
                        Vel = abs(Vel1 + Vel2 + Vel3);
                }
                else
                {
                        Vel = 0;
                }

                PT1 = 2*P0 - rho*c*Vel/cos(Angle)+Patm + (rho*9.81*h) ;

                //Pressure cut-off activation. - Ajai.
                if (PT1 < 0)
                        PT1 = 0;

                PT = PT+PT1;
                if (tempCoordinates[k][0]==2357 && k==3)
                        outfile<<fsiTime<<" "<<P0<<" "<<PT1 <<std::endl;

        }//End of hydrostaticP == 1 if statement
        else
        {
                PT1 = (rho*9.81*h)+Patm;
                PT = PT+PT1;
        }

}//Loops over each node of an element for pressure values.

//Area Calculation
if (mesh[i][1] ==  4)
{
        tx = tempCoordinates[0][1]-tempCoordinates[3][1];
        ty = tempCoordinates[0][2]-tempCoordinates[3][2];
        tz = tempCoordinates[0][3]-tempCoordinates[3][3];
        V3[0] = tx;
        V3[1] = ty;
        V3[2] = tz;
        tx = tempCoordinates[2][1]-tempCoordinates[3][1];
```

```
                ty = tempCoordinates[2][2]-tempCoordinates[3][2];
                tz = tempCoordinates[2][3]-tempCoordinates[3][3];
                V4[0] = tx;
                V4[1] = ty;
                V4[2] = tz;
                T1 = pow(pow(V1[0],2)+pow(V1[1],2)+pow(V1[2],2),.5);
                T2 = pow(pow(V2[0],2)+pow(V2[1],2)+pow(V2[2],2),.5);
                T3 = .5*T1*T2;
                T4 = pow(pow(V3[0],2)+pow(V3[1],2)+pow(V3[2],2),.5);
                T5 = pow(pow(V4[0],2)+pow(V4[1],2)+pow(V4[2],2),.5);
                T6 = .5*T4*T5;
                Area = T3+T6;
        }
        else
        {
                T1 = pow(pow(V1[0],2)+pow(V1[1],2)+pow(V1[2],2),.5);
                T2 = pow(pow(V2[0],2)+pow(V2[1],2)+pow(V2[2],2),.5);
                T3 = .5*T1*T2;
                Area = T3;
        }

        //(((PT - (mesh[i][1] * Patm))*Area / mesh[i][1]) is basically P_avg*Element
        //Area which is the elemental force.
        //(((PT - (mesh[i][1] * Patm))*Area / mesh[i][1])/ mesh[i][1] is thus the
        //nodal force.
        FT = ((PT - (mesh[i][1] * Patm))*Area / mesh[i][1])/ mesh[i][1];
        //This force is in the normal direction to the element. Not to the vector
        //connecting the node to the source.
        //We need to make sure when the force components are added, it is done
        //along proper directions b/w elements too.

        for (j=0; j < mesh[i][1]; j++)
        {
                //Force is along the direction opposite to normal
                force_x = FT*N[0];
                force_y = FT*N[1];
                force_z = FT*N[2];
                z = 0;
                temp = 0;
                while (z == 0)
                {
                        //Apply force to node
                        if (tempCoordinates[j][0] ==  myForce[temp][0])
                        {
                                myForce[temp][1] = myForce[temp][1] - force_x ;
                                myForce[temp][2] = myForce[temp][2] - force_y ;
                                myForce[temp][3] = myForce[temp][3] - force_z ;
                                z = 1;
                        }
                        temp++;
                }
        }//Second loop over each node of an element for updating the force due to
        //shock on a node.

}//End of element calculation loop.

for (i=0; i< nWetNodes; ++i)
{
```

```
                nodalForce.id = wetNodes[i];
                nodalForce.x = myForce[i][1];
                nodalForce.y = myForce[i][2];
                nodalForce.z = myForce[i][3];
                wettedNodeForces[i] = nodalForce;
        }

        //End Addition
        std::cout << "       ----------------------------" << std::endl << std::flush;
        outfile.close();
        return SCA::SCASuccess;
}
```

The initialize function is defined as below to initialize all the variables as well as define the normal direction.

```
SCA::SCAResult myOpenFSI::initialize(const SCA::MDSolver::Util::OpenFSI::wettedNodeSeq&
wettedNodeCoordinates,const SCA::MDSolver::Util::OpenFSI::wettedNodeSeq&
wettedNodeDisplacements, const SCA::MDSolver::Util::OpenFSI::wettedNodeSeq&
wettedNodeVelocities, const SCA::MDSolver::Util::OpenFSI::wettedSurfaceSeq&
wettedSurfaces)
{
long int i, j, k, z, temp;                  // loop counter.
double tx, ty, tz;
double V1[3], V2[3], V3[3], V4[3], N[4];

std::ofstream outfile;
outfile.open ("initialize.fsi");
outfile << "In Initialize call\n" << std::endl ;

std::ifstream addedMass;
addedMass.open("addedmass.txt");
//Add loop here to input the added mass data
//
//
//
//End of Loop
std::cout << "                                    In --- initialize --- service method."
<< std::endl << std::flush;
nWetNodes = wettedNodeCoordinates.size();
wetNodes = (long int *) malloc(nWetNodes * sizeof(long int));
if ( wetNodes == NULL )
{
        fprintf(stderr, "Error allocating memory\n");
        exit(1);
}

std::cout << "                                       # Wetted Nodes " << nWetNodes <<
std::endl ;
for (i=0; i<nWetNodes; ++i)
{
        wetNodes[i] = wettedNodeCoordinates[i].id ;
}

outfile << "Number of coordinates:  " << nWetNodes << std::endl;
```

```
for (i=0; i<nWetNodes; ++i)
{
	outfile << wettedNodeCoordinates[i].id << " "
		<<  wettedNodeCoordinates[i].x << " "
		<<  wettedNodeCoordinates[i].y << " "
		<<  wettedNodeCoordinates[i].z << "\n" ;
}
outfile << std::endl;

for (i=0; i<wettedSurfaces.size(); i++)
{
	outfile << "Surface " << i <<  " id: " << wettedSurfaces[i].wettedSurfaceId
		<< " name: " << wettedSurfaces[i].wettedSurfaceTag << std::endl;
	for (j=0; j<(wettedSurfaces[i].wettedElements).size(); j++)
	{
		outfile << j << "  id: " << wettedSurfaces[i].wettedElements[j].id ;
		outfile << "  type: " << wettedSurfaces[i].wettedElements[j].type << "
nodes: ";
		for (k=0; k<wettedSurfaces[i].wettedElements[j].numberOfNodes; k++)
		{
			outfile << " " <<
wettedSurfaces[i].wettedElements[j].connectivity[k];
		}
		outfile << std::endl;
	}
	outfile << std::endl;
}

//////////////////////////////////////////////////////////////////////////
// Declaration of Pressure array, mesh and number of wetted elements: /JC
pressureIn = (double*) malloc( (j) * sizeof(double) );
mesh = new __int64*[j];
nWettedElements = j;
hydrostaticP = (long int *) malloc(nWetNodes * sizeof(long int));
for ( i = 0;i<(j);i++)
{
	mesh[i] = new __int64[6];
}
for (i=0; i<(wettedSurfaces[0].wettedElements).size(); i++)
{
	j = 0;
	mesh[i][j] = wettedSurfaces[0].wettedElements[i].id;
	j = 1;
	mesh[i][j] = wettedSurfaces[0].wettedElements[i].numberOfNodes;
	if (wettedSurfaces[0].wettedElements[i].numberOfNodes == 4)
	{
		for (j=2; j<(wettedSurfaces[0].wettedElements[i].numberOfNodes+2); j++)
		{
			mesh[i][j] = wettedSurfaces[0].wettedElements[i].connectivity[j-2];
		}
	}
	else
	{
		for (j=2; j<(wettedSurfaces[0].wettedElements[i].numberOfNodes+2); j++)
		{
			mesh[i][j] = wettedSurfaces[0].wettedElements[i].connectivity[j-2];
		}
		mesh[i][j+1] = -1;
```

```
	}

}
tempCoordinates = new double*[nWettedElements];
for (int i = 0;i<(nWettedElements);i++)
{
	tempCoordinates[i] = new double[4];
}
// End addition
//Defines myDisplacement and myVelocity:
myCoordinates = new double*[nWetNodes];
for (int i = 0;i<(nWetNodes);i++)
{
	myCoordinates[i] = new double[4];
}
myVelocity = new double*[nWetNodes];
for (int i = 0;i<(nWetNodes);i++)
{
	myVelocity[i] = new double[4];
}
myDisplacement = new double*[nWetNodes];
for (int i = 0;i<(nWetNodes);i++)
{
	myDisplacement[i] = new double[4];
}
myForce = new double*[nWetNodes];
for (int i = 0;i<(nWetNodes);i++)
{
	myForce[i] = new double[4];
}
for (i=0; i<nWetNodes; ++i)
{
	myDisplacement[i][0] =  wettedNodeDisplacements[i].id ;
	myDisplacement[i][1] =  wettedNodeDisplacements[i].x ;
	myDisplacement[i][2] =  wettedNodeDisplacements[i].y ;
	myDisplacement[i][3] =  wettedNodeDisplacements[i].z ;
	myCoordinates[i][3] = wettedNodeCoordinates[i].z;
	myCoordinates[i][0] = wettedNodeCoordinates[i].id;
	myCoordinates[i][1] = wettedNodeCoordinates[i].x;
	myCoordinates[i][2] = wettedNodeCoordinates[i].y;
	myCoordinates[i][3] = wettedNodeCoordinates[i].z;
	myVelocity[i][0] = wettedNodeVelocities[i].id;
	myVelocity[i][1] = wettedNodeVelocities[i].x;
	myVelocity[i][2] = wettedNodeVelocities[i].y;
	myVelocity[i][3] = wettedNodeVelocities[i].z;
	myForce[i][0] = wettedNodeVelocities[i].id;
	myForce[i][1] = 0;
	myForce[i][2] = 0;
	myForce[i][3] = 0;
	outfile <<"Vertical Displacement for wetted node id," << myDisplacement[i][0]<<" =
"<<wettedNodeDisplacements[i].y<<" before interface with OpenFSI. \n";
	outfile <<"Vertical Velocity for wetted node id," << myVelocity[i][0]<<" =
"<<myVelocity[i][2]<<" before interface with OpenFSI. \n";
}

for (i=0; i<nWettedElements; i++)
{
	for (j=0; j < mesh[i][1]; j++)
```

```
	{
		z = 0;
		temp = 0;
		while (z == 0)
		{
			//Sets up temporary coordinates for each element and temporary
			//velocity for each element
			if (myCoordinates[temp][0] == mesh[i][j+2])
			{
				tempCoordinates[j][0] = myCoordinates[temp][0];
				tempCoordinates[j][1] = myCoordinates[temp][1];
				tempCoordinates[j][2] = myCoordinates[temp][2];
				tempCoordinates[j][3] = myCoordinates[temp][3];
				z = 1;
			}
			temp++;
		}
	}
	tx = tempCoordinates[0][1]-tempCoordinates[1][1];
	ty = tempCoordinates[0][2]-tempCoordinates[1][2];
	tz = tempCoordinates[0][3]-tempCoordinates[1][3];
	V1[0] = tx;
	V1[1] = ty;
	V1[2] = tz;
	//MAESTRO models wetted side(pressure side) normals inward i.e. Nastran input file
- nodes are clockwise - i.e in OpenFSI we need to flip to make normal outward.
/*Here the vertices are assumed to be oriented in the clockwise direction in the grid
data. Hence for the bottom panel normal to point in the -y direction, we take the cross-
product accordingly. Let position vectors be a,b,c,d for each node. Then we have V1 = a-
b(vector directed from b to a) and V2 = c-b(vector directed from b to c). The cross-
product to get a anti-clockwise rotation would be V2 x V1.*/
	tx = tempCoordinates[2][1]-tempCoordinates[1][1];
	ty = tempCoordinates[2][2]-tempCoordinates[1][2];
	tz = tempCoordinates[2][3]-tempCoordinates[1][3];
	V2[0] = tx;
	V2[1] = ty;
	V2[2] = tz;
	tx = V1[2]*V2[1]-V2[2]*V1[1];
	ty = -(V1[2]*V2[0]-V1[0]*V2[2]);
	tz = V1[1]*V2[0]-V1[0]*V2[1];
	N[3] = -(pow(pow(tx,2)+pow(ty,2)+pow(tz,2),.5));
	N[0] = tx/N[3]; //Components of a unit vector
	N[1] = ty/N[3]; //Components of a unit vector
	N[2] = tz/N[3]; //Components of a unit vector
	outfile << "Element: " << mesh[i][0] << " N[0]: " << N[0] << " N[1]: " << N[1] <<
std::endl;
}

//End addition
for (i=0; i<wettedSurfaces.size(); i++)
{
	std::cout << "                                          " <<
		wettedSurfaces[0].wettedSurfaceId <<
		": " <<  wettedSurfaces[0].wettedSurfaceTag <<
		": " << (wettedSurfaces[0].wettedElements).size() << " elements" <<
std::endl;
}
```

```
std::cout << "                                        --------------------------" <<
std::endl << std::flush;
outfile.close();
addedMass.close();

return SCA::SCASuccess;
}
```

The updation of the displacements and velocities are done in

putWettedNodeDisplacementsAndVelocities() function:

```
SCA::SCAResult myOpenFSI::putWettedNodeDisplacementsAndVelocities(const
SCA::MDSolver::Util::OpenFSI::wettedNodeSeq& wettedNodeDisplacements,
       const SCA::MDSolver::Util::OpenFSI::wettedNodeSeq& wettedNodeVelocities,
       SCA::SCABool& redoTimeStep)
{
       long int i;                  // loop counter.
       std::ofstream outfile;
       outfile.open ("displacements.fsi", std::ios::out | std::ios::app);
       outfile << "In putDisplacements call at t=" << fsiTime << std::endl;

       nWetNodes = wettedNodeDisplacements.size();

       std::cout << "                                        In ---
putWettedNodeDisplacementsAndVelocities --- service method." << std::endl;
       std::cout << "                                      `  Displacement size " <<
wettedNodeDisplacements.size() << std::endl;
       std::cout << "                                         Velocity size " <<
wettedNodeVelocities.size() << std::endl;

       //Updates nodal location and velocities:
       for ( i = 0 ; i < nWetNodes ; i++ ) {
              myDisplacement[i][0] =  wettedNodeDisplacements[i].id ;
              myDisplacement[i][1] =  wettedNodeDisplacements[i].x ;
              myDisplacement[i][2] =  wettedNodeDisplacements[i].y ;
              myDisplacement[i][3] =  wettedNodeDisplacements[i].z ;
              myVelocity[i][0] = wettedNodeVelocities[i].id;
              myVelocity[i][1] = wettedNodeVelocities[i].x;
              myVelocity[i][2] = wettedNodeVelocities[i].y;
              myVelocity[i][3] = wettedNodeVelocities[i].z;
       }
       //End addition
       ////////////////////////////////////////////////////////////////////////
       std::cout << " ----------------------------------" << std::endl << std::flush;
       outfile.close();
       return SCA::SCASuccess;
}
```

Other Functions are only slightly modified for logging values etc.

```
SCA::SCAResult myOpenFSI::initializeTimeStep(const SCA::SCAReal64 time,
       SCA::SCAReal64& deltaTime,
       const SCA::SCAReal64 minimumDeltaTime,
       const SCA::SCAReal64 maximumDeltaTime,
       SCA::SCABool& redoTimeStep)
{
```

```
	fsiTime = (double) time;
	std::ofstream outfile;
	outfile.open ("initializeTime.fsi", std::ios::out | std::ios::app);
	outfile << "In InitializeTimeStep call\n" ;
	outfile << "time = " << time << "\n";
	outfile << "deltaTime = " << deltaTime << "\n";
	outfile << "minimumDeltaTime = " << minimumDeltaTime << "\n";
	outfile << "maximumDeltaTime = " << maximumDeltaTime << "\n";

	std::cout << "                                        In --- initializeTimeStep ---
service method." << std::endl << std::flush;
	std::cout << "                                                time = " << time <<
std::endl;
	std::cout << "                                            deltaTime = " <<
deltaTime << std::endl;
	std::cout << "                                          -------------------------"
<< std::endl << std::flush;
	outfile.close();

	return SCA::SCASuccess;
}


SCA::SCAResult myOpenFSI::finalizeTimeStep(const SCA::SCAReal64 time,
	SCA::SCABool& endSimulation,
	SCA::SCABool& redoTimeStep)
{
	std::ofstream outfile;
	outfile.open ("finalizeTime.fsi", std::ios::out | std::ios::app);
	outfile << "In finalizeTimeStep call\n" ;
	outfile << "time = " << time << "\n";

	std::cout << "                                        In --- finalizeTimeStep ---
service method." << std::endl << std::flush;
	std::cout << "                                                time = " << time <<
std::endl;
	std::cout << "                                          -------------------------" <<
std::endl << std::flush;

	outfile.close();

	return SCA::SCASuccess;
}



SCA::SCAResult myOpenFSI::terminate(const SCA::SCAString message)
{
	std::ofstream outfile;
	outfile.open ("terminate.fsi");
	outfile << "In terminate call:" << message << std::endl;
	std::cout << "                In --- Terminate --- service method." << std::endl;
	std::cout << "                                Message: " << message << std:: endl;
	std::cout << "                      -----------------" << std::endl << std::flush;
	free(wetNodes);
	outfile.close();
	return SCA::SCASuccess;
}
```

APPENDIX B: NASTRAN INPUT FILE CREATION FROM MAESTRO MODEL

This section describes the process assuming that the below-mentioned steps are completed as mentioned in the other parts of the appendix.

- Creation of a refined mesh model in Maestro with appropriate mesh size. This would imply that we have a Maestro '. mdl' file of the ship model with the refined mesh in the 'top' section in the 'Parts' tab.
- The OpenFSI code is compiled and relevant executables are created.
- The environment variables are set properly to run the case.
- A folder for running the wetted surface elements formulation code is setup.

Steps involved

- Create an active folder (a folder to set up the simulation and results) with the maestro file

  Open the Maestro file and using Export option in File Menu, export the wetted surfaces file. In this case, it would be FSP_latest_0.1_refined.wet

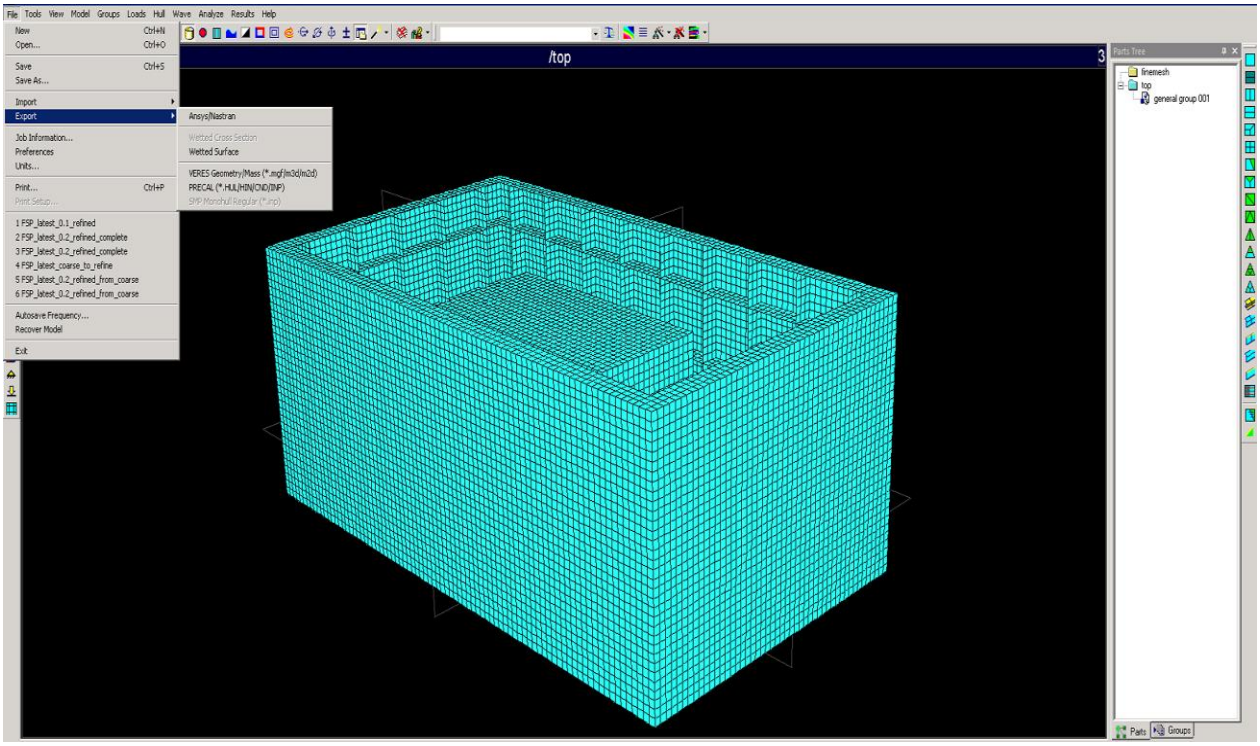

Figure 67: Sample fine-mesh model in MAESTRO

- Again using the Export option in the file menu, export the model to a .nas file. No need to make any changes in the export options.

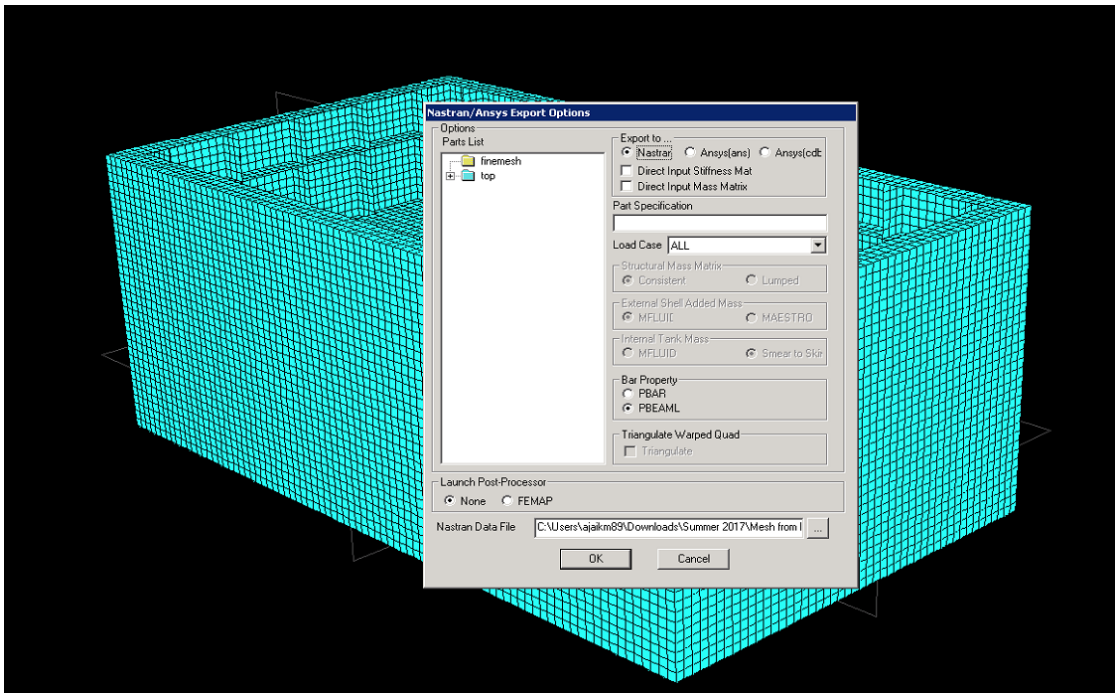

Figure 68:Export options in MAESTRO

A prompt may appear warning about no restraints defined. Just press Yes to continue to create a .nas file. The example filename given is FSP_latest_0.1_refined.nas. Now open the .nas file and take note of the last element in the CQUAD list. Say 70865 in the case of the barge's 0.1 size mesh as can be seen below.

```
100271 CQUADR      70858       2     3069     2607     2606     3068
100272 CQUADR      70859       2     2607      953      952     2606
100273 CQUADR      70860       2     3275     3070     3069     3276
100274 CQUADR      70861       2     3070     2608     2607     3069
100275 CQUADR      70862       2     2608      954      953     2607
100276 CQUADR      70863       2     3079     2636     3070     3275
100277 CQUADR      70864       2     2636     2174     2608     3070
100278 CQUADR      70865       2     2174      482      954     2608
100279 $Maestro Springs as DOF Springs
100280 $Write Property Cards
100281 PSHELL             1       30.003175        3                3
100282 PSHELL             2       1  0.0254        1                1
100283 PSHELL             3       2  0.0127        2                2
100284 PSHELL             4       4  0.0254        4                4
100285 PSHELL             5       10.022225        1                1
100286 ENDDATA
```

Figure 69:Node assignment for wetted element creation

- Copy the FSP_latest_0.1_refined.wet file created in Step 1 to the C++ project folder meant for creation of wetted element code format. Here the project folder is wetsurface_2.0. The .wet file is to be pasted in the folder wetsurface (inside the main folder) which contains the source code.
- Modify the code with the correct index for the Wetted elements section. Open the source code file and edit the line 109(highlighted in blue) to specify the starting index of wetted element. Here in this case the value is set to 70866 as the last element from CQUAD file was 70865.

```
92          if (g < waterline) {
93              std::cout << str1 << a << b << c << d << e << f << std::endl;
94              //ELE[count][0] = a; //element numbering
95              ELE[count][0] = b; //the first local node
96              ELE[count][1] = c; //the second local node
97              ELE[count][2] = d; //the third local node
98              ELE[count][3] = e; //the fourth local node
99              coord[count][0] = f; //x coordinate of element central point
100             coord[count][1] = g; //y coordinate of element central point
101             coord[count][2] = h; //z coordinate of element central point
102             count += 1;
103         }
104     }
105
106     int element_num = count;
107     std::ofstream myfile;
108     myfile.open("wetsurface_formated.txt");
109     int element_start = 70866;    //to be changed by the user
110     int element = element_start; //the starting numbering of wsf element
```

Figure 70:Modifications in wetted element extraction code

- Running the wetsurface project.

  In the main project folder, open the file named wetsurface_2.0.sln (File type: Microsoft Visual Studio Solution) using Visual Studio. In this case Visual Studio 2015 community version is used.Click on the Local Windows Debugger to execute the code. A prompt comes asking if you would like to build the code. Click yes. Then code execution occurs and during runtime, the filename of the. wet file is to be given as an input.

Figure 71:File name input for wetted element list creation

Once the code has finished running, the formatted wetted surface element Nastran code is found in the file wetsurface_formatted in the folder containing the source code.

Figure 72:Output File created by the code

NOTE:

- The wetted surface formatting code is meant for the barge case only i.e. the code can be used for different mesh refinements. The waterline and any other loading condition if modified has to be reflected in the code also.
- Also the WETSURF command contents for the OpenFSI code requires only one single surface (with all elements specified) whereas the given code generates multiple surfaces. Hence the code can be shortened based on the starting and ending wetted element indices. The example is shown below.

```
WETSURF       1    back                                                    +
+        73155   73156   73157   73158   73159   73160   73161   73162+
+        73163   73164   73165   73166   73167   73168   73169   73170+
.
.
+        74999   75000   75001   75002
WETSURF       2    left                                                    +
+        70866   70867   70868   70869   70870   70871   70872   70873+
```

And it goes on. In the example considered, the element indices considered are consecutive. Hence the WETSURF command in the C++ output file can be modified as:

```
WETSURF       2pressure                                                    +
+        70866    THRU   77801     BY     1                               +
```

Where:

70866: The first wetted element index

77801: The last wetted element index

1: Interval

BY: The interval between each index.

THRU: Connects the start and end element indices

Step 7: Formatting the Nastran input file.

Now the Nastran input file from Maestro has to be formatted. The Nastran input file exported from MAESTRO will have the following sections:

```
INIT MASTER(S)
ID, Untitled
SOL SESTATICS
TIME 10000
CEND
$Write SubCase if any
  TITLE=MAESTRO
  ECHO = UNSORT
  DISPLACEMENT = ALL
  OLOAD = ALL
  SPCFORCE = ALL
  FORCE = ALL
  STRESS = ALL
  SPC=1
  LOAD=1
BEGIN BULK
PARAM,POST,-1
PARAM,AUTOSPC,YES
PARAM,GRDPNT,0

$******1*******2*******3*******4*******5*******6*******7*******8*******9
$Write Constraints
FORCE         1    481       0     1.0-9.03E-4 -2.51778.592E-9
FORCE         1    482       0     1.0 54.8069 33.87117.747E-8
FORCE         1    483       0     1.0 54.8069 33.87117.747E-8
.
.
FORCE         1   33112       0     1.0-9.03E-4 -2.51778.592E-9
GRID        481       0 -4.2672  2.4384  2.1336
GRID        482       0 -4.2672 -0.9144 -1.8288
```

.

.

GRID 33112 0 4.2672 1.9304 -2.3368

MAT1 12.040E117.846E10 0.300 7850.

MAT1 22.041E117.849E10 0.300 7833.42

MAT1 32.041E117.849E10 0.300 7833.42

MAT1 42.040E117.846E10 0.300 8931.4

CQUADR 35881 1 13153 12782 12781 13152

CQUADR 35882 1 12782 12460 12459 12781

.

.

CQUADR 70865 2 2174 482 954 2608

$Maestro Springs as DOF Springs

$Write Property Cards

PSHELL 1 30.003175 3 3

PSHELL 2 1 0.0254 1 1

PSHELL 3 2 0.0127 2 2

PSHELL 4 4 0.0254 4 4

PSHELL 5 10.022225 1 1

ENDDATA

The following changes are to be made for using the OpenFSI interface with Nastran.

1. Append the contents of the ‘wetsurface_formatted’ file to the end of the Nastran input file right before the ENDDATA command as shown below. Refer the actual file for reference.

CQUADR 35882 1 12782 12460 12459 12781

.

.

CQUADR 70865 2 2174 482 954 2608

$Maestro Springs as DOF Springs

$Write Property Cards

PSHELL 1 30.003175 3 3

```
PSHELL       2      1  0.0254      1            1
PSHELL       3      2  0.0127      2            2
PSHELL       4      4  0.0254      4            4
PSHELL       5      10.022225      1            1
WETELMG    70866   QUAD4                              +
+          987     733     486     714
.
.
.
WETELMG    77801   QUAD4                              +
+          2174    482     954     2608
WETSURF    2pressure                          +
+          70866   THRU   77801    BY     1                  +
ENDDATA
```

2. Before beginning the wetted element section, you have to once again specify the OpenFSI service using the FSICTRL keyword. The line is to be added right after CQUAD prescription before the wetted element description.
3. Remove all FORCE command prescriptions as generated by Maestro.
4. Modifications are to be made in the NASTRAN control section, File Management Section, as below:

```
$! NASTRAN CONTROL SECTION
NASTRAN SYSTEM(444)=1
$! FILE MANAGEMENT SECTION
CONNECT SERVICE FSISRV1 'myOpenFSI'
$! EXECUTIVE CONTROL SECTION
SOL 400
TIME 10000
CEND
$WRITE SUBCASE IF ANY
  SUBTITLE = ONE
  TITLE = MAESTRO
```

```
  ECHO = UNSORT
  DISPLACEMENT (SORT1, PLOT) = ALL
  VELOCITY (SORT1, PLOT) = ALL
  ACCELERATION (SORT1, PLOT) = ALL
  DLOAD = 50
  ANALYSIS = NLTRAN
  TSTEPNL = 1
BEGIN BULK
GRID        481       0 -4.2672  2.4384  2.1336
GRID        482       0 -4.2672 -0.9144 -1.8288
.
.
```

And so on.

5. Now the relevant load sections are to be added. In this simulation setup, we specify gravity through Nastran but hydrostatic pressure and the shock wave pressure are specified through OpenFSI to simulate the effect of Fluid Structure Interaction. Thus we specify the OpenFSI interface for the second load condition. Also the OpenFSI source code in this case is setup for a single wetted surface as mentioned earlier and hence there would be only one TLOAD and WETLOAD command specification which is to be linked with OpenFSI. See the code below.

```
BEGIN BULK
$ Femap with NX Nastran Load Set 50 : NASTRAN 50
DLOAD        50       1       1     101       1     103
TLOAD1       101     101           LOAD       3
TLOAD1       103     103           LOAD       3
GRAV         101       0      1.      0.   -9.81      0.
WETLOAD      103       2 FSISRV1
TABLED1        3                                       +
+            0.      1.  10000.      1.    ENDT
CORD2C         1       0      0.      0.      0.      0.      0.      1.+FEMAPC1
+FEMAPC1       1.      0.      1.
CORD2S         2       0      0.      0.      0.      0.      0.      1.+FEMAPC2
```

```
+FEMAPC2      1.      0.      1.
$******1*******2*******3*******4*******5*******6*******7*******8*******9
GRID         481       0 -4.2672  2.4384  2.1336
```

6. Now this being a SOL 400 simulation, the time step is to be specified.
7. The default automatic constraints in Nastran is also to be disabled using the AUTOSPC keyword.

The relevant modifications to Nastran input file will be as below:

```
TSTEPNL       1    1000    2.-5      1             1      1     PW+
+          .001    .001    1.-7      3     10      4     .2      +
+            5       5            .75            .1     20.
PARAM,PRGPST,NO
PARAM,POST,0
PARAM,OGEOM,NO
PARAM,AUTOSPC,NO
$ Femap with NX Nastran Load Set 50 : NASTRAN 50
DLOAD         50       1       1     101       1     103
TLOAD1        101     101           LOAD       3
.
```

Now the simulation can be run directly through MSC Nastran to obtain the results in .xdb file format.

Results Visualization

The results can be easily viewed in FEMAP based on the '.xdb' file created. One of the typical output required is vertical velocity of the barge as the shockwave reaches the floating shock platform. Assuming there is the Nastran input file and the results file, one can visualize in FEMAP itself.

Step 1: Import the .nas file into FEMAP as an MSC/MD Nastran model

Step 2: Attach the results file('.xdb')

Step 3: To obtain the velocity plot, click the charting icon

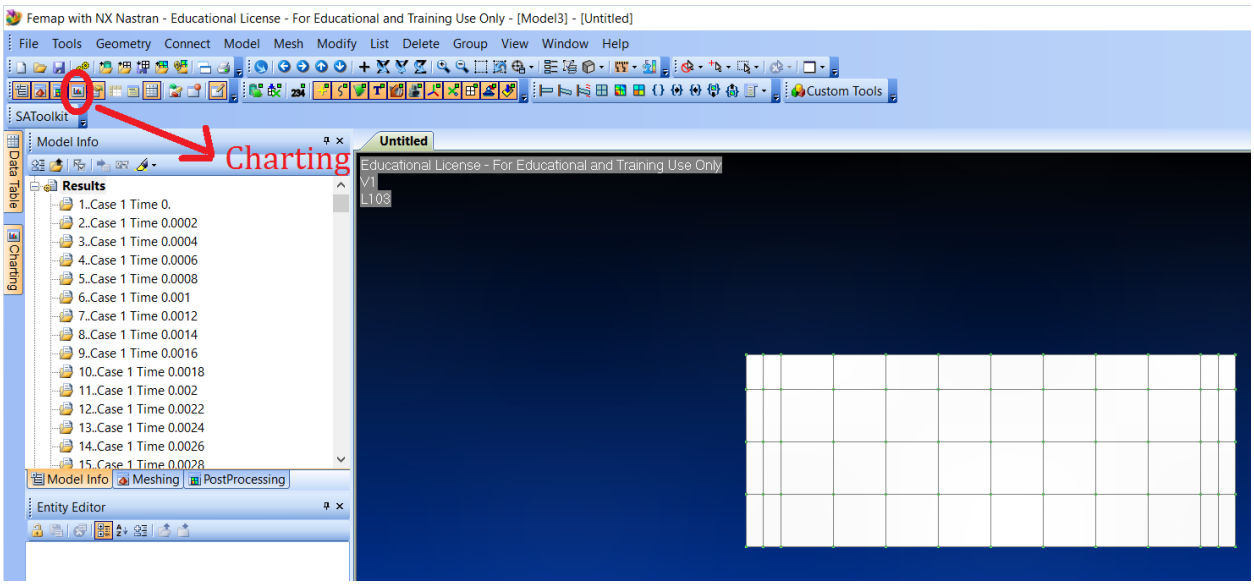

Figure 73:Charting icon in FEMAP

Once the charting window appears, select the xy icon to select the plot series.

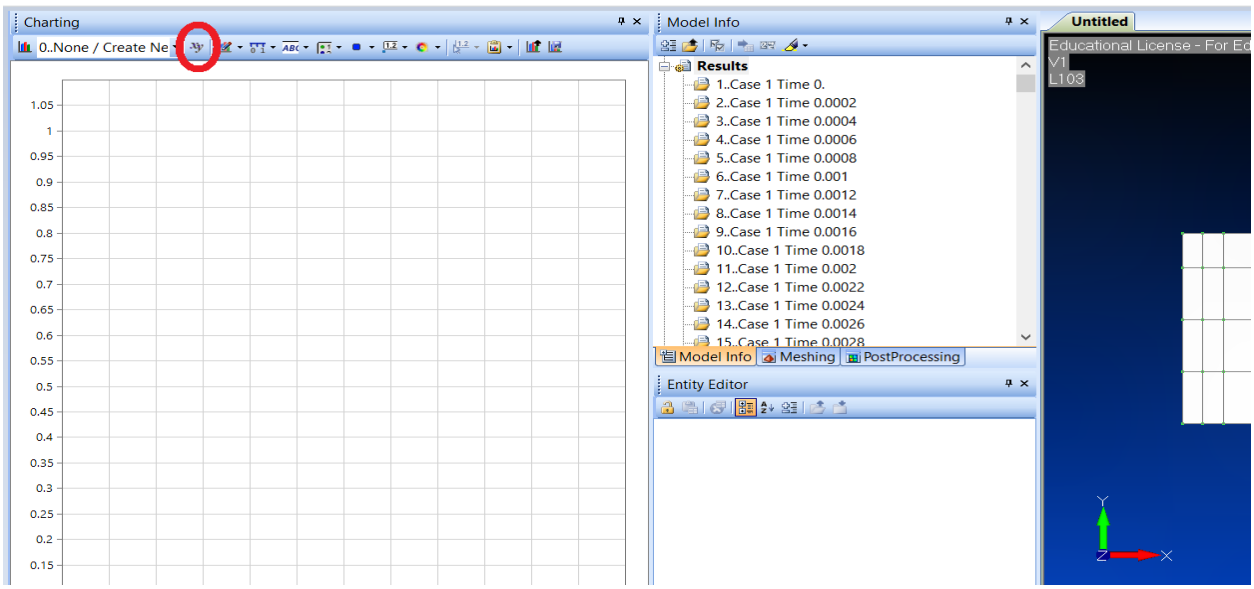

Figure 74:Plot selection in FEMAP charting

Click on ‘New Data series’ icon.

Use Vector vs Output set tab in the prompt window. Sample entries are shown in the *Figure* below.

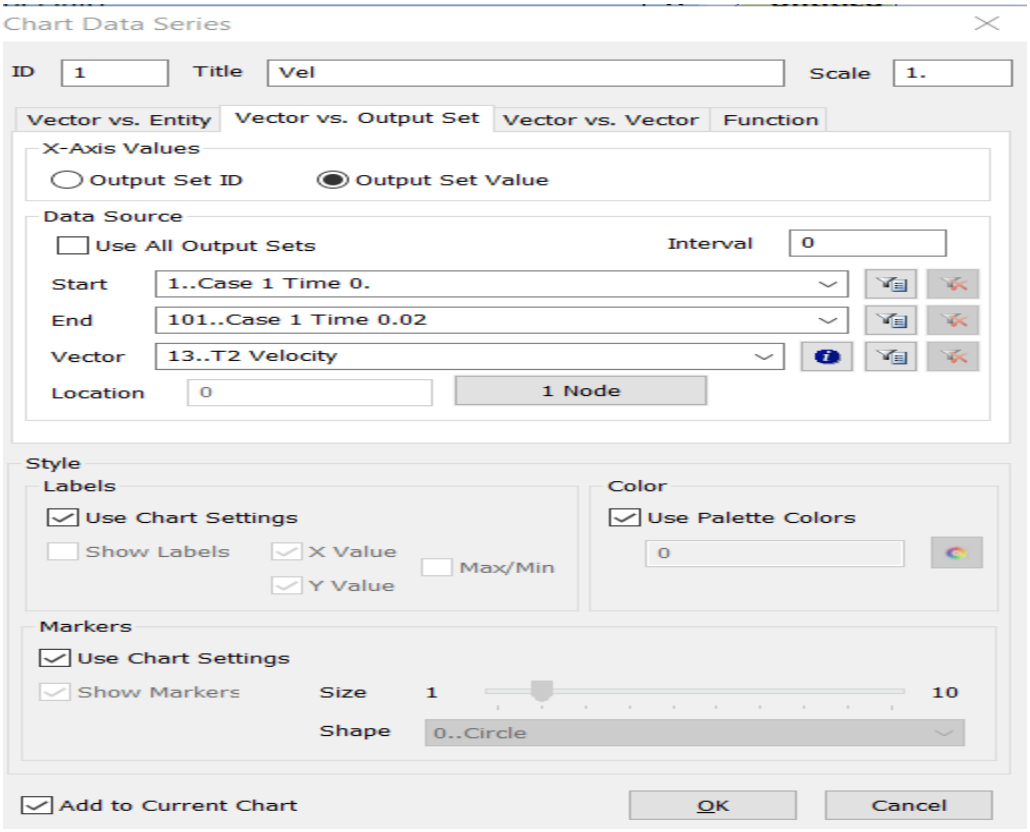

Figure 75:Sample input for plot creation

The Output set value gives the x-axis as time instead of a serial number whereas the vector T2 velocity gives the Vertical velocity at the node specified.

In this case choose, for the coarse mesh, node 212 which corresponds to the center of the bottom plating.

The output plot appears in the Charting window as shown below in *Figure 35*.

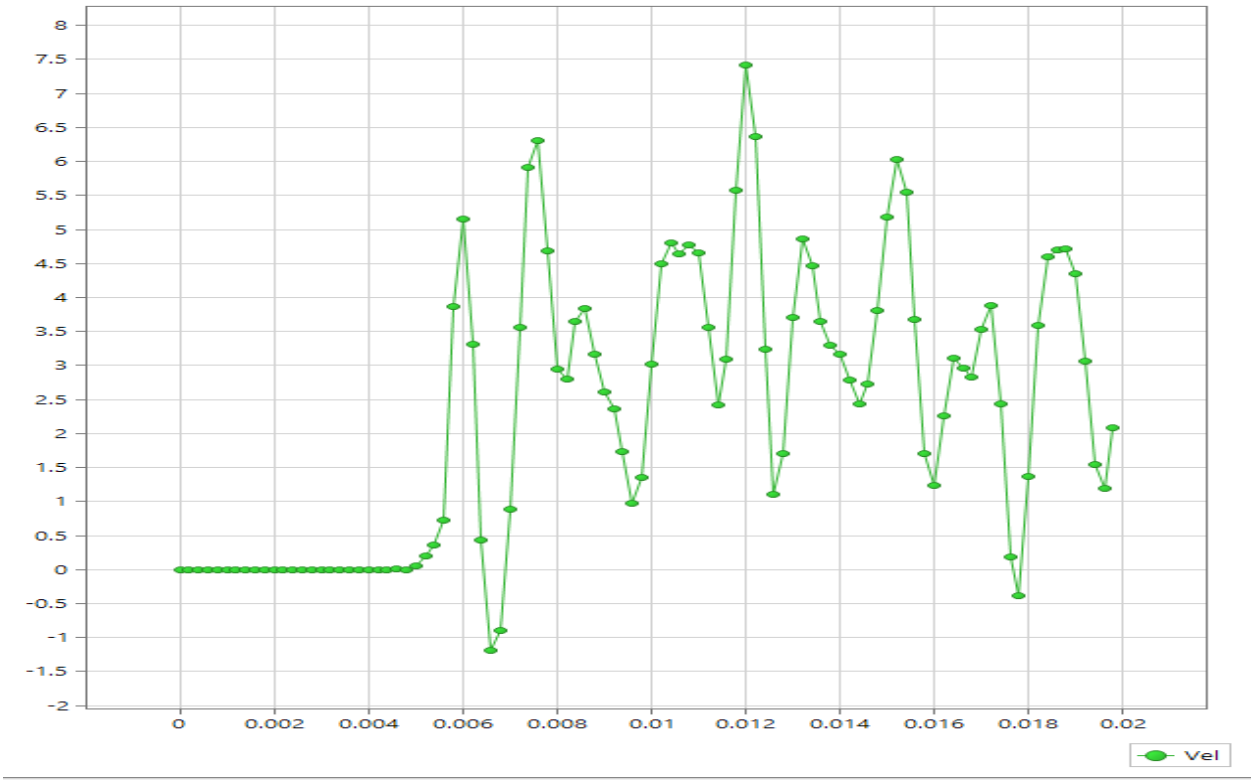

Figure 76:Sample plot in FEMAP

The axis labels and chart title can be specified under chart settings.

APPENDIX C: NASTRAN INPUT FILE FOR FLOATING SHOCK PLATFORM

```
$! NASTRAN CONTROL SECTION
NASTRAN SYSTEM(444)=1
NASTRAN SYSTEM(151)=1
NASTRAN BUFFSIZE=65537
NASTRAN DBCFACT=4
$! FILE MANAGEMENT SECTION
CONNECT SERVICE FSISRV1 'myOpenFSI'
$! EXECUTIVE CONTROL SECTION
SOL 400
TIME 10000
CEND
$WRITE SUBCASE IF ANY
  SUBTITLE = ONE
  TITLE = MAESTRO
  ECHO = UNSORT
  SET 1 = 1646, 1647, 2320, 1493, 1502, 1500, 1550, 1549, 1488,
2361,1613,2295,1731,1818
  DISPLACEMENT (SORT1, PLOT) = 1
  VELOCITY(SORT1, PLOT) = 1
  ACCELERATION(SORT1, PLOT) = 1
  DLOAD = 50
  ANALYSIS = NLTRAN
  TSTEPNL = 1
  SPC=2
$
BEGIN BULK
$
```

```
TSTEPNL       1   10001    2.-6      1             1       1      PW+
+           1.-4    1.-4    1.-5      3      10       4      .2       +
+             5       0            .75             .1     20.
PARAM,PRGPST,NO
PARAM,POST,0
PARAM,OGEOM,NO
PARAM,AUTOSPC,NO
CORD2C        1       0      0.      0.      0.      0.      0.      1.+FEMAPC1
+FEMAPC1      1.      0.      1.
CORD2S        2       0      0.      0.      0.      0.      0.      1.+FEMAPC2
+FEMAPC2      1.      0.      1.
$
DLOAD         50      1       1     101       1     103
TLOAD1        101     101           LOAD       3
TLOAD1        103     103           LOAD       3
GRAV          101       0      1.      0.   -9.81      0.
WETLOAD       103       2 FSISRV1
TABLED1       3                                          +
+             0.      1.  10000.      1.    ENDT
$ Write Property Cards
PSHELL        1       1 0.022225      1             1
PSHELL        2       3 0.003175      3             3
PSHELL        3       5  0.0127       5             5
PSHELL        4       1  0.0254       1             1
PSHELL        5       2  0.0127       2             2
PSHELL        6       4  0.0254       4             4
$ PLATE Property
MAT1          1  2.040E11          0.300   7850.
MAT1          2  2.050E11          0.280  7861.1
MAT1          3  2e+011            0.303   7850.
MAT1          4  2.040E11          0.300  7540.8
```

```
MAT1          5   2.040E11          0.300   4393.
$******1*******2*******3*******4*******5*******6*******7*******8*******9
GRID     564          -4.2672 1.524   -.9144
GRID     565          -4.2672 2.1336  .9144
GRID     566          -4.2672 1.524   -2.4384
GRID     567          -4.2672 2.1336  1.8288
$.
$.
$     Truncated for brevity
$.
$.
GRID     2518         6.866-9 1.2192  -.3048
GRID     2519         6.866-9 .9144   -.6096
GRID     2520         6.866-9 1.2192  -.6096
SPC1          2      1    2356 THRU   2520
CQUADR     3324      1    2275    2160    2137    2252
CQUADR     3325      1    2383    2275    2252    2371
CQUADR     3326      1    2160    1989    1966    2137
CQUADR     3327      1    1275    1197    1196    1276
$.
$.
$     Truncated for brevity
$.
$.
CQUADR     5536      5    1976    2066    2018    1967
CQUADR     5537      5    2462    2383    2371    2412
CQUADR     5538      5    2461    2462    2412    2413
CQUADR     5539      5    2425    2461    2413    2360
$!
FSICTRL  FSISRV1EXPLICIT      1
$!
```

```
WETELMG    5540  QUAD4                          +
+          569   797   796   570
WETELMG    5541  QUAD4                          +
+          797   955   954   796
$.
$.
$    Truncated for brevity
$.
$.
WETELMG    5937  QUAD4                          +
+          651   691   574   577
WETELMG    5938  QUAD4                          +
+          650   692   691   651
WETELMG    5939  QUAD4                          +
+          583   581   692   650
WETSURF     2pressure                        +
+         5540  THRU  5939   BY    1           +
ENDDATA
```