

Development of An Open-Source Methodology for Simulation of Civil Engineering Structures Subject to Multi-Hazards

Nicolette S. Lewis

A dissertation

submitted in partial fulfillment of the
requirements for the degree of

Doctor of Philosophy

University of Washington
2023

Reading Committee:
Michael R. Motley, Co-Chair
Marc O. Eberhard, Co-Chair
Andrew O. Winter
Pedro Arduino

Program Authorized to Offer Degree:
Civil and Environmental Engineering

© Copyright 2023

Nicolette S. Lewis

University of Washington

Abstract

Development of An Open-Source Methodology for Simulation of Civil Engineering Structures
Subject to Multi-Hazards

Nicolette S. Lewis

Co-chairs of the Supervisory Committee:

Michael R. Motley

Civil and Environmental Engineering

Marc O. Eberhard

Civil and Environmental Engineering

Design standards for structural earthquake and tsunami resilience provide guidelines to construct buildings that will remain standing during either a maximum considered tsunami or seismic event. Cascading hazards from earthquakes and tsunamis present a greater challenge to structural designers than individual hazards, as multi-hazard events involve a series of extreme loading stages, where structural damage sustained during one stage might influence performance or resilience during subsequent loading stages. To assist in the investigation of site-specific resilience of structures subject to cascading or multi-hazards through numerical analysis methods, an analysis approach for coupling disparate numerical analysis methods used for earthquake simulation and tsunami inundation simulations was developed and validated against analytical solutions and the results of experiments. This led to the development of a coupling driver, FOAMySees, which facilitates simulation of fluid-structure-interaction (FSI) between standard computational fluid dynamics

and finite element analysis models constructed within the OpenFOAM toolbox and OpenSees framework, respectively. The capabilities of OpenFOAM and OpenSees, when combined, allow for the evaluation of nonlinear structural response to multi-hazard excitation. Communication between component analyses is established by the coupling driver through preCICE, a library for multi-physics partitioned coupling. FOAMySees accompanies standard practices for modelling of structures subject to seismic excitation within OpenSees as part of an FSI simulation, which will be useful to analysts aiming to assess the residual capacity or performance of models of structural designs subject to complex three-dimensional loading from tsunami flows following a design-level seismic event. The approach allows for flexible construction of coupled models with validated, open-source, community-maintained software libraries, establishing the capability to conduct analyses that could assist in validating predictive force equations and structural design recommendations for structures subject to various natural hazards.

Acknowledgements

The work within this thesis was funded by the National Science Foundation (NSF) and Joy Pauschke (program manager) through Grants CMMI-1726326, CMMI-1933184, and CMMI-2131111. Thank you to NHERI Computational Modeling and Simulation Center (SimCenter), as well as their developers, funding sources, and staff for their continued support. It was a great experience to work with the SimCenter to implement this tool allowing for partitioned coupling of OpenSees and OpenFOAM as part of a digital-twin module within the NHERI SimCenter Hydro-UQ framework. Much of the development work of the research tool presented was conducted using University of Washington's HYAK Supercomputing resources. Thank you to UW HYAK and to the support staff of the UW HPC resources for their maintenance of the supercomputer cluster and for offering a stable platform for HPC development and computation, as well as for all of the great support over the last few years.

A huge thanks to my advisors, Michael Motley, Marc Eberhard, Pedro Arduino, and Andrew Winter. The patience they showed me through this doctoral process was immense, and I am so grateful that I had such understanding and flexible advisors during my time at UW. The personal growth this experience of getting the PhD has facilitated within me is something I can only compare to a few other things in my life - thank you all so much for the opportunity to continue to learn and investigate. Thanks also to my Graduate Student Representative Antonino Ferrante, and to the other faculty who have taught me and helped me along the way.

CFD modelling approaches implemented here as routines within a coupled solution were largely

informed by the work and research of Dr. Andrew Winter, whose guidance has been immensely helpful over the last few years. I likely would have learned nothing about OpenFOAM without Andrew's help, and would not have been able to perform the analyses which led to my first publication without his work constructing CFD models of the Oregon State University O.H. Hinsdale Large Wave Flume.

Thank you to Frank McKenna, Dawn Lehman, Richard Wiebe, and Vince Chaijaroen, for all of their patience and time over the last few years. It was so great to work with all of you, and I learned more than I could possibly include here in words.

Thank you to the UW Department of Civil Engineering, all of the faculty at University of Washington for your guidance, and to all my colleagues at the University of Washington. It was a joy to work with you all on the projects I was involved in during my time at UW - hopefully I will have the chance to work with you all again in the future!

DEDICATION

This thesis is dedicated to my family and friends. I'd particularly like to thank my mother Nancy, my father James, my sister Maddie, and my brother-in-law Tuk. I could fill up pages giving individual shout-outs to cousins, aunts, uncles, grandparents, great-aunts and uncles, second and third cousins, cousins of cousins and all else in my extended family, as well as family friends - but there are too many of you to list. I love you all so much, and am so grateful for all the support everyone has given me throughout my life. Though I have moved away from my original home in California and am excited about the future and where it will take me, I am finding that my heart and soul will always be with the people who raised me and were with me along the way - my community, my friends, my professors and the folks who taught me at every step of my education process so far, the people who struggled with me, succeeded with me, advocated for me, and believed in me, even when I did not. Thank you all so much, I don't know where I would be without the wonderful people in my life.

Contents

1	Introduction	1
1.1	Approach	2
1.2	Research Objectives	5
1.3	Scope of Research	6
2	Background	9
2.1	Conventional Approaches to Tsunami Resilient Design	9
2.1.1	Standard Probabilistic Approaches for Determination of Tsunami Hazards and Loading to Structures	10
2.2	Emerging Approaches for Tsunami-Resilient Design	17
2.2.1	Investigation of Effects of Time-Accurate Pressure Profiles from Incom- pressible Hydrodynamic Flows on Calculated Structural Resilience	17
2.2.2	Assessment of Topographic and Geometric Effects on Tsunami Loading to Structures	18
2.2.3	Deterministic Assessment of Resilience of Structures Expected to Provide Refuge in Tsunami Conditions	19
2.2.4	Validation of Probabilistic Approaches for Combined Hazard Analysis	21
2.2.5	Summary	23
3	Methods	25

3.1	Finite Element Method	25
3.1.1	Overview	25
3.1.2	Open System for Earthquake Engineering Simulation (OpenSees)	28
3.2	Finite Volume Method	32
3.2.1	Overview	32
3.2.2	Open-source Field Operation And Manipulation (OpenFOAM)	34
3.3	Partitioned Coupling	39
3.3.1	Overview	39
3.3.2	Open-Source Partitioned Multi-physics Coupling Library (preCICE)	47
4	Coupling Between Non-conformal Interfaces for Simulation of Nonlinear Fluid-Structure-Interaction	54
4.0.1	Terminology Utilized in this Chapter	55
4.1	Coupling Approach	56
4.1.1	Force Mapping Between Solutions	58
4.1.2	Displacement Mapping Between Solutions	61
4.2	Implementation of Coupling Approach	63
4.2.1	Overview	63
4.2.2	Coupled Model Participant Configuration and Initialization	66
4.2.3	FOAMySees Initialization	71
4.2.4	FOAMySees Coupling Driver Solution Loop Procedure	76
4.2.5	preCICE API Methods and OpenSees Framework Commands within FOAMySees Coupling Driver FSI Solution Loop	78
5	Analytical Test Cases	83
5.1	Hydrostatic Force FSI Case	83
5.1.1	Results	86

5.1.2	Mesh Convergence Study of Branch Interpolator	87
5.2	Fixed-end Frame Under Shifting Hydrodynamic Gravity Loading	94
5.3	Laminar Multiphase Hydrodynamic Impact	102
5.3.1	Analysis Details	103
5.3.2	Coupling Settings Study	104
5.3.3	Results Comparison	111
5.4	Test Cases With Sequential or Concurrent Ground Motions	115
5.4.1	Coupled Analysis Settings	116
5.4.2	Results	117
6	Experimental Test Cases	121
6.1	FSI Simulation of Concrete Core Wall Model at 1:6 Scale	122
6.1.1	Experimental Investigation	122
6.1.2	Numerical Investigation With Computational Fluid Dynamics	128
6.1.3	Fluid-Structure-Interaction Analysis	136
6.1.4	Discussion	137
6.2	Three Dimensional Elevated Structure Under Turbulent Breaking Solitary Wave Impingement with Real Structural Sections and Materials	143
6.2.1	Results	146
7	Discussion	152
7.1	Effect of Coupling Timestep on Solution	152
7.1.1	Work Equivalence Over the Coupling Data Projection Mesh	153
7.1.2	Coupling Settings and Effect on Work Resolved Across Coupling Interface	162
7.2	Computational Cost of Walhorn FSI Case	166
7.3	Limitations Of Coupling Methodology for Analysis of Structures Subject to Tsunami Hazards	169

7.3.1	Buoyancy Forces	169
7.3.2	Scour Effects During Tsunami Inundation	170
7.3.3	Debris Impact, Damming	171
7.4	Modelling Considerations	171
7.4.1	OpenFOAM Modelling Considerations	171
7.4.2	OpenSees Modelling Considerations	174
8	Future Work	179
8.1	Deterministic Simulation of Probabilistic Combined Hazard Scenarios to Existing Civil Engineering Structures for Assessment of Structural Resilience	179
8.2	Investigation of Effects of Structural Motion or Flexibility of Structures on Imparted Flow Forces	181
8.3	FOAMySees As an Approach For Numerical-Experimental Validation Studies and Investigations of Systems at Larger Scales	182
8.4	Study of Use of Coupling Approach for Other Structural Engineering Applications	183
8.5	Implementation of Methodologies for Acceleration of Computational Speeds . . .	183
9	Conclusions	185
A	Appendix	209
A.1	OpenFOAM Appendix	209
A.1.1	Finite Volume Schemes and Finite Volume Solution Settings	209
A.1.2	Coupling Related Files And Settings	219
A.1.3	Example Case Files	222
A.2	OpenSees Appendix	229
A.3	OpenSees Models	229
A.4	FOAMySees Appendix	239

A.5	preCICE Files	239
A.6	FOAMySees Code Listings	243

List of Figures

2.1	Tsunami Building Performance Objectives By Risk Category	14
3.1	OpenSees Class Structure (Figure obtained from McKenna et al. [2010])	29
3.2	OpenSees <i>Analysis</i> Class Composition (Figure obtained from McKenna et al. [2010] and Utilized Here to Supplement Discussion of OpenSees Class Structure)	30
3.3	OpenFOAM Polyhedral Cell and Referenced Components	35
3.4	Conventional-Serial-Staggered Explicit Coupling - Top of Figure (Structural Operator), Bottom of Figure (Fluid Operator)	41
3.5	Conventional-Parallel-Staggered Explicit Coupling - Top of Figure (Structural Operator), Bottom of Figure (Fluid Operator)	41
3.6	Improved-Serial-Staggered Explicit Coupling - Top of Figure (Structural Operator), Bottom of Figure (Fluid Operator)	42
3.7	Implicit Coupling with Gauss-Seidel Iterations	44
3.8	Implicit Coupling with IQN-ILS Iterations	45
3.9	Diagram Showing General Clustering of Points to Reference Points Utilizing K-means Clustering Algorithm	48
3.10	preCICE Python Language Binding commands utilized within initialization of a coupled solution	52

3.11	Diagram of preCICE Subroutine Calls Within An Explicitly-Coupled FSI code where the structural solution code is controlled with preCICE Python language bindings commands. This diagram is solely intended for informing about preCICE API function calls and where these calls are necessary, and is not indicative of the complete procedures required for a fully-coupled FSI solution	52
4.1	OpenFOAM FSI Patch	58
4.2	Conversion of Pressures to Nodal Forces: preCICE converts surface pressures along the FSI patch into nodal forces which may then be mapped to branch nodes	58
4.3	Nearest Neighbor or Radial Basis Function (RBF) mapping is calculated between branch node positions and CFD patch surface nodes: Isometric View of Model-to-Model Mapping from OpenFOAM surfaces to FOAMySees branch nodes and subsequent 1D interpolation to OpenSees FEA nodes	59
4.4	Elevation View of RBF Mapping from OpenFOAM surfaces to branch nodes to FEA nodes	59
4.5	Components of Force and Moment Utilized for Calculation of FEA Applied Forces and Moments From the Nodes of the Branch Coupling Mesh	61
4.6	Illustration of Rotation of Branch Nodes Completed by FOAMySees	63
4.7	Coupling Procedure for FOAMySees (abridged)	66
4.8	Objects From OpenSees Framework Which Are Maintained from Initial OpenSees Analysis or Redefined as Part of Solution Procedure. Bold Bordered Objects Highlighted in Green Are Redefined Each Coupling Iteration and/or Timestep Within the FOAMySeesCouplingDriver	71
4.9	FOAMySees Initialization of Coupling Data Projection Mesh	73
4.10	Overview of Use of Coupling Data Projection Mesh	76

4.11	FOAMySees Coupling Driver Solution Loop Summary. Steps within the solution loop summary are marked on the left hand side of the figure by symbols associated with each API or framework used in the solution loop. Overlaid symbols indicate combined contributions to completion of the step within the solution loop. Meshes used for the step are shown on the right hand side of each step.	78
4.12	FOAMySees Coupling Driver	80
5.1	Beam with properties E , I , L , A , ρ_B , within a domain t meters wide, topped with water h meters deep, under uniform gravity loading g	84
5.2	Phase-fraction (α) field for the CFD model	88
5.3	Pressure (p) field for the CFD model	88
5.4	Center-span Displacement Normalized by Euler Bernoulli Beam Theory	89
5.5	Applied Forces Normalized by Expected Load (4905 N)	89
5.6	Total OpenSees Constrained Node Reaction Forces and OpenFOAM Applied Forces	90
5.7	FEM nodal locations for case with 48 beam elements and 40 CFD cells in x direction	92
5.8	Total Reaction Force Errors - Mesh Convergence	94
5.9	Change of Loading Contribution with Displacement of Beams Under Shifting Water	95
5.10	FOAMySees Solution: center-span Displacement of $E = 1E6$ Pascals Beam Under Shifting Water Loading	96
5.11	Point A (1.5 s)	96
5.12	Point B (2.2 s)	96
5.13	Point C (4.7 s)	96
5.14	Point D (25 s)	96
5.15	FOAMySees Solution: center-span Displacement of Beams Under Shifting Water Loading	101

5.16	Normalized by Uniform Load Deflection: FOAMySees Solution for center-span Displacement of Beams Under Shifting Water Loading	102
5.17	Laminar Multiphase Hydrodynamic Impact Validation Case Setup	103
5.18	OpenFOAM Mesh and Initial State	105
5.19	Explicit Simulation Divergence - Strong FSIs in Partitioned Simulations Require Implicit Coupling Schemes for Stability	106
5.20	Effect of Timestep Reduction on Unconverging Model - With Too Few Coupling Iterations Provided, Simulations Experience High Fluctuations in Force	108
5.21	Simulation Instabilities Increase with Decreased Timestep for Models with Poorly Selected Coupling Settings	109
5.22	Variation in Coupled Solution Displacements With Implicit Coupling Timestep Size	110
5.23	Variation in Coupled Solution Forces With Implicit Coupling Timestep Size	111
5.24	Walhorn et. al Validation Case: FOAMySees Comparison to Other Solutions . . .	113
5.25	Walhorn et. al Fluid Free Surface Comparison: Left, grey background - FOAMySees. Right, white background - Walhorn et. al	113
5.26	CFD Model Mesh Distortion - Phase Fraction	114
5.27	CFD Model Mesh Distortion - Pressure	114
5.28	CFD Model Mesh Distortion - Velocity	114
5.29	Displacement Laplacian Study - Flap Tip Displacement	115
5.30	Displacement Laplacian Study - Applied Forces	115
5.31	Ground Motions Utilized in Preliminary Seismic Analysis	117
5.32	Preliminary Analysis Flap Tip Displacement for Cases with Base Ground Motion Excitation Prior to Fluid-Structure Analysis	117
5.33	Comparison Between FOAMySees Laminar Hydrodynamic Impact Solutions - Demonstration of Effects of Material Yielding and Initial Kinematic Conditions Due to Preliminary Seismic Loading Upon Flap Tip Displacement	118

5.34	Comparison Between FOAMySees Laminar Hydrodynamic Impact Solutions - Demonstration of Concurrent Seismic and FSI	119
5.35	Snapshot of FOAMySees Concurrent FSI Ground Motion Simulation	120
6.1	Floor Plan of Representative Full Scale Vertical Evacuation Structure: From this design, the prototype specimen which was utilized in testing and numerical analysis was developed. Figure modified from Pyke [2020].	123
6.2	Specimen Dimensions (Pyke [2020])	124
6.3	Specimen In Testing Configuration	126
6.4	Strain Gauge Locations	126
6.5	Specimen Location in Flume (Figure from Pyke [2020])	126
6.6	Wave Impact Forces - All Trials (Grey), Trial Averages from 1.40m (Solid) and 1.45m (Dashed) Wave Height Sequences: Force Comparison Between Load Cell Measurements and Strain Gauge Measurements	128
6.7	Model Dimensions	130
6.8	Side Elevation of OpenFOAM Model at Selected Times	131
6.9	145cm Wave Impact: Comparison	132
6.10	Test-Simulation Free Surface Height Comparison - 145cm Wave	133
6.11	Test-Simulation Velocity Comparison - 145cm Wave	133
6.12	Trial Averaged Upstream Face Pressures at Peak Base Shear (Time:9 seconds) . . .	134
6.13	Trial Averaged Pressure and Time-Integrated Pressure Time Histories for Two Representative Seaward Face Sensors, B06 and D11	135
6.14	145cm Wave Impact Forces: Comparison Between Trial Averaged Experimental and Simulated Streamwise Forces. Experimental streamwise forces are measured through load cells, pressure sensors, and strain gauges. Simulated forces were calculated with OpenFOAM and FOAMySees	140

6.15	145cm Wave Impact Forces: Comparison Between Trial Averaged Experimental and Simulated Overturn Moment. Experimental values are measured through load cells and strain gauges. Simulated forces were calculated with OpenFOAM and FOAMySees	141
6.16	145cm Wave Impact Forces: Comparison Between Trial Averaged Experimental and Simulated Principal Vertical Forces. Simulated forces were calculated with FOAMySees	141
6.17	145cm Wave Impact: Explicit and Implicit Work Into OpenSees from OpenFOAM - Explicit Simulation Experienced Instability, Resulting in Large Amounts of Erroneous Work Transferred	142
6.18	145cm Wave Impact: Explicit and Implicit Work Errors WRT Work Transfer Rate. The Explicitly-Coupled Simulation Experienced Instabilities, Resulting in Large Amounts of Erroneous Work Transferred, but This Erroneous Work is Transferred Almost Entirely to OpenSees	142
6.19	OpenSeesPy model overlaid by OpenFOAM CFD patch surface	145
6.20	Branch Clusters Utilized in Analysis (branch clusters are given color to show clusters more clearly. Clusters of similar color are <i>not</i> associated with one another)	145
6.21	OpenFOAM and OpenSeesPy Meshes at 0.7 s with fluid free surface	147
6.22	OpenFOAM and OpenSeesPy Meshes at 0.9 s with fluid free surface	147
6.23	OpenFOAM and OpenSeesPy Meshes at 1.1 s with fluid free surface	148
6.24	OpenFOAM and OpenSeesPy Displacement Over Time for Point of Maximum Lateral Displacement (Top of Left Front Column)	148
6.25	Streamwise Force Comparison Between Experiment, OpenFOAM CFD, and FOAMySees FSI	149
7.1	Work Across the Coupling Interface Over Time	155

7.2	Work Across the Coupling Interface Over Time, Large vs. Small Coupling Timesteps and Timestep Size Effect on Resolution of Work Transfer	156
7.3	Top: Explicit Scheme, Bottom: Implicit Scheme. Work Across the Coupling Interface Over Time - Left: Work In, Right: Work Out, Walhorn Case	158
7.4	Work Across the Coupling Interface Over Time - Top: Explicit Scheme, Bottom: Implicit Scheme. Work In Over Work Out, Walhorn Case	158
7.5	Top: Explicit Scheme, Bottom: Implicit Scheme. Work Resolution Error Over Time, Walhorn Case	159
7.6	Walhorn Case, Error in Work Across the Coupling Interface as a Function of Work Rate, By Timestep Increment Size and Coupling Scheme - Top: Explicit Scheme, Bottom: Implicit Scheme	160
7.7	Walhorn Case, Error in Work Across the Coupling Interface as a Function of Work Rate, Effect of Relaxation Factor on Work Transfer Errors	163
7.8	Walhorn Case, Effect of Relaxation Factor (α) on Number of Coupling Iterations Required for Convergence	164
7.9	Walhorn Case, Error in Work Across the Coupling Interface as a Function of Work Rate, Effect of Preconditioner on Work Transfer Errors	165
7.10	Walhorn Case, Effect of Preconditioner on Number of Coupling Iterations Required for Convergence	166
7.11	Walhorn Case, Coupling Settings Investigated	166

List of Tables

5.1	Closed Domain - Model Boundary Conditions for All Field Variables	87
5.2	Fixed Fixed Beam: Predicted $[WL^4/384EI]$ vs. Simulated Displacements, $W = 490.5[N/m]$, $L = 10[m]$, $I = 100[m^4]$	87
5.3	Closed Domain - Model Boundary Conditions for All Field Variables	104
6.1	Instrument Locations	125
6.2	Peak Forces and Moments by Study Parameter: F_x (Streamwise), F_y (Transverse), F_z (Lift), M_x (Roll), M_y (Pitch), M_z (Yaw)	127
6.3	CFD Model Boundary Conditions for All Field Variables	130
6.4	1.45 m Wave Force Comparison: Simulated vs. Experimentally Measured	139
7.1	Walhorn Dam Break Case Simulation Speed WRT OpenFOAM Parallelization and Computational Resources	168

Chapter 1

Introduction

Many coastal structures have the potential to be subjected to multiple hazards, such as the combination of strong earthquake motions followed by the forces induced by tsunami inundation. Little is known about the combined impacts of these hazards on structures. The present work hopes to add to techniques for assessing structural resilience to loading from multi-hazard events with the introduction of a methodology which utilizes previously validated simulation codes for coastal engineering and seismic engineering together for the solution of coupled fluid-structure-interaction problems.

This dissertation covers the development, testing, and some applications of an open-source nonlinear fluid-structure-interaction software tool comprised of Python, OpenFOAM, OpenSeesPy, and preCICE. The code has been named FOAMySees, which is a portmanteau of the latter halves of 'OpenFOAM' and 'OpenSees', and is comprised of the Open-source Field Operation And Manipulation (OpenFOAM, Weller et al. [1998b]) software package and OpenSeesPy (Zhu et al. [2018b]), a Python 3 interpreter of OpenSees (McKenna et al. [2010]). The programs are coupled via preCICE (Bungartz et al. [2016]), a coupling library for partitioned multi-physics simulation.

The FOAMySees tool was developed to assist engineers in coupled simulation of domain-approximating finite element models of civil engineering structures (three-dimensional structures

represented by center-line extruded beams, trusses, etc) subject to dynamic pressure loading from complex three-dimensional flows as calculated by a body-fitted mesh computational fluid dynamics model. The approach allows for the determination of complex three-dimensional time dependent pressure distributions for specific geometries subject to extreme hydrodynamic scenarios in turbulent conditions, the reduction of those pressure distributions to a set of forces which are applied to an FEA model, and the evaluation of the structural response of nonlinear finite element analysis models subject to such loading patterns within a partitioned strongly-coupled solution. The presented coupling methodology and research tool allow for multi-hazard simulation of a strongly-coupled solution between CFD models in three dimensions and conventional centerline-extruded nonlinear FEA models of civil engineering structures before, during, or after simulations of seismic excitation or other structural analyses, creating a flexible approach within which cascading hazards against structures can be investigated.

1.1 Approach

Computational continuum mechanics solver codes exist which provide capabilities for simulation of nonlinear FSI that may satisfy the desired objectives for performance analysis of civil engineering structures to loading from cascading or combined hazards. However, most software with these capabilities is packaged within expensive commercial software suites with limited or no access to source code. These limitations make such software undesirable for research purposes or development of specialized analysis routines.

Open-source software libraries popular with researchers within civil engineering are readily available and have been validated for the solution of various physical problems applicable to certain aspects of tsunami resilience design. Libraries exist for the solution of specific problems within engineering (turbulent multi-phase flow, nonlinear structural mechanics, elastic fluid-structure-interaction), but the solution of the disparate physics of structural mechanics and fluid dynamics

simultaneously as a monolithic calculation can be numerically challenging. Furthermore, there are few experimentally-validated open-source methodologies with all capabilities necessary to simulate strong fluid-structure-interactions between nonlinear dynamic structural response and multi-phase turbulent flows.

Thus, it may be difficult to utilize a single numerical method to suit the desired objectives of strongly-coupled fluid-structure-interaction for analysis of natural hazards. Software libraries which are specialized for the particular problems they are solving and are popular with civil engineers could be utilized in parallel and coupled together for the calculation of interactions between various physical solutions. If the software libraries utilized in coupling are modular or are part of a greater framework of specialized modules, coupling such modular libraries together could allow for a flexible methodology to perform any desired series of analyses on a given set of models.

OpenSees is a commonly used open-source framework for nonlinear FEM analysis of structures. OpenSees is often utilized for earthquake simulation, pushover analyses, and nonlinear time history analysis of structures. OpenSees has a long-standing history in the civil engineering field and has been validated against experimental results, providing confidence in the accuracy of the program (Agüero Ramón Llin et al. [2007], Bian et al. [2015], Furtado et al. [2015], Kolozvari et al. [2017], Zhu et al. [2018a], Vaiana et al. [2021]).

Similarly, OpenFOAM is used within the civil engineering community for CFD simulation of wind and water hazards to civil engineering structures, and has been validated against experiments and utilized for studies examining the behavior of waves and tsunami-like flows around structures (Douglas and Nistor [2014], Douglas et al. [2015], Wong [2015], Motley et al. [2016], Sarjamee et al. [2017a], Sarjamee et al. [2017b], Qin et al. [2018b], Winter [2019], Qin [2019], Croquer et al. [2022], Elsheikh et al. [2022a], Elsheikh et al. [2022b], Liu et al. [2022], Seiffert et al. [2015], Xiang et al. [2020], Winter et al. [2020], Yang et al. [2021], Dang et al. [2022]).

OpenFOAM and OpenSees are both free, scalable, and plenty of example cases exist for both codes, which allows engineers to utilize the tools handily and learn through trial and error. For

these reasons, OpenSees and OpenFOAM were selected as the FEM and CFD simulation software, respectively, to be used for this research. OpenSees was coupled to OpenFOAM through the use of an open-source coupling library, preCICE. A coupling driver, FOAMySees, was developed to initialize an OpenSees model from a standard OpenSeesPy script to be utilized as part of a strongly coupled FSI simulation. This coupling driver connects to preCICE to receive timestepping and force data from a parallel OpenFOAM simulation, applies loads to the initialized OpenSees model for the coupling timestep and advances the OpenSees model forwards in time, and sends displacements calculated from OpenSees to OpenFOAM through preCICE. The coupling driver also establishes analysis checkpoints for the OpenSees model such that implicit coupling methods can be utilized to stabilize the fluid-structure-interaction solution.

The geometries and nodal locations of a CFD model and a FEM model can be vastly different. Coupling of CFD surfaces to FEM nodes with FOAMySees is accomplished through the use of an additional mesh for data interpolation. FEM and interpolation meshes are connected through spatial clustering techniques performed on the nodal positions of each mesh. Forces and moments as calculated by the CFD model over the interpolation mesh are applied to the nodes of the finite element model for solution of structural dynamics over the next increment of time, and the motion of the structure is applied within the CFD model as a boundary condition, allowing for the calculation of the dynamic fluid forces imparted against the structure during its response to the imparted flow. This approach conserves work across the coupling interface and permits models of different external geometries to be coupled together for the purposes of strongly-coupled fluid-structure-interaction.

While FOAMySees is specialized to work with OpenFOAM, other CFD and FEM solvers could be coupled with OpenSees via FOAMySees by leveraging preCICE's suite of available adapters for other computational continuum mechanics solvers. The developed tool is specifically for OpenSeesPy generated OpenSees finite element models, as the main coupling code is structured within Python and uses preCICE's Python language API bindings. Similar procedures could possibly be conducted in another programming language, but that is beyond the scope of this thesis.

Coupling OpenFOAM and OpenSees offers a methodology for determining structure-specific tsunami forces at realistic scales while accounting for accumulated damage, which may be useful in evaluating the resilience of structures under extreme fluid loading. Site-specific tsunami inundation time histories or flow properties could be applied as input boundary conditions to a FOAMySees model of the structure, allowing for simulation of the nonlinear behavior of civil engineering structures under code-prescribed tsunami flow conditions. The approach can assist engineers in analysis of structures subject to various hazards such as earthquakes and tsunamis, extreme wind events, storm surges, and floods.

1.2 Research Objectives

This work discusses the motivation for and procedures followed to couple simulations used for nonlinear structural analysis and earthquake engineering within a finite element analysis (FEA) software framework (OpenSees) to computational fluid dynamics (CFD) models commonly used for simulation of turbulent flows and fluid behaviors which generate forces of interest to designers of civil engineering structures (OpenFOAM). The research tool was verified against several test cases, and the flexibility of the developed tool for various use cases was demonstrated. This thesis elaborates steps taken by the author in implementing the coupling approach and the methodology used for open-source simulation of the interaction between models commonly utilized for civil engineering analysis of structures and multi-phase flows.

The main objectives of this research are to:

1. Identify and implement an approach for the deterministic simulation of tsunami-like flows and coupled fluid-structure-interaction using validated open-source software components
2. Outline the steps taken to develop the research tool for investigation of physical problems involving nonlinear fluid-structure-interaction (FSI)

3. Investigate the accuracy of the coupling approach with respect to a series of analytical and experimental test cases
4. Demonstrate the capabilities of the approach with regards to multi-hazard analysis of structures
5. Offer modelling recommendations and considerations of utilizing the approach for general FSI simulation
6. Discuss limitations and applications of the approach, and
7. Suggest points of future improvement for the approach and future work which could be conducted with the developed research tool

1.3 Scope of Research

The presented research covers a verified approach to simulation of strongly-coupled fluid-structure-interaction using validated modelling methodologies for the coupled sub-problems. A tool was developed which allows for the evaluation of the response of finite element models to a series of staged or concurrent nonlinear earthquake and FSI analyses while preserving structural state and damage information between analyses. The tool was designed to expand the current applications of validated open-source modelling methodologies with no modification of source code of component analysis software, allowing for a flexible connection between solutions conducted in a computational fluid dynamics model and a finite element model through the use of a coupling mesh across which work is passed.

This Introduction is followed by a chapter covering the background of and motivation for the presented research (Chapter 2). A review of the numerical methodologies utilized in the development and deployment of the proposed fluid-structure-interaction coupling approach as components of the coupled solution is then provided (Chapter 3). Following this is the main chapter of the thesis

(Chapter 4), wherein a technique is proposed for reconciliation of work preservation between non-conformal geometries (Section 4.1). The approach utilized to implement the technique as part of the research tool is then described along with computational details and sequences of analysis procedures (Section 4.2), provided such that the approach is clear to others aiming to make improvements or implement the approach in a different manner. A series of analytical cases which demonstrate the usability of the code within a limited context are provided, with which the accuracy of calculated solutions with respect to analytical predictions and solutions from other fluid-structure-interaction simulation software is investigated (Chapter 5). The tool was also tested with dynamically applied ground motion boundary conditions applied to the structural analysis model prior and during the FSI analysis as examples of the tool's capabilities for multi-hazard analysis. An experimental-numerical correlation study using data from large-scale small-deformation tests conducted at the University of Oregon within the O.H. Hinsdale Wave Research Laboratory Large Wave Flume was then modelled with the tool to demonstrate usability of the approach for realistic engineering applications (Chapter 6). Challenges, drawbacks, limitations of the methodology, and future points of improvement of the research tool are then elaborated (Chapter 7), along with considerations for modelling fluid-structure-interaction problems with the presented approach. Possible applications of the approach are then discussed (Chapter 8), along with conclusions (Chapter 9). References are then provided along with an Appendix (A) containing pertinent information about utilizing the developed coupling approach and files for models presented within the aforementioned chapters.

Chapter 2

Background

This chapter presents the background for the conducted research into coupling of open-source libraries for the purpose of simulation of nonlinear fluid-structure-interaction.

2.1 Conventional Approaches to Tsunami Resilient Design

Civil engineering design standards offer guidelines for the evaluation of a given structure to site-specific loading from tsunami inundation events. A statistically dependable methodology for determining overall magnitudes of force which can be imparted to structures from probabilistic or model-generated tsunami characteristics at a given site is the ASCE/SEI 7-22 Minimum Design Loads and Associated Criteria for Buildings and Other Structures provisions ASCE [2022], referred to from this point forward as "ASCE 7-22." ASCE 7-22 design procedures implement Load and Resistance Factor Design procedures for newly constructed buildings and for performance assessment of certain structures to ensure building resilience to expected combinations of loads to a structure from various sources. Probabilistic methods, empirical force equations, and theoretical force approximations offer estimates for peak resultant force magnitudes from a tsunami event which can then be utilized in procedures for the design of structures at risk to tsunami loading. A summary

of ASCE 7-22 Chapter 6 procedures for tsunami resilient design is available in FEMA [2019]. Advances made in the procedures from those in ASCE 7-16 Chapter 6 are elaborated in Robertson [2023]. An overview of ASCE procedures for tsunami resilient structural design is provided here for background to the discussion of the motivating factors behind the research conducted into coupled FSI for multi-hazard structural resilience applications.

2.1.1 Standard Probabilistic Approaches for Determination of Tsunami Hazards and Loading to Structures

Probabilistic hazard methods utilize large amounts of historical data of seismic and tsunami risks from multiple sources to quantify expected risk to structures from certain hazards within particular geographic regions. Tsunami risk to a given site can be evaluated through probabilistic methods or numerical and statistical modelling of tsunami propagation and inundation from deterministic tsunami source scenarios.

From calculated tsunami characteristics for a particular site and tsunami hazard (tsunami inundation height, velocity, run-up height), empirical equations developed from test data or theoretical formulae derived from the Navier-Stokes equations for fluid transport can be utilized to approximate the fluid forces generated by a tsunami inundation flow against a given structure.

Approximation of Site Tsunami Characteristics

Methods for evaluation of tsunami source generation and propagation modelling (PTHA, Geist and Parsons [2006], Geist and Lynett [2014]) can be utilized to calculate expected tsunami flow heights and velocities for a given seismic hazard and fault source. Tsunami generation and propagation information is aggregated for multiple hazard sources and scenarios to establish hazard maps of a given region to inform about tsunami risk to structures. The regional assessment of tsunami hazards is not of specific interest in the present work, thus, further discussion of the history, gaps in research,

and advances within this field over the last decade are left to other sources (Behrens et al. [2021]).

Estimation of Tsunami Forces at Design Scales from Maximum Considered Tsunami Inundation Characteristics

Structures are classified through ASCE 7-22 designated criteria to establish required levels of resilience to a particular tsunami hazard depending on the occupation type and location of the structure. Regions of expected tsunami inundation are mapped and zoned by ranges of predicted tsunami intensities (Tsunami Design Zones) with respect to inundation risk from the aggregated expected Maximum Considered Tsunami ('MCT', tsunamis of large magnitude which reoccur at a maximum period of every 2450 years) Chock et al. [2016]. ASCE requires that structures be designed to a tsunami hazard level that has a 2 percent chance of exceedence within 50 years, a prerequisite that aims to preclude the possibility of collapse for a structure by ensuring that conservative estimates of force are utilized for structural design. Tsunami Risk Category (TRC) of a building is assigned based on structure occupancy type and other considerations. Tsunami risk categories are I-IV with IV representing critical facilities and structures and I representing structures with low occupancy or importance. Tsunami design 'importance factors' (Sec 6.8 Table 6.8-1) are associated with these risk categories to apply more conservative estimates of load as Risk Category increases. ASCE 7-22 demands more rigorous analysis procedures for structures of higher Risk Categories, and loads are increased depending on the importance factor of the particular structure. With these considerations for structural location and occupancy type, the capacity of newly-designed or retrofitted structures with respect to the demands from a maximum considered tsunami is increased through element strength reduction factors and applied load amplification factors, raising the resilience of structures to preclude catastrophic structural failure during extreme loading events. Though not all structures are subject to tsunami design requirements (only TRC III and IV), from Robertson [2023], *"local jurisdiction is encouraged to require tsunami design for TRC II buildings with sufficient height to provide emergency refuge for people stranded within the TDZ."*

Following ASCE 7-22 procedures, designers utilize approximate tsunami height and velocity characteristics obtained from probabilistic tsunami source generation methods or risk maps to calculate maximum considered tsunami run-up height and velocity at a structure. Site-specific tsunami velocities and inundation heights can be calculated using local topography and bathymetry information and energy-based methods (energy grade line analysis) or inundation modelling (Sec. 6.7.5). Various modification factors are available within ASCE 7 to amplify flow velocity depending on congestion of the region and to account for the effects of upstream structures (Sec. 6.7.6).

Calculated local tsunami flow properties are then used along with theoretically and empirically derived formulae for hydrostatic and hydrodynamic force magnitudes and distributions imparted to affected components of a structure during several tsunami cycles (inundation and receding flow from tsunami waves). The forces accounted for by ASCE 7 include tsunami bore entrapment forces, flow stagnation pressures, tsunami bore impact forces, hydrostatic bearing forces, debris impact forces, debris damming forces, buoyancy forces, and scour effects, with equations and factors provided to modify these forces depending on various expected conditions.

These forces and effects are modified by factors accounting for structural external closure ratio (how 'open' the structure is to flow), the direction of force application (± 22.5 degrees from direction of primary flow), and the density and specific weight of sediment-laden tsunami flow (for tsunamis the amount of sediment carried can be significant and thus an amplification factor of 1.1 on the typical density of seawater). Considerations for development of alternative load-resisting paths in a structure in the event of debris-induced local failure are also included in Chapter 6.

Three load cases are considered for tsunami loading to structures within ASCE 7 design guidelines. The three load cases considered are:

1. Load Case 1: Combined Hydrodynamic Drag with Maximum Buoyancy Force During Initial Inundation to Consider Maximum Overturn and Sliding Forces

2. Load Case 2: Combined Hydrodynamic Drag Including Effects of Debris Damming, at Max.

Flow Velocity (assumed to act at 2/3 maximum site inundation depth)

3. Load Case 3: Combined Hydrodynamic Drag Including Effects of Debris Damming, at Max.

Flow Depth (assumed to act at 1/3 maximum site inundation *velocity*)

The forces calculated to each component from each load case are then applied to a structure using appropriate load combinations as defined by ASCE 7 to assess the probabilistic combination of tsunami loading to all elements as prescribed by the design procedures. The load combinations for tsunami design as prescribed by ASCE 7-22 Ch. 6 Eqns. 6.8.3.3-1a and 6.8.3.3-1b are listed in Eqns. 2.1 and 2.2.

$$0.9D + F_{TSU} + 1.0H_{TSU} \quad (2.1)$$

$$1.2D + F_{TSU} + 0.5L + 0.2S + 1.0H_{TSU} \quad (2.2)$$

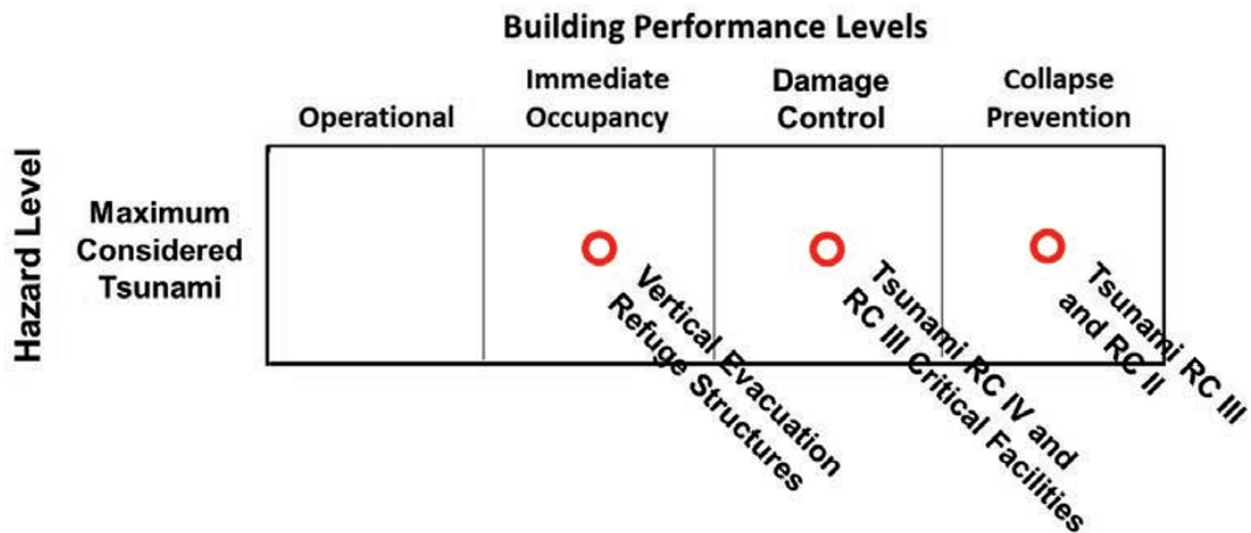
In these equations, D , L , S , F_{TSU} , and H_{TSU} , respectively, are the force patterns imparted to the structure by various sources, including 'dead' loads, 'live' loads, snow loads, the combined lateral forces the the structure, and the combined vertical forces to the structure.

ASCE 7-22 Tsunami Design provisions (Chapter 6, ASCE [2022]) offer conservative estimates of the tsunami-induced forces to a structure and target structural reliability analysis procedures which should be followed to ensure satisfaction of desired building performance measures (Chock et al. [2016]). The estimates for tsunami-induced forces are experimentally-validated (Robertson et al. [2013], Robertson [2014]), and have been built upon decades of research of tsunami inundation events and the damage imparted to structures within their region of effect (Carden et al. [2012], Chock [2015]). Chapter 6 of ASCE 7-22 offers a suite of tsunami force demand equations and design provisions which are intended to improve both tsunami resilience and overall life safety of coastal communities and their buildings (Chock et al. [2013], 2018; Chock [2016], Foster et al. [2017a]).

Building Performance Requirements and Performance Analysis

ASCE 7-22 Chapter 6 Tsunami Loads and Effects procedures when utilized properly can offer guidelines for development of structural designs which preclude catastrophic failure and ensure basic life-safety of occupants during a natural hazard event. Life-safety in ASCE is defined as *"The post-event damage state is that in which a structure has damaged components but retains a margin against onset of partial or total collapse."* 'Life-safety' performance objectives are quantified into the 'Collapse-Prevention' category for Tsunami Building Performance. Building Performance Requirements are shown in Fig. 2.1 from FEMA P-646 for buildings of different Risk Categories as defined by ASCE 7.

Fig. 2.1. Tsunami Building Performance Objectives By Risk Category



Methods for determining performance of structures to tsunami-induced forces include nonlinear dynamic structural analysis and tsunami pushover analysis of structural systems. Pushover analyses can be used to examine the inelastic behavior of systems when under sustained lateral acceleration or forces which could cause a structure to be 'pushed' over to the point of collapse. An expected parameter of the tsunami (the inundation height and velocity, the tsunami run-up elevation, or maximum expected tsunami shear force) is used as a variable in calculating maximum expected

lateral forces and force distributions imparted to a structure during a given probabilistic tsunami event which are then applied incrementally until no load-resisting paths remain in the structure. Varying approaches to pushover analyses have been conducted in studies of structures subject to tsunami loading and efforts have been made to implement nonlinear pushover analysis procedures within ASCE-7 design guidelines (Rossetto et al. [2018], Petrone et al. [2020], Xu et al. [2021], Baiguera et al. [2019], 2022).

Performance Assessment and Limit State Analysis of Structures

Nonlinear time history analyses and pushover analyses of structures can offer large amounts of information about building performance. These methods can be utilized to quantify the expected response of a structure with respect to designer or code-prescribed building performance measures, commonly referred to as a 'limit-state-analysis.' The 'limit-state' of a structure can be assessed in many ways, but generally through ASCE 7 procedures limit states are determined by the structure's satisfaction of the design and performance requirements as elaborated by the ASCE 7 design code for the structure's expected hazards, Risk Category, and occupancy type. Inter-story drift ratios are commonly utilized to assess a structure's performance to a given loading. For most structures, the amount of rotation experienced at a connection or drift in between stories can be utilized to assess the degree of damage that a structure's components will experience.

The performance of a structure to design-level tsunami loading can be assessed from results of multiple quasi-static pushover analyses or nonlinear response history analyses (NLRHA). Structure-specific vulnerability assessments can be performed on results from suites of pushover analyses or nonlinear time history analyses examining structural performance over various hazard intensity measures (Stephens et al. [2023]). A parameter which represents the tsunami hazard intensity is varied and the response of the system is recorded for each parameterized value allowing for creation of a function of the maximum structural demands and characterization of response with respect to the magnitude of a given parameter. The parameter quantifying a hazard's strength is commonly

referred to as an "intensity measure" - for tsunamis, the intensity measure may be tsunami height and velocity or peak tsunami shear force. Realistic ranges of hazard intensity measures to be utilized for tsunami pushover analyses can be estimated through probabilistic methods for tsunami source generation and energy-based tsunami run-up calculations or from site-specific inundation analysis.

Building Damage Assessment from Tsunami Risks

As the lateral forces imparted to structures from tsunamis and earthquakes can be roughly of the same magnitude in some cases, efforts have been made to extend validated metrics for seismic building performance to be applied to tsunamis. Fragility functions utilized for the assessment of earthquake or tsunami damage to structures are supported by field research conducted in the aftermath of historical natural hazard events and surveys of structural damage imparted to buildings within their regions of effect (Charvet et al. [2015]). Rigorous regression analyses of the damage states observed from structures subject to various measures of tsunami intensity are performed to establish fragility functions, which can then be utilized to estimate structural damage depending on expected hazard intensity at a given site from a particular scenario. Probabilities of failure of given structural types when subject to various intensity measures of multiple types of natural hazard have been quantified. A comprehensive guide to building damage functions established for various structure and occupancy types is available in FEMA [2017] Section 5.5. FEMA HAZUS guidelines aim to provide engineers with a metric for assessing risk of damage to structures subject to seismic and tsunami hazards.

Leveraging building damage functions or fragility curves, the risk of damage to structures can be assessed for hazards from multiple seismic and tsunami sources using combined probabilistic analyses. Several approaches for evaluation of structural resilience to multi-hazards are well-established in literature, including performance based tsunami engineering (PBTE, Park et al. [2012], Park et al. [2013], Attary et al. [2017], Attary et al. [2021]) probabilistic seismic and tsunami damage analysis (PSTDA, Park et al. [2019], Goda and De Risi [2023], Alam [2019], Burns et al.

[2021]), and performance based coastal engineering (PBCE González-Dueñas and Padgett [2021]). Probabilistic hazard analysis methods for assessing risk can provide meaningful information about general resilience of communities to a maximum considered seismic and tsunami hazard.

2.2 Emerging Approaches for Tsunami-Resilient Design

This section discusses the motivating factors for research into development of an open-source approach for multi-hazard analysis of structures. Existing standards and analytical approaches cover a large amount of tsunami hazard and loading circumstances, but uncertainties remain about the resilience of structures not subject to ASCE 7 tsunami design requirements due to their Risk Category designation or existing seismic capacity. Additionally, further work is needed to understand the effects of local topographic and geometric conditions on the resilience of structures to tsunami hazards. This applies especially to structures not designed to current standards which may reasonably be expected to provide refuge when subject to conditions that demand building performance to ensure life-safety. The design standards for tsunamis are relatively new, which means that there could be many structures which will meet not tsunami resilience requirements for life-safety when their performance is assessed.

2.2.1 Investigation of Effects of Time-Accurate Pressure Profiles from Incompressible Hydrodynamic Flows on Calculated Structural Resilience

Pushover analyses can be used to investigate nonlinear structural response to approximated loading patterns from velocities calculated by the site-specific inundation analysis. Development of tsunami loading patterns requires careful consideration of the flow behavior in and around the structure during various points of the inundation, which itself demands extensive experience and knowledge of the behavior of fluid loading. Loading patterns (typically uniform, triangular, inverted triangular, or trapezoidal distributions) are applied to structures based on assumptions of the tsunami behavior

concerned with the associated components of force at the structural interface. These different patterns also can be implemented in different application procedures, which can result in variation in calculated resilience of a structure.

Constant depth pushover analyses apply the maximum expected tsunami loading to the structure as a boundary condition for the pushover analysis and is a conservative estimate of the tsunami shear forces. Variable depth pushover analyses offer more reasonable approximation of the expected shear demands to the structure when compared to results from dynamic time history analyses, and are considered an improvement upon the constant depth method for tsunami pushover analyses (Petrone et al. [2017], Rossetto et al. [2019]). Research into the effects of time-dependent vertical buoyancy loads on the resilience of RC frames demonstrate that incorporating buoyancy loading during a pushover analysis can reduce the inundation height at which failure occurs by as much as 14 percent (Del Zoppo et al. [2020], Del Zoppo et al. [2023]).

These factors indicate that additional work could be done to investigate the effects of load pattern form and application method on the calculated resilience of structures to sustained lateral loading derived from approximations of MCT conditions and to validate these loading patterns for general use in all cases, particularly circumstances which might result in complex forms of monotonic loading to a structure from surrounding topographic and geometric factors.

2.2.2 Assessment of Topographic and Geometric Effects on Tsunami Loading to Structures

Site-specific inundation analyses offer flow height, velocity, and momentum flux time histories at a given region from probabilistic tsunami scenario, from which loading profiles for tsunami pushover analyses can be generated. Despite this, conditions and forces experienced by a structure during a tsunami inundation are largely subjective, as is individual structural resilience. Temporal and steady state tsunami inundation flow behavior around a structure and the locality and direction of forces

imparted by the flow to the structure are contingent on the surrounding topography as well as the structural geometry.

Large variations in the local velocities of a tsunami can occur as the tsunami propagates through a congested region of structures or around obstacles. Experimentally-validated CFD simulations of arrays of structures conducted indicate that momentum-flux methods for approximation of tsunami forces do well to predict overall magnitude of imparted stream-wise forces for structures subject to direct flow conditions, but uncertainty remains in the effects of local topographies on the load profiles generated to structural geometries during various stages of inundation (Qin et al. [2018a]). Channeling, shielding, and reflections of the flow from obstructions neighboring a structure can create velocity changes which induce changes in measured pressure distributions and thus resultant force patterns to a structure from predictions assuming uniformly applied flow (Winter et al. [2020]).

Care must be taken in both the assessment of forces to specific structures in particular topographic configurations and the current structural state to preclude errors in approximation of structural resilience to tsunami imparted forces. Fluid dynamics simulations of tsunami inundation to a structure offer large amounts of information which cannot be obtained from probabilistic approaches, risk maps, or experiments and are useful for assessment of geometry-and-topography-specific tsunami forces imparted to critical structures. Simulation of flow in the near region of a structure using partial differential equations representing 3D fluid dynamics could aid in higher fidelity calculation of force application patterns to structures located in congested regions.

2.2.3 Deterministic Assessment of Resilience of Structures Expected to Provide Refuge in Tsunami Conditions

There will likely be locations and tsunami hazards where evacuation of an entire populace from tsunami inundation is not possible, and evacuees who cannot escape inundation regions in these cases would take refuge in structures which may not be specifically designed to resist tsunami

loading. It is possible that buildings tall enough to provide levels above the inundation height in these cases might be sought out by evacuees as locations of refuge regardless of the structure's Risk Category designation and building performance objectives, raising potential risks to life-safety.

Carden et al. [2016] conducted rigorous analyses of the expected tsunami inundation and resilience of nearly 500 structures in several coastal U.S. states at risk to tsunami risks through comparisons between ASCE seismic design requirements for structures and the expected maximum forces from a MCT as calculated by ASCE 7 design procedures. Structures in Risk Category II were examined, as many residential and commercial structures of sufficient height are placed into this category and are subject to adequate seismic design requirements to be comparable to requirements applied to structures for tsunami resilient designs. The calculations for maximum expected tsunami force were determined from seismic-source specific PTHA and inundation modelling to develop site-specific tsunami inundation maps (Tsunami Design Zone maps in ASCE 7-22) to be utilized for quantification of expected forces against structures. A primary assumption made was that buildings studied were current to existing seismic design code. Existing seismic design requirements for Seismic Design Category D in Oregon, California, and Washington were found to provide sufficient design strength to resist expected tsunami forces when the structure in question was tall enough to possess at least one story above the MCT inundation height. However, particular TDZ regions exist wherein the magnitude of MCT loading exceeded that of the seismic demand forces and thus the capacities of structures within these regions may need to be specifically designed to resist the tsunami loading (Carden et al. [2016]).

As egress from the inundation region may not be possible, local jurisdictions of cities and regions subject to effects from tsunamis have been encouraged to adopt building codes which will maximize the number of structures subject to more rigorous analysis procedures such that alternative means of evacuation from tsunamis is provided to a community through tall buildings of Risk Category II (most structures fall within this Risk Category).

Due to the large and uncommon demands of tsunami-induced loading to various parts of

structures, it is not known if stability of typical structures or structural elements which are not specifically designed to resist tsunami-specific hazards will be compromised even if lateral force resisting systems are sufficient in strength to resist forces induced by a tsunami. Failure of particular structural components such as exterior columns could possibly result in progressive collapse of reduced structural performance with respect to that expected solely based on the lateral force resisting system Baiguera et al. [2022]. Finite element models have been utilized with success to study sequential earthquake and soil-fluid-structure interactions of nonlinear structures, indicating that moderate to severe damage accumulated during a preceding earthquake can affect the resilience of a structure during a subsequent tsunami inundation, and thus cannot be neglected (Cortez et al. [2022]). Thus, means for investigating building-specific response to site-specific ground motions followed by three-dimensional pressure profiles arising from their interaction with incompressible dynamic flows in and around structural geometries could assist in characterizing weaknesses in existing structures to tsunami hazards.

2.2.4 Validation of Probabilistic Approaches for Combined Hazard Analysis

Risk to a particular structure for a certain tsunami hazard is calculated through establishment of maximum considered tsunami hazard intensity, measures for a particular site and comparison of expected tsunami intensity measures with estimated lateral force resisting system strengths and associated lateral force capacities of structures within the tsunami region of effect. Recently, efforts have been made to quantify resilience of structures to multi-hazard events through statistical analysis and probabilistic methods, combining existing probabilistic seismic hazard models with those developed for tsunamis.

As metrics for building damage to hazard intensity are largely derived from field observations of past hazard events, large uncertainties remain in these approaches and care must be taken when applying them. The low frequency of occurrence of tsunami events of design-scale magnitude results in a limited number of source events from which resilience data can be drawn for a given

region. Furthermore, for any structure the resilience likely depends on site-specific conditions, the quality of construction, the design of the structure itself according to design standards at the time of construction, preventative measures undertaken to reduce structural damage, and many other factors. This results in large variability in the expected resilience of structures when classified solely by the predominant construction material, which may need to be carefully considered through additional analyses.

Building damage metrics derived from historical tsunami events and their imparted damage to structures may not be applicable to new forms of construction, new structural lateral force resisting systems, or to structures about which there is limited data pertaining to tsunami damage. Construction methods and design codes vary by geographic region, and similarly the structures in various geographic regions may possess differing levels of resilience to particular forms of loading when compared to buildings designed to ASCE 7 standards. As such, care must be taken to assess the resilience of new construction to cascading hazards and resilience of structures in regions which have not previously been subject to a tsunami.

As individual structures will possess different levels of resilience to combined hazards, the use of combined probabilistic fragility functions for the assessment of building damage from combined hazards may not be applicable to structures generally without consideration of individual building resilience to individual hazards and their expected residual capacities following a deterministic realization of a probabilistic seismic hazard scenario. The present method offers means to conduct pushover analyses and nonlinear time history analyses of structures subject to three-dimensional forces following seismic excitation. This will allow designers to assess the residual capacity of structures to forces whose direction, magnitude, and distribution over the faces of the structure are true to those which will be experienced by the structure in-situ during inundation. This offers an alternative approach to assess the resilience of structural forms to tsunami-like loading, such that existing approaches for probabilistic damage assessment of structures can be verified or validated.

2.2.5 Summary

In summary, structural response and resilience to tsunamis could be better understood if investigation of the fluid-structure-interaction of structural systems and incompressible multi-phase flows is conducted. An accessible approach for conducting nonlinear analyses of structural response to forces calculated by 3D fluid dynamics simulations would offer means to assess resilience of structures to specific fluid forcing conditions that are difficult to quantify with approximate methods or static approximations, lowering uncertainty in calculated resilience.

Additionally, cascading earthquake-tsunami multi-hazard events present a series of extreme loading scenarios, where damage sustained during the earthquake could influence structural performance during the subsequent inundation. If earthquake simulations could be conducted on a nonlinear model of a structure, followed by a coastal region inundation fluid-structure-interaction simulation, the force and displacement response of all structural elements in a structure could be analyzed during specific design-level cascading and combined hazard scenarios.

In the case of yielding of structural elements or main load-resisting structural components, FSI simulation also allow for the evaluation of alternative load-resisting paths such that structures can effectively resist sustained tsunami forces without catastrophic failure. Thus, methods for evaluating site-and-geometry-specific performance of existing structures to extreme hydrodynamic loading while maintaining damage from preceding seismicity could help engineers to improve overall life-safety and understand methods for retrofitting structures to be resilient when subject to cascading hazards.

Chapter 3

Methods

The methods utilized in the implemented coupled FSI simulation methodology are:

- the finite element method for nonlinear structural analysis (Section 3.1),
- the finite volume method for computational fluid dynamics (Section 3.2), and
- partitioned coupling (Section 3.3).

The coupling is implemented with k-means clustering and k-nearest-neighbor search, as well as Python class programming for calculations and for management of coupling data, communication between programs with TCP/IP sockets, and use of Python language bindings to multiple C++ libraries. These methods are described in the following sections.

3.1 Finite Element Method

3.1.1 Overview

The finite element method is a numerical method developed in the mid-1900s (Turner et al. [1956]) for the solution of partial differential equations (PDEs). The finite element method (FEM) relies on

the principles of the method of weighted residuals, the calculus of variations, and/or the theory of minimum potential energy. Applying the finite element method to a PDE results in the reduction of continuous field equations and functions into spatially discretized arrays and vectors, allowing for the solution of PDEs by means of linear algebra methods and by computers through numerical techniques.

Finite elements can be derived in a variety of ways depending on the approach utilized, but all function to minimize the potential energy of a given system that is under inspection. Finite element analysis (FEA) allows for calculation of the solution of PDEs through satisfaction of the 'weak form' of a given continuous partial differential equation. This is accomplished by means of the utilization and integration of interpolation (or 'shape') functions between discretization points of a continuous domain. These shape functions (and their order) are selected for a given differential equation to allow for the order of interpolation required such that the resolved field variables and their derivatives fully capture the physics and satisfy the continuity, completeness, and compatibility requirements of the equation(s) to be solved.

Spaces between selected discretization points within the discretized continuous domain are classified into regions called 'elements,' over which the fluxes or fields of the given governing equation are calculated. Continuous fields are integrated over the shape functions within the finite elements to be calculated as discrete values that are then applied at the continuum discretization points from which the finite element is constructed. Similarly, discrete nodal values are interpolated to represent continuous fields by means of utilizing the interpolation functions to distribute discrete values from nodes at which the solution is calculated to the spaces within the element region. A number of numerical integration points and locations (Gauss points) are selected along these elements' spatial domains to allow for the satisfactory calculation of integrated values within a shape function through numerical integration.

The finite element method has been generalized since its inception and validated for many purposes, including both linear and nonlinear structural analysis of civil engineering structures.

In structural engineering and dynamics, the finite element method is used to approximate the stiffness, damping, and mass of continuous structures at a series of selected points, maintaining the relationships between the integrated quantities and their locations by means of correlation matrices, or symmetric arrays, which are sized by the number of degrees of freedom for a given discretization and governing equation that has been discretized. The generation of matrices for each structural quantity (mass, damping, stiffness) can then be utilized as components in the solution of ordinary differential equations over the multiple degrees of freedom of the structural domain, namely the characteristic equation of structural motion (see below, $\underline{x} = \text{Vector}$, $\underline{X} = \text{Matrix}$):

$$\underline{f}(t) = \underline{M}\ddot{\underline{x}}(t) + \underline{C}\dot{\underline{x}}(t) + \underline{K}\underline{x}(t) \quad (3.1)$$

where

$\underline{f}(t) = \text{Vector of Externally Applied Forces}$

$\ddot{\underline{x}}(t), \dot{\underline{x}}(t), \underline{x}(t) = \text{Acceleration, Velocity, and Displacement Vectors}$

$\underline{K}, \underline{C}, \underline{M} = \text{Stiffness Matrix, Damping Matrix, and Mass Matrix}$

For linear systems, solution of Eqn. 3.1 most generally involves the modal decomposition of the equation for the characteristic equation of structural motion through the use of eigen-analyses, 'mass-normalization', and orthogonalization of the equation into principle components. This allows for the utilization of the principal deformation shapes (mode shapes of the structure) as single-degree-of-freedom oscillators (with stiffness, damping, and mass) to be integrated over time using standard methods for the solution of dynamical ordinary-differential equations. The magnitudes of the oscillations of the mode shapes of the system can then be projected back to three-dimensional space through the use of the calculated principle modes of deformation. For more detail on numerical routines utilized for finite element analysis of structures, structural dynamics of multiple-degree-of-freedom systems and otherwise, see Bathe [2006], Zienkiewicz et al. [2005],

and Chopra [2017].

3.1.2 Open System for Earthquake Engineering Simulation (OpenSees)

Originally developed by researchers at the Pacific Earthquake Engineering Research (PEER) Center, OpenSees, the Open System for Earthquake Engineering Simulation (McKenna et al. [2010]), is an object-oriented, finite-element-analysis library containing numerical routines, methods, and techniques commonly utilized for the linear and nonlinear solution of the structural dynamics equations described previously. OpenSees operates as a framework upon which structural models or applications can be created for the purpose of earthquake engineering. For the sake of clarity, software classes, objects, and methods part of the *OpenSees* Framework will be shown in ***bold and italicized*** text.

OpenSees Classes

OpenSees leverages a class structure which delegates various components of the finite element solution and assembly of the finite element model to various discrete parts of the computer program in memory. OpenSees utilizes these classes in a way which allows for the 'uncoupling' of the domain and state of a finite element method solution from the time integration and analysis techniques utilized for a given segment of a structural analysis. This is broadly achieved through utilizing broker objects to keep components of the solution (namely the domain and the analysis routines) separate enough to allow for flexibility and mid-solution modification of the analysis routines and 'commitment' of the state of the model after convergence to an object specifically constructed to store model states.

This aspect of OpenSees allows for high flexibility of the solution routines for a given finite element model, and is the hallmark feature of the library as this allows for development of sections, elements, solution algorithms, and applications of OpenSees which are entirely discrete from other components of the program. This means that a given civil engineering or structural engineering

programmer does not have to be an expert in computer science to develop and implement an element formulation or structural engineering component that can be utilized in OpenSees.

Each class or object in the OpenSees library handles a different component of the overall construction of finite element model data or calculation of nonlinear structural analysis and/or dynamics in the context of the finite element method. In OpenSees, these objects can loosely be grouped into two categories - objects which handle the calculation of finite-element associated quantities for construction of the system of equations and discretization of the domain, and those which handle construction of the systems of equations and the solution of the dynamics of the system which is being modelled (the former and latter described in McKenna et al. [2010] under sections 'Classes for Representing Finite-Element Equations' and 'Strategies for Solving Finite-Element Equations', respectively).

The main classes of OpenSees are shown in Fig. 3.1. Within this section, the names of OpenSees classes will be formatted with ***bold and italic text***.

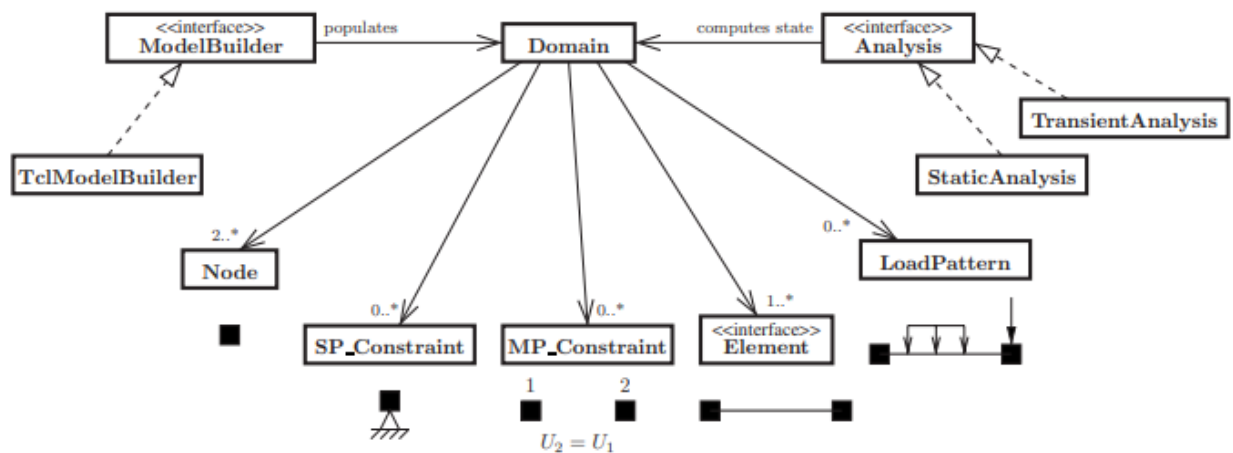


Fig. 3.1. OpenSees Class Structure (Figure obtained from McKenna et al. [2010])

A ***ModelBuilder*** class is comprised of routines, which build the arrays required for static and dynamic analysis of structures via the finite element discretization, as well as definition of nodes, elements, sections, load patterns, and constraints for various degrees of freedom of the model. An

object of the **Domain** class holds the state of the model in memory, and allows access of other objects of particular classes to the nodes, elements, load patterns, time series, and constraints as defined by the **ModelBuilder** class, as well as the current state of the model. Objects of the **Analysis** class are abstract (meaning its components are constructed of other objects, rather than inherited from another class) and are aggregated from sub-classes that are not abstract. These sub-classes include the **Integrator**, **ConstraintHandler**, **SolutionAlgorithm**, **ConvergenceTest**, **SystemOfEqn**, and **Solver** classes. These classes are associated with the **Analysis** class as shown in Fig. 3.2.

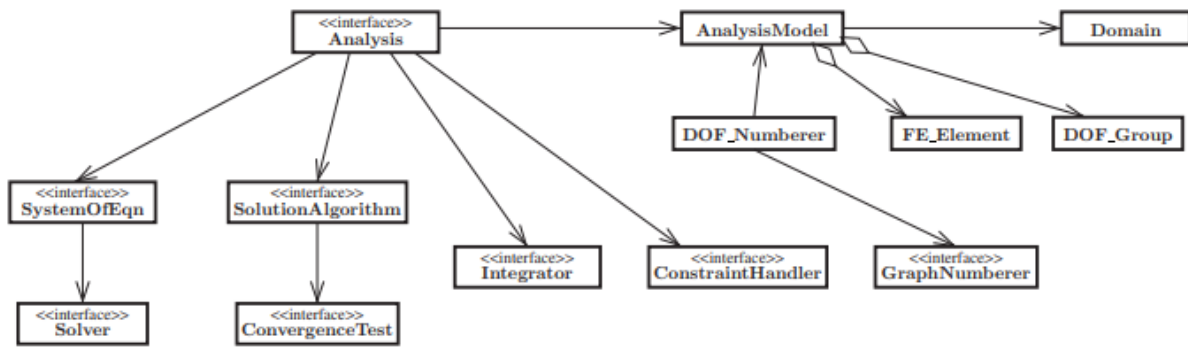


Fig. 3.2. OpenSees **Analysis** Class Composition (Figure obtained from McKenna et al. [2010] and Utilized Here to Supplement Discussion of OpenSees Class Structure)

The details of each of these classes are specified by the OpenSees user for a given analysis through construction of an object of each type. Once these sub-classes have been constructed, the **Analysis** class object can be assembled and utilized to solve for the model's next state. This method of comprising analysis objects via aggregation rather than inheritance is useful due to the flexibility it offers in the definition of analysis objects. Construction of the **Analysis** object of the finite element analysis via aggregation from other objects allows for the re-definition of the analysis routine and solution procedure at any given state of the finite element model. As shown in Fig. 3.2, an **AnalysisModel** class is utilized by OpenSees to hold the information about finite elements and degrees of freedom within a model separate from the classes within OpenSees which manage the system of equations (the **Analysis** class). The **AnalysisModel** class object manages the preservation of a state of a given **Analysis** object within the **Domain** object. The **Domain**

object holds the information about the finite elements, nodes, and degrees of freedom within the domain, and is aggregated of two sub-classes, *DOF_Group* and *FE_Element*. The management and communication of degrees of freedom between the *AnalysisModel* object and the system of equations is handled through objects of the *DOF_Group* subclass, where finite elements and their contributions to the systems of equations are handled by objects of the *FE_Element* class. Degrees of freedom are numbered and assembled into those of the system of equations using an associated object under the class *DOF_Numberer*. Further details of the classes of OpenSees and their uses are omitted here, but are available in McKenna et al. [2010] and in the OpenSees documentation (OpenSees [Documentation(e)]).

Standard OpenSees Model Building and Analysis Procedure

The flexibility and modular nature of OpenSees facilitates the implementation of sequential analysis of structures. From a user standpoint, OpenSees analyses are typically structured in the following manner, built through commands in scripting languages like Tcl, Python, or MATLAB:

1. Initialization of OpenSees Domain
2. Definition of Materials, Sections, Nodes
3. Definition of Elements
4. Application of Boundary Conditions
5. Definition of Convergence Tests and Time Integration
6. Application of Loading
7. Solution Loop (s)
8. Repeat No. 5, 6, and 7 as Needed, Discarding or Maintaining Displacements, Damage, or Loading Patterns as Needed

OpenSeesPy: Python Interpreter for OpenSees

OpenSeesPy (Zhu et al. [2018b]) is a Python language interpreter for the OpenSees finite element analysis framework which was developed as a refactor of the Tcl script language interpreter utilized by OpenSees. Commands through OpenSeesPy offer a Python-based way of communicating with the OpenSees library as though the commands issued were made in the original Tcl scripting language of OpenSees. OpenSeesPy offers a way for other Python modules to be used in concert with OpenSees for various use cases, while also offering an environment where calculations can be made prior to or during an analysis using Python. The present work utilized OpenSeesPy as the chosen OpenSees interpreter not only for the simplicity of utilizing OpenSeesPy for the creation and analysis of OpenSees models, but for the availability of language bindings to libraries which would allow for the coupling of the Python-based OpenSees solution algorithm with other programs.

3.2 Finite Volume Method

3.2.1 Overview

The finite volume method (first proposed in McDonald [1971]) is a numerical method for calculating the solution to partial differential equations, commonly utilized for the solution of the Navier-Stokes equations for fluid forces in the context of wind or water analyses. In the context of utilizing the finite volume method (FVM) for computational fluid dynamics (CFD), the FVM allows for the calculation of flux transfer between a series of control volumes, or finite regions of space where physical quantities are stored and accumulated. Most commonly when utilizing the FVM for CFD, the partial differential equations to be solved are the transport equations for fluid motion derived from the Navier-Stokes equations. Using the integral form of the transport equation to be solved, the divergence theorem, and a domain discretization comprised of a grid (co-located mesh) or set of grids (staggered mesh), the integral form of the transport equation can be reduced via numerical

differentiation and integration into a series of linear equations which are utilized for calculating the flux of fluid mass between control volumes (cells).

When enough of these control volumes are assembled into a mesh, complicated three-dimensional geometries can be represented and modelled, and the flux of fields associated with the fluid behavior can be calculated around these geometries. As pressures and velocities are solved at different locations within the mesh, these and other field values are interpolated between nodes within the solution mesh through the use of second-order linear interpolation.

As with other numerical modelling methods for the solution of partial differential equations, boundary conditions must be specified in FVM modelling to allow for proper solution of the equations of interest in the desired context. Boundary conditions are applied to 'patches,' or geometric regions of the model, which contain the faces of the control volume cells that are connected to those geometric regions. These patches are initialized with prescribed values for various physical quantities contained within the solution or are given function-based conditions for calculation of field variables at the domain boundary. How patches handle the solution fields and their calculation at the boundary interface is contingent on the functions utilized and prescribed to them. Improper selection of boundary conditions can result in a simulation which will not converge, or converges on an incorrect solution. As such, much care must be taken in the selection of boundary conditions for the particular use case intended by the modeller. Furthermore, as the quality of the numerical interpolation and derived/integrated quantities depends largely on the spatial discretization of the control volumes which represent the geometry of the problem, the domain must be carefully discretized in a way which allows for proper calculation of the order/type of behavior expected by the user. This is of particular importance near the boundary of bodies when the modelling of viscous effects of fluid motion and the forces generated. Further discussion and recommendations on the finite volume method in the context of computational fluid dynamics are available in Hirsch [2007].

Governing Equations

The Volume-Averaged Reynolds-Averaged Navier Stokes governing equations utilized in this work were derived by Dr. Pablo Higuera (developer of IHFOAM and olaFlow) in his Ph.D. thesis (Higuera [2018]). The governing equations for VARANS fluids consist of the continuity (conservation of mass) and momentum equations defined by Eqns. 3.2 and 3.3, respectively, where ϕ represents the porosity, D_{50} is the mean nominal diameter for porous media, and α , β , and C are optional porosity tuning parameters. $\hat{u}$, ρ , $\hat{p}^{*f}$, μ , g , X , and F_i^{ST} represent velocity, density, volume-averaged pseudo-dynamic pressure, dynamic viscosity, gravity, position vector representing height from free-surface, and surface tension force, respectively.

$$\frac{\partial \hat{u}_i}{\partial x_i} = 0 \quad (3.2)$$

$$\begin{aligned} \frac{1+C}{\phi} \frac{\partial \rho \hat{u}_i}{\partial t} + \frac{1}{\phi} \frac{\partial}{\partial x_j} \left[\frac{1}{\phi} \rho \hat{u}_i \hat{u}_j \right] = \\ - \frac{\partial \hat{p}^{*f}}{\partial x_i} - g_j X_j \frac{\partial \rho}{\partial x_i} + \frac{1}{\phi} \frac{\partial}{\partial x_j} \left[\mu_e \frac{\partial \hat{u}_i}{\partial x_j} \right] + F_i^{ST} - \alpha \frac{(1-\phi)^3}{\phi^3} \frac{\mu}{D_{50}^2} \hat{u}_i - \beta \left(1 + \frac{7.5}{CK} \right) \frac{(1-\phi)^3}{\phi^3} \frac{\rho}{D_{50}} \sqrt{\hat{u}_j \hat{u}_j} \hat{u}_i \end{aligned} \quad (3.3)$$

By assuming all media is nonporous (i.e., $\phi=1$), Eqn. 3.3 reduces to:

$$\frac{\partial \rho \hat{u}_i}{\partial t} + \frac{\partial}{\partial x_j} \left[\rho \hat{u}_i \hat{u}_j \right] = - \frac{\partial \hat{p}^{*f}}{\partial x_i} - g_j X_j \frac{\partial \rho}{\partial x_i} + \frac{\partial}{\partial x_j} \left[\mu_e \frac{\partial \hat{u}_i}{\partial x_j} \right] + F_i^{ST} \quad (3.4)$$

3.2.2 Open-source Field Operation And Manipulation (OpenFOAM)

Modelling of fluids in the presented work was completed using CFD with the Open-source Field Operation And Manipulation (OpenFOAM) (Weller et al. [1998b]) software package. OpenFOAM is a collection of C++ libraries that can be compiled to create individual applications, which

are broadly categorized as either solvers or utilities. Within this section and throughout the text, OpenFOAM applications and library components will be shown in **bold** text. The OpenFOAM Foundation's OpenFOAM.org - Version 8 was utilized in the development and analysis work in this study. OpenFOAM utilizes an 'unstructured polyhedral cell' mesh format to allow for an arbitrary number of a faces, edges, and vertices to be utilized for the cells (small discrete volumes) which comprise the volume discretization of the domain. Quantities associated with the solution of the partial differential equations are solved and stored at the centers of these control volume cells and are interpolated to the points and faces of the cells as needed. Pressures and velocities are calculated and located at the cell centers. Forces upon the boundary of the control volume are calculated from pressures within the cells surrounding the boundary of interest. The pressure value is integrated over the cell face area of interest and projected into the direction of the cell face normal for determination of the force vector at a given boundary of a cell. A diagram showing a polyhedral cell with an arbitrary number of faces and the components of the cell mentioned previously is shown in Fig. 3.3.

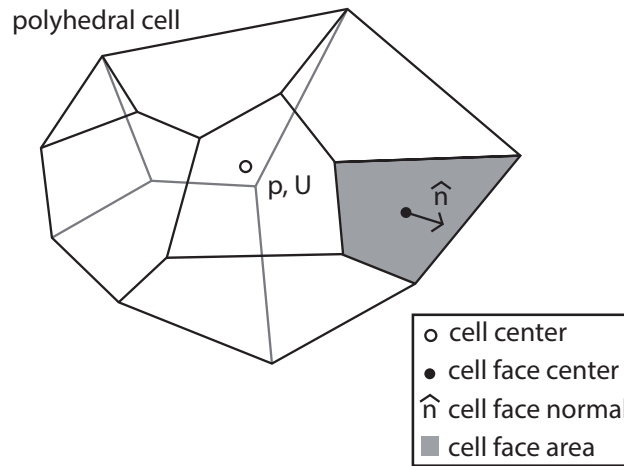


Fig. 3.3. OpenFOAM Polyhedral Cell and Referenced Components

Analytical and experimental test cases examined in this work were simulated using the **olaFlow** solvers (Higuera [2018]), an open-source project developed within the OpenFOAM framework. **OlaFlow** solves the Unsteady Volume-Averaged Reynolds-Averaged Navier-Stokes (VARANS) equations using the finite volume method, allowing for the simulation of physically correct two-phase

incompressible fluids. **OlaFlow** provides wave generation and active absorption functionality, which is useful for coastal engineering applications including tsunami-like wave generation. As the intention of developing FOAMySees was for the purpose of investigating resilience of structures to coastal hazards and the response of nonlinear structural systems to wave impingement and inundation, **OlaFlow** is a convenient selection for the fluid model, and was utilized for the test and validation studies completed later on in this thesis. In cases when pure fluid phases are modeled where porosity is neglected, the VARANS equations utilized within **olaFlow** are reduced to the classical RANS equations utilized in the standard OpenFOAM application **interFoam**, from which the solvers for **olaFlow** were derived. Since porosity effects were not considered here, similar CFD modelling procedures to those conducted by Motley et al. [2016] were performed for the following analytical correlation analyses. Modelling procedures within OpenFOAM for CFD (namely the use of OpenFOAM in the context of simulating multi-phase free-surface flows with waves generated by prescribed boundary motion, and the finite volume schemes and solution settings utilized) were largely informed and advised by Dr. Andrew Winter. The procedures used presently are based upon the modelling procedures conducted in his Ph.D thesis and associated work (Winter et al. [2018], Winter [2019], Winter et al. [2020]).

Modelling of Multi-phase Air-Water Mixture in OpenFOAM

OpenFOAM computationally manages incompressible multi-phase fluids by considering the volume fraction of components of a mixture within the individual cells of the finite volume mesh. An indicator function α is defined for the volume fraction of the two-phase fluid, which has a value of 1.0 corresponding to regions occupied by one phase, in this case water, and a value of 0.0 for the other, in this case air. Intermediate values indicate cells contain a mixture of water and air, where the free-surface between the fluids is not resolved explicitly. Using this method, the interface between the two incompressible phases (water and air) is tracked using the Volume of Fluid (VOF) technique to represent complex free-surface configurations of the flow of water and air within the domain (Hirt

and Nichols [1981]). Where free-surface tracking was necessary in this study, a volume fraction of 0.5 was used to identify an approximate free-surface. The VOF method assumes the fluid phases are immiscible (cannot be mixed), where each phase remains separate from one another. Due to the assumed immiscible condition, a single set of continuity and momentum governing equations is solved for both fluid phases.

Turbulence Modelling

When turbulence was considered, the k - ω Shear Stress Transport (k - ω SST) model was utilized for the modelling of turbulence. The k - ω SST model (Menter and Langtry [2003a]) models the transport of turbulent kinetic energy, k , turbulent kinetic energy dissipation rate within the boundary region, ω , and turbulence viscosity, ν_t . This turbulence model formulation captures the turbulent kinetic energy generation in the near field of the wall much better than k - ϵ models, resulting in higher fidelity of shear stresses and forces imparted upon objects or structures within the flow. ϵ in these equations represents rate of dissipation of turbulent kinetic energy further away from the boundary. The k - ω SST model is essentially an extension of k - ϵ models, with almost identical behavior of the k - ω SST model and k - ϵ models far away from boundaries. In Qin et al. [2018a], the authors found slightly better results for viscous shear are generated near structures with the k - ω SST model when studying structures subject to flows similar to those of tsunamis at scales of interest to structural engineers and as such the k - ω SST model is utilized here.

Mesh Motion in OpenFOAM

Motion of FSI interfaces for the present study is handled within OpenFOAM by the **displacementLaplacianFvMotionSolver** Class, which calculates the near field cell displacements required to satisfy a Laplacian diffusivity scheme specified by the user. This allows for smooth motion and deformation of the entire mesh during movement of boundary patches. Many other options for solution of cell deformation are available within OpenFOAM; however, the current coupling

methodology only utilizes **displacementLaplacian**-based mesh motion. As such, a discussion of alternative mesh motion techniques within OpenFOAM are omitted here for brevity. The **displacementLaplacian** finite volume motion solver only requires the displacement field at the boundary patch be specified either in memory or within the case files. This is handled dynamically in memory by the proposed coupling method, allowing real-time updates of the patch deformation during coupled fluid-structural analysis. Explanation of the displacement Laplacian methodology for finite volume method mesh motion is available in the original paper by Jasak and Tukovic [2006]. For more detailed information on dynamic mesh capabilities within OpenFOAM, see Jasak [2009].

Boundary Conditions

Boundary conditions for various surfaces within OpenFOAM are largely contingent on the location of the surface and its use case in a given model. Boundary conditions for walls (impermeable regions through which fluxes cannot pass) vary greatly from those of inlets or outlets, where fluid fluxes pass in and out of the finite volume domain. In addition to this, boundary conditions for moving patches are often different than those for static patches, namely for the velocity and mesh motion fields of the OpenFOAM solution. Furthermore, surfaces near moving patches must sufficiently accommodate the motion of the moving surface such that cells within the mesh do not experience deformations large enough to prohibit accurate calculation of the derivatives in the near field of the deformed mesh. With too much deformation, cells can experience cell-edge shortening to the point where cells are not an adequate quality for computation of numerical fields. In extreme cases, cells can be inverted, resulting in a zero or negative-valued cell volume. In the cases with which this code has been validated, boundary patches for each specific case will be tabulated and presented to demonstrate the use of various CFD boundary conditions for the generation or calculation of various solution field variables.

3.3 Partitioned Coupling

3.3.1 Overview

Partitioned Coupling for Fluid Structure Interaction

In this subsection, two approaches for coupling of fluid solvers and structural solvers based on a Dirichlet-Neumann decomposition of the coupling program are discussed. Dirichlet-Neumann decomposition means that the coupled solution between the fluid solver and the structural solver is controlled by the application of Dirichlet boundary conditions to the solution of the fluid and the application of Neumann boundary conditions for the structural solver (Widlund [1988]). In other words, velocities or displacements from structural motion are applied to a boundary within the fluid domain at the beginning of the coupling timestep to be integrated over time for end-of-timestep calculation of fluid shear stresses, pressures, and forces that are derived from the action of the fluid against the coupled boundary. These calculated forces or stresses are applied to the coupled surface of the structure at the beginning of the timestep to be integrated over time into the next structural state's displacements and velocities.

Explicit-Partitioned (Weak) FSI Coupling

Explicit coupling is defined here as the connection of two methodologies by means of the transfer of data once each coupling timestep between participants of the coupling. Explicit coupling does not use any iterative techniques for checking of overall data magnitudes or the satisfaction of a specified coupling convergence tolerance(s) by means of residuals. In the context of FSI, this is most commonly referred to as 'loose' or 'weak' coupling of the fluid-structure interface. Explicit coupling is the simplest way of achieving a coupled solution wherein the output of one program is utilized as an input of another and vice versa.

In the explicit or weakly coupled approach, data exchange can be easily be completed in a variety

of ways across the coupling interface with respect to the time progression of the coupled solution. The simplest of the coupling time progression methods is a "conventional-serial-staggered" (CSS) approach. In CSS time progression, one program transfers data, waits for the other program to return data, then uses returned data to calculate the next state of the program - this process is repeated until the solution is completed. An example of CSS time progression utilized in a FSI solution is provided in Fig. 3.4. Fluid solver forces calculated at the FSI boundary at time $t(n)$ are transferred to the structural solver. These forces are applied to a structural analysis model and the structural model is time-integrated from $t(n)$ to $t(n + 1)$ to calculate displacements and velocities. The motion of the system as calculated by the structural solver at $t(n + 1)$ is then transferred to the fluid solver to be applied to the fluid solver's boundaries during time $t(n)$ to $t(n + 1)$. The forces from the fluid solver at $t(n + 1)$ are then calculated, and the process is repeated with subsequent coupling timestep increments until the simulation is complete.

Similarly, data could be exchanged in a 'conventional-parallel-staggered' (CPS) approach. An example of CPS time progression within the context of FSI is provided in Fig. 3.5. In the CPS approach, each program participating in the coupling transfers data to one another at the same time and calculates its next state in parallel to the other coupling participants. Parallel-staggered computation of the next state can be more time-efficient than serial-staggered computation, as processes do not need to wait for the other processes to return data before progression to the next state. In the context of solutions where added-mass-driven instabilities are not present, these explicit partitioned methods for coupling of fluid and structural solutions require relatively small coupling time steps in order to be stable in time and to resolve high-frequency fluctuations of data transferred between participants. Instability of explicit coupling solutions can be minimized by the use of a 'staggered' time grid, where fluid and structural solvers operate on time discretizations spaced at half of a coupling timestep from each other. For example, the deformation of the structure is solved at $t(n + 1/2)$, $t(n + 3/2)$, and so on, and the fluid solution is solved at $t(n)$, $t(n + 1)$, utilizing displacements from a structural analysis solution which is running slightly ahead of the fluid solution.

832 This is referred to as an "improved-serial-staggered" coupling method, as shown in Fig. 3.6, ISS
833 has been shown to be more stable for the case of partitioned FSI solutions between 'black-box'
834 solvers than CPS or CSS. For more information about these explicit coupling methodologies
835 and time stability of partitioned conventional-serial-staggered and conventional-parallel-staggered,
836 improved-serial-staggered coupling, as well as recommendations for improving stability of explicit
837 partitioned coupling, see Farhat and Lesoinne [2000].

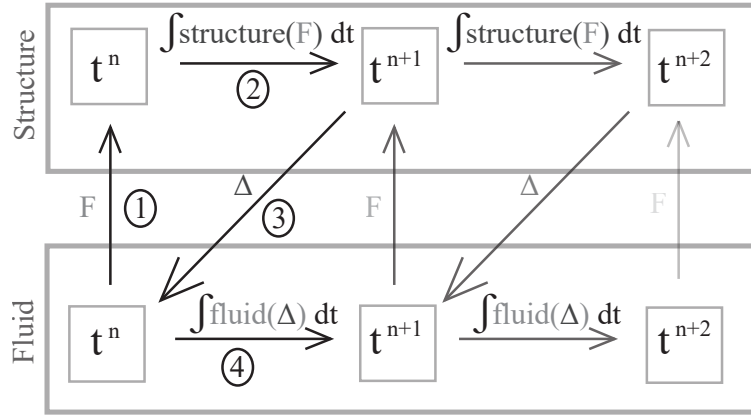


Fig. 3.4. Conventional-Serial-Staggered Explicit Coupling - Top of Figure (Structural Operator), Bottom of Figure (Fluid Operator)

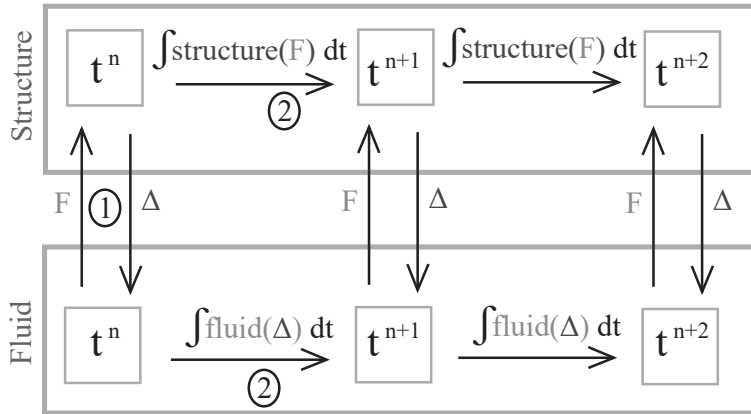


Fig. 3.5. Conventional-Parallel-Staggered Explicit Coupling - Top of Figure (Structural Operator), Bottom of Figure (Fluid Operator)

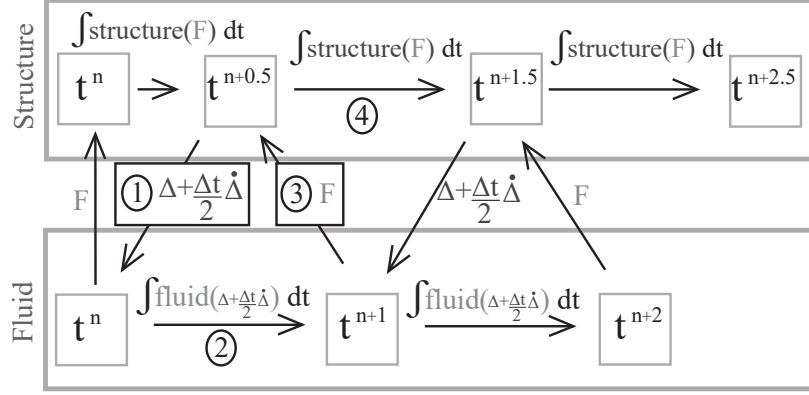


Fig. 3.6. Improved-Serial-Staggered Explicit Coupling - Top of Figure (Structural Operator), Bottom of Figure (Fluid Operator)

Implicit-Partitioned (Strong) FSI Coupling

In FSI applications, such as the simulation of flow of incompressible fluids through and about highly flexible structures, implicit coupling via fixed-point iteration is sometimes crucial in establishing the convergence of the solution and avoiding instabilities. This is especially true when the added-mass effect of the fluid surrounding the FSI interface plays a large role in the dynamics of the motion of the structure. Implicit coupling of partitioned codes has traditionally been handled by a variety of methods, including use of Gauss-Seidel iterations, Gauss-Seidel iterations with dynamic relaxation factors such as Aitken Under-Relaxation (Küttler and Wall [2008]), and indirect Quasi-Newton methods such as Interface Quasi-Newton - Inverse Least Squares (IQN-ILS, Degroote et al. [2009]), and Interface Quasi-Newton - Inverse Multi-Vector Jacobian (IQN-IMVJ, Scheufole [2015]). The implicit methods considered here depend on the solution of a fixed-point iteration on the output of each solver to ensure the work applied to the coupling meshes from both solutions is equivalent in magnitude across each interface. From the preCICE documentation, "the relative coupling residual is defined as $R(x^k) = ||H(x^k) - x^k||_2 / ||x^k||_2$ for a fixed-point equation H " with input data x^k (preCICE User Documentation).

When coupling is handled in a partitioned approach, many of the variables involved in each solution are not available to the coupling algorithms, The Jacobians of both the fluid and structural

solvers are not directly accessible in the partitioned coupled approach, so these Jacobians cannot be utilized to approximate tangent behavior of the coupled system or to stabilize coupling iterations based on the amount of energy on either side of the coupling interface. As such, alternative means must be utilized for predicting a coupled system response which would result in convergence of coupling iterations.

One means of quantifying convergence of a partitioned-coupled system is through a 'coupling residual operator.' The coupling residual operator is a function of the changes in data between coupling iterations based on the input data, resolved by solving the fixed-point iteration for the coupling data residual over several iterations. When the changes in coupling data residual between iterations are sufficiently small (below a coupling tolerance), the coupling timestep has 'converged,' meaning that the data within the coupling timestep has reached a limit.

Some implicit partitioned-coupling methods achieve work satisfaction over the coupling interface between structural and fluid solvers by considering the relative convergence of coupling data (in this case, forces and displacements) from both solvers over the coupling iterations within a timestep. In partitioned-coupled FSI, satisfying a coupling data relative residual convergence ensures that the coupling data does not grow unbounded and allows for the work passed over the interface between both solvers to be roughly equivalent.

In the IQN-ILS method, acceleration of coupling iterations and thus convergence of a coupling timestep is completed using an approximation of the required changes in coupling data that would result in a coupling residual which is below the specified tolerance. As the only available information to the coupling algorithm is the input and output data, the IQN-ILS algorithm works by using a function constructed of changes in coupling residuals and changes in input data. This is completed by creating arrays of changes in residual vectors between coupling iterations, using several iterations to generate a function. The function of the changes in input data that is used to calculate a change in coupling data which satisfies the coupling residual tolerance is calculated using an inverse least-squares model. This referred to as 'approximating of the inverse of the Jacobian of the coupling

881 residual operator (Degroote et al. [2009]).

882 The least-squares modelling process utilized in IQN-ILS can loosely be described as using a
 883 QR decomposition of the stored residual changes to estimate the secants of the inverse Jacobian of
 884 the residual operator at the present timestep. Filtering of the arrays of coupling residual changes is
 885 completed by removing residual vectors from iterations which are near linearly dependent on each
 886 other (identical or otherwise). Iteration and filtering of data from iterations is completed within a
 887 coupling timestep until combinations of changes in the coupling residual that satisfy convergence of
 888 the coupling residual tolerance can be found. These combinations are then used to approximate the
 889 required corrections to the coupling data during the next coupling iteration which would result in
 890 acceleration of the convergence of the root finding iteration of the coupling residual operator. This
 891 works by using the secants of the residual operator for a given number of iterations and timesteps to
 892 approximate the current tangent of the residual operator, which can then be utilized with coupling
 893 data from previous iterations to predict what coupling data should be passed at the next iteration.

894 Diagrams showing the algorithm and time progression of implicit coupling with Gauss-Seidel
 895 iterations and with the IQN-ILS method are shown in Figs. 3.7 and 3.8, respectively.

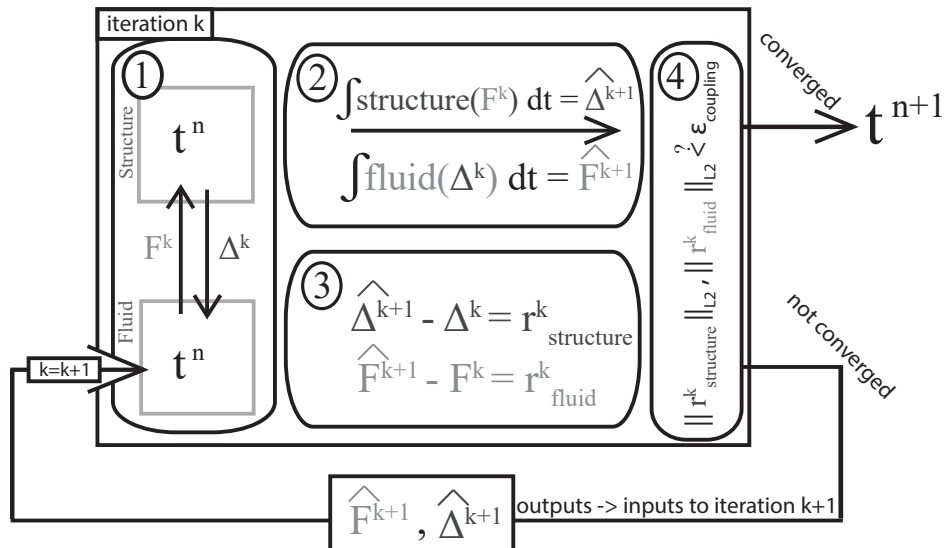


Fig. 3.7. Implicit Coupling with Gauss-Seidel Iterations

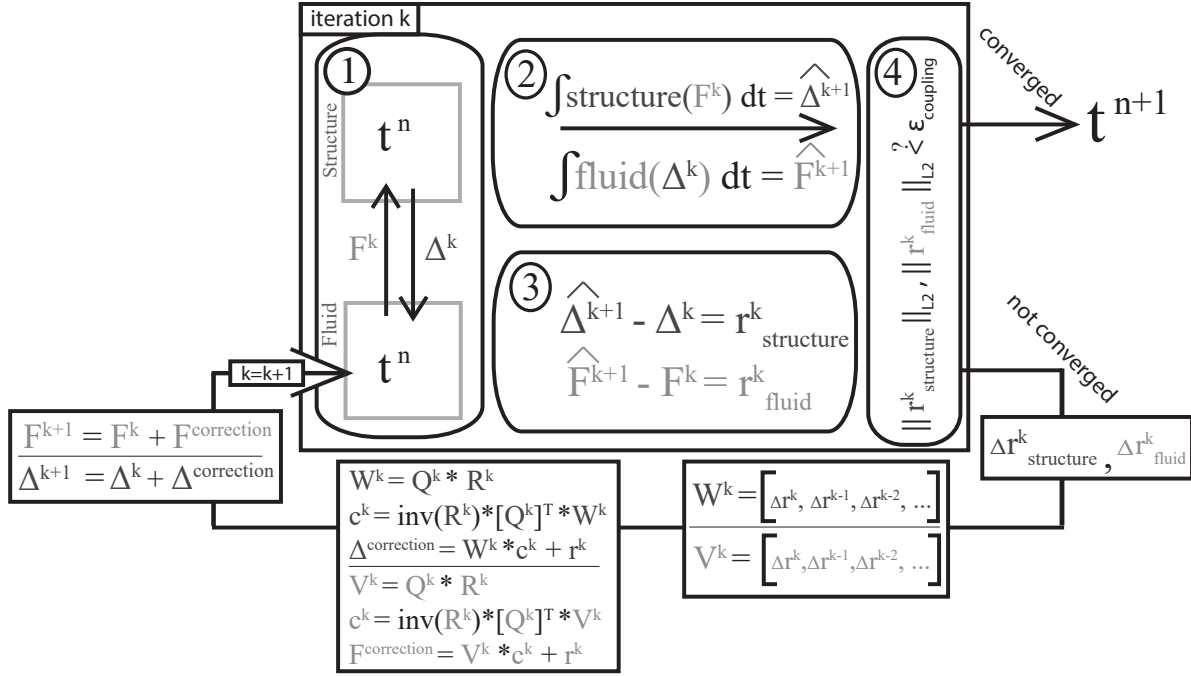


Fig. 3.8. Implicit Coupling with IQN-ILS Iterations

Stability of Partitioned Coupling in the Context of Incompressible FSI

While explicit coupling methodologies work well in many cases, the stability of the weakly-coupled partitioned solution cannot always be ensured by the reduction of the coupling timestep due to the global effect of structural motion throughout the body of the incompressible fluid on fluid pressure as well as motion and the presence of the added-mass effect and its interaction with the lower modes of vibration of the structure (Causin et al. [2005]). When the added mass introduced by the motion of the structure within a fluid is greater than the mass of the structure plus a factor related to the acting stiffness of the structure, staggered coupling between the structure and the fluid is inherently unstable regardless of the timestep used. This is particularly problematic when looking at highly flexible structures subject to incompressible flows.

In situations where an incompressible fluid's inertia plays a large factor in the dynamics of the structure, implicit coupling techniques are required to ensure convergence of the solution. Use of Gauss-Seidel iterations in the case of a partitioned-coupled FSI simulation does not ensure

convergence of the solution when the added-mass of the action of the fluid participates largely in the dynamics of the solution. Reduction of the timestep may even make the instability worse, as more modes of deformation of the fluid and structure may be resolved and therefore more added mass may be added to the solution.

It has been shown that both the Gauss-Seidel iteration method with dynamic relaxation (Aitken Under-Relaxation) and indirect Quasi-Newton methods such as IQN-ILS can be utilized successfully in the stable simulation of strongly-coupled FSI between multi-phase flows with free surfaces and highly flexible structures (Degroote et al. [2010c]). Further discussion of efficiency of implicit coupling algorithms, the stability of partitioned strongly-coupled FSI problems with respect to the engaged added mass of fluid, and the effects of the interplay of the inertia of the actively engaged added mass of fluid with the dynamics of the structural inertia is left to other sources (Küttler and Wall [2008], Degroote et al. [2009], Scheufele [2015], Degroote et al. [2010b], Degroote et al. [2010c], Causin et al. [2005], Delaissé et al. [2023]). For the simulation of incompressible FSI with the presented methodology, OpenFOAM coupled to OpenSees in a partitioned approach for the simulation of wave impingement against flexible structures, it is recommended that implicit partitioned coupling routines are utilized, particularly if simulating without thorough knowledge of the expected stability of the simulation.

K-Means Clustering, K-Nearest Neighbor Search

K-means clustering and K-Nearest Neighbor search are two methods which are utilized within the present coupling methodology for the association of nodes between meshes. The algorithm for K-means clustering was developed in the mid-twentieth century for the purpose of classification of voltage signals in the context of 'pulse-code modulation' (Lloyd [1982]). It has since been expanded, refined, and utilized in applications that require clustering of data points regularly, such as machine learning, principle component analysis, and classification. K-means methods allow for the clustering of a set of points to another through the use of normalized measures for distance, with the aim of

934 minimizing the variance between the locations of a set of clustering reference points P and a set
 935 of query points to be clustered x . The classification of the query points of x into the clustering
 936 reference points of P is achieved through solution of Eqn. 3.5, from Lloyd [1982].

$$\operatorname{argmin} \sum_{i=1}^k \sum_{x \in P_i} \|x - \frac{1}{|P_i|} \sum_{x \in P_i} x\|^2 = \operatorname{argmin} \sum_{i=1}^k |P_i| \operatorname{variance}(P_i) \quad (3.5)$$

937 where $\frac{1}{|P_i|} \sum_{x \in P_i} x$ is the centroid of the set of reference points to which the query points are
 938 clustered. This methodology minimizes the $L2$ norm between the query points and the reference
 939 points, determining the subsets of $\{x\}$ that reduce the variance of the means of the subsets of $\{x\}$
 940 from the mean of each reference point P_i . The end result is a set of sets that can loosely be described
 941 as $\{P\} \ni \{y_i\}$ for all subsets $\{y_i\}$, where $\{y_i\} \subset \{x\} \forall i$ and $\{y_i\} \cap \{y_j\} = \emptyset$ for $j \neq i$. This allows for
 942 clustering of a series of points to a set of another, creating a superset $\{x\}$ of mutually exclusive
 943 subsets $\{y_i\}$ of query points that are each associated with a single reference point. K-Nearest
 944 Neighbor Search (and mapping) also utilizes this principle, determining for each query point (points
 945 within $\{x\}$) which point in another set $\{z\}$ is the nearest via Euclidean distance.

946 3.3.2 Open-Source Partitioned Multi-physics Coupling Library (preCICE)

947 preCICE (Chourdakis et al. [2022]) is a library developed for the purpose of partitioned coupling
 948 of multiphysics simulations, utilizing iteration to maintain numerical stability and to ensure work
 949 convergence between coupled surfaces. The preCICE library functions by transferring data arrays
 950 between coupling partners for serial or parallel computation of fully-coupled solutions between
 951 codes. Convergence measures, data transfer, timestep iteration, and surface-to-surface mapping
 952 between coupling partners are managed by the preCICE library. These surface-to-surface mappings
 953 are most generally handled with radial basis function (RBF) mesh interpolation from a source mesh
 954 to a destination mesh.

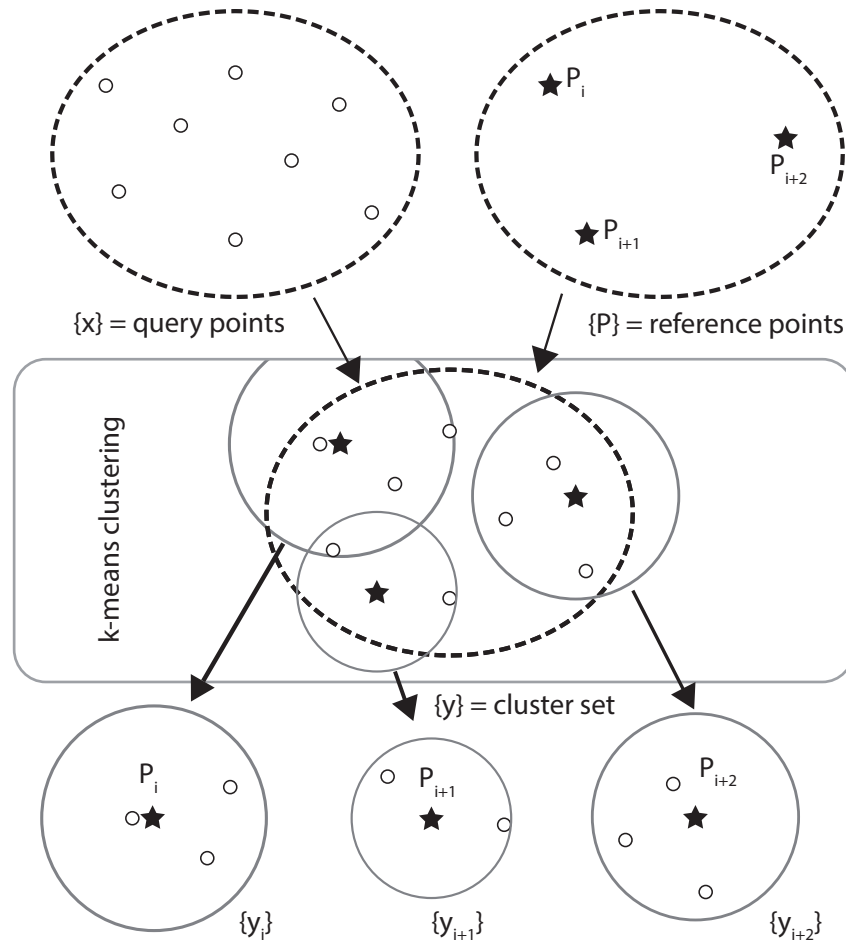


Fig. 3.9. Diagram Showing General Clustering of Points to Reference Points Utilizing K-means Clustering Algorithm

Data Communication Between Coupled Programs

While MPI methods are available for coupling while utilizing the preCICE library (for more information, see Totounferoush et al. [2021]), the present work utilizes preCICE's capability to allow programs to communicate via TCP/IP sockets to pass data between coupling participants. This selection was made to ease the difficulty of the compilation process of the various libraries required to assemble the coupled solution. MPI communication between participants requires a shared version of MPI amongst the programs utilized for coupling. A short summary of TCP/IP communication is provided below for context.

TCP/IP (sockets)

A TCP/IP 'socket' is a file transfer protocol, which allows remote processes or programs to communicate by exchanging data via an IP address (IBM), a protocol that a set of rules which are followed, and the utilization of particular 'ports' through which data can be passed to and from a program or IP address. On Unix-like systems, the TCP/IP connection can be completed with the establishment of a file through which all connected programs can communicate. In the context of the preCICE library, this location is established in the 'precice-configuration.xml' file, which is a JSON file utilized for the configuration of the exchange of coupling data between participants. When all coupling participants connected to the coupling library have loaded the configuration file and read the location of the TCP/IP folder and file (or 'socket address') through which information will be passed, and this TCP/IP connection is established and maintained throughout the rest of the analysis.

A detailed description of the TCP/IP socket methodology utilized by preCICE is available in Chapter 3, Section 3.2 of the Master's thesis of Alexander K. Shukaev (Shukaev [2015]). TCP/IP connections offer a method that is robust albeit slower than MPI for the communication of data between programs while precluding the likely probability that the programs coupled together do not utilize the same version of MPI and will experience conflicts. Using TCP/IP sockets allows for a more flexible approach to the software requirements of each program, and eases adoption and future development of the coupled solution with newer versions of coupled participant software. For example, if OpenFOAM releases a new version of their libraries, a user could theoretically ignore which version of MPI is utilized during compilation and upgrade OpenFOAM and the OpenFOAMpreCICE adapter to utilize a new version of OpenFOAM in the coupling approach. A user could do similarly for OpenSeesPy, upgrading the library as necessary without concern of breaking the functionality of the coupled solution code. While speed is theoretically compromised when utilizing a TCP/IP connection, this has not been studied presently and future work could be done to examine the utilization of MPI for the passage of data between coupling participants and the effects this may have on computation time.

Mapping Routines

Mapping between meshes with varying numbers of coupling nodes is handled generally within preCICE through the use of Radial-Basis-Function Mapping (RBF Mapping). Alternatively, Nearest Neighbor or Nearest Projection mapping can be utilized for interpolation between co-located meshes quicker than Radial-Basis-Function mapping. For descriptions of the mapping routines available within preCICE and recommendations for their utilization in the context of partitioned coupled multi-physics solutions, see Lindner [2019].

preCICE Adapters

preCICE is structured in a modular manner, with applications interfacing with the preCICE library via API function calls, which allows access to the routines of the library. Applications may be 'plugged-in' to the preCICE library coupling routines by means of a coupling 'adapter.' These adapters interface with the coupled program, taking data from the program and passing it to the preCICE library routines to be utilized in a manner specified by the user through configuration files. preCICE offers many adapters for open-source finite element and finite volume codes. For example, see Uekermann et al. [2017], which covers the preCICE provided adapters for other open-source computational continuum mechanics codes. It is important to note that many of these codes and adapters previously coupled via preCICE have not been validated for civil/structural engineering applications, and are not optimized to allow for a structural engineer to pick up and readily utilize them for engineering design purposes. The standard OpenFOAM adapter as described by Chourdakis et al. [2023] has been utilized for the coupling of OpenFOAM to the presented FSI coupling solution procedures.

1010 **preCICE Python Language Bindings**

1011 The preCICE API function calls are available in many languages via bindings. These bindings allow
1012 for usage of the preCICE library within code written in the language shared by the bindings. This
1013 offers flexibility for users to implement their own code for dynamically or temporally assigning
1014 boundary conditions within a desired numerical simulation. The mechanics of how the language
1015 bindings for preCICE interface with the background software architecture can be found at within
1016 the preCICE source code (preCICE Python Bindings). Calls to the preCICE API are managed
1017 through a *SolverInterface* object. This object has various methods that control data communication
1018 and coupled solver progression. The methods of *SolverInterface* utilized in the presented work are
1019 shown in Figs. 3.10 and 3.11.

1020 A simplified version of initialization routines for a preCICE coupled python code along with their
1021 associated preCICE python language binding commands are shown in Fig. 3.10. The *SolverInterface*
1022 class, methods and sub-classes are utilized within the main solution loop of the coupling driver
1023 program to control the progression of the solution and manage iteration of the structural solver
1024 until preCICE determines the coupling residual tolerance determined by the user has been met. A
1025 simplified version of the main solution loop is shown schematically in Fig. 3.11. Calls to preCICE
1026 API functions are highlighted in shaded ovals, wherein the functions themselves are shown in a
1027 bolded text.

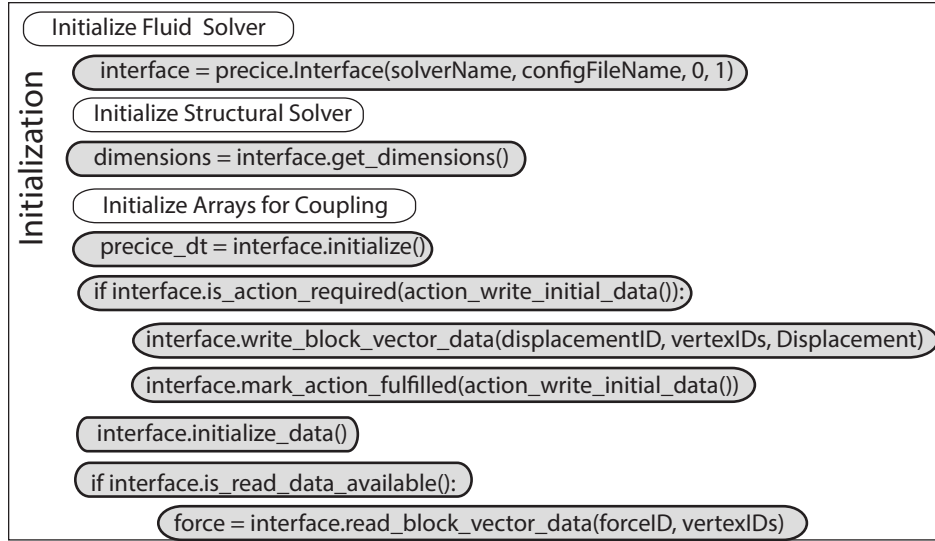


Fig. 3.10. preCICE Python Language Binding commands utilized within initialization of a coupled solution

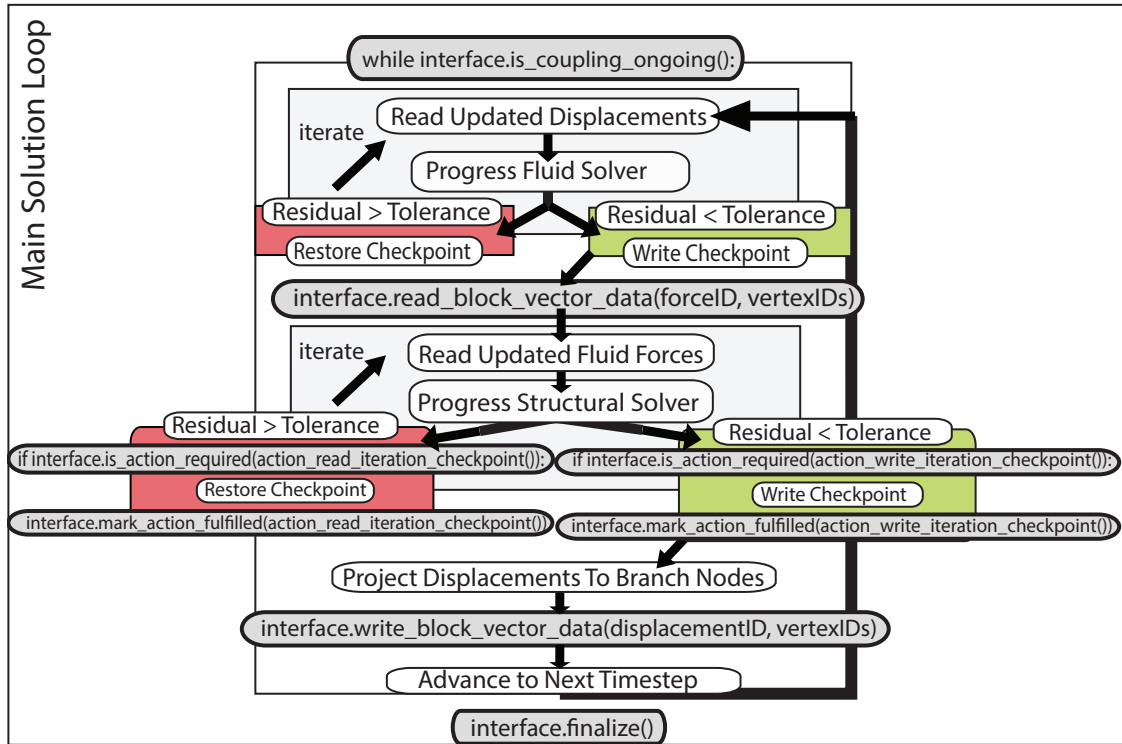


Fig. 3.11. Diagram of preCICE Subroutine Calls Within An Explicitly-Coupled FSI code where the structural solution code is controlled with preCICE Python language bindings commands. This diagram is solely intended for informing about preCICE API function calls and where these calls are necessary, and is not indicative of the complete procedures required for a fully-coupled FSI solution

Chapter 4

Coupling Between Non-conformal Interfaces for Simulation of Nonlinear Fluid-Structure-Interaction

This chapter covers the approach to and the implementation of the coupling between non-conformal FEA and CFD meshes within OpenSees and OpenFOAM through the preCICE coupling library for the purposes of simulation of nonlinear FSI. The chapter is comprised of an approach section and an implementation section. The approach section begins with the motivation for and an overview of the coupling approach, and concludes with detailed descriptions of how geometric mapping of force and displacement data between CFD and FEA solutions was completed within the coupling approach. The implementation section begins with an overview of the entire coupled simulation methodology, followed with descriptions of how each component of the coupled solution was implemented as part of the coupling routines. This is followed by subsections covering coupled model initialization, configuration of the preCICE coupling library, and required steps for preparation of the coupling participants OpenFOAM and OpenSees in the context of the current coupling approach. The chapter concludes with a description of how the preCICE coupling library was utilized along with OpenSees

and their Python language bindings for the development of a coupling driver code within Python to implement finite element models constructed within the OpenSees framework as part of a flexible approach to open-source simulation of nonlinear FSI using partitioned coupling.

4.0.1 Terminology Utilized in this Chapter

Terminology used for various components of the coupling approach are provided here for clarity.

Coupling Code Nomenclature

The coupling routines developed by the author are referred to by the author as "FOAMySees" in an attempt preclude any confusion between what software and procedures have and have not been developed as part of this research. FOAMySees is a collection of files and Python objects which control the interior solution loop and convergence of the structural analysis model contained within an exterior coupling solution loop for partitioned-staggered FSI within the OpenSees framework through the preCICE multiphysics coupling library. The code is available online through the FOAMySees Github repository (Lewis [2023]).

Coupling Mesh Terminology

The additional coupling mesh implemented in the current methodology is the 'coupling data projection mesh.' The nodes of the coupling data projection mesh are referred to as 'data communication branch nodes,' 'branch nodes,' or 'branches.' The coupling data projection mesh and its branch nodes exist solely outside of and are inaccessible to the CFD solution and FEA solutions. 'Branch node groups' consist of multiple branch nodes that are associated to one FEA node. The complete assembly of these branch node groups for all coupled FEA nodes comprises the coupling data projection mesh.

Formatting Associated With Particular Libraries

For the sake of clarity of the source of each component utilized in the coupling, classes, objects, and methods from *OpenSees*, **OpenFOAM**, and *preCICE*, will be shown in ***bold and italicized***, **bold**, and *italicized* text, respectively.

4.1 Coupling Approach

In order to appropriately model turbulent interactions of fluid flows with flexible structures, the response and dynamics of the flows and the structures of interest must individually be modelled appropriately. For many flows interacting with civil engineering structures, determining accurate forces imparted by the flow on structures of interest within the flow involves modelling upstream and downstream behavior of the flow and the incorporation and solution of the fluid's turbulence at, near and beyond any structures in the flow. Since turbulence is a three-dimensional phenomenon, turbulent flows must be modelled in three dimensions. When modelling turbulent flows with a body-fitted mesh CFD model, structures modelled as objects in the flow need to be represented as three-dimensional closed surfaces. CFD models also require appropriate amounts of mesh resolution around the immediate boundary of and in the wake of objects of interest to capture the velocity gradients near the structure for proper calculation of the turbulent flow characteristics and viscous forces. Additionally, the CFD mesh must possess an appropriate discretization resolution along structural boundaries and geometries within the fluid domain such that calculation of flow properties is not affected by the mesh itself. Oftentimes the mesh resolution requirements of the near field flow result in a high density point cloud of mesh nodes in the near region of CFD boundaries of interest.

Structural finite element analysis models of civil engineering superstructures are commonly geometrically or spatially simplified for the sake of reducing computational cost and time while maintaining enough pertinent information about the response behavior of components of the structure within the model through approximation to justify doing so. It is more efficient for large

structures comprised of various structural components to be modelled in a way that reduces the geometric complexity for finite element analysis. Reduction of modelling complexity allows for faster iteration of structural system design configurations and structural element design, and is standard practice for determining gross behavior of building superstructures within civil-structural engineering applications. Despite the benefits of reduced complexity in structural modelling, civil engineering superstructures are often modelled in three dimensions without through-thickness for each component as a result of this geometric simplification. Superstructure finite element models omit architectural features such as walls or parapets, but as these non-structural components comprise the majority of the building envelope they are largely participatory in interacting with flows and collect the forces from wind or waves. Furthermore, the external bounding surface comprised of the nodes of a finite element model of a building or bridge superstructure is not necessarily congruent with the structure's external shape or the building envelope, making use of typical superstructure finite element models in the context of a three-dimensional coupling to fluid dynamics solvers with turbulence a challenge.

To account for the discrepancy in geometric space and discretized representations between the two analysis components to be coupled (OpenFOAM and OpenSees), surface-to-surface, surface-to-edge, or surface-to-node interpolation must be completed. The interpolation between each point of the meshes of the coupled FEA and CFD models is managed by introducing an additional coupling mesh to the solution. This coupling mesh is utilized to calculate the forces and moments that should be applied to a finite element model that is nested within the dense 3D point cloud consisting of the nodes of the coupled surface of the CFD model. The coupling mesh is also used for the spatial projection of displacements and rotations from the finite element model to the three dimensional surface of nodal locations in a body-fitted CFD mesh. This methodology allows for but is not limited to the coupling of single degree of freedom and coarse multiple degree of freedom finite element models to CFD surfaces for evaluation of the nonlinear response of structures to loading calculated by three-dimensional CFD models that incorporate turbulence modelling.

Forces and displacements are both mapped between coupled meshes by preCICE at the end of the converged coupling timestep for application as boundary conditions during next timestep, but are managed separately in the solution. The different mapping routines and calculations utilized for forces and displacements are conducted by the developed coupling driver and are described in the following subsections.

4.1.1 Force Mapping Between Solutions

At the end of a coupling timestep, preCICE calculates forces to be sent to FOAMySees by integrating surface pressures over each polyhedral cell's face along the coupled boundary of the OpenFOAM model and multiplying the magnitude of the calculated force by the normal vector of the cell face of interest. This is illustrated in Figs. 4.1 and 4.2.

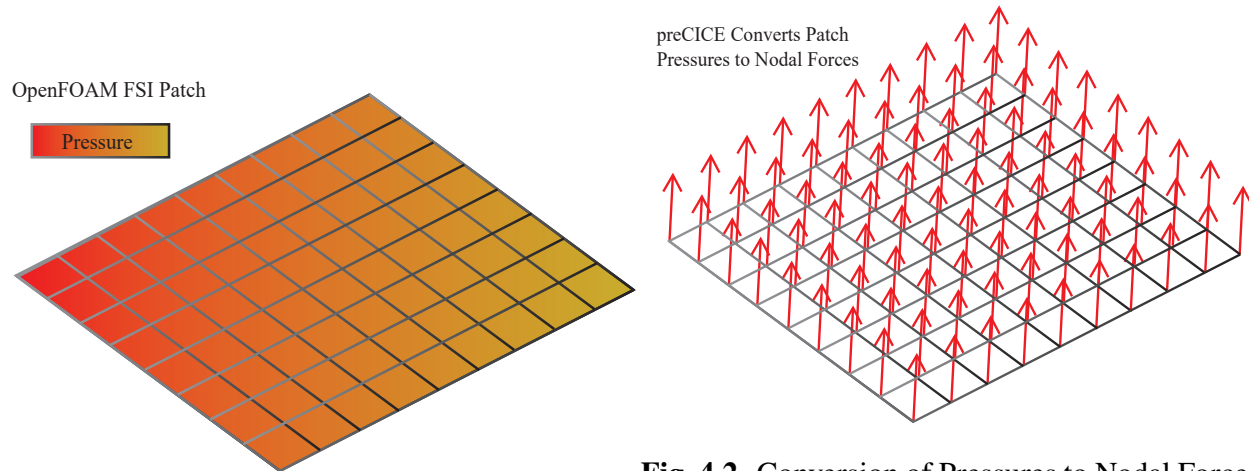


Fig. 4.1. OpenFOAM FSI Patch

Fig. 4.2. Conversion of Pressures to Nodal Forces: preCICE converts surface pressures along the FSI patch into nodal forces which may then be mapped to branch nodes

The coupling data projection mesh is coupled by preCICE to the three-dimensional mesh surfaces, which represent the structure within the CFD model. Forces from the CFD model are positioned at the cell face center nodes and are conservatively applied to the coupling data projection mesh using radial basis functions (RBFs) or nearest neighbor mapping. The mapping between the

Radial Basis Function (RBF) Mapping
from OpenFOAM Patch Nodal Positions
to Branch Nodes of FOAMySees

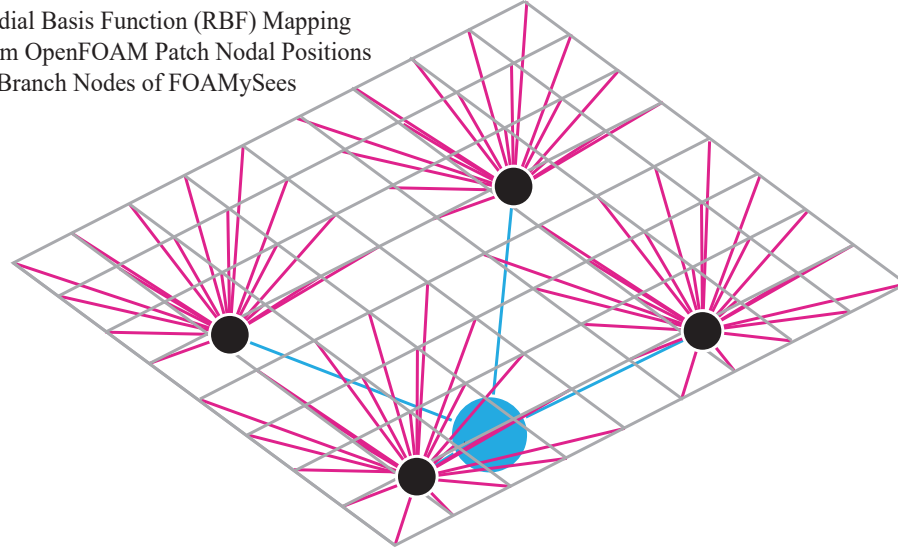


Fig. 4.3. Nearest Neighbor or Radial Basis Function (RBF) mapping is calculated between branch node positions and CFD patch surface nodes: Isometric View of Model-to-Model Mapping from OpenFOAM surfaces to FOAMySees branch nodes and subsequent 1D interpolation to OpenSees FEA nodes

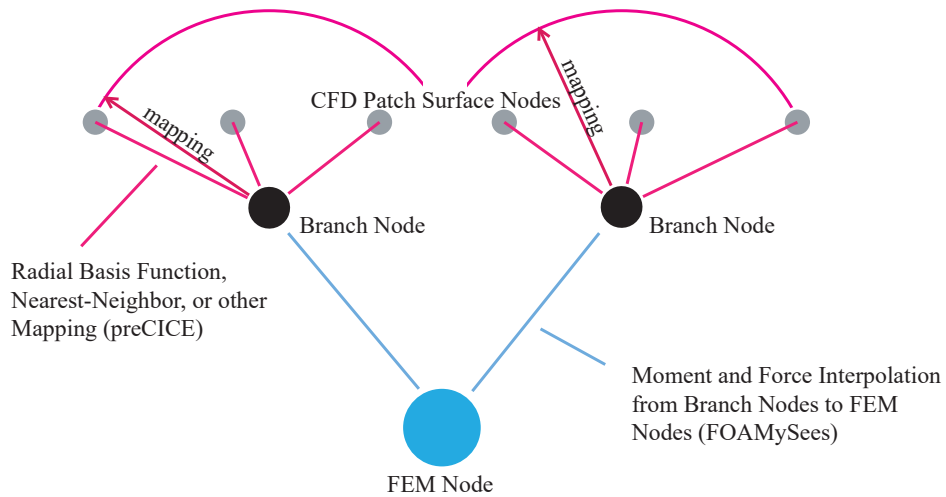


Fig. 4.4. Elevation View of RBF Mapping from OpenFOAM surfaces to branch nodes to FEA nodes

1129 nodes of the coupled CFD surface and the coupling data projection mesh is illustrated in Figs. 4.3
 1130 and 4.4. After forces are mapped by preCICE to the branch coupling mesh nodes associated with
 1131 each coupled FEA node, branch node forces and moments are calculated and applied to the FEA
 1132 nodes for time integration of the structural analysis model during the next timestep or coupling

iteration. For all FEA nodes, the force vectors of the branch nodes associated with a given FEA node are summed to construct the force vector $\hat{F}_i(t^n)$ at each timestep. More specifically, for a standard FOAMySees model FEA node with $\mathbf{j}=\mathbf{N}$ branches, the force relationship between an FEA node and its branch nodes at any particular time is described in Eqn. 4.1. Each FEA node's applied forces are a combination of the forces applied to all of its branches. Similarly, each FEA node's moments ($\hat{M}_i(t^n) = [M_{xx_i}(t^n), M_{yy_i}(t^n), M_{zz_i}(t^n)]$, Eqns. 4.2 through 4.4) are calculated with the sum of the contribution of each of the branch nodal forces to the main axis moments about each FEA node, with moment arms calculated from the updated position of the branch with respect to the current location of the FEA node. The branch locations utilized for the calculation of applied moments are updated each timestep.

$$\hat{F}_i(t^n) = \sum_{j=1}^N \hat{F}_{branch_j}(t^n) \quad (4.1)$$

$$M_{xx_i}(t^n) = \sum_{j=1}^N F_{z_j}(t^n) * dy'_j(t^n) - F_{y_j}(t^n) * dz'_j(t^n) \quad (4.2)$$

$$M_{yy_i}(t^n) = \sum_{j=1}^N F_{x_j}(t^n) * dz'_j(t^n) - F_{z_j}(t^n) * dx'_j(t^n) \quad (4.3)$$

$$M_{zz_i}(t^n) = \sum_{j=1}^N F_{y_j}(t^n) * dx'_j(t^n) - F_{x_j}(t^n) * dy'_j(t^n) \quad (4.4)$$

These operations for moment calculation can be written as a linear combination of dot products between vectors (Eqns. 4.5 through 4.7), where each vector $(\hat{\cdot})_i$ is comprised of the force and moment arm components of the entire branch group for each coupled FEA node i .

$$M_{xx_i}(t^n) = \langle \hat{F}_{z_i}(t^n)_{[1 \times N]} | \hat{d}'_{y_i}(t^n)_{[N \times 1]} \rangle - \langle \hat{F}_{y_i}(t^n)_{[1 \times N]} | \hat{d}'_{z_i}(t^n)_{[N \times 1]} \rangle \quad (4.5)$$

$$M_{yy_i}(t^n) = \langle \hat{F}_{x_i}(t^n)_{[1 \times N]} | \hat{d}'_{z_i}(t^n)_{[N \times 1]} \rangle - \langle \hat{F}_{z_i}(t^n)_{[1 \times N]} | \hat{d}'_{x_i}(t^n)_{[N \times 1]} \rangle \quad (4.6)$$

$$M_{zz_i}(t^n) = \langle \hat{F}_{y_i}(t^n)_{[1 \times N]} | \hat{d}x'_i(t^n)_{[N \times 1]} \rangle - \langle \hat{F}_{x_i}(t^n)_{[1 \times N]} | \hat{d}y'_i(t^n)_{[N \times 1]} \rangle \quad (4.7)$$

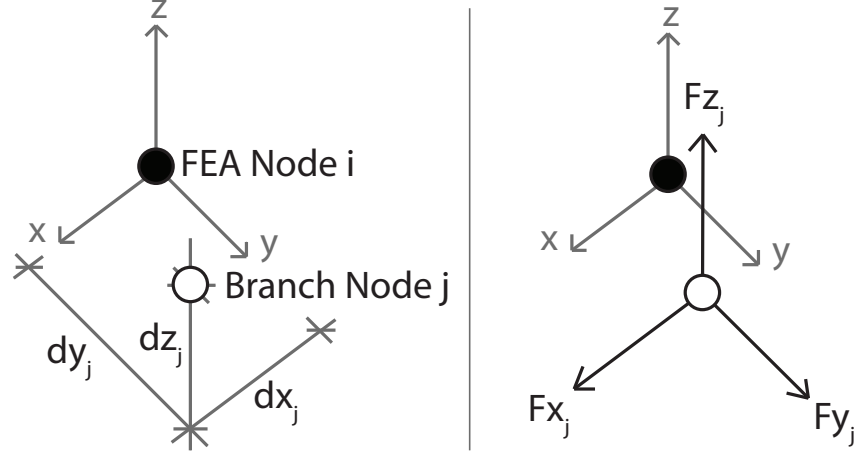


Fig. 4.5. Components of Force and Moment Utilized for Calculation of FEA Applied Forces and Moments From the Nodes of the Branch Coupling Mesh

4.1.2 Displacement Mapping Between Solutions

The displacements and rotations of the coupled FEA nodes are calculated by OpenSees and are retrieved from the finite element solution at the end of each converged structural analysis timestep. Displacements are mapped to the branch nodes associated with each FEA node via translation such that each of the N branch nodes per FEA node moves rigidly with its associated FEA node. The branch nodes then are rotated rigidly about the FEA nodes via rotation matrix transformation, according to each FEA node's calculated local slope. The rotation matrix $\underline{\mathbf{R}}$ is shown in Eqn. 4.8, where ϕ , θ , and ψ are FEA nodal rotations in degrees (improper Euler angles) about the x, y, and z axes respectively.

$$\underline{\underline{R}} = \begin{bmatrix} \cos(\theta)\cos(\psi) & \sin(\phi)\sin(\theta)\cos(\psi) - \cos(\phi)\sin(\psi) & \cos(\phi)\sin(\theta)\cos(\psi) + \sin(\phi)\sin(\psi) \\ \cos(\theta)\sin(\psi) & \sin(\phi)\sin(\theta)\sin(\psi) + \cos(\phi)\cos(\psi) & \cos(\phi)\sin(\theta)\sin(\psi) - \sin(\phi)\cos(\psi) \\ -\sin(\theta) & \sin(\phi)\cos(\theta) & \cos(\phi)\cos(\theta) \end{bmatrix} \quad (4.8)$$

$\underline{\underline{v}}$ is an array of column vectors comprised of the distances of each of the FEM node's branches from its root FEM node.

$$\underline{\underline{v}} = \begin{bmatrix} dx_1 & \dots & dx_N \\ dy_1 & \dots & dy_N \\ dz_1 & \dots & dz_N \end{bmatrix} \quad (4.9)$$

The rotation of all branches within a group simultaneously is completed with a left multiplication of the branch distance array $\underline{\underline{v}}$ (Eqn. 4.9) with the rotation matrix $\underline{\underline{R}}$. The transformation for the branch group is completed as shown in Eqn. 4.10 to produce an array of rotated branches, $\underline{\underline{w}}$.

$$\underline{\underline{w}} = \langle \underline{\underline{R}} | \underline{\underline{v}} \rangle = \begin{bmatrix} dx'_1 & \dots & dx'_N \\ dy'_1 & \dots & dy'_N \\ dz'_1 & \dots & dz'_N \end{bmatrix} \quad (4.10)$$

1155 The finite element model's displacements are interpolated to the coupling data projection mesh
 1156 by the FEM node/branch node relationship described in Eqn. 4.12 for the N branches of each branch
 1157 node group. This rigid body rotation is added to a translation array sized $[3 \times N]$ tiled with the
 1158 displacement vector ($\hat{x}_i$) of the root FEA model node (i) to which the branches of $\underline{\underline{w}}$ are associated.

$$[\hat{\delta}_i, \hat{\delta}_i, \dots, \hat{\delta}_i]_{[3 \times N]} = \hat{x}_i_{[3 \times 1]} \otimes [1, 1, \dots, 1]_{[N \times 1]} \quad (4.11)$$

$$\underline{\underline{\Delta}}_{branches(i)} = [\hat{\delta}_i, \hat{\delta}_i, \dots, \hat{\delta}_i]_{[3 \times N]} + \langle \underline{\underline{R}}(\phi_i, \theta_i, \psi_i) | \underline{\underline{v}}_i \rangle_{[3 \times N]} \quad (4.12)$$

1159 The displacements of each branch are assembled into their respective places within the coupling

data projection mesh displacement array. Branch node displacements are exchanged to OpenFOAM via preCICE for application as Dirichlet boundary conditions within the CFD model during the next coupling timestep. The process of rigid-body translation and rotation of branch groups about their respective FEM nodes is illustrated in Fig. 4.6.

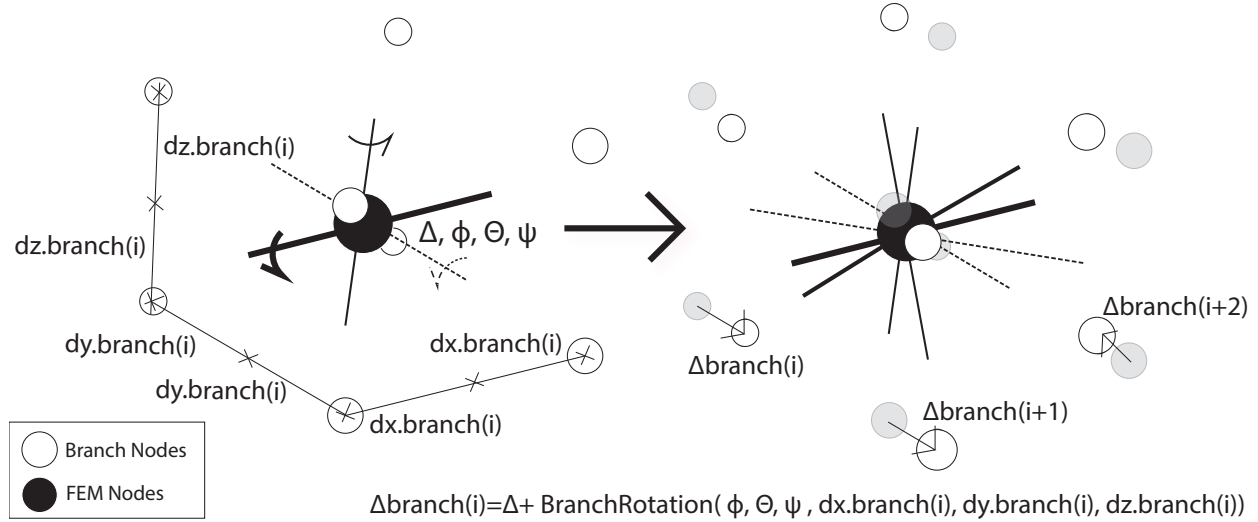


Fig. 4.6. Illustration of Rotation of Branch Nodes Completed by FOAMySees

4.2 Implementation of Coupling Approach

4.2.1 Overview

To summarize the coupling approach, OpenSees was utilized as the structural analysis component within an open-source methodology for simulation of nonlinear FSI through the development of the FOAMySeesCouplingDriver code. The code was developed to control the structural analysis solution within OpenSees during a coupled FSI simulation through the use of the OpenSees Framework Python bindings. Python bindings to the preCICE coupling library are also utilized within the FOAMySeesCouplingDriver for communication of data between the preCICE coupling library and the developed coupling driver code.

OpenFOAM and OpenSees models are first generated separately for use within a coupled FSI simulation. These models are organized into subfolders within a primary coupled analysis directory. The OpenFOAM model is constructed with FSI boundary conditions at the locations of interest which are coupled to the OpenSees model. The coupled finite elements and nodes of the OpenSees model should be located within the volume generated by the 3D surface of the OpenFOAM CFD boundary region to which it is to be coupled. Residuals for OpenFOAM and OpenSees internal solution loops are defined through the finite volume solution settings within OpenFOAM and through the establishment of convergence tests within OpenSees.

The preCICE coupling library is configured for the given coupled model use case. The FOAMySeesCouplingDriver code is initialized at the beginning of the coupled FSI simulation. This driver code constructs the OpenSees model of interest within the same Python instance as the main solution algorithm loop of the coupling driver. The FOAMySeesCouplingDriver then establishes nodal connectivity relationships for the geometric mapping routines between CFD and FEA mesh nodal locations through K-means clustering techniques. Checkpoints are stored and loaded as necessary by the FOAMySeesCouplingDriver during the coupled solution for iteration of the structural model upon a set of applied forces from a particular state within a given coupling timestep.

The OpenFOAM simulation runs in parallel to the FOAMySeesCouplingDriver and is connected to the coupled solution through the standard OpenFOAM coupling adapter for the preCICE library. Each timestep the OpenFOAM preCICE coupling adapter is used to calculate the forces at the coupled interface from the CFD model running in parallel and to apply the displacements calculated by the FOAMySeesCouplingDriver as boundary conditions. The FOAMySeesCouplingDriver is also connected to the coupled FSI solution through the preCICE library and utilizes Python language binding function calls to the preCICE coupling library API within each coupling timestep. Python commands to OpenSees are utilized each timestep for dynamic application of loads calculated by the parallel CFD model to the OpenSees FEA model and for retrieval of kinematic response

quantities from OpenSees for application as boundary conditions within the CFD model at the coupled interface. Progression of the coupled solution to the next timestep is controlled by the preCICE library through coupling settings defined within the preCICE coupling library configuration XML file. How and when the exterior loop of the coupling algorithm progresses to the next coupling timestep is controlled by the preCICE coupling library. Coupling iterations can be utilized in the external coupling loop or not, but depending on specifics of the problem of interest the use of coupling iterations may largely be necessary in maintaining stability of the solution.

Each participant in the coupling must be prepared before or during model initialization for use in partitioned coupling of the simulations through the preCICE library. The details of these procedures for initializing a coupled solution between OpenFOAM and OpenSees through the preCICE coupling library in the context of the FOAMySees coupling approach are presented in the following Sections 4.2.2 and 4.2.3. A graphical overview of the contributions of all participants in the coupling process is shown in Fig. 4.7. Software participants are listed along the left hand side of the figure. The order of operations for both FOAMySees initialization and time-stepping is illustrated as an initialization phase, two interior loops (one solution loop for CFD solution internal residuals and one for those of the structural solution internal residuals), and one exterior loop per coupling timestep which ensures satisfaction of convergences of coupling variables passed between software participants.

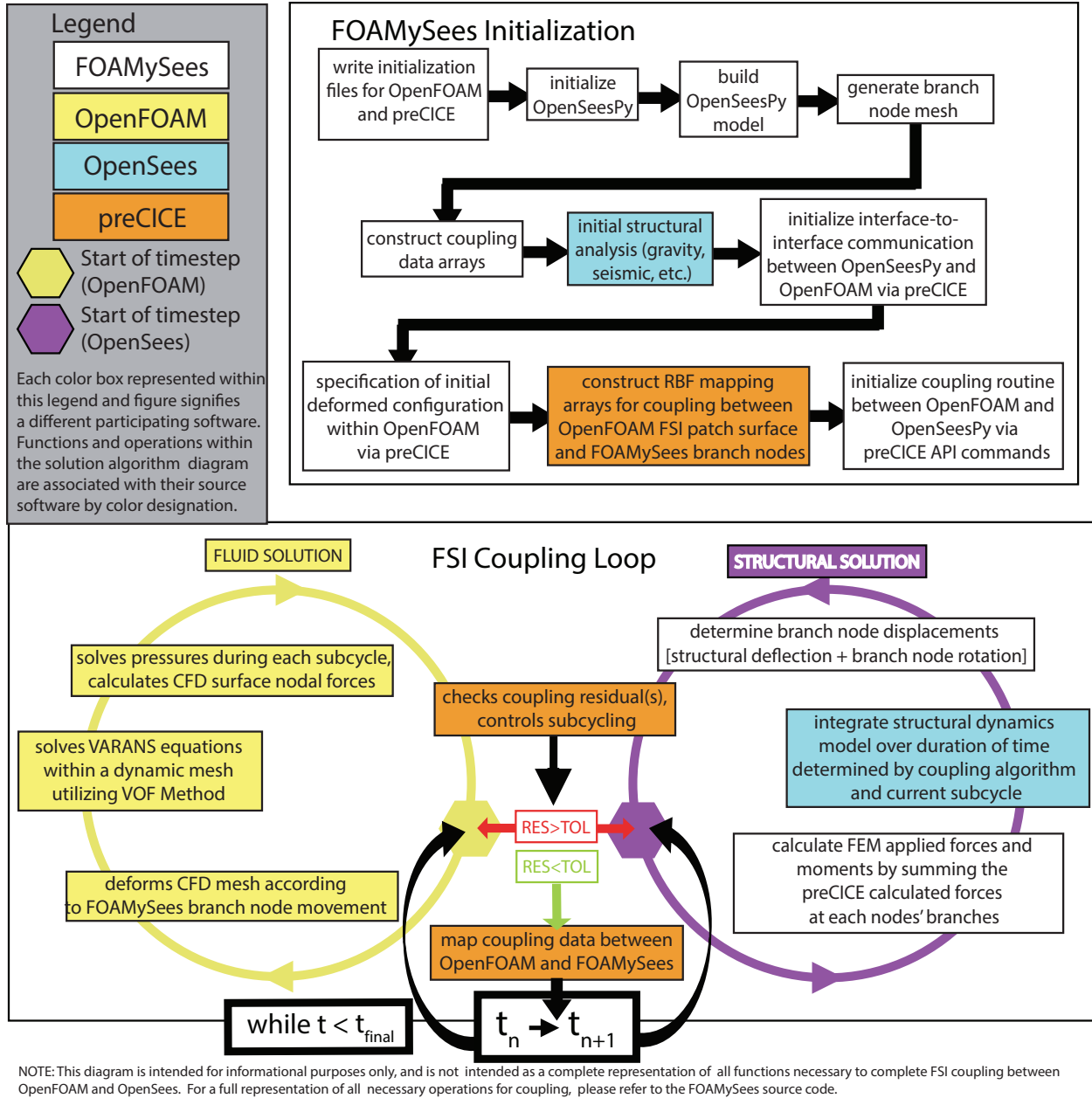


Fig. 4.7. Coupling Procedure for FOAMySees (abridged)

4.2.2 Coupled Model Participant Configuration and Initialization

Preparation of OpenFOAM model and OpenSees model components for utilization within a coupled FSI simulation does not vary greatly from the definition of individual FEA simulations with OpenSees or CFD simulations within OpenFOAM. Each participant of the coupling is started

individually as a process with a bash shell script in Linux operating systems. Communication of data between software programs is conducted with the preCICE coupling library. Details on configuration and initialization of each software component of the coupling approach are provided in the following subsections.

preCICE Configuration

Coupling settings are managed and defined in a preCICE configuration XML file, which includes the coupling timestep data and partners, residual convergence tolerances, coupling acceleration techniques (Aitken, Broyden, IQN-ILS, IQN-IMVJ), and data exchange mechanisms for coupling (TCP/IP sockets vs. MPI). The file is structured as follows:

- preCICE Output Logging Settings
- Coupling Data, Dimensions of Solution
- Coupling Participants and Their Data
- Participant Data-To-Mesh Association, Mesh Usage For Each Participant, Mapping Method Specification
- Data Exchange Mechanism (MPI vs TCP/IP Sockets)
- Coupling Scheme, Data Exchange Scheme, Explicit/Implicit Coupling Options

This configuration file is read by FOAMySees and OpenFOAM for a given analysis, and is utilized by preCICE during the coupled solution to ensure that coupling procedures as specified by the user are followed. For more information about the configuration of coupling partners within preCICE, see the preCICE documentation on coupling configuration (preCICE [2022b]). An example preCICE configuration XML, which is utilized for an explicitly-coupled simulation between the FOAMySeesCouplingDriver and OpenFOAM, is available in the Appendix under Listing "preCICE Configuration XML File Example".

OpenFOAM

The preCICE coupling adapter for OpenFOAM reads displacements from and writes forces to the preCICE coupling library. The standard release version of the OpenFOAM preCICE adapter is available on Github (preCICE [2022a]). A standard OpenFOAM file structure for a CFD model with dynamic mesh capabilities is maintained for use within a partitioned coupled FSI solution with preCICE, with an additional settings file and OpenFOAM function object for initializing connections to coupling routines in preCICE.

OpenFOAM preCICE Function Object

The preCICE coupling adapter is utilized within an OpenFOAM analysis via a function object of type `preciceAdapterFunctionObject`. This function object is defined along with other OpenFOAM function objects in the "controlDict" file in the "system/" subfolder of the OpenFOAM model. The `preciceAdapterFunctionObject` may be utilized with an entry in the OpenFOAM controlDict file similar to that shown in Listing "OpenFOAM preCICE Adapter Function Object" upon installation and addition of the preCICE library and OpenFOAM preCICE adapter to the appropriate environment variable paths.

OpenFOAM "preciceDict" preCICE Dictionary

The settings file necessary for coupling OpenFOAM to other software using the preCICE coupling library is the "preciceDict" file, which is stored within the OpenFOAM case "system/" subfolder. The `preciceDict` OpenFOAM dictionary manages how OpenFOAM receives data from preCICE and what locations in the OpenFOAM model data should be acquired from or passed to. The OpenFOAM preCICE adapter allows for specification of the variables within the OpenFOAM object registry that should be utilized for the calculation of forces and the application of boundary conditions within the CFD model. To summarize the `preciceDict`'s extents and purpose, the OpenFOAM

system/preciceDict file contains an OpenFOAM file header to register the object in the OpenFOAM namespace for use of its contents within OpenFOAM, the location of the preCICE configuration XML file with respect to the main folder of the analysis, the name of the coupling participant, and the coupling module of preCICE that should be utilized.

The coupling interface is specified in an 'interfaces' sub-dictionary, which includes the names of the coupled patches within OpenFOAM and the data that should be applied to the coupling interface during the solution (readData) as well as the data that should be sent to the other participants of the coupled solution (writeData). This sub-dictionary also allows for the extraction of OpenFOAM data from various components of the patches, which are specified in the coupling library. In the current version of the coupling implementation, forces are read from and displacements are written to the face center nodes of the cells that are attached to the coupled interface in the CFD model. The values written to these cell face centers by preCICE during the coupled solution are interpolated to the locations those solution variables are utilized at within OpenFOAM.

A sub-dictionary associated with the loaded preCICE coupling module is specified within the system/preciceDict file. For FSI simulations this is the 'FSI' sub-dictionary. For FSI simulations involving OpenFOAM solvers for multi-phase incompressible flows, the FSI sub-dictionary is defined with the pressure field specified as the OpenFOAM solution field variable 'p'. 'p_rgh' is a derived field utilized within the CFD solution and is not equivalent to the total fluid pressure, but rather the dynamic pressure. This field is reassigned as the temperature variable within the preCICE library to ensure that the 'p' field is utilized by preCICE for calculation of forces, and not 'p_rgh'. Setting this variable as the temperature variable in the preCICE FSI module does not affect the OpenFOAM solution, as the specified pressure field is simply read from OpenFOAM. However, without such a definition, the preCICE OpenFOAM coupling adapter FSI module could potentially calculate forces utilizing the 'p_rgh' field which would omit hydrostatic pressures from the force calculations. This seems to be related to the object registry in OpenFOAM, but can be circumvented with changes to the preciceDict FSI subdictionary as specified. The version of the

1292 preciceDict dictionary utilized in the coupled solution approach can be seen in the Appendix under
1293 Listing "OpenFOAM preCICE Dictionary".

1294 **OpenSees**

1295 An OpenSees model is defined within a standard OpenSees script for use within a FSI analysis
1296 conducted via FOAMySees. In the context of FOAMySees, this OpenSees script is placed
1297 within a directory to be utilized by the FOAMySeesCouplingDriver to initialize an instance of
1298 the structural finite element model which is to be analyzed. This is done as part of the same
1299 Python instance as the coupled FSI simulation contained within the FOAMySeesCouplingDriver
1300 Main Solution Loop (described in Section 4.2.4). Utilization of a shared Python instance for
1301 the initialization of the OpenSees model, the preliminary analysis, and the main solution loop
1302 of the FOAMySeesCouplingDriver allows for the preservation of structural analysis and loading
1303 performed within the OpenSees model initialization script throughout any subsequent FSI analysis.
1304 OpenSees models are defined through standard OpenSees framework Python language interpreter
1305 commands, and this script is loaded into the coupling driver to be utilized in the coupled analysis.
1306 A list of coupled node numbers can be specified, otherwise all nodes of the finite element model
1307 defined within OpenSees are coupled to the CFD model. The locations of the coupled nodes of
1308 the OpenSees model are used by the coupling driver upon initialization of the coupled model for
1309 clustering of CFD mesh surface nodes to the coupled nodes of the finite element mesh. OpenSees
1310 model settings such as convergence test and algorithm options are controlled via files contained
1311 within the OpenSeesSettings/ subfolder or with definitions within the OpenSees model construction
1312 script. FOAMySees manages the time progression of the OpenSees model and mapping to and from
1313 the FEA mesh in the coupled solution.

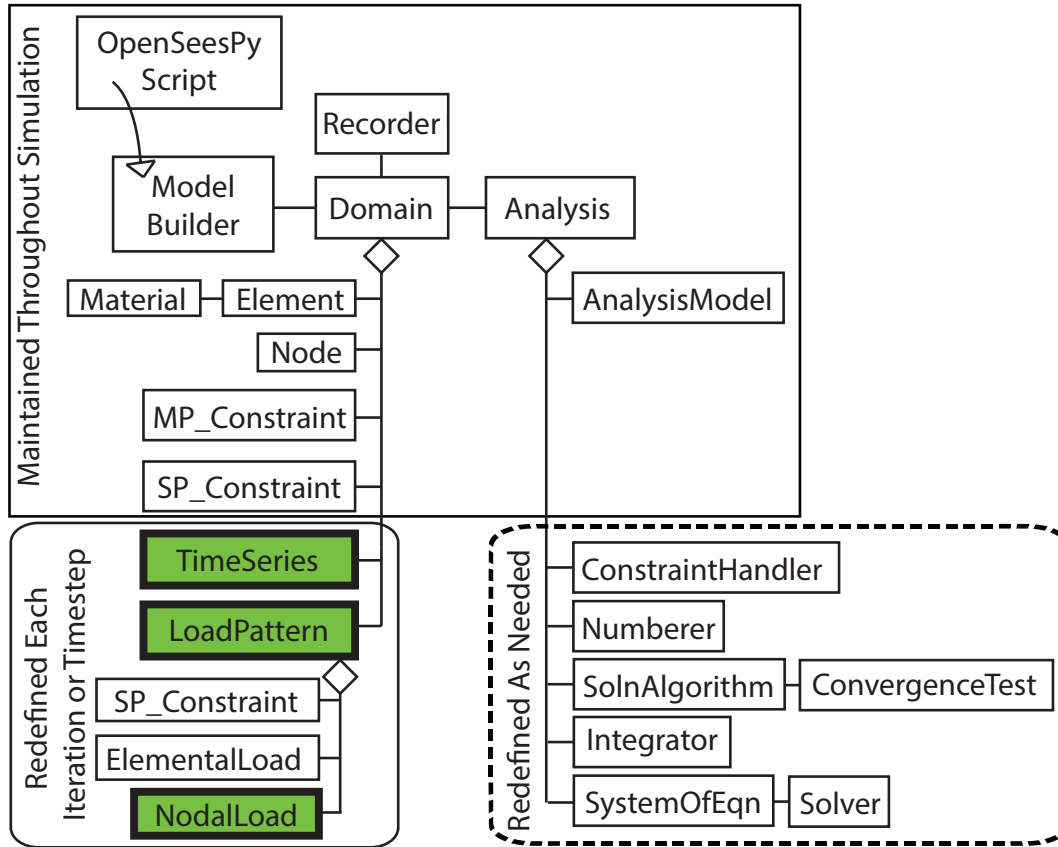


Fig. 4.8. Objects From OpenSees Framework Which Are Maintained from Initial OpenSees Analysis or Redefined as Part of Solution Procedure. Bold Bordered Objects Highlighted in Green Are Redefined Each Coupling Iteration and/or Timestep Within the FOAMySeesCouplingDriver

4.2.3 FOAMySees Initialization

Since the Python instance that constructs the OpenSees finite element model and completed the initial structural analysis within OpenSees is the same Python instance utilized for the coupled FSI simulation, access to the OpenSees **Domain** object associated with the constructed model is maintained and the state of the model after the preliminary structural analysis routines are completed is preserved to be utilized for additional analyses. The constructed OpenSees **Domain** object, as well as the OpenSees **Domain**-to-**Analysis** broker object, '**AnalysisModel**,' are preserved from the model definition and are utilized during the coupled FSI procedures to store the state of the FEA model at the conclusion of a given timestep or converged timestep iteration. OpenSees **TimeSeries**

and ***LoadPattern*** objects are defined within each coupling iteration for application of new loads via creation of OpenSees ***NodalLoad*** objects for each coupled node within the solution. A graphical overview of the OpenSees framework objects, which are constructed within the initial structural analysis or model definition and maintained throughout the analysis, is provided in Fig. 4.8.

Prior to entering the coupled FSI solution loop, the FOAMySeesCouplingDriver initializes the FOAMySeesInstance data interpolator. This involves loading the coupling data projection mesh into the coupling driver as a point cloud and clustering its nodes to the coupled nodes of the OpenSees model, establishing FEA-to-branch node relationships. The coupling data projection mesh is sent to preCICE by the FOAMySeesCouplingDriver for calculation of the interpolation mapping for force and displacement data exchange between OpenFOAM surface patch nodes and coupling data projection mesh nodes. The mapping type between meshes used by preCICE can be selected by changing the configuration settings within the preCICE configuration XML file.

Generation and Clustering of Coupling Data Projection Mesh

Initialization of the coupling data projection mesh is illustrated in Fig. 4.9. If the coupling interface geometry doesn't exist as a boundary within the OpenFOAM model to which the structure will be coupled, the boundary of the structural envelope can be meshed into the CFD model using OpenFOAM's **snappyHexMesh** tool. This will generate a body-fitted mesh around the structure within the OpenFOAM model. From a fully constructed OpenFOAM CFD model with a body-fitted mesh, coupling surfaces can be output to file using OpenFOAM's **surfaceMeshExtract** function. The surface representing the coupled CFD region is loaded into the FOAMySeesCouplingDriver for initialization as the coupling data projection mesh through the pyVista Python package (Sullivan and Kaszynski [2019]). The pyVista package allows for quick visualization and array storage of the vertex locations of an input surface file, and is used by FOAMySees to generate arrays of the vertices of the surface file as well as the center points between the vertices on each cell face of the coupled CFD model boundary surface. In the current version of the coupling approach, the cell face

center nodal locations function as the branch node locations for the coupling data projection mesh. Forces along the coupled surface within OpenFOAM are calculated at the face center nodes of the cells attached to the CFD patch surfaces. With cell face center nodes selected as the locations of the branch nodes comprising the coupling data projection mesh, forces for each cell in the CFD model are directly associated with branch forces. Due to this, FEA nodal applied forces and moments and CFD prescribed displacements can be calculated without the use of additional radial basis functions or mapping routines within OpenFOAM or preCICE.

The SciPy (Virtanen et al. [2020]) spatial package KD-Tree function (Archibald [2008]) is

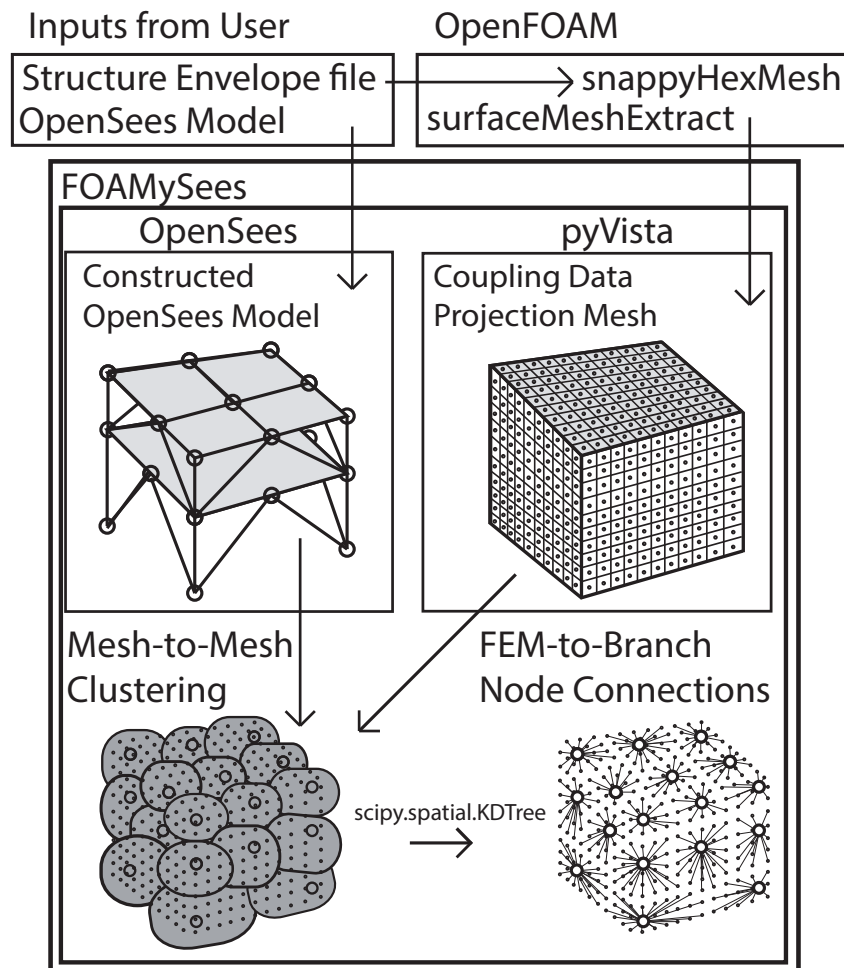


Fig. 4.9. FOAMySees Initialization of Coupling Data Projection Mesh

utilized for calculation of K-means clustering of points of each mesh to one another. The method used is more specifically called a 'sliding-midpoint KD-Tree', as described in Maneewongvatana and Mount [1999]. Using an array of coupled finite element analysis nodal locations and the array of the cell face center nodal locations of the coupled CFD surface, the SciPy.Spatial.KDTree method contained within the SciPy Spatial package is used to query the location of the nodes within one mesh and determine the nearest matching nodal locations within a second mesh, returning lists of clustered nodes representing the FEA-to-branch node relationships. These node-to-node relationships are stored in lists of lists with FEM node numbers and their associated branch node numbers for reference and indexing of arrays within coupling routines.

Initialization of a Coupled Interface Mesh and Data Within FOAMySees through preCICE Python Language Bindings

Connection between the FOAMySeesCouplingDriver and the preCICE coupling library is established in Python by importing the preCICE API Python language bindings package and utilizing the bindings to create a preCICE object of the class *SolverInterface*. The name and location of the preCICE coupling configuration XML file is sent to this object, as well as the name of the coupling participant which is currently being connected to the coupling library. The coupling configuration XML file is the same as that of the OpenFOAM model to which the OpenSees analysis will be coupled. The method to create a *SolverInterface* object through the preCICE API requires an additional two entries pertaining to the process index and the process size of the coupling participant connecting to the preCICE library. For an un-parallelized instance of the FOAMySeesCouplingDriver, the values for the solver process index and the process size (number of processors of the current participant that should communicate with the preCICE library) are 0 and 1, respectively. In the case of FOAMySees, the Python instance hosting the coupling driver and the OpenSees model is running on a single processor, but if a parallelized OpenSees model were run the process size and the process index may differ. Further details can be found within the preCICE C++ library documentation under the

SolverInterface class (preCICE C++ API Documentation [m], preCICE C++ API Documentation [a]).

Connection and association of the meshes and data defined in the preCICE coupling configuration XML file with the meshes and data contained in NumPy (Harris et al. [2020]) arrays within a Python instance is achieved through methods of the constructed preCICE API *SolverInterface* object. The nodal locations of the coupling data projection mesh are sent to the preCICE library with preCICE API commands '*getMeshID*' and '*setMeshVertices*' (preCICE C++ API Documentation [c], preCICE C++ API Documentation [l]). The '*getMeshID*' and '*setMeshVertices*' commands should be utilized after the *SolverInterface* object has been created and before the coupling mapping calculation has been completed. The data arrays for force and displacement vectors at the nodes of the coupling data projection mesh are associated with a particular data ID number, which is stored for use within the coupled solution loop. The ID number for each data array can be returned from preCICE with the '*getDataID*' method of the preCICE *SolverInterface* class (preCICE C++ API Documentation [b]). The data arrays are initialized before the main coupling solution loop and are read from or written to by preCICE each coupling iteration using their unique ID numbers.

After initialization of the coupling data communication mesh within the FOAMySeesCouplingDriver code, and the definition of the participant coupling data arrays and vertices within preCICE, the '*initialize*' method of the *SolverInterface* preCICE class is called (preCICE C++ API Documentation [d]). The initialize method establishes the mapping between the coupled meshes within preCICE, returns the coupling timestep length, and creates the necessary TCP/IP connection for data exchange during the coupled solution. The TCP/IP connection is created between the processes on all processors which are participating in the coupled solution in OpenFOAM and the individual process hosting the Python instance with the OpenSees model and FOAMySeesCouplingDriver. When connections between coupling partners and their meshes have been completed, connections between nodes of the various coupling participants are established and mapping between meshes is calculated as specified within the preCICE coupling configuration XML file. Initial

data can be exchanged between FOAMySees and OpenFOAM using the *'initializeData'* method of the *SolverInterface* preCICE class (preCICE C++ API Documentation [e]). From this point, the FOAMySeesCouplingDriver progresses to the main coupling solution loop.

4.2.4 FOAMySees Coupling Driver Solution Loop Procedure

The conceptual progression of a coupled model using a coupling data projection mesh within a single coupling timestep or timestep iteration can be described in 6 steps as seen in Fig. 4.10:

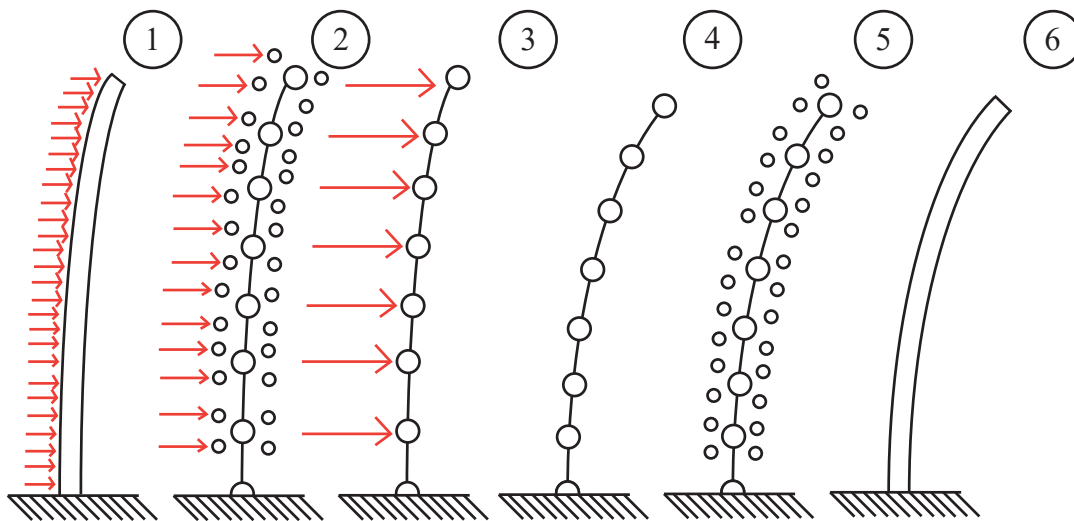


Fig. 4.10. Overview of Use of Coupling Data Projection Mesh

Upon entering the main solution loop, the following operations are performed each timestep in sequence by FOAMySees components preCICE, OpenFOAM, and OpenSees.

- OpenFOAM: Reads data communication mesh displacements and integrates the CFD model over time with mesh motion, calculates pressures and viscous shear stresses within a CFD timestep iteration
- preCICE: Integrates surface pressures and viscous shear stresses over the coupling boundary to forces at the coupling interface, maps forces to FOAMySees coupling data projection mesh via selected mapping methods and TCP/IP Socket Connection

- 1421 • FOAMySees: Reads forces from preCICE into arrays, calculates finite element model nodal

1422 applied forces, calculates relative changes in branch location for calculation of updated

1423 moments, calculates finite element node applied moments using nodal eccentricities between

1424 CFD cell face nodal locations and FEA nodal locations, and applies new net nodal forces and

1425 moments to coupled OpenSees nodes
- 1426 • OpenSees: Integrates structural dynamics finite element analysis model over time and returns

1427 nodal displacements and rotations to FOAMySees when converged
- 1428 • FOAMySees: Projects displacements and rotations from each FEA node to its associated

1429 branch nodes on the CFD surface as 3D point motion via rotation matrices and rigid body

1430 translations.
- 1431 • preCICE: Sends FOAMySees calculated data communication mesh displacement array to

1432 OpenFOAM for application of displacement boundary conditions and calculation of branch

1433 forces during the next coupling timestep or iteration

1434 Previously, an unnecessary step was added to the solution procedure and upon further study is no
 1435 longer implemented as part of the FOAMySees coupling procedure. The step was described in Eqn.
 1436 5 in Section 1.3.1 and is mentioned in Fig. 2 in Lewis et al. [2023] as part of the coupling routine.

1437 The mesh and operator of each step of the exterior coupling loop is shown in Fig. 4.11. Steps are
 1438 listed with their associated meshes on the right hand side and the operator of the action on the left
 1439 hand side. When two operator symbols are overlaid, this indicates that both operators were used in
 1440 the step. When another operator symbol is combined with FOAMySees, this indicates the utilization
 1441 of the Python language bindings for a given operator (OpenSees Framework or preCICE API) for
 1442 extraction of data from the operator to be used in calculations within that or a following step.

FOAMySees Coupling Driver Solution Loop Outline

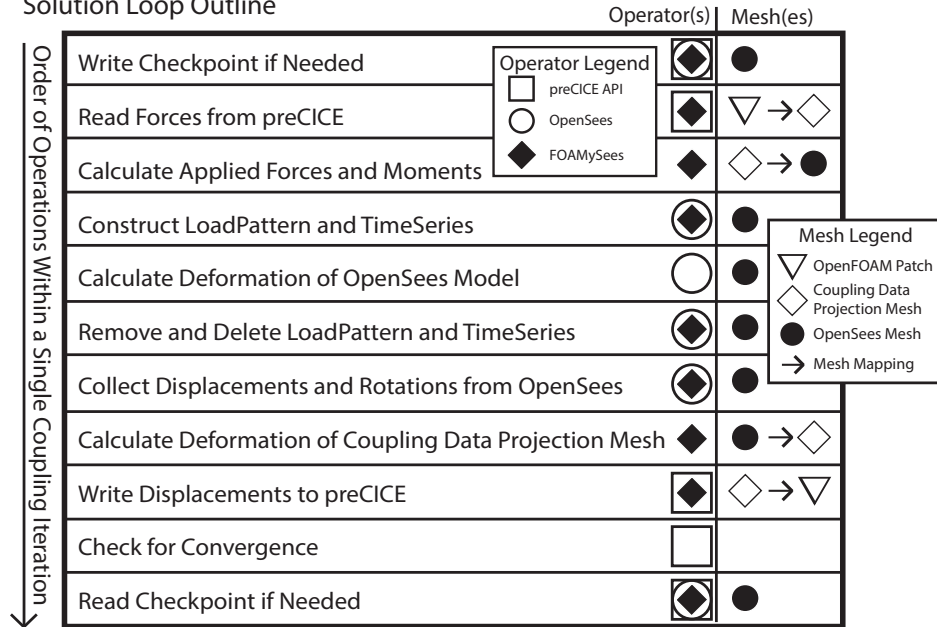


Fig. 4.11. FOAMySees Coupling Driver Solution Loop Summary. Steps within the solution loop summary are marked on the left hand side of the figure by symbols associated with each API or framework used in the solution loop. Overlaid symbols indicate combined contributions to completion of the step within the solution loop. Meshes used for the step are shown on the right hand side of each step.

4.2.5 preCICE API Methods and OpenSees Framework Commands within FOAMySees Coupling Driver FSI Solution Loop

The main solution loop for the coupling driver is available online, at the FOAMySees Github repository (FOAMySees, Lewis [2023]). The coupling solution steering is controlled primarily through methods of the *SolverInterface* object of the preCICE API, connection to which is established during the coupling driver initialization (preCICE C++ API Documentation [a]). The solution algorithm is contained within a while loop, which operates for as long as function calls to the method '*SolverInterface.isCouplingOngoing*' within the preCICE API return a boolean value of 'true' (preCICE C++ API Documentation [g]). The duration of the simulation is controlled by the preCICE configuration XML file. When either OpenFOAM or the Python instance connected to

the preCICE library and the coupled OpenSees analysis are terminated or crash, the TCP/IP socket connection is broken. In these cases, the function call to *SolverInterface.isCouplingOngoing* returns a boolean value of 'false' and the coupling solution loop is exited.

Within each cycle of the while loop, a coupling iteration is completed. At the beginning of the iteration, the *SolverInterface* object is queried through the preCICE API to determine if a checkpoint should be written by OpenSees. A checkpoint saves the state of the model at the onset of a new coupling timestep, which allows for the model to be utilized as part of the fixed-point iteration implicit coupling methods required for stabilization of certain FSI problems. Determining if a checkpoint needs to be written is assessed by passing an 'action' flag to the method '*isActionRequired*' of the *SolverInterface* object, namely the '*precice.action_write_iteration_checkpoint*' flag (preCICE C++ API Documentation [f]). If the *SolverInterface.isActionRequired* method returns a value of 'true' when sent a flag associated with writing a checkpoint, Typically, checkpoints are written prior to the first coupling iteration, allowing for the state at the end of the previously converged coupling timestep to be used as the start of the present iteration. FOAMySees uses OpenSees commands to write a checkpoint at the current state through the '*database*' and '*save*' commands in the OpenSees library (OpenSees [Documentation(b)]. OpenSees [Documentation(c)]). A new database save commit tag is utilized for every coupling checkpoint. Checkpoints for the OpenSees finite element model state are established by writing a new state to an OpenSees database which has been previously constructed at the start of each new coupling timestep. The checkpoint writing action is then utilized again through a method of the *SolverInterface* object '*markActionFulfilled*' (preCICE C++ API Documentation [h]). The steps conducted within a timestep iteration within the FOAMySees coupling driver solution loop are schematically summarized in Fig. 4.12.

Forces at the coupling data projection mesh are then read into the coupling driver from the preCICE *SolverInterface* object as a NumPy array through the '*readBlockVectorData*' method of the *SolverInterface* class (preCICE C++ API Documentation [j]). For each coupled node in the finite element solution, the node-to-branch relationship list constructed in the initialization through the

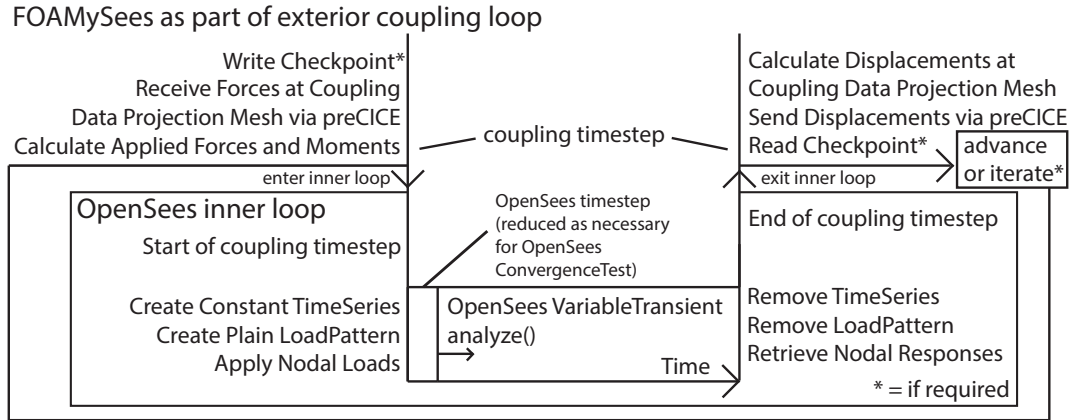


Fig. 4.12. FOAMySees Coupling Driver

1479 K-means clustering is utilized to index the NumPy array of forces and a NumPy array of branches.
1480 The indexed forces for each FEM node are summed to calculate the applied nodal forces for the
1481 coupled node of interest, and moments for the coupled node are computed by differences of a series
1482 of dot products between vertical slices of the force array and deformed branch node group arrays (see
1483 Eqns. 4.5 through 4.7). These forces and moments are applied for the coupling timestep as a nodal
1484 loads contained within a 'Plain' **LoadPattern** object in the OpenSees framework, and are associated
1485 with a 'Constant' **TimeSeries** object. After application of the nodal loads for all coupled FEM nodes,
1486 a **VariableTransient** time integration is conducted for the duration of one coupling timestep iteration
1487 within the OpenSees analysis through the '**analyze**' command (OpenSees [Documentation(a)]). The
1488 **LoadPattern** and **TimeSeries** objects are utilized for the coupling iteration only, and are immediately
1489 removed from the OpenSees model after completion of the analyze command with the '**remove**'
1490 command (OpenSees [Documentation(f)]). The displacements and rotations of the coupled nodes
1491 of the finite element solution are then requested from OpenSees using the '**nodeDisp**' command
1492 (OpenSees [Documentation(d)]). These displacement and rotation values are utilized along with
1493 the nodal branch locations to project displacements through rotation matrices to the coupling data
1494 projection mesh, and are stored in the displacement array of the coupling data projection mesh. The
1495 displacement array of the coupling data projection mesh is sent to the preCICE library with the

1496 method '*SolverInterface.writeBlockVectorData*' (preCICE C++ API Documentation [k]), and the
1497 model is advanced using a function call to the '*SolverInterface.advance*' method (preCICE C++
1498 API Documentation [i]). The coupling driver uses the *SolverInterface.isActionRequired* preCICE
1499 method with a '*precice.action_read_iteration_checkpoint*' flag to check if the timestep has converged
1500 or if a database checkpoint needs to be restored through the OpenSees '*restore*' command (OpenSees
1501 [Documentation(g)]). The while loop which contains the coupling solution procedure is then
1502 restarted as either the next iteration of the coupling external loop or as the next timestep of the
1503 coupled solution.

Chapter 5

Analytical Test Cases

Multiple idealized test cases were examined in this chapter to demonstrate the utilization of the coupling methodology in various conditions, which can be validated with predictions from analytical methods and equations. This chapter includes a quasi-static test case, a test case comparing end states of a dynamic simulation and an analytical prediction within the context of a parametric study, and a dynamic test case simulating a dam break of a volume of water against a flexible structure. The dam break test case was compared to results from the simulation conducted within other simulation codes for free end displacement of the structure and free surface profiles of the water. The dam break test case was then investigated as part of a study on the transfer of states between initial structural analysis routines involving nonlinear materials and a subsequent FSI analysis, and as part of a FSI simulation with a concurrent ground-motion.

5.1 Hydrostatic Force FSI Case

A quasi-static FSI case was created using a beam with fixed-end conditions to test the calculation of displacements of a simple structure under uniform loading in quasi-static conditions. The CFD domain was 10 m x 2 m with a 0.1 m thickness in the direction perpendicular to the plane of

deformation. The finite element model was comprised of beam elements, which spanned along the base of the CFD model. Water was initialized at a depth of 0.5 m within the CFD domain. Under hydrostatic conditions, this configuration can be simplified as a uniform distributed load across the bottom of the domain with a value of 490.5 N/m, for a total applied vertical load of 4905 Newtons on the beam. A diagram of the case showing reduction of the fluid loading to a distributed load is shown in Fig. 5.1. The modulus of elasticity of the beam was varied and the displacements calculated by the coupled model were compared to expected displacements from Euler-Bernoulli beam theory for a fixed-end beam under uniform loading.

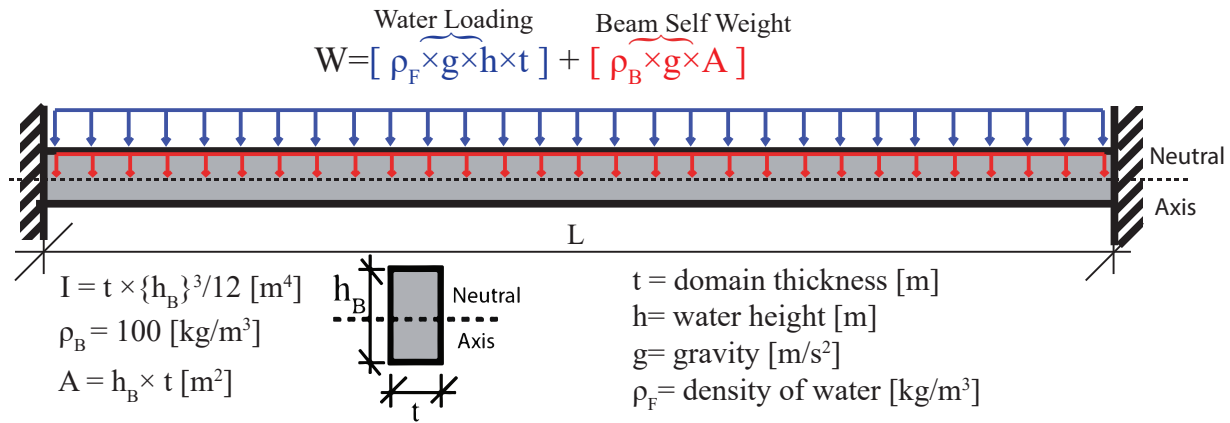


Fig. 5.1. Beam with properties E , I , L , A , ρ_B , within a domain t meters wide, topped with water h meters deep, under uniform gravity loading g

Coupled Analysis Details

An implicit coupling scheme with a coupling timestep of $\Delta t = 1e - 3s$ was utilized in this simulation to ensure stability of the partitioned solution. As the mass of the water is bearing directly on the coupled interface of the CFD model, motions of the structural boundary largely affect the forces imparted by the water. Without interface acceleration techniques or sufficient damping, the oscillations between the displacement of the beam and the applied forces can result in divergence of the coupled solution. The coupling data acceleration method utilized was an IQN-ILS algorithm

with a coupling convergence tolerance of $1e-3$, ensuring that data transferred over subsequent coupling iterations reaches a relative convergence of 0.1 % before progressing to the next coupling timestep.

OpenFOAM Analysis Details

An olaFlow OpenFOAM model was utilized for the following study. The finite volume domain was discretized with 40 cells in the horizontal direction and 8 cells in the vertical direction, with the direction perpendicular to the page given 1 cell thickness. The OpenFOAM fvSchemes and fvSettings files can be found in the OpenFOAM Appendix (A.1.3). Standard values for density and viscosity were utilized for the air and water phases ($\rho_{water} = 1,000 kg/m^3$, $\nu_{water} = 1.0 \times 10^{-6} m/s^2$, $\rho_{air} = 1.22 kg/m^3$, $\nu_{air} = 1.48 \times 10^{-5} m/s^2$, where ρ = mass density of the fluid; and ν = kinematic viscosity). Boundary conditions as described by Table 5.1 were utilized for the patches of the OpenFOAM model. The atmosphere was given an inlet/outlet boundary condition for velocity. The walls perpendicular to the plane of deformation were modelled in OpenFOAM as 'empty' patches for proper simulation of a 2D case. The walls at the fixed ends of the beam were prescribed static boundary conditions for both velocity and pressure. The FSI interface was given a velocity boundary condition of 'movingWallVelocity,' and a point motion boundary condition as prescribed by the coupling code through a 'fixedValue' boundary condition. Pressure boundary conditions are assigned a 'fixedFluxPressure' condition except at the atmosphere, where a 'totalPressure' boundary condition was specified to allow for mass inflow and outflow through the top of the domain.

OpenSees Analysis Details

Within OpenSees, the beam was modelled with 'dispBeamColumn' elements. The Average Acceleration form of the Newmark time integration scheme with $\gamma = 0.5$ and $\beta = 0.25$ was utilized throughout the analysis. A 'NormUnbalance' convergence test was utilized with a tolerance of $1e-8$ and a maximum number of iterations of 1000. The 'RCM' numberer was used, with

transformation constraint handling, a 'BandGen' system, and a Kyrlov Newton algorithm. Legendre beam integration with two integration points was utilized. Uniform modal damping was applied on the model, with a value of 5 % from the first mode frequency to five times the first mode frequency. The OpenSeesPy script utilized for model construction can be seen in the OpenSees Appendix under Listing A.5. A moment of inertia of 100 m^4 was utilized to reduce the order of magnitude of displacements to maintain a uniform distributing load pattern from the hydrostatic pressures within the coupled analysis.

Study Variable

The modulus of elasticity of the elastic beam was varied to verify expected forces are applied from OpenFOAM to the structural nodes within OpenSees and to verify that correct displacements are returned from OpenSees to OpenFOAM. The modulus of elasticity of the beam was deemed the least consequential variable to change with respect to other variables, as varying only the modulus of elasticity between cases for a parametric study allows for a consistent domain size, VOF phase-fraction, beam mass, beam element size, cell size, and beam cross section dimensions while providing a linear change in displacement with respect to the elastic modulus.

5.1.1 Results

The expected center-span displacement for a fixed-fixed Euler-Bernoulli beam under a uniform distributed load is $WL^4/384EI$. With an equivalent distributed load of 490.5 N/m ($0.5\text{m} \times 0.1\text{m} \times 9.81\text{m/s}^2 \times 1000\text{kg/m}^3$), a length of 10 m, a moment of inertia of 100 m^4 , and with elastic modulus of $5\text{E}+09 \text{ Pa}$, the expected center-span displacement is $-2.55469\text{E}-08 \text{ m}$. center-span displacements for beams with lower values for modulus of elasticity are shown in Table 5.2, along with the calculated center-span displacement for the beam from the presented coupled FSI model with identical properties.

Table 5.1. Closed Domain - Model Boundary Conditions for All Field Variables

Field	Atmosphere	Walls	Structural Interface
U	pressureInletOutletVelocity	noSlip	movingWallVelocity
p_{rgh}	totalPressure	fixedFluxPressure	fixedFluxPressure
pointDisplacement	fixedNormalSlip/fixedValue	fixedNormalSlip/fixedValue	fixedValue
α	inletOutlet	zeroGradient	zeroGradient

Table 5.2. Fixed Fixed Beam: Predicted $[WL^4/384EI]$ vs. Simulated Displacements, $W = 490.5[N/m]$, $L = 10[m]$, $I = 100[m^4]$

E [Pa]	Theoretical Δ	Simulated Δ	Theoretical/Simulated
1.00E+09	-1.27E-07	-1.28E-07	0.992
5.00E+08	-2.55E-07	-2.56E-07	0.996
2.50E+08	-5.10E-07	-5.11E-07	0.998
1.25E+08	-1.02E-06	-1.02E-06	1.000
6.25E+07	-2.04E-06	-2.04E-06	1.000

5.1.2 Mesh Convergence Study of Branch Interpolator

A mesh convergence study was conducted on the fixed-end beam case examined in the previous subsection using a reduced moment of inertia and cross-sectional area. This study was conducted to examine the effects of the selection the finite element mesh on the forces and displacements resolved by the OpenSees analysis. The efficacy of the displacement mapping routines at various finite element mesh resolutions and effects of the clustering on the applied and reaction forces were evaluated by comparing simulated results to theoretically expected values for force and displacement. The beam mesh was varied from 8 elements to 48 elements, with intermediate values of 12, 16, 24, and 32 beam elements. The mesh of the CFD domain was held constant at 40 elements along the beam to investigate the effect of the ratio of the number of CFD face centers to finite element nodes and the effect of relative mesh node locations on the mapped forces. The CFD model mesh with fields for phase fraction and pressure are shown in Fig. 5.2 and Fig. 5.3, respectively.

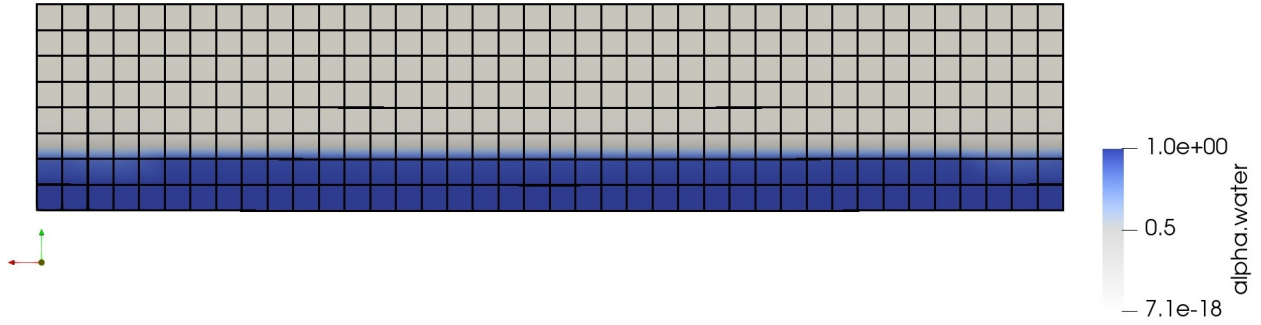


Fig. 5.2. Phase-fraction (α) field for the CFD model

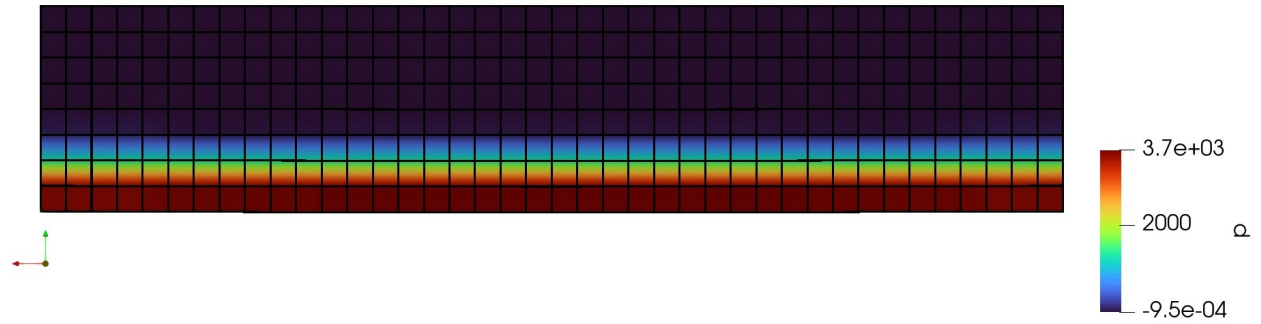


Fig. 5.3. Pressure (p) field for the CFD model

OpenSees Analysis Details

The finite element model was unchanged from the previous section, except for the structural properties. A modulus of elasticity (E) of 200 MPa was utilized for the beam finite elements with a Poisson's ratio (ν) of 0.3, a shear modulus (G) equal to $G=E/(2*(1+\nu))$, a flexural moment of inertia of 0.00833 m^4 , and a cross-sectional area of 0.1 m^2 . These section properties were changed from the previous study to increase the displacement by a few orders of magnitude, and to correspond with the section properties of the beam's geometry in three dimensions for comparison in the next section with a monolithic three dimensional solution. Constrained finite element nodes were included in the coupling mesh K-means clustering routine, which affected the resolved reaction forces depending on the finite element mesh resolution with respect to the CFD interface mesh resolution.

Mesh Convergence Study Results

All mesh resolutions offered displacements at the end state of the analysis that are identical to the prediction offered by theory. The time history of the center-span displacement for all simulations in the mesh convergence study is shown in Fig. 5.4. center-span displacements from simulations with all finite element mesh resolutions were normalized by the expected displacement from Euler-Bernoulli beam theory.

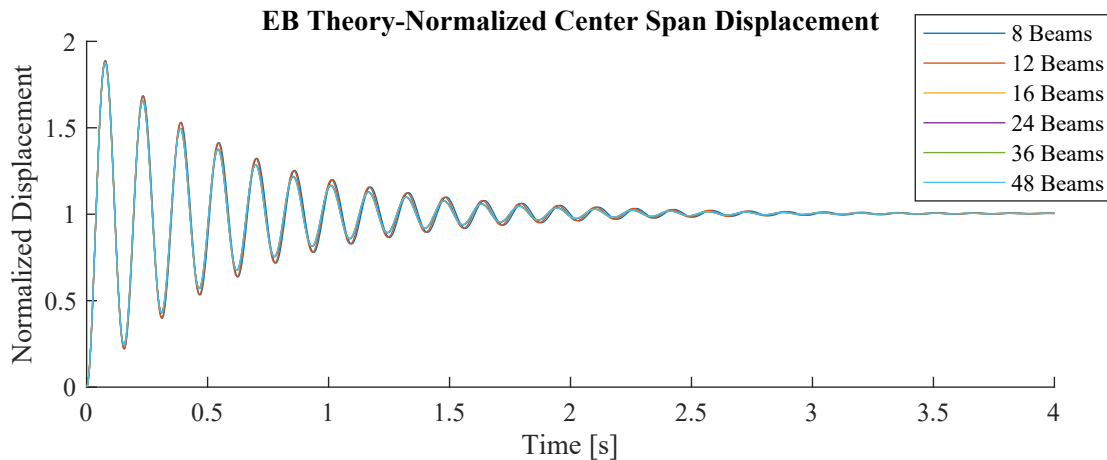


Fig. 5.4. Center-span Displacement Normalized by Euler Bernoulli Beam Theory

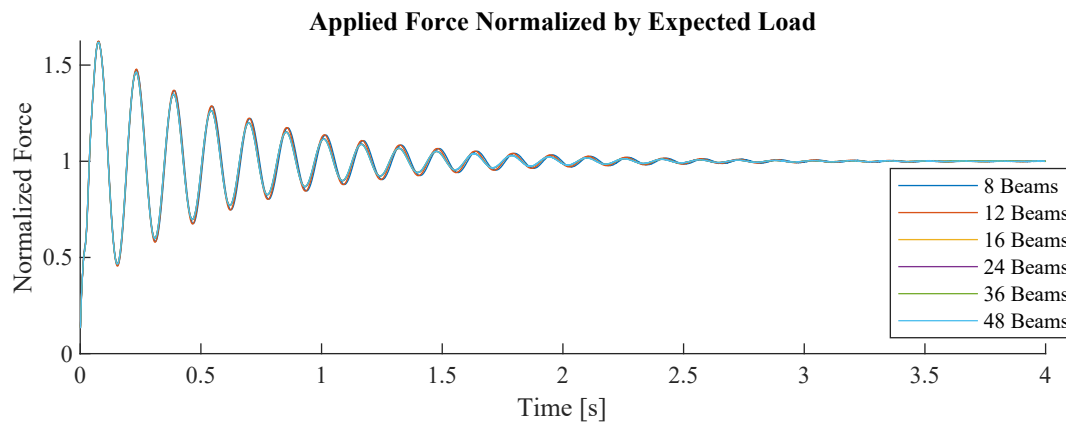


Fig. 5.5. Applied Forces Normalized by Expected Load (4905 N)

The applied forces for each case were normalized by expected vertical loading (4905 N) and plotted in Fig. 5.5. The applied forces from the CFD simulation converged on the expected loading

with little to no variation as the finite element mesh resolution is increased. Slight changes in peak force during the dynamic stages of the analysis between solutions with different finite element mesh resolutions were present, caused by over-prediction of center-span displacements during the dynamic peaks of the simulation from simulations with fewer beam elements. This over-prediction of dynamic peak displacement can be attributed to less acting mass within the beam and the reduction in concentration of forces and masses at nodes along the beam with an increasing amount of finite elements and nodes. As more elements were added to the beam, the mass of the beam was captured with higher accuracy (less mass was integrated to the beam ends through the shape functions) and the beam loading became more uniform, resulting in lower displacements. This effect is most clear in Fig. 5.4, in which the displacements for the FEA domain discretized with 48 elements are lower than those of the models with coarser finite element discretizations.

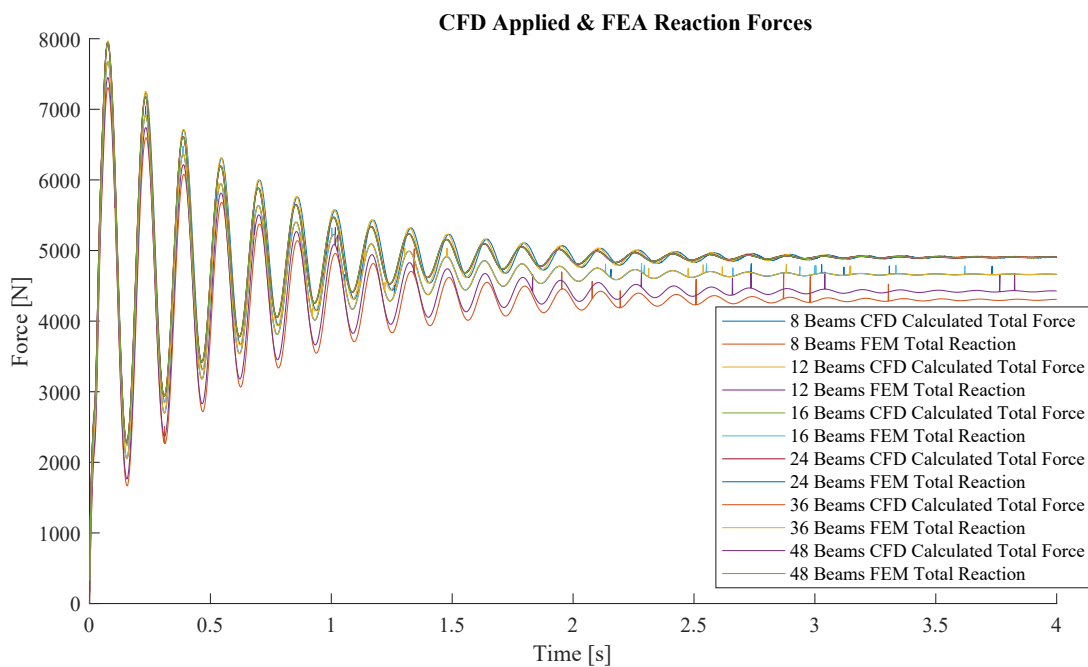


Fig. 5.6. Total OpenSees Constrained Node Reaction Forces and OpenFOAM Applied Forces

Forces from both the OpenFOAM model and the OpenSees models at various finite element mesh resolutions can be seen in Fig. 5.6. The beam settled over time under the weight of the water,

slowly being damped by the applied Rayleigh damping and the viscous motion of the water above the beam. OpenSees reaction forces changed with the beam mesh resolution due to the nature of the K-means clustering of the finite element nodes to the cell face center nodes of the CFD model. As the fixed nodes of the beam were included in the nodes utilized in the coupling routine, the K-means clustering routine of the coupling initialization clustered some of the cell face center nodes to the nodes which are constrained within the model. The OpenFOAM mesh resolution in the horizontal direction was 40 cells, resulting in each CFD cell face center node possessing 2.5 % of the total applied load. With an increased finite element mesh resolution, fewer cell face centers were clustered to the fixed nodes at the ends of the beam and more of the applied load was captured in increments of 2.5 %, resulting in higher reaction forces.

Coupling Mesh Convergence

When the FEA domain was discretized with 8 beam elements, 5 cell face center nodes near the edges of the CFD model were coupled to the constrained nodes of the finite element model, resulting in a total reaction force which was 87.5 % of the applied load from OpenFOAM. Increasing the mesh resolution to 12 beams resulted in a total reaction force which was 90 % of the applied load, with 4 CFD face center nodes coupled to constrained finite element nodes. The 8 element and 12 element finite element models experienced spikes in reaction force of magnitudes 5 % and 7.5 % of the total applied force, demonstrating the need for more elements to reduce the discrepancy in motion of the CFD model and the FEA model's coupling data projection mesh nodes and to capture more reaction load. Increasing the mesh resolution further to 16 beam elements reduced the number of CFD cell face center nodes coupled to constrained FEM nodes to 2, resulting in a total reaction of 95 % of the applied load. Spikes of force of 2.5 % of the applied force occurred, causing the total reaction load to jump to 97.5 % of the total load during individual coupling iterations. Results did not change until the finite element mesh resolution was increased beyond the point that allowed the CFD cell face center nodes to be geometrically closer to unconstrained FEM nodes than constrained nodes.

For the current CFD model at its present mesh resolution, the use of 48 beam elements in a coupled finite element model fully-captured the applied load from OpenFOAM. Force spikes due to changes in the mapping routine do not occur in the 48 beam element mesh results as the coupling mapping has converged. The converged beam mesh is shown along with the CFD model for reference in Fig. 5.7, with finite element nodes represented at the base of the CFD model as rendered points. If the constraint at the constrained node is not rotational, moments from the forces will be applied to the node, but shear forces will be still be lost. Increasing the CFD mesh resolution would similarly resolve the amount of force captured, but some face cell center nodes could still end up coupled to constrained nodes resulting in force losses during the mapping routines. Rather, it is recommended that constrained finite element nodes are removed from the coupling routines except when meshes are fine enough in the near region of constrained nodes to fully capture loads without their removal. Removal of constrained nodes from the coupling routine can assist in capturing the full applied load from OpenFOAM as the CFD face center nodes which previously would be clustered to the constrained FEA nodes would instead be clustered to the nearest unconstrained FEA node.

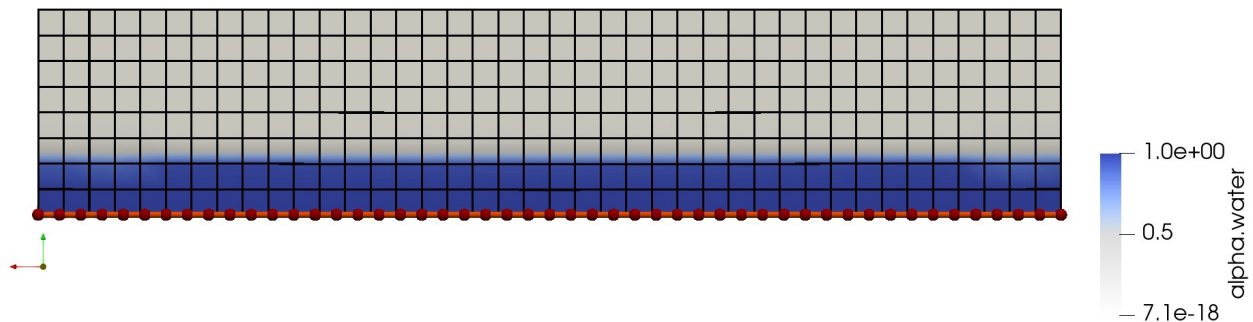


Fig. 5.7. FEM nodal locations for case with 48 beam elements and 40 CFD cells in x direction

Spurious Fluctuations in Reaction Force Due to Mapping Changes

Spikes in reaction force can be seen in reaction forces for the unconverged meshes. As the force spikes don't exist in the OpenFOAM applied force time histories or in the converged mesh reaction

1668 force time history, and also occur in increments of roughly 2.5, 5, and 7.5 % (with noise), these
1669 reaction force spikes can be attributed to the effect of the motion of the unconverged coupling data
1670 projection mesh nodes on which CFD face center nodes they are mapped to within the coupling
1671 routine. As the beam deforms, the CFD mesh attempts to deform along with it according to its
1672 displacementLaplacian mesh motion solver, resulting in new locations of the CFD cell face centers
1673 with respect to the coupling data projection mesh. As this connection between the coupling data
1674 projection mesh and the CFD cell face center nodes is not rigid, motion of the two interfaces
1675 (coupling data projection mesh and CFD patch surface) can differ during the coupling iteration until
1676 the timestep of both models have converged. If a CFD cell face center node switches mapping nodes
1677 during a timestep iteration, this is because its cell was stretched by the point displacement boundary
1678 condition enough to make the cell face center geometrically closer to another branch node of the
1679 coupling data projection mesh. This could happen in certain circumstances (such as large rotations
1680 or displacements, large coupling branch lengths, and/or large timesteps) which result in differences
1681 in relative position during a timestep iteration of the coupling data projection mesh and CFD cell
1682 face center nodes.

1683 In the case of the unconverged beam meshes, some of the CFD mesh face center nodes, which
1684 previously were mapped to the constrained finite element nodes at the exterior, were instead mapped
1685 to unconstrained finite element nodes during particular timesteps or timestep iterations. This
1686 happened to multiple CFD cell face center nodes during these timestep iterations, perhaps due to the
1687 symmetric motion of the beam during the loading resulting in the change in mapping routine on both
1688 ends of the beam. The spikes in reaction force show up until the motion of the points of the coupled
1689 CFD model follow the motion of the coupling data projection mesh closely enough to change the
1690 mapping back to the original mapping, whereupon the forces from the spike are transferred back
1691 to a constrained finite element node and the reaction force is lost. The duration and magnitude of
1692 the spikes in reaction force in Fig. 5.8 imply that these changes to the node coupling in the force
1693 mapping routine may happen spuriously as the relative positions of the nodes of the coupling data

projection mesh and CFD face center mesh vary incrementally over time. This only occurs when there are constrained finite element nodes included as part of the coupling mesh during the model initialization, particularly with finite element meshes which are not sufficiently resolved for the mapping routine with respect to the CFD model mesh resolution.

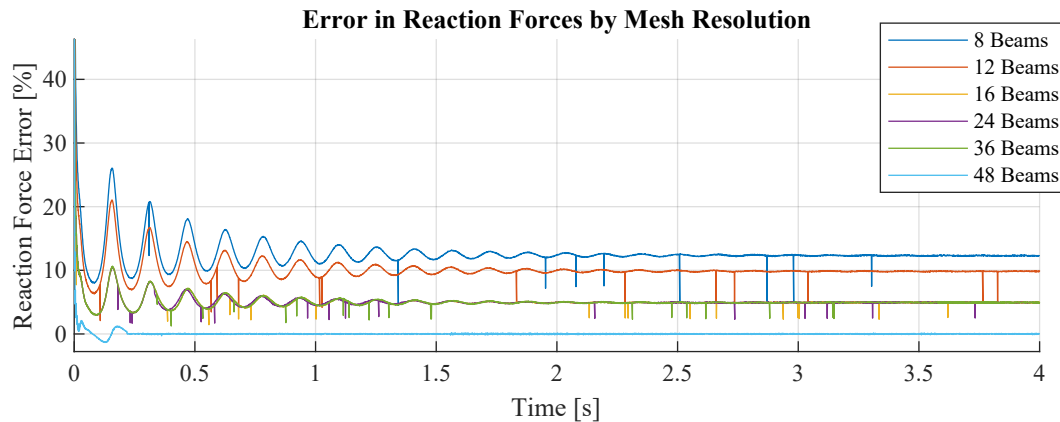


Fig. 5.8. Total Reaction Force Errors - Mesh Convergence

5.2 Fixed-end Frame Under Shifting Hydrodynamic Gravity Loading

The same domain geometry as described in the Hydrostatic Force FSI Test Case section was utilized to examine the effect of large displacements upon the shifting of water sitting atop fixed-end frames. The structural properties were chosen to amplify the effects of fluid pooling upon the deflection imparted against frame elements. As the deflection increases, the loading distribution becomes more concentrated near the beam mid-span, as fluid redistributes itself naturally to find a free surface that results in equilibrium between the internal structural forces and external fluid pressure. An illustration of this behavior is shown in Fig. 5.9.

Since the geometry, sections, materials, and loading chosen in this example result in large displacements, multiple phenomena contribute to the behavior of the frame elements, including

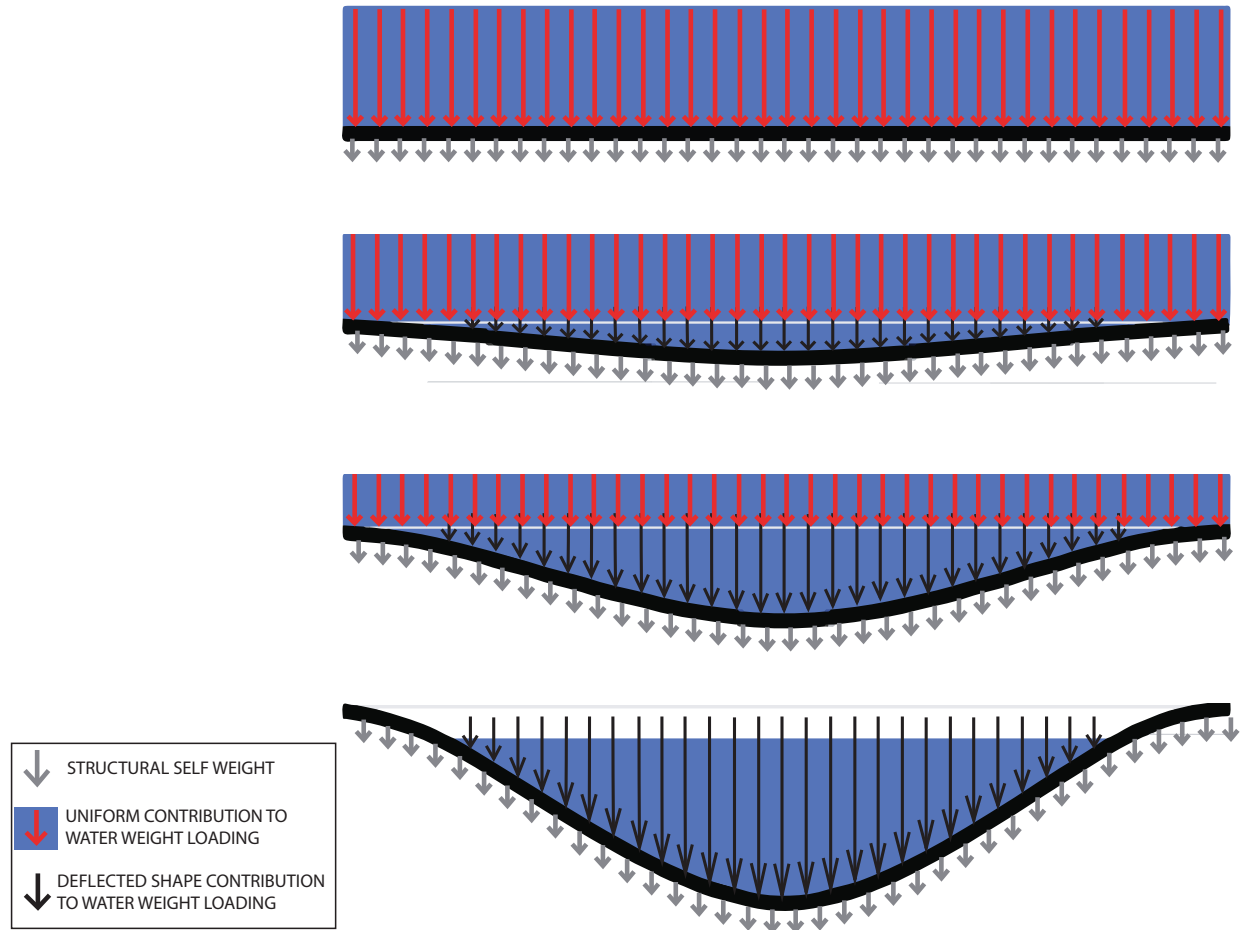


Fig. 5.9. Change of Loading Contribution with Displacement of Beams Under Shifting Water

1709 frame rotational stiffening effects due to tension. The modulus of elasticity was chosen as the
 1710 parameter to be varied, as maintaining section geometric properties allows for comparison of the
 1711 solution calculated by FOAMySees with solutions calculated by the Material Point Method (MPM),
 1712 for which structural sections must be modelled through-thickness with integration points. A loading
 1713 prediction function (Eqn. 5.1) for the force distribution across a beam under a pool of water was
 1714 developed for circumstances where beam deformations result in changes in hydrostatic pressure
 1715 under gravity. This loading prediction function was tested for the range of elastic modulus values
 1716 within the parametric study. Displacement results from a finite element analysis with only the
 1717 predictive loading function and gravity loads applied were compared to the displacements from both

FOAMySees and MPM. The predictive loading function performs well when predicting the loading from hydrostatic pressures for the cases tested, but the extents to which the function works with respect to other ranges of still water level and beam length was not investigated.

To illustrate a typical analysis within this study, snapshots were selected from a case within this example, particularly the case with the largest displacements and therefore the largest displacement to length ratio. In Fig. 5.10, four points in time during the simulation are marked as points A, B, C, and D respectively. In sub-figures Fig. 5.11 through 5.14, free surface elevation snapshots are displayed, showing the redistribution of water along the length of the frame and the settlement of the frame into a loading configuration that resulted in much larger deflections than what would occur for a uniform distributed load case. The end states of these dynamic analyses were utilized in the present comparison study.

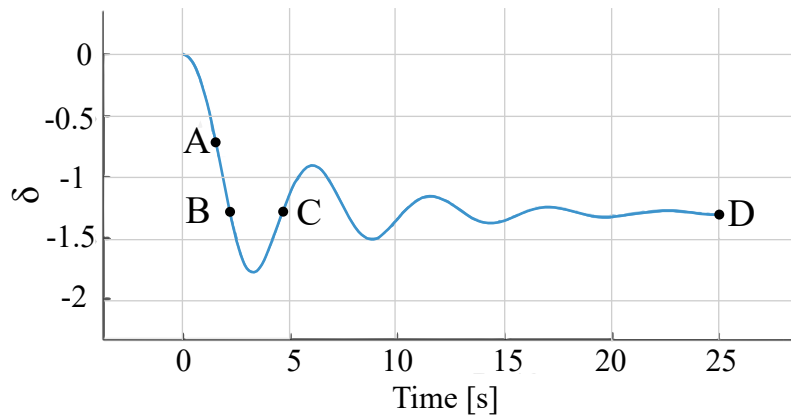


Fig. 5.10. FOAMySees Solution: center-span Displacement of $E = 1\text{E}6$ Pascals Beam Under Shifting Water Loading

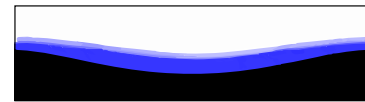


Fig. 5.11. Point A (1.5 s)

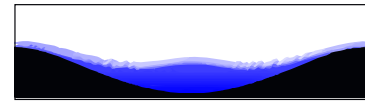


Fig. 5.12. Point B (2.2 s)

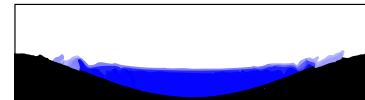


Fig. 5.13. Point C (4.7 s)

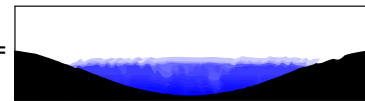


Fig. 5.14. Point D (25 s)

OpenSees Analysis Details

Frames for this test case possessed a moment of inertia of $8.33\text{E-}3 \text{ m}^4$, a length of 10 meters, a cross sectional area of 0.1 m^2 , and elastic modulus ranging from $1\text{E+}08$ Pa to $1\text{E+}06$ Pa. The beam

1732 mesh resolution was increased from 48 elements to 150 elements. The same OpenSees script as the
 1733 previous subsection was utilized in this study, with the modulus of elasticity and moment of inertia
 1734 variables modified accordingly.

1735 **Predictive Loading Function for Pooling**

1736 As the deflection to length ratio increases, so does the effect of the pooling of the water at the
 1737 center of the frame upon the expected maximum deflection. A loading prediction function for
 1738 determining quasi-static loading of beams under pooling water at various deflections was developed,
 1739 and is defined by Eqn. 5.1. Discrete loads calculated by integrating Eqn. 5.1 over the chosen frame
 1740 element length were applied dynamically in OpenSeesPy. Identical beam properties to those used in
 1741 the FSI simulations were utilized with the loading prediction function to provide an independent
 1742 benchmark against which FOAMySees calculated loads and results could be compared.

1743 The loading prediction function estimates the fluid elevation required above a particular deformed
 1744 frame shape to achieve quasi-static equilibrium between the beam in its deformed shape and the
 1745 'pooled' water sitting atop the beam. This calculation is completed after every timestep within a
 1746 dynamic analysis to account for the current deformed shape.

$$F(x) = t \times \rho_f \times g \times \left(\frac{\int_0^L \delta(x) dx - \frac{V_f}{t}}{L} - \delta(x) \right) \quad (5.1)$$

where

$g = \text{gravity}$

$\delta(x) = \text{displaced shape function}$

$t = \text{domain thickness}$

$\rho_f = \text{fluid density}$

$SWL = \text{still water level}$

$L = \text{domain length}$

$V_f = \text{fluid volume} = (SWL * L * t)$

This is handled within Python relatively simply. The normalized shape of a deformed Euler-Bernoulli beam under fixed-end conditions (maximum magnitude 1) is stored in a vector with the same number of components as nodes within the finite element model (in this case, 150). The script shown in the Code Appendix under Listing A.9 is then utilized for analysis, where calls to 'FOAMySees' refer to FOAMySees class methods for applying loads to nodes within the model each iteration and performing a single analysis step.

OpenFOAM Analysis Details

The same OpenFOAM CFD model utilized in the previous subsection was utilized for this study, with a finer mesh resolution. The OpenFOAM CFD domain mesh x resolution was increased to 120 cells from 40 cells and the z mesh resolution was also increased accordingly.

Results

Loading Prediction Function

Application of the loading prediction function to an uncoupled finite element model generates identical displacements when compared to coupled FSI simulations for cases the volume produced within the region bounded by the deformed shape of the beam and the undeformed configuration of

the beam is a value which is less than the total fluid volume bearing on the beam. When the deflection of the beam generates a volume greater than that of the original volume of the fluid, errors in the loading approximation provided by the function with respect to the coupled FSI increase. This can be attributed to the fact that the loading approximation function does not account for these situations explicitly and over-predicts the loading magnitude, which results in a final value for displacement that is slightly more than that offered by the FOAMySees FSI simulation. This effect is evident in the case with the largest δ/L value, which shows slightly different results between the prediction function and the FSI simulation conducted within FOAMySees. If a correction term is added, which accounts for the volumetric calculation error in cases where the deflections are much larger than what would be required to lower the still water level below the undeformed configuration of the beam, these errors can be eliminated and the function could possibly be used in more circumstances, including when water levels are much lower than the test cases. Alternatively, in these circumstances the loading function shape and domain can be switched to a parabolic loading function spanning only across the wet surface of the beam. As these cases were not examined in detail in this study, development of a loading function condition or correction factor for this case is left to future work.

Axial Effects on Frame Rotational Stiffness

OpenSees-only simulations were run with uniform distributed loads due to the fluid pressure in its initial configuration to assess the affects of axial load (i.e. tension stiffening) in this example case. The effect of axial force upon rotational stiffening is evident in Fig. 5.15. All calculated deflection curves, most notably the curve with 'static' loading, tend to decrease in magnitude beyond a deflection to length ratio of 1:100 for this given case. For sections with different ratios of moment of inertia or modulus of elasticity to axial loading, this behavioral inflection point due to frame stiffening effects will also differ. The differential equation describing the deformation of the beam with shear effects is shown in Eqn. 5.2. As the modulus of elasticity increases, the effects of axial load on the overall behavior of deflection decreases, whereas with a reduced modulus of

elasticity, this effect increases. The term for shear deformation includes a shear correction factor (κ), which accounts for the shape of the section being changed through shear deformation. The shear deformation term is not accounted for in the element formulation utilized in the OpenSees analysis, resulting in differences between the results calculated by the material point method model and the coupled FEA-CFD model.

$$EI \frac{dw^4}{dx^4} - P \frac{dw^2}{dx^2} + \frac{EI}{\kappa AG} \frac{d^2 q(x)}{dx^2} = q(x) \quad (5.2)$$

Effect of Fluid Density and Elevation on Ratio of Simulated to Predicted Displacement

A fluid with a higher density and a lower still water level was utilized to both simulate a case within FOAMySees and compare with the quasi-static load shape calculation described previously in this section. This case is labeled as "OpenSeesPy, Shifting Load, H=0.1 m, $\rho_f = 3000 \text{ kg/m}^3$ " in Fig. 5.16. Since the still 'water' level is much closer to the level of deformation of the beam, the ponding effects are more significant at lower levels of deformation for an overall load that is 60 % of the overall magnitude of the case with a lower fluid density and higher still water level.

Comparison Between Parametric Studies With Coupled CFD-FEA Model and Material Point Method Model

Results are compared with a solution provided by Bonus et al. [2022], which utilized an in-house Material Point Method (MPM) software, ClaymoreUW, for simulation of both fluid and structure solutions. Slight variations between both solutions exist due to the presence of structural shear deformation effects within the MPM solution for the elastic beam. As the modulus of elasticity is decreased, the solutions from FOAMySees vary more and more from those provided by the ClaymoreUW program due to the presence of shear deformation in the MPM model resulting in a

higher deflection than the finite element beams, which do not account for the shear deformation. Fig. 5.16 is simply a normalized Fig. 5.15. Each curve in Fig. 5.15 was normalized by the 'OpenSeesPy, Uniform Load' results calculated in OpenSeesPy to isolate the effects of load redistribution due to fluid pooling upon the frame near the center of its span. This normalization resulted in slight distortion of the curve for the MPM solutions in Fig. 5.16, as the curves were normalized as a function of E and not the δ/L value. Since the deflections for the two lowest values of modulus of elasticity for the MPM model were larger than the results from the uniform distributed load by a factor of two, these last two points of the MPM curve are not truly normalizations, which results in points lying below the curve for FOAMySees rather than above it.

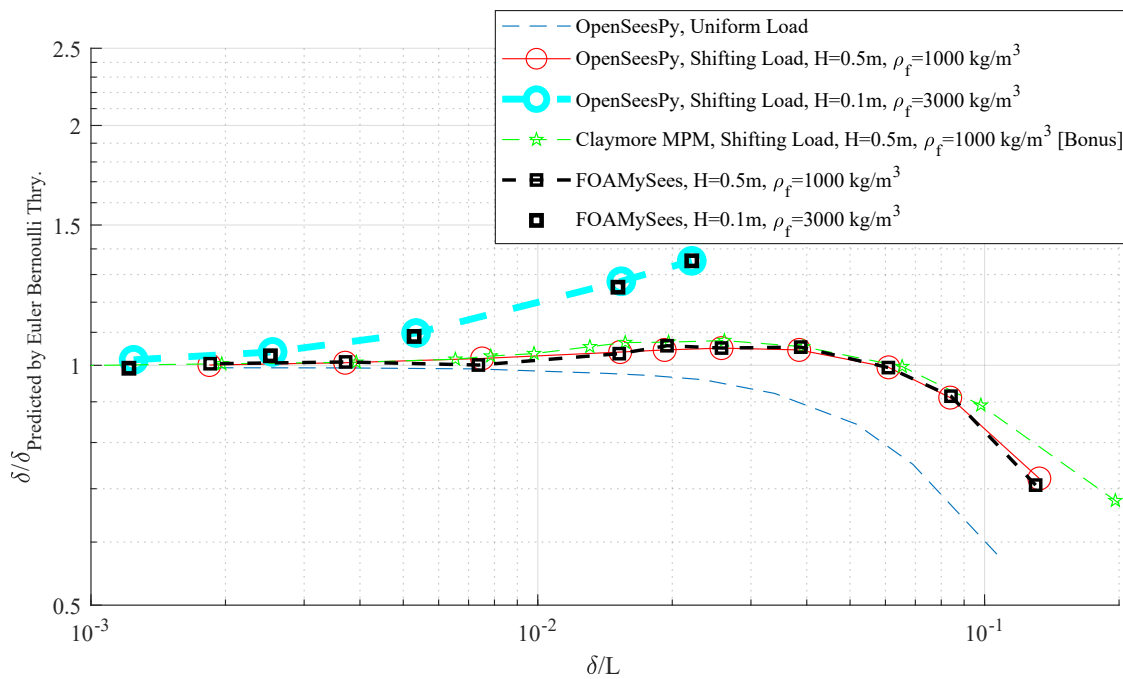


Fig. 5.15. FOAMySees Solution: center-span Displacement of Beams Under Shifting Water Loading

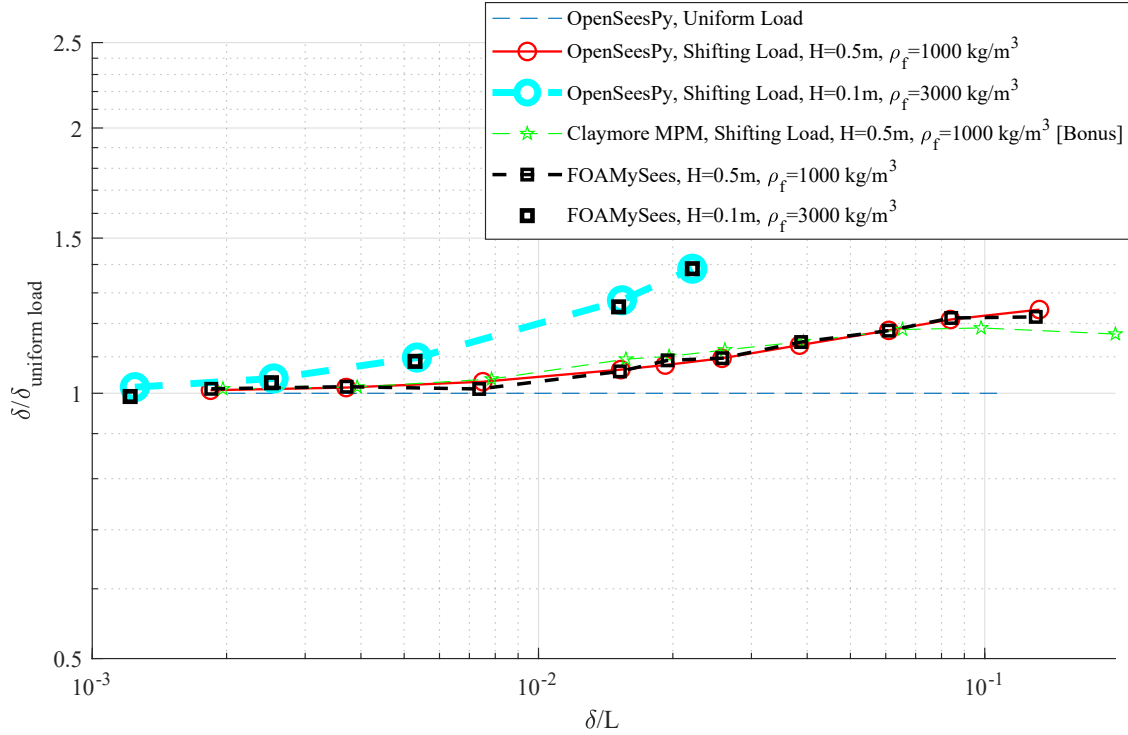


Fig. 5.16. Normalized by Uniform Load Deflection: FOAMySees Solution for center-span Displacement of Beams Under Shifting Water Loading

5.3 Laminar Multiphase Hydrodynamic Impact

An analytical benchmark test case historically solved with other numerical methods was investigated with the FOAMySees simulation code. Walhorn et al. [2005] is an FSI case comprised of a 2D multiphase fluid dam break within a tank with initial conditions as shown in Fig. 5.17. The dam is initialized and collapses upon itself, creating a bore, which impinges upon a flexible flap affixed to the floor of the tank. This case was utilized for an investigation into the coupling settings for the partitioned-coupled solution through the preCICE library (Subsection 5.3.2). Explicit coupling was first investigated for a number of timestep sizes, and an implicit coupling scheme with interface acceleration was then implemented. First, the implicitly-coupled solution was forced to diverge to study the effects of unconverged coupling timesteps and timestep size on the forces and displacements resolved by the diverging solution. The coupling settings were then changed to an

appropriate configuration for a converging solution, and timestepping of the converging solution and mesh motion options within OpenFOAM were investigated. Results from the simulation are then compared against simulations conducted in other simulation softwares for validation (Subsection 5.3.3).

5.3.1 Analysis Details

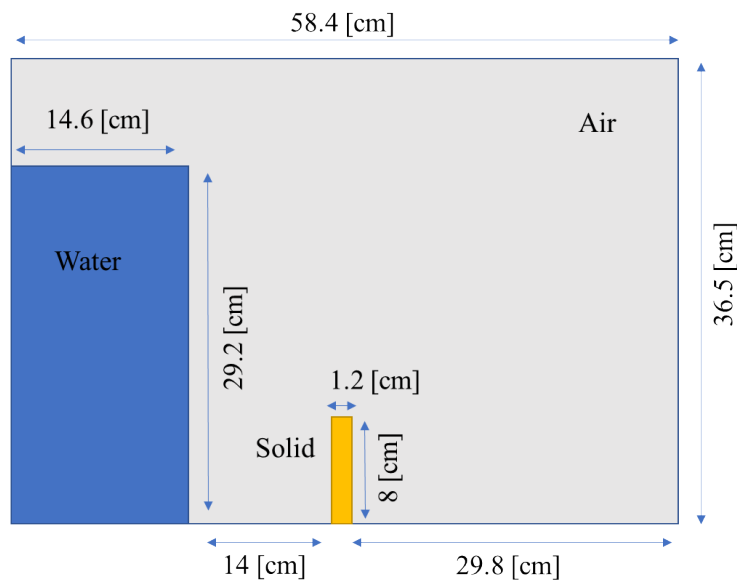


Fig. 5.17. Laminar Multiphase Hydrodynamic Impact Validation Case Setup

OpenSees Analysis Details

The structure is elastic, 1.2 cm wide by 8 cm tall, with a density of 2500 kg/m^3 , elastic modulus (E) of 1 MPa and Poisson's ratio (ν) of 0.3. The flap as modelled within a domain with a thickness of 0.01 meters has a total mass of 0.024 kg, a moment of inertia of $1.44\text{E-}9 \text{ m}^4$, and was modelled using 'elasticBeamColumn' type beam elements in OpenSees. The cantilevered flap was constrained at its base, and tip displacements and base reactions were recorded.

OpenFOAM Analysis Details

Standard values for density and viscosity were utilized for the air and water phases ($\rho_{water} = 1,000 \text{ kg/m}^3$, $\nu_{water} = 1.0 \times 10^{-6} \text{ m}^2/\text{s}$, $\rho_{air} = 1.22 \text{ kg/m}^3$, $\nu_{air} = 1.48 \times 10^{-5} \text{ m}^2/\text{s}$, where ρ = mass density of the fluid; and ν = kinematic viscosity). Boundary conditions for the model were as shown in Table 5.3. The case was modelled in two dimensions with a thickness of 0.01 m, with patches assigned 'empty' conditions in the unused direction. The blockMesh utilized for the OpenFOAM component of the coupled FSI analysis is available in the OpenFOAM Appendix under Listing A.4, and the initial state of the analysis is shown in Fig. 5.18 with the mesh resolution. A 'simpleGrading' scheme was utilized in the CFD mesh of the analysis to increase the mesh resolution near the tip of the flap (Guide).

Table 5.3. Closed Domain - Model Boundary Conditions for All Field Variables

Field	Atmosphere	Walls	Flap
U	pressureInletOutletVelocity	noSlip	movingWallVelocity
p_{rgh}	totalPressure	fixedFluxPressure	fixedFluxPressure
pointDisplacement	fixedNormalSlip/fixedValue	fixedNormalSlip/fixedValue	fixedValue
α	inletOutlet	zeroGradient	zeroGradient

5.3.2 Coupling Settings Study

In this subsection, the coupling settings for the analysis were investigated as part of a study to demonstrate the requirements for strongly-coupled FSI with the presented coupling methodology. The use of an partitioned-explicit-coupling algorithm was investigated, and the effects of timestep on the stability of explicitly-coupled simulations is discussed. An implicit coupling algorithm was then implemented, and a timestep and coupling convergence tolerance study was conducted with the implicit coupling algorithm to investigate the effect of coupling settings on the resolved forces and displacements of the coupled model.

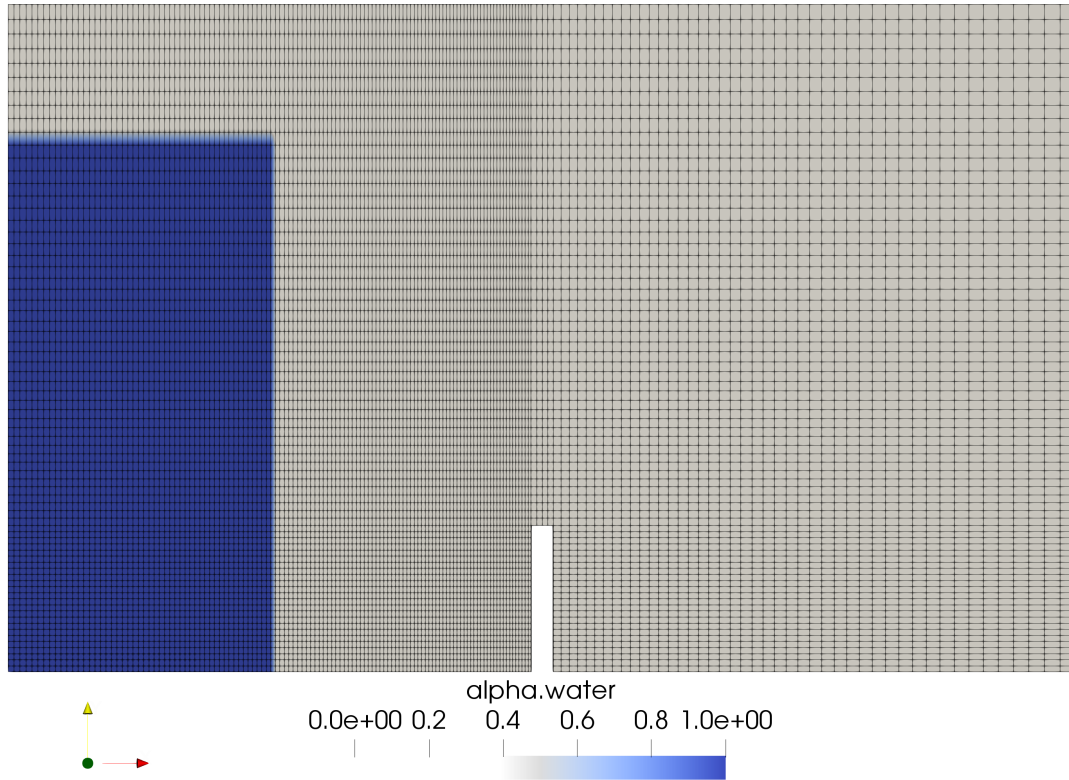


Fig. 5.18. OpenFOAM Mesh and Initial State

Instability of Explicitly-Coupled Analyses with Strong FSIs

The use of an explicit coupling scheme was investigated for this case. Results for force and displacement are provided from a study of an explicit simulation with repeatedly reduced timesteps for demonstration of the inherent instability of the partitioned explicit solution with a large number of interacting frequency modes. The displacements and forces for the diverging explicit solutions are provided in Fig. 5.19. The largest timestep investigated ($5e - 4s$) allows the simulation to progress well into the strong FSI phase, but quickly leads to divergence of the solution during the second point of high acceleration. As the timestep of the solution is decreased, more modes of deformation of the structure are captured and become unstable in the strong interactions between the motion of the structure and the motion of the breaking dam. Reduction of the coupling timestep in the explicit

simulation results in instabilities at earlier times in the simulation. The lowest timestep investigated ($1e-5s$) results in the simulation crashing at the instance of the breaking dam's impact with the flap ($0.13s$). Force instabilities are present in this simulation from the before the dam impact phase due to unstable frequency modes. This indicates the need for the use of an implicit coupling scheme for the present case.

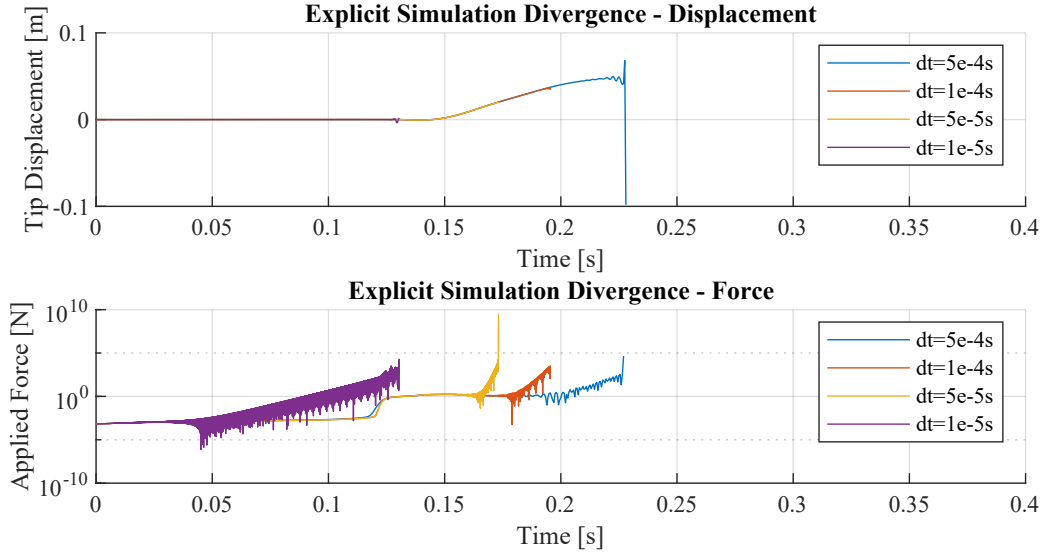


Fig. 5.19. Explicit Simulation Divergence - Strong FSI in Partitioned Simulations Require Implicit Coupling Schemes for Stability

Implicit Coupling Settings Investigation

Unconverged Coupling Iterations and Spikes in Force

The coupling tolerance was set to a low value and the maximum number of coupling iterations was set to a value of 10 to try to force divergence of the coupling solution to investigate the effect of decreasing the coupling time increment size on the stability of an already unstable coupled solution. An implicit coupling scheme with IQN-ILS interface acceleration and an initial relaxation factor of 0.67 was utilized for this study of implicit coupling behavior, with a coupling data iteration convergence tolerance of $1e-5$. In this subsection, effects of timestep reduction on the instability

of the model and the displacements and forces resolved by the diverging coupled solution are investigated and discussed. The coupling timestep for the analysis was varied with an initial value of $5e-4$ s which was then reduced to $1e-4$ s, and then reduced again to a minimum of $5e-5$ s. The force time histories, tip displacements, and coupling convergence information for the simulations run with timesteps of $5e-4$ s and $1e-4$ s are provided in Fig. 5.20. The simulation with a timestep of $5e-4$ s first fails to converge around the inflection point of the displacement curve, just after peak displacement is achieved (0.16 s). From this point onward, small fluctuations in force can be seen as the solution diverges. With a reduced timestep of $5e-5$ s, the model diverges much sooner - at around 0.13 s, or at the initial point of impact with the bore of the dam break. Force instabilities calculated within OpenFOAM due to the divergence of the coupling iterations can be seen for the three cases studied in Fig. 5.21.

Due to the insufficient number of coupling iterations provided in the coupling scheme, the coupling algorithm failed to stabilize the solution across all frequency modes of the interaction of the structure with the fluid. When a coupling timestep fails to converge, the simulation experiences errors in the applied displacement and calculated force fields, which result in spikes in the resolved forces and reactions in the next coupling timestep.

This can be precluded by allowing for enough coupling iterations within the coupling scheme to ensure that all unstable frequency modes have been accelerated to convergence by the implicit coupling scheme. With larger timestep sizes, fewer frequency interactions in the solution are resolved, resulting in an overall lower number of unstable frequency modes. As the timestep is reduced, more frequency modes are captured and may become unstable in the strong FSI, and thus more coupling iterations may need to be performed to ensure convergence of the coupling timestep.

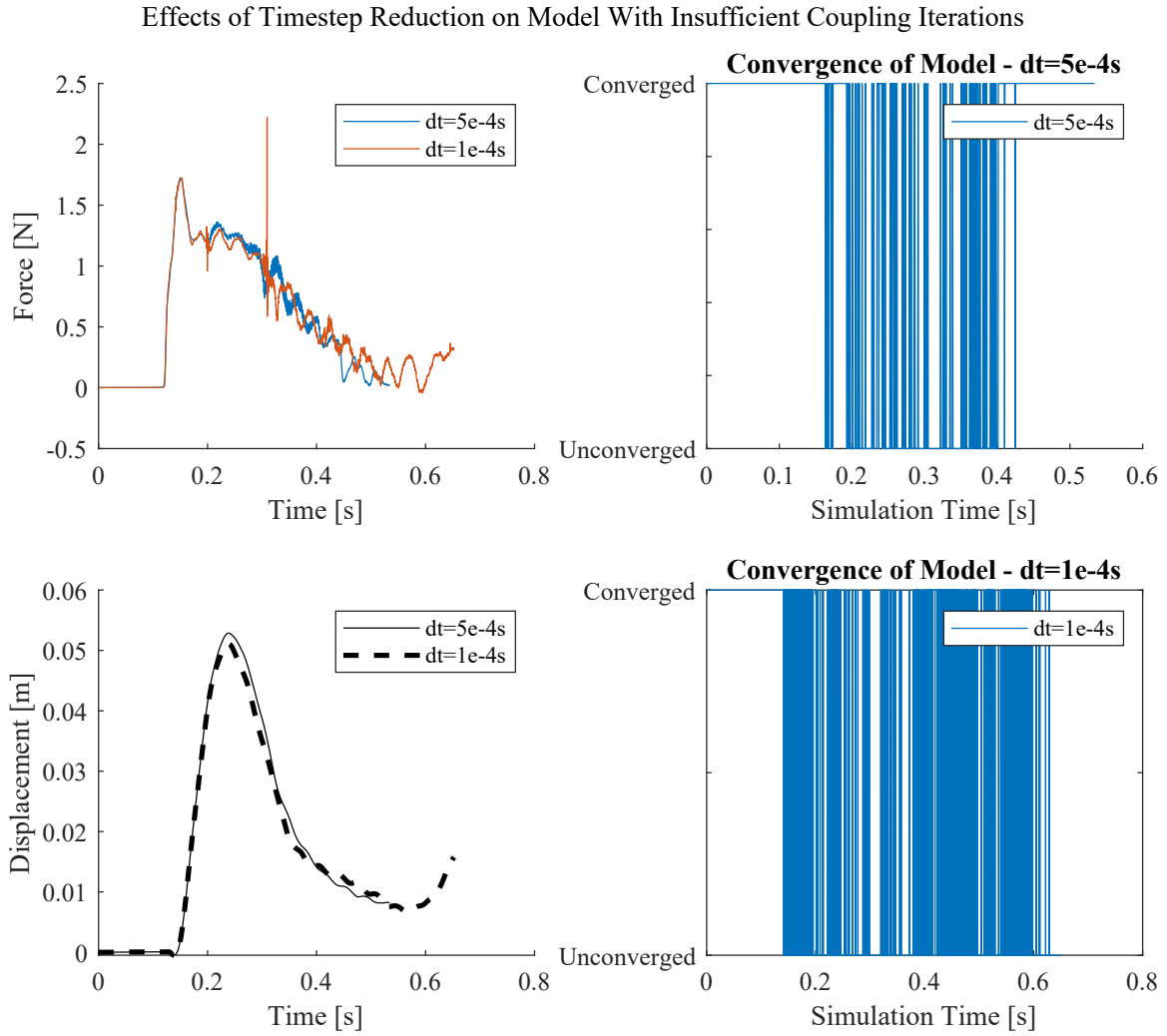


Fig. 5.20. Effect of Timestep Reduction on Unconverging Model - With Too Few Coupling Iterations Provided, Simulations Experience High Fluctuations in Force

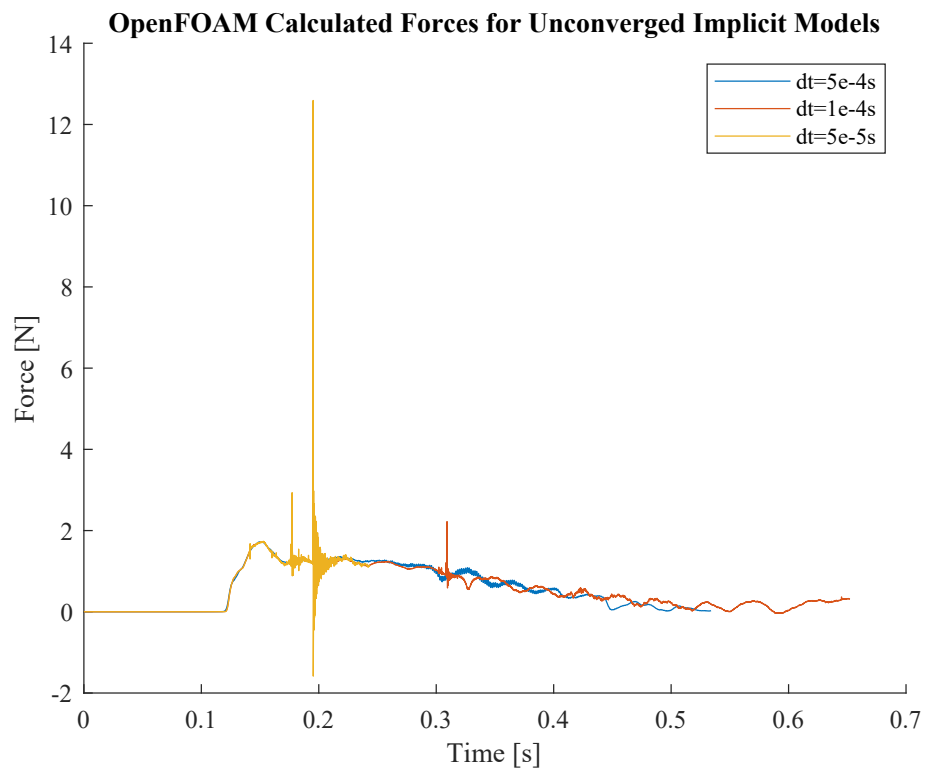


Fig. 5.21. Simulation Instabilities Increase with Decreased Timestep for Models with Poorly Selected Coupling Settings

Converged Model Timestep Investigation

Maximum coupling iteration counts for the models investigated in this section were increased from 10 to a value of 100 iterations, eliminating the model instabilities through additional coupling iterations. The converged solutions for displacement and force were investigated with respect to an increasing coupling timestep size and are plotted in Figs. 5.22 and 5.23, respectively. While CFD applied loads and FEA model reactions do not vary in magnitude much between the analyses, the deformation of the finite element model is under-approximated with larger timestep sizes. As seen in Fig. 5.23, the applied forces are fully transferred to the finite element model regardless of timestep size, and can be measured as a reaction force of the same magnitude as the applied load for all timestep sizes. This indicates that the selection of too large of a coupling timestep with an implicit coupling scheme can result in the loss of energy in the simulation.

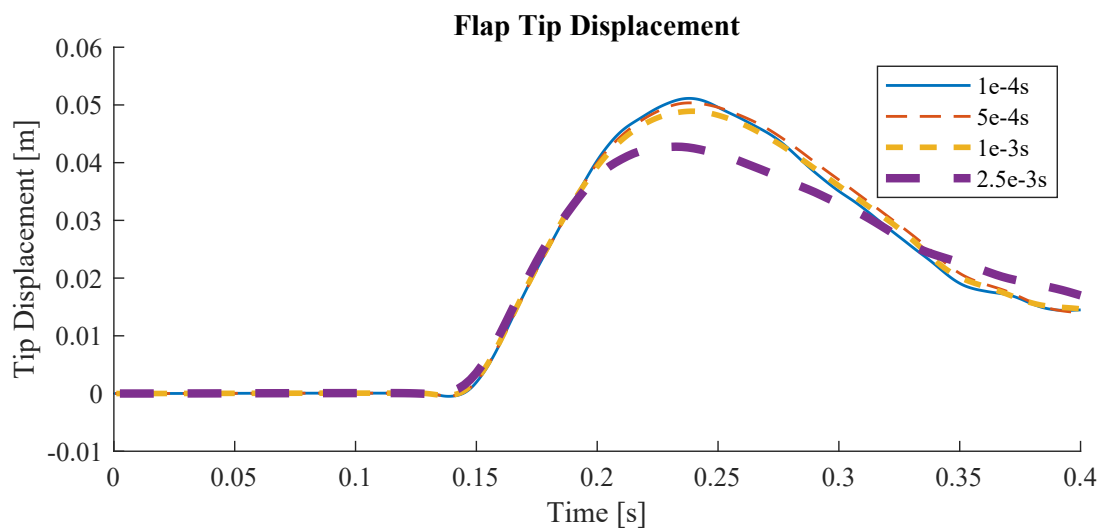


Fig. 5.22. Variation in Coupled Solution Displacements With Implicit Coupling Timestep Size

Implicit Simulation Timestep Convergence Study

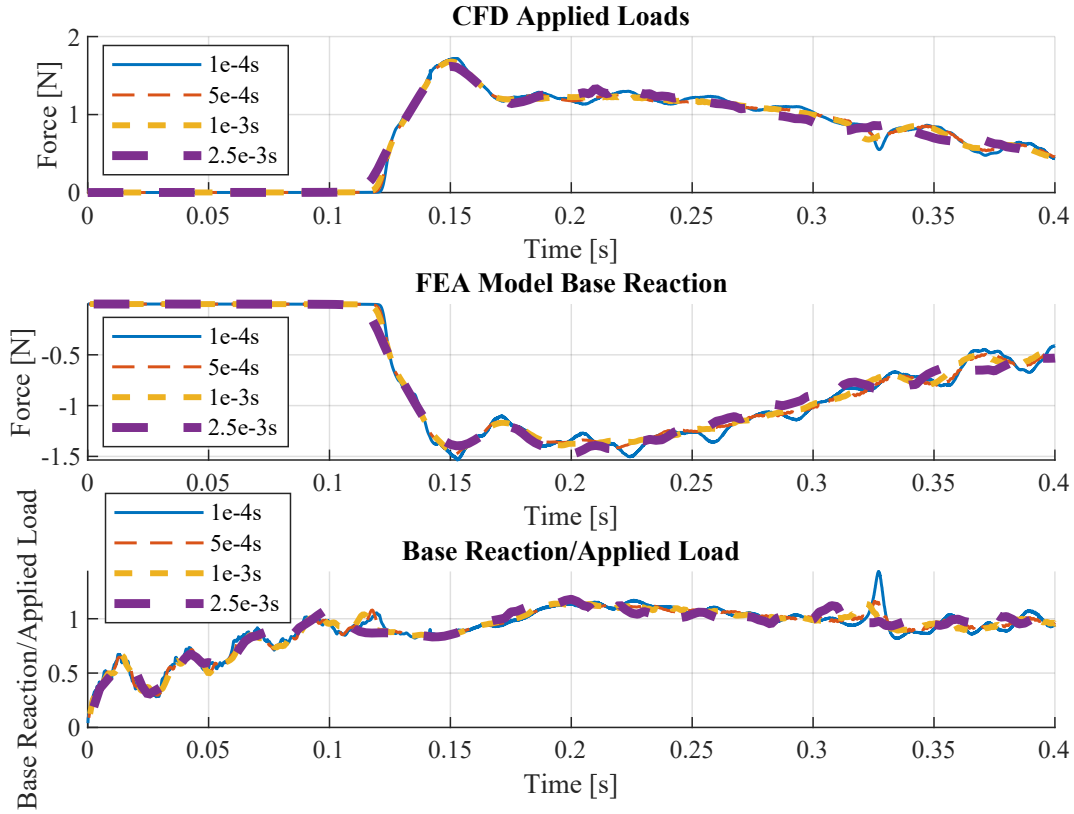


Fig. 5.23. Variation in Coupled Solution Forces With Implicit Coupling Timestep Size

5.3.3 Results Comparison

Based on the convergence study, a time increment of 1e-4 s was selected for the coupling time progression to fully capture the deformation of the flap during the dam break case. A coupling convergence tolerance of 1e-4 was utilized to ensure convergence of the strongly coupled solution within timestep iterations, with a maximum number of coupling iterations of 100. The flap tip displacements from the case were plotted against results from other simulations conducted in different softwares for cross-correlation. Results from FOAMySees show reasonable correlation between the coupled FEM - CFD solution and historical results for the same geometry and initial conditions modelled with finite elements entirely, using the particle finite element method, the

material point method, and with another coupled CFD-FEA model simulated with the Simcenter STAR-CCM+ commercial software. Walhorn originally solved the problem monolithically with finite element solutions for both the water and the structure. Idelson and Marti utilized particle finite element methods (PFEM) for the fluid solution coupled with finite elements for the structure. Bonus utilized Material Point Method (MPM) for simulation of both fluid and structure components of the solution. The case was modelled in STAR-CCM+ by Dr. Andrew Winter using 8-node bricks with nonlinear geometry model enabled to compare results with commercial software. STAR-CCM+ utilized FVM for solution of the fluid motion and FEM for motion of the flexible flap. The lateral displacement time history for the top left corner of the flap is shown in Fig. 5.24. Results from Marti et al. [2006], Idelsohn et al. [2008], STAR-CCM+, and Bonus et al. [2022] are shown as well, demonstrating large variability in the post-bore-impact phase from the dam break amongst various codes. Free surface comparisons to results obtained from Walhorn et. al and from the STAR-CCM+ simulations are shown in Fig. 5.25. Snapshots from various points along the solution progression show good agreement with the original solution provided by Walhorn and with the results from STAR-CCM+. A solution with a lower beam mesh resolution in FOAMySees is shown to demonstrate the effect of constrained nodes in the coupled solution on the 'impact' time of the bore and the tip displacement of the flap. The finite volume domain along the flap contained 24 cell face center nodes. Initially, a finite element mesh with 20 elements was chosen, with constrained nodes included in the coupling routine. As the cell face center nodes nearest the flume floor in the model with the lower resolution finite element mesh were coupled to a constrained finite element node, the forces from those cells were not transferred to the free nodes of the finite element model and thus were not accounted for. When the bore reaches the cells which are coupled to unconstrained nodes, the force is transferred to the unconstrained nodes and motion of the structure begins. This demonstrates the need for removal of constrained nodes from the coupling routines or the use of fine meshes near constrained finite element nodes to preclude transfer of coupling forces to constrained nodes.

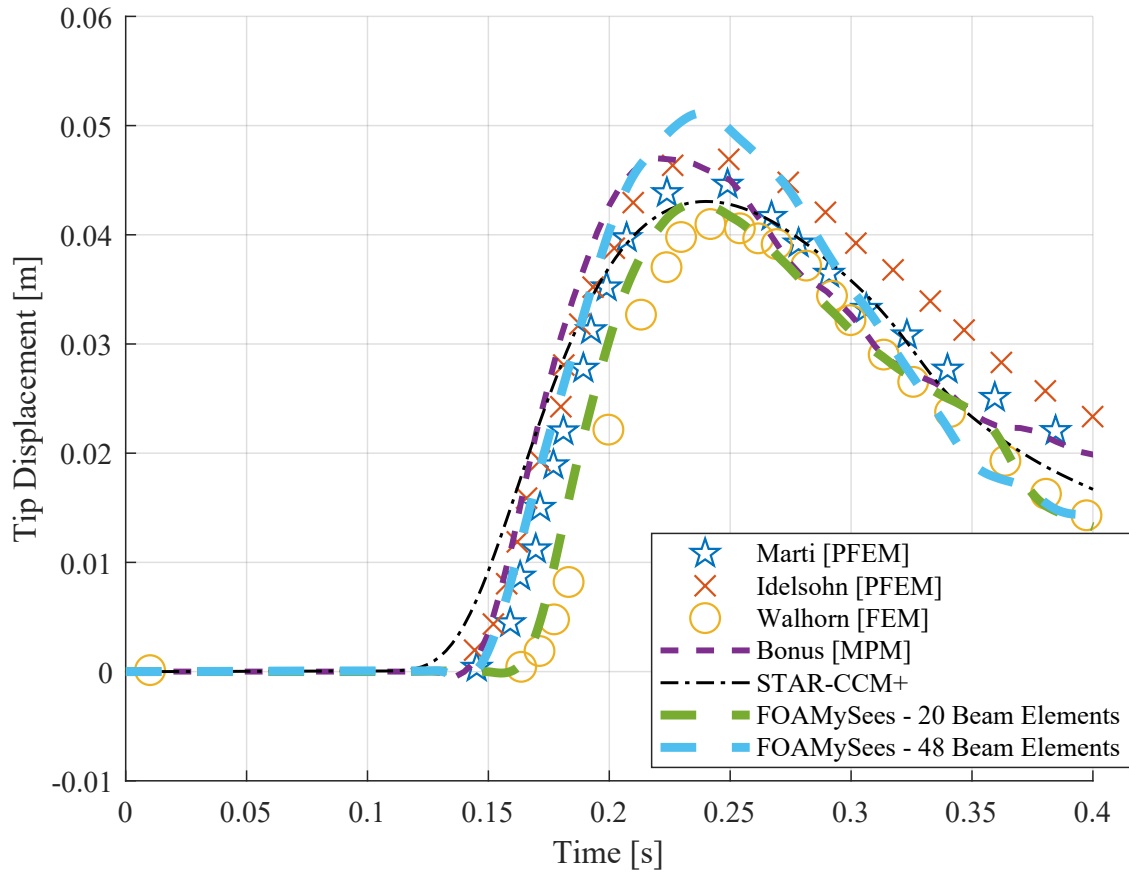


Fig. 5.24. Walhorn et. al Validation Case: FOAMySees Comparison to Other Solutions

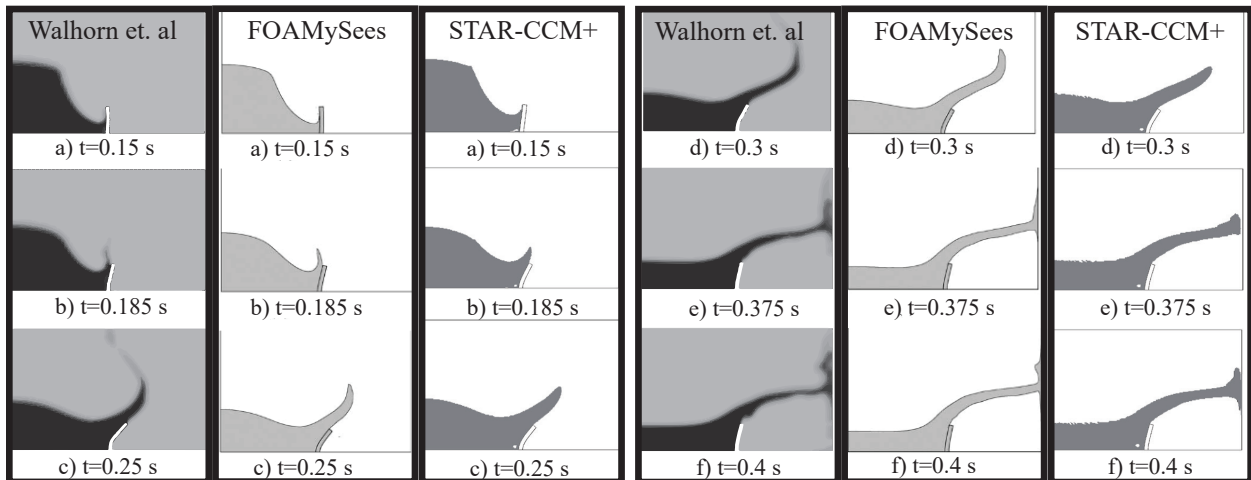


Fig. 5.25. Walhorn et. al Fluid Free Surface Comparison: Left, grey background - FOAMySees. Right, white background - Walhorn et. al

Discussion

The large deflections of the flap during this case induce a large mesh distortion in the CFD model. Mesh distortion can result in changes in numerical derivatives between the points of the CFD domain grid which must be corrected by incorporating nonOrthogonal correctors in the OpenFOAM solution. Without correctors, mesh distortion causes the pressure and velocity fields to not be properly resolved and this can result in instabilities and subsequently divergence of the solution. In Fig. 5.26, Fig. 5.27, and Fig. 5.28, volume phase-fraction, pressure, and velocity during a point of large mesh deformation were plotted along with mesh distortion to show the effect of non-orthogonality on the resolved pressures and velocities.

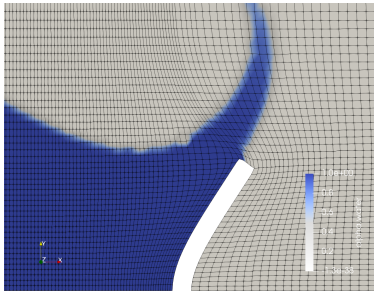


Fig. 5.26. CFD Model Mesh Distortion - Phase Fraction

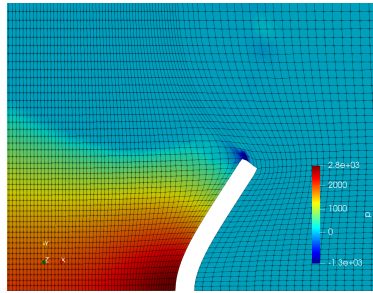


Fig. 5.27. CFD Model Mesh Distortion - Pressure

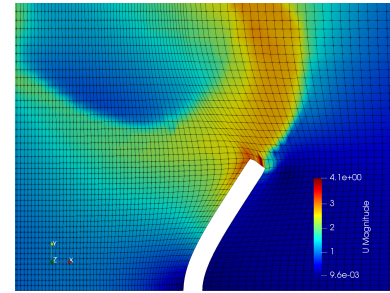


Fig. 5.28. CFD Model Mesh Distortion - Velocity

CFD Mesh Motion Scheme Effects on Resolved Forces and Displacements

Using an implicit model with a timestep of $5e-4$ s and a coupling convergence tolerance of $1e-4$, a study was conducted on the displacementLaplacian mesh motion solver options within OpenFOAM. Displacements and forces resolved by the coupled solution for different mesh motion schemes are plotted in Figs. 5.29 and 5.30. While applied forces do not change much between different pointDisplacement schemes, the flap tip displacement is heavily damped in the case using a 'uniform' laplacian of the interface displacement across the CFD domain. This indicates that usage of a uniform laplacian scheme for diffusion of the moving boundary displacement results in erroneous

diffusivity in the CFD model due to the motion of the surface of the coupled interface. Use of 'inverseDistance' or 'quadraticInverseDistance' displacementLaplacian schemes in the OpenFOAM dynamicMesh dictionary is thus recommended for coupled FSI in simulations with significant mesh deformation.

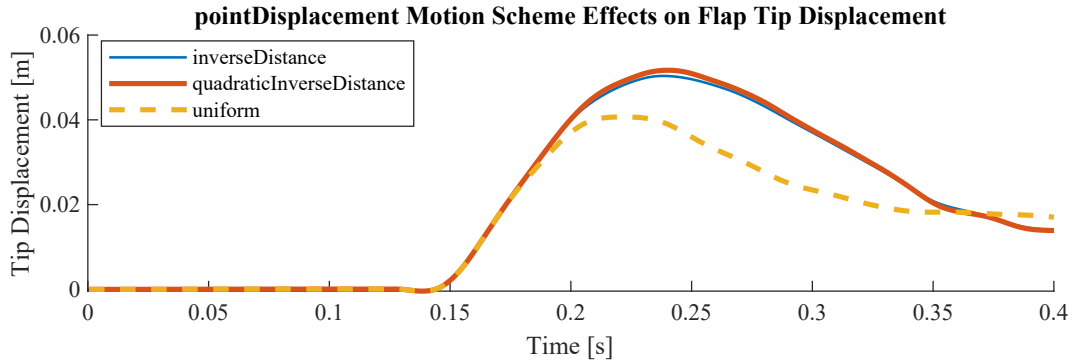


Fig. 5.29. Displacement Laplacian Study - Flap Tip Displacement

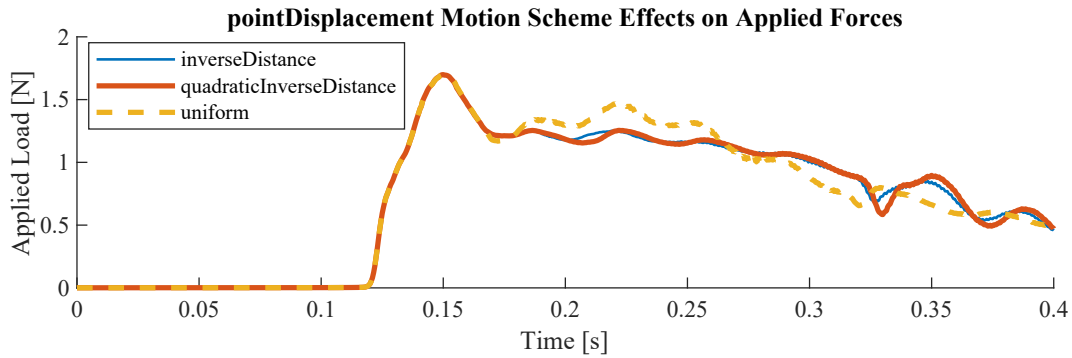


Fig. 5.30. Displacement Laplacian Study - Applied Forces

5.4 Test Cases With Sequential or Concurrent Ground Motions

The FOAMySees model of the Walhorn case from this chapter was modified to investigate the transfer of initial structural state to an FSI analysis. This study was conducted to demonstrate the capability for FSI with nonlinear materials as well as the possibility for sequential hazard analysis within the presented coupling methodology.

5.4.1 Coupled Analysis Settings

As with the previous analyses of the Walhorn case, an implicit coupling methodology (namely IQN-ILS interface acceleration) was utilized for this case. A coupling convergence tolerance of $1e-4$ was used with a coupling timestep of $5e-4$ s. The timestep size for the initial structural analysis routines was 0.001 s.

OpenFOAM Analysis Details

Identical domain geometry, fluid properties, and fluid initial conditions as described in the Laminar Multiphase Hydrodynamic Impact section (Sec. 5.3) were utilized to demonstrate the multi-hazard analysis capabilities of FOAMySees. The blockMesh utilized for the OpenFOAM component of the coupled FSI analysis is available in the OpenFOAM Appendix under Listing A.4, and the initial state of the analysis is shown in Fig. 5.18 along with the mesh.

OpenSees Analysis Details

The FEA domain was discretized into 48 beam elements with a fiber section to represent the flap, using 'dispBeamColumnElements' and 6 fibers through the thickness of the section. The nonlinear flap material chosen was the OpenSeesPy Uniaxial Hardening material, with an initial elastic modulus, E , equal to 1 MPa. Two materials were studied, both with arbitrary hardening parameters $H_{iso} = 0.15$ and $H_{kin} = 0.25$, one with an arbitrary yield stress (1 Mpa), followed by one with a reduced arbitrary yield stress (125 Kpa). Legendre beam integration with 2 Gauss points along the length of the beam were used. The OpenSees script utilized for this analysis can be seen in the OpenSees Appendix under Listing A.7. In cases with a preliminary excitation, ground motion was stopped after 4 s and free vibration of the flap was allowed until 10 s. A plot of the horizontal accelerations and acceleration magnitudes for both ground motions used over time are shown in Fig. 5.31. Flap tip responses from all cases subject to preliminary ground motions are shown in Fig.

5.32.

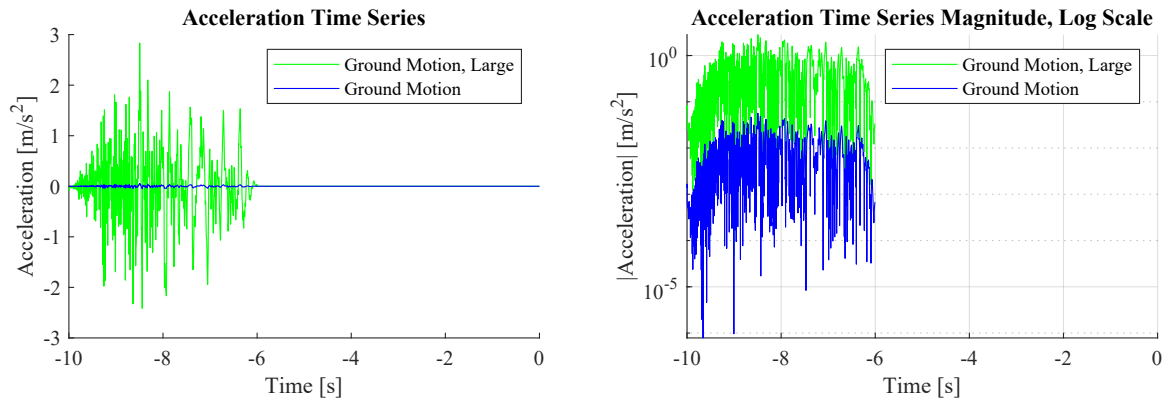


Fig. 5.31. Ground Motions Utilized in Preliminary Seismic Analysis

1997

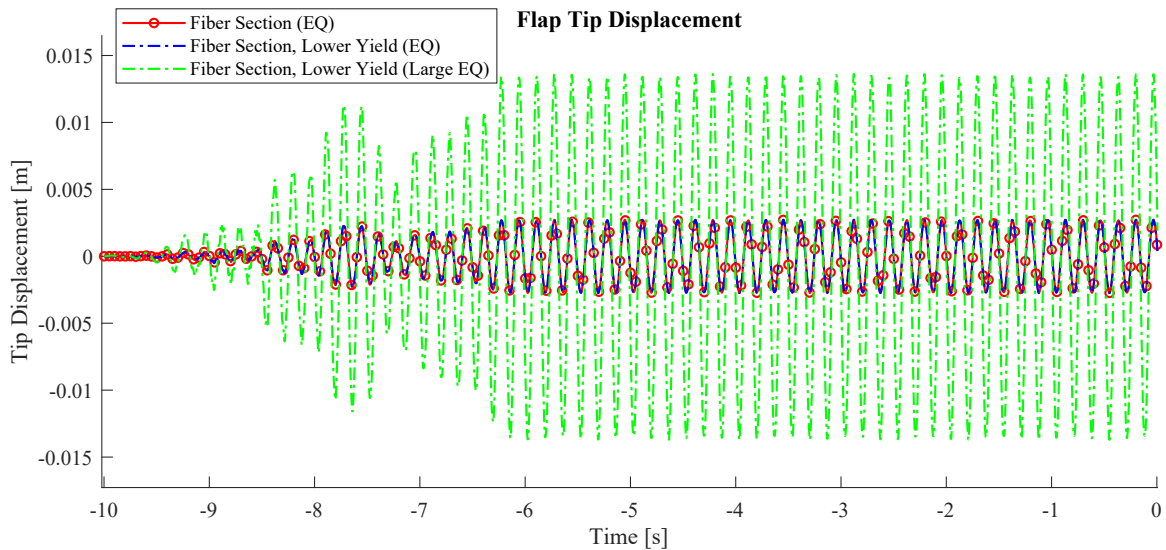


Fig. 5.32. Preliminary Analysis Flap Tip Displacement for Cases with Base Ground Motion Excitation Prior to Fluid-Structure Analysis

5.4.2 Results

The simulation was run a total of six times in this study. The analysis was run initially with the same elastic material properties as the previous section using a 'dispBeamColumn' element formulation as a control. This is shown in Fig. 5.33 as the curve 'Elastic Section'. The flap section was then

changed to a fiber section with 6 integration points through the thickness of the section utilizing the nonlinear uniaxial hardening material. The yield stress was first set to a high value (1 MPa) to prevent yielding as a check on the material model against the solution with an elastic material. The model with the hardening material was run without a preliminary earthquake analysis to establish a baseline displacement time history with a nonlinear material with which to compare, then was run again with a low magnitude preliminary seismic analysis. Results for these simulations are shown as the curves named 'Fiber Section' and 'Fiber Section (EQ)', in Fig. 5.33. The hardening material yield stress was then reduced from 1 MPa to a value of 125 KPa. The case was run without a preliminary analysis, showing material yielding during the bore impact phase of the FSI analysis ('Fiber Section, Lower Yield'). The case was then run twice more with the yielding material and preliminary seismic excitations of two different magnitudes ('Fiber Section, Lower Yield (EQ)', 'Fiber Section, Lower Yield (Large EQ)'). All cases are plotted alongside the original

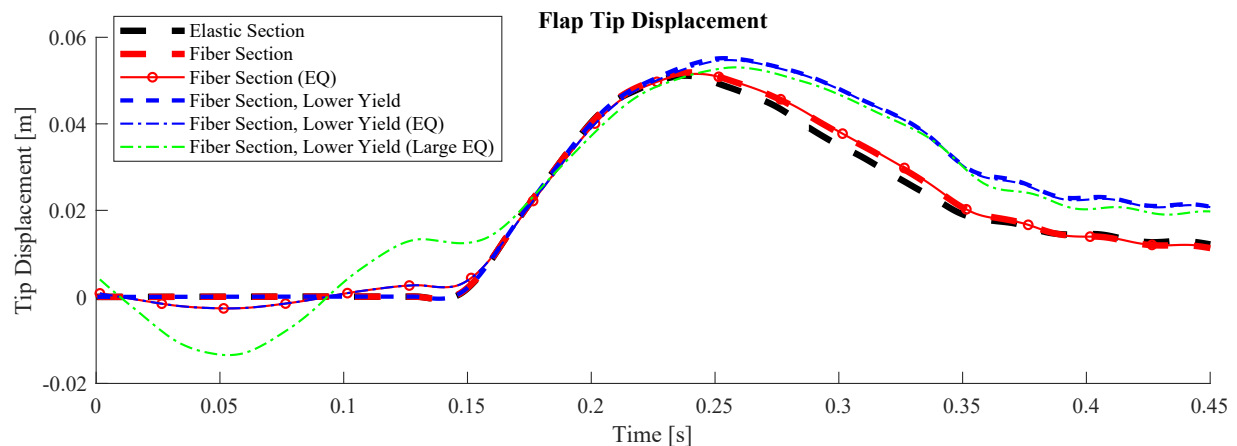


Fig. 5.33. Comparison Between FOAMySees Laminar Hydrodynamic Impact Solutions - Demonstration of Effects of Material Yielding and Initial Kinematic Conditions Due to Preliminary Seismic Loading Upon Flap Tip Displacement

FOAMySees solution for the Walhorn validation case with elastic beam elements in Fig. 5.33. Comparing between curves 'Fiber Section' and 'Fiber Section, Lower Yield', it is evident that material nonlinearity plays a role in the flap tip displacement response to impact from the breaking dam. The state from the preliminary analysis is carried over to the subsequent FSI, as shown in

the curve named 'Fiber Section, Lower Yield (Large EQ)' in Fig. 5.33. This curve shows a slight reduction in the tip displacement and velocity with respect to the same case with a smaller ground motion, possibly due to initial kinematic conditions.

Concurrent Seismic and FSI

The FOAMySees main solution routines allow for the implementation of time-based loading functions and ground motion application during implicitly-coupled analyses. The case with the highest yield stress from this section was tested with a concurrent triangular periodic ground motion displacement time history applied to the base node of the flap. The applied motion had a magnitude of $5\text{e-}4$ meters and began at a time of 0.01 s, stopping at an analysis time of 0.25 s, with a repeat period of 0.01 s. The flap tip displacement time history from this case is shown in Fig. 5.35 along with that of the same case without a ground motion.

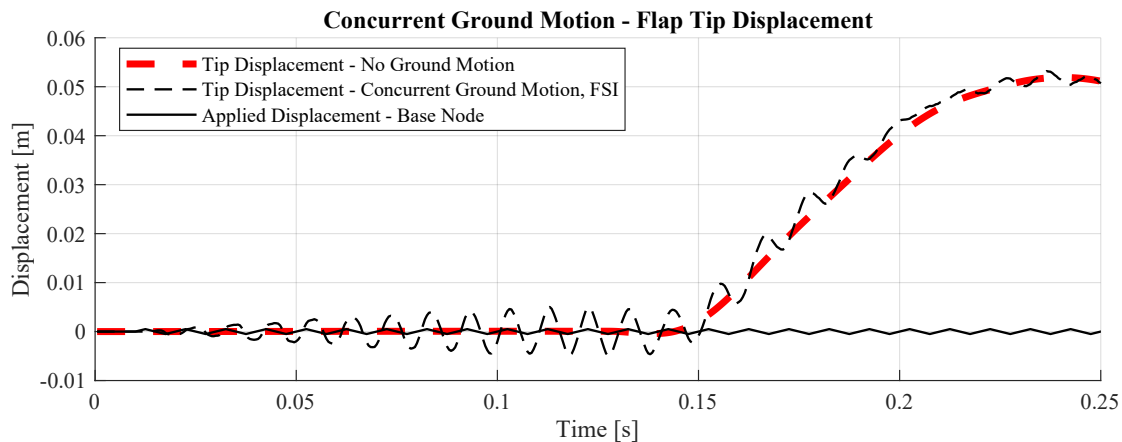


Fig. 5.34. Comparison Between FOAMySees Laminar Hydrodynamic Impact Solutions - Demonstration of Concurrent Seismic and FSI

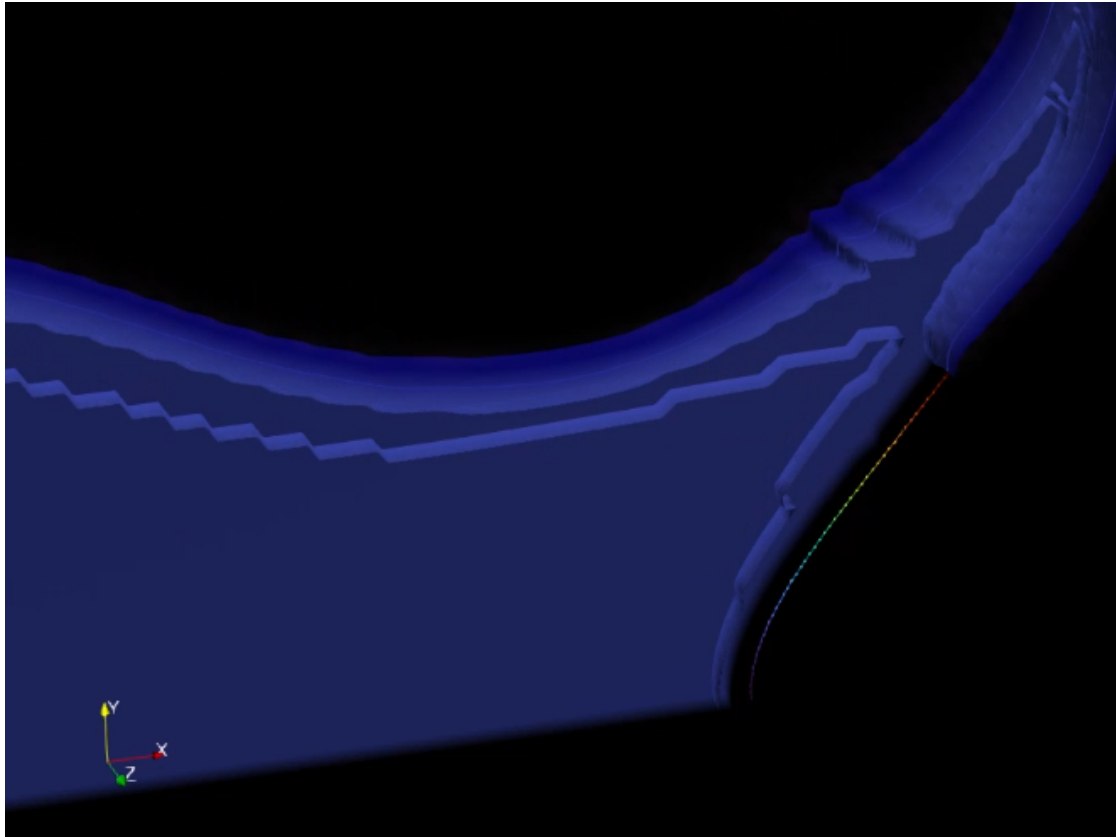


Fig. 5.35. Snapshot of FOAMySees Concurrent FSI Ground Motion Simulation

Chapter 6

Experimental Test Cases

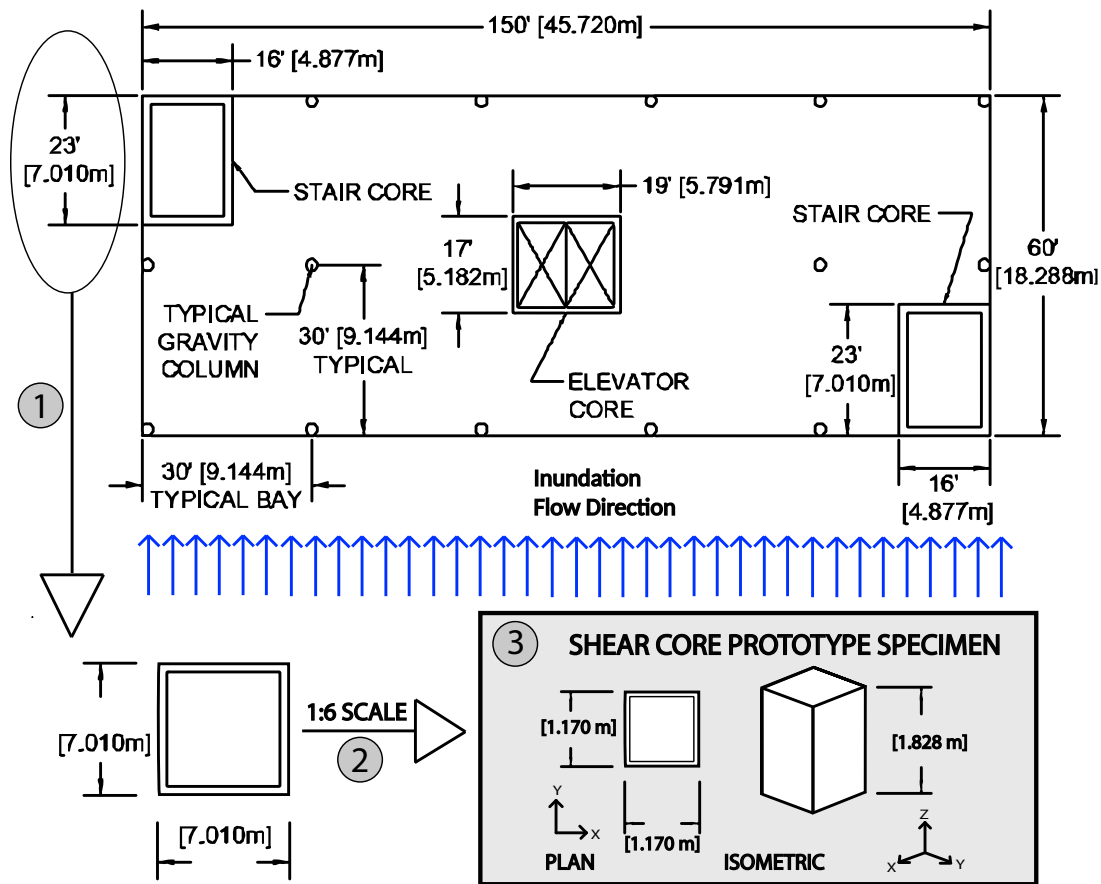
To validate FOAMySees calculated forces with experimental data, an experimental campaign of two specimens tested at the Oregon State Univeristy O.H Hinsdale Wave Research Laboratory Large Wave Flume was modelled. Experimental work was conducted by a collaborative team from the University of Washington and Oregon State University led by master's students Christopher Pyke and Kenneth Sullivan. CFD simulations and computational analysis were completed with the assistance of Chris Pyke, Kenneth Sullivan, Dakota Mascarenas, Andrew O. Winter, Ph.D, Dawn E. Lehman, Ph.D, Michael R. Motley, Ph.D., Pedro Arduino, Ph.D, and Charles W. Roeder, Ph.D. The faculty and staff at O.H. Hinsdale Wave Research Laboratory also contributed to this project and assisted with experimentation and data curation. As the experimental setup and results have been published previously in Lewis et al. [2022] and are available in Pyke [2020] and Sullivan [2021], information pertaining to motivation, background, and irrelevant testing parameters are omitted here for brevity and clarity. Results from the strongly-coupled FSI analyses were compared to experimental results and analogous CFD analyses (identical mesh) with a rigid boundary representing each structure.

6.1 FSI Simulation of Concrete Core Wall Model at 1:6 Scale

6.1.1 Experimental Investigation

Using ASCE 7-16 to determine both site-specific earthquake and tsunami demands, Pyke [2020] developed a representative vertical evacuation structure containing concrete shear core walls was designed to remain linearly elastic under design loads. All calculations for the demands and the design of the representative vertical evacuation structure can be found in Chapter 3 and Appendix A of Pyke [2020]. From this representative VES, a prototype concrete core shear wall test specimen was developed by Pyke by scaling down the design of the full-scale lateral force resisting system (LFRS) from the representative vertical evacuation structure to meet the dimensional constraints of the experimental facility. More specifically, the geometry of the specimen was based off taking the longest wall of the LFRS shown in Fig. 6.1 and making a 7 m x 7 m square box. At the scale possible within the flume (1:6 scale) this translated to a test specimen that was 1.168 m by 1.168 m with 0.127 m thick walls, with a height of 1.830 m and a 0.165 m thick slab at the base of the walls. This rectangular concrete core specimen sat on four piles which were embedded into the corners of the walls and the base of a box beneath which contained soil. This box was filled to various levels with soil during the tests to examine the effects on structural response, but as the cases which are simulated possessed no soil in this box, further information about this study parameter is omitted in this thesis. This soil box was covered by impermeable floor panels, which spanned to the edge of the soil box from the face of the specimen. The soil box was attached to a series of pin-pin connected load cells, which themselves were attached to the flume walls and floor via a reaction frame. The specimen design, construction drawings, and more information about the development of experimental procedures utilized are described in detail in Appendix B of Pyke [2020].

FULL SCALE REPRESENTATIVE VERTICAL EVACUATION STRUCTURE (VES)



(1) The longest wall of the shear core system was taken and made into a square. (2) This square was scaled down to 1:6 of original size. (3) The prism was shortened to 1.83 meters in height to fit within the test facility.

Fig. 6.1. Floor Plan of Representative Full Scale Vertical Evacuation Structure: From this design, the prototype specimen which was utilized in testing and numerical analysis was developed. Figure modified from Pyke [2020].

Test Facility and Instrumentation

The dimensions of the specimen and test set up including instrumentation can be seen in Fig. 6.2, taken from Pyke [2020]. This 1:6 scale VES prototype was tested in the Large Wave Flume (LWF) at the Hinsdale Wave Research Lab (HWRL) at Oregon State University (OSU). The LWF is 104.0 m long, 3.7 m wide, and 4.6 m deep. Waves are generated by a piston-type wavemaker on the upstream end of the flume. The wavemaker at HWRL is capable of producing any user-defined wave capable

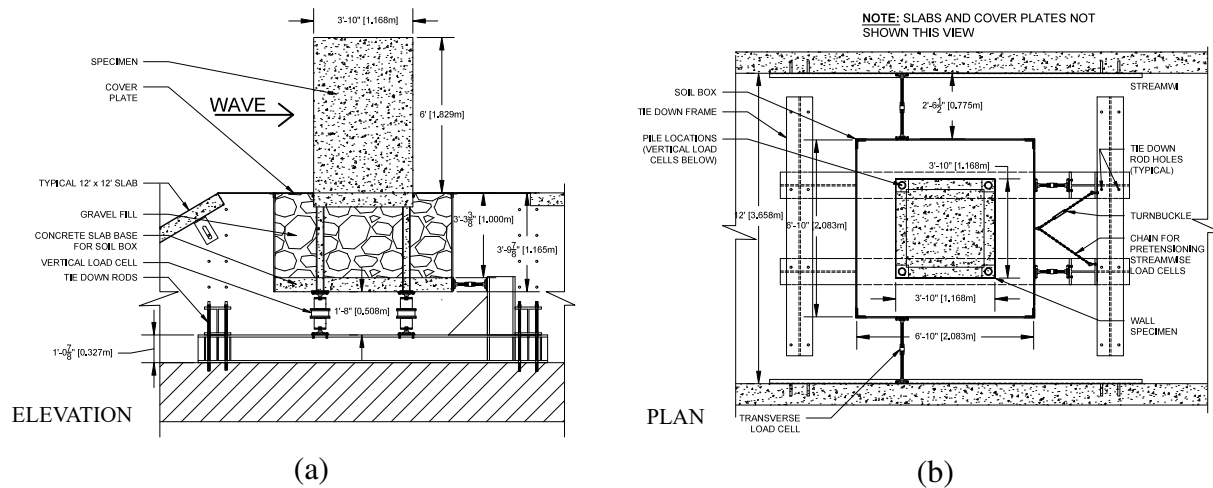


Fig. 6.2. Specimen Dimensions (Pyke [2020])

of being generated within the physical constraints of the motion of the pistons with a provided still water level. Its maximum stroke is 4.0 m with a maximum velocity of 4.0 m/s. The maximum still water level (SWL) in the flume that a wave can be generated in is 2.0 m. The maximum stable wave height the facility is capable of creating given these constraints is a 1.40 m solitary wave. Unstable waves can be produced by generating waves of taller height than this, which will then break and propagate as a turbulent bore. To provide an adjustable flume bathymetry, a set of 0.15 m thick, 3.66 m square, movable concrete slabs are used as the flume bed. The flume bathymetry used in this study consisted of a submerged horizontal bottom section starting at the wavemaker, which was followed by sloped beach and dry horizontal sections formed by the movable slabs.

Vertical forces, transverse forces (forces perpendicular to the length of the flume), and stream-wise forces (forces acting parallel to the length of the flume) were measured during the wave impact. A variety of instruments including load cells, pressure sensors, and strain gauges facilitated redundant measurements and validation for base shear and overturning moment. Exact geometric locations of wave gauges, ADVs, and pressure sensors utilized in the experimental-numerical correlation study are listed in Table 6.1. Wire resistance wave gauges were used to record free-surface elevations from these tests and fluid velocities were measured using Acoustic Doppler Velocimeters (ADV)

located at points along the flume. Four strain gauges were located along the height of each concrete filled tube foundation pile, near the bottom and top of both the front and back of each pile. Exact locations of the strain gauges can be seen in Fig. 6.4.

Table 6.1. Instrument Locations

Sensor Name	X Pos. [m]	Y Pos. [m]	Z Pos. [m]	Sensor Type
WG1	13.96	-1.39	1.33	Wire Wave Gauge
WG2	17.6	-1.39	1.32	Wire Wave Gauge
WG3	21.28	-1.39	1.31	Wire Wave Gauge
WG4	24.93	-1.38	1.32	Wire Wave Gauge
WG5	28.59	-1.38	1.32	Wire Wave Gauge
WG6	32.24	-1.38	1.32	Wire Wave Gauge
USWG1	36.04	-1.38	3.63	Ultrasonic Wave Gauge
USWG2	39.97	-0.03	3.82	Ultrasonic Wave Gauge
ADV1	28.57	-1.45	0.92	Acoustic Doppler Velocimeter
ADV2	32.22	-1.45	0.93	Acoustic Doppler Velocimeter
ADV3	36.08	-1.46	0.93	Acoustic Doppler Velocimeter
B06	40.77	0.00	0.76	Pressure Sensor
D11	40.77	0.39	0.02	Pressure Sensor

Overview of Experimental Results

Complete experimental results are available in Pyke [2020], with further discussion in Lewis et al. [2022]. Two wave types of nearly identical time scale were utilized in Pyke's experimental study ($T \approx 2.25$ s). The first wave type is referred to as the 1.40 m wave, and can be classified as an 'unbreaking solitary' wave. The second was slightly taller, is referred to as the 1.45 m wave, and broke almost immediately after forming. This wave approached the structure as a broken bore and can be characterized as a 'spilling breaker' wave. These two waves of similar height were chosen to investigate the effects of the turbulent bore of the broken wave upon the forces imparted to the specimen. As the forces from the 1.45 m wave were found to be larger in both the experiments conducted by Pyke and in the associated CFD simulation with respect to the results from the 1.40m wave, the 1.45 m wave height experimental results were chosen to be investigated for FSI and CFD

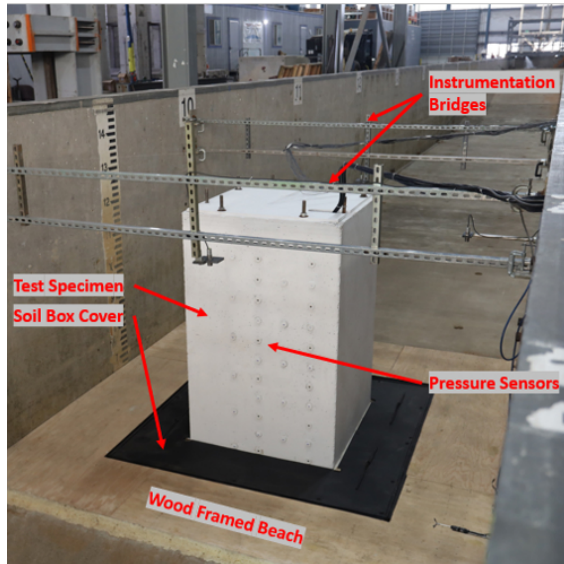


Fig. 6.3. Specimen In Testing Configuration

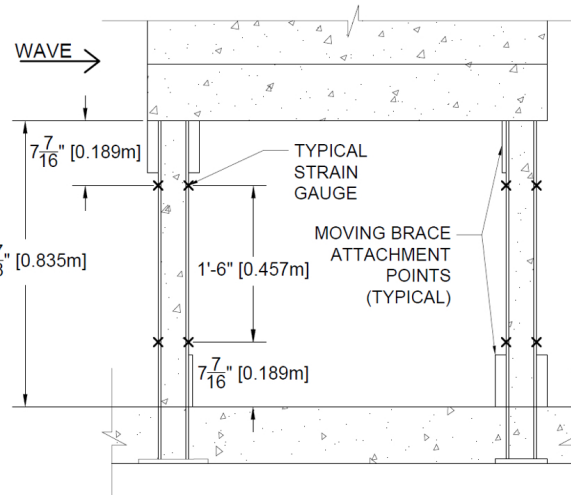


Fig. 6.4. Strain Gauge Locations

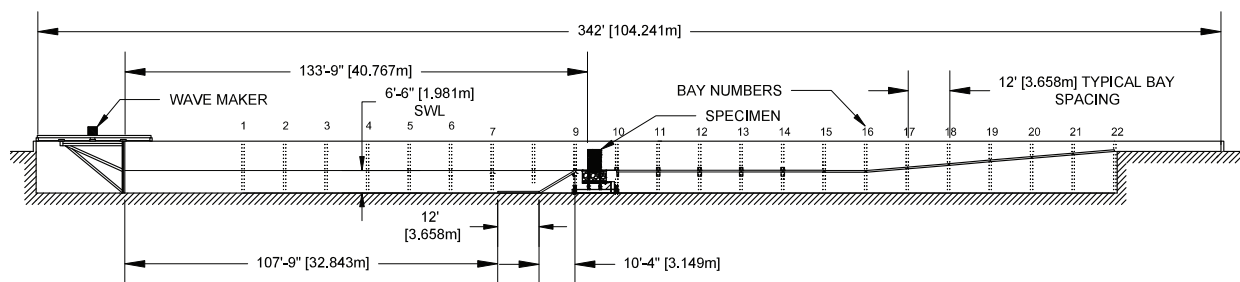


Fig. 6.5. Specimen Location in Flume (Figure from Pyke [2020])

analysis comparison, and thus information pertaining only to this case was reported in this section.

Overall, 20 experimental trials were run for the case examined. Peak forces and moments in all directions for the experimental results from Pyke for the examined case are listed in Table 6.2. Peak streamwise force was computed in multiple ways, using load cells, strain gauges, and through the integration of pressure sensors by means of a linear interpolation between the locations of the pressure sensors. Forces from specimen response were measured using the load cells, as well as with the strain gauges attached to the concrete filled tubes (CFTs) located below the concrete specimen. Overall specimen shear forces, axial forces, and moments were then computed by summing forces at known stations about the specimen's center of rigidity, which was presumed to be at the center of

the base of the concrete portion of the specimen.

For a wave with a maximum crest height of 1.45 m and a celerity (the velocity with which a wave advances) of 5.82 m/s the structure experienced forces near 30 kN in the streamwise direction, with an overturning moment of over 20 kN-m in the pitching direction. These are among the largest forces that can be generated in the flume for a still water level of 2.0 m.

Table 6.2. Peak Forces and Moments by Study Parameter: Fx (Streamwise), Fy (Transverse), Fz (Lift), Mx (Roll), My (Pitch), Mz (Yaw)

Wave Hgt. [m]	Measurement Origin	Soil Box Condition	Fx [kN]	Fy [kN]	Fz [kN]	Mx [kN – m]	My [kN – m]	Mz [kN – m]
1.45	Pressure Sensors	Empty	25.91	N/A	N/A	N/A	N/A	N/A
1.45	Strain Gauges	Empty	25.74	0	11.13	2.28	24.31	0.41
1.45	Load Cell	Empty	37.62	0.61	106.09	0.8	22.41	2.57

Floor panels which spanned from the edge of the soil box to the face of the specimen took vertical force from water bearing upon the panels near the specimen as the wave passed by. These forces were measured by the load cells which were located under the soil box directly beneath the specimen's foundation piles. The location of the flooring from which the additional forces originated can be seen in Fig. 6.3, as black regions near the base of the structure. Force in the streamwise direction from the impact of the wave against the flooring surrounding the specimen was transferred through the soil box and measured by load cells positioned downstream from the specimen, resulting in the discrepancy in force in the streamwise direction between the load cell and strain gauge resultants.

A small asymmetry in either the placement of the specimen within the flume or the stiffness of the instrumentation assembly holding the specimen resulted in slight rolling and yawing moments upon the specimen about its minor axes of rotation, as well as translation reaction forces associated with these actions. This can be seen in the upper and lower right panels of Fig. 6.6. The specimen first experienced a force in the positive roll direction when the wave initially struck, then a reversal of the sign of the moment back to the negative direction as the wave bore rolled around its sides. At

2131 this instance, the soil box appeared to experience a slight transverse force (<0.5 kN) associated with
 2132 this rolling moment and its effects on the flow around the structure.

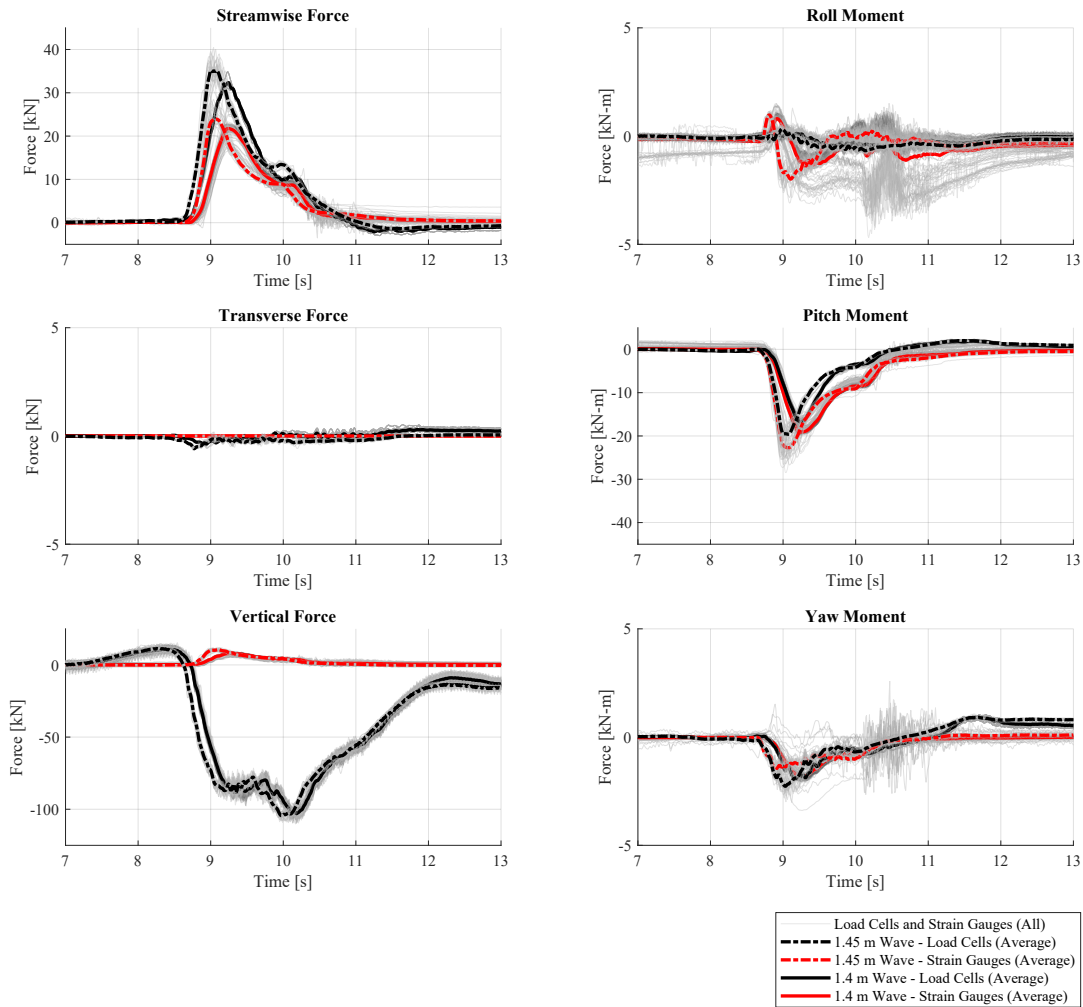


Fig. 6.6. Wave Impact Forces - All Trials (Grey), Trial Averages from 1.40m (Solid) and 1.45m (Dashed) Wave Height Sequences: Force Comparison Between Load Cell Measurements and Strain Gauge Measurements

2133 6.1.2 Numerical Investigation With Computational Fluid Dynamics

2134 The following section investigates the efficacy of CFD in simulating wave-structure interaction and
 2135 hydrodynamics from paddle-generated breaking solitary waves and resolving experimental forces

for integrated study of fluid behavior using a model at 1:1 scale with the experiments.

Model Description

Model dimensions can be seen in Fig. 6.7. Atmospheric boundary conditions for variables U (velocity) and p_{rgh} (dynamic pressure) were applied to the topmost boundary of the model. All walls except the atmosphere boundary were assigned standard wall boundary conditions for the simulation. Specific boundary condition types for the turbulence model and dynamic mesh library were applied to necessary regions of the model for initialization of mesh movement and satisfaction of the requirements for numerical stability. Waves were generated by a pneumatically-powered piston at one end of the wave flume, which then propagated down the length of the flume to a sloped bathymetry, forcing the waves to initially spill and then fully break, which results in a turbulent bore prior to the flow reaching the test structure.

In order to replicate the experimental wave motion and measured fluid loads imparted on the structure, the displacement of the paddle during experiments was recorded as a time history. This paddle displacement history was imposed on the wavemaker paddle surface in the OpenFOAM model using the olaFlow boundary condition function `wavemakerMovement` to specify the mesh motion. More specifically, mesh-morphing wave generation boundary conditions were used to mimic the piston-driven paddle wavemaker at the Hinsdale Wave Research Lab by prescribing the variable 'pointDisplacement' within OpenFOAM. All boundary conditions applied in the models are listed in Table 6.3. For further information and a comprehensive evaluation of OpenFOAM boundary conditions used in Large Wave Flume Models such as those utilized in this study, see Winter et al. [2020].

The $k-\omega$ Shear Stress Transport ($k-\omega$ SST) model was utilized for the modelling of turbulence. The $k-\omega$ SST model, described by Menter and Langtry [2003b], models turbulent kinetic energy, k , turbulent kinetic energy dissipation rate within the boundary region, ω , and turbulence viscosity,

2161 ν_t . This turbulence model formulation captures the turbulent kinetic energy generation in the near
 2162 field of the wall much better than $k-\epsilon$ models, resulting in higher fidelity of shear stresses and
 2163 forces imparted upon objects or structures within the flow. ϵ in these equations represents rate of
 2164 dissipation of turbulent kinetic energy further away from the boundary. The $k-\omega$ SST model is
 2165 essentially an extension of $k-\epsilon$ models, with almost identical behavior of the $k-\omega$ SST model and
 2166 $k-\epsilon$ models far away from boundaries; however, the authors found slightly better results for viscous
 2167 shear are generated near structures with the $k-\omega$ SST model. The turbulence model was initialized
 2168 with the following values for turbulence coefficients: k ($9.54E-3$), ϵ ($4.45E-4$), C_μ ($0.90E-1$), and
 2169 ω (0.52).

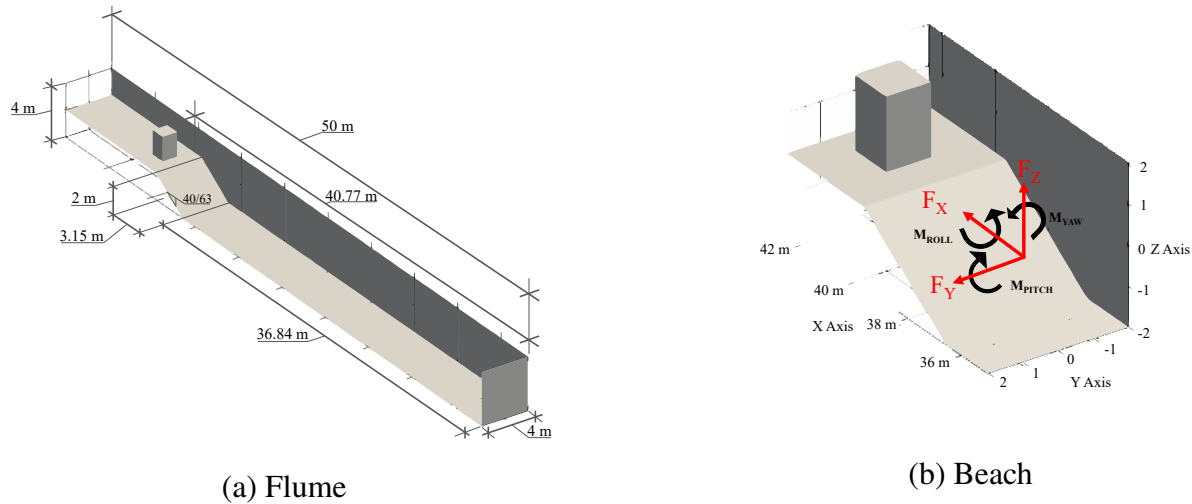


Fig. 6.7. Model Dimensions

Table 6.3. CFD Model Boundary Conditions for All Field Variables

Field	Atmosphere	Walls	Wavemaker
U	pressureInletOutletVelocity	noSlip	movingWallVelocity
p_{rgh}	totalPressure	fixedFluxPressure	fixedFluxPressure
ν_t	calculated	nutkWallFunction	nutkWallFunction
k	inletOutlet	kqRWallFunction	kqRWallFunction
ω	inletOutlet	omegaWallFunction	omegaWallFunction
ϵ	inletOutlet	epsilonWallFuction	epsilonWallFuction
α	inletOutlet	zeroGradient	zeroGradient
pointDisplacement	fixedNormalSlip/fixedValue	fixedNormalSlip/fixedValue	wavemakerMovement

The computational domain was comprised of 3.2 million points and 3 million cells, with cell sizes varying from 10 cm $\times$ 10 cm $\times$ 10 cm far from the structure to 1.5 cm $\times$ 1.5 cm $\times$ 1.5 cm near the structure. The CFD model was initialized with a maximum timestep of 1E-4 s and a minimum timestep of 1E-9 s, with time step size governed by a Courant-Friedrichs-Lewy (CFL) condition of 0.5. Models were run on UW Hyak K1one with 40 processors with 128G of memory per processor and took approximately 20 hours to complete.

Hydrodynamics Comparison and CFD Computational Results for Velocity and Free Surface

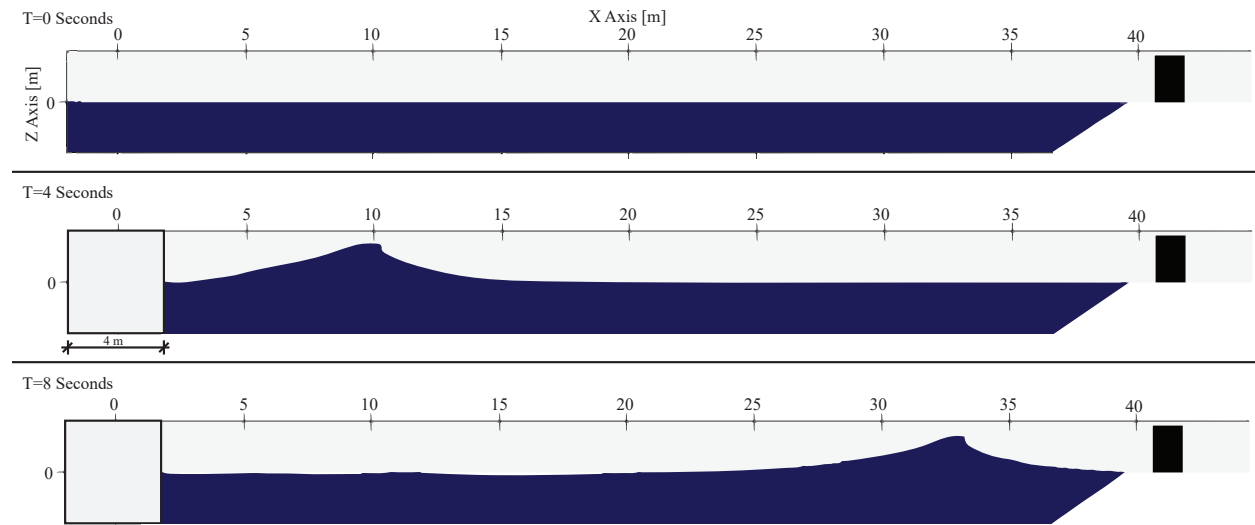


Fig. 6.8. Side Elevation of OpenFOAM Model at Selected Times

The simulated results for wave movement, pressure distribution over the specimen, and structural forces were compared against the measured responses to evaluate the accuracy of the model. Fig. 6.8 shows elevation views of the simulated wave at various points of travel down the flume, as well as the end state (fully-extended) of the prescribed motion of the wavemaker as the water moves toward the test specimen.

Side-by-side comparisons of simulation results and videos obtained from experiment trials are depicted in Figs. 6.9(a) to 6.9(c). General shapes of the free surface of the multi-phase air-water

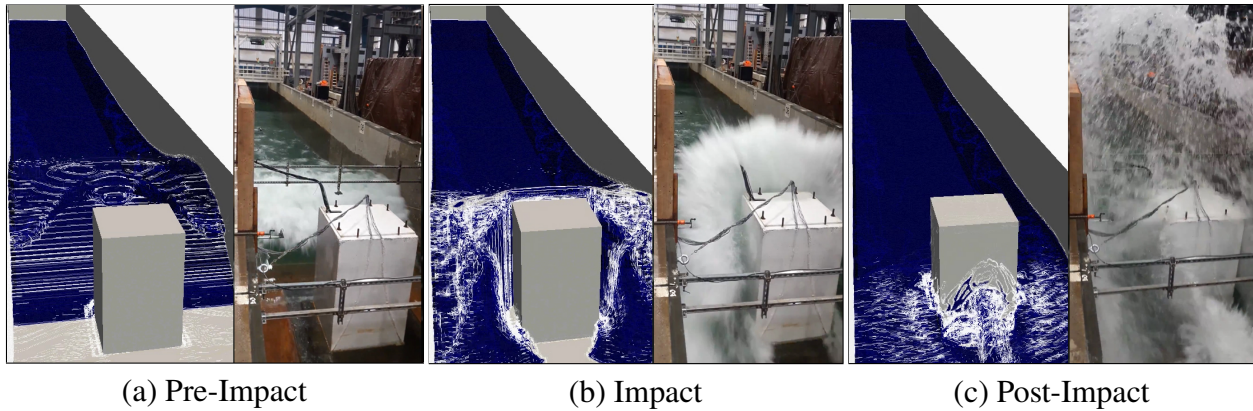


Fig. 6.9. 145cm Wave Impact: Comparison

mixture agree well with those observed in videos of the experiments, with a reasonable resolution of the splashing of the fluid around the structure upon impact. As shown in Fig. 6.9(c), the experimental and numerical post-impact wake shapes behind the specimen match well.

Experimental time histories of the free-surface height for the wave-gauge instruments as well as the calculated location of the free surface of the simulated fluid air-water boundary at identically located sample points within the computational model are shown for comparison in Fig. 6.10.

Comparison between measured values for the free surface and the location of the free surface calculated in OpenFOAM show strong agreement. Velocities for ADVs 1 and 2 match well with velocities calculated in OpenFOAM. ADVs 3 and 4 experienced a large degree of noise from bubbles being created and collapsing, producing acoustic waves, and the broken wave bore near the test structure. A comparison of ADV 3 with its associated OpenFOAM velocity probe are shown in Fig. 6.11, in which the substantial amounts of noise for ADV 3's measured values are shown.

Contour plots of the pressure over the upstream face of the specimen obtained from OpenFOAM simulations and experiments at the instant the peak base shear occurred are shown in Fig. 6.12. The experimentally obtained pressures are shown in the right panel - this contour was constructed by reflecting pressure measurements from all pressure sensors about the center-line of the specimen and using a cubic interpolation function between values of pressure at discrete locations. The signal from the pressure sensors utilized in experiments seems to be attenuated around the peak pressure,

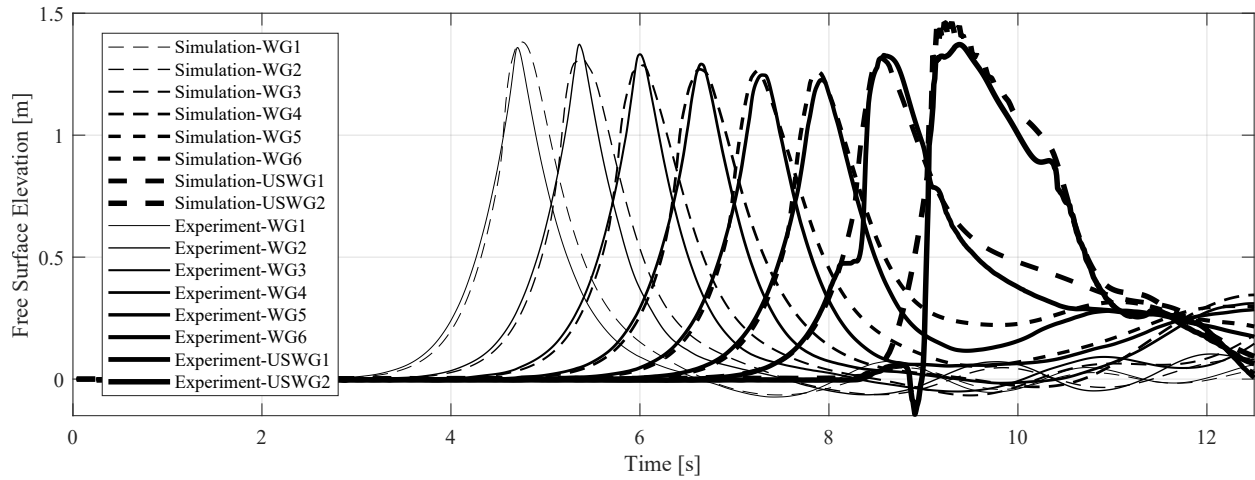


Fig. 6.10. Test-Simulation Free Surface Height Comparison - 145cm Wave

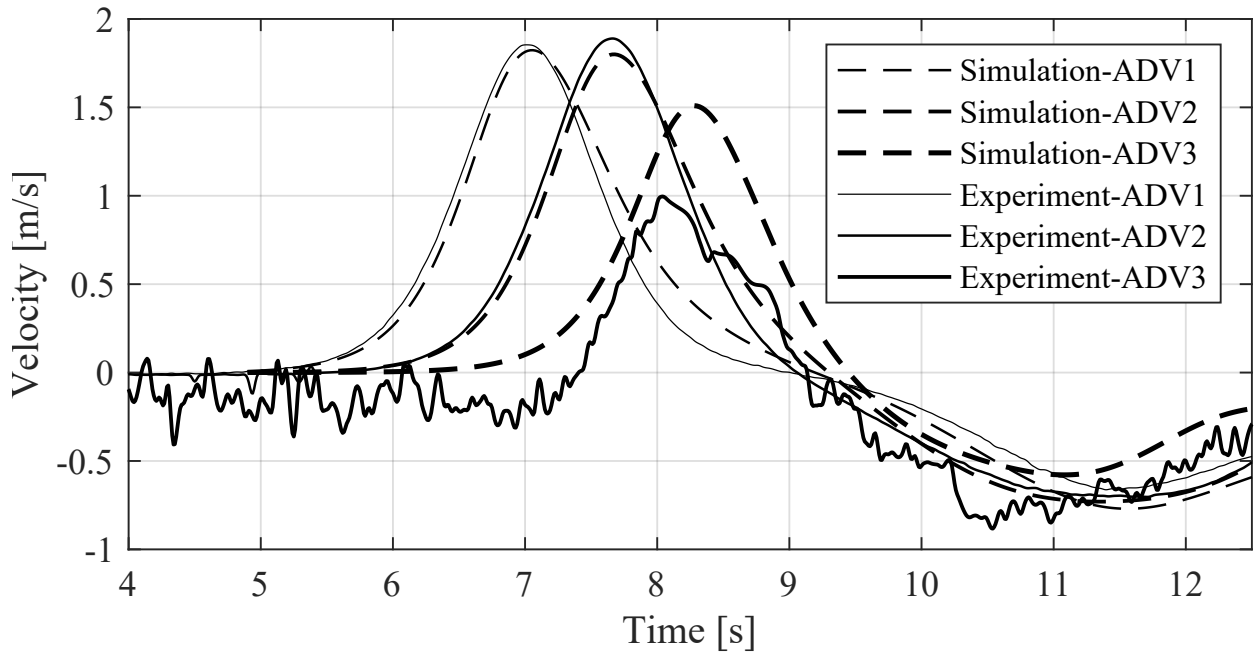


Fig. 6.11. Test-Simulation Velocity Comparison - 145cm Wave

which manifests as lower overall pressure magnitudes over the upstream face of the specimen at the instant of peak streamwise force. As such, the contours of experimental and simulated pressures do not appear to match very well at the peak force instance, however local pressure sensor location comparisons show reasonable agreement with slight temporal and magnitude variations.

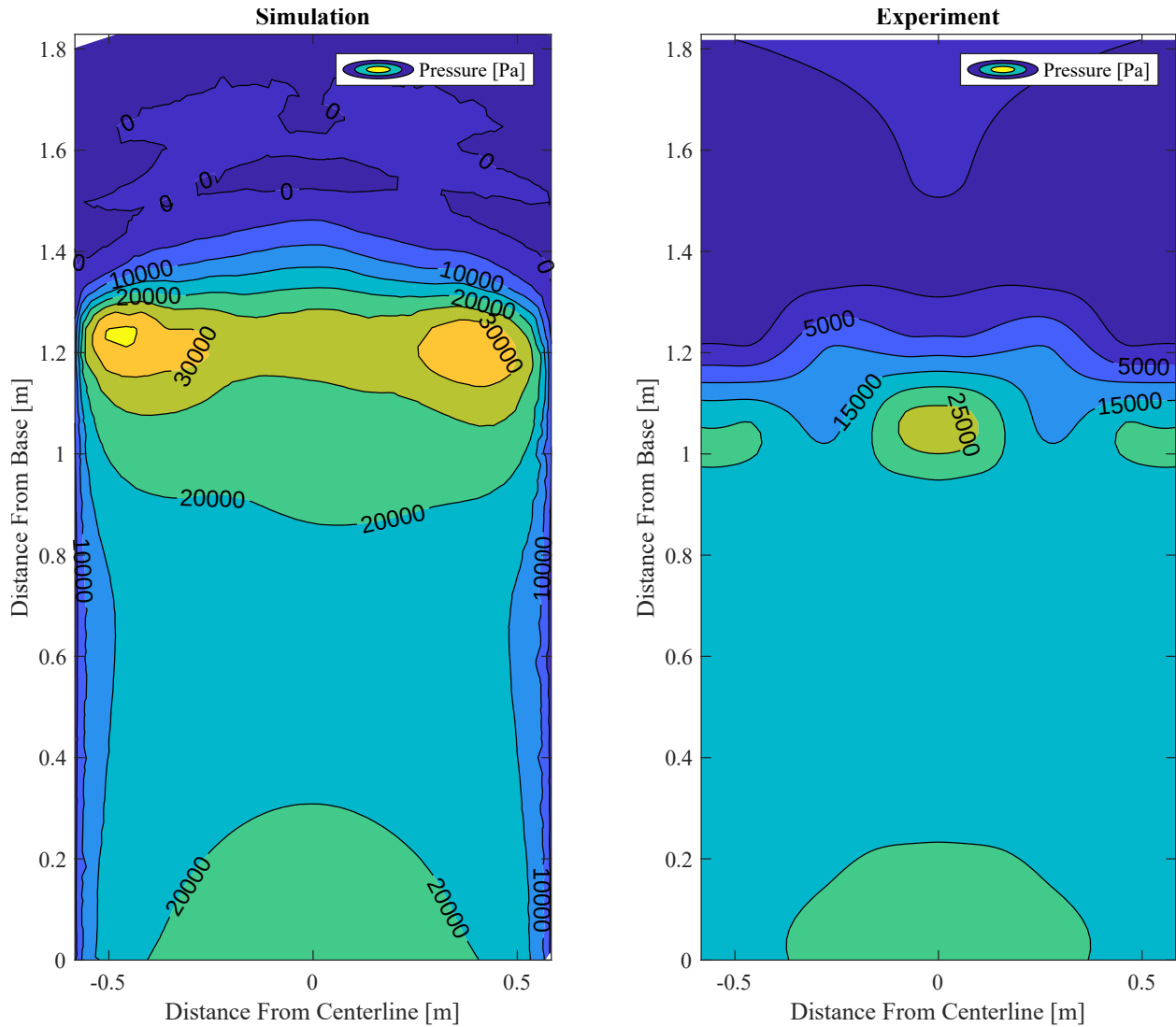
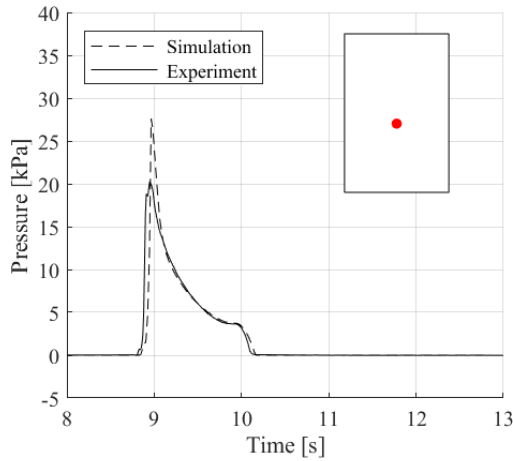


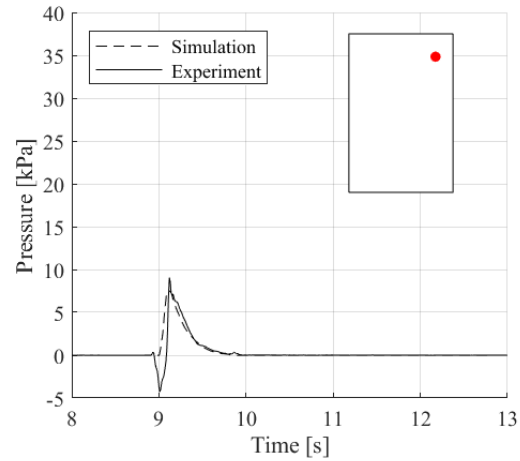
Fig. 6.12. Trial Averaged Upstream Face Pressures at Peak Base Shear (Time:9 seconds)

Pressure contours upon the upstream face of the specimen help determine the nature of the instantaneous and sustained loading profiles upon structural elements and walls subject to bore impact and flow entrapment, for which the shape of the loading profile is significant to evaluating the force response of critical structural elements. Locally measured and simulated pressures show strong agreement over the duration of the loading. This also applies to time-integrated pressure values and resultant impulses, particularly near the base of the structure, where cell sizes are smaller than higher up on the wall. Trial-averaged pressure and impulse time histories at two representative points along

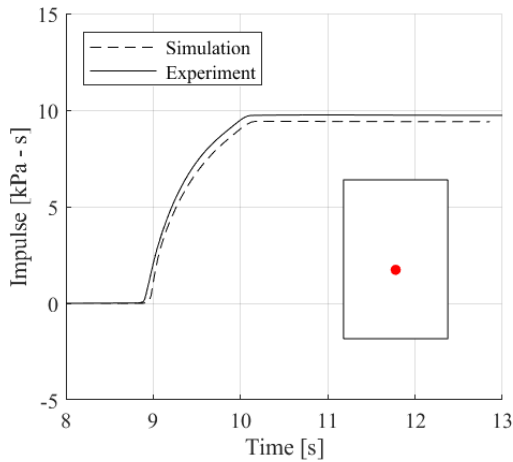
the seaward face of the structure were chosen for comparison of experimentally measured values and those simulated by the numerical model are shown in Figs. 6.13(a)-(d).



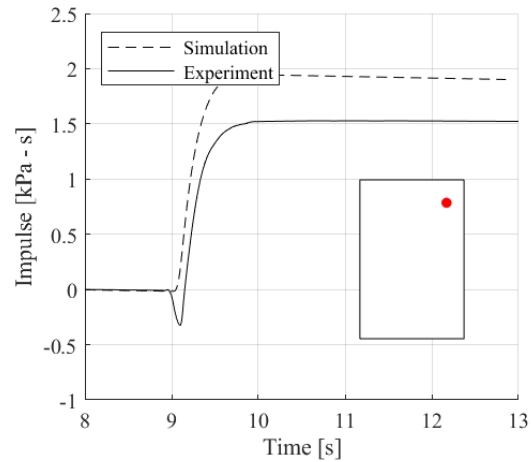
(a) Sensor B06 Pressure



(b) Sensor D11 Pressure



(c) Sensor B06 Time-Integrated Pressure



(d) Sensor D11 Time-Integrated Pressure

Fig. 6.13. Trial Averaged Pressure and Time-Integrated Pressure Time Histories for Two Representative Seaward Face Sensors, B06 and D11

These sensors, B06 and D11, were located near the center and near the lower right corner of the specimen's upstream face respectively. While pressures match well at at order of magnitude, at the local level the pressure distribution over the face of the specimen could be better resolved temporally

and spatially with a finer mesh in the plane perpendicular to the flow of the wave. It is presumed that due to the size of the cells, some averaging is occurring between peaks and troughs in pressures which otherwise would have resulted in a closer match to experimental pressure measurements. For this study the mesh was kept at a moderate level of refinement to keep numerical model run-times within reasonable limits, which was deemed acceptable for the phenomenological purposes of this investigation.

6.1.3 Fluid-Structure-Interaction Analysis

The case was run first to completion with a CFD-only analysis. To save time, the fields from this analysis were mapped to a FOAMySees FSI model utilizing OpenFOAM's '**mapFields**' utility at $T=8.75$ s, shortly after the wave had crested at the flume bathymetry and before it had reached the specimen.

OpenSees Model Details

A simplified OpenSees model was built of the concrete core wall specimen, along with the concrete-filled-tubes which supported the core wall. An elastic material with a modulus of elasticity of $5e9$ Pascals, density of 2400 kg/m^3 , and a Poisson ratio of 0.25 was utilized to model the concrete. An 'ElasticMembranePlateSection' with 6 inch thickness was used with 'ShellMITC4' elements roughly 10 cm x 10 cm in size comprising the core wall specimen geometry at the wall and slab centerlines. Uni-axial material properties for structural steel ('Steel02' model, with a yield stress of 344.75 MPa, initial elastic tangent of 200 GPa, strain-hardening ratio of 0.1, and isotropic hardening parameters of a_1 , a_2 , and a_3 of 18.0, 0.925, and 0.15, respectively) and concrete ('Concrete02' model, with concrete compressive strength at 28 days of -49.64 MPa, concrete strain at maximum strength of -0.00326, concrete crushing strength of -9.93 MPa, concrete strain at crushing strength of -0.01631, ratio between unloading slope and initial slope of 0.1, tensile strength of 4.39 MPa, and tension softening stiffness of 2 GPa) respectively were chosen for modelling materials of the

CFT tubes within OpenSeePy. Fiber sections with elastic uniaxial materials and 64 fibers each were utilized to represent the composite sections of the columns, which were standard 10.16 cm (4 in) steel pipes with 1.27 cm (1/2 in) thick walls filled with concrete. EqualDOF commands were utilized for connection of structural elements of different formulations within OpenSeesPy. The structure was fixed at its base.

The model was given Rayleigh damping in OpenSeesPy with a value of 7.5 % from the frequency of the first structural mode (f_1) to five times that frequency ($5f_1$) with Rayleigh mass coefficients of $\alpha_{mass} = 0.0$ and Rayleigh stiffness coefficients of $\beta_{tangent} = 0.0$, $\beta_{initial} = \frac{\zeta * 5f_1 - \zeta * f_1}{\pi * (5f_1^2 - f_1^2)}$, and $\beta_{committed} = 0.0$.

6.1.4 Discussion

Challenges

Time increments required by the CFD model at the resolution needed to obtain good forces are small. Standard adaptive timestepping to maintain CFL criteria cannot be utilized without modification of the OpenFOAM coupling adapter ('adjustTimeStep' in system/controlDict must be set to 'false'). Since the fluid is incompressible, heavy numerical instabilities could arise without an implicit coupling scheme stabilizing the solution at large timestep sizes or other means to control the work transfer across the coupling interface. Furthermore, the explicit coupling scheme has a hard time converging during later points of the wave impact due to the high frequency interactions between the models and the effects of added mass instabilities in the partitioned coupling.

A CFD-only simulation takes roughly 14 hours of execution time to propagate the wave for 9 s of simulation time to the structure while using adaptive time-stepping to allow the timestep to increase in situations where the CFL number is less than 0.5. If an implicit coupling scheme were utilized at a timestep of 1e-6 s (the required timestep to achieve a CFL condition of 0.5

throughout the whole simulation), an FSI analysis would take significant amounts of time. Using an implicit coupling scheme means iterating over the timestep multiple times, which may be costly for large CFD models. If a CFD-only analysis running at timestep increments of $1\text{e-}6$ s takes 1 day to complete, an implicitly-coupled analysis would take as many days as the duration of the CFD analysis multiplied by the number of iterations required on average for the model to converge. As the timestep is reduced, more high-frequency interactions between the two analyses are captured, and thus more coupling iterations may be required to ensure convergence.

Approach

The OpenFOAM model was progressed in 50 increments of the displacement calculated by OpenSees during the coupling timestep. Using these modifications to the timestepping and time data interpolation of the coupling participants, work across the coupling interface is preserved, and the CFL criterion of the CFD model can be satisfied without the utilization of small coupling timestep sizes. This results in a simulation which is 'tightly-coupled,' but maintains a faster simulation time due to the ability of each solver to utilize the accelerated data from the indirect Quasi-Newton implicit coupling scheme to predict a solution trajectory to the next coupling timestep which results in equivalent amounts of work across the coupling interface after a sufficient number of coupling iterations. For this case, these settings resulted in satisfaction of the residual convergence tolerance fairly quickly. The easing of the temporal work transfer requirements across the interface along with reduction of the number of coupling timesteps and iterations taken along with time-interpolation of data vastly improves numerical solution speeds. This smooths the work transfer functional (See 7.1.1), but if timesteps are small enough to capture the governing physics and forces of the problem of interest while maintaining a reasonable simulation duration then the methodology could possibly be utilized for solutions over larger domains or more quickly than if the timestep were to be kept at the same size for each component analysis.

FSI Simulation Results

As seen in Table 6.4, with respect to the values for base shear and overturn moment calculated from the strain gauges upon the supporting CFTs and the forces obtained from the numerical model, shear and vertical forces were over-measured by the load cells, but overturn moments were under-measured. This is assumed to be due to the presence of large magnitudes of water near the upstream face of the specimen weighing the panels surrounding the specimen down, and thus imparting the force of its weight on the load cells nearest the front of the structure and reducing their contribution to the overturn moment. This 'floor-loading' phenomena was simulated within OpenFOAM and compared to the measured values from the load cells, seen in the Fz column of Table 6.4.

Table 6.4. 1.45 m Wave Force Comparison: Simulated vs. Experimentally Measured

	Fx [kN]	Fy [kN]	Fz [kN]	Mx [kN - m]	My [kN - m]	Mz [kN - m]
Strain Gauges	25.74	0	11.13	2.28	24.31	0.41
Load Cell	37.62	0.61	-106.09	0.8	22.41	2.57
Pressure Sensors	25.91	0	0	0	0	0
OpenFOAM CFD Simulation	31.79	0	-111.01	0	22.65	0
FOAMySees FSI Simulation, OpenFOAM $\delta_t = 1e - 5s$, Coupling $\delta_t = 5e - 4s$, IQN-ILS, Coupling Residual $\epsilon_{rel} = 5e - 3$, Init. Relaxation Factor $\alpha = 0.1$ Time to completion (1 Simulation Second) = 56 hours (real time), 40 Procs, 1 Node, UW HYAK Klone						
Avg. Max Courant No = 2	Fx [kN]	Fy [kN]	Fz [kN]	Mx [kN - m]	My [kN - m]	Mz [kN - m]
OpenSees Reaction	42.445	0.30	31.35	0.11	23.19	0

Forces and moments from the CFD simulations slightly vary from those obtained from the experiments, but these force trends and magnitudes fall within 10 % of the measured loads which are unaffected by slight asymmetries in the test frame's stiffness. These differences are still to be investigated in future research, however the agreement between forces obtained from experiments and numerical models is assumed to be within acceptable ranges for justifying further refinements to the modelling methodologies and advancement of the techniques implemented for tsunami-resilience-engineering purposes. Streamwise forces, overturn moment, and vertical forces from the experiments, CFD models, and FSI models are shown in Fig. 6.14, Fig. 6.15, and Fig. 6.16,

2308 respectively. The simulation was run first with an implicit coupling scheme, with a coupling relative
 2309 residual tolerance of $5e-3$ and a coupling timestep of $5e-4$ s. The work input into the OpenSees
 2310 model for both the implicit and explicit simulation are shown in Fig. 6.17. The explicit simulation
 2311 experiences an instability around the time of maximum force, resulting in large spikes in the resolved
 2312 forces and large amounts of generated spurious work. The implicit simulation ensures that this work
 2313 remains bounded through relative convergence of the data within each coupling timestep. Errors
 2314 for work transfer across interface with respect to work transfer rate is shown in Fig. 6.18. It is
 2315 expected that with a reduced timestep the work transfer errors which arise from the large coupling
 2316 timestep size would be minimized - however, reduction of the timestep will increase the probability
 2317 that the explicitly-coupled simulation will diverge due to the inherent added-mass instabilities of
 2318 partitioned-explicit coupling.

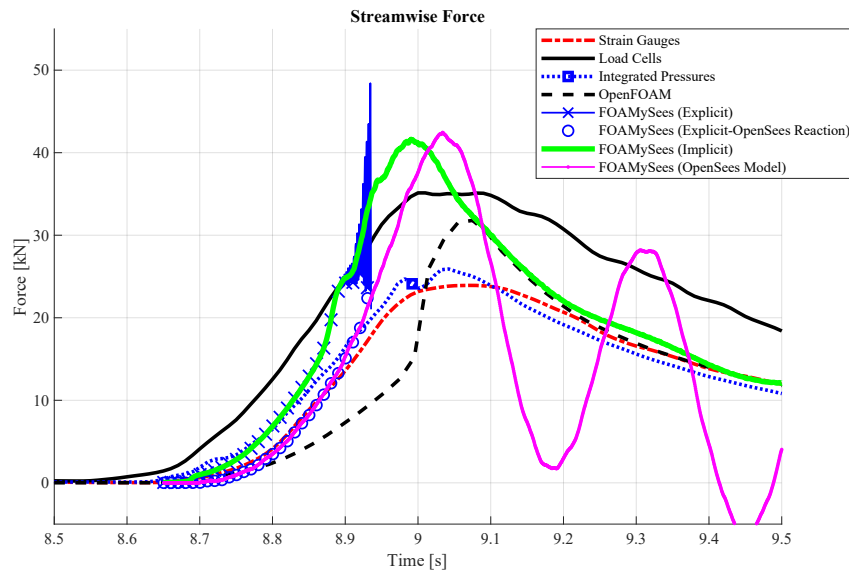


Fig. 6.14. 145cm Wave Impact Forces: Comparison Between Trial Averaged Experimental and Simulated Streamwise Forces. Experimental streamwise forces are measured through load cells, pressure sensors, and strain gauges. Simulated forces were calculated with OpenFOAM and FOAMySees

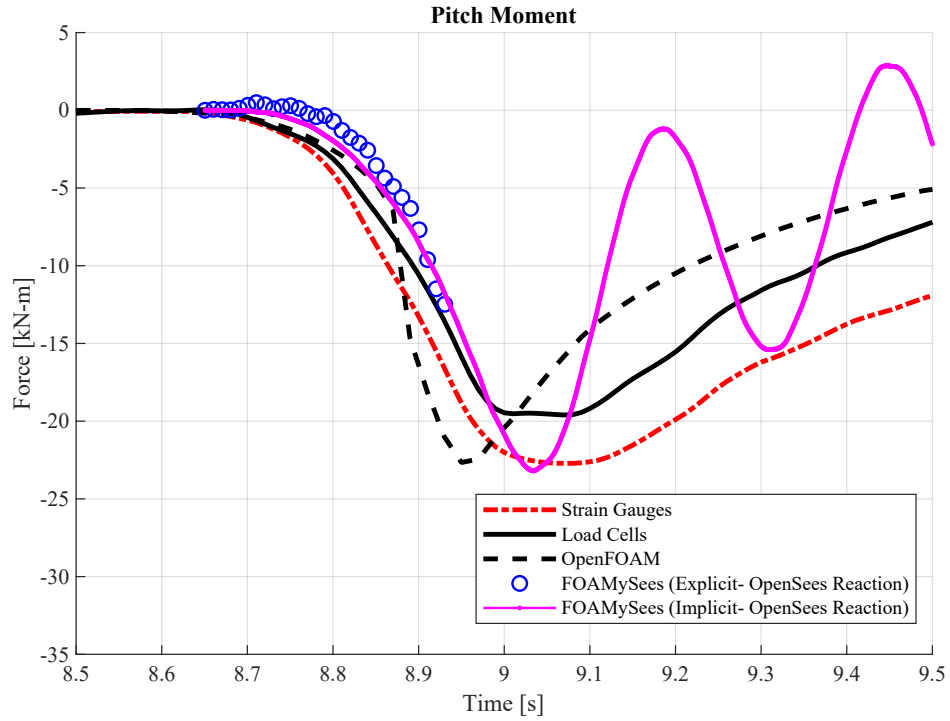


Fig. 6.15. 145cm Wave Impact Forces: Comparison Between Trial Averaged Experimental and Simulated Overturn Moment. Experimental values are measured through load cells and strain gauges. Simulated forces were calculated with OpenFOAM and FOAMySees

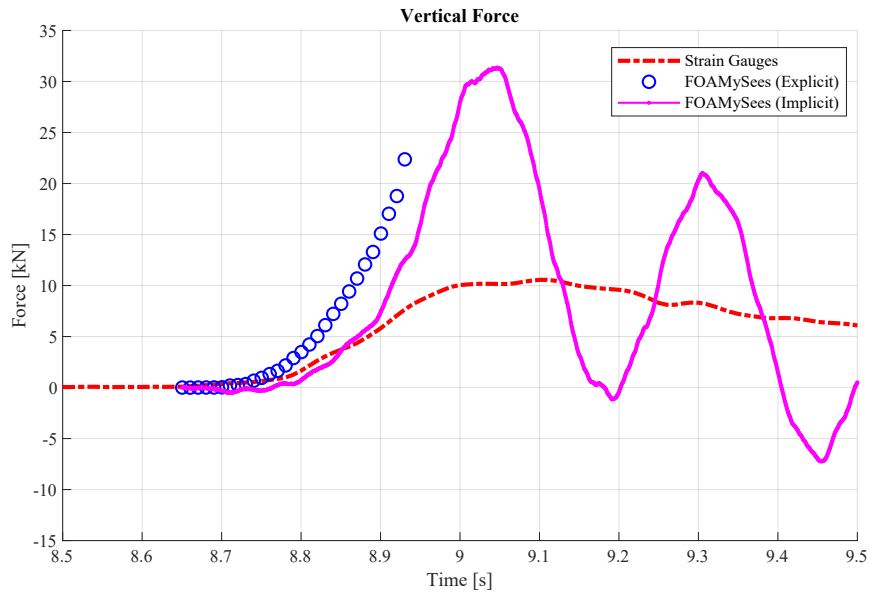


Fig. 6.16. 145cm Wave Impact Forces: Comparison Between Trial Averaged Experimental and Simulated Principal Vertical Forces. Simulated forces were calculated with FOAMySees

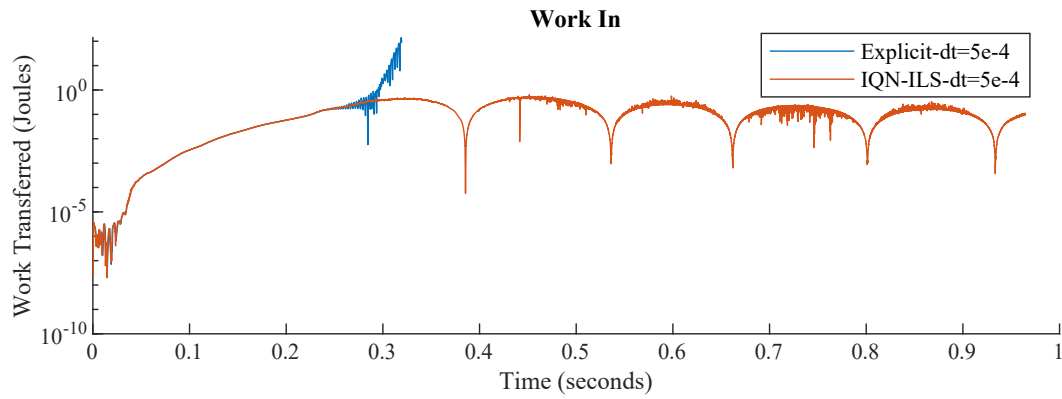


Fig. 6.17. 145cm Wave Impact: Explicit and Implicit Work Into OpenSees from OpenFOAM - Explicit Simulation Experienced Instability, Resulting in Large Amounts of Erroneous Work Transferred

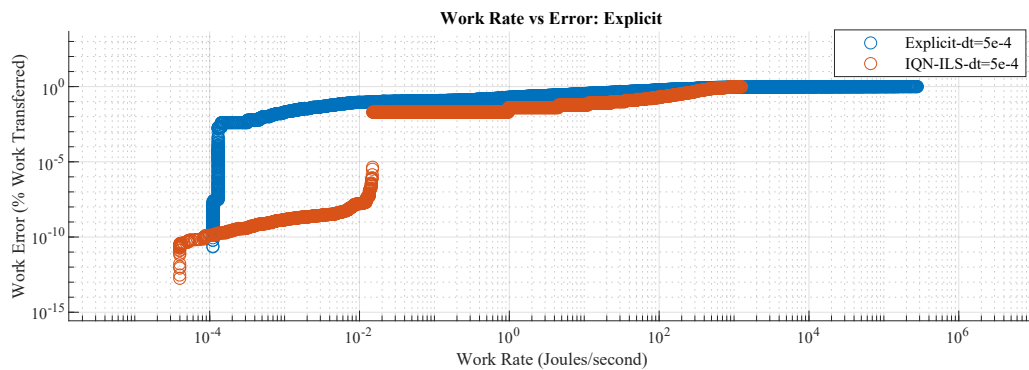


Fig. 6.18. 145cm Wave Impact: Explicit and Implicit Work Errors WRT Work Transfer Rate. The Explicitly-Coupled Simulation Experienced Instabilities, Resulting in Large Amounts of Erroneous Work Transferred, but This Erroneous Work is Transferred Almost Entirely to OpenSees

6.2 Three Dimensional Elevated Structure Under Turbulent Breaking Solitary Wave Impingement with Real Structural Sections and Materials

In order to evaluate the efficacy of the force and displacement mapping routines for complex three dimensional geometries with an explicit coupling scheme, a 1:6 full scale structure studied experimentally by Sullivan [2021] within the Large Wave Flume at Oregon State University O.H. Hinsdale Wave Research Laboratory was modelled in OpenFOAM and FOAMySees. This experiment was almost identical in flume geometry, wave properties, and bathymetry to that of the previous section, so further details pertaining to these parameters are omitted here for brevity. Full experimental results are available in Kenneth Sullivan's Masters Thesis (Sullivan [2021]). Limited results are presented here for general force comparison between an explicit simulation and the experimental base shear measured by Sullivan. The model was run with an explicit coupling scheme for 4 s of simulation time at a timestep of 0.0005 s on 1 UW HYAK K1one HPC Node with 40 processors for a total computational time of 14 hours. This resulted in relaxation of the work transfer across the interface by means of numerical damping of the CFD model and omission of high-frequency interactions between the FEA model and the interface forces, but the simulation did not crash, which indicates that possibly the amount of participating artificial added-mass for this particular model and for these coupling settings was lower than the effective inertial resistance of the structure. This study was primarily conducted to demonstrate usability of the software for three-dimensional turbulent cases, and is not intended as an experimental-validation study or as indication that explicit coupling schemes are stable or offer meaningful results.

OpenSees Model Description

The structure was 1.016 m by 1.016 m in plan from column center to column center, comprised of steel frame elements, steel gusset plates, and concrete filled steel tubes. HSS2x2x1/8 elements were framed horizontally between columns with their center-line at heights of 1.3208 m, 1.8288 m, and 2.3368 m. HSS1.5x1.5x1/4 elements were utilized for chevron bracing from 1.3208 m to 1.8288 m along all four sides of the structure. Panels spanned between HSS2x2 elements forming diaphragms within the structural plan footprint. For exact material properties of the experimental specimen and exact structural dimensions, see Sullivan [2021]. The concrete-filled-tubes were modelled in the same manner as those in the previous section, as the same concrete-filled-tubes were utilized for both specimens which were tested. Elastic beam sections were utilized for the HSS components of the frame. Elastic MITC4 shell elements were utilized for the modelling of the panels spanning across each story, which were given a thickness of 12.7 mm (1/2 in) and material properties of elastic steel. The model was again given Rayleigh damping in OpenSeesPy with a value of 7.5 % from the frequency of the first structural mode (f_1) to five times that frequency ($5f_1$).

Branch Meshes

A select number of FEM nodes were chosen for coupling to CFD patch surfaces as the full geometry of the structure was not represented in OpenFOAM. The CFD patch surface utilized in this analysis with the OpenSeesPy model used in FOAMySees overlaid is shown in Fig. 6.19 and the displacement branch groupings utilized are shown in Fig. 6.20.

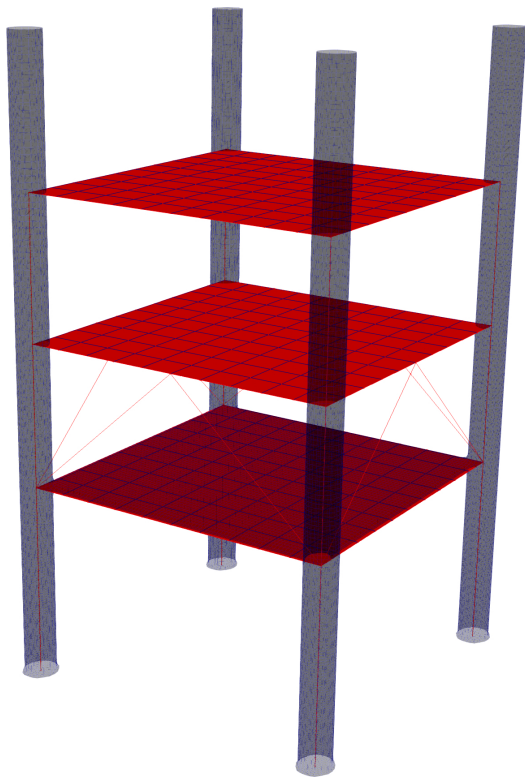


Fig. 6.19. OpenSeesPy model overlaid by OpenFOAM CFD patch surface

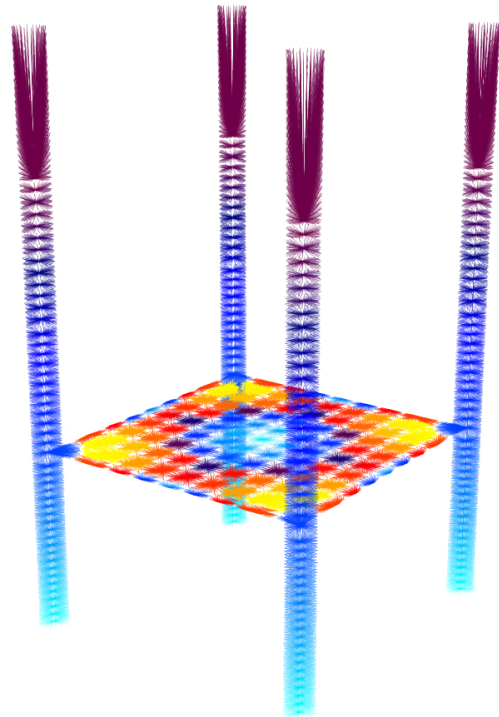


Fig. 6.20. Branch Clusters Utilized in Analysis (branch clusters are given color to show clusters more clearly. Clusters of similar color are *not* associated with one another)

6.2.1 Results

Computational Cost

In comparison to a geometrically similar model utilized for benchmarking computational cost with respect to an equivalent CFD model, (9,960 branch nodes coupled to 35,960 CFD patch surface nodes) the first second of the simulation with the present model (50,000 branch nodes coupled to 50,000 CFD patch nodes) took roughly 3.5 hours, which is nearly twice the computational time of the model described in Sec. 7.2 with nearly identical node and cell counts within the CFD models compared (4,034,512 points and 3,894,298 cells versus 4,061,836 points and 3,917,848 cells). For more information, see Sec. 7.2.

Displacements, Forces, and Impulses

Mesh motion of OpenSeesPy, the FOAMySees displacement branch mesh, and OpenFOAM surface patches along with fluid free surface (isosurface at $\alpha = 0.5$) overlaid for selected times of the simulation are shown in Figs. 6.21 through 6.23. In Fig. 6.24, a time history of the displacement of the top left corner on the upstream face of the structure in OpenSeesPy and its associated location in OpenFOAM are plotted alongside each other to demonstrate proper transfer of displacement during the simulation. To demonstrate proper transfer of force between coupled models, the forces applied to the branch nodes within FOAMySees were output and summed for comparison with the force calculated within OpenFOAM via functionObjects. The streamwise force time histories for both OpenFOAM and FOAMySees are shown along with experimentally measured forces from Sullivan [2021] for the duration of wave impingement in Fig. 6.25. This analysis and experiment will be investigated with an implicitly-coupled model utilizing incremental time stepping within the OpenFOAM model in future work.

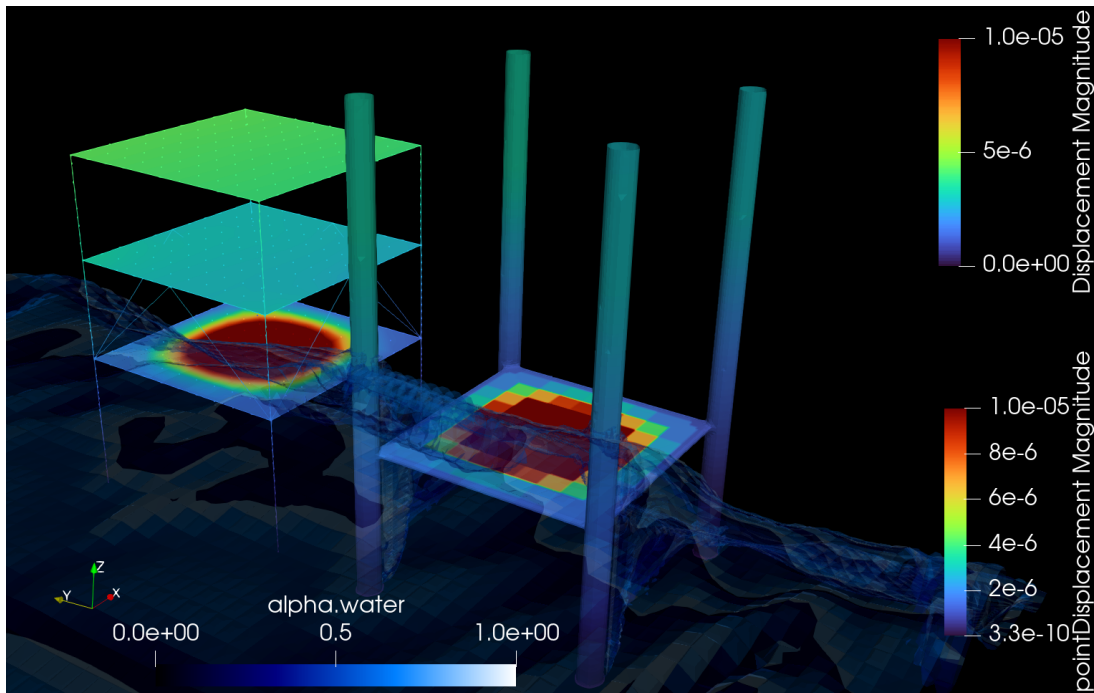


Fig. 6.21. OpenFOAM and OpenSeesPy Meshes at 0.7 s with fluid free surface

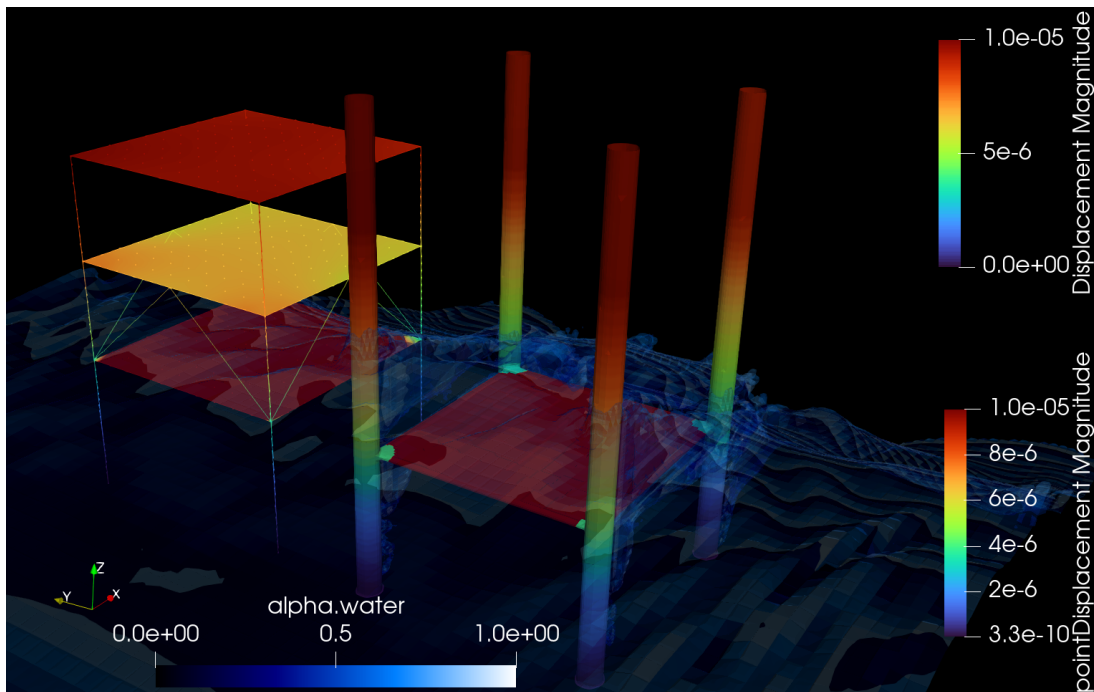


Fig. 6.22. OpenFOAM and OpenSeesPy Meshes at 0.9 s with fluid free surface

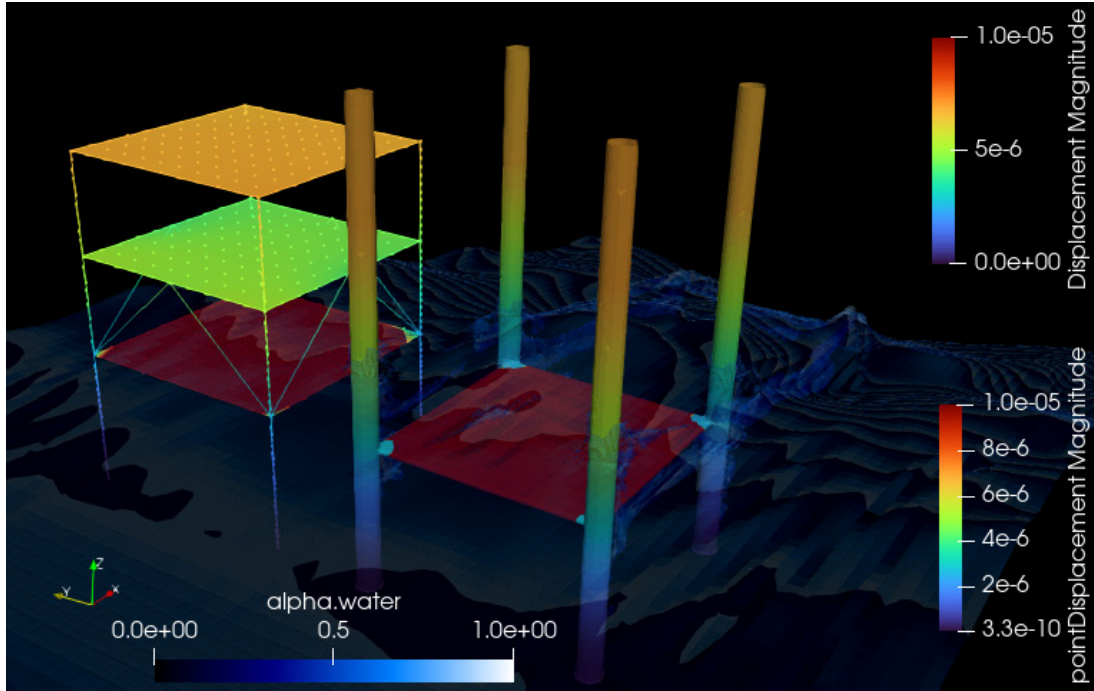


Fig. 6.23. OpenFOAM and OpenSeesPy Meshes at 1.1 s with fluid free surface

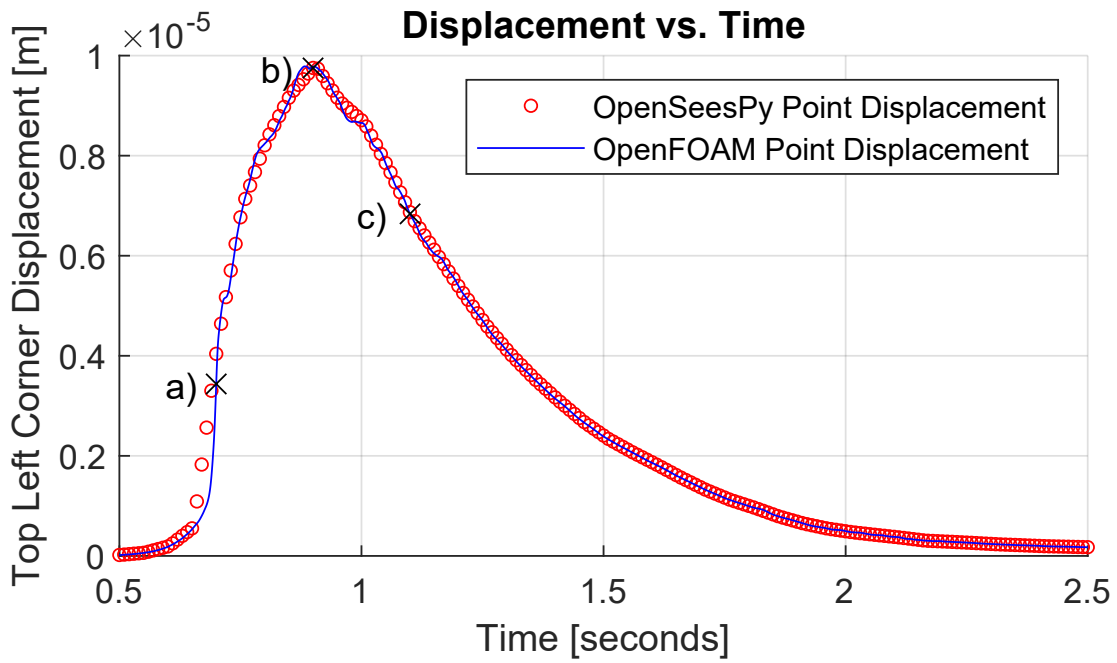


Fig. 6.24. OpenFOAM and OpenSeesPy Displacement Over Time for Point of Maximum Lateral Displacement (Top of Left Front Column)

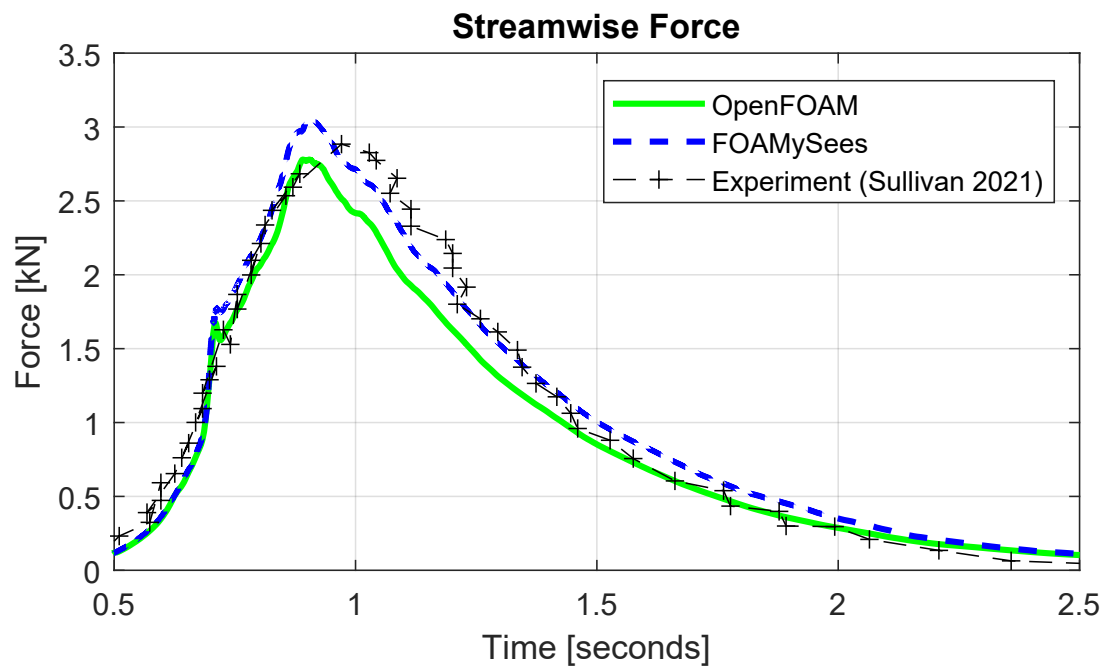


Fig. 6.25. Streamwise Force Comparison Between Experiment, OpenFOAM CFD, and FOAMySees FSI

Chapter 7

Discussion

This chapter contains discussion of the work transfer across the coupling interface with respect to coupling settings and time increment size, computational performance of the proposed coupling method with respect to analogous CFD models, modelling considerations for the presented coupling methodology, limitations, as well as a supplemental discussion on the results obtained from convergence studies conducted within the previously presented Analytical Test Cases, Chapter 5.

7.1 Effect of Coupling Timestep on Solution

The effects of coupling timestep size on the accuracy of a coupled solution are discussed in this subsection. Time integration step size selection and the use of explicit and implicit coupling schemes were investigated briefly in the Analytical Test Cases Chapter (Chapter 5). This section is intended as a supplement to the previous study conducted with the Walhorn analytical test case, with the purpose of elaborating on the nuances of the effects of timestep selection in modelling strongly-coupled fluid-structure-interaction with the presented methodology.

7.1.1 Work Equivalence Over the Coupling Data Projection Mesh

For equivalence of work transfer over an interface between a structural dynamics model and a projected coupling mesh to be satisfied, input and output work for a period of time must be equal or within satisfaction of a tolerance for norm of coupling data residual difference between the two source meshes. This can be described as keeping the amounts of input and output work roughly equivalent at a given instance or over a given period of time.

As the geometries of the FEA and CFD model meshes are different, this results in a pair of sets of work which must contain norms that are roughly equivalent. To prevent errors in particular nodal responses, work equivalency across the interface must be satisfied for every node at the coupling interface and each node at the coupling interface is tied to a finite element node, this means externally applied work as calculated by the product of the forces applied to the coupling mesh with the displacement field of the coupling mesh for a given timestep or increment must be balanced by the work calculated through the product of the displacement and rotation of the structure with the internal resistance generated from the motion of the structural finite element model for the same time increment.

For sake of simplicity, consider a single spring finite element with six degrees of freedom, constrained at one node, resulting in a configuration of the entire CFD mesh at the coupled interface being coupled to one FEA node. From the context of the finite element model, the work equivalency over the interface between coupling participants is more accurately described if split into the discrete components from which the work equivalency requirement satisfaction must be satisfied. Using this context, the work equivalency can be loosely defined as follows, with each of the following components are considered to be time-increment-averaged values from (t^n) to (t^{n+1}) .

$$\int_{t^n}^{t^{n+1}} \langle \hat{F}_{CFD} | \hat{\Delta}_{CFD} \rangle dt = \int_{t^n}^{t^{n+1}} \langle \hat{F}_{FEA} | \hat{\Delta}_{FEA} \rangle + \langle \hat{M}_{FEA} | \hat{\Theta}_{FEA} \rangle dt \quad (7.1)$$

where:

2421 $dt = (t^{n+1}) - (t^n),$

2422 $(\dot{})$ represents a single time derivative of displacement of an arbitrary field variable

2423 $\hat{F}_{CFD}$ = force field on the coupling interface mesh calculated from the CFD model

2424 $\hat{\Delta}_{CFD}$ = the displacement field of the coupling interface mesh to be projected to the CFD boundary
2425 surface,

2426 $\hat{F}_{FEA}$ = force field calculated to the FEA model,

2427 $\hat{\Delta}_{FEA}$ = the displacement field of the FEA mesh,

2428 $\hat{M}_{FEA}$ = internal moment field calculated by the FEA model,

2429 and $\hat{\Theta}_{FEA}$ = the rotation field of the FEA mesh.

2430

2431 Starting with the forces calculated from an initial fluid analysis step within the CFD model,
2432 $\hat{F}_{CFD}$, forces and moments which are applied to the free finite element node are computed. Eqn.
2433 4.1 for force and Eqns. 4.2 through 4.4 for moment function as the operations which result in the
2434 reduction of the set of forces represented by $\hat{F}_{CFD}$ to the sets of forces and moments applied to the
2435 FEA model, $\hat{F}_{FEA}$ and $\hat{M}_{FEA}$. These forces and moments are calculated at each timestep and at each
2436 timestep iteration. From these applied forces and moments, the finite element model is integrated
2437 over time to determine the state of the model at the next time increment or at the next timestep
2438 iteration.

2439 After integration of the structural model over a given timestep, internal energies of the FEA
2440 model must sufficiently resolve the applied work in order for momentum and energy from the current
2441 timestep iteration to be conserved throughout the duration of the simulation. Once the OpenSees
2442 model has converged for a given timestep iteration and set of applied forces and moments, the
2443 displacements calculated at the FEM nodes ($\hat{\Delta}_{FEA}$) are projected through Eqn. 4.12 and applied to
2444 the CFD model's coupled surface as boundary conditions to be incorporated as velocities of the
2445 boundary over the timestep ($\hat{\Delta}_{CFD}$).

2446 The components of Eqn. 7.1 within a single timestep are thus all interdependent, which can

roughly be expressed as a function of functions as in Eqn. 7.2.

$$W(t) = f(F(\dot{\Delta}(t)), \Delta(F(t))) \quad (7.2)$$

$\hat{F}_{FEA}$ and $\hat{M}_{FEA}$ are computed from $\hat{F}_{CFD}$ by means of Eqn. 4.1 for force and Eqns. 4.2 through 4.4 for moment, and are integrated over the timestep to be fully resisted by internal and dynamic forces. $\hat{\Delta}_{CFD}$ is directly computed through Eqn. 4.12 via FEA displacement and rotation components $\hat{\Delta}_{FEA}$ and $\hat{\Theta}_{FEA}$ after convergence of the FEA model upon forces and moments $\hat{F}_{FEA}$ and $\hat{M}_{FEA}$.

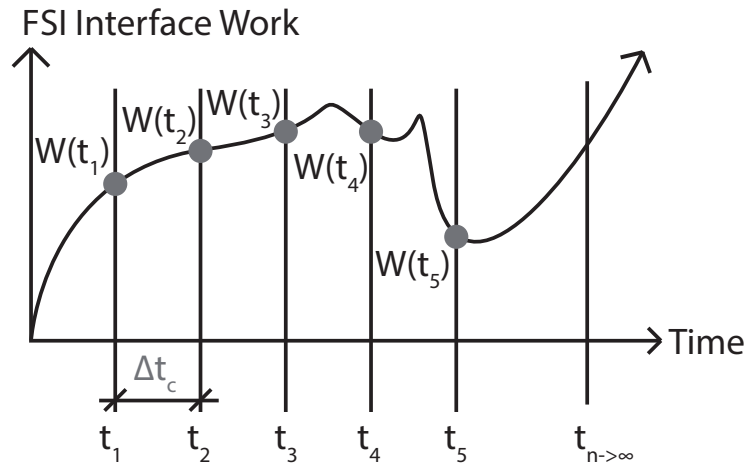


Fig. 7.1. Work Across the Coupling Interface Over Time

The evolution of an arbitrary work function across the interface over time is shown in Fig. 7.1. At the conclusion of each timestep ($W(t_n)$, Fig. 7.1), the change in work from (t^n) to (t^{n+1}) from both the CFD and FEA model should be equivalent in order for both the structural and fluid solutions to have reached the same point along the work function $W(t)$. By choosing a finer time discretization, the discrete changes in work between timesteps can be reduced and the behavior of the work transfer at the interface between coupling timesteps and each participant can be better captured.

As each variable within Eqn. 7.1 is dependent on the other variables within the equation at the current state of the analysis, the satisfaction of work equivalency across the coupling interface as

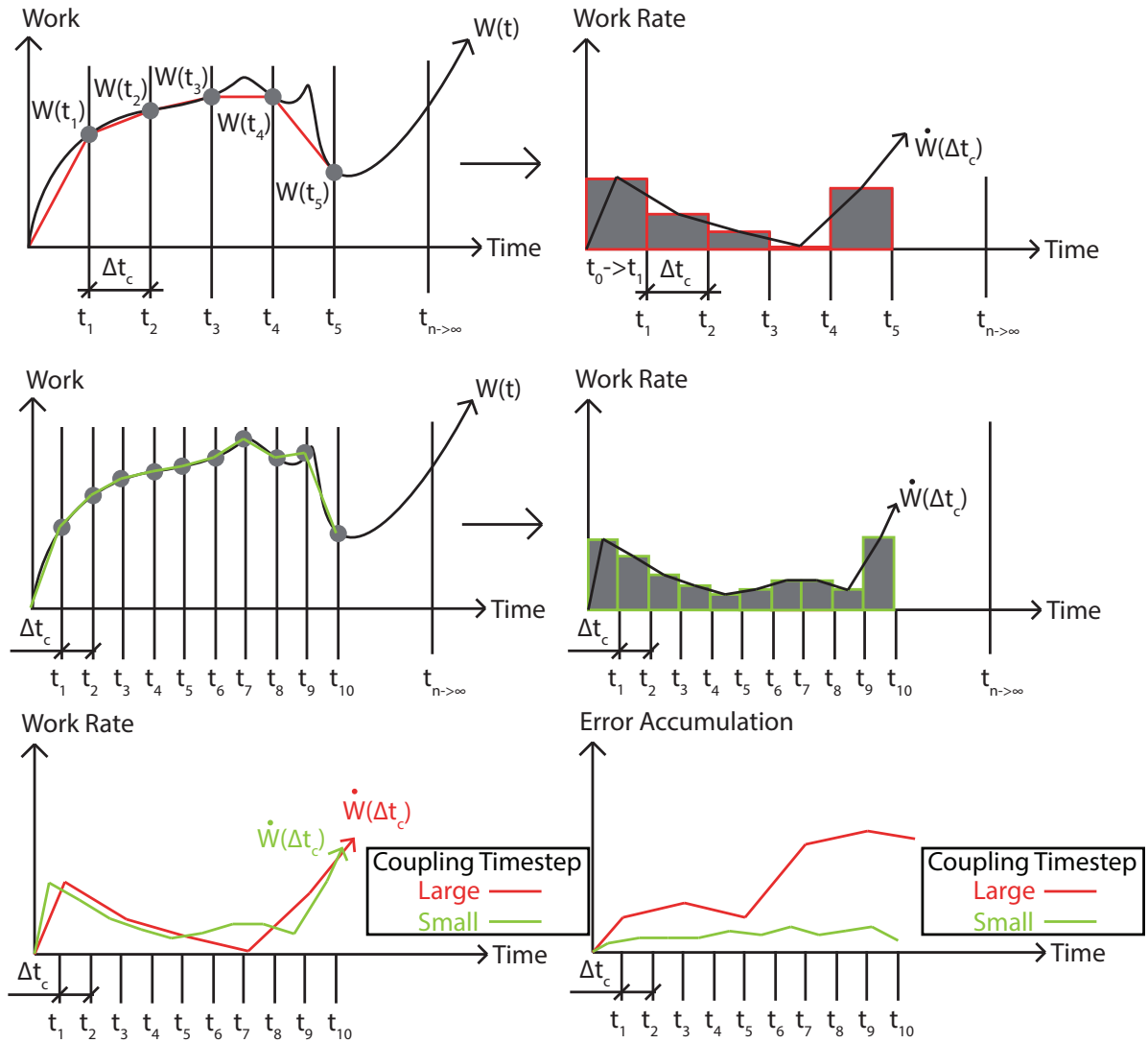


Fig. 7.2. Work Across the Coupling Interface Over Time, Large vs. Small Coupling Timesteps and Timestep Size Effect on Resolution of Work Transfer

described by Eqn. 7.1 must be achieved within each timestep. When timesteps or displacements approach zero and the coupling data is relatively smooth with respect to time, work in and work out are presumed to be equivalent. As displacement increment magnitudes increase, errors in the calculated work from either the left or right hand side of Eqn. 7.1 can increase - when these errors go uncorrected, accumulation of errors can result in a diverging solution which crashes or is not representative of the true dynamics of the coupled problem.

2467 The Walhorn simulation was again selected to demonstrate the effects of the work in and work
2468 out from each component analysis on the stability of the coupled solution with respect to the amount
2469 of work transferred and the amount of errors accumulated. Work into the OpenSees model from the
2470 OpenFOAM forces and out from the OpenSees model displacements to the OpenFOAM model was
2471 calculated and plotted with respect to coupling scheme and timestep in Fig. 7.3. This was done
2472 by calculating the work conducted on the coupling interface mesh by the OpenSees model ('Work
2473 Out') and the work conducted by the coupling interface mesh on the OpenSees finite element mesh
2474 ('Work In') at each timestep. The input and output work curves were normalized with respect to
2475 each other for each simulation and plotted in Fig. 7.4. Work across the coupling interface ('Work In'
2476 versus 'Work Out') is well-preserved by the geometric mapping and force and moment calculation
2477 routines within FOAMySees. Timestep and coupling scheme play a large role in the preservation
2478 of work throughout the simulation and in the errors calculated at various work transfer rates. The
2479 implicit coupling schemes used in this section were not optimized for the problem of interest, and
2480 no preconditioner was utilized on the coupling data.

2481 The error in work transferred across the coupling interface versus time was calculated from the
2482 input and output work for each converged coupling timestep, and plotted in Fig. 7.5. The error in
2483 work transferred across the coupling interface as a function of the work transfer rate is plotted in Fig.
2484 7.6. The results from these simulations imply that as timestep is decreased, errors in work transfer
2485 across the coupling interface are reduced. In addition, implicit coupling schemes ensure that work
2486 within the coupling solution does not grow unbounded due to the effects of numerical instabilities or
2487 errors in work accumulated from previous timesteps.

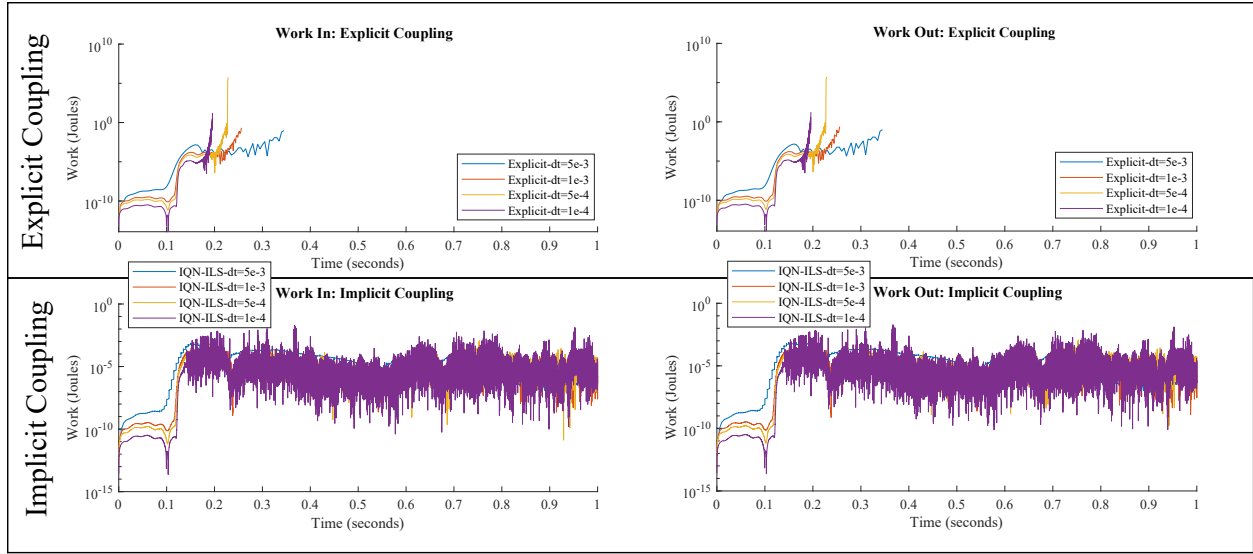


Fig. 7.3. Top: Explicit Scheme, Bottom: Implicit Scheme. Work Across the Coupling Interface Over Time - Left: Work In, Right: Work Out, Walhorn Case

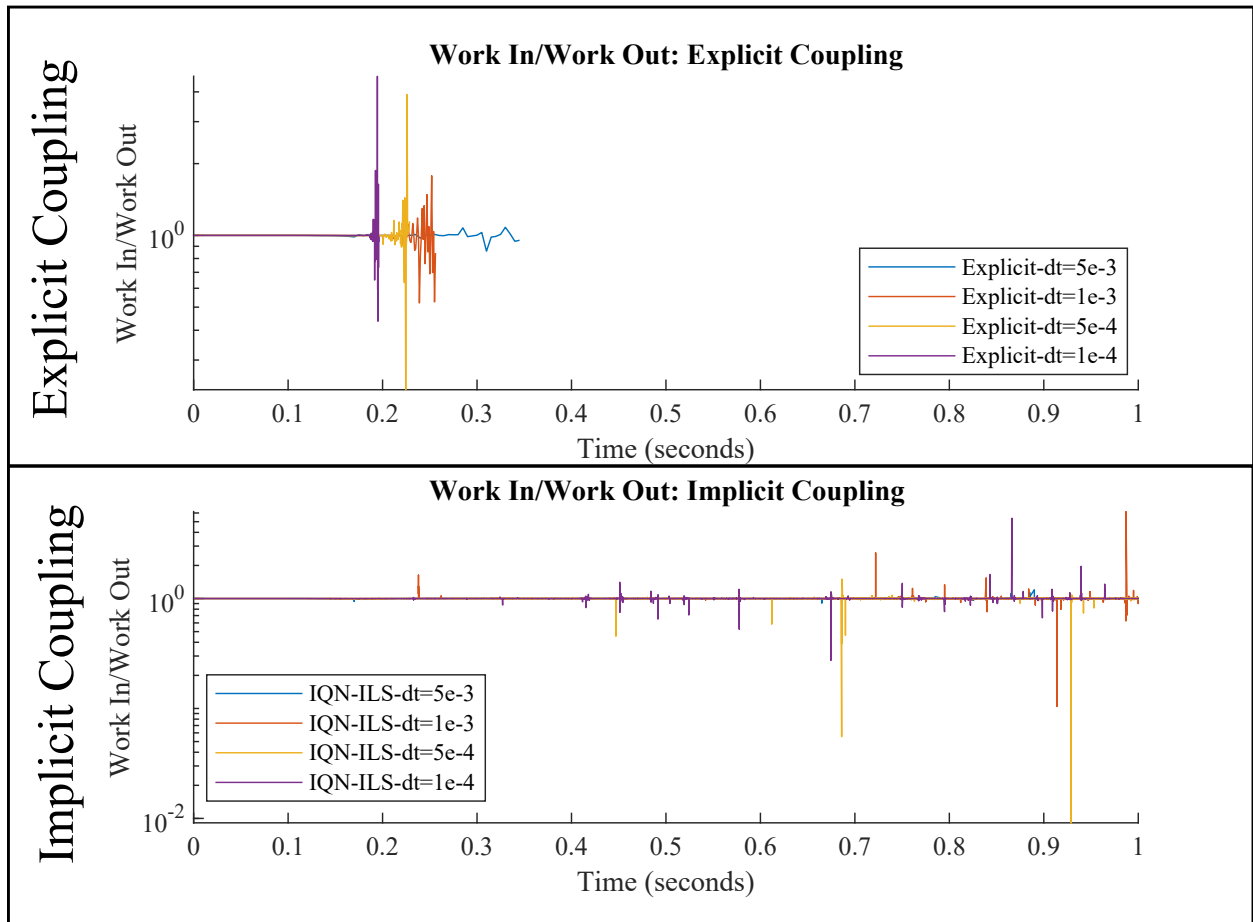


Fig. 7.4. Work Across the Coupling Interface Over Time - Top: Explicit Scheme, Bottom: Implicit Scheme. Work In Over Work Out, Walhorn Case

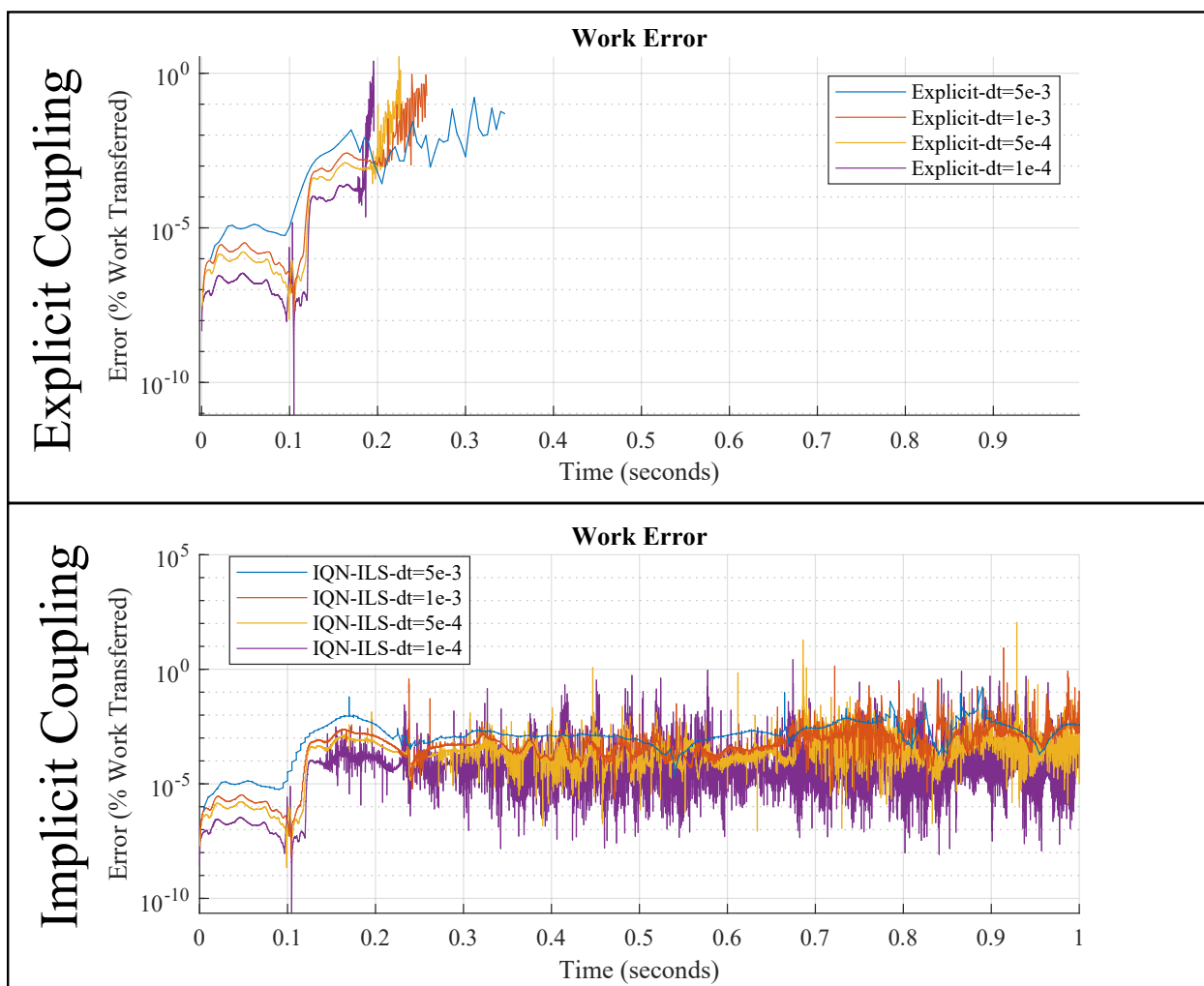


Fig. 7.5. Top: Explicit Scheme, Bottom: Implicit Scheme. Work Resolution Error Over Time, Walhorn Case

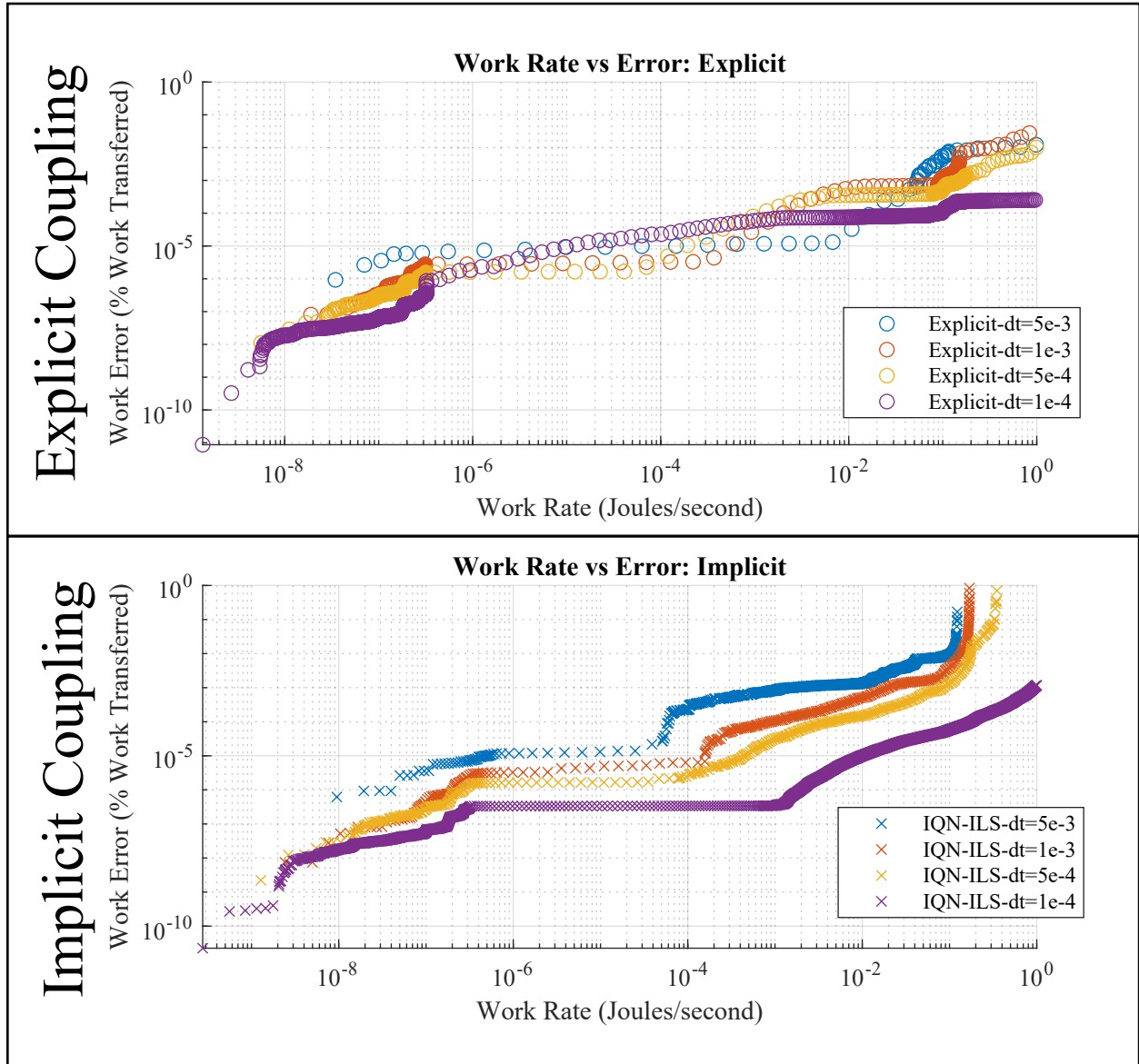


Fig. 7.6. Walhorn Case, Error in Work Across the Coupling Interface as a Function of Work Rate, By Timestep Increment Size and Coupling Scheme - Top: Explicit Scheme, Bottom: Implicit Scheme

Added-Mass-Effect in Partitioned Coupling

To summarize added-mass-effects, incompressibility of a fluid results in a 'stiff' numerical solution on the Poisson pressure equation in the CFD model through which displacements can generate large forces easily. When a fluid is dense these effects are amplified, as the effective inertial fluid forces of an accelerated fluid volume increases with density. Various frequencies of deformation of the structure will have forces associated with them in those frequencies when motions in those frequencies are applied to the CFD solution as displacement boundary conditions. When the fluid and structure are interacting strongly (when fluid inertial forces are near or greater than the inertial forces of a structure combined with the effective resistance, causing fluid-driven or flow-influenced motion) structural motions will generate force fluctuations at those frequencies within the CFD domain, which are then applied back to the FEA model if not appropriately corrected. This can result in large errors in work transfer which subsequently cause simulation divergence.

Small timestep sizes can result in development of high-frequency unstable interactions between partitioned-coupled models without corrections for errors in work calculated by the participant analyses. As more modes of deformation are captured, more frequency modes of this force generated by the fluid's participating added-mass are engaged, increasing the effective added mass of the fluid. If the added-mass of the fluid grows too large with respect to a particular threshold based on the acting structural resistance and mass, the simulation becomes unstable and might diverge. It can be helpful to think of the work from each frequency mode of the data transferred across the coupling mesh as individual parts of the work satisfaction relationship presented in Eqn. 7.1.

Implicit coupling methods are often required to simulate partitioned-coupled fluid-structure-interaction of incompressible fluids acting against highly flexible structures. When utilizing implicit coupling schemes to stabilize simulations with large amounts of participating fluid added-mass, at least one coupling iteration is required per unstable frequency mode to stabilize the interface. If a coupling timestep does not converge, the motion or forces of the unstabilized frequency modes can cause large force spikes within the next coupling timestep which need be corrected in following

timestep iterations. Errors within participant solutions could be resolved by reducing the internal solution convergence tolerances within OpenFOAM and OpenSees (fvSettings and ConvergenceTest, respectively) forcing stricter convergence upon applied boundary conditions, but doing so to attempt to improve stability of explicitly-coupled analyses is futile (i.e., better resolution of the displacement response to spuriously generated forces results in further accumulation in force error).

Coupling timesteps should be sufficiently small such that each component model does not experience numerical damping or diffusion due to the time integration scheme. Additionally, with too large of a coupling timestep, an implicit-partitioned-coupled model may appear to be stable as coupling timesteps can converge prematurely. This is due to both the acceleration techniques utilized by the implicit coupling methodologies utilized as well as the solutions within the component structural analysis and computational fluid dynamics models. Larger time increments means larger increments in coupling data. Larger increments in data will result in smaller variations between guesses from iterations with respect to the rough magnitude of the data increment.

As the coupling convergence tolerance is contingent on the relative convergence of the coupling data with respect to the data from the initial iteration and number of coupling iterations, large increments in coupling data may result in the implicit coupling technique reaching the relative convergence tolerance sooner than it should were it to satisfy the work equivalency requirement across the coupling interface. Due to this, the possibility for errors in the amount of work calculated by the motion of the coupled interface (and thus the amount of errors in energy in the solution) increases with timestep size.

7.1.2 Coupling Settings and Effect on Work Resolved Across Coupling Interface

The use of relaxation factor and its effect on the convergence rate and errors in work across the coupling interface was investigated. The work rate error results for the Walhorn case run at a

coupling time increment size of $5e-4$ s using a range of relaxation factors ($\alpha = 0.1, 0.333, 0.67, 0.9$) are shown in Fig. 7.7. The number of iterations required for convergence for each relaxation factor as a function of time is shown in Fig. 7.8. For this particular case, the relaxation factor has little effect on the convergence of the solution and on the work transfer error for most of range of work rate within the solution. A relaxation factor of 0.67 seems to result in higher amounts of work transfer error than the other relaxation factors - it is not clear why this is the case.

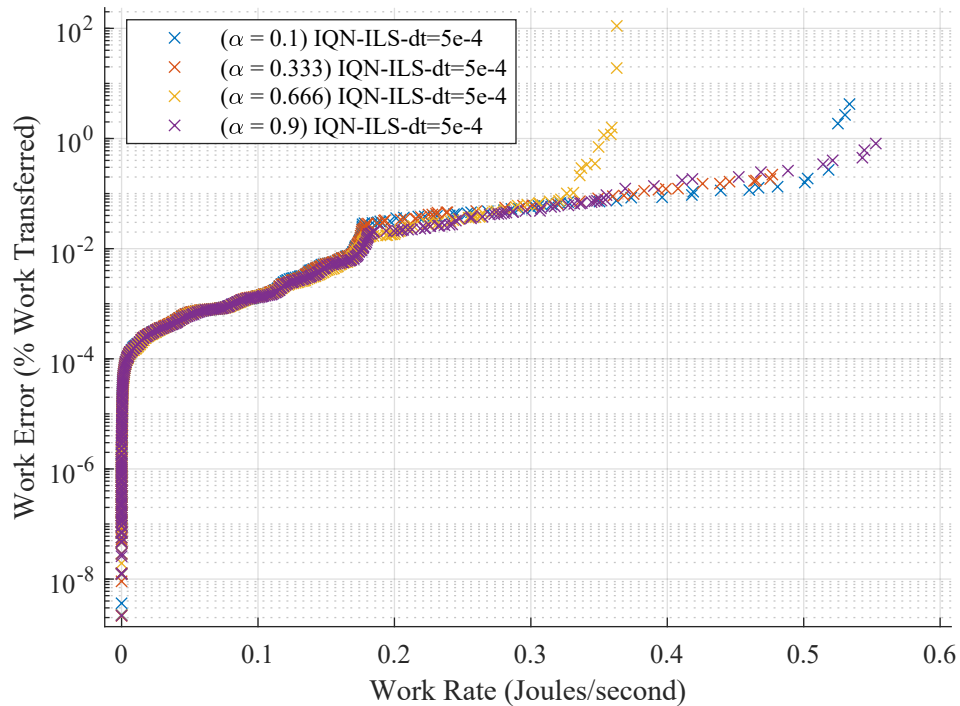


Fig. 7.7. Walhorn Case, Error in Work Across the Coupling Interface as a Function of Work Rate, Effect of Relaxation Factor on Work Transfer Errors

The effect of the use of "residual-sum" preconditioners on the coupling data was also investigated. Preconditioners were applied to three models of varying timestep utilizing the IQN-ILS implicit coupling algorithm. For these cases, a relaxation factor (α) of 0.67 was utilized. A timestep increment of $5e-3$ s, $1e-3$ s, and $5e-4$ s was investigated with and without preconditioners applied. Results from a comparison of work errors by work rate for the six simulations are shown in Fig. 7.9,

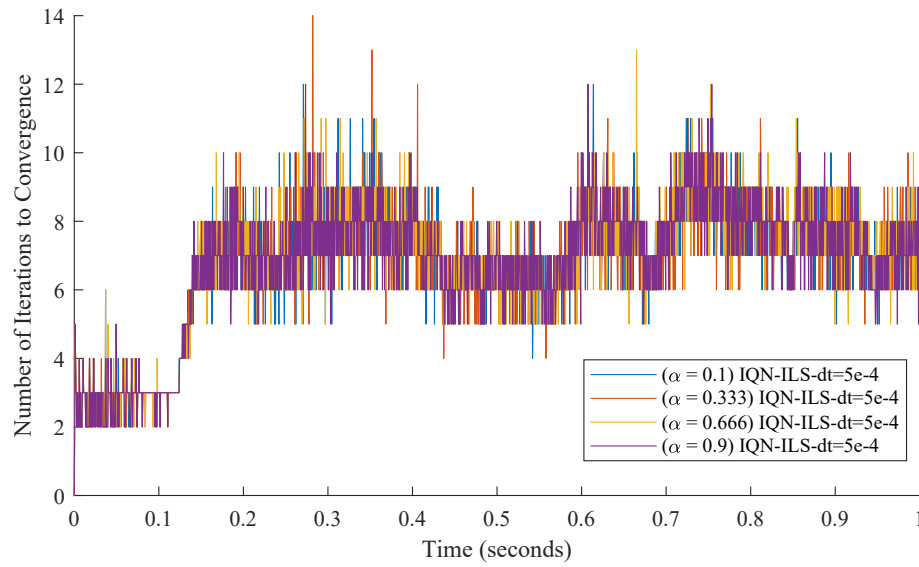


Fig. 7.8. Walhorn Case, Effect of Relaxation Factor (α) on Number of Coupling Iterations Required for Convergence

and the number of coupling iterations required for convergence for each simulation is shown in Fig. 7.10.

The use of the preconditioner resulted in a more stable simulation for a larger timestep, a nearly identically stable simulation for the intermediate timestep size, and a reduction in stability of the simulation for the smallest timestep size. It is not clear why this is the case, but perhaps as the timestep decreases, displacement increments decrease in size and the preconditioner must do more work to attempt to make the coupling data for both displacement and force of roughly the same magnitude. The number of coupling iterations required for the smallest timestep with a preconditioner is much less than that of other timesteps. The quick convergence of this model occurs during a phase of high frequency excitation and large deformation and is it possible that the preconditioner is forcing premature convergence of the solution before the work across the coupling interface is roughly equivalent, but the implicit coupling scheme keeps the work errors bounded such that the simulation does not diverge, even if the premature convergence of coupling timesteps incurs work errors. The effect of preconditioner on simulation stability should be investigated further for

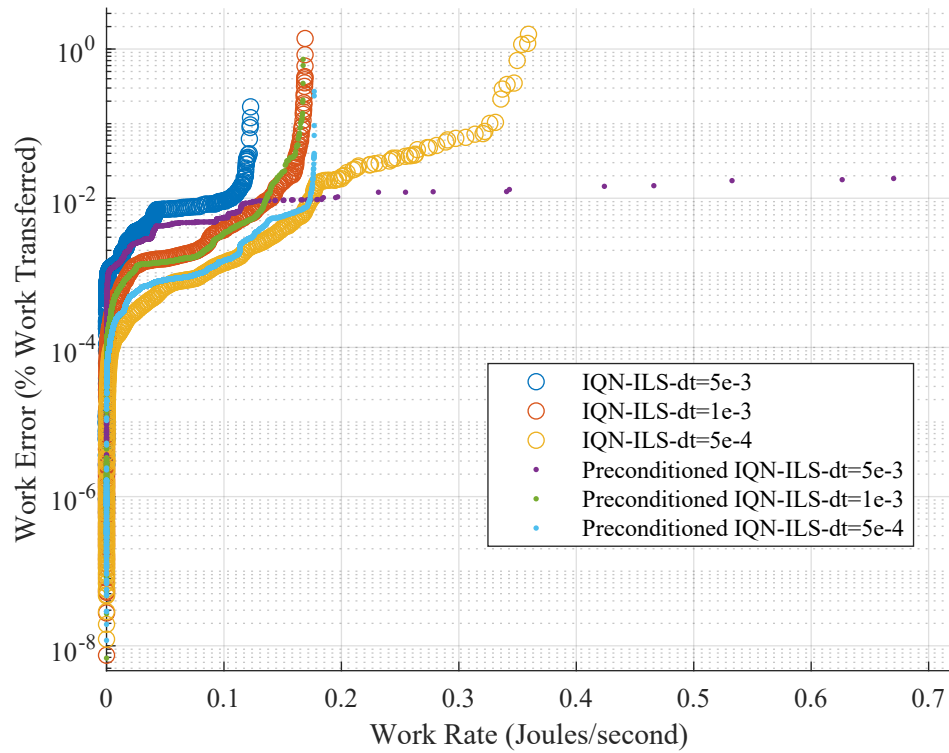


Fig. 7.9. Walhorn Case, Error in Work Across the Coupling Interface as a Function of Work Rate, Effect of Preconditioner on Work Transfer Errors

more timestep sizes and for different relaxation factors, as these may play a significant role in the efficacy of preconditioning of the coupling data. In addition, when conducting simulations with implicit coupling schemes, it is recommended that the work in and out of the solution across the coupling interface from all participants is calculated and assessed to ensure there are not spurious energies or energy lost within the simulation due to coupling errors.

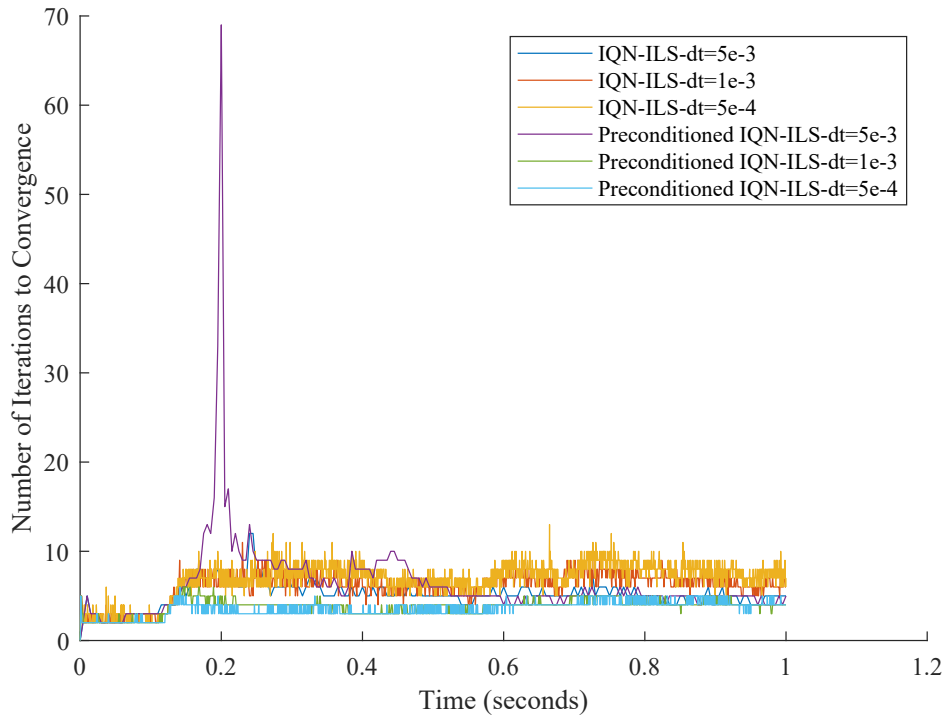


Fig. 7.10. Walhorn Case, Effect of Preconditioner on Number of Coupling Iterations Required for Convergence

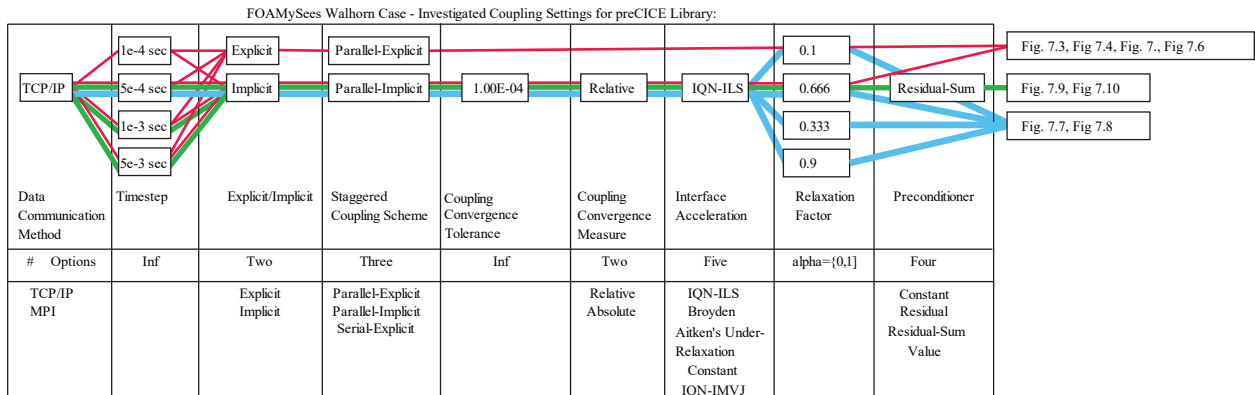


Fig. 7.11. Walhorn Case, Coupling Settings Investigated

7.2 Computational Cost of Walhorn FSI Case

Parallel-staggered coupling is utilized, which means that all coupling participants conduct simulation of the coupling timestep simultaneously. If one participant completes a coupling iteration prior

to other coupling participants, that participant must wait for the other participants to complete their coupling iterations to receive data from the other coupling partners. Upon completion of the coupling timestep iteration by all coupling partners, data is transferred, coupling residuals are checked, and further iterations are completed or the solution is progressed to the next timestep.

Thus, the progression speed of each step of the coupled simulation is controlled by the slowest speed of convergence amongst all coupling participants in addition to the time taken to map data from coupling participant to coupling participant. As direct node-to-node mapping is utilized between the coupling data projection mesh and the OpenFOAM coupling surface, the time taken to map the 1:1 data from coupling data projection mesh to OpenFOAM and back depends on the speed through which preCICE can transfer data.

In addition to this, other thresholds for computation performance may affect simulation speeds, in particular, the I/O speed of the system on which the computations are being performed. TCP/IP connections require transfer of data through use of a connection on the file system. The implicit coupling routines within FOAMySees utilize the '*database*' function of OpenSees, which implements files for the storage of solution variables to be restored upon loading of a database state commit. This operation requires that files be written to disk each coupling timestep, which can limit performance on systems without sufficient I/O resources. The size of the OpenSees model will also have a considerable effect on this component of the computational resources utilized, as database files must be read from disk for each coupling iteration and written to disk for each converged coupling timestep.

It is assumed that a finite element model will possess fewer elements than a computational fluid dynamics model and overall will demand less computational resources than a CFD model within most coupled analyses involving external aerodynamic and hydrodynamic flows. The speed of computation within OpenFOAM is contingent on the amount of memory available to compute solution of the systems of equations and the settings chosen within the OpenFOAM finite volume method solution options, as well as the particular physics of a chosen problem. OpenSees and

Table 7.1. Walhorn Dam Break Case Simulation Speed WRT OpenFOAM Parallelization and Computational Resources

Processors	Machine	Execution Time (s)	(mins)	(hours)
36	UW HYAK Klone	2680	44.67	0.74
18	UW HYAK Klone	3089	51.49	0.86
12	UW HYAK Klone	4001	66.69	1.11
6	UW HYAK Klone	5142	85.69	1.43
2	UW HYAK Klone	12058	200.97	3.35

OpenFOAM can be parallelized across processors to speed up simulation times, but as the utilized OpenSees models are quite simple with respect to the OpenFOAM models, the OpenFOAM model is assumed to control the solution speed of the coupled FSI simulation. The computation speed of the Walhorn dam break case ($\Delta t = 5e-4$ s seconds, IQN-ILS, no preconditioner, relaxation factor of 0.67) was studied across multiple processors on University of Washington's HYAK HPC Cluster to investigate the effects of parallelization of this particular case on the speed of computation. The analysis speeds for the case parallelized on one HPC node are presented in Table 7.1. The OpenFOAM model was comprised of 24144 cells, with a total of 85034 cell faces and 59830 internal faces. The OpenSees model was the same utilized in the previous study, with 40 elastic beam elements. The model performed an average of around 5 iterations per coupling timestep, with a total number of coupling timesteps equal to 2000, resulting in a total approximate number of solution timestep iterations of roughly 10,000. Further work can be done to improve the speed of computation within OpenFOAM, and perhaps to optimize the speed of the FOAMySees coupling code. Approaches for accelerating OpenFOAM simulations are discussed in 8.5.

The use of MPI communication between OpenSees and OpenFOAM through preCICE was not investigated in this thesis, as conflicts between MPI installations utilized within OpenFOAM and OpenSeesPy made implementation of MPI communication between coupled models challenging. MPI communication with preCICE and FOAMySees is beyond the scope of the current thesis, but should be investigated as MPI communication could accelerate the speed of simulations in the case that TCP/IP communication is a bottleneck in the speed of the coupling procedures.

7.3 Limitations Of Coupling Methodology for Analysis of Structures Subject to Tsunami Hazards

While the presented coupling methodology allows for the nonlinear analysis of finite element models of civil engineering structures and may be useful for evaluation of structural response to extreme hydrodynamic forcing, the approach still needs to be validated for assessment of structural resilience to hazards from a tsunami. In addition to this, the methodology does not account for several important forcing-generating phenomena experienced by a structure during a tsunami event. These phenomena include the buoyancy forces of particular building geometries, the effects of soil scour around the base of a structure during a fluid-structure interaction, and the effects of debris impact, accumulation, and subsequent damming force from increased drag and lift area.

7.3.1 Buoyancy Forces

CFD analyses utilizing multi-phase flows can be utilized to capture geometry specific uplift and buoyancy forces (including forces from entrapped air) within geometries which are fully represented within the analysis model. An important consideration is that some CFD analyses with body fitted meshes do not explicitly account for the presence of buoyancy forces for some CFD patch geometries, such as those whose entire boundary is not connected to volumes within the numerical domain. If structures are modelled with their lowermost boundaries (all surface faces with normals slightly toward the ground) contained within the CFD domain, hydrostatic pressure fields in the CFD model will result in calculation of appropriate buoyancy forces through integration of these pressures over the boundary of the structure.

For structures modelled as surfaces whose boundaries do not contain face normals pointing towards the ground within the CFD domain, hydrostatic pressures which would allow for the development of buoyancy forces through integration over the surface of the entire structure will not be accounted for. The omission of CFD calculated buoyancy forces can be rectified by utilizing

approximate methods for the calculation of the buoyancy forces around a structure in post-processing, or as a user defined loading function within FOAMySees.

For example, if the flow elevation is applied as a time dependent boundary condition or is known ahead of time, the flow elevation time history could be utilized to estimate approximate buoyancy forces which could be applied to the FEA model during the FSI analysis. Hydrostatic pressures at the base of a vertically uniform structure when multiplied by a given plan area will offer an approximation for buoyancy forces equal to the weight of the displaced water. If vertical irregularities exist within a structure, this process can be repeated at each floor with components of floor area and sampled or predicted hydrostatic pressures along the height of the structure during the inundation time history.

7.3.2 Scour Effects During Tsunami Inundation

Scour during tsunami inundation can significantly change the strength of the soil around a structure and the overall stability of the structure through reduction of foundation soil constraining pressures through the erosion of soil around the foundation. Soil erosion can lead to the foundation of a structure being unable to support the loads of the structure as-designed, possibly leading to failure. Additional work needs to be done to investigate methods for simulation or prediction of scour effects and associated reductions in strength capacity of a structure when subject to scour. Recent advances have been made to model scour of soils within OpenFOAM (Cheng et al. [2017], Meyer et al. [2021], Nagel et al. [2020]), but whether these approaches can be utilized within a coupled fluid-structure-interaction analysis within the presented methodology for reasonable simulation of scour around flexible structures has not been investigated.

2662 **7.3.3 Debris Impact, Damming**

2663 The proposed coupling methodology between FEA models and CFD models of tsunami flows
2664 offers no supported capability for the simulation of transport of debris by tsunamis and subsequent
2665 debris impact against structures. Simplified debris loading may be applied as a time-dependent
2666 load pattern within FOAMySees during a concurrent fluid-structure-interaction analysis, but the
2667 simulation of transport of debris within a tsunami requires extensions of CFD approaches beyond
2668 OpenFOAM's current capabilities or the utilization of other numerical methods for simulation of
2669 debris-fluid-structure-interaction such as SPH, DEM, PFEM, or MPM. The use of overset CFD
2670 methods for the transport of floating debris with the presented coupling methodology may be
2671 possible, but the use of the tool for these applications has not been investigated. Furthermore, the
2672 overset mesh capabilities of OpenFOAM are computationally expensive, which make their use for
2673 the calculation of large deformations of transported debris within a flow undesirable.

2674 **7.4 Modelling Considerations**

2675 **7.4.1 OpenFOAM Modelling Considerations**

2676 As mentioned previously, the standard OpenFOAM adapter as described by Chourdakis et al.
2677 [2023] is utilized for the coupling of OpenFOAM to the FOAMySees and OpenSees by means
2678 of the preCICE Multiphysics Coupling Library. While this adapter establishes the required data
2679 communication and timestep iteration routines necessary for the execution of a coupled solution
2680 between OpenFOAM and other programs, care must be taken within the OpenFOAM solution when
2681 modelling strongly-coupled fluid-structure-interaction to ensure the stability of the coupled solution
2682 and convergence of the pressure and velocity fields calculated by the CFD model upon the applied
2683 Dirichlet boundary conditions within a given coupling or subcycling timestep. Without a sufficient
2684 number of outer corrector loops within the PIMPLE solution algorithm (a combination of PISO

and SIMPLE algorithms) which is utilized by OpenFOAM, convergence of the CFD model upon the applied displacement and velocity boundary conditions at the coupling interface within a given timestep cannot be ensured (CFD-Online, OpenFOAMwiki). Presently, three outer correctors are utilized within the fvSettings of the OpenFOAM model, as seen in the OpenFOAM Appendix under Listing A.2.

Timestepping of OpenFOAM Solution

Satisfaction of the Courant-Frederics-Lewy criterion for CFD model stability is achieved more readily with smaller timestep sizes, and CFD mesh displacement Laplacian routines can more effectively morph the finite volume domain without incurring diffusion or errors in the CFD solution. Turbulence can also be resolved at higher frequencies with smaller timesteps, and overall pressures and velocities can be captured with better accuracy. More frequency modes of deformation and force are captured by both numerical models with lower timestep sizes resulting in better simulation of high frequency interactions over the coupling interface.

Due to the limitation of the CFL number criterion on the solution, OpenFOAM offers the option to automatically reduce the global timestep to ensure the stability of the finite volume solution. While this is advantageous numerically for the CFD solution, subcycling of the CFD model's timestep within the partitioned-staggered approach utilized by the presented methodology is not supported; waveform interpolation (time interpolation of data) may be possible in newer versions of preCICE, but this was not investigated in this thesis.

With standard OpenFOAM CFD model subcycling, the displacement of the interface is not transferred over all subcycling timesteps within the OpenFOAM model, but is instead applied during the first subcycle. Motion of the coupled interface is then stopped during subsequent subcycling timesteps until the next coupling timestep increment is reached.

This results in erroneous large force spikes during coupled simulations with subcycling implemented, which in turn can result in divergence of the solution. Even with implicit coupling

2710 schemes implemented, these force spikes can go uncorrected and as such it is recommended that a
2711 fixed timestep size equal to the coupling timestep size be used within OpenFOAM during coupled
2712 analyses. Changes could be made in the OpenFOAM coupling adapter or preCICE settings to allow
2713 for time interpolation of the displacement data between CFD timestep subcycles, but this is outside
2714 of the scope of the current work.

2715 For the purpose of two-way coupling with the presented methodology, it is recommended that
2716 timesteps are not reduced automatically by OpenFOAM below the coupling timestep and that
2717 subcycling of the CFD solution is not conducted using OpenFOAM calculated timestep sizes. For
2718 the purpose of one-way coupling with the presented methodology (application of OpenFOAM
2719 calculated forces to the FEM nodes), this is not an issue as Dirichlet boundary conditions are not
2720 applied to the FSI interface in one-way coupling and thus do not generate spurious added-mass-effect
2721 forces.

2722 One possible way to circumvent CFL criterion limitations on timestep size is the use of Local
2723 Time Stepping and time interpolation of coupling data (applying displacements calculated by
2724 OpenSees incrementally over a series of smaller CFD timesteps) or reduction of the timestep of
2725 the coupled solution to a value which ensures satisfaction of the CFL criterion at all points of the
2726 simulation duration. These methods have been shown to produce reasonable results for steady
2727 state solutions of complex flows (Espinoza et al. [2015]). LTS schemes speed up analysis times
2728 significantly, but significant damping of the transient solution can occur as the Local Time Stepping
2729 method solves for the steady state solution of the CFD model. While Local Time Stepping may not
2730 work well for cases such as those presented involving hydrodynamic impact, it likely will provide
2731 meaningful results when running models which are expected to converge on a steady state solution,
2732 which may be useful for cases involving extended flow durations at relatively steady flow rates
2733 and free-surface levels. Local Time Stepping utilizes the standard Euler integration scheme and a
2734 'timestep field' along with additional calculations to allow for stability of a solution at larger global
2735 timesteps when cell sizes are small or velocities are large in particular regions, which is especially
2736 useful for cases involving incompressible flows impacting body fitted meshes around finely resolved

geometries. Local Time Stepping in OpenFOAM can be enabled by replacing the ddtSchemes subdictionary in the fvSchemes file of the OpenFOAM case system folder:

```
ddtSchemes
{
    default          localEuler;
    ddt(phi)         localEuler;
}
```

pointDisplacement Boundary Conditions At and Around the FSI Coupling Interface

Mesh motion of coupled patches within OpenFOAM can be inhibited by boundary conditions applied to the pointDisplacement field of uncoupled CFD patches within the near field of the coupled surface. If points along patches near the coupled interface are constrained, the motion of the mesh near the structure within OpenFOAM could be erroneously restricted by the zero displacement values prescribed to the CFD patch mesh nodes at these locations. Any boundary within the proximity of the coupled interface should be given zeroGradient or fixedNormalSlip boundary conditions for mesh surface point displacement to allow the points near the coupled surface to move along with the displacement of the structure and to preclude artificial stiffening of the CFD mesh.

7.4.2 OpenSees Modelling Considerations

Parallelization of OpenSeesPy Models

Sometimes MPI versioning or implementation can differ between the OpenFOAM installation and the OpenSees installation, making initialization of parallelized OpenSees models difficult or impossible. In order to utilize parallelized OpenSees models within the current approach on HPC systems, the OpenSees analysis must be initialized on a different computing node than the OpenFOAM model to which it is coupled. Communication between the models is maintained

through the TCP/IP connection, and this precludes conflicts between the installations of MPI utilized by the OpenSees model and the OpenFOAM model.

Structural Model Timestep

A timestep sufficiently small enough to meet all stability criteria for the numerical integration routines of both structural and fluid solvers should be chosen as the timestep for both the exchange of coupling data and individual solver time progression. The Newmark method with coefficients of γ and β valued 0.5 and 0.25 respectively was utilized for time integration of the structural model. As such, the maximum timestep increment for the structural dynamics simulation depends on structural material wave propagation and minimum element eigenvalue timestep requirements. Furthermore, work input to and output from the OpenSees model should be checked using FOAMySees (this is possible through setting the correct options in the coupling driver).

Transfer of Full Force Between OpenFOAM and OpenSees During Coupling Iterations

Unresolved forces within the OpenSees model should be near or equal to zero for a given coupling timestep in order to ensure all force from fluid loading has been effectively transferred to the structural analysis model. Convergence of an OpenSees model at a given state upon the applied loads can be ensured with proper selection of the ConvergenceTest object type within OpenSees and can be checked with OpenSees output. All OpenSees models examined in this thesis were run with either a 'NormUnbalance' or 'EnergyUnbalance' convergence test, typically with a convergence tolerance of 1e-8.

Depending on the convergence test specified within OpenSees, timesteps of the coupled solution may not progress at all - thus it is imperative that an appropriate convergence measure is selected for the OpenSees model. Reduction of unbalanced applied forces can be ensured by lowering the coupling timestep (and thus the force increment between timesteps) and lowering the structural analysis subcycling timestep to sufficiently capture the structural response to the impulse generated

by the forces applied over a given timestep. The effect of the OpenSees model convergence test type and tolerance has not been investigated as part of this work, but may play a considerable role in the convergence of the structural analysis model when subject to the forces calculated by the CFD model that it is coupled to.

Caution is advised when relaxing the convergence test measures within OpenSees, as making the convergence test less strict may result in higher variability of the structural response within a given timestep between coupling iterations, slower overall convergence, and accumulation of errors across the coupling interface. As a general rule, it is recommended that internal solver tolerances are kept at $1/100^{th}$ of the coupling convergence tolerance defined for an implicit coupling scheme. Making the convergence test too strict may in turn result in a model which will never converge upon the forces applied to it within the timestep regardless of the number of structural analysis model subcycling timesteps are completed for the given loading.

Use of Elements or Materials without sendSelf or recvSelf Methods Implemented

Some element and material formulations within OpenSees cannot be utilized as part of an OpenSees database, due to their lack of functionality in the sendSelf and recvSelf methods of the objects. Before running an implicitly-coupled analysis with a model, try running the the following operation after constructing the model in OpenSeesPy. This writes and reads a checkpoint file, which will crash if there are components of the OpenSees model which do not support usage in a database.

```
ops.database('File',"checkpoint")
ops.save(0)
ops.restore(0)
```

Any element formulation or material model which does not support database usage cannot be used in an implicitly-coupled analysis through the current capabilities of FOAMySees unless the sendSelf and recvSelf methods are implemented for the OpenSees object. This does not restrict their

2808 use in the presented approach when utilizing an explicit coupling scheme, but any object which does
2809 not support storage of state values in an OpenSees database will need to be modified to allow for
2810 these capabilities to be used in an implicitly-coupled analysis or replaced by an object which does.
2811 The 'sendSelf' and 'recvSelf' methods take the internal data from the object of question to send
2812 through the OpenSees broker objects to be stored within a Database object. Some of the elements,
2813 materials, and objects in OpenSees don't have these methods implemented, and can't be used as part
2814 of a database until the sendSelf and recvSelf methods have been implemented for the given object.
2815 As the database method is utilized in implicit coupling routines within FOAMySees, these objects,
2816 elements, materials, etc can only be used in the context of an explicitly-coupled analysis with the
2817 presented methodology, unless the objects' sendSelf and recvSelf methods are implemented in the
2818 OpenSees framework (Scott [2020], Scott [2022]).

Chapter 8

Future Work

The presented coupling methodology opens possibilities of several investigations which may be of interest to structural engineers and researchers of computational nonlinear fluid-structure-interaction. This chapter discusses several topics of interest which may be addressed with the FOAMySees research tool, as well as where the approach could be improved in the future.

8.1 Deterministic Simulation of Probabilistic Combined Hazard Scenarios to Existing Civil Engineering Structures for Assessment of Structural Resilience

Cascading seismic-tsunami hazard effects are current topics of study and are considered to be highly subjective to geographic location, local topography, soil and site conditions, and structural properties (Reis et al. [2022b]). The effects of damage from seismic excitation on the residual capacity of typical structures under tsunami inundation are largely unknown but likely depend on structural design, soil conditions, and individual hazard intensities at the site. Simulation of source-fault tsunami generation, propagation, and subsequent coastal inundation tsunami modelling is possible

to assess the characteristics of a tsunami which is associated with a particular expected ground motion (Reis et al. [2021]). Numerical modelling of sequential tsunami propagation and inundation utilizing coupled finite volume methods and smoothed-particle-hydrodynamics models has been conducted with success (Reis) to achieve a deterministic simulation of tsunami source generation and propagation to site inundation through coupled shallow water equation-FVM/Navier-Stokes SPH modelling.

Similar deterministic cascading hazard analyses were conducted on models of a coastal roadway by Román-de la Sancha et al. [2022], demonstrating an approach to assess resilience of structures to a specific multi-hazard scenario for validation of probabilistic methodologies. Experimentally validated coupling procedures have been conducted with OpenFOAM and nonlinear potential flow solvers Mata et al. [2023]. The use of complicated tsunami source and propagation models could therefore also possibly be coupled to FOAMySees for generation of appropriate tsunami flows as a boundary condition within a coupled fluid-structure-interaction analysis, allowing for a meaningful deterministic simulation of particular hazards.

Since CFD simulations are prohibitively computationally expensive for large domains, topography-specific FOAMySees coupled analyses could be created in accordance to the three load cases with the tsunami acting in either direction as prescribed by ASCE 7 Chapter 6 instead of modelling the full tsunami inundation cycles (6 shorter FSI analyses instead of full time history of inundation). These analyses could be used to investigate the demands to components with respect to the loading prescribed by design procedures and be utilized to assess resulting structural element capacities. Investigating full structural system response to these load cases would be possible with a FOAMySees numerical model of the structure, with boundary and initial conditions representing the load case of interest.

8.2 Investigation of Effects of Structural Motion or Flexibility of Structures on Imparted Flow Forces

Coupled eulerian-lagrangian numerical analyses conducted on wood-frame residential structures subject to wave loading indicate that flexibility of the structure in a fluid-structure-interaction analysis can result in forces which are up to 40 percent larger than those calculated using a rigid body approximation of the structural motion (Kotzamanis and Kalliontzis [2023]). It is not clear if similar effects exist for structures of taller heights or of differing construction material, but it is expected that as the mass or stiffness of the structure is increased such effects are attenuated.

For tall, slender structures or structures with long spans such as bridges or highways, small changes in rotation at connections can result in large displacements of components of the structure. A wide bridge with a long span could experience twisting of the bridge deck along the axis of its span, changing the position of the leading edge of the bridge deck structure within a flow transverse to the bridge span and the angle of inclination of the deck with respect to the flow. Structures in these conditions function almost as hydrofoils - the pressures generated by the flow underneath the structure during the flooding stage result in large uplift forces along the length of the bridge span, possibly resulting in torsional and flexural bending of the deck.

Changes in deck inclination and slope can further affect the forces generated through changes in flow-normal cross-sectional area of the overall superstructure, the orientation of flow-bearing surfaces, and the dynamic motion of the structure. Furthermore, structural nonlinearity and changes in resistance can result in increased motion during softening and possible reapplication of dynamic pressure loading against a structure upon re-establishment of a sufficient load resisting configuration to cease these motions. Whether such motions have an effect on imparted dynamic fluid forces during high velocity flows is not well understood, but forces generated within compensating structural elements upon failure of particular elements may have a considerable impact on the resilience of a structure to sustained loading.

8.3 FOAMySees As an Approach For Numerical-Experimental Validation Studies and Investigations of Systems at Larger Scales

Experimentation and testing of structures at full scale for performance-based-design purposes is not only difficult and prohibitively expensive for most researchers, but highly infeasible in the case of generation of realistic tsunami flows and loading. Furthermore, extrapolation of experimental studies conducted at small scales to design scales requires thorough consideration of the laws of scaling of the components of force associated with the physics of the problem, which may be challenging or impossible to achieve for certain situations. For example, experimental results involving high wind and wave loading against flexible structures simultaneously may be challenging to scale to larger lengths due to the varying length scale factors required for viscous fluid effects (Reynold's Number scaling) and inertial fluid effects (Froude Number scaling). Alternative to scaling experiments utilizing nondimensional parameters, measured values from experiments can be cross-validated with preliminary simulations, and simulations and data therefrom can be used to advance the CFD models by comparing the experimental and numerical results. Once validated, the model can provide data that could not be measured to advance the understanding of the demands and system response. Coupled numerical models created of such experiments could then be scaled to examine the behavior of the interactions at full design scales, and possibly inform about scaling laws and effects and magnitudes of various forces at certain length scales.

2901 **8.4 Study of Use of Coupling Approach for Other Structural** 2902 **Engineering Applications**

2903 OpenFOAM is a modular software library containing CFD solvers for multitudes of applications.
2904 Many of the applications (external aerodynamic simulation, thermal heat transfer, fire simulation,
2905 compressible flows, blast simulation) might be useful to structural engineers if incorporated as part
2906 of a coupled multi-physics simulation to assess structural response. Coupling of an OpenFOAM
2907 simulation to OpenSees is achieved quite readily through the use of the preCICE Adapter function
2908 object within OpenFOAM, meaning it could be fairly simple to couple a CFD model which has been
2909 built within OpenFOAM for other reasons than hydrodynamic simulation to a structural analysis
2910 model to assess structural resilience to the boundary forces calculated by the OpenFOAM model.
2911 This opens the possibility of utilizing FOAMySees for wind and gust analysis of structures, blast
2912 resilience analysis, and other applications pertinent in nonlinear structural analysis and structural
2913 resilience. For example: Temperatures can be passed from OpenFOAM to preCICE, and (with
2914 modification of FOAMySees) could be applied to and retrieved from OpenSees for conjugate heat
2915 transfer analysis of structures.

2916 **8.5 Implementation of Methodologies for Acceleration of Com-** 2917 **putational Speeds**

2918 The size and complexity of models required for finite volume simulations of tsunami inundation
2919 demand extreme amounts of computational resources using traditional computing techniques. Further
2920 work is being done to investigate methods for utilization of OpenFOAM for simulations at the
2921 'exascale' level of computing, implementing computing resources from clusters of processing units
2922 for massively-parallelized simulations (exaFOAM project). The use of computational acceleration
2923 techniques through graphical processing units (GPUs) offers a promising technique for reducing the

required memory and simulation time to complete finite volume simulations of large domains Qin [2019]. With the technology of GPUs progressing at a rate much faster than that of traditional CPUs, it follows that these advancements in computational capabilities can be utilized for acceleration of simulation speeds for complicated models of physical phenomena (Porter and Krishnamoorthy).

GPU-based parallelization techniques for acceleration of linear algebra calculations within OpenFOAM have been recently developed, which indicate promising results for the acceleration of simulation times of calculation of external aerodynamic flows, particularly for solutions which require large numbers of pressure Poisson solution steps (Jaiswal et al. [2016], Ghioldi and Piscaglia [2023], Zampini et al. [2020]). Implementation of external libraries such as amgx4Foam (Piscaglia and Ghioldi [2023]) or PETSc4FOAM (Bnà et al. [2020]) for the acceleration of OpenFOAM simulations may ease the adoption of strongly-coupled fluid-structure-interaction of tsunami-like flows about flexible structure with the presented methodology by offering faster simulation times to analysts. Furthermore, utilization of single and mixed-precision calculations rather than double precision is a new capability in OpenFOAM which may offer significant improvements in computational speed while utilizing hybrid CPU-GPU resources for parallelization of CFD simulations (Brogi et al. [2022]).

Chapter 9

Conclusions

The presented work offers a versatile, open-source methodology for numerical simulation of nonlinear fluid-structure-interaction between CFD and FEA models using validated softwares OpenFOAM, OpenSees, and preCICE. By allowing for geometric discrepancies between structural and fluid solvers at the coupling interface and compensating for this through the use of an intermediate interpolation mesh, high-resolution reduced order center-line element and shell element comprised finite element models may be utilized to simulate the response of structures to fluid loading in three dimensions.

The approach was verified through several static analysis cases, one high-energy dam break case involving a flexible rubber flap placed within the path of flow, and two large-scale cases representing a 1:6 scale structures comprised of steel and concrete subject to breaking solitary wave impingement tested previously at Oregon State University. The high energy dam break case was run with seismic excitation preceding the fluid-structure interaction simulation to demonstrate kinetic and damage state coherence between termination of the preliminary structural analysis subroutines and the initial state of the fluid-structure analysis.

Simulation execution times for coupled analyses involving large CFD meshes and fast incompressible flows can be prohibitively massive. The use of larger implicit coupling timesteps, along with

substepping and time-interpolation of applied Dirichlet boundary conditions within OpenFOAM and variable timestepping within OpenSees, can ease computational difficulties, preserve memory resources, and allow for simulations to run exponentially faster than unmodified CFD models using implicit timestepping schemes with smaller timesteps to maintain CFL stability criterion. However, additional work should be done investigating the efficacy of methods for accelerating OpenFOAM simulation execution times and assessing accuracy of these methods for use in simulation of dynamic incompressible multi-phase flows. Despite the demonstrated usability for FSI purposes, more testing of the tool and coupling methodology is necessary to ensure that the tool will work in cases not examined in this thesis. Determining rate of convergence of solution with respect to the coupling settings chosen could be useful to quantify, though this will likely depend on the problem of interest which is simulated as well as the computational stability and quality of the sub-models of the coupled solution.

Investigations into memory usage of the approach should be completed, as there may be components of the current approach that could be improved. While implementing the coupling technique in another programming language may reduce the flexibility of the approach, specific problems could possibly be modelled in C++ with OpenSees objects and coupled to OpenFOAM via C++ function calls to the preCICE coupling library. Many possibilities exist that cannot be further studied within the current scope of work, and other researchers are encouraged to investigate how this approach could be better implemented to improve computational times.

Additionally, timestep stability regimes of the presented coupled solution technique with respect to limiting factors within both the finite element and computational fluid dynamics methodologies could be investigated. Outlining best practices for modelling civil engineering structures at design-scales within the present methodology with respect to OpenFOAM, OpenSees, and implicit and explicit partitioned coupling schemes could be useful to engineers looking to utilize this approach for evaluation of structures to natural hazards. Finally, more work must be done to validate the presented solution technique with experimental studies conducted at large scales. A detailed investigation

2984 of how turbulence modelling affects results was not considered in the present study, thus more
2985 research must be completed to assess the accuracy of results provided by FOAMySees when utilizing
2986 OpenFOAM fluid solvers which incorporate turbulence modelling.

Bibliography

- Antonio Agüero Ramón Llin, Carmen Izvernari, and Robert Tremblay. 2007. Modeling of the seismic response of concentrically braced steel frames using the opensees analysis environment. A SIMPLIFIED METHOD FOR SEISMIC PERFORMANCE EVALUATION OF STEEL BRIDGE PIERS WITH THIN-WALLED STIFFENED BOX SECTIONS, 2:242.
- Taofiq Al-Faesly, Dan Palermo, Ioan Nistor, and Andrew Cornett. 2012. Impact of tsunami forces on structures: The university of ottawa experience. volume 32.
- Mohammad S. Alam, Andrew O. Winter, Glen Galant, Krishnendu Shekhar, Andre R. Barbosa, Michael R. Motley, Marc O. Eberhard, Daniel T. Cox, Pedro Arduino, and Pedro Lomonaco. 2020. Tsunami-like wave-induced lateral and uplift pressures and forces on an elevated coastal structure. Journal of Waterway, Port, Coastal, and Ocean Engineering, 146(4):04020008.
- Mohammad Shafiqul Alam. 2019. Probabilistic multi-hazard earthquake-tsunami structural risk assessment framework.
- Anne Archibald. 2008. Scipy spatial package, kd-tree method.
- ASCE. 2022. Minimum Design Loads and Associated Criteria for Buildings and Other Structures.
- Navid Attary, John W. Van De Lindt, Andre R. Barbosa, Daniel T. Cox, and Vipin U. Unnikrishnan. 2021. Performance-based tsunami engineering for risk assessment of structures subjected to multi-hazards: Tsunami following earthquake. Journal of Earthquake Engineering, 25(10):2065–2084.

- 3005 Navid Attary, Vipin U. Unnikrishnan, John W. van de Lindt, Daniel T. Cox, and Andre R. Barbosa. 2017.
 3006 Performance-based tsunami engineering methodology for risk assessment of structures. Engineering
 3007 Structures, 141:676–686.
- 3008 Brian Atwater, Musumi-Rokkaku Satoko, Kenji Satake, Tsuji Yoshinobu, Ueda Kazue, and David Yamaguchi.
 3009 2005. The orphan tsunami of 1700.
- 3010 Marco Baiguera, Tiziana Rossetto, Ian Robertson, and Crescenzo Petrone. 2019. A nonlinear static procedure
 3011 for the tsunami design of a reinforced concrete building to the asce7 standard.
- 3012 Marco Baiguera, Tiziana Rossetto, Ian N. Robertson, and Crescenzo Petrone. 2022. A procedure for
 3013 performing nonlinear pushover analysis for tsunami loading to asce 7. Journal of Structural Engineering,
 3014 148(2):04021270.
- 3015 K.J. Bathe. 2006. Finite Element Procedures. Prentice Hall.
- 3016 Jörn Behrens, Finn Løvholt, Fatemeh Jalayer, Stefano Lorito, Mario A. Salgado-Gálvez, Mathilde Sørensen,
 3017 Stephane Abadie, Ignacio Aguirre-Ayerbe, Iñigo Aniel-Quiroga, Andrey Babeyko, Marco Baiguera,
 3018 Roberto Basili, Stefano Belliazzi, Anita Grezio, Kendra Johnson, Shane Murphy, Raphaël Paris, Irina
 3019 Rafliana, Raffaele De Risi, Tiziana Rossetto, Jacopo Selva, Matteo Taroni, Marta Del Zoppo, Alberto
 3020 Armigliato, Vladimír Bureš, Pavel Cech, Claudia Cecioni, Paul Christodoulides, Gareth Davies, Frédéric
 3021 Dias, Hafize Başak Bayraktar, Mauricio González, Maria Gritsevich, Serge Guillas, Carl Bonnevie Harbitz,
 3022 Utku Kânoğlu, Jorge Macías, Gerassimos A. Papadopoulos, Jascha Polet, Fabrizio Romano, Amos Salamon,
 3023 Antonio Scala, Mislav Stepinac, David R. Tappin, Hong Kie Thio, Roberto Tonini, Ioanna Triantafyllou,
 3024 Thomas Ulrich, Elisa Varini, Manuela Volpe, and Eduardo Vyhmeister. 2021. Probabilistic tsunami hazard
 3025 and risk analysis: A review of research gaps. Frontiers in Earth Science, 9.
- 3026 Guanbo Bian, David Padilla-Llano, Stephen Buonopane, Cristopher Moen, and Benjamin Schafer. 2015.
 3027 Opensees modeling of wood sheathed cold-formed steel framed shear walls. Structural Stability Research
 3028 Council Annual Stability Conference 2015, SSRC 2015, pages 219–232.

3029 Simone Bnà, Ivan Spisso, M.H. Olesen, and Giacomo Rossi. 2020. Petsc4foam: a library to plug-in petsc
3030 into the openfoam framework.

3031 J. Bonus, P. Arduino, M. Motley, and M. O. Eberhard. 2022. Debris-fluid-structure interaction: Numerical
3032 approach with validation.

3033 Federico Brogi, Simone Bnà, Gabriele Boga, Giorgio Amati, Tomaso Esposti Ongaro, and Matteo Cerminara.
3034 2022. On floating point precision in computational fluid dynamics using openfoam.

3035 E. H. van Brummelen. 2009. Added Mass Effects of Compressible and Incompressible Flows in Fluid-Structure
3036 Interaction. Journal of Applied Mechanics, 76(2):021206.

3037 Hans-Joachim Bungartz, Florian Lindner, Bernhard Gatzhammer, Miriam Mehl, Klaudius Scheufele,
3038 Alexander Shukaev, and Benjamin Uekermann. 2016. precice – a fully parallel library for multi-physics
3039 surface coupling. Computers and Fluids, 141:250–258. Advances in Fluid-Structure Interaction.

3040 Patrick O. Burns, Andre R. Barbosa, Michael J. Olsen, and Haizhong Wang. 2021. Multihazard damage and
3041 loss assessment of bridges in a highway network subjected to earthquake and tsunami hazards. Natural
3042 Hazards Review, 22(2):05021002.

3043 preCICE C++ API Documentation. a. solverinterface class. Accessed Sept, 2023.

3044 preCICE C++ API Documentation. b. solverinterface class getdataid method. Accessed Sept, 2023.

3045 preCICE C++ API Documentation. c. solverinterface class getmeshid method. Accessed Sept, 2023.

3046 preCICE C++ API Documentation. d. solverinterface class initialize method. Accessed Sept, 2023.

3047 preCICE C++ API Documentation. e. solverinterface class initializedata method. Accessed Sept, 2023.

3048 preCICE C++ API Documentation. f. solverinterface class isactionrequired method. Accessed Sept, 2023.

3049 preCICE C++ API Documentation. g. solverinterface class iscouplingongoing method. Accessed Sept, 2023.

3050 preCICE C++ API Documentation. h. solverinterface class markactionfulfilled method. Accessed Sept, 2023.

3051 preCICE C++ API Documentation. i. solverinterface class preciceadvance method. Accessed Sept, 2023.

3052 preCICE C++ API Documentation. j. solverinterface class readblockvectordata method. Accessed Sept, 2023.

3053 preCICE C++ API Documentation. k. solverinterface class readblockvectordata method. Accessed Sept,
3054 2023.

3055 preCICE C++ API Documentation. l. solverinterface class setmeshvertices method. Accessed Sept, 2023.

3056 preCICE C++ API Documentation. m. (version 2.5.0). Accessed Sept, 2023.

3057 Suncica Canic. 2020. Moving boundary problems. Bulletin of the American Mathematical Society, 58:1.

3058 Lyle Carden, G. Chock, Ian Robertson, and G. Yu. 2012. Structural analysis case studies of buildings damaged
3059 during the tohoku tsunami.

3060 Lyle Carden, Guangren Yu, G. Chock, and Ian Robertson. 2016. Tsunami-resilient building designs for hawaii
3061 and other high hazard regions.

3062 Philip Cardiff. 2012. Development of the Finite Volume Method for Hip Joint Stress Analysis. Ph.D. thesis.

3063 P. Causin, J.F. Gerbeau, and F. Nobile. 2005. Added-mass effect in the design of partitioned algorithms for
3064 fluid–structure problems. Computer Methods in Applied Mechanics and Engineering, 194(42):4506–4527.

3065 CFD-Online. Openfoam pimple outer correctors for fsi. Accessed Sept, 2023.

3066 Ingrid Charvet, Anawat Suppasri, H Kimura, Daisuke Sugawara, and Fumihiko Imamura. 2015. A multivariate
3067 generalized linear tsunami fragility model for kesennuma city based on maximum flow depths, velocities
3068 and debris impact, with evaluation of predictive accuracy. Natural Hazards, 79.

3069 Zhen Cheng, Tian-Jian Hsu, and Joseph Calantoni. 2017. Sedfoam: A multi-dimensional eulerian two-phase
3070 model for sediment transport and its application to momentary bed failure. Coastal Engineering, 119:32–50.

3071 G. Chock, Lyle Carden, Ian Robertson, Michael Olsen, and Guangren Yu. 2013. Tohoku tsunami-induced
3072 building failure analysis with implications for usa tsunami and seismic design codes. Earthquake Spectra,
3073 135:021601–1.

3074 Gary Chock. 2015. Chapter 21 - the asce 7 tsunami loads and effects design standard for the united states. In
 3075 Miguel Esteban, Hiroshi Takagi, and Tomoya Shibayama, editors, Handbook of Coastal Disaster Mitigation
 3076 for Engineers and Planners, pages 437–460. Butterworth-Heinemann, Boston.

3077 Gary Chock, Guangren Yu, Hong Kie Thio, and Patrick J. Lynett. 2016. Target structural reliability analysis for
 3078 tsunami hydrodynamic loads of the asce 7 standard. Journal of Structural Engineering, 142(11):04016092.

3079 Gary Y. K. Chock. 2016. Design for tsunami loads and effects in the asce 7-16 standard. Journal of Structural
 3080 Engineering, 142(11):04016093.

3081 Gary Y. K. Chock, Lyle Carden, Ian Robertson, Yong Wei, Rick Wilson, and John Hooper. 2018. Tsunami-
 3082 resilient building design considerations for coastal communities of washington, oregon, and california.
 3083 Journal of Structural Engineering, 144(8):04018116.

3084 A.K. Chopra. 2017. Dynamics of Structures: Theory and Applications to Earthquake Engineering. Always
 3085 learning. Pearson.

3086 G Chourdakis, K Davis, B Rodenberg, M Schulte, F Simonis, B Uekermann, G Abrams, HJ Bungartz,
 3087 L Cheung Yau, I Desai, K Eder, R Hertrich, F Lindner, A Rusch, D Sashko, D Schneider, A Totounferoush,
 3088 D Volland, P Vollmer, and OZ Koseomur. 2022. preCICE v2: A sustainable and user-friendly coupling
 3089 library [version 2; peer review: 2 approved]. Open Research Europe, 2(51).

3090 Gerasimos Chourdakis, David Schneider, and Benjamin Uekermann. 2023. Openfoam-precice: Coupling
 3091 openfoam with external solvers for multi-physics simulations. OpenFOAM® Journal, 3:1–25.

3092 C. Cortez, R. Jünemann, C. Fernández, A. Urrutia, J.G.F. Crempien, and R. Cienfuegos. 2022. Performance
 3093 of an rc building under seismic and tsunami actions in sequence via nonlinear dynamic analysis including
 3094 soil-structure interaction. Engineering Structures, 272:114942.

3095 Sergio Croquer, Pilar Diaz-Carrasco, Sébastien Poncet, Jay Lacey, and Ioan Nistor. 2022. Numerical
 3096 benchmark for modelling wave impacts against vertical walls using openfoam.

3097 R. Cross. 1967. Tsunami surge forces. Journal of the Waterways and Harbors Division, 93:201–234.

3098 Hai Dang, Sungwon Shin, Eunju Lee, Hyoungsu Park, and Jun-Nyeong Park. 2022. Numerical investigation
 3099 of countermeasure effects on overland flow hydrodynamic and force mitigation in coastal communities.
 3100 Journal of Ocean Engineering and Technology, 36.

3101 Joris Degroote. 2013. Partitioned simulation of fluid-structure interaction. Archives of Computational
 3102 Methods in Engineering, 20.

3103 Joris Degroote, Sebastiaan Annerel, and Jan Vierendeels. 2010a. Stability analysis of gauss-seidel iterations
 3104 in a partitioned simulation of fluid-structure interaction. Computers Structures - COMPUT STRUCT, 88.

3105 Joris Degroote, Klaus-Jürgen Bathe, and Jan Vierendeels. 2009. Performance of a new partitioned procedure
 3106 versus a monolithic procedure in fluid–structure interaction. Computers & Structures, 87(11-12):793–801.

3107 Joris Degroote, Robby Haelterman, Sebastiaan Annerel, Peter Bruggeman, and Jan Vierendeels. 2010b.
 3108 Performance of partitioned procedures in fluid–structure interaction. Computers Structures, 88(7):446–457.

3109 Joris Degroote, Antonio Souto-Iglesias, Wim Van Paepegem, Sebastiaan Annerel, Peter Bruggeman, and Jan
 3110 Vierendeels. 2010c. Partitioned simulation of the interaction between an elastic structure and free surface
 3111 flow. Computer Methods in Applied Mechanics and Engineering, 199(33):2085–2098.

3112 Joris Degroote and Jan Vierendeels. 2011. A partitioned quasi-newton solution technique for fluid-structure
 3113 interaction problems using a coarsened grid to accelerate the convergence of the coupling iterations. In
 3114 COMPUTATIONAL METHODS FOR COUPLED PROBLEMS IN SCIENCE AND ENGINEERING
 3115 IV, pages 100–111. European Community on Computational Methods in Applied Sciences (ECCOMAS).

3116 Marta Del Zoppo, Tiziana Rossetto, M. Di Ludovico, and Andrea Prota. 2020. Assessing the effect of
 3117 tsunami-induced vertical loads on rc frames.

3118 Marta Del Zoppo, Tiziana Rossetto, M. Di Ludovico, and Andrea Prota. 2023. Effect of buoyancy loads
 3119 on the tsunami fragility of existing reinforced concrete frames including consideration of blow-out slabs.
 3120 Scientific Reports, 13.

3121 Nicolas Delaissé, Toon Demeester, Rob Haelterman, and Joris Degroote. 2023. Quasi-newton methods
 3122 for partitioned simulation of fluid–structure interaction reviewed in the generalized broyden framework.
 3123 Archives of Computational Methods in Engineering.

3124 Steven Douglas and Ioan Nistor. 2014. On the effect of bed condition on the development of tsunami-induced
 3125 loading on structures using openfoam. Natural Hazards, 76.

3126 Steven Douglas, Ioan Nistor, and Phillippe St-Germain. 2015. 3-d multi-phase numerical modelling of
 3127 tsunami-induced hydrodynamic loading on nearshore structures.

3128 Nuri Elsheikh, Amir Hossein Azimi, Ioan Nistor, and Abdolmajid Mohammadian. 2022a. Tsunami-induced
 3129 bores propagating over a canal, part ii: Numerical experiments using the standard k- turbulence model.
 3130 Fluids, 7:214.

3131 Nuri Elsheikh, Ioan Nistor, Amir Hossein Azimi, and Abdolmajid Mohammadian. 2022b. Tsunami-induced
 3132 bore propagating over a canal—part 1: Laboratory experiments and numerical validation. Fluids, 7:213.

3133 Daniel Espinoza, Thomas Scanlon, and Richard Brown. 2015. Validation of tools to accelerate high-speed
 3134 cfd simulations using openfoam.

3135 C. Farhat and M. Lesoinne. 2000. Two efficient staggered algorithms for the serial and parallel solution of
 3136 three-dimensional nonlinear transient aeroelastic problems. Computer Methods in Applied Mechanics
 3137 and Engineering, 182(3):499–515.

3138 FEMA. 2017. Hazus tsunami model technical guidance.

3139 FEMA. 2019. P646 Guidelines for Design of Structures for Vertical Evacuation from Tsunamis.

3140 A.S.J. Foster, T. Rossetto, and W. Allsop. 2017a. An experimentally validated approach for evaluating tsunami
 3141 inundation forces on rectangular buildings. Coastal Engineering, 128:44–57.

3142 A.S.J. Foster, T. Rossetto, and W. Allsop. 2017b. An experimentally validated approach for evaluating tsunami
 3143 inundation forces on rectangular buildings. Coastal Engineering, 128:44–57.

- 3144 André Furtado, Hugo Rodrigues, and António Arêde. 2015. Modelling of masonry infill walls participation
3145 in the seismic behaviour of rc buildings using opensees. International Journal of Advanced Structural
3146 Engineering.
- 3147 Eric Geist and Patrick Lynett. 2014. Source processes for the probabilistic assessment of tsunami hazards.
3148 Oceanography, 27:86–93.
- 3149 Eric Geist and Tom Parsons. 2006. Probabilistic analysis of tsunami hazards*. Natural Hazards, 37:277–314.
- 3150 F. Ghioldi and F. Piscaglia. 2023. Acceleration of supersonic/hypersonic reactive cfd simulations via
3151 heterogeneous cpu-gpu supercomputing. Computers Fluids, 266:106041.
- 3152 Cassidy J. Gills. 2018. Experimental and numerical validation of three-dimensional tsunami wave pressures
3153 and forces on an elevated structure.
- 3154 Katsuichiro Goda and Raffaele De Risi. 2023. Future perspectives of earthquake-tsunami catastrophe modelling:
3155 From single-hazards to cascading and compounding multi-hazards. Frontiers in Built Environment, 8.
- 3156 Catalina González-Dueñas and Jamie E. Padgett. 2021. Performance-based coastal engineering framework.
3157 Frontiers in Built Environment, 7.
- 3158 OpenFOAM User Guide. blockmesh. Accessed Sept, 2023.
- 3159 Charles R. Harris, K. Jarrod Millman, Stéfan J. van der Walt, Ralf Gommers, Pauli Virtanen, David
3160 Cournapeau, Eric Wieser, Julian Taylor, Sebastian Berg, Nathaniel J. Smith, Robert Kern, Matti Pícus,
3161 Stephan Hoyer, Marten H. van Kerkwijk, Matthew Brett, Allan Haldane, Jaime Fernández del Río, Mark
3162 Wiebe, Pearu Peterson, Pierre Gérard-Marchant, Kevin Sheppard, Tyler Reddy, Warren Weckesser, Hameer
3163 Abbasi, Christoph Gohlke, and Travis E. Oliphant. 2020. Array programming with NumPy. Nature,
3164 585(7825):357–362.
- 3165 Anis Hasanpour and Denis Istrati. 2022. Extreme storm wave impact on elevated coastal buildings.
- 3166 J Heintz and M Mahoney. 2008. Guidelines for design of structures for vertical evacuation from tsunamis.

3167 Pablo Higuera. 2018. olaflow: Cfd for waves.

3168 Charles Hirsch. 2007. Numerical computation of internal and external flows (second edition). In Charles
3169 Hirsch, editor, Numerical Computation of Internal and External Flows (Second Edition), second edition
3170 edition. Butterworth-Heinemann, Oxford.

3171 C.W Hirt and B.D Nichols. 1981. Volume of fluid (vof) method for the dynamics of free boundaries. Journal
3172 of Computational Physics, 39(1):201–225.

3173 C.W. Hirt and B.D. (1981) Volume of Fluid (VOF) Method for the Dynamics of Free Boundaries. Journal of
3174 Computational Physics Nichols. 2001. J. Engrg. Mech., 127(10):987–993.

3175 IBM. Tcp/ip. Accessed August 2023.

3176 S.R. Idelsohn, J. Marti, A. Limache, and E. Oñate. 2008. Unified lagrangian formulation for elastic solids and
3177 incompressible fluids: Application to fluid–structure interaction problems via the pfem. Computer Methods
3178 in Applied Mechanics and Engineering, 197(19):1762–1776. Computational Methods in Fluid–Structure
3179 Interaction.

3180 Kenshiro Ishiki, Lee S. Cunningham, and Benedict D. Rogers. 2023. Existing design approaches to nuclear
3181 power plants subject to tsunamis: A critical review. Structures, 57:105109.

3182 Daigoro Isobe and Seizo Tanaka. 2021. Sequential simulations of steel frame buildings under multi-phase
3183 hazardous loads during earthquake and tsunami. Frontiers in Built Environment, 7.

3184 Denis Istrati and Ian Buckle. 2019. Role of trapped air on the tsunami-induced transient loads and response
3185 of coastal bridges. Geosciences, 9:191.

3186 Denis Istrati, Ian Buckle, Ahmad Itani, Pedro Lomonaco, and Solomon Yim. 2017. Large-scale fsi experiments
3187 on tsunami-induced forces in bridges.

3188 Shashank Jaiswal, Rajesh Reddy, R. Banerjee, Shingo Sato, Daisuke Komagata, Makoto Ando, and Jun
3189 Okada. 2016. An efficient gpu parallelization for arbitrary collocated polyhedral finite volume grids and its
3190 application to incompressible fluid flows.

3191 Hrvoje Jasak. 2009. Dynamic mesh handling in openfoam.

3192 Hrvoje Jasak and Zeljko Tukovic. 2006. Automatic mesh motion for the unstructured finite volume method.
 3193 Transactions of FAMENA, 30:1–20.

3194 Kristijan Kolozvari, Kutay Orakcal, and John Wallace. 2017. New opensees models for simulating nonlinear
 3195 flexural and coupled shear-flexural behavior of rc walls and columns. Computers Structures, 196.

3196 Vasileios Kotzamanis and Dimitrios Kalliontzis. 2023. Understanding flexibility effects in the interaction of
 3197 light-frame wood structures with wave action. Journal of Structural Engineering, pages 04023187–1.

3198 Ulrich Küttler and Wolfgang Wall. 2008. Fixed-point fluid-structure interaction solvers with dynamic
 3199 relaxation. Computational Mechanics, 43:61–72.

3200 Randall J. LeVeque. 2002. Finite Volume Methods for Hyperbolic Problems. Cambridge Texts in Applied
 3201 Mathematics. Cambridge University Press.

3202 Nicolette S. Lewis. 2023. Foamysees coupling code repository.

3203 Nicolette S. Lewis, Dawn E. Lehman, Michael R. Motley, Pedro Arduino, Charles W. Roeder, Christopher N.
 3204 Pyke, and Kenneth P. Sullivan. 2022. Integrated study of existing tsunami design standards. Journal of
 3205 Structural Engineering, 148(12):04022200.

3206 Nicolette S. Lewis, Andrew O. Winter, Justin Bonus, Michael R. Motley, Marc O. Eberhard, Pedro Arduino,
 3207 and Dawn E. Lehman. 2023. Open-source simulation of strongly-coupled fluid-structure interaction
 3208 between non-conformal interfaces. Frontiers in Built Environment, 9.

3209 Florian Lindner. 2019. Data transfer in partitioned multi-physics simulations: interpolation & communication.

3210 David Linton, Rakesh Gupta, Dan Cox, John van de Lindt, Mary Elizabeth Oshnack, and Milo Clauson.
 3211 2013. Evaluation of tsunami loads on wood-frame walls at full scale. Journal of Structural Engineering,
 3212 139(8):1318–1325.

3213 Shilong Liu, Ioan Nistor, Abdolmajid Mohammadian, and Amir Hossein Azimi. 2022. Experimental and
 3214 numerical investigation of beach slope effects on the hydrodynamic loading of tsunami-like surges on a
 3215 vertical wall. Journal of Marine Science and Engineering, 10:1580.

3216 S. Lloyd. 1982. Least squares quantization in pcm. IEEE Transactions on Information Theory, 28(2):129–137.

3217 Patrick Lynett. 2009. Tsunami inundation, modeling of. pages 117–133.

3218 Joshua Macabuag, Alison Raby, Antonios Pomonis, Ioan Nistor, Sean Wilkinson, and Tiziana Rossetto. 2018.
 3219 Tsunami design procedures for engineered buildings. Proceedings of the Institution of Civil Engineers -
 3220 Civil Engineering, 171:1–60.

3221 Per A. Madsen, David R. Fuhrman, and Hemming A. Schäffer. 2008. On the solitary wave paradigm for
 3222 tsunamis. Journal of Geophysical Research: Oceans, 113(C12).

3223 Songrit Maneewongvatana and David M. Mount. 1999. Analysis of approximate nearest neighbor searching
 3224 with clustered point sets.

3225 Julio Marcelo Marti, Sergio Idelsohn, Alejandro Limache, Nestor Calvo, and Jorge D’Elía. 2006. A fully
 3226 coupled particle method for quasi-incompressible fluid-hypoelastic structure interactions.

3227 Marisol Mata, Stef Boersen, Marcel Gent, Alessandro Antonini, Bjarne Jensen, and Cock van der Lem.
 3228 2023. A validation of wave loads on crest walls on top of composite breakwaters using openfoam. Coastal
 3229 Engineering Proceedings, page 23.

3230 Patrick W McDonald. 1971. The computation of transonic flow through two-dimensional gas turbine cascades,
 3231 volume 79825. American Society of Mechanical Engineers.

3232 Frank McKenna, Michael H Scott, and Gregory L Fenves. 2010. Nonlinear finite-element analysis software
 3233 architecture using object composition. Journal of Computing in Civil Engineering.

3234 John E. Melton. 1989. An integrated cfd/experimental analysis of aerodynamic forces and moments: August
 3235 1989 nasa ames research center moffett field, ca.

3236 M. Menter, F.R.. Kuntz and R. Langtry. 2003a. Ten years of industrial experience with the sst turbulence
 3237 model. In Proc., 8th Annual Int. Bridge Conf., page 625–632, Antalya, Turkey. the fourth international
 3238 symposium on turbulence, heat and mass transfer, Begell House.

3239 M. Menter, F.R.. Kuntz and R. Langtry. 2003b. Ten years of industrial experience with the sst turbulence
 3240 model. page 625–632.

3241 Janek Meyer, Kai Graf, and Thomas Slawig. 2021. Simulation of scour around arbitrary offshore foundations
 3242 based on the volume-of-fluid method combined with a bingham model.

3243 Daniel P Mok and WA Wall. 2001. Partitioned analysis schemes for the transient interaction of incompressible
 3244 flows and nonlinear flexible structures. Trends in computational structural mechanics, 1.

3245 J.R. Morison, J.W. Johnson, and S.A. Schaaf. 1950. The force exerted by surface waves on piles. Journal of
 3246 Petroleum Technology, 2(05):149–154.

3247 Michael R. Motley, Hin Kei Wong, Xinsheng Qin, Andrew O. Winter, and Marc O. Eberhard. 2016.
 3248 Tsunami-induced forces on skewed bridges. Journal of Waterway, Port, Coastal, and Ocean Engineering,
 3249 142(3).

3250 Tim Nagel, Julien Chauchat, Cyrille Bonamy, Xiaofeng Liu, Zhen Cheng, and Tian-Jian Hsu. 2020. Three-
 3251 dimensional scour simulations with a two-phase flow model. Advances in Water Resources, 138:103544.

3252 Christopher Neumann, Barbara Simpson, Andreas Schellenberg, Pedro Lomonaco, and Shawn You. 2023.
 3253 Hydrodynamic real-time hybrid simulation demonstrated for cascading seismic and tsunami events. Journal
 3254 of Waterway, Port, Coastal, and Ocean Engineering, 149(1):04022029.

3255 OpenFOAMwiki. Pimple algorithm: Openfoam user guide. Accessed Sept, 2023.

3256 OpenSees. Documentation(a). Analyze command. Accessed Aug, 2023.

3257 OpenSees. Documentation(b). Database class. Accessed Aug, 2023.

3258 OpenSees. Documentation(c). Database class save method. Accessed Aug, 2023.

3259 OpenSees. Documentation(d). Nodedisp command. Accessed Aug, 2023.

3260 OpenSees. Documentation(e). The open system for earthquake engineering simulation. Accessed Aug, 2023.

3261 OpenSees. Documentation(f). Remove command. Accessed Aug, 2023.

3262 OpenSees. Documentation(g). Restore command. Accessed Aug, 2023.

3263 Hyounghsu Park, Mohammad S. Alam, Daniel T. Cox, Andre R. Barbosa, and John W. van de Lindt. 2019.

3264 Probabilistic seismic and tsunami damage analysis (pstda) of the cascadia subduction zone applied to

3265 seaside, oregon. International Journal of Disaster Risk Reduction, 35:101076.

3266 Sangki Park, John W. van de Lindt, Daniel Cox, and Rakesh Gupta. 2013. Concept of community fragilities

3267 for tsunami coastal inundation studies. Natural Hazards Review, 14(4):220–228.

3268 Sangki Park, John W. van de Lindt, Daniel Cox, Rakesh Gupta, and Francisco Aguiniga. 2012. Successive

3269 earthquake-tsunami analysis to develop collapse fragilities. Journal of Earthquake Engineering, 16(6):851–

3270 863.

3271 Mark Petersen, Morgan Moschetti, Peter Powers, Charles Mueller, Kathleen Haller, Arthur Frankel, Yuehua

3272 Zeng, Sanaz Rezaeian, Stephen Harmsen, Oliver Boyd, Ned Field, Rui Chen, Kenneth Rukstales, Nico

3273 Luco, Russell Wheeler, Robert Williams, and Anna Olsen. 2014. Documentation for the 2014 update of

3274 the united states national seismic hazard maps. USGS Open-File Report 2014–1091.

3275 Crescenzo Petrone, Tiziana Rossetto, Marco Baiguera, Camilo De la Barra Bustamante, and Ioanna Ioannou.

3276 2020. Fragility functions for a reinforced concrete structure subjected to earthquake and tsunami in

3277 sequence. Engineering Structures, 205:110120.

3278 Crescenzo Petrone, Tiziana Rossetto, and Katsuichiro Goda. 2017. Fragility assessment of a rc structure

3279 under tsunami actions via nonlinear static and dynamic analyses. Engineering Structures, 136:36–53.

3280 Federico Piscaglia and Federico Ghioldi. 2023. Gpu acceleration of cfd simulations in openfoam. Aerospace,

3281 10(9).

3282 Chris Porter and Niveditha Krishnamoorthy. The computational fluid dynamics revolution driven by gpu accel-
 3283 eration. [https://developer.nvidia.com/blog/computational-fluid-dynamics-](https://developer.nvidia.com/blog/computational-fluid-dynamics-revolution-driven-by-gpu-acceleration/)
 3284 [revolution-driven-by-gpu-acceleration/](https://developer.nvidia.com/blog/computational-fluid-dynamics-revolution-driven-by-gpu-acceleration/). Accessed: 2023-10-12.

3285 preCICE. 2022a. Openfoam coupling adapter (for precice version 2.5.0).

3286 preCICE. 2022b. (version 2.5.0). Accessed August 2023.

3287 exaFOAM project. exafoam project for exascale computing with openfoam. <https://exafoam.eu/>.
 3288 Accessed: 2023-10-12.

3289 Christopher N. Pyke. 2020. Testing of a reinforced concrete core wall for tsunami vertical evacuation shelter
 3290 structures in a wave flume.

3291 preCICE Python Bindings. Version 2.5.0. Accessed Sept, 2023.

3292 Xinsheng Qin. 2019. Efficient Tsunami Simulation at Local and Global Scales. Ph.D. thesis.

3293 Xinsheng Qin, Michael Motley, Randall Leveque, Frank Gonzalez, and Kaspar Mueller. 2018a. A comparison
 3294 of a two-dimensional depth averaged flow model and a three-dimensional rans model for predicting tsunami
 3295 inundation and fluid forces. Natural Hazards and Earth System Sciences, 18.

3296 Xinsheng Qin, Michael R. Motley, and Nasser A. Marafi. 2018b. Three-dimensional modeling of tsunami
 3297 forces on coastal communities. Coastal Engineering, 140:43–59.

3298 Cláudia Reis. Pile-Supported Quay Submitted to Successive Earthquake and Tsunami Actions, pages 32–43.

3299 Cláudia Reis, Maria Ana Baptista, Mário Lopes, Carlos Sousa Oliveira, and Stéphane Clain. 2022a.
 3300 Cascade earthquake and tsunami hazard assessment: A deterministic perspective for engineering purposes.
 3301 International Journal of Disaster Risk Reduction, 75:102952.

3302 Cláudia Reis, Stéphane Clain, Jorge Figueiredo, André R. Barbosa, Maria Ana Baptista, and Mário Lopes.
 3303 2021. Experimentally validated numerical models to assess tsunami hydrodynamic force on an elevated
 3304 structure. Engineering Structures, 249:113280.

3305 Cláudia Reis, Mário Lopes, Maria Ana Baptista, and Stéphane Clain. 2022b. Towards an integrated framework
 3306 for the risk assessment of coastal structures exposed to earthquake and tsunami hazards. Resilient Cities
 3307 and Structures, 1(2):57–75.

3308 I. N. Robertson, K. Paczkowski, H. R. Riggs, and A. Mohamed. 2013. Experimental Investigation of Tsunami
 3309 Bore Forces on Vertical Walls. Journal of Offshore Mechanics and Arctic Engineering, 135(2):021601.

3310 Ian Robertson. 2014. Development of tsunami loading expressions for design of coastal infrastructure. volume
 3311 102.

3312 Ian Robertson and Abdulla Mohamed. 2009. Development of performance based tsunami engineering, pbte.
 3313 IABSE Symposium Report, pages 16–23.

3314 I.N. Robertson. 2023. Recent advances in tsunami design of coastal structures. Coastal Engineering
 3315 Proceedings, page 83.

3316 Tiziana Rossetto, Camilo Barra, Crescenzo Petrone, Juan De la Llera, Jorge Vásquez G., and Marco Baiguera.
 3317 2019. Comparative assessment of nonlinear static and dynamic methods for analysing building response
 3318 under sequential earthquake and tsunami. Earthquake Engineering Structural Dynamics, 48.

3319 Tiziana Rossetto, Crescenzo Petrone, Ian Eames, Camilo Barra, Andrew Foster, and Joshua Macabuag. 2018.
 3320 Advances in the Assessment of Buildings Subjected to Earthquakes and Tsunami, pages 545–562.

3321 Matthew Rueben, Robert Holman, D. Cox, Sungwon Shin, Jason Killian, and J. Stanley. 2011. Optical
 3322 measurements of tsunami inundation through an urban waterfront modeled in a large-scale laboratory
 3323 basin. Coastal Engineering, 58:229–238.

3324 A. Román-de la Sancha, R. Silva, O. S. Areu-Rangel, M. G. Verduzco-Zapata, E. Mendoza, N. P. López-Acosta,
 3325 A. Ossa, and S. García. 2022. Modelling the sequential earthquake–tsunami response of coastal road
 3326 embankment infrastructure. Natural Hazards and Earth System Sciences, 22(8):2589–2609.

3327 Samieh Sarjamee, Ioan Nistor, and Abdolmajid Mohammadian. 2017a. Large eddy simulation of extreme
 3328 hydrodynamic forces on structures with mitigation walls using openfoam. Natural Hazards, 85.

3329 Samieh Sarjamee, Ioan Nistor, and Abdolmajid Mohammadian. 2017b. Numerical investigation of the
 3330 influence of extreme hydrodynamic forces on the geometry of structures using openfoam. Natural Hazards,
 3331 87.

3332 Klaudius Scheufele. 2015. Robust Quasi-Newton Methods for Partitioned Fluid-Structure Simulations. Ph.D.
 3333 thesis.

3334 Michael Scott. 2020. Opensees sendself and recvself.

3335 Michael Scott. 2022. An opensees special request (sendself, recvself).

3336 Betsy R. Seiffert, R. Cengiz Ertekin, and Ian N. Robertson. 2015. Wave loads on a coastal bridge deck and
 3337 the role of entrapped air. Applied Ocean Research, 53:91–106.

3338 Seyedreza Shafiei, Bruce W. Melville, and Asaad Y. Shamseldin. 2016. Experimental investigation of tsunami
 3339 bore impact force and pressure on a square prism. Coastal Engineering, 110:1–16.

3340 Alexander K. Shukaev. 2015. A fully parallel process-to-process intercommunication technique for precice.

3341 Barbara Simpson. 2022. Real-time hybrid experimental-numerical simulation of bridge infrastructure
 3342 subject to cascading earthquake-tsunami hazards. In Pacific Northwest Transportation Consortium [204].
 3343 University of Washington ResearchWorks Archive.

3344 Max Stephens, Zhonghou Xu, Colin Whittaker, and Liam Wotherspoon. 2023. Vulnerability of power
 3345 distribution utility poles to tsunami bore impacts. 3:22.

3346 Bane Sullivan and Alexander Kaszynski. 2019. PyVista: 3D plotting and mesh analysis through a streamlined
 3347 interface for the Visualization Toolkit (VTK). Journal of Open Source Software, 4(37):1450.

3348 Kenneth P. Sullivan. 2021. Testing of a new composite framing system for vertical evacuation structures.

3349 D.T. Swigler. 2009. Laboratory study investigating the three-dimensional turbulence and kinematic properties
 3350 associated with a breaking solitary wave.

3351 Amin Totounferoush, Frédéric Simonis, Benjamin Uekermann, and Miriam Schulte. 2021. Efficient and
 3352 scalable initialization of partitioned coupled simulations with precice. Algorithms, 14(6):166.

3353 M. J. Turner, R. W. Clough, H. C. Martin, and L. J. Topp. 1956. Stiffness and deflection analysis of complex
 3354 structures. Journal of the Aeronautical Sciences, 23(9):805–823.

3355 Benjamin Uekermann, Hans-Joachim Bungartz, Lucia Cheung Yau, Gerasimos Chourdakis, and Alexander
 3356 Rusch. 2017. Official preCICE adapters for standard open-source solvers. In Proceedings of the 7th GACM
 3357 Colloquium on Computational Mechanics for Young Scientists from Academia.

3358 preCICE User Documentation. (version 2.5.0). Accessed Sept, 2023.

3359 Nicolò Vaiana, Raffaele Capuano, Salvatore Sessa, Francesco Marmo, and Luciano Rosati. 2021. Nonlinear
 3360 dynamic analysis of seismically base-isolated structures by a novel openses hysteretic material model.
 3361 Applied Sciences, 11:900.

3362 Pauli Virtanen, Ralf Gommers, Travis E. Oliphant, Matt Haberland, Tyler Reddy, David Cournapeau, Evgeni
 3363 Burovski, Pearu Peterson, Warren Weckesser, Jonathan Bright, Stéfan J. van der Walt, Matthew Brett,
 3364 Joshua Wilson, K. Jarrod Millman, Nikolay Mayorov, Andrew R. J. Nelson, Eric Jones, Robert Kern, Eric
 3365 Larson, C J Carey, İlhan Polat, Yu Feng, Eric W. Moore, Jake VanderPlas, Denis Laxalde, Josef Perktold,
 3366 Robert Cimrman, Ian Henriksen, E. A. Quintero, Charles R. Harris, Anne M. Archibald, Antônio H.
 3367 Ribeiro, Fabian Pedregosa, Paul van Mulbregt, and SciPy 1.0 Contributors. 2020. SciPy 1.0: Fundamental
 3368 Algorithms for Scientific Computing in Python. Nature Methods, 17:261–272.

3369 E. Walhorn, A. Kölke, B. Hübner, and D. Dinkler. 2005. Fluid–structure coupling within a monolithic model
 3370 involving free surface flows. Computers Structures, 83(25):2100–2111.

3371 H. G. Weller, G. Tabor, H. Jasak, and C. Fureby. 1998a. Openfoam: A tensorial approach to computational
 3372 continuum mechanics using object-oriented techniques,. COMPUTERS IN PHYSICS, 12(6).

3373 H. G. Weller, G. Tabor, H. Jasak, and C. Fureby. 1998b. A tensorial approach to computational continuum
 3374 mechanics using object-oriented techniques. Computers in Physics, 12(6):620–631.

- 3375 Olof B Widlund. 1988. Iterative substructuring methods: Algorithms and theory for elliptic problems in
3376 the plane. In First International Symposium on Domain Decomposition Methods for Partial Differential
3377 Equations, Philadelphia, PA.
- 3378 Jan Wienke and Hocine Oumeraci. 2005. Breaking wave impact force on a vertical and inclined slender pile -
3379 theoretical and large-scale model investigations. Coastal Engineering, 52:435–462.
- 3380 Jebediah Wilson, Rakesh Gupta, M Asce, John Lindt, Milo Clauson, and Rachel Garcia. 2009. Behavior
3381 of a one-sixth scale wood-framed residential structure under wave loading. Journal of Performance of
3382 Constructed Facilities - J PERFORM CONSTR FACIL, 23.
- 3383 Andrew Winter. 2019. Effects of Flow Shielding and Channeling on Tsunami-Induced Loading of Coastal
3384 Structures. Ph.D. thesis.
- 3385 Andrew O. Winter, Mohammad S. Alam, Krishnendu Shekhar, Michael R. Motley, Marc O. Eberhard,
3386 Andre R. Barbosa, Pedro Lomonaco, Pedro Arduino, and Daniel T. Cox. 2020. Tsunami-like wave forces
3387 on an elevated coastal structure: Effects of flow shielding and channeling. Journal of Waterway, Port,
3388 Coastal, and Ocean Engineering, 146(4):04020021.
- 3389 Andrew O. Winter, Michael R. Motley, and Marc O. Eberhard. 2018. Tsunami-like wave loading of individual
3390 bridge components. Journal of Bridge Engineering, 23(2):04017137.
- 3391 H. K. W. Wong. 2015. Three-dimensional effects of tsunami impact on bridges.
- 3392 Tao Xiang, Denis Istrati, Solomon C. Yim, Ian G. Buckle, and Pedro Lomonaco. 2020. Tsunami loads on
3393 a representative coastal bridge deck: Experimental study and validation of design equations. Journal of
3394 Waterway, Port, Coastal, and Ocean Engineering, 146(5):04020022.
- 3395 Ji-Gang Xu, Gang Wu, De-Cheng Feng, and Jia-Jun Fan. 2021. Probabilistic multi-hazard fragility analysis
3396 of rc bridges under earthquake-tsunami sequential events. Engineering Structures, 238:112250.
- 3397 Zhiying Yang, Bo Huang, Bing Zhu, Jiawei Zhang, and Azhen Kang. 2021. Comparative study of tsunami-like

3398 wave-induced forces on medium-scale models of box girder and t-girder bridges. Journal of Bridge
3399 Engineering, 26(2):04020125.

3400 Harry Yeh, Ian Robertson, and Jane Preuss. 2005. Development of design guidelines for structures that serve
3401 as tsunami vertical evacuation sites.

3402 S. Zampini, S. Bnà, M. Valentini, and I. Spisso. 2020. Gpu-accelerated openfoam simulations using
3403 petsc4foam. In in proceedings of the 8th OpenFOAM conference.

3404 Minjie Zhu, Ibrahim Elkhetafi, and Michael Scott. 2018a. Validation of opensees for tsunami loading on
3405 bridge superstructures. Journal of Bridge Engineering, 23.

3406 Minjie Zhu, Frank McKenna, and Michael Scott. 2018b. Openseespy: Python library for the opensees finite
3407 element framework. SoftwareX, 7:6–11.

3408 O.C. Zienkiewicz, R.L. Taylor, and J.Z. Zhu. 2005. The Finite Element Method: Its Basis and Fundamentals.
3409 Elsevier Science.

3410 **Chapter A**

3411 **Appendix**

3412 **A.1 OpenFOAM Appendix**

3413 **A.1.1 Finite Volume Schemes and Finite Volume Solution Settings**

Listing A.1. OpenFOAM fvSchemes Dictionary

```
3414
3415 FoamFile
3416 {
3417     version 2.0;
3418     format ascii;
3419     class dictionary;
3420     location "system";
3421     object fvSchemes;
3422 }
3423 // * * * * *
3424
3425 ddtSchemes
3426 {
3427     default Euler;
3428 }
3429
3430 gradSchemes
3431 {
3432     default Gauss linear;
3433 }
3434
3435 divSchemes
3436 {
3437     div(rhoPhi,U) Gauss linearUpwindV cellLimited Gauss linear 1;
3438     div(U) Gauss linearUpwindV cellLimited Gauss linear 1;
3439     div((rhoPhi|interpolate(porosity)),U) Gauss limitedLinearV 1;
3440     div(rhoPhiPor,UPor) Gauss linearUpwindV cellLimited Gauss linear 1;
```

```

3441 div(rhoPhi,UPor) Gauss linearUpwindV cellLimited Gauss linear 1;
3442 div(rhoPhiPor,U) Gauss linearUpwindV cellLimited Gauss linear 1;
3443 div(phi,alpha) Gauss vanLeer01;
3444 div(phirb,alpha) Gauss upwind; interfaceCompression;
3445 div((muEff*dev(T(grad(U)))) Gauss linearUpwindV cellLimited Gauss linear 1;
3446 div(phi,epsilon) Gauss upwind;
3447 div((phi|interpolate(porosity)),k) Gauss upwind;
3448 div((phi|interpolate(porosity)),epsilon) Gauss upwind;
3449 div(phi,k) Gauss linearUpwind limitedGrad;
3450 div(phi,omega) Gauss linearUpwind limitedGrad;
3451 div((phi|interpolate(porosity)),omega) Gauss upwind;
3452     div((nuEff*dev2(T(grad(U)))) Gauss upwind;
3453     div(((rho*nuEff)*dev2(T(grad(U)))) Gauss linear;
3454 }
3455
3456 laplacianSchemes
3457 {
3458     default Gauss linear corrected;
3459     laplacian(yPsi) Gauss linear corrected;
3460
3461 }
3462
3463 interpolationSchemes
3464 {
3465     default linear;
3466 }
3467

```



```

3468 snGradSchemes
3469 {
3470     default corrected;
3471 }
3472
3473 fluxRequired
3474 {
3475     default no;
3476     p_rgh;
3477     pcorr;
3478     alpha.water;
3479 }
3480
3481 wallDist
3482 {
3483     method Poisson;
3484     //method meshWave;
3485 }
3486
3487
3488 // *****
3489  //
3490

```

Listing A.2. OpenFOAM fvSettings Dictionary

```

3491 FoamFile
3492 {
3493     version 2.0;
3494     format ascii;
3495     class dictionary;
3496     location "system";
3497     object fvSolution;
3498 }
3499
3500 // * * * * *
3501
3502 solvers
3503 {
3504     "alpha.water"
3505     {
3506         nAlphaCorr 2; // changed from 1, which was probably left over from trying the CrankNicolson
3507         ↪ ddt scheme
3508         nAlphaSubCycles 2;
3509         alphaOuterCorrectors yes;
3510         cAlpha 1;
3511         icAlpha 0; // newly added
3512         MULESCorr yes; // changed from no
3513         nLimiterIter 10; // changed from 3
3514         applyAlphaPrevCorr yes; // newly added
3515         solver smoothSolver;
3516         smoother symGaussSeidel;
3517         tolerance 1e-8;

```

```

3518         relTol 0;
3519         minIter 1; // newly added
3520     }
3521 "alpha.waterFinal"
3522 {
3523     nAlphaCorr 2; // changed from 1, which was probably left over from trying the CrankNicolson
3524     ↪ ddt scheme
3525     nAlphaSubCycles 1;
3526     alphaOuterCorrectors yes;
3527     cAlpha 1;
3528     icAlpha 0; // newly added
3529     MULESCorr yes; // changed from no
3530     nLimiterIter 10; // changed from 3
3531     applyAlphaPrevCorr yes; // newly added
3532     solver smoothSolver;
3533     smoother symGaussSeidel;
3534     tolerance 1e-8;
3535     relTol 0;
3536     minIter 1; // newly added
3537 }
3538
3539
3540 "cellDisplacement.*"
3541 {
3542     solver PCG;
3543     preconditioner DIC;
3544

```

```

3545     tolerance 1e-06;
3546     relTol 0;
3547     maxIter 100;
3548 }
3549 "yPsi.*"
3550 {
3551     solver PBiCGStab;
3552     preconditioner DILU;
3553
3554     tolerance 1e-06;
3555     relTol 0;
3556     maxIter 100;
3557 }
3558
3559 "alpha.water.*"
3560 {
3561     nAlphaCorr 2;
3562     nAlphaSubCycles 2;
3563     cAlpha 1;
3564     icAlpha 0;
3565
3566     MULESCorr yes;
3567     nLimiterIter 5;
3568     alphaApplyPrevCorr no;
3569
3570     solver smoothSolver;
3571     smother symGaussSeidel;

```

```

3572     tolerance 1e-8;
3573     relTol 0;
3574 }
3575
3576 "pcorr.*"
3577 {
3578     solver PCG;
3579     preconditioner DIC;
3580     tolerance 1e-9;
3581     relTol 0;
3582 }
3583
3584 p_rgh
3585 {
3586     solver PCG;
3587     preconditioner DIC;
3588     tolerance 1e-9;
3589     relTol 0.01;
3590 }
3591
3592 p_rghFinal
3593 {
3594     $p_rgh;
3595     relTol 0;
3596 }
3597
3598 "(U|k|omega|epsilon).*"

```

```

3599 {
3600     solver smoothSolver;
3601     smoother symGaussSeidel;
3602     tolerance 1e-08;
3603     relTol 0;
3604 }
3605
3606
3607 "cellDisplacementFinal"
3608 {
3609     $cellDisplacement;
3610     relTol 0;
3611 }
3612 }
3613
3614 PIMPLE
3615 {
3616     momentumPredictor yes;
3617     nOuterCorrectors 5;
3618     nCorrectors 3;
3619     nNonOrthogonalCorrectors 1;
3620     ddtCorr no;
3621     correctPhi no;
3622     moveMeshOuterCorrectors yes;
3623     consistent yes;
3624     turbOnFinalIterOnly no;
3625     pRefCell 0;

```

```

3626     pRefValue 0;
3627 }
3628
3629 relaxationFactors
3630 {
3631     fields
3632     {
3633     }
3634     equations
3635     {
3636         ".*" 1;
3637     }
3638 }
3639
3640 // *****
3641     ↪ //
3642

```

A.1.2 Coupling Related Files And Settings

Listing A.3. OpenFOAM preCICE Dictionary

```
3644
3645 FoamFile
3646 {
3647     version 2.0;
3648     format ascii;
3649     class dictionary;
3650     location "system";
3651     object preciceDict;
3652 }
3653
3654 preciceConfig "./precice—config.xml";
3655 participant Fluid;
3656 modules (FSI);
3657 interfaces
3658 {
3659     Interface1
3660     {
3661         mesh Fluid—Mesh;
3662         locations faceCenters;
3663         connectivity false;
3664         patches (interface);
3665         readData
3666         (
3667             Displacement
3668         );
3669         writeData
3670         (
```

```
3671         Force
3672     );
3673 };
3674 };
3675
3676 FSI
3677 {
3678     namePointDisplacement pointDisplacement;
3679     nameT p_rgh;
3680     nameP p;
3681 }
3682
```

A.1.3 Example Case Files

Listing A.4. OpenFOAM blockMeshDict for Walhorn Case

```
3684
3685 FoamFile
3686 {
3687     version 2.0;
3688     format ascii;
3689     class dictionary;
3690     object blockMeshDict;
3691 }
3692 // * * * * *
3693
3694 x0 -0.286;
3695 x1 0.0;
3696 x2 0.012;
3697 x3 0.298;
3698
3699 y0 0.;
3700 y1 0.08;
3701 y2 0.365;
3702
3703 z0 0;
3704 z1 0.01;
3705 zc1 2;
3706
3707 vertices
3708 (
3709     ($x0 $y0 $z0 ) // 0
3710     ($x1 $y0 $z0 ) // 1
```

```

3711 ($x2 $y0 $z0 ) // 2
3712 ($x3 $y0 $z0 ) // 3
3713 ($x0 $y1 $z0 ) // 4
3714 ($x1 $y1 $z0 ) // 5
3715 ($x2 $y1 $z0 ) // 6
3716 ($x3 $y1 $z0 ) // 7
3717 ($x0 $y2 $z0 ) // 8
3718 ($x1 $y2 $z0 ) // 9
3719 ($x2 $y2 $z0 ) // 10
3720 ($x3 $y2 $z0 ) // 11
3721
3722 ($x0 $y0 $z1 ) // 12
3723 ($x1 $y0 $z1 ) // 13
3724 ($x2 $y0 $z1 ) // 14
3725 ($x3 $y0 $z1 ) // 15
3726 ($x0 $y1 $z1 ) // 16
3727 ($x1 $y1 $z1 ) // 17
3728 ($x2 $y1 $z1 ) // 18
3729 ($x3 $y1 $z1 ) // 19
3730 ($x0 $y2 $z1 ) // 20
3731 ($x1 $y2 $z1 ) // 21
3732 ($x2 $y2 $z1 ) // 22
3733 ($x3 $y2 $z1 ) // 23
3734 );
3735
3736 // Grading
3737 h1 120;

```

```

3738 h2 4;
3739 h3 45;
3740 v1 24;
3741 v2 48;
3742
3743 blocks
3744 (
3745     hex ( 0 1 5 4 12 13 17 16 )
3746     ($h1 $v1 $zc1 )
3747     simpleGrading (0.5 1 1)
3748
3749     hex ( 2 3 7 6 14 15 19 18 )
3750     ($h3 $v1 $zc1)
3751     simpleGrading (2 1 1)
3752
3753     hex ( 4 5 9 8 16 17 21 20 )
3754     ($h1 $v2 $zc1)
3755     simpleGrading (0.5 2 1)
3756
3757     hex ( 5 6 10 9 17 18 22 21 )
3758     ($h2 $v2 $zc1)
3759     simpleGrading (1 2 1)
3760
3761     hex ( 6 7 11 10 18 19 23 22 )
3762     ($h3 $v2 $zc1 )
3763     simpleGrading (2 2 1)
3764 );

```

```

3765
3766 edges
3767 (
3768 );
3769
3770 boundary
3771 (
3772     leftWall
3773     {
3774         type wall;
3775         faces
3776         (
3777             ( 0 4 16 12 )
3778             ( 4 8 20 16 )
3779         );
3780     }
3781     rightWall
3782     {
3783         type wall;
3784         faces
3785         (
3786             ( 3 7 19 15 )
3787             ( 7 11 23 19 )
3788         );
3789     }
3790     interface
3791     {

```

```

3792     type wall;
3793     faces
3794     (
3795         ( 1 5 17 13 )
3796         ( 5 6 18 17 )
3797         ( 6 2 14 18 )
3798     );
3799 }
3800 atmosphere
3801 {
3802     type patch;
3803     faces
3804     (
3805         ( 8 9 21 20 )
3806         ( 9 10 22 21 )
3807         ( 10 11 23 22 )
3808     );
3809 }
3810 lowerWall
3811 {
3812     type wall;
3813     faces
3814     (
3815         ( 0 1 13 12 )
3816         ( 2 3 15 14 )
3817     );
3818 }

```



```

3819 defaultFaces
3820 {
3821     type patch;
3822     faces
3823     (
3824         ( 0 1 5 4 )
3825         ( 2 3 7 6 )
3826         ( 4 5 9 8 )
3827         ( 5 6 10 9 )
3828         ( 6 7 11 10 )
3829         ( 12 13 17 16 )
3830         ( 14 15 19 18 )
3831         ( 16 17 21 20 )
3832         ( 17 18 22 21 )
3833         ( 18 19 23 22 )
3834     );
3835 }
3836 );
3837
3838 // *****
3839  //
3840

```

3841 **A.2 OpenSees Appendix**

3842 **A.3 OpenSees Models**

Listing A.5. OpenSees Fixed-End Beam Model for FSI Analysis

```
3843
3844 ops.wipe()
3845 coupledNodes=[]
3846 E=2e8
3847 nu=0.3
3848 G=E/(2*(1+nu))
3849 numDOF=6
3850 ops.model('basic', '-ndm', 3, '-ndf', numDOF)
3851 nElem=8
3852 density=100
3853 domainThickness=0.1
3854 nNodes=nElem+1
3855 node1=[0. , 0.0, 0.0]
3856 node2=[10., 0.0, 0.0]
3857 beamNormal=[0., 0., 1.]
3858 beamLength=node2[0]-node1[0]
3859 A =0.1
3860 b=0.1
3861 h=1
3862 beamThickness=h
3863 Iz =b*(h**3)/12
3864 Iy = Iz
3865 J = 0.333
3866 matTag=1
3867 ops.nDMaterial('ElasticIsotropic', matTag, E, nu, 1000.0)
3868 xNodeList=np.linspace(node1[0],node2[0],nElem+1)
3869 yNodeList=np.linspace(node1[1],node2[1],nElem+1)
```

```

3870 zNodeList=np.linspace(node1[2],node2[2],nElem+1)
3871 MASS=density*beamLength*domainThickness*beamThickness
3872 for nodeNum in range(0, len(xNodeList)):
3873     ops.node(nodeNum, xNodeList[nodeNum],yNodeList[nodeNum],zNodeList[nodeNum])
3874     coupledNodes.append(nodeNum)
3875 print(coupledNodes, '_Coupled_nodes')
3876 coordTransf = 'Corotational'
3877 coordTransf='PDelta'
3878 #####
3879 nodeList=ops.getNodeTags()
3880 nodalMass=MASS/nNodes
3881 secTag=1
3882 integTag=1
3883 ops.section('Elastic', secTag, E, A, Iz, Iy, G, J)
3884 ops.beamIntegration('Legendre', integTag, secTag, 2)
3885 nodRotMass=0 #(1/12)*nodalMass*(leme**2)
3886 for node_num in range(nNodes):
3887     ops.mass(nodeNum,*[nodalMass,nodalMass,nodalMass,nodRotMass,nodRotMass,nodRotMass])
3888 for nodeNum in range(1,nNodes):
3889     ops.geomTransf(coordTransf, nodeNum+100000, beamNormal[0],beamNormal[1],beamNormal
3890         ↪ [2])
3891     ops.element('dispBeamColumn', nodeNum, *[nodeNum-1, nodeNum], nodeNum+100000,
3892         ↪ integTag) #, '-cMass', '-mass', mass=0.0)
3893 ops.fixX(0.0,*[1, 1, 1, 1, 1, 1])
3894 ops.fixX(10.0,*[1, 1, 1, 1, 1, 1])
3895 ops.recorder('Node', '-file', 'reactionNodeLHS.out', '-time', '-node', 0, '-dof', 1,2,3,4,5,6, 'reaction')
3896 ops.recorder('Node', '-file', 'reactionNodeRHS.out', '-time', '-node', nNodes-1, '-dof', 1,2,3,4,5,6, '

```

```

3897     ↪ reaction')
3898 ops.recorder('Node', '-file', 'dispNodeCenter.out', '-time', '-node', nNodes/2, '-dof', 1,2,3,4,5,6, 'disp')
3899 f1= 5.5
3900 f2=f1*5
3901 z1=0.05
3902 z2=0.05
3903 alphaM = 0.000 # (4*3.1415*f1*f2)*((z1*f2 - z2*f1)/(f2**2 - f1**2)) # M-prop. damping; D =
3904     ↪ alphaM*M
3905 betaKcurr = 00.00 # K-proportional damping; +beatKcurr*KCurrent <- not this
3906 betaKinit = ((z1*f2 - z2*f1)/(3.1415*(f2**2 - f1**2))) # initial-stiffness proportional damping +
3907     ↪ beatKinit*Kini <<<<<<<<----- use this
3908 betaKcomm = 0.0
3909 ops.rayleigh(alphaM,betaKcurr, betaKinit, betaKcomm) # RAYLEIGH damping
3910

```

Listing A.6. OpenSees Cantilever Beam Model for FSI Analysis

```

3911 ops.wipe()
3912 FOAMySeesInstance.osi=ops.model('basic','—ndm',3,'—ndf',6)
3913 RunEQ=1
3914 nElem=48
3915 nu=0.45
3916 EScaleFactor=1
3917 E=1000000*EScaleFactor #10^7 dyne or 1MPa
3918 structuralDensity=2500
3919 MASS=structuralDensity*0.08*0.012*0.01
3920 beamNormal=[1., 0., 0.]
3921 structuralDensity=2500
3922 node1=[0.006,0.00,0.005]
3923 node2=[0.006,0.08,0.005]
3924 I=0.01*(0.012**3)/12
3925 A = .01*0.012
3926 Iz = I
3927 Iy = Iz
3928 J = 0.333
3929 beamLength=node2[1]—node1[1]
3930 G=E/(2*(1+nu))
3931 sigmaY=1.25e5
3932 H_iso=0.5
3933 H_kin=0.25
3934 matTag=1
3935 node1=[0.006,0.00,0.005]
3936 node2=[0.006,0.08,0.005]
3937 beamNormal=[— 1,0,0]
3938 xNodeList=np.linspace(node1[0],node2[0],nElem+1)
3939 yNodeList=np.linspace(node1[1],node2[1],nElem+1)
3940 zNodeList=np.linspace(node1[2],node2[2],nElem+1)
3941 for nodeNum in range(0, len(xNodeList)):
3942     ops.node(nodeNum, xNodeList[nodeNum],yNodeList[nodeNum],zNodeList[nodeNum])
3943 K=E*G/(3*(3*G — E))
3944 coordTransf='PDelta'
3945 nodeList=ops.getNodeTags()
3946 nodeLocs=np.zeros([len(nodeList),3])
3947 nodalMass=MASS/len(nodeLocs)
3948 secTag=1
3949 matTag=1
3950 integTag=1
3951 ops.beamIntegration('Legendre', integTag, secTag, 2)
3952 for node in range(1,len(nodeList)):
3953     nodeLocs[node,:]=ops.nodeCoord(nodeList[node])
3954 nodRotMass=0
3955 for node_num in range(len(nodeLocs)):
3956     ops.mass(nodeList[node_num],*[nodalMass,nodalMass,nodalMass,nodRotMass,nodRotMass,nodRotMass])
3957 for nodeNum in range(1,len(nodeLocs)):
3958     ops.geomTransf(coordTransf, nodeNum+100000, beamNormal[0],beamNormal[1],beamNormal[2])
3959     ops.element('elasticBeamColumn', nodeNum, nodeNum—1, nodeNum, A, E, G, J, Iy, Iz, nodeNum+100000)
3960 ops.fixY(0.00,*[1,1,1,1,1,1])
3961 ops.recInfoList=['reactionBase.out',0,'reaction'],['tipDisplacement.out',nElem,'disp']]
3962

```

```
3963 ops.recorder('Node', '-file', 'DFree.out', '-time', '-node', 48, '-dof', 1,2,3, 'disp')
3964 ops.recorder('Node', '-file', 'AFixed.out', '-time', 'timeSeries', 'TSTAG', '-node', 0, '-dof', 1,2,3, 'reaction')
3965
```

Listing A.7. OpenSees Cantilever Beam Model for FSI Analysis

```
3966 ops.wipe()
3967
3968 FOAMySeesInstance.osi=ops.model('basic','-ndm',3,'-ndf',6)
3969 RunEQ=1
3970 nElem=48
3971 nu=0.45
3972 EScaleFactor=1
3973 E=1000000*EScaleFactor #10^7 dyne or 1MPa
3974 structuralDensity=2500
3975 MASS=structuralDensity*0.08*0.012*0.01
3976 beamNormal=[1., 0., 0.]
3977 structuralDensity=2500
3978 node1=[0.006,0.00,0.005]
3979 node2=[0.006,0.08,0.005]
3980 I=0.01*(0.012**3)/12
3981 A = .01*0.012
3982 Iz = I
3983 Iy = Iz
3984 J = 0.333
3985 beamLength=node2[1]-node1[1]
3986 G=E/(2*(1+nu))
3987 sigmaY=1.25e5
3988 H_iso=0.5
3989 H_kin=0.25
3990 matTag=1
3991 node1=[0.006,0.00,0.005]
3992 node2=[0.006,0.08,0.005]
```



```

3993 beamNormal=[-1,0,0]
3994 xNodeList=np.linspace(node1[0],node2[0],nElem+1)
3995 yNodeList=np.linspace(node1[1],node2[1],nElem+1)
3996 zNodeList=np.linspace(node1[2],node2[2],nElem+1)
3997 for nodeNum in range(0, len(xNodeList)):
3998     ops.node(nodeNum, xNodeList[nodeNum],yNodeList[nodeNum],zNodeList[nodeNum])
3999 # #####
4000 # ELEMENTS
4001 # #####
4002 matTag=1
4003 K=E*G/(3*(3*G - E))
4004 ops.uniaxialMaterial('Hardening', matTag, E, sigmaY, H_iso, H_kin)
4005 coordTransf='PDelta'
4006 #####
4007 nodeList=ops.getNodeTags()
4008 nodeLocs=np.zeros([len(nodeList),3])
4009 nodalMass=MASS/len(nodeLocs)
4010 secTag=1
4011 matTag=1
4012 ops.section('Fiber', secTag, '-GJ', G*J)
4013 ops.patch('rect', matTag, 2,12, *[-0.005,-0.006], *[0.005,0.006])
4014 for node in range(1,len(nodeList)):
4015     nodeLocs[node,:]=ops.nodeCoord(nodeList[node])
4016 integTag=1
4017 ops.beamIntegration('Legendre', integTag, secTag, 2)
4018 nodRotMass=0
4019 for node_num in range(len(nodeLocs)):

```

```

4020 ops.mass(nodeList[node_num],*[nodalMass,nodalMass,nodalMass,nodRotMass,nodRotMass,
4021     ↪ nodRotMass])
4022 for nodeNum in range(1,len(nodeLocs)):
4023     ops.geomTransf(coordTransf, nodeNum+100000, beamNormal[0],beamNormal[1],beamNormal[2])
4024     ops.element('dispBeamColumn', nodeNum, *[nodeNum-1, nodeNum], nodeNum+100000, secTag) #,
4025     ↪ '-cMass', '-mass', mass=0.0)
4026 ops.fixY(0.00,*[1,1,1,1,1,1])
4027 nodeRecInfoList=[['reactionBase.out',0,'reaction'],['tipDisplacement.out',nElem,'disp']]
4028 res=['disp','vel','accel','incrDisp','reaction','pressure','unbalancedLoad','mass']
4029 args='force'
4030 eleTags=[0,1]
4031 ops.recorder('Element', '-file', 'ElementOut.txt', '-precision', 6, '-time', '-dT', 0.001, '-ele', *eleTags, *
4032     ↪ args)
4033 os.system('rm rf_SeesoutPrelim')
4034 os.system('mkdir SeesoutPrelim')
4035 os.system('touch SeesoutPrelim.pvd')
4036 ops.recorder('PVD', 'SeesoutPrelim', '-precision', 4, '-dT', 0.01, *res)
4037 if RunEQ==1:
4038     GMfile='./fromUser/GroundMotion.acc'
4039     # Uniform EXCITATION: acceleration input
4040     IDloadTag = 400 # load tag
4041     dt = 0.001 # time step for input ground motion
4042     GMfact = 10 # data in input file is in g Unifts — ACCELERATION TH
4043     maxNumIter = 10
4044     GMdirection=1
4045     Tol=1e-3
4046     TSTAG=2

```

```

4047 ops.timeSeries('Path', TSTAG, '-dt', dt, '-filePath', GMfile, '-factor', GMfact, '-useLast')
4048 ops.pattern('UniformExcitation', IDloadTag, GMdirection, '-accel', 2)
4049 ops.recorder('Node', '-file', 'DFree.out', '-time', '-node', 48, '-dof', 1,2,3, 'disp')
4050 ops.recorder('Node', '-file', 'AFixed.out', '-time', 'timeSeries',TSTAG,'-node', 1, '-dof', 1,2,3, 'accel')
4051 ops.constraints('Transformation')
4052 ops.numberer('Plain')
4053 ops.system('BandGeneral')
4054 ops.test('EnergyIncr', Tol, maxNumIter)
4055 ops.algorithm('ModifiedNewton')
4056 NewmarkGamma = 0.5
4057 NewmarkBeta = 0.25
4058 ops.integrator('Newmark', NewmarkGamma, NewmarkBeta)
4059 ops.analysis('VariableTransient')
4060 DtAnalysis = 0.001
4061 TmaxAnalysis = 10
4062 Nsteps = int(TmaxAnalysis/ DtAnalysis)
4063 ok=1
4064 ops.algorithm('KrylovNewton')
4065 ok = ops.analyze(Nsteps, DtAnalysis,DtAnalysis/10,DtAnalysis,1)
4066 ops.remove('loadPattern',IDloadTag)
4067 ops.remove('timeSeries', 2)
4068

```

4069 **A.4 FOAMySees Appendix**

4070 **A.5 preCICE Files**

Listing A.8. preCICE Configuration XML File Example

```
4071 <?xml version="1.0" encoding="UTF-8" ?>
4072
4073 <precice-configuration>
4074   <log>
4075     <sink
4076       filter="%Severity%>_debug_and_%Rank%=_0"
4077       format="---[precice]_%ColorizedSeverity%_%Message%"
4078       enabled="true" />
4079   </log>
4080
4081   <solver-interface dimensions="3">
4082     <data:vector name="Force" />
4083     <data:vector name="Displacement" />
4084
4085     <mesh name="Fluid-Mesh">
4086       <use-data name="Displacement" />
4087       <use-data name="Force" />
4088     </mesh>
4089
4090     <mesh name="Coupling-Data-Projection-Mesh">
4091       <use-data name="Displacement" />
4092       <use-data name="Force" />
4093     </mesh>
4094
4095     <participant name="FOAMySeesCouplingDriver">
4096       <use-mesh name="Coupling-Data-Projection-Mesh" provide="yes" />
4097       <write-data name="Displacement" mesh="Coupling-Data-Projection-Mesh" />
```

```

4098     <read-data name="Force" mesh="Coupling-Data-Projection-Mesh" />
4099 </participant>
4100
4101 <participant name="Fluid">
4102     <use-mesh name="Fluid-Mesh" provide="yes" />
4103     <use-mesh name="Coupling-Data-Projection-Mesh" from="FOAMySeesCouplingDriver" /
4104     ↪ >
4105     <write-data name="Force" mesh="Fluid-Mesh" />
4106     <read-data name="Displacement" mesh="Fluid-Mesh" />
4107     <mapping:nearest-neighbor
4108         direction="write"
4109         from="Fluid-Mesh"
4110         to="Coupling-Data-Projection-Mesh"
4111         constraint="conservative" />
4112     <mapping:nearest-neighbor
4113         direction="read"
4114         from="Coupling-Data-Projection-Mesh"
4115         to="Fluid-Mesh"
4116         constraint="consistent" />
4117 </participant>
4118 <m2n:sockets from="Fluid" to="FOAMySeesCouplingDriver" exchange-directory="Fluid" />
4119 <coupling-scheme:parallel-explicit>
4120     <time-window-size value="0.001" />
4121     <max-time value="12.5"/>
4122     <participants first="Fluid" second="FOAMySeesCouplingDriver"/>
4123     <exchange data="Force" mesh="Coupling-Data-Projection-Mesh" from="Fluid" to="
4124     ↪ FOAMySeesCouplingDriver" />

```

```
4125     <exchange data="Displacement" mesh="Coupling-Data-Projection-Mesh" from="
4126         ↪ FOAMySeesCouplingDriver" to="Fluid" />
4127 </coupling-scheme:parallel-explicit>
4128 </solver-interface>
4129 </precice-configuration>
4130
```

A.6 FOAMySees Code Listings

Listing A.9. Loading Prediction Function Implementation in Python

```
4132 SHAPE=[]
4133
4134 f = open('Shape.txt','r')
4135
4136 for row in f:
4137     row = row.split('\n')
4138     SHAPE.append(float(row[0]))
4139
4140 NS=np.zeros((151,3))
4141
4142 crs=0
4143
4144 for R in NS:
4145     R[2]=SHAPE[crs]
4146     crs+=1
4147
4148 BL=10
4149
4150 SHAPE=NS
4151
4152 print(SHAPE)
4153
4154 OHG=0.5*1000
4155
4156 rhoWater=FOAMySees.config.rhoWater
4157
4158 HWATER=rhoWater/FOAMySees.config.rhoOverHGoal
4159
4160 maxdisp=0
4161
4162 nnnodes=151
4163
4164 W=-0.1*rhoWater*9.81*HWATER
4165
4166 SW=-0.1*100*9.81*1
4167
4168 CW2=W+SW
4169
4170 expD=(CW2*(BL**4))/(384*Sees.config.E*0.1/12)
4171
4172 W*=BL/nnnodes
4173
4174 SELFWEIGHT=np.ones((nnnodes,3))*SW*BL/nnnodes
4175
4176 for x in SELFWEIGHT:
4177     x[0]=0
```

```

4159         x[1]=0
4160     print(SELFWEIGHT)
4161     BSW=np.ones((nnnodes,3))
4162     for x in BSW:
4163         x[0]=0
4164         x[1]=0
4165     while t<tEnd:
4166         for node_num in range(1,len(FOAMySees.nodeLocs)):
4167             FOAMySees.displacement[node_num] ...
4168             ... [0]=ops.nodeDisp(FOAMySees.nodeList[node_num],1)
4169             FOAMySees.displacement[node_num] ...
4170             ... [1]=ops.nodeDisp(FOAMySees.nodeList[node_num],2)
4171             FOAMySees.displacement[node_num] ...
4172             ... [2]=ops.nodeDisp(FOAMySees.nodeList[node_num],3)
4173     maxdisp=FOAMySees.displacement.flat[abs(FOAMySees.displacement).argmax()]
4174     totalload=-HWATER*rhoWater*0.1*9.81*BL
4175     SINLOAD=(maxdisp*SHAPE*9.81*rhoWater*0.1)*BL/nnnodes
4176     SINWATER=(np.sum(np.abs(maxdisp*SHAPE))*BL/nnnodes)
4177     FLATWATER=((HWATER*BL)-SINWATER)
4178     FLATHT=FLATWATER/BL
4179     if FLATWATER>0:
4180         FLATLOAD=-BSW*(rhoWater*0.1*9.81*(FLATHT)*BL)/nnnodes
4181     else:
4182         FLATLOAD=BSW*(rhoWater*9.81*0.1* ...
4183         ... ((np.sum(np.abs(maxdisp*SHAPE))*BL/nnnodes)- ...
4184         ... (HWATER*BL))/BL)*BL)/nnnodes
4185     TFL=np.sum(FLATLOAD)+1e-12

```

```
4186     TSL=np.sum(SINLOAD)+1e-12
4187     F_RAT=(TFL+TSL)/totalload
4188     Force=SELFWEIGHT+ SINLOAD + FLATLOAD
4189     FOAMySees.force = Force
4190     FOAMySees.stepForward()
4191     t += FOAMySees.dt
4192
```

Listing A.10. pyVista Command to Read In Fluid Coupling Surface

```
4193
4194 import pyvista as pv
4195
4196 Branches = pv.read("FluidCouplingSurface.obj").cell_centers().points
```

Listing A.11. SciPy Spatial package KDTree usage

```
4197
4198 from scipy.spatial import KDTree
4199
4200 Tree=KDTree(FOAMySees.coupledNodeLocs)
4201
4202 BranchToNodeRelationships=Tree.query(Branches)[1]
```

Listing A.12. Python preCICE Solver Interface Initialization

```
4202
4203 from precice import *
4204
4205 interface = precice.Interface(solverName, configFileName, 0, 1)
```

Listing A.13. Python preCICE Solver Mesh Initialization

```
4206
4207 vertices=Branches
4208
4209 meshID = interface.get_mesh_id("Coupling-Data-Projection-Mesh")
4210
4211 vertexIDs = interface.set_mesh_vertices(meshID, vertices)
4212
4213 displacementID = interface.get_data_id("Displacement", meshID)
4214
4215 forceID = interface.get_data_id("Force", meshID)
```

Listing A.14. Python preCICE Coupling Scheme Initialization

```
4213
4214 precice_dt = interface.initialize()
4215
```

Listing A.15. OpenFOAM preCICE Adapter Function Object

```
4216
4217 functions {
4218     preCICE_Adapter
4219     { type preCICEAdapterFunctionObject;
```

```

4220         libs ("libpreciceAdapterFunctionObject.so");
4221     } ...
4222

```

Listing A.16. OpenFOAM preCICE preciceDict FSI Subdictionary

```

4223
4224 FSI {
4225     namePointDisplacement pointDisplacement;
4226     nameT p_rgh;
4227     nameP p;
4228 }
4229

```

Listing A.17. OpenFOAM Surface Mesh Extraction to OBJ File

```

4230
4231 surfaceMeshExtract —case Fluid —patches 'interface' —latestTime FluidCouplingSurface.obj
4232

```