Numerical Modeling of Large-Displacement Fluid-Structure Interaction: Preliminary Study Aimed at Analysis of Heart Valve Dynamics

by

Kyle Williston

Submitted in partial fulfillment of the
requirements for the degree of Master of Science

at

Dalhousie University
Halifax, Nova Scotia
August 2012

DALHOUSIE UNIVERSITY

DEPARTMENT OF ENGINEERING MATHEMATICS AND INTERNETWORKING

The undersigned hereby certify that they have read and recommend to the Faculty of Graduate Studies for acceptance a thesis entitled "Numerical Modeling of Large-Displacement Fluid-Structure Interaction: Preliminary Study Aimed at Analysis of Heart Valve Dynamics" by Kyle Williston in partial fulfilment of the requirements for the degree of Master of Science.

Dated: August 17, 2012

Supervisor: _____

Readers: _____

_____

DALHOUSIE UNIVERSITY

DATE: August 17, 2012

AUTHOR: Kyle Williston

TITLE: Numerical Modeling of Large-Displacement Fluid-Structure Interaction: Preliminary Study Aimed at Analysis of Heart Valve Dynamics

DEPARTMENT OR SCHOOL: Department of Engineering Mathematics and Internetworking

DEGREE: MSc CONVOCATION: October YEAR: 2012

_____

Signature of Author

Table of Contents

List of Tables vi

List of Figures vii

Abstract ix

List of Abbreviations Used x

Acknowledgements xi

Chapter 1 - Introduction 1

1.1 State of the Art in Numerical Heart Valve Modeling 2

1.1.1 Moving Mesh Methods 3

1.1.2 Fixed Mesh Methods 3

1.1.3 Discussion of Fixed Mesh Methods 3

1.2 Motivation and Research Objectives 5

1.3 Thesis Overview 6

Chapter 2 - FSI Modeling Using LS-DYNA 7

2.1 About LS-DYNA 7

2.2 Arbitrary Lagrangian-Eulerian Methodology 7

2.3 Conservation Equations 8

2.4 Explicit Solver 11

2.5 Equation of State Models 13

2.6 Material Models 14

2.7 FSI Coupling 16

2.8 ALE Multi-Material 17

2.9 Mesh Rezoning Algorithm 18

2.10 Advection 19

2.11 Boundary Conditions 21

Chapter 3 - Model Specifics 22
3.1 Dimensions and Meshing 22
3.2 Boundary Conditions 26
3.3 Materials 27
3.4 Fluid-structure coupling 28
3.5 LS-DYNA Controls 29
3.6 Equation of State 30
3.7 Element Formulation and Ambient Element Type 30
3.8 Simulations 31
Chapter 4 - Results 33
4.1 Stiff Flap 33
4.1.1 Physical Analysis 33
4.1.2 Convergence 39
4.2 Soft Flap 44
4.2.1 Physical Analysis 44
4.2.2 Convergence 51
4.3 Pressure Drop Test 56
4.3 Large Deformation 57
4.4.1 Constant Flow 58
4.4.2 Pulsatile Flow 63
Chapter 5 - Conclusion 68
5.1 Conclusions 68
5.2 Future Work 70
Bibliography 71
Appendix A - LS-DYNA Example Code 74

List of Tables

Table 3.1 - PIDs 23
Table 3.2 - Number of Elements in Each Mesh Size 23
Table 3.3 - Boundary condition types 26
Table 3.4 - Fluid Material Card 28
Table 3.5 - Flap Material Cards 28
Table 3.6 - Coupling Variables 29
Table 3.7 - *control_ale Card 29
Table 3.8 - *control_contact Card 29
Table 3.9 - *control_termination Card 30
Table 3.10 - *eos_gruneisen Card for Constant Flow 30
Table 3.11 - Element Details 31
Table 3.12 - Simulations 32
Table 4.1 - Relative Error for the Stiff Material Card 43
Table 4.2 - Relative Error for the Soft Material Card 55

List of Figures

Figure 2.1 - Lagrangian Mesh Movement 9
Figure 2.2 - Eulerian Mesh Movement 10
Figure 2.3 - ALE Mesh Movement 11
Figure 2.4 - LS-DYNA Explicit Solver 13
Figure 2.5 - LS-DYNA Penalty Based Coupling 17
Figure 2.6 - ALE Multi-Material Example 18
Figure 2.7 - LS-DYNA Advection Examples 20
Figure 3.1 - Channel Model 22
Figure 3.2 - Mesh 1 - # of Elements (a) 125, (b) 500, (c) 2000, (d) 8000 24
Figure 3.3 - Mesh 2 - # of Elements: (a) 10, (b) 30, (c) 80, (d) 200 25
Figure 3.4 - Mesh 2 overlapping Mesh 1 (base mesh) 25
Figure 3.5 - Boundary Conditions for Mesh 1 26
Figure 3.6 - Boundary Conditions for Mesh 2 27
Figure 3.7 - Materials Used in Channel Model 27
Figure 4.1 - Stiff Flap, Constant Flow: 0 to 1750ms, 250ms Spacing 35
Figure 4.2 - Stiff Flap, Constant Flow: 2000 to 3750ms, 250ms Spacing 36
Figure 4.3 - Stiff Flap, Constant Flow: 4000 to 5750ms, 250ms Spacing 37
Figure 4.4 - Stiff Flap, Constant Flow: 6000 to 9500ms, 500ms Spacing 38
Figure 4.5 - Stiff Flap, Constant Flow: 1500ms, Simulations (a)1, (b)2, (c)3, (d)4 40
Figure 4.6 - Stiff Flap, Constant Flow : 5000ms, Simulations (a)1, (b)2, (c)3, (d)4 41
Figure 4.7 - Stiff Flap, Constant Flow: 8500ms, Simulations (a)1, (b)2, (c)3, (d)4 42
Figure 4.8 - Simulation 1-4: Resultant Displacement vs. Time 43
Figure 4.9 - Soft Flap, Constant Flow: 0 to 1750ms, 250ms Spacing 46
Figure 4.10 - Soft Flap, Constant Flow: 2000 to 3750ms, 250ms Spacing 47
Figure 4.11 - Soft Flap, Constant Flow: 4000 to 5750ms, 250ms Spacing 48
Figure 4.12 - Soft Flap, Constant Flow: 6000 to 9500ms, 250ms Spacing 49
Figure 4.13 - Comparison of Flap Dynamics at Selected Frames [23] 50
Figure 4.14 - Soft Flap, Constant Flow: 1500ms, Simulations (a)5, (b)6, (c)7, (d)8 52

Figure 4.15 - Soft Flap, Constant Flow: 5000ms, Simulations (a)5, (b)6, (c)7, (d)8 53
Figure 4.16 - Soft Flap, Constant Flow: 8500ms, Simulations (a)5, (b)6, (c)7, (d)8 54
Figure 4.17 - Simulation 5-8: Resultant Displacement vs. Time 55
Figure 4.18 - Pressure Gradient 3D-Axisymmetric [18] 56
Figure 4.19 - Pressure Drop Test Schematic 57
Figure 4.20 - Simulation 9: Pressure Drop Test Results 57
Figure 4.21 - Soft Flap (Mod), Cons. Flow: 0 to 3500ms, 500ms Between Frames 59
Figure 4.22 - Soft Flap (Mod), Constant Flow: 4000 to 7500ms, 500ms Spacing 60
Figure 4.23 - Soft Flap (Mod), Constant Flow: 8000 to 11500ms, 500ms Spacing 61
Figure 4.24 - Soft Flap (Mod), Constant Flow: 12000 to 15500ms, 500ms Spacing 62
Figure 4.25 - Pulsatile Flow Graph 64
Figure 4.26 - Soft Flap (Mod), Pulsatile Flow: 0 to 8750ms, 1250ms Spacing 65
Figure 4.27 - Soft Flap (Mod), Pulsatile Flow: 10000 to 18750ms, 1250ms Spacing 66
Figure 4.28 - Soft Flap (Mod), Pulsatile Flow: 20000 to 28750ms, 1250ms Spacing 67

Abstract

The demand for artificial heart valve replacements is increasing as a result of birth defects, ageing and disease. Collaboration between engineers, biologists and mathematicians is necessary to handle problems related to biocompatibility and fluid dynamics. As a result of the increased demand for artificial heart valves, many new designs have been developed recently. A method to test those designs is to use mathematical modeling. This method has a relatively low-cost and can be used as a preliminary tool before expensive prototypes are created. This research analyzes the use of the numerical modeling software LS-DYNA for large-displacement fluid-structure interaction. It is a preliminary study aimed at the analysis of heart valve dynamics. In particular, a channel with flap model is created in LS-DYNA. The model's physics, convergence and ability to handle large deformations is investigated.

List of Abbreviations Used

Abbreviation	Definition
ALE	Arbitrary Lagrangian-Eulerian
CURVIB	Curvilinear Immersed Boundary
EOS	Equation of State
FEM	Finite Element Method
FSI	Fluid-Structure Interaction
FVM	Finite Volume Method
HCIB	Hybrid Cartesian/Immersed-Boundary
IBM	Immersed Boundary Method
PID	Part Identification
SPO	Split-Operator Method

Acknowledgements

First and foremost I thank my thesis supervisor Dr. Serguei Iakovlev. His knowledge and supervision for this work has provided me with the support needed to see it through until the end. He has been a great teacher and friend to me over the past few years and it has been a privilege to work for him. I could not have asked for a better supervisor for this work.

A very special thanks also goes to my co-supervisor Dr. Ferdinando Auricchio of the Department of Structural Mechanics located at the Università di Pavia, Italy. His expertise and passion for biomedical engineering has inspired me. Additionally, financial support from Università di Pavia has made this work possible and given me the opportunity to travel to Italy to meet with Dr. Auricchio's research team.

I would also like to thank Adrien Lefieux of the Department of Structural Mechanics located at the Università di Pavia, Italy. His guidance played an important role to the success of this work.

Additionally, I would like to thank my family and friends who have supported me from the beginning until the end. Without you this work would have not been possible.

Chapter 1 - Introduction

Dysfunctional heart valves as a result of birth defects, ageing or disease often lead to the use of mechanical or bioprosthetic replacements. This year it is expected that over 300,000 heart valve procedures will take place and that number will expand to 850,000 by the year 2050 [**1**]. The demand for artificial heart valves is increasing at a rate of 10-12% per year as a result of those procedures [**2**]. The complicated features of artificial heart valves such as biocompatibility and fluid dynamics requires experts from multiple backgrounds to work together [**3**].

Mathematical modeling has become an important tool for analyzing the complex dynamics of heart valves [**4**]. It can be considered as a preliminary tool for analyzing heart valve designs before expensive prototypes are created and tested. Results from the modeling procedure can provide valuable insights on heart valve designs or also improve our understanding of heart disease [**5**]. Additionally, modeling provides a flexible framework for modifying important variables quickly at a relatively low cost in comparison to physical models [**6**].

This study analyzes the fluid-structure interaction (FSI) of a channel containing a pressurized liquid interacting with a flexible flap. From this point onward the channel with flexible flap model will be referred to simply as the channel model. The channel model is analyzed using the commercial finite element method software LS-DYNA (version 971, revision 65543) developed by Livermore Software Technology Corporation. The channel model is to be used as a preliminary study before delving into the more complicated full heart valve model.

The contributions of this work is the success of showing that LS-DYNA is able to handle large-deformation FSI simulations. To prove this, a physical analysis, convergence analysis, pressure drop test, and comparison has been completed. As this is a preliminary study, this work is also an initial step into the development of a full heart-valve model within LS-DYNA. In future studies, the model can be adapted to include parameters that will bring it closer to an anatomically correct model. This will require the inclusion of additional flaps, new materials, heart-beat flow schemes, and an anatomically accurate

mesh. It is also important to note on the economical advantage of LS-DYNA. It is not as expensive as other commercially available finite element packages and this allows it to be much more accessible.

1.1 State of the Art in Numerical Heart Valve Modeling

Researchers have developed multiple approaches to modeling heart valve dynamics numerically, each with their own advantages and disadvantages. Broadly speaking, two main categories exist for how geometries are handled for numerical heart valve modeling:

1. Moving mesh models: the discretized fluid-continuum is able to move with the structural component. Mesh velocities for the fluid are included in the governing equations. The most popular case of a moving mesh model is the arbitrary Lagrangian-Eulerian (ALE) method.
2. Fixed mesh models: the discretized fluid-continuum is unable to move. The Immersed boundary method (IBM) is commonly used for this model [**7**].

Moving mesh models can be applied to variety of engineering problems such as metal forming, casting, FSI, and coupling of multi-physics fields that contain multiple materials. Additionally, several enhancements have been used to improve the quality of moving mesh models such as adaptive meshing and parallel computation [**8**]. The fixed mesh models such as the IBM were first developed with the FSI of cardiac dynamics in mind. They prove to have an advantage in cardiac modeling due to its ability to handle large deformations more easily. Moving mesh models tend to deform too much during the dynamics of the heart valve leading to unreliable results [**9**].

Additionally, multiple methodologies have been developed to solve the governing equations. The two most common methods are the finite element method (FEM) which is used by LS-DYNA and the finite volume method (FVM), however, many other methodologies exist and have all shown to provide reasonable and insightful results [**6**]. In general, FEM is considered to be more accurate than FVM because its ability to handle a higher number of degrees of freedom (DOF) [**10**].

In [**10**] a study is presented that compares FEM and FVM. In the study they simulate oscillatory flow through the abdominal aortic bifurcation, both the FEM and FVM flow

solvers were parallelized. The result of the study showed that the FEM model produced converged results 20 times faster than the FVM model. The results themselves were quite similar but the significant time difference makes FEM superior in this case.

1.1.1 Moving Mesh Methods

These methods are based on a fluid grid that moves during the deformation of the flap/heart valve. The movement of the grid is processed using the ALE formulation of governing equations. This approach has been found to be very restrictive as updating the mesh every time step becomes very computationally costly. Additionally, for large displacement problems such as in the case of heart valves, obtaining smooth and acceptable meshes is nearly impossible. For these reasons, the moving mesh methods are not the best choice for simulating heart valve dynamics [**9**].

1.1.2 Fixed Mesh Methods

Fixed mesh methods are the more favorable option for large deformation FSI problems because they have shown to be very versatile and able to handle the large deformations in heart valves [**11**]. In this methodology, the fluid mesh is fixed and discretized separately from the solid mesh. The fluid mesh remains static throughout the simulation while the solid mesh is moved by adding body forces to the governing equations of fluid motion. Large deformations become very applicable for this type of methodology because the fluid mesh does not conform to the solid mesh [**9**].

1.1.3 Discussion of Fixed Mesh Methods

Peskins in 1972 was the first to simulate heart valve motion using IBM [**12**]. In general, for Newtonian fluids governed by the Navier-Stokes equations the IBM is described as:

$$\rho\left(\frac{\partial u(x,t)}{\partial t} + u \cdot \nabla u\right) = \mu \Delta u(x,t) - \nabla p + f(x,t)\Big), \tag{1.1}$$

$$\nabla \cdot u = 0, \tag{1.2}$$

where ρ is the constant fluid density, u is the fluid velocity, x is the position, t is time, μ is the dynamic viscosity, p is pressure, and f is the force density. The force density is described as:

$$f(x,t) = \sum_{j=1}^{N} \delta_a(x - Z_j)F_j, \tag{1.3}$$

where Z_j is a collection of interacting particles, F_j is the force acting on the jth particle, and δ_a is an approximation of the Dirac delta function smoothed over the length a. Lastly, the immersed boundary is updated:

$$\frac{dZ_j}{dt} = \int \delta_a(x - Z_j)u(x,t)dx. \tag{1.4}$$

Peskin and McQueen later applied this method to a full heart model [**12**], [**13**].

Another type of IBM is the fictitious domain method [**14**]. Again, the fluid and solid mesh are separated, however, unique to this case, the fluid and solid interface are coupled using local body forces. Several studies have shown this method to work for 2D and 3D heart models with a non-physiological Reynolds numbers [**9**]. The main concern with this method is that accurate results for viscous shear stresses on the solid boundaries are not possible [**15**]. However, in [**15**] they eliminate the inaccurate viscous shear stresses by using an adaptive mesh, which refines the mesh in the area of highest activity (boundaries) as the simulation progresses. Using this modified version of the fictitious domain method has shown promising results but also very high computational costs. Additionally, it has still not been used to simulate a full 3D heart model under realistic physiologic conditions [**9**].

Most recently, a new class of the IBM has been developed which in itself contains several sub-classes. Broadly speaking, it is known as sharp-interface IBMs and it aims to overcome the need for a high resolution mesh around the fluid-structure interface [**9**]. Unlike the previous IBMs that diffuse the no-slip, no-penetration boundary condition to areas adjacent to the fluid-structure interface, these methods impose the boundary precisely on the prescribed boundary. These methods give second order accuracy of the boundary layer, unlike the first order accuracy that the diffusive methods provide. In short, the following are examples of sharp-interface IBMs:

1. Cut-cell method: grid cell shape around the boundary is fitted to create a local boundary fitted-mesh [**9**]
2. Immersed interface method: jump condition inherent in the IBM is used in a finite difference framework to avoid the smearing effect and to avoid approximation of the discrete delta function using a smooth function [**9**].
3. Hybrid Cartesian/Immersed-boundary method (HCIB): Reconstructs the boundary condition on the immersed boundary using interpolation techniques [**9**].
4. Curvilinear Immersed Boundary method (CURVIB): The HCIB method is modified using a body fitted curvilinear mesh [**9**].

All sharp-interface IBMs have shown to be very successful for modeling heart valve dynamics and researchers continue to develop new codes. For instance, [**11**] developed a very efficient version of the CURVIB which overcomes inefficient time stepping inherent in CURVIB.

1.2 Motivation and Research Objectives

The goal of this study is to develop a model that can later be advanced to analyze the inner workings of the aortic heart valve. Developing and testing the channel model is a logical first step to take before approaching a full aortic valve model, the channel model will provide insight into the workings of LS-DYNA and its capabilities as it applies to FSI modeling in the context of aortic valves.

LS-DYNA uses a class of numerical methods known as hydrocodes to perform FSI [**16**]. Details outlining this methodology are given in Chapter 2 [**17**]. The hydrocodes use ALE methodologies to solve the channel model, however, in this study all fluid meshes are stationary (pure Eulerian). This is done using the Split-Operator Method (SPO) which is explained in Chapter 2.

This research aims to fulfil a number of objectives with respect to the problem statement:

1. To design a functional channel model in LS-DYNA that has FSI, modifiable materials, constant and pulsatile flow options, and the ability to refine the continuum mesh.

2. To test for convergence of the model using mesh refinement techniques.
3. To compare the model to an existing pressure drop test.
4. To analyze large deformations of the channel model to test LS-DYNA's capabilities.
5. To suggest future work for the channel model and its adaptation to a full heart-valve model.

1.3 Thesis Overview

Chapter 2 shows the background theory used in LS-DYNA that pertains to the present study. Governing equations, time integration, equation of state models, material models, FSI coupling, multi-material handling, advection, and boundary conditions are discussed. Chapter 3 develops the channel model used in this research. All important parameters are discussed in detail and the various simulations are outlined. Example code used to create the channel model is provided in Appendix A. Lastly, Chapter 4 analyzes the physics, convergence, and feasibility of the channel model. Physics is analyzed qualitatively and compared to existing papers when possible. Convergence is tested by refining the channel model mesh and comparing the resultant displacement of the flap. Pressure is tested by modifying the channel model to what is shown by [**18**]. Feasibility looks at the dynamic capabilities of the LS-DYNA using a highly flexible flap under constant and pulsatile flow.

Chapter 2 - FSI Modeling Using LS-DYNA

LS-DYNA version 971 is used to model constant and pulsatile flow through a channel containing a thin flexible flap. This chapter discusses the methods used in LS-DYNA as it pertains to the channel model.

2.1 About LS-DYNA

LS-DYNA is a general-purpose finite element program used to simulate highly complex nonlinear and transient dynamic problems using explicit time integration. LS-DYNA is command line driven and all that is required for proper execution is the executable file, input file, and free disk space. Many third-party preprocessing programs exist along with the manufacturers own non-licensed LS-PrePost which aids in the graphic visualization and setup of a problem [**19**].

2.2 Arbitrary Lagrangian-Eulerian Methodology

Three methods that LS-DYNA can use to obtain solutions to a finite element problem are Lagrangian, Eulerian, or ALE. Lagrangian and Eulerian methods are special cases of ALE. For the channel model a Eulerian approach is used for the fluid and a Lagrangian approach is used for the flap. This forces the fluid mesh to be stationary and the flap to move according to the body forces and FSI coupling parameters.

Typically the Lagrangian method is used primarily to solve structural deformations. The mesh in this case is attached to the material and moves with it as forces based on the physics of the problem result in displacement of the nodes. This is advantageous as it follows the nodes of the material and can handle complex geometries more easily. A pure Lagrangian method cannot be used for this model because it contains a mixture of fluid flow and a Lagrangian flap part.

The Eulerian method is most commonly used for the advection of fluids through a mesh. The mesh itself does not deform in any way during the calculations making the Eulerian model ideal for the fluid part of the channel model. A common example of the Eulerian method is flow of water through a straight pipe. Given a pressure gradient and proper boundary conditions the resulting velocities and pressure can be obtained at various times and locations in the pipe.

Finally, the ALE method which takes into account both the deformation of the mesh and the advection of a fluid (or fluid-like) medium. This method is not ideal for this problem because it is expected that large deformations will occur in the flap. These deformations in the flap would have a negative impact on the fluid mesh by skewing its shape.

2.3 Conservation Equations

Conservation equations in LS-DYNA are used to compute displacements and mesh velocities. Equations for the conservation of mass (2.1), momentum(2.2), and energy(2.3) are given as:

$$\frac{\partial \rho}{\partial t} = (-\rho) div(v) - (v - u)\nabla \rho, \tag{2.1}$$

$$\rho \frac{\partial v}{\partial t} = \sigma - \rho(v - u)\nabla v, \tag{2.2}$$

$$\rho \frac{\partial e}{\partial t} = \sigma \varepsilon - \rho(v - u)\nabla \rho, \tag{2.3}$$

where ρ is the fluid density, t is time, u and v are the mesh and particle velocity, e is the total mechanical energy, σ is the stress and ε is the strain [**8**].

If a Lagrangian formulation is being used by LS-DYNA the velocity components are set equal to each other ($u = v$). It is possible during large deformations that this will lead to distortion of the elements. In this case the calculations will cease and a manual mesh rezoning will be required before continuation [**8**]. An example of the Lagrangian mesh movement is shown in Figure 2.1.

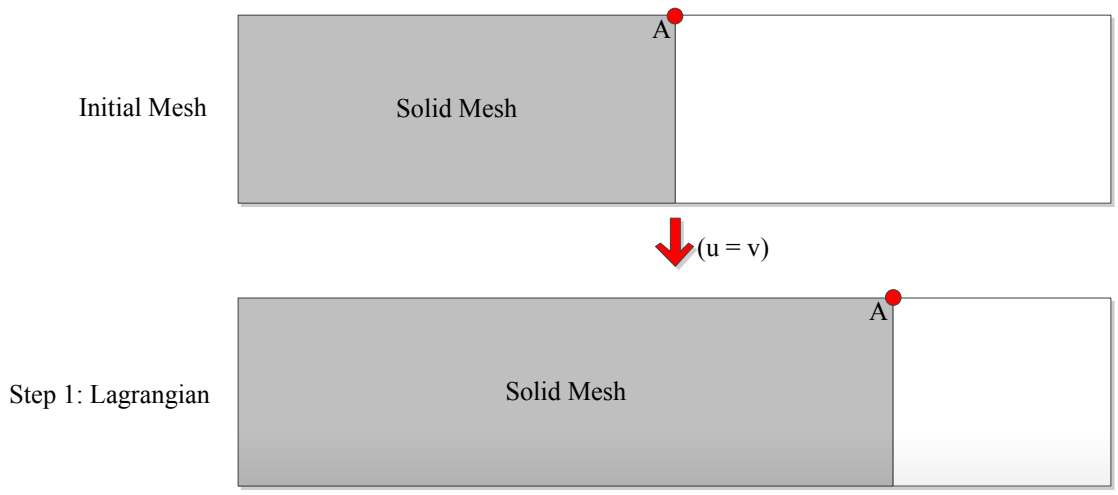

Figure 2.1 - Lagrangian Mesh Movement

In Figure 2.1 a 1D representation of the Lagrangian mesh movement is shown. First the mesh is initialized with the edge of the material ending at 'A'. After the Lagrangian step (guided by governing equations) takes place the value of 'A' moves to its new position.

During the Eulerian formulation the mesh is stationary and does that move, therefore in this case the mesh velocity is set to zero (u = 0). However, the mesh velocity is not explicitly set to zero, calculations using the Eulerian formulation requires two steps and use of the SPO in order to solve the equations. The two steps are as follows:

1. Mesh velocity (u) is set equal to the particle velocity (v). During this step this mesh will move with the fluid exactly like the Lagrangian formulation. This ensures that mass will be conserved. Mesh node characteristics, displacements and velocities are calculated here until the deformation becomes excessive. Step 2 is triggered by a drop in the time-step due to excessive deformation.
2. Mesh velocity is set to zero and mesh rezoning algorithms are used to place the mesh back into its original location. After successfully rezoning the mesh, advection of material is transported by means of material flux.

A 1D representation of the Eulerian mesh movement is shown in Figure 2.2. In this figure the initialized mesh and step 1 are the same as shown in the Lagrangian methodology. After the Lagrangian step takes place the point A is rezoned back to its original position

in the initial mesh. During rezoning of the fluid mesh in step 2, the material flux is calculated by subtracting the difference of point A in the step 2 to step 1 mesh.

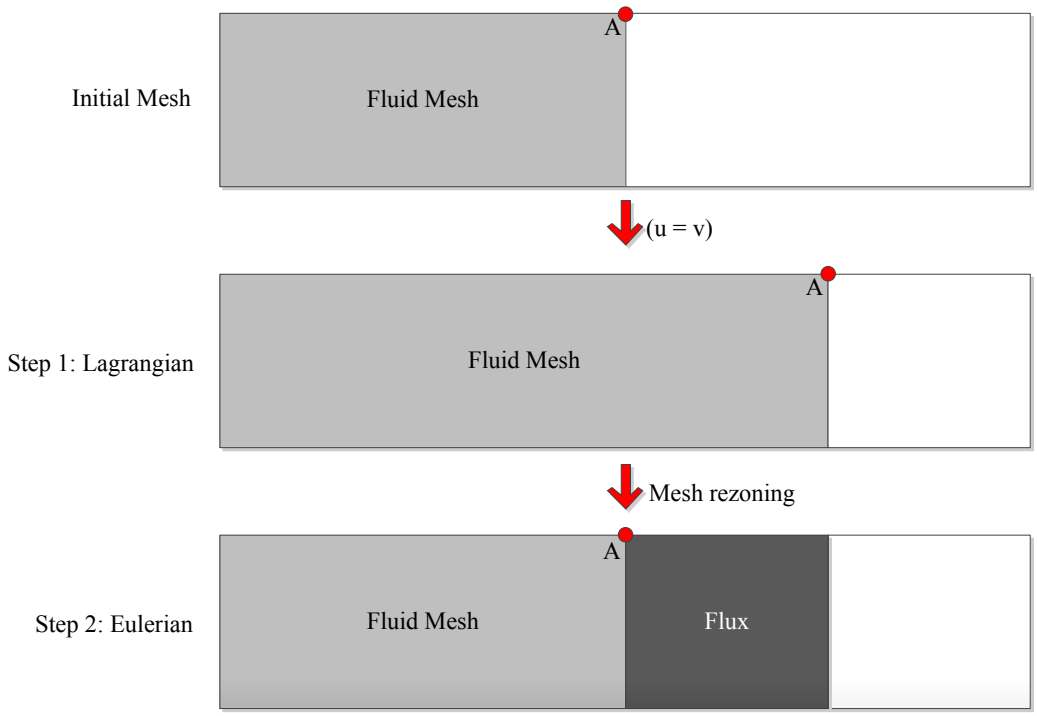

Figure 2.2 - Eulerian Mesh Movement

The ALE formulation is similar to the Eulerian formulation with one major difference. During the mesh rezoning phase, instead of returning the mesh back into its original position it places it to an optimal position decided by a mesh rezoning algorithm. At this point the remaining flux is calculated. The 1D representation of the ALE mesh movement can be seen in Figure 2.3. Again, the initial mesh and step 1 mesh are identical to the Lagrangian mesh movement. In step 2 the mesh is moved to an optimal position decided by the mesh rezoning algorithm explained later in this chapter. Again, the difference between point A in the step 2 and the step 1 mesh is calculated to determine the material flux.

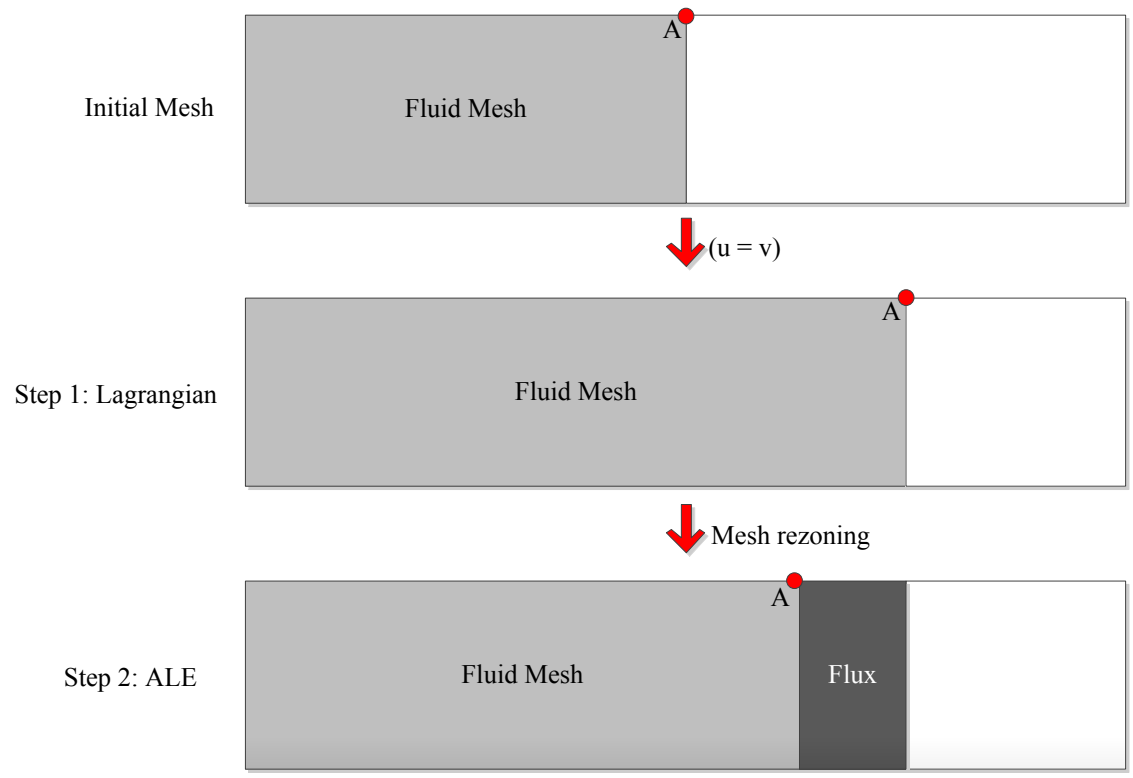

Figure 2.3 - ALE Mesh Movement

2.4 Explicit Solver

The explicit solver in LS-DYNA aims to solve the displacement, u^{n+1}, at time t^{n+1}. The explicit solver is only valid for dynamic problems and solves the following equation of motion [**20**]:

$$m\ddot{u}_n + c\dot{u}_n + ku_n = f^{ext}, \tag{2.4}$$

where m is the mass, $\ddot{u}$ is acceleration, c is a damping factor, $\dot{u}$ is the velocity, k is the linear stiffness, u is the displacement, f^{ext} is the external forces, and n is the time state. The ku^n part of Equation 2.4 can also be called the internal forces (f^{int}) [**20**]. The concept of the explicit solver can be seen in the time step variables used. It can be written as follows:

$$Explicit: d^{n+1} = f(u_n, \dot{u}_n, \ddot{u}_n, u_{n-1}, \dot{u}_{n-1}, , \dots). \tag{2.5}$$

Here all of the terms at the present time state "n" are known. This information is used to solve for the next time directly.

First velocity and displacement values are updated:

$$\dot{u}_{n+1} = \dot{u}_{n-1} + \Delta t_n \ddot{u}_n, \tag{2.6}$$

$$u_{n+1} = \dot{u}_{n+1/2} \Delta t_{n+1/2} + u_n, \tag{2.7}$$

where Δt is the time step. In Equation (2.6), initial conditions are used to imposed on the model are used. Equation 2.7 then uses the updated velocity from Equation 2.6 to update the displacement. Next, the internal forces are computed:

$$\varepsilon_{n+1} = B\ddot{u}_{n+1}, \tag{2.8}$$

$$B = \begin{pmatrix} \frac{\partial}{\partial x} & 0 & 0 \\ 0 & \frac{\partial}{\partial y} & 0 \\ 0 & 0 & \frac{\partial}{\partial z} \\ \frac{\partial}{\partial y} & \frac{\partial}{\partial x} & 0 \\ 0 & \frac{\partial}{\partial z} & \frac{\partial}{\partial y} \\ \frac{\partial}{\partial z} & 0 & \frac{\partial}{\partial x} \end{pmatrix} N, \tag{2.9}$$

$$\sigma_{n+1} = F(\varepsilon_{n+1}), \tag{2.10}$$

$$f_{n+1}^{int} = \int_{\dot{V}} B^t \sigma_{n+1} \cdot dV, \tag{2.11}$$

where B is the strain-displacement matrix, N is an interpolation matrix which contains the shape functions for the 8-noded hexahedron solid element, σ is the stress, ε is the strain, t denotes a transpose matrix, and V is the volume of the element. Stress is updated using a function of strain which depends on the material model used. Next, the external forces are calculated using body loads and boundary conditions. Lastly, the accelerations are calculated:

$$a_{n+1} = m^{-1}(f_{n+1}^{ext} - f_{n+1}^{int} - c\dot{u}_{n+1/2}). \quad (2.12)$$

After the new acceleration is calculated the next time step takes places and repeats the process. The explicit solver is summarized in the flow-chart provided in Figure 2.4.

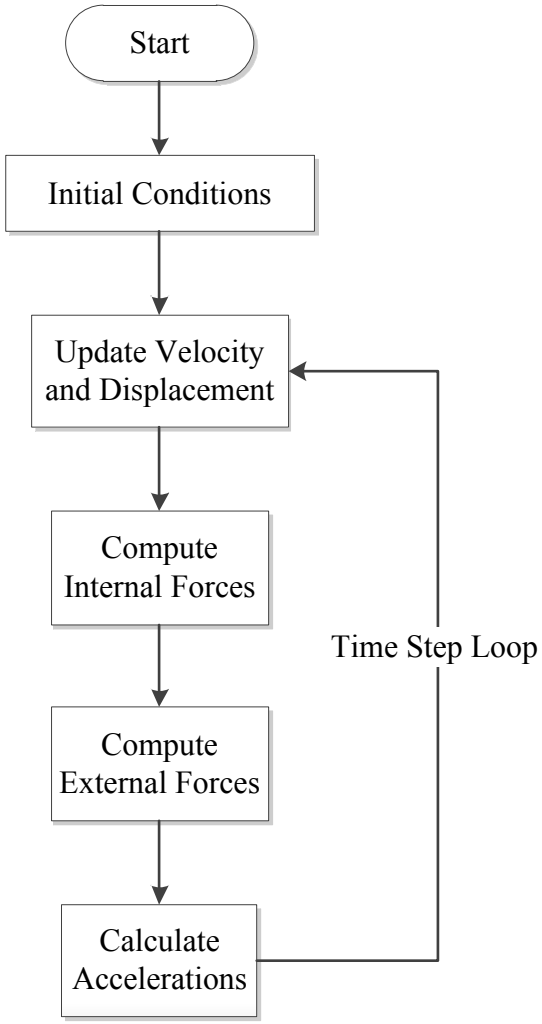

Figure 2.4 - LS-DYNA Explicit Solver

2.5 Equation of State Models

LS-DYNA uses equation of state (EOS) models to relate materials to their associated pressure, density, volume, and internal energy. Additionally they can be used to initialize parameters (for example the initial pressure of an inflow tank). In this study fluid and pressurized fluid use the Gruneisen equation of state model.

The Gruneisen EOS model defines pressure for compressible materials as:

$$p = \frac{\rho_0 C^2 \mu \left[1 + \left(1 - \frac{\gamma_0}{2}\right)\right] \mu - \frac{a}{2}\mu^2}{\left[1 - (S_1 - 1)\mu - S_2 \frac{\mu^2}{\mu + 1} - S_3 \frac{\mu^3}{(\mu + 1)^2}\right]^2} + (\gamma_0 + a\mu)E, \tag{2.13}$$

where E is the internal energy per initial volume, C is the intercept of the shock velocity and particle velocity curve, S_1, S_2 and S_3 are coefficients of the slope of the shock velocity and particle velocity curve, γ_0 is the Gruneisen gamma, a is the first order volume correction to γ_0, ρ_0 is the initial density, and μ is the compression state defined by Equation (2.14) [**19**]. In Equation 2.14, ρ defines the material density.

$$\mu = \frac{\rho}{\rho_0} - 1, \tag{2.14}$$

2.6 Material Models

Material models in LS-DYNA are used to distinguish between different physical mediums that may interact with each other. Two different types of material models are used in this study, they are the so called null material (used for fluid) and Mooney-Rivlin Rubber (used for the thin flexible flap) model [**20**].

The null material model allows for a user-defined mass density and viscosity coefficient. This material model was setup to use the Gruneisen EOS allowing for pressure initialization in the fluid medium. In particular, this is used to create a pressurized source at the inflow of the channel model. The null material model calculates viscous stress (σ_v) as:

$$\sigma_v = 2\mu\dot{\varepsilon}, \tag{2.15}$$

where μ is the user-defined dynamic viscosity and $\dot{\varepsilon}$ is the strain rate calculated from the explicit solver [**19**]. Lastly, total stress (σ_t) is calculated using the pressure obtained from Equation 2.13:

$$\sigma_t = \sigma_v - p \cdot I_d, \tag{2.16}$$

where I_d is the identity matrix [**19**]. This stress can then be used to calculate the internal forces shown in Equation 2.11.

The Mooney-Rivlin rubber was chosen for the flap material because of its ability to handle high levels of deformation. This material is based on the strain energy function, W, as follows:

$$W = A(I_1 - 3) + B(I_2 - 3) + C\left(\frac{1}{I_3^2} - 1\right) + D(I_3 - 1)^2, \tag{2.17}$$

where A and B are user-defined constants [**20**]. C and D are defined as follows:

$$C = \frac{1}{2}A + B, \tag{2.18}$$

$$D = \frac{A(5v - 2) + B(11v - 5)}{2(1 - 2v)}, \tag{2.19}$$

where v is Poisson's ratio [**20**]. I_1-I_3 are invariants related to the Cauchy-Green tensor (C) as shown here [**20**]:

$$I_1 = tr(C), \tag{2.20}$$

$$I_2 = \frac{1}{2}[(tr(C))^2 - tr(C^2)], \tag{2.21}$$

$$I_3 = \det(C). \tag{2.22}$$

Physically speaking, the Cauchy-Green tensor is used to give the square local change in distance due to deformation, it is related to a deformation gradient (F) as shown here [**20**]:

$$C = F^T F. \tag{2.23}$$

Taking the partial-derivative of W and the derivative of the invariants (I_1-I_3) yields the second Piola-Kirchhoff stress equation given as [**20**]:

$$S_{ij} = 2A \cdot I_d + 2B\left(I_1 \cdot I_d - C_{ij}\right) - 4C\frac{1}{I_3^2}C_{ij}^{-1} + 4D(I_3 - 1)I_3 C_{ij}^{-1} \tag{2.24}$$

Equation 2.24 can then be transformed into the Cauchy stress by using a push forward operation. The push forward operation on Equation 2.24 is not given in the LS-DYNA theory manual. The equations are:

$$\sigma_{ij} = \frac{1}{J} F_{ik} S_{kl} F_{jl}, \tag{2.25}$$

$$J = \det(F_{ij}), \tag{2.26}$$

where σ_{ij} is the Cauchy stress, F is the deformation gradient, S is the Piola-Kirchhoff stress, and the subscript variables represent Christoff symbols of the first kind [**20**]. After calculating the Cauchy stress it is used to calculate internal forces with Equation 2.11.

2.7 FSI Coupling

LS-DYNA uses a penalty based coupling approach to model FSI coupling between the Eulerian and Lagrangian interfaces [**20**]. The penalty based approach acts like a spring system where one end of the spring is attached to a slave node (solid) and the other is attached to a master node (fluid).

The spring system adds coupling forces to the model during penetration of a slave node through a master node, this is defined as:

$$F = k \cdot d, \tag{2.27}$$

where k is the spring stiffness and d is the penetration [**21**]. This force is applied to both the slave and master node to satisfy force equilibrium at the fluid-structure interface. The force applied to the structure (F_s) and fluid (F_f) node is defined as:

$$F_s = -F, \tag{2.28}$$

$$F_f^i = N_i \cdot F, \tag{2.29}$$

where N_i is the shape function at node i [**21**]. Calculation of the stiffness coefficient is calculated by the following equation:

$$k = p_f \frac{KA^2}{V}, \tag{2.30}$$

where p_f is a scalar factor ($0 \leq p_f \leq 1$), K is the bulk modulus of the fluid element, A is the average area of structure elements connected to the slave node and V is the volume of the fluid element that contains the master node [**21**].

Figure 2.5 demonstrates the penalty based coupling used in LS-DYNA. The figure contains two pictures that are separated by one time step, Δt. In the left picture the master node (fluid) is near the slave node (solid). After advection occurs in the second picture the master node penetrates through the solid into the void space. A spring is attached to the master and slave node with the head of the spring located on the structure. Coupling forces are applied at the nodes to counteract the penetration thus removing the master node from the void space.

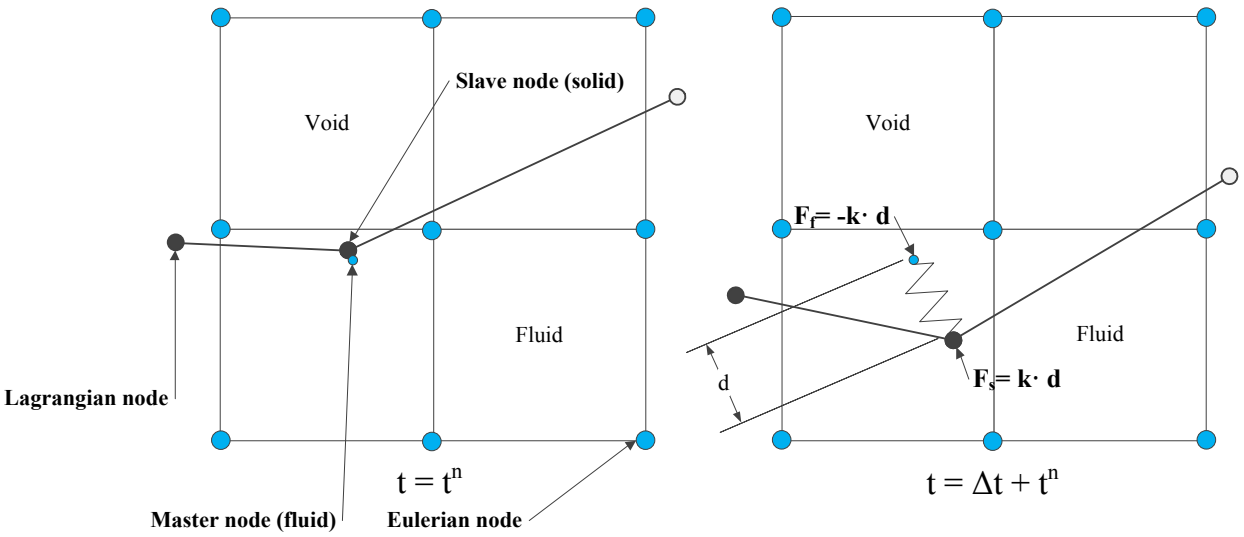

Figure 2.5 - LS-DYNA Penalty Based Coupling

2.8 ALE Multi-Material

In order to distinguish between multiple materials existing in the same element, LS-DYNA uses an ALE multi-material algorithm. The composite stress (σ^*) for an element is calculated by taking a volume-fraction-weighted average across the element. To do this, the number of materials contained in an element (k), the volume-fraction of each material in the element (η) and the stress of each material in the element (σ) are required. by looking at the multi-material ALE element is the volume-fraction-weighted average stress of the individual material stresses [**22**]. The equations are as follows:

$$\sigma^* = \sum_{k=1}^{nmat} \eta_k \sigma_k, \tag{2.31}$$

$$\sum_{k=1}^{nmat} \eta_k = 1, \tag{2.32}$$

where $nmat$ is the number of materials present in a given element [**20**]. An example containing three materials is shown in Figure 2.6. In this example three materials are shown to take space in a single element. First, each material is given a volume-fraction based on the percentage of space they hold in the element. For example, if material 1 was 15% of the element by volume then $\eta_1 = 0.15$. Next, the stresses are given to each material based on the material model equations given in Section 2.6. Finally, the composite stress is calculated using Equation 2.31. This composite stress is applied to the element containing multiple materials. This process is completed for every element that contains more than one material [**20**].

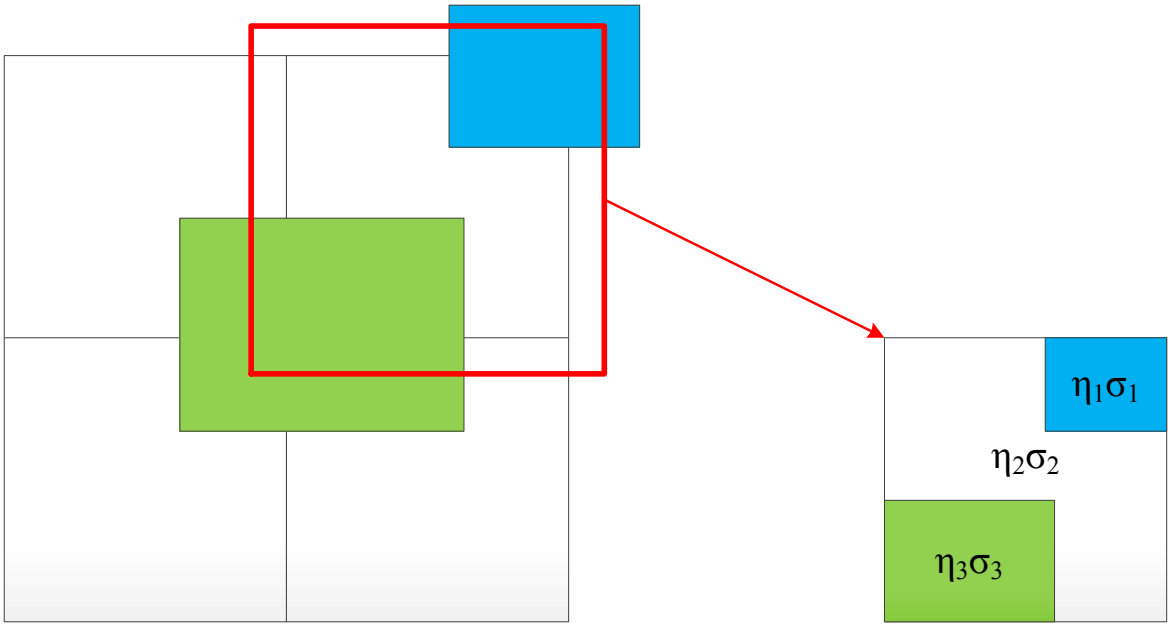

Figure 2.6 - ALE Multi-Material Example

2.9 Mesh Rezoning Algorithm

Eulerian mesh rezoning in LS-DYNA allows for the fluid-mesh to return to its original position after the Lagrangian step (see Figure 2.2). LS-DYNA accomplishes this by saving the global nodal positions in the initial mesh and returning them to that position

after the Lagrangian step [**20**]. Subtracting the initial mesh from the mesh obtain from the Lagrangian step yields the material flux required for advection (Section 2.10) [**20**].

2.10 Advection

Equations used for advection facilitate the movement of material flux through each element. Mass, momentum, and internal energy are all accounted for during the advection of fluid materials [**8**]. For solid materials the mass, momentum, stress, internal energy, and plastic strain are transferred [**8**]. The amount of flux is determined by the mesh rezoning talked about in Section 2.9. Flux is the volume of material moving from one element to another.

For this study the Van Leer algorithm (second order accuracy) is used for the advection calculations. However, the general (first order) advection equations are given to show why they will not be used here. The general advection equations are as follows:

$$s_e \cdot V_e = s_l \cdot V_l + \sum_{faces} s_l^j \cdot Flux_j \tag{2.33}$$

$$\Delta M = \sum_{faces} s_l^j \cdot Flux_j \tag{2.34}$$

where s_e is the Eulerian element that is the target of the advection, V_e is the volume of that Eulerian element, s_l is the Lagrangian element that is the source of advection, V_l is the volume of that Lagrangian element, s_l^j is the average value in the Lagrangian element (on the j^{th} face) which is being advected (the donor cell), and ΔM is the change in the momentum [**8**].

The main concern with this first order accurate equation is that the parameters will dissipate as they are advected through each element face [**8**]. This is the case because as soon as the material is advected into a new element, the material is spread across the entire element length [**8**]. As an example of this dissipative effect we look at the first order momentum equation for s:

$$s = \rho \cdot U \tag{2.35}$$

where ρ is the density and U is the average element velocity in the x, y and z directions. The element velocities is computed by taking the average of the nodal velocities in a given direction. Second, we look at the second order Van Leer momentum equation for s:

$$s = \rho_0 + \frac{1}{2}(X_1 - X_0)\frac{\partial \rho}{\partial X}(X_1), \tag{2.36}$$

where ρ_0 is the initial density and X_n represents an element face. The Van Leer equation uses a parabolic fit to transfer material flux at the element face [**20**]. This formulation allows for a gradual change in material flux as opposed to the instantaneous change in the general algorithm. Figure 2.7 shows an example of both the general and Van Leer algorithm for advection. On the left side of the figure the general equation is used and material flux changes abruptly at each element face. Additionally, the spreading of the value ρ across the element length leads to the dissipative effect. That is, as ρ is transferred successfully to each element face it is reduced. On the right hand side of the figure shows the Van Leer equation is used. Here the material flux changes gradually following the parabolic curve. This in turn leads to the non dissipative transfer of the value ρ. For this reason, the second order Van Leer algorithm is used in this study.

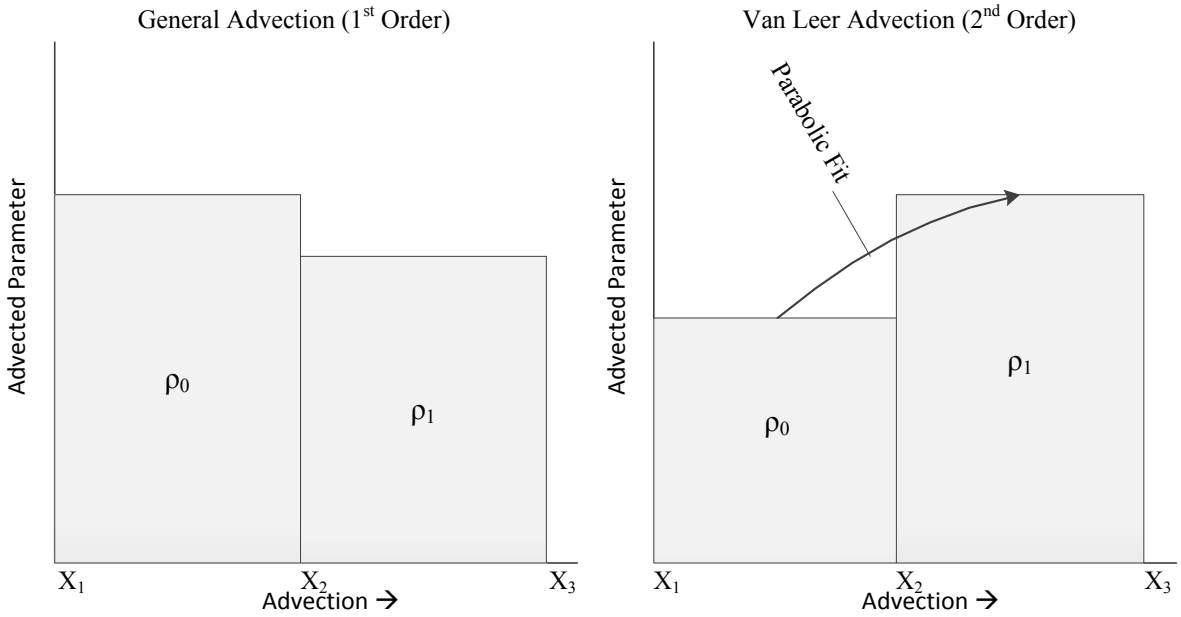

Figure 2.7 - LS-DYNA Advection Examples

2.11 Boundary Conditions

Translational and rotational boundary conditions in LS-DYNA are applied by setting the acceleration component to zero either locally or globally [**20**]. In this study single point constraints are used (boundary conditions are setup uniquely at each node) which forces LS-DYNA to use a local coordinate system [**20**]. The local coordinate system is created by setting up a vector (u_1) in the direction of the local x-axis (x_l) and a local in-plane vector (v_l) [**20**]. Next, the vector u_1 is normalized and the local axis equations are given as [**20**]:

$$x_l = \frac{u_1}{\|u_1\|}, \tag{2.20}$$

$$z_l = \frac{x_l \times v_l}{\|x_l \times v_l\|}, \tag{2.21}$$

$$y_l = z_l \times x_l. \tag{2.22}$$

Next, a transformation matrix (q) is created to change the acceleration components from global to local coordinates [**20**]:

$$q = \begin{bmatrix} x_l^t \\ y_l^t \\ z_l^t \end{bmatrix}. \tag{2.35}$$

Using the transformation matrix the nodal translation (a_I) and rotation ($\dot{\omega}_I$) acceleration vectors are transformed to local coordinates [**20**]:

$$a_{I_l} = q a_I, \tag{2.33}$$

$$\dot{\omega}_{I_l} = q \dot{\omega}_I, \tag{2.34}$$

where I represents the node number. Constrained components are zeroed at this stage in the calculations. Finally, the translation and rotation acceleration vectors are transformed back into the global coordinate system [**20**]:

$$a_I = q^t a_{I_l}, \tag{2.33}$$

$$\dot{\omega}_I = q^t \dot{\omega}_{I_l}. \tag{2.34}$$

Chapter 3 - Model Specifics

A FSI model has been created with the adaptation into LS-DYNA in mind. This chapter discusses the development of the channel model and its transition into LS-DYNA. Model specifics such as dimensions and mesh sizes, boundary conditions, materials, fluid-structure coupling parameters, LS-DYNA controls, equation of state specifics, element formulation, and ambient element type will be discussed. As a note, LS-DYNA specifies parameters and cards (topic with several parameters) by having a '*' in front of their name. Finally, all of the simulations ran in this thesis are discussed.

3.1 Dimensions and Meshing

A continuum of 100mm by 20mm is the base of the channel model. On the leftmost edge of the channel model is the pressurized inflow water, in the center of the channel is the flap (16.5mm x 1mm) hanging from the top, and the remaining area is the initially stationary water, Figure 3.1.

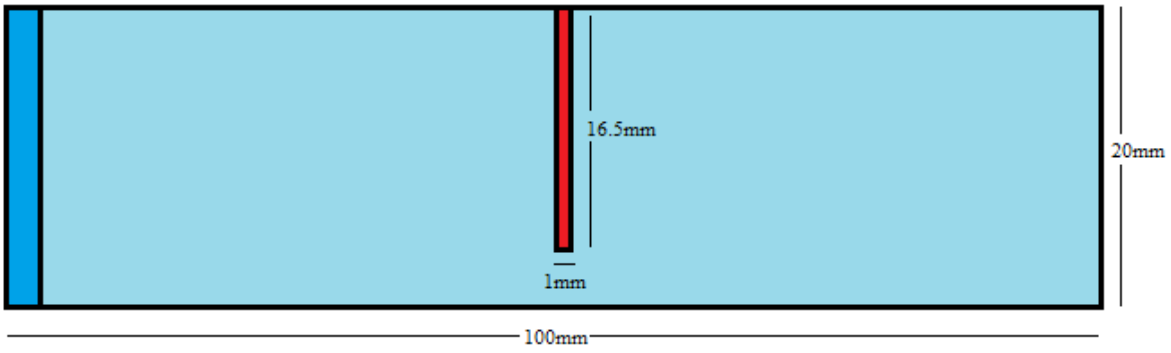

Figure 3.1 - Channel Model

Creation of this model in LS-DYNA was done by creating three part identifications (PIDs), one for each area. Table 3.1 shows the channel model part and its designated ID. These PIDs become a very important aspect of creating the full model in LS-DYNA because they are the reference tag used to match coupling parts and materials to parts, among many other variables.

Table 3.1 - PIDs

PID	Area
1	Stationary water
2	Pressurized source
3	Flap

Four different mesh sizes were used to discretize the continuum. Different mesh sizes were used solely to determine convergence of the system which is discussed further in Section 4.1. PID 1 and PID 2 were meshed together (Mesh 1) while PID 3 (Mesh 2) was meshed independently. The number of elements in each mesh size is shown in Table 3.2.

Table 3.2 - Number of Elements in Each Mesh Size

Mesh	Mesh 1 # elements (N)	Mesh 2 # elements (N)
Coarse	125	10
Base	500	30
Refined	2000	80
Very Refined	8000	200

Images from LS-DYNA of Mesh 1 and Mesh 2 are provided in Figure 3.2 and Figure 3.3 respectively. An example of both meshes together is provided in Figure 3.4 using the base level of discretization.

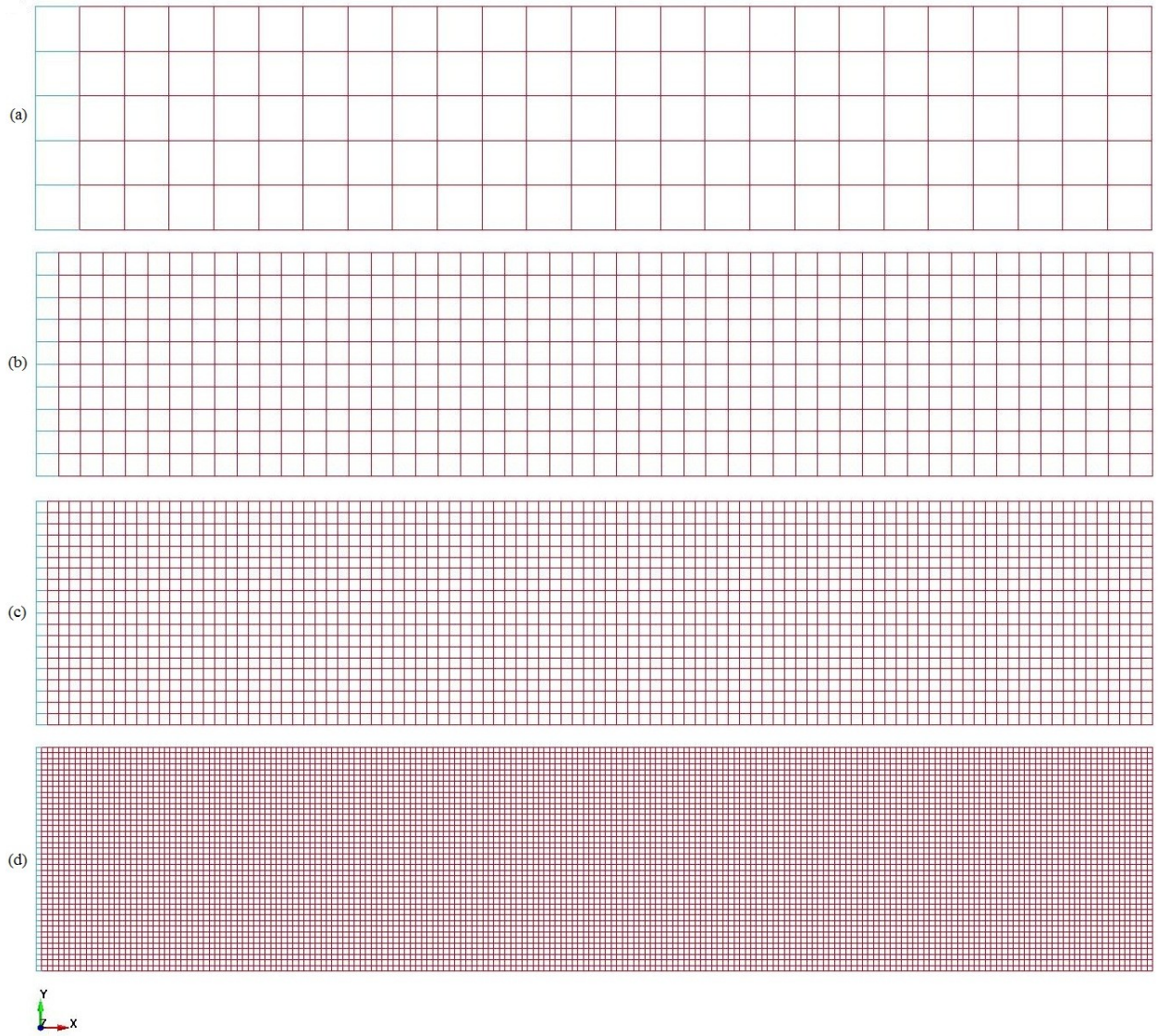

Figure 3.2 - Mesh 1 - # of Elements (a) 125, (b) 500, (c) 2000, (d) 8000

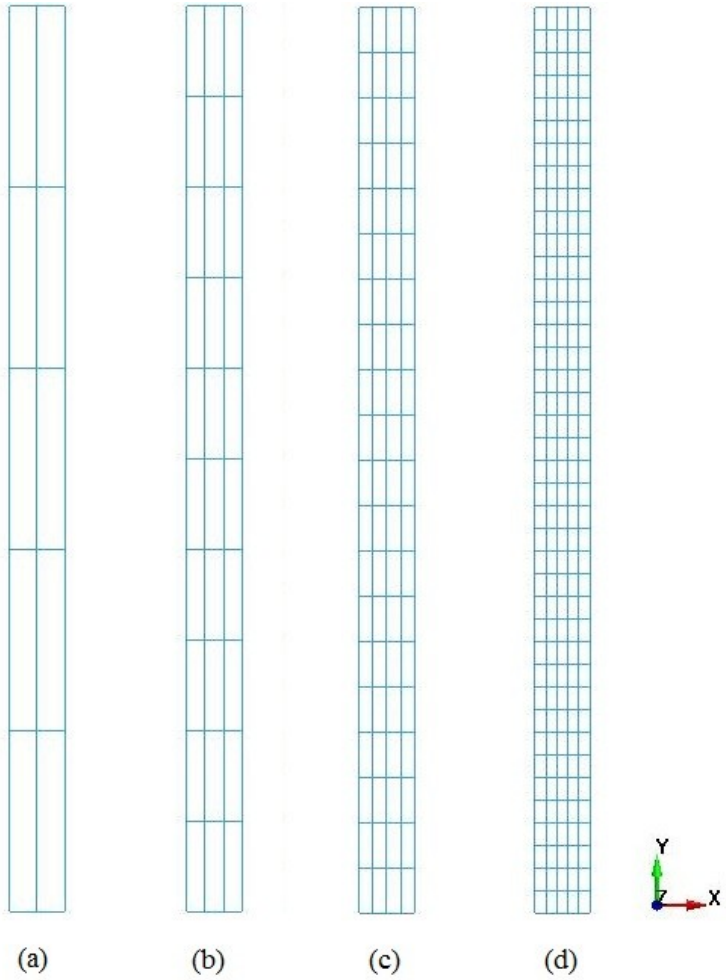

Figure 3.3 - Mesh 2 - # of Elements: (a) 10, (b) 30, (c) 80, (d) 200

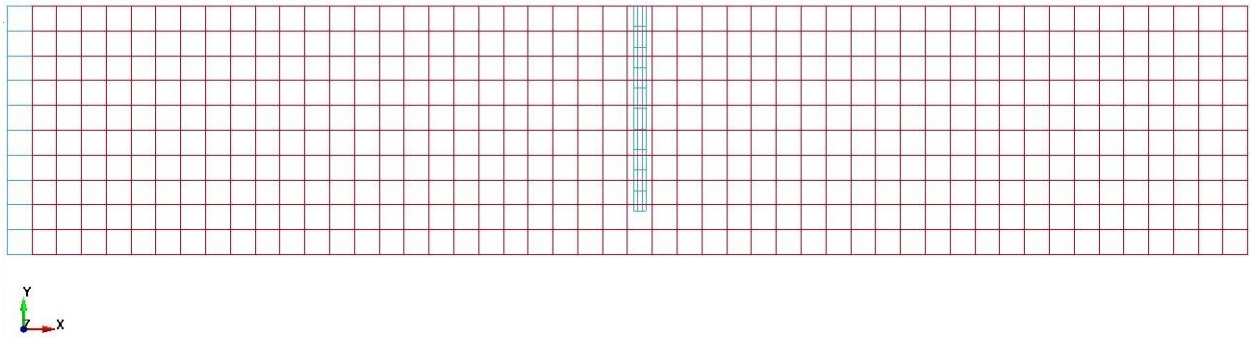

Figure 3.4 - Mesh 2 overlapping Mesh 1 (base mesh)

3.2 Boundary Conditions

The boundary conditions are very important in the model as they dictate how the flap can move or where water can flow. In this model four different types of boundary conditions are imposed on the model, they are presented in Table 3.3. The boundary conditions for mesh 1 and mesh 2 can be seen in Figure 3.5 and Figure 3.6 respectively.

Table 3.3 - Boundary condition types

Type	Boundary Condition
1	Constrained Translation (z), Constrained Rotation (x,y)
2	Constrained Translation (y,z), Constrained Rotation (x,y,z)
3	Constrained Translation (x,z), Constrained Rotation (x,y,z)
4	Constrained Translation (x,y,z), Constrained Rotation (x,y,z)

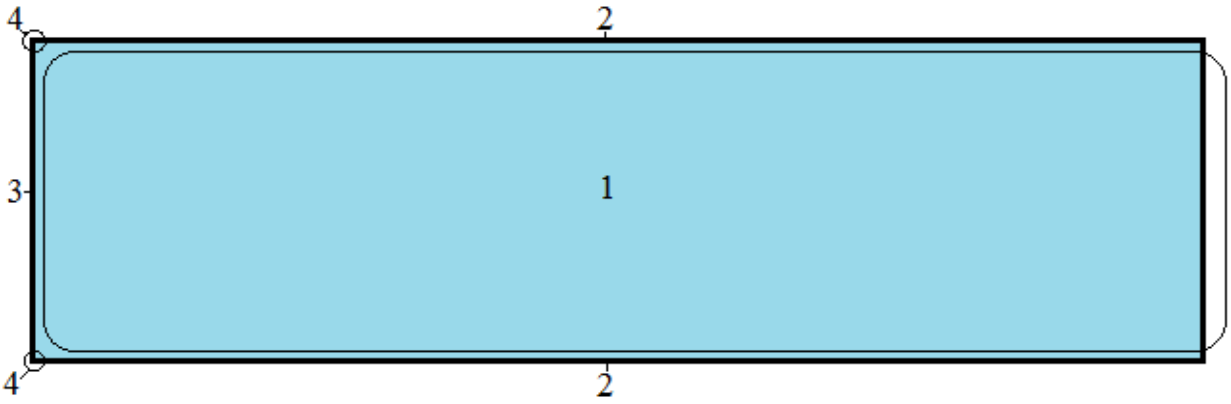

Figure 3.5 - Boundary Conditions for Mesh 1

Boundary type 1 for the fluid continuum allows free flow from the source on the leftmost end to the exit on the rightmost end. Boundary type 2, 3, and 4 provide restrictions that prevent water from leaking out of those nodes. This effectively makes the model equivalent to a pipe.

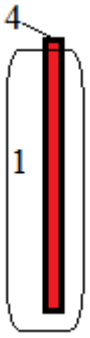

Figure 3.6 - Boundary Conditions for Mesh 2

Boundary type 1 for the flap continuum allows for it to flow freely in the x and y direction. Boundary type 4 clasps the top of the flap to the top of the channel model. All boundary conditions on each mesh was setup manually using the *boundary_spc_node card in LS-DYNA. The *boundary_spc_node card allows for the restriction of translation and rotation in any direction.

3.3 Materials

Two different types of materials are used for the channel model, one for the fluid (mesh 1) and one for the flap (mesh 2). This material allocation can be seen in Figure 3.7.

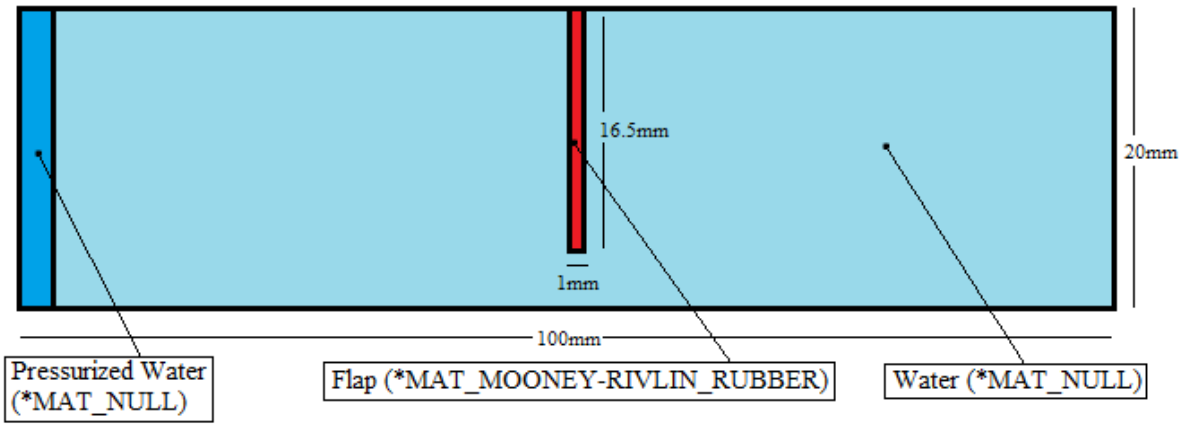

Figure 3.7 - Materials Used in Channel Model

Water is modeled using *MAT_NULL as recommended in the LS-DYNA manual [**19**]. The material card is setup as follows.

Table 3.4 - Fluid Material Card

Parameter	Value	Units
Material ID	1	-
Mass density	0.9982	g/mm^3
Viscosity coefficient	9.982e-009	$ms\text{-}N/mm^2$

The flap is modeled using *MAT_MOONEY-RIVLIN_RUBBER. This material was chosen for its ability to sustain extreme deformations. Two material cards have been created for *MAT_MOONEY-RIVLIN_RUBBER one which is much stiffer than the other. Before each simulation the stiffer or softer material is chosen. The material cards are setup as follows.

Table 3.5 - Flap Material Cards

Parameter	Value	Units
Stiffer card		
Material ID	2	-
Mass density	1.01	g/mm^3
Poisson ratio	0.499	-
Constant A	0.13292	MPa
Constant B	0.0263	MPa
Softer card		
Material ID	3	-
Mass density	1.01	g/mm^3
Poisson ratio	0.499	-
Constant A	0.013292	MPa
Constant B	0.00263	MPa

3.4 Fluid-structure coupling

To allow for the interaction of the fluid (mesh 1) with the flap (mesh 2) a coupling algorithm is required. In LS-DYNA this is done using the *constrained_lagrange_in_solid card. This card is required because the fluid and flap mesh do not share nodes, it ensures the transfer of forces and the resistance of penetration through the solid. This section describes the variables chosen in the card to ensure good coupling forces.

First a slave and master is chosen that dictates which parts will interact. The slave is always set up as the movable piece, so in this case the flap is chosen. The master must be the combination of both the fluid and the pressurized fluid source in order to contain all fluid parts. This is accomplished by creating a part list using the *set_part_list card. Variables used after setting up the slave and master are shown in Table 3.6.

Table 3.6 - Coupling Variables

Variable	Value
NQUAD	2
CTYPE	4
DIREC	2

3.5 LS-DYNA Controls

Several controls are used within LS-DYNA that take care of background simulation variables. Five main control cards are used in this model, they are *control_ale, *control_contact, *control_cpu, *control_termination, and *control_timestep.

The purpose of *control_ale is to set global parameters for ALE and Eulerian calculations [**19**]. The parameters accompanied with descriptions used for this model are as follows.

Table 3.7 - *control_ale Card

Parameter	Value	Description
NADV	1	Number of cycles between advections
Advection Method	2	Equation:Van Leer
ALE smoothing factor	-1	Off (No ALE rezoning)

The purpose of *control_contact is to setup parameters related to contact surfaces in the model (flap). The parameters used are as follows.

Table 3.8 - *control_contact Card

Parameter	Value	Description
ISLCHK	1	No initial penetration checking
ORIEN	1	Contact surfaces are given by *PART definition

The *control_cpu card is used to tell to terminate the simulation at a specific time. For this model, the time is set to 0.0 which allows it can run for an unlimited amount of time or until it reaches the simulation termination time. The *control_termination and *control_timestep card controls the simulation run-time and the time step used for each calculation. The total termination times are given in Table 3.9. The time step is calculated automatically within LS-DYNA based on the size of the smallest element [**20**].

Table 3.9 - *control_termination Card

Scenario	Time (ms)
Constant Flow	10000 to 20000
Pulsatile Flow	30000

3.6 Equation of State

The Gruneisen EOS is used to apply a pressurized source to PID 2. In this study two variation of the Gruneisen EOS are required, one for a constant pressure and the other for a time varied pressure. For a constant pressure the *eos_gruneisen card is used and connected to PID2. The card used for constant pressure can be seen in Table 3.10.

Table 3.10 - *eos_gruneisen Card for Constant Flow

Parameter	Value	Units
C	0.165	-
S1	1.92	-
Gamma	0.1	-
E0	0.005	N-mm

For varied pressure over time the *eos_gruneisen card in conjunction with the *boundary_ambient_eos and *define_curve card is used. For *eos_gruneisen the same values as shown in Table 3.11 are used with the exception that E0 is now set to 0 (no initial internal energy). The *boundary_ambient_eos card controls the value of E (internal energy) over a period of time using the *define_curve card. Within the *define_curve card a half period sin curve has been setup with a peak of 0.01 N-mm.

3.7 Element Formulation and Ambient Element Type

Each part of the channel model contains its own unique element formulation and ambient element type that coincide with its purpose. Each part uses 8-noded hexahedrons created

in *element_solid to define the geometry of the system. It was possible to choose triangle-type elements for the flap, however, doing so could potentially cause elements to be overly stiff [**20**].

The element formulation and ambient element type for each PID is selected using the *section_solid_ale card for the water elements (PID 1 and PID2) and *section_solid for the flap elements (PID3). Element formulation 11 (ELFORM 11), 1-point multi-material ALE, has been chosen for the water elements. This formulation allows for PID1 and PID2 to share the same element which allows for the advection of the pressurized water into the original PID2 domain. Element formulation 1 (ELFORM 1), constant stress solid element, has been chosen for the flap elements. The ambient element type is only applicable to PID 2 (pressurized source) and allows for it to act as a pressure inflow. This is accomplished by setting the ambient element type to 4. A summary of these formulations is shown in Table 3.11.

Table 3.11 - Element Details

PID	Element type	Element formulation	Ambient element type
1	8-noded hexahedron	ELFORM 11	-
2	8-noded hexahedron	ELFORM 11	4
3	8-noded hexahedron	ELFORM 1	-

3.8 Simulations

A wide range of simulations have been completed to test for physics, convergence and to study the feasibility of using LS-DYNA in large-displacement FSI. Variables in each simulation can be mesh size, flow type (constant or time variant), material type, and termination time. A summary of the simulations is provided in Table 3.12.

The physics and convergence analysis consists of eight simulations. Four of the simulations use a constant flow scheme, the stiffer material card, and vary in mesh size; the remaining four simulations use a constant flow scheme, the softer material card and vary in mesh size. First, a physical analysis of simulations 3 and 7 is completed. Second, simulations 1 to 8 are tested for the convergence of the resultant displacement (at the flap tip) at steady-state and over time. Third, the steady-state value of the resultant

displacement is compared as the mesh size increases for a given material type. An error relative to the most refined mesh size for a given material type is computed in this case. Lastly, the velocity contour plot is compared as the mesh size increases for a given material type.

For simulation 9, the channel model is modified to correlate with a study completed by [**18**]. Here the flap is extended to the bottom of the channel model and pressure is tested on the left and right hand side of the flap.

Finally, simulation 10 and 11 were created for the purpose of testing the limits of LS-DYNA's FSI. In these simulation the material has been modified to provide a flap with extreme motion over the duration of the simulation time. This was accomplished by greatly increasing the density in the material card to 101 g/mm^3. Simulation 10 is a constant flow scheme and Simulation 11 is a pulsatile flow scheme. The purpose of this simulation was to test contact with the channel wall and extreme deformations.

Table 3.12 - Simulations

Simulation	Flow	Material Card	Mesh	Time (ms)	Purpose
1	Constant	Stiff	Coarse	10000	Convergence
2	Constant	Stiff	Base	10000	Convergence
3	Constant	Stiff	Refined	10000	Physics/Convergence
4	Constant	Stiff	Very Refined	10000	Convergence
5	Constant	Soft	Coarse	10000	Convergence
6	Constant	Soft	Base	10000	Convergence
7	Constant	Soft	Refined	10000	Physics/Convergence
8	Constant	Soft	Very Refined	10000	Convergence
9	Constant	Soft	Base	20000	Pressure Test
10	Constant	Soft (101 g/mm^3)	Base	20000	Physical Analysis
11	Pulsatile	Soft (101 g/mm^3)	Base	30000	Physical Analysis

Chapter 4 - Results

4.1 Stiff Flap

This section provides an in-depth look at the physical analysis and convergence test for the stiff flap (Simulations 1 through 4).

4.1.1 Physical Analysis

A qualitative approach has been used to analyze the physical characteristics of the stiff flap. Simulation 3 was chosen for the physical analysis of the stiff flap because it showed good convergence (see Section 4.1.2). Figure 4.1, 4.2, 4.3 and 4.4 show the stages of the simulation from the beginning and onwards, until steady-state is reached. The figures show a velocity profile for the fluid mesh along with the physical movement of the stiff flap.

Figure 4.1 begins with the initialization of the problem. The channel shows no signs of flow at this point and the flap is in its initial position setup in the geometry of the problem. Shortly after initialization of the channel model, the fluid begins to flow. This can be seen by looking at the slight change in colour of the velocity contour left of the flap. As the fluid continues to flow it eventually hits the flap at a time of 500ms. Contours of velocity begin to increase in magnitude at the bottom of the flap because it is the only flow path available. As a result of the impacting fluid flow, the flap begins to shift to the right. As the flow continues to build up on the left side of the flap, higher velocities are seen throughout. As the velocities become higher the flap displacement also continues to increase. The velocities at the top of the flap continue to be near-zero.

Figure 4.2 continues similarly to Figure 4.1. Velocities under the flap continue to increase as a result of the fluid build up on the left side of the channel. As a result of the flow increasing in the channel, the near-zero velocity area around the top of the flap begin to dissipate on the left side. This is due to the increased activity on the left side, it was inevitable that the flow would creep towards that corner. The shape of the contours in that same area are oddly shaped (spikes) around 2000ms. The reason for the contour shapes is unknown, however, as the simulation progresses they disappear.

In Figure 4.3 the velocities continue to increase in the channel model. The flap no longer moves because it has reached steady-state. The near-zero flow directly right of the flap expands towards the channel model exit as the fluid flow under the flap becomes more streamlined in one direction. The cause for the streamlined velocity is the stable positioning of the flap. At the end of Figure 4.3 the flap reaches its steady-state displacement. Finally, in Figure 4.4 the velocities reach their peak and the velocity contours reach steady-state.

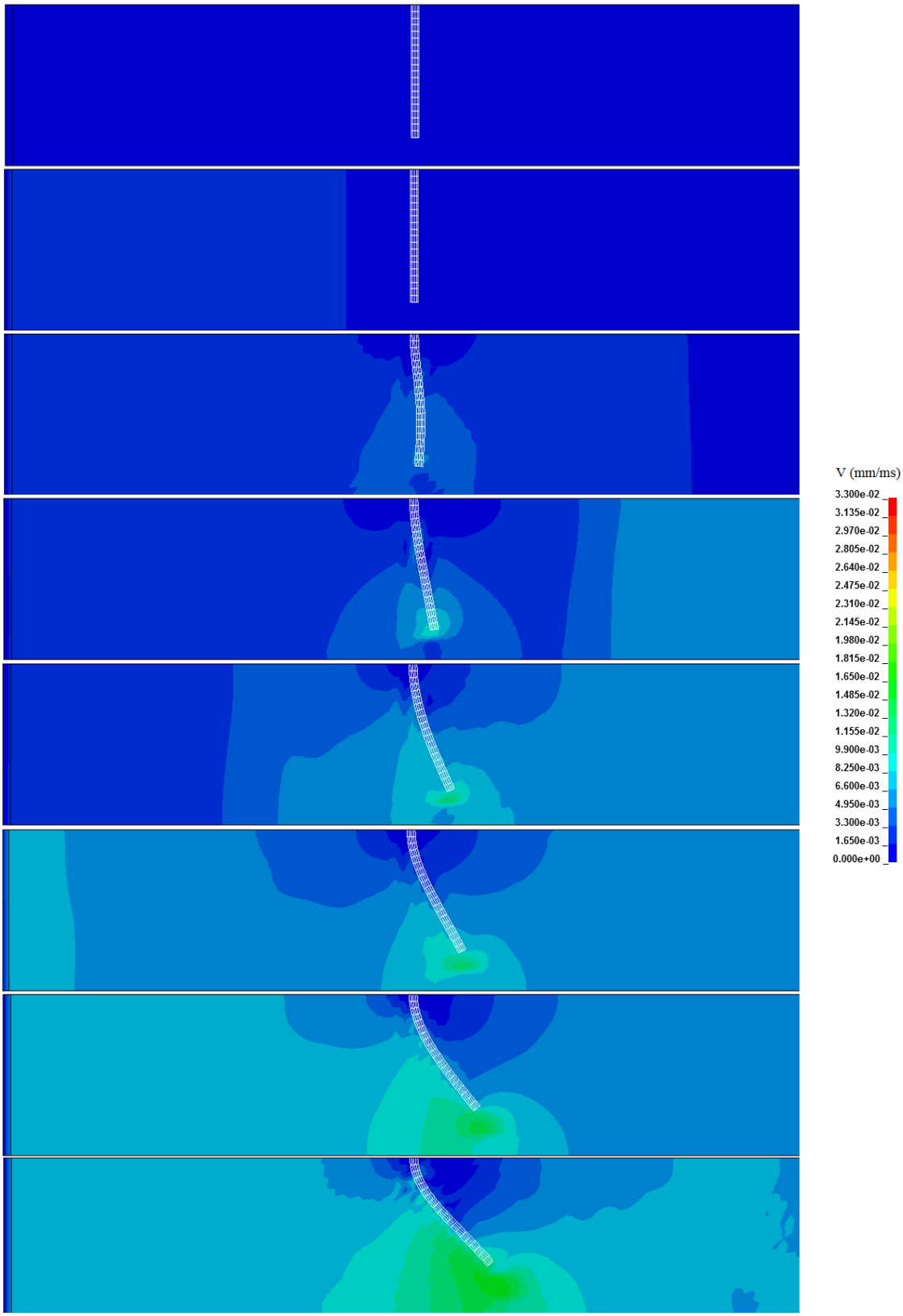

Figure 4.1 - Stiff Flap, Constant Flow: 0 to 1750ms, 250ms Spacing

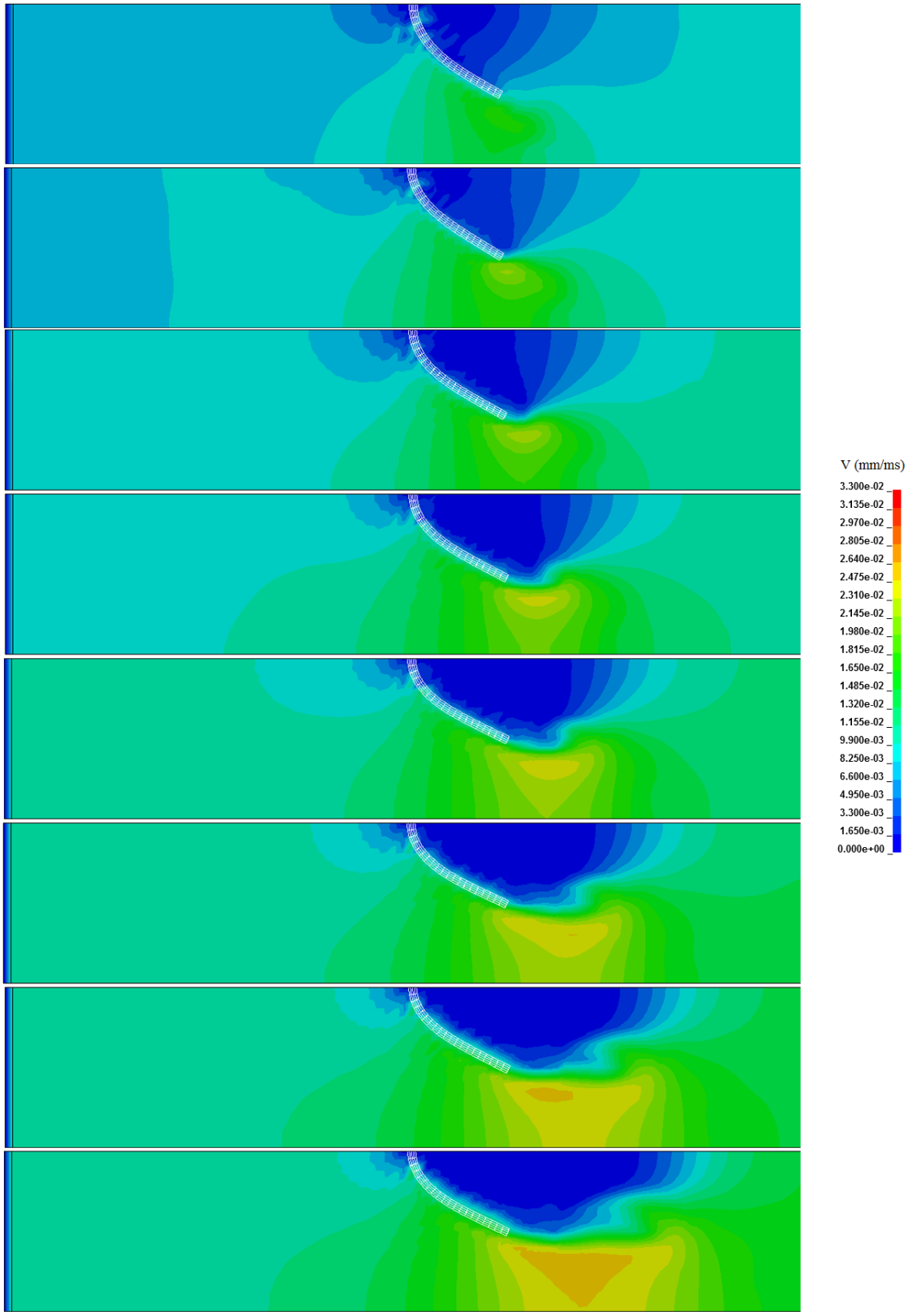

Figure 4.2 - Stiff Flap, Constant Flow: 2000 to 3750ms, 250ms Spacing

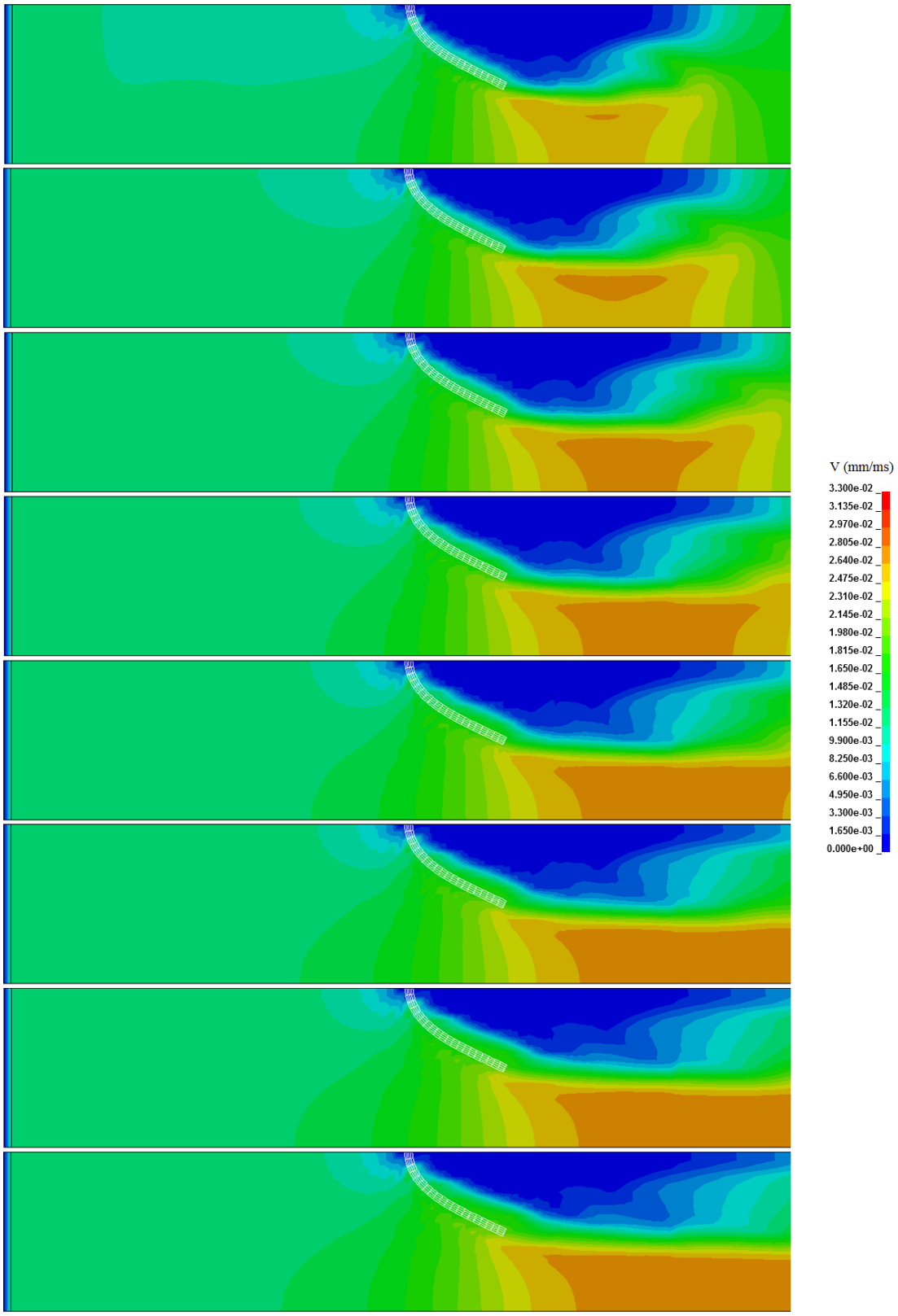

Figure 4.3 - Stiff Flap, Constant Flow: 4000 to 5750ms, 250ms Spacing

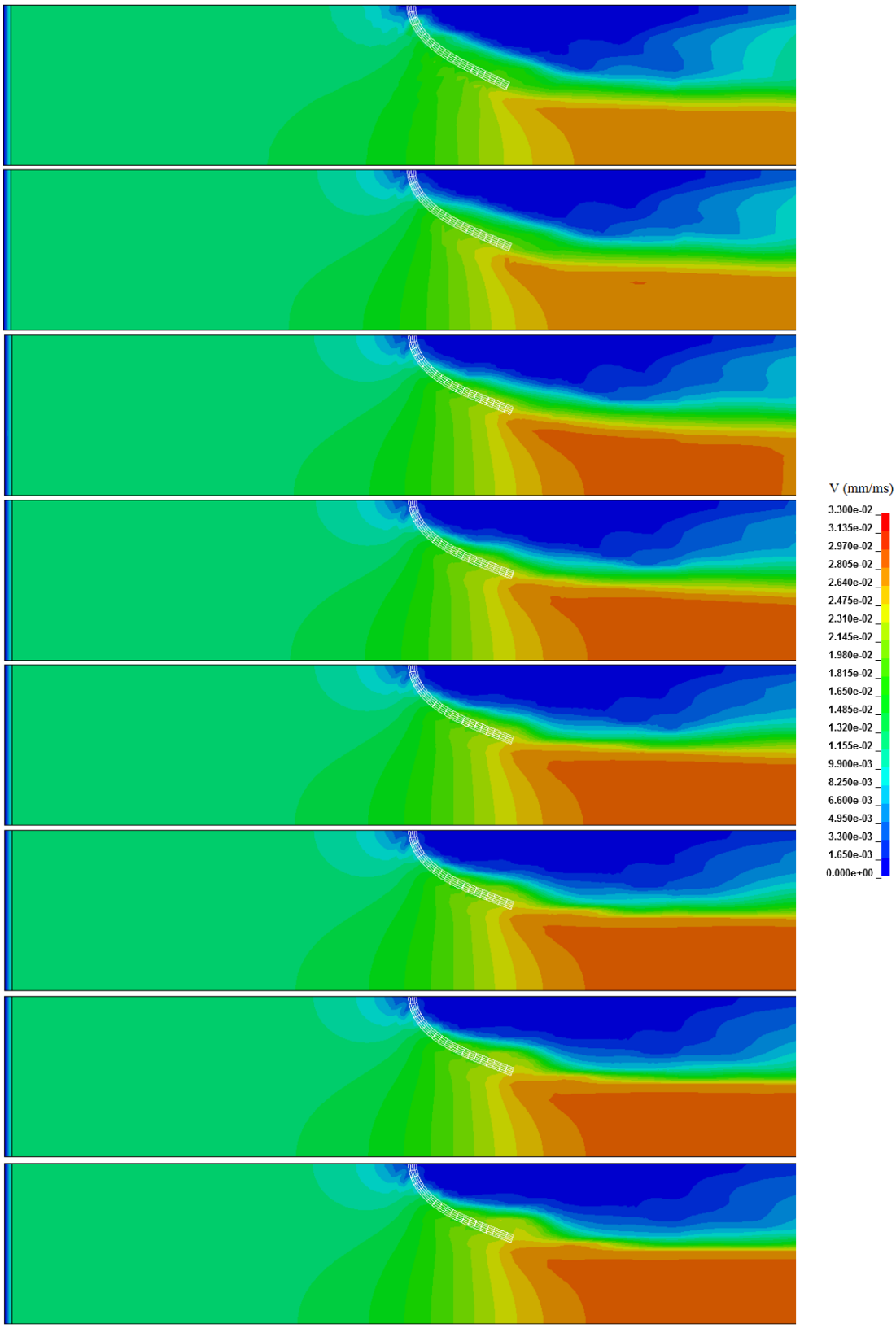

Figure 4.4 - Stiff Flap, Constant Flow: 6000 to 9500ms, 500ms Spacing

4.1.2 Convergence

The velocity contours at time 1500, 5000, and 8500 ms is shown in Figure 4.5, Figure 4.6, and Figure 4.7. Each figure is a combination of four pictures which represent the four mesh sizes used in this study. From top to bottom they are the coarse, base, refined, and very refined mesh.

At time 1500ms the pressure source from the left side has begun to show significant interaction with flexible flap. Visually each flap appears to have similar resultant displacements. Velocities increase as the flow reaches the more narrow path under the flap which is an expected result derived from fluid continuum mechanics. The major differences at this stage of the simulation can be seen in the velocity contours. This difference is most evident in the coarse mesh where additional contours can be seen closer to the pressure source. As the mesh becomes more refined the contours appear to converge.

At time 5000ms the flap starts to stabilize at its maximum displacement for the simulation. At this time the differences in the contours is more prevalent, especially behind the flap. In this area the velocities are much higher for the coarse mesh and appear to dissipate as it becomes more refined. High velocities in this location could impose unwarranted forces opposing the flap motion, however, they do not appear to have an impact on the flap displacement in this case. Again the contours converge as the mesh becomes more refined.

At time 8500ms the flap has stabilized at its maximum displacement. The contours at this stage of the simulation are very similar to the 5000ms time frame, except for a few major differences. First, the area behind the flap in the coarse mesh still contains the high velocity but it is now elongated following the path of the pipeline lengthwise. Second, in the base mesh there appears to be a very high velocity leak. This is a result of poor coupling forces in the lesser refined meshes. This becomes a non-issue because it disappears with more refinement. Lastly, the velocity on a whole has increased and stabilized. The contours in this case also converge.

To further test for convergence the resultant displacement over time and at steady-state of the flap is examined. Figure 4.8 shows a graph of the resultant displacement over time for each mesh size. As time progresses it is evident that every mesh converges to a steady-state value. As the mesh becomes more refined the displacement over time converges. That is, the coarse mesh is the least like the very refined mesh, whereas, the refined mesh is close to the very refined mesh.

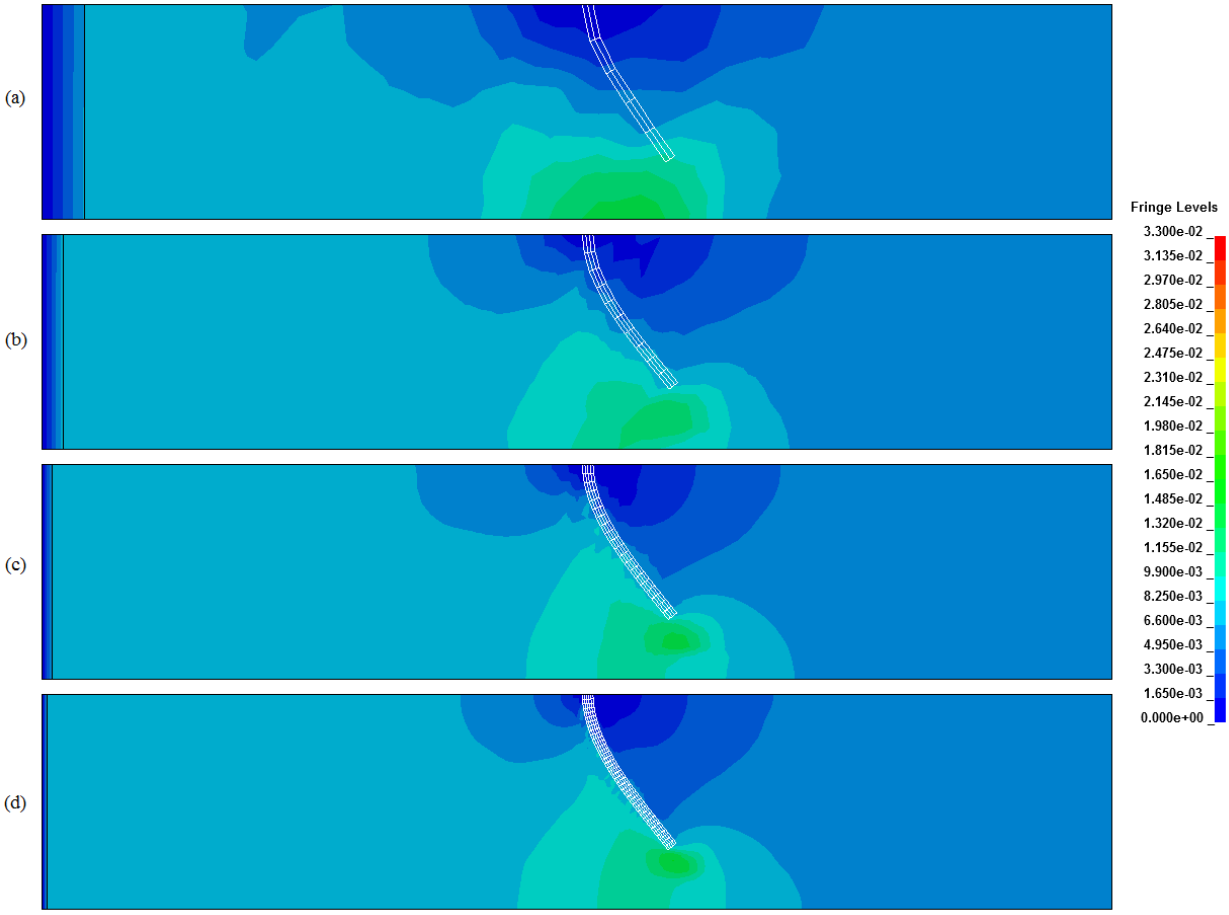

Figure 4.5 - Stiff Flap, Constant Flow: 1500ms, Simulations (a)1, (b)2, (c)3, (d)4

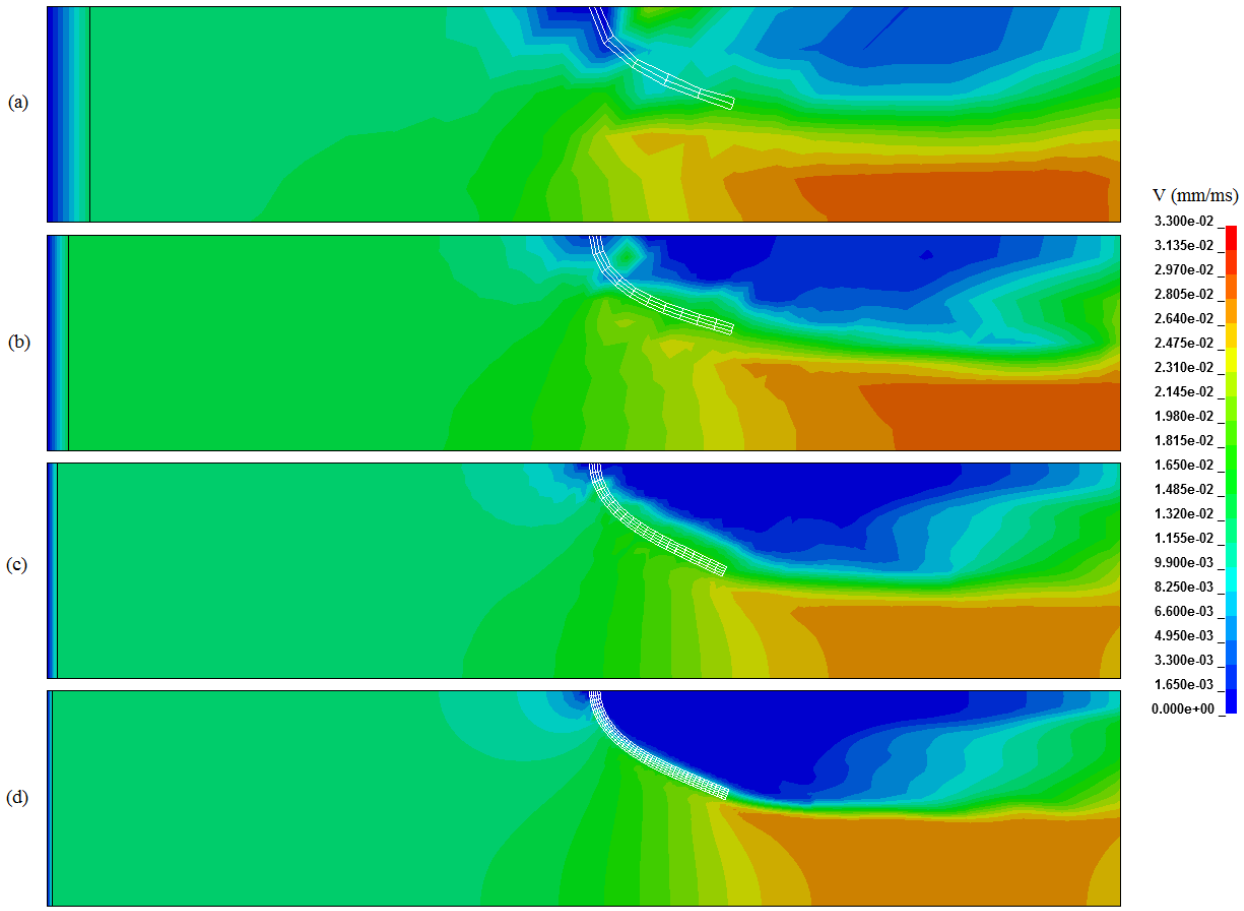

Figure 4.6 - Stiff Flap, Constant Flow : 5000ms, Simulations (a)1, (b)2, (c)3, (d)4

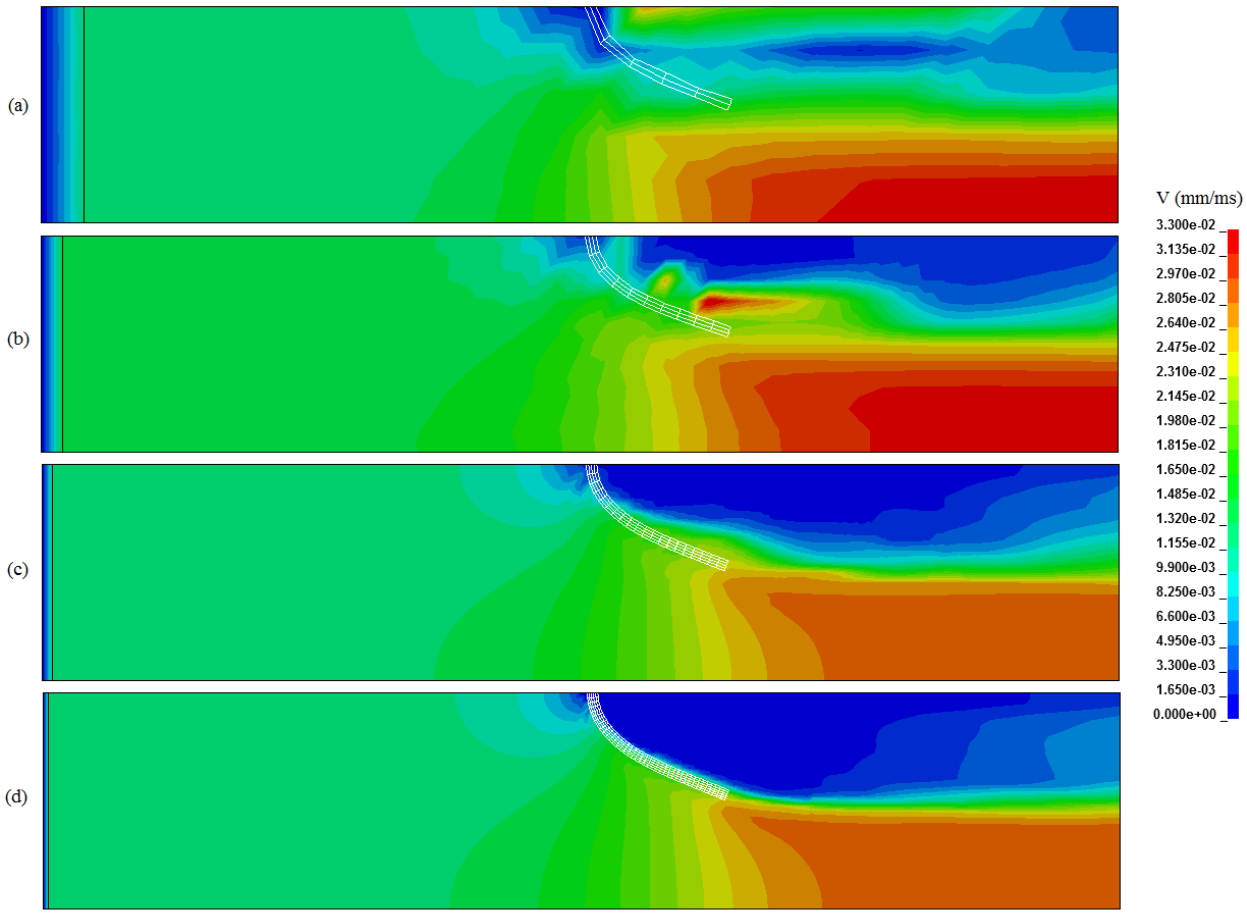

Figure 4.7 - Stiff Flap, Constant Flow: 8500ms, Simulations (a)1, (b)2, (c)3, (d)4

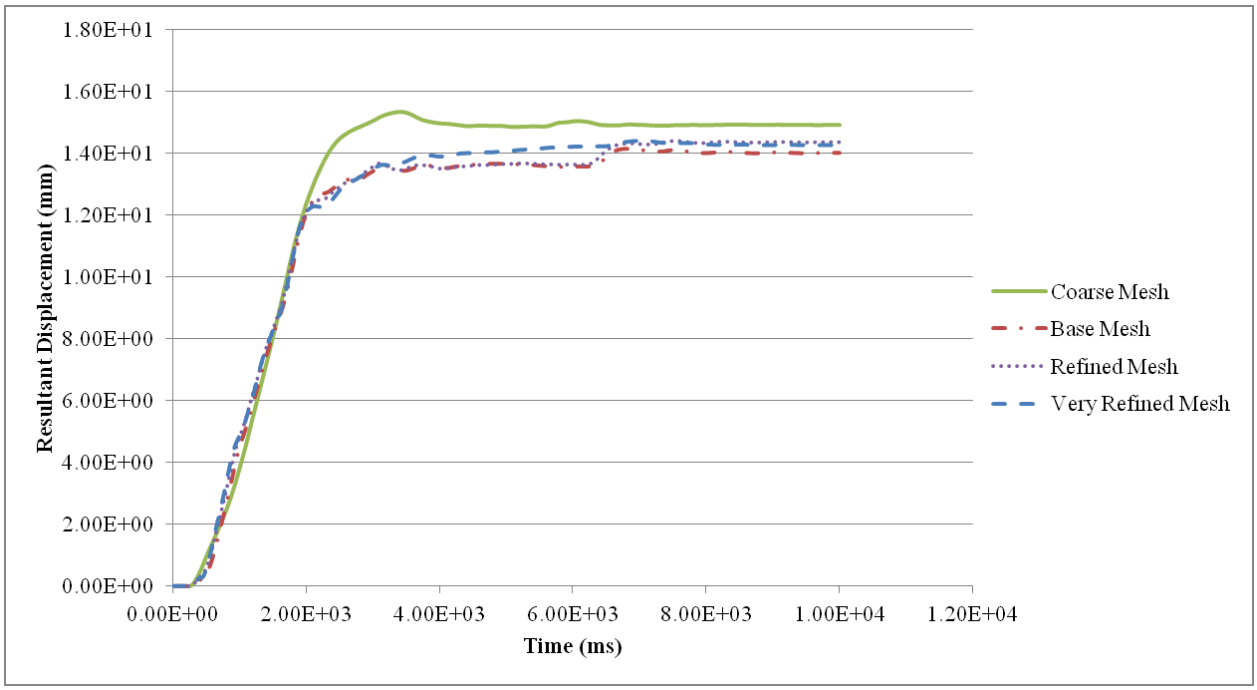

Figure 4.8 - Simulation 1-4: Resultant Displacement vs. Time

This can be seen in the steady-state resultant displacement of the coarse, base, and refined mesh which are compared to the very refined mesh. An error relative to the very refined mesh (Eq 4.1) is computed.

$$Error\ \% = \frac{(A - B)}{A} * 100 \tag{4.1}$$

Where A and B are the steady-state resultant displacements for the very refined mesh and other mesh respectively. The computed errors are shown in Table 4.1.

Table 4.1 - Relative Error for the Stiff Material Card

Mesh	Steady-state Resultant Displacement	Error relative to the Very Refined mesh (%)
Coarse	14.90722	4.47
Base	14.01292	1.80
Refined	14.35158	0.58
Very Refined	14.26907	-

The errors computed for the stiff material under constant pressure show excellent convergence as the mesh size becomes more refined.

4.2 Soft Flap

This section provides an in-depth look at the physical analysis and convergence test for the soft flap (Simulations 5 through 8).

4.2.1 Physical Analysis

A qualitative approach has been used to analyze the physical characteristics of the soft flap. Additionally, physical differences between the stiff and soft flap will be discussed here. Simulation 7 was chosen for the physical analysis of the stiff flap because it showed good convergence for flap resultant displacement (see Section 4.2.2). Figure 4.9, 4.10, 4.11 and 4.12 show the stages of the simulation from the beginning and onwards, until steady-state is reached. Lastly, Figure 4.13 shows a comparison of the soft flap dynamics to work done by [**23**].

The first two frames of Figure 4.9 are equal to those shown in Figure 4.1. This is expected as no interaction with the flap has yet to occur. The third frame, when the fluid flow initially hits soft flap is also very similar to what is shown in Figure 4.1. The soft flap is shifted slightly to the right and concaves slightly. In the fourth frame we notice the first major difference between the stiff and soft flap. In this frame (and several frames after) the soft flap continues to hold the concave shape while the stiff flap does not. Overall, the shape of the velocity contours are very similar between the stiff and soft flap. However, looking closely at the contours shows that the soft flap has lower values for velocity.

In Figure 4.10 the concave shape changes to the standard shape (curved edge in the direction of flow) between frame one and two. The displacement of the soft flap is much greater than the stiff flap as expected.

In Figure 4.11 and 4.12 the velocity contours advance to steady-state. Overall the magnitude of velocity for the soft flap are lower than the stiff flap. This lower velocity is a by-product of the higher resultant-displacement in the soft flap causing a larger pathway for the fluid to flow through. Identical to the stiff flap, the soft flap reaches its steady-state resultant displacement by the last frame of Figure 4.11. In Figure 4.12 an example of leakage occurs on the right hand side of the soft flap. A larger than normal velocity contour spawns there as a result of insufficient coupling forces. Increasing coupling

forces would fix this problem, however, it was found during convergence testing that further refinement of the mesh resolves the issue.

In Figure 4.13, selected frames from the channel model are compared to [**23**]. In [**23**] a modification of the fictitious domain method is used to model flap dynamics in a channel. Frames from [**23**] showing velocity vectors are given on the right hand side of the figure. Exact material parameters and velocity values were not provided in the work making a quantitative analysis not possible. The qualitative comparison here examines the shapes of the flap along with the relative magnitude of the velocity.

The first frame in Figure 4.13 shows the initial flow phase of the simulation. Both flaps at this frame have a similar concave shape with the tail of the flap falling behind the center. The second frame follows briefly after the first in both cases. At this frame the concave shape holds for both simulations at a similar displacement. Additionally, velocity magnitudes at the top of the flap are both low as a result of minimum flow in that area. The third frame shows the point in time where the flap begins to straighten out, removing the concave shape. In this frame, the curvature of the flaps differ slightly. In [**23**] the flap at this frame is straight, however, in the work presented here it is curved slightly. This is the case because in the channel model presented in this paper, the flap straightens out at an earlier frame when the resultant displacement is not as high. This can be seen in the last frame of Figure 4.9. The fourth frame of Figure 4.13 shows the flap nearing its maximum displacement. Curvature of the flap in this case is very similar and both flaps are close to parallel with the upper and bottom channel walls. Velocity magnitudes at this frame are also very similar, with the high velocity stream on the bottom and near-zero levels to the right of the flap. It can be said that both models are very similar with the exception of the flap straightening time. The exception could be a result of differing flap materials or flap width.

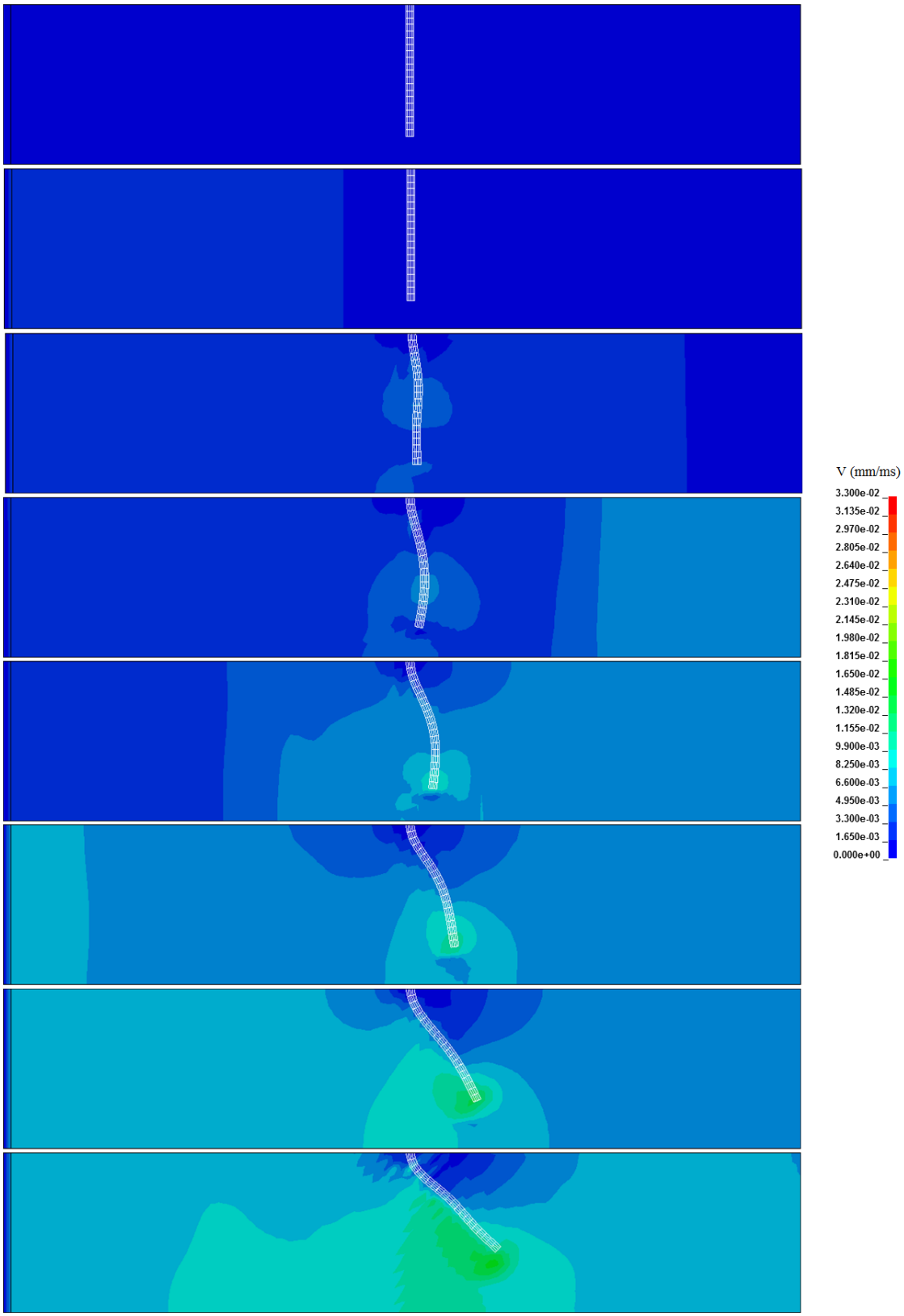

Figure 4.9 - Soft Flap, Constant Flow: 0 to 1750ms, 250ms Spacing

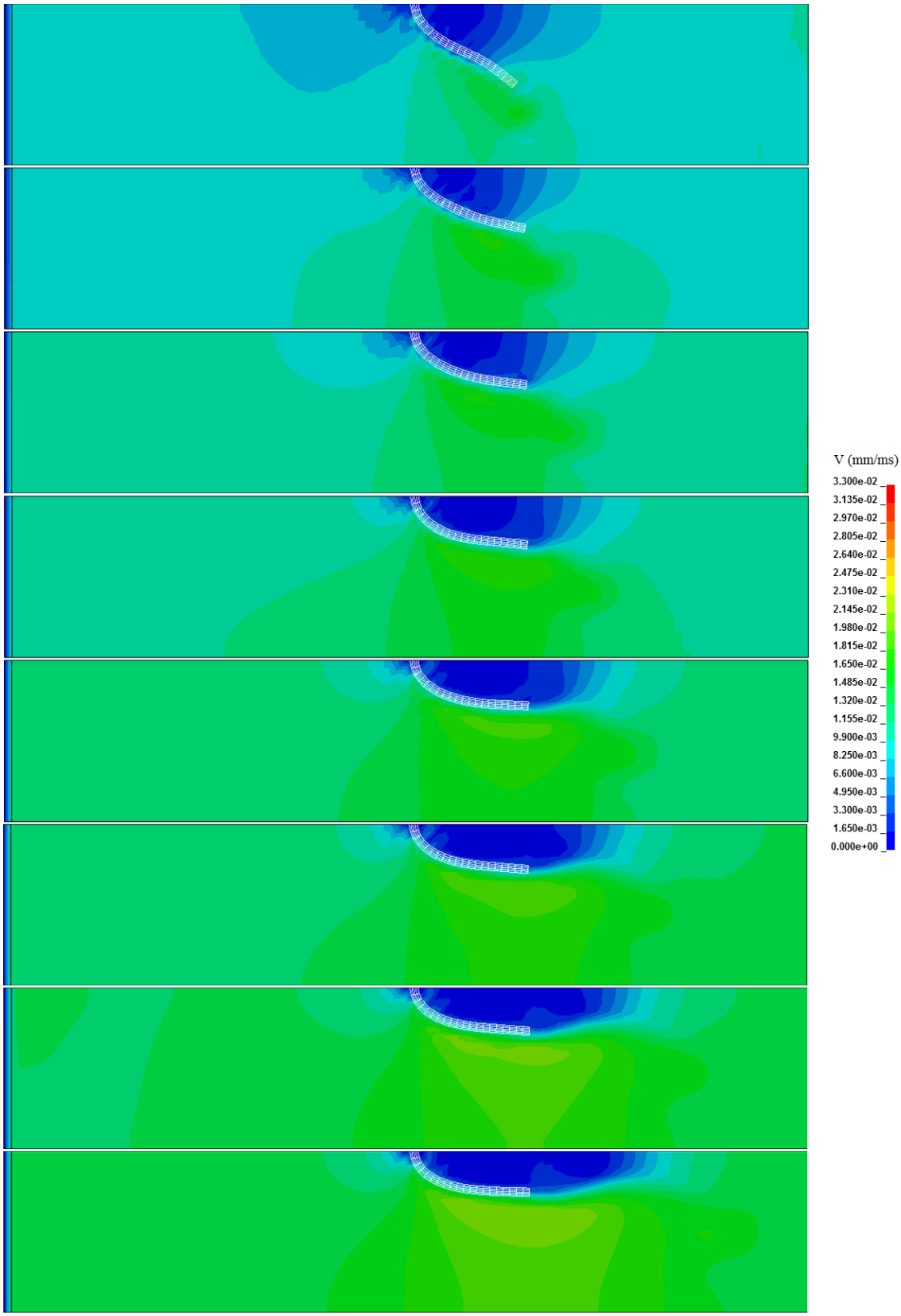

Figure 4.10 - Soft Flap, Constant Flow: 2000 to 3750ms, 250ms Spacing

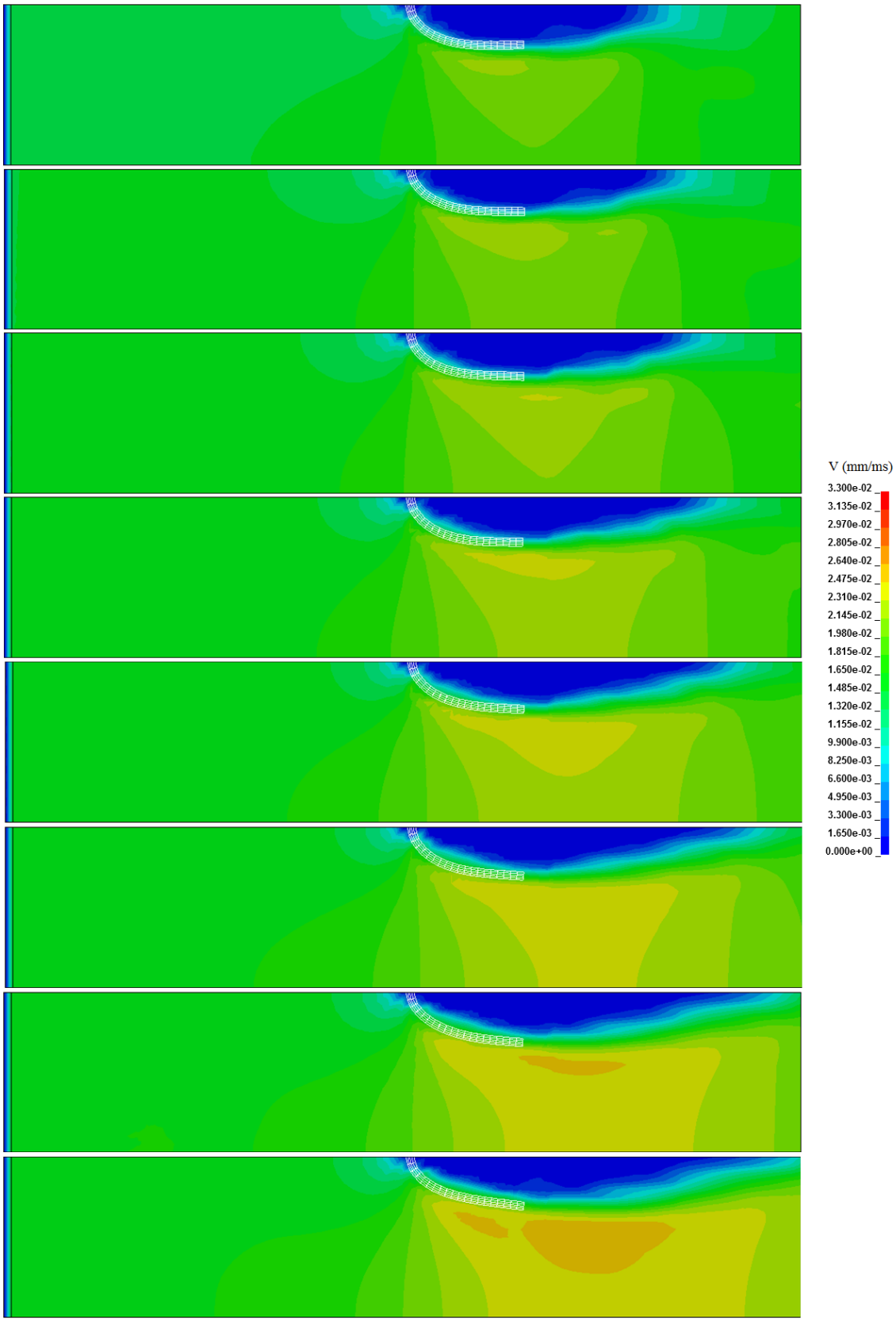

Figure 4.11 - Soft Flap, Constant Flow: 4000 to 5750ms, 250ms Spacing

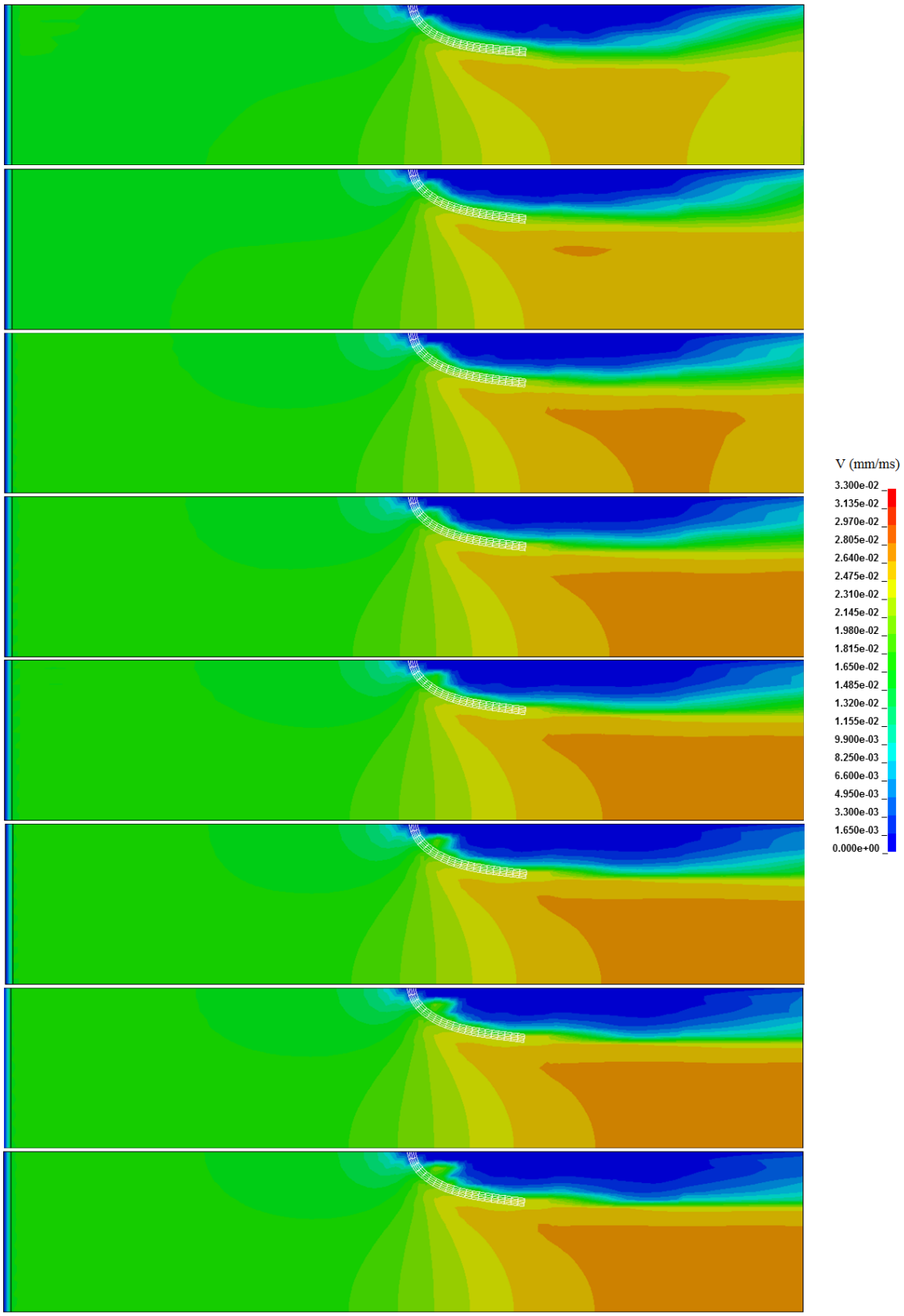

Figure 4.12 - Soft Flap, Constant Flow: 6000 to 9500ms, 250ms Spacing

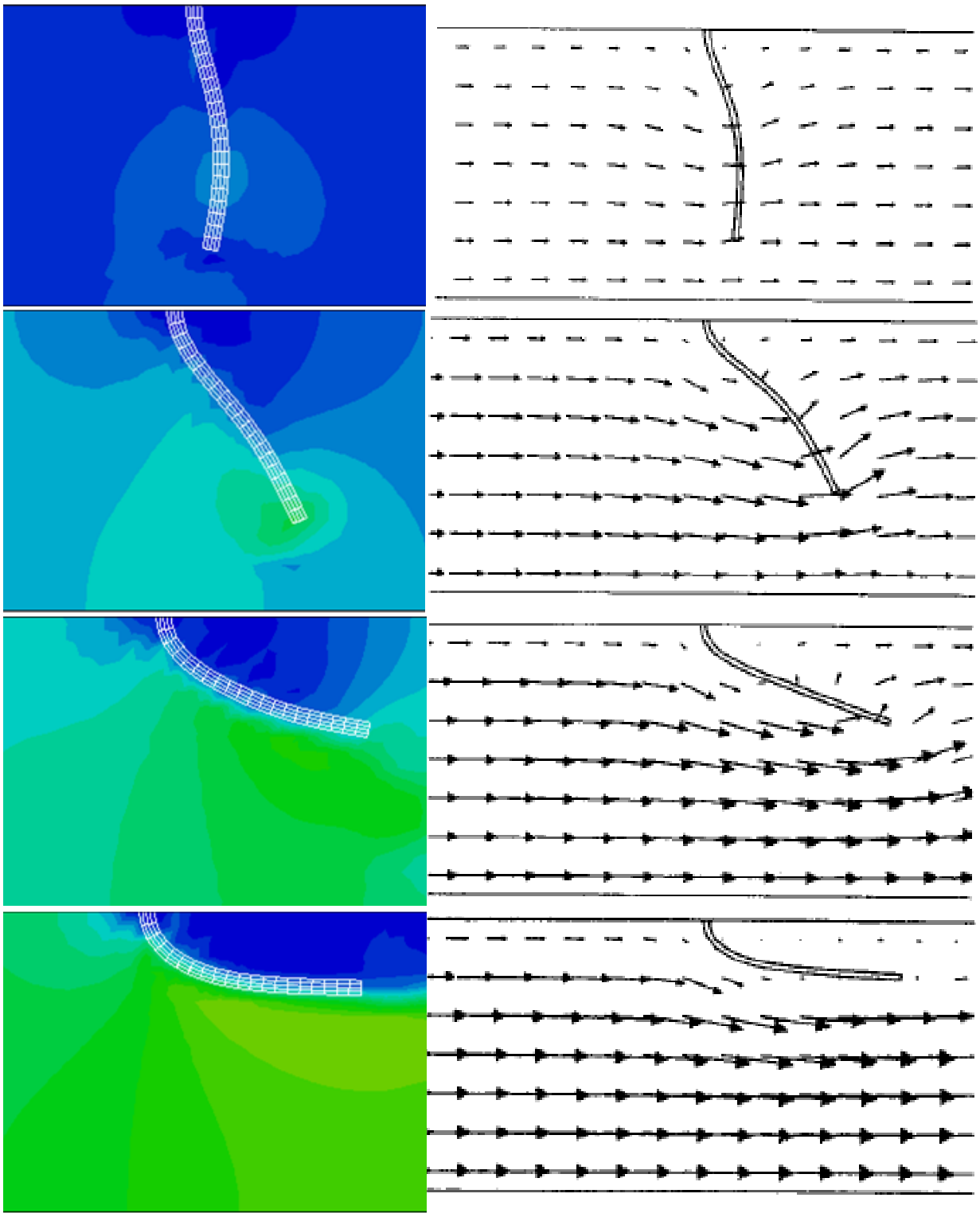

Figure 4.13 - Comparison of Flap Dynamics at Selected Frames [23]

4.2.2 Convergence

Similar to the stiffer flap, the velocity profile at time 1500, 5000, and 8500 ms is shown in Figure 4.14, Figure 4.15, and Figure 4.16. The progression of mesh refinement is analyzed for each time frame. In all cases the velocity contours converge as the mesh is refined.

At time 1500ms the flap has started to ascend as a result of the pressurized source. At this stage the flap behaves much differently than the stiffer flap. The center tends to concave and the tip falls behind it.

At time 5000ms the flap has begun to stabilize at its peak displacement. An unwanted shape in the flap elements can be seen in the coarse mesh at this stage. This is known as an hourglass or keystone effect (in reference to their shape) and is caused by the elements being too soft [**24**]. It occurs when modes, not including rigid body modes, are not resisted by elements [**24**], [**25**]. In LS-DYNA hourglass controls can be used to counteract hourglass forces, in addition the way that the solid is constructed with elements can reduce or eliminate hourglass modes [**26**]. In the case of this study the hourglass modes are eliminated by the refinement of the flap mesh. The opposite is also possible, that is as you apply too many forces to counteract the hourglass mode or you construct a mesh in certain ways (triangular elements especially) that it will become too stiff.

At time 8500ms all of the flaps with the exception of the coarse mesh have reached steady-state. An additional test was completed to see if the coarse mesh does eventually converge. The termination time was set to 30000ms and it was found that the coarse mesh converges around the 20000ms mark. The base mesh in this case also contains a high velocity leak to the right of the flap which disappears with further refinement.

Next, the steady-state resultant displacement is tested for convergence. Figure 4.17 shows a graph of the resultant displacement over time for each mesh size. As time progresses it is evident that every mesh converges to a steady-state value, with the exception of the coarse mesh (explained earlier in this section). As the mesh becomes more refined the displacement over time begins follows a more standardized path. That is, the coarse mesh

is the least like the very refined mesh, whereas, the refined mesh is close to the very refined mesh.

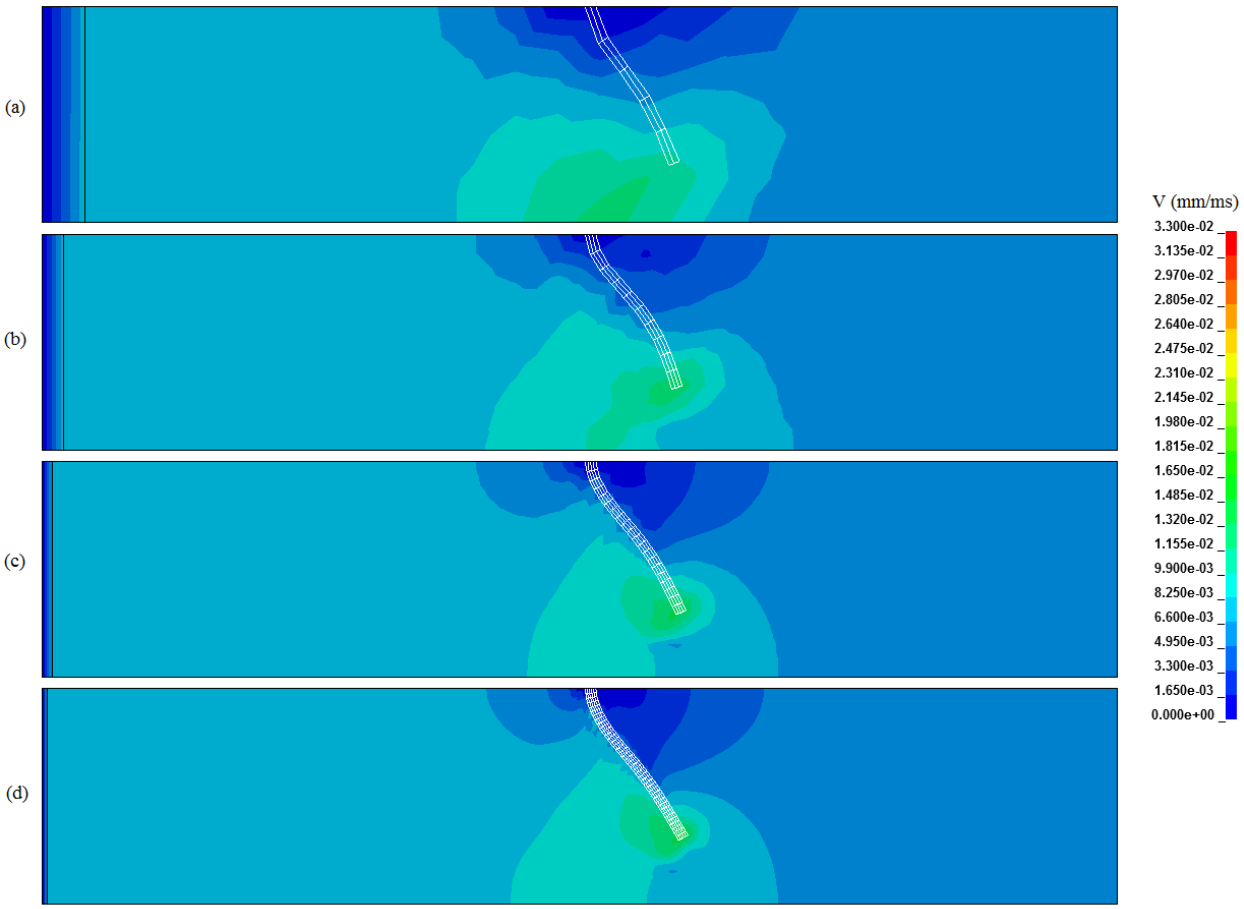

Figure 4.14 - Soft Flap, Constant Flow: 1500ms, Simulations (a)5, (b)6, (c)7, (d)8

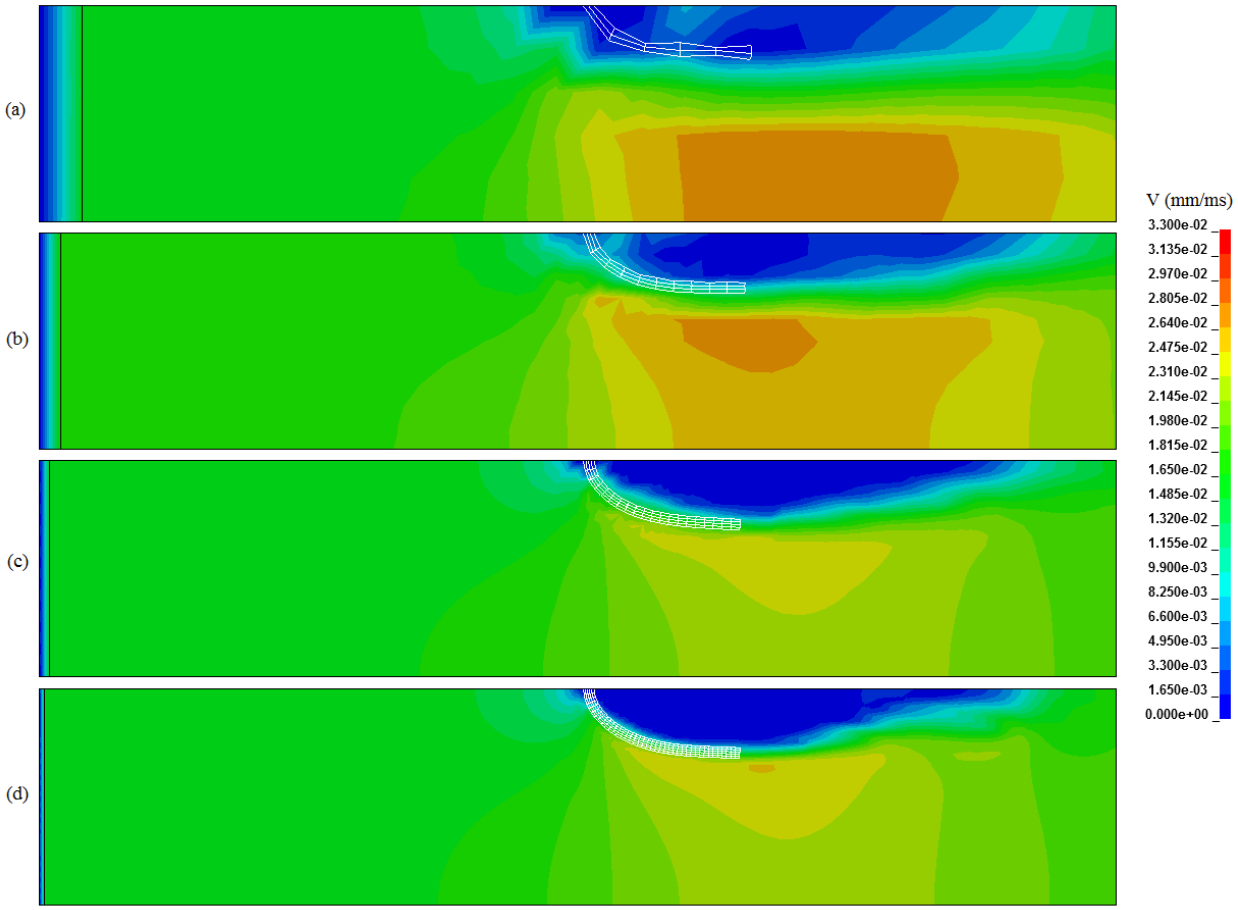

Figure 4.15 - Soft Flap, Constant Flow: 5000ms, Simulations (a)5, (b)6, (c)7, (d)8

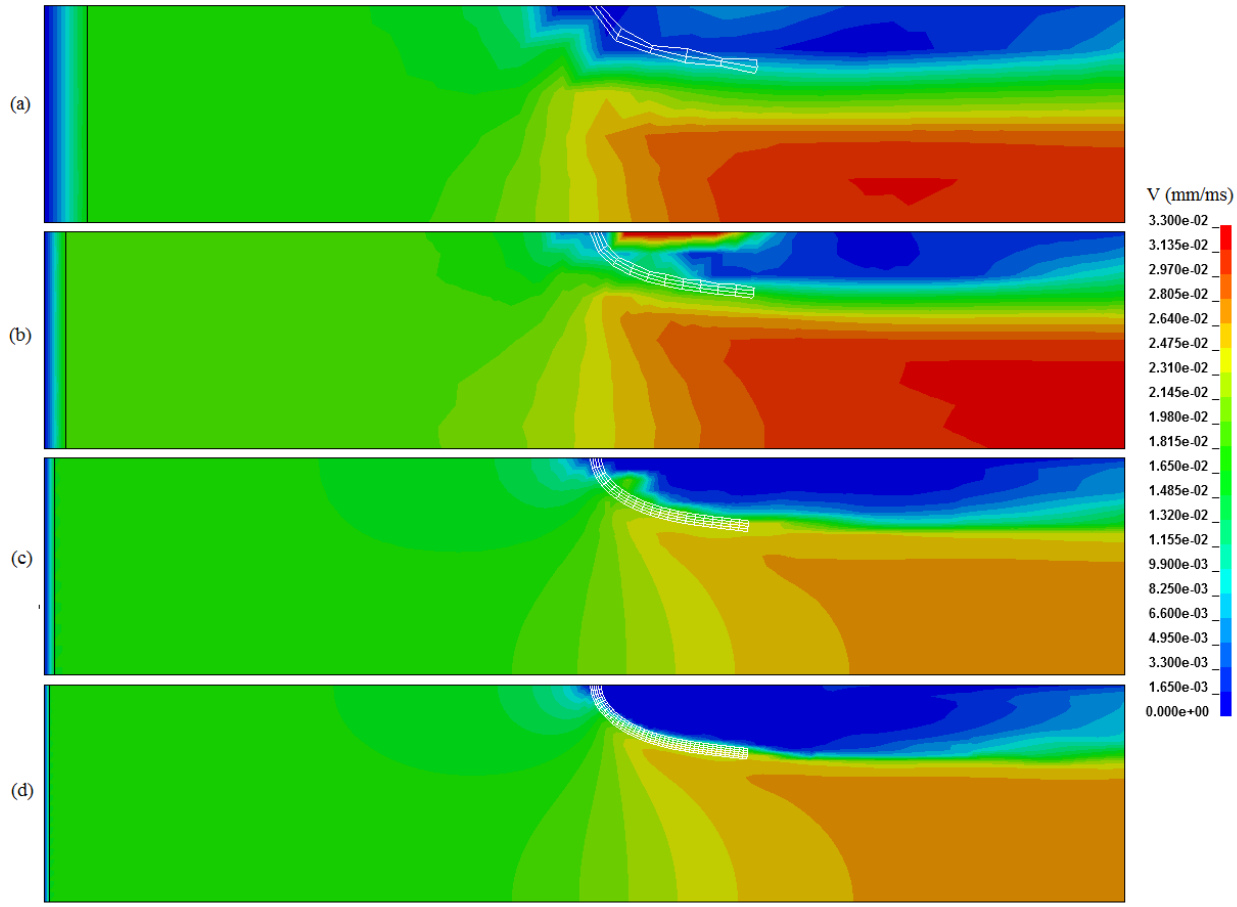

Figure 4.16 - Soft Flap, Constant Flow: 8500ms, Simulations (a)5, (b)6, (c)7, (d)8

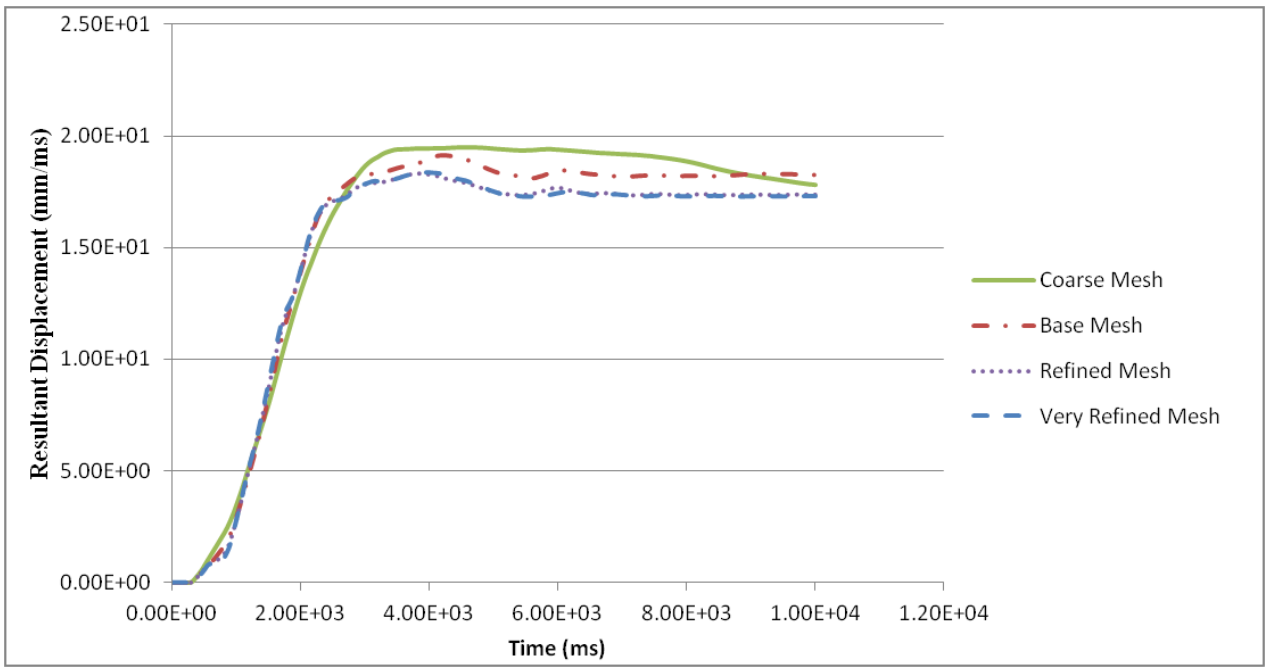

Figure 4.17 - Simulation 5-8: Resultant Displacement vs. Time

The error for the steady-state resultant displacement is calculated for this case using the same method shown by equation 4.1. For the coarse mesh, because it does not reach steady-state in the allotted time, the value at the time where the other meshes reach steady-state is used. This occurs approximately at 6000ms. Table 4.2 shows the error results for this case.

Table 4.2 - Relative Error for the Soft Material Card

Mesh	Steady-state Resultant Displacement	Error relative to the Very Refined mesh (%)
Coarse	19.37939	12.06
Base	18.26056	5.59
Refined	17.37947	0.50
Very Refined	17.29350	-

The errors computed for the soft material under constant pressure show excellent convergence as the mesh size becomes more refined. The error are in general larger between each mesh size and this is a result from a more dynamically active system.

4.3 Pressure Drop Test

To further validate the channel model the pressure drop test outlined by [**18**] has been completed. In this test the flap is extended to the bottom of the channel model. Boundary conditions are setup so that the flap is restricted in all degrees of freedom on both the bottom and top of the flap. The remainder of the model remains the same. The purpose of this test is to determine the ability for a model to handle sharp pressure gradients.

In [**18**], fluid is described by the unsteady Navier-Stokes equation, and the flap described by an incompressible Neo-Hookean model. Pressure drop is tested in their model by extending the flap to the bottom of their channel as described above. In their case they use an axisymmetric meshing scheme. That is, only a quarter of a the model is computed (3D) and the remainder is constructed using symmetry boundary conditions. The purpose of their testing, is to evaluate the ability for their model to handle sharp pressure gradients inside the fluid mesh. The results of their test is shown in Figure 4.18. In this figure the left side of the chamber stabilizes at their induced pressure while the right side of the chamber remains zero. They conclude by saying their model is able to handle large pressure gradients, thus passing the pressure drop test.

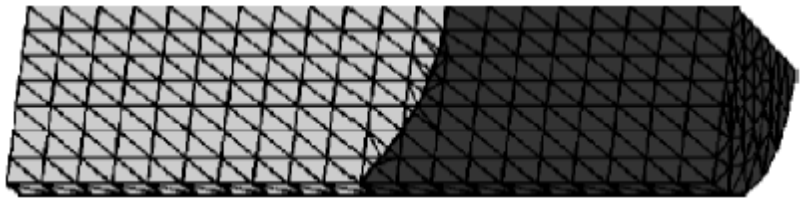

Figure 4.18 - Pressure Gradient 3D-Axisymmetric [18**]**

To perform the pressure drop test in this study, the soft material and constant pressure cards were chosen. The purpose of this test was to determine if the channel model was able to handle large pressure gradients, similar to [**18**]. This is model schematic used is shown in Figure 4.19.

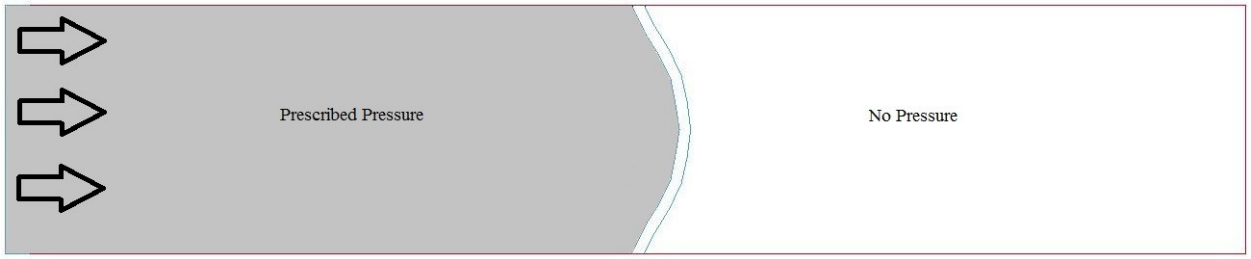

Figure 4.19 - Pressure Drop Test Schematic

The pressure on the left hand side of Figure 4.19 is expected to fluctuate over time due to the inertial forces of the wall; Additionally, after initial loading of the model, the pressure on the right hand side should read zero [**18**]. Pressure readings for elements on the left and right hand side are given in Figure 4.20.

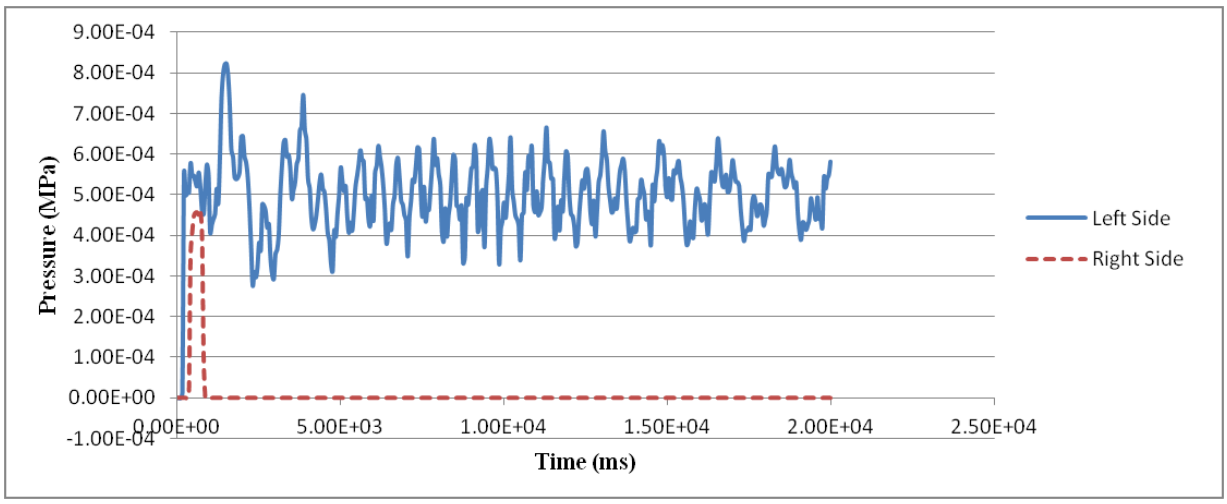

Figure 4.20 - Simulation 9: Pressure Drop Test Results

The above figure shows that the pressures are behaving correctly in the model. The initial pressure wave results in a brief surge of pressure on the right hand side but quickly stabilizes at 0. The left hand side oscillates around the imposed value of 5.00E-4 MPa. This oscillation is a direct result from the inertial forces from the wall. Based on the results shown, the channel model is able to handle large pressure gradients, thus passing the pressure drop test.

4.3 Large Deformation

The channel model was simulated with the mass density of the flap changed to 101g/mm^3 for a constant (Simulation 10) and pulsatile (Simulation 11) flow scheme. The high mass

density was chosen to simulate large deformations because the added momentum would result in a very dynamic flap. The simulations will begin more slowly than previously because of the increased mass density (flap is more difficult to move), however, this will also lead to a larger build up of flow on the left hand side of the flap.

4.4.1 Constant Flow

Figure 4.21, 4.22, 4.24 and 4.23 show the progression of the large deformation constant flow scheme. Figure 4.21 begins similarly to the other simulations. This includes the concave shape of the flap and the low velocity levels at the top of the flap. The major difference is the time in which it takes for this to occur. It takes approximately twice the amount of time when compared to the soft flap simulations. This is evident when you notice the timescales are doubled for this simulation 10 figures. In the last frames of Figure 4.21 the flap begins to straighten out to the follow the path flow. This occurs at a higher resultant displacement than the soft flap, simulation 7.

In Figure 4.22 the flap successfully straightens out and in doing so makes contact with the upper channel wall. Upon impacting the channel wall the flap bounces in the direction of its starting position. In the later frames of Figure 4.22 the flap holds a high level of curvature as a result of the bounce. Velocity contours are less regular than seen in other simulations after the flap has made contact with the wall. However, after bouncing towards the starting position they stabilize quickly.

In Figure 4.23 the flap begins to straight for a second time as a result of the bounce and momentum of the flap tip. The flap then continues beyond the straightening point into the concave position. The position of the flap at this time causes a large increase in velocity of the flow stream. This is caused by the reduction in the flow path width at that time. The increased velocity then leads to a second repetition of flap contact with the upper channel wall. Again, the flap begins to fall after the bounce in the last frame of Figure 4.23.

Figure 4.24 is very similar to Figure 4.23. The flap falls back into a concave shape, the velocity increases and in the last frame the flap is nearing a third bounce. This whip-like motion continues as the simulation progresses.

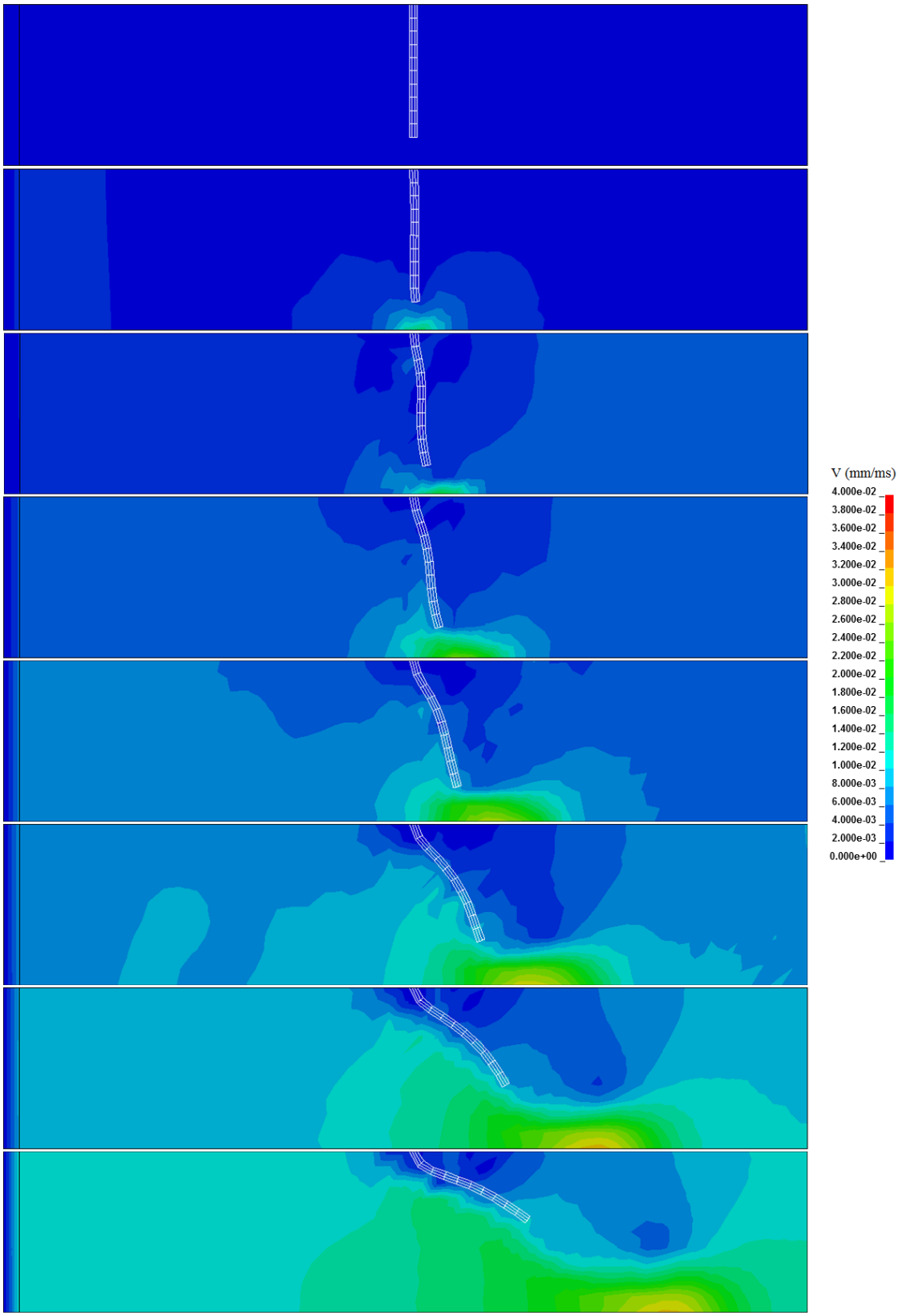

Figure 4.21 - Soft Flap (Mod), Cons. Flow: 0 to 3500ms, 500ms Between Frames

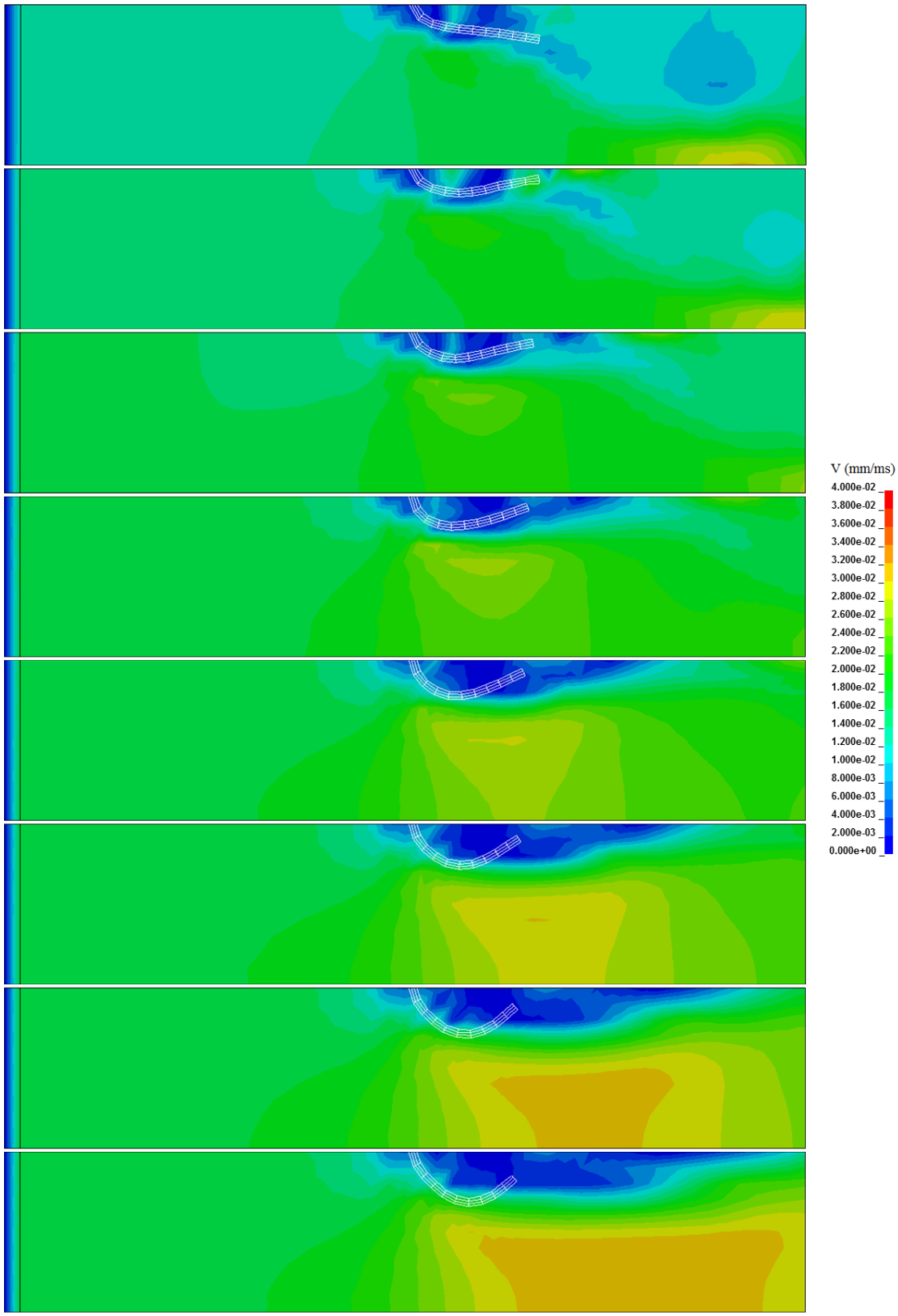

Figure 4.22 - Soft Flap (Mod), Constant Flow: 4000 to 7500ms, 500ms Spacing

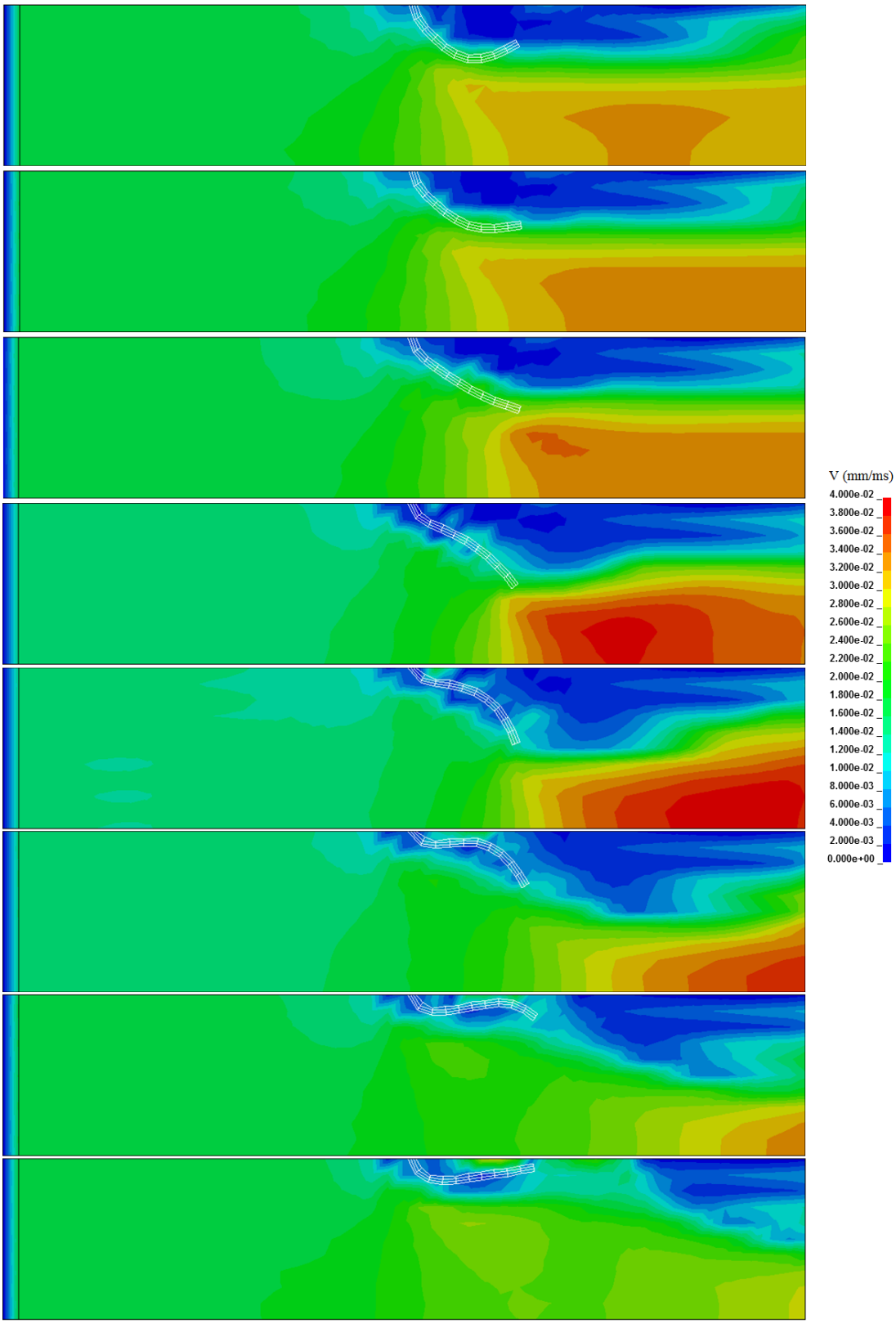

Figure 4.23 - Soft Flap (Mod), Constant Flow: 8000 to 11500ms, 500ms Spacing

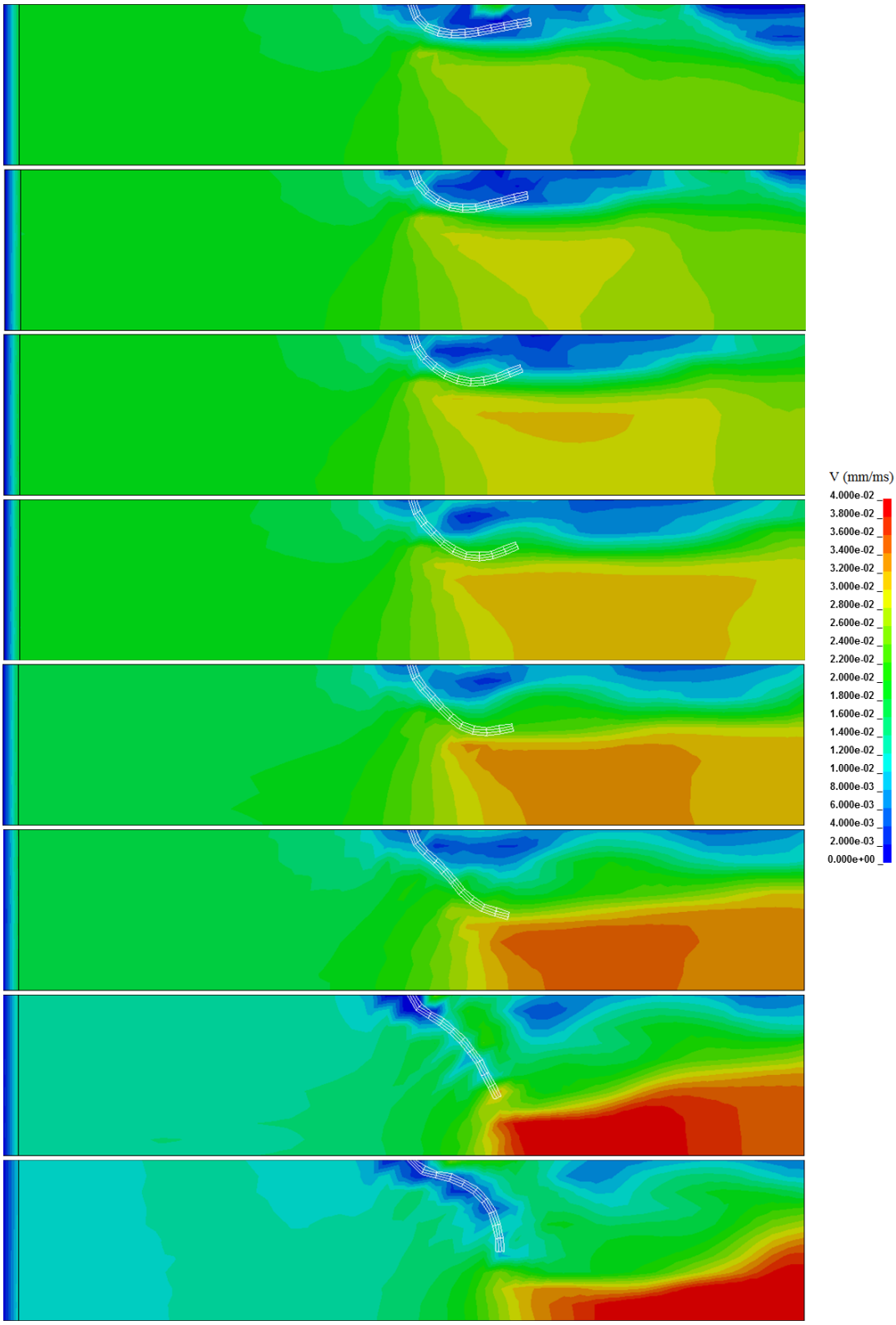

Figure 4.24 - Soft Flap (Mod), Constant Flow: 12000 to 15500ms, 500ms Spacing

4.4.2 Pulsatile Flow

Half period sinusoidal flows were introduced into the channel model. The flow can be broken into the three segments shown in Figure 4.25. In this figure the first segment is the initial sine curve, the second is the zeroed line and the third is the last sine curve. Figure 4.26, 4.27, and 4.28 show the velocity profiles of the simulation at segment one, two and three respectively. The flow in this cases was doubled to 1.00E-3 MPa at the peak of the sine curve. Values of 5.00E-4 MPa at the sine peak resulted in deformations that were not large enough for the purpose of this section.

In Figure 4.26 (segment one) the flap begins to move identically to simulation 10. The major difference between these two simulations is the timescale. Again, the timescale is more than doubled. During segment one the flap makes contact with the upper channel wall at 6250ms, or just after the peak of the sine curve. After contact with the wall the flap bounces back into a concave position by the last frame of Figure 4.26.

In Figure 4.27 the flow begins to dissipate because there is no longer any pressure source providing flow. The remaining flow in the channel is enough for the flap to continue upwards and bounce once for a second time. After bouncing for the second time the flap again goes into a concave shape, however, instead of elevating back to the top of the channel as seen previously, it begins to fall into its initial position. This is a result of the reduced flow in the channel which is near-zero throughout in the later frames of Figure 4.27.

In Figure 4.28 the flow increases once again. The flap begins to move towards the upper channel wall with its shape following the direction of the flow. Just before contact with the wall in the fourth frame, the flap transitions into a concave shape. Shortly after switching shapes, the flap makes contact with the upper channel wall for the third time and bounces towards the bottom. As the flap falls it takes the concave shape and begins ascension to its fourth possible bounce during the simulation. Flows at this point begin to decrease drastically as the source pressure begins to disappear. However, it is likely that the fourth bounce would occur similarly to the frames between Figure 4.26 and 4.27.

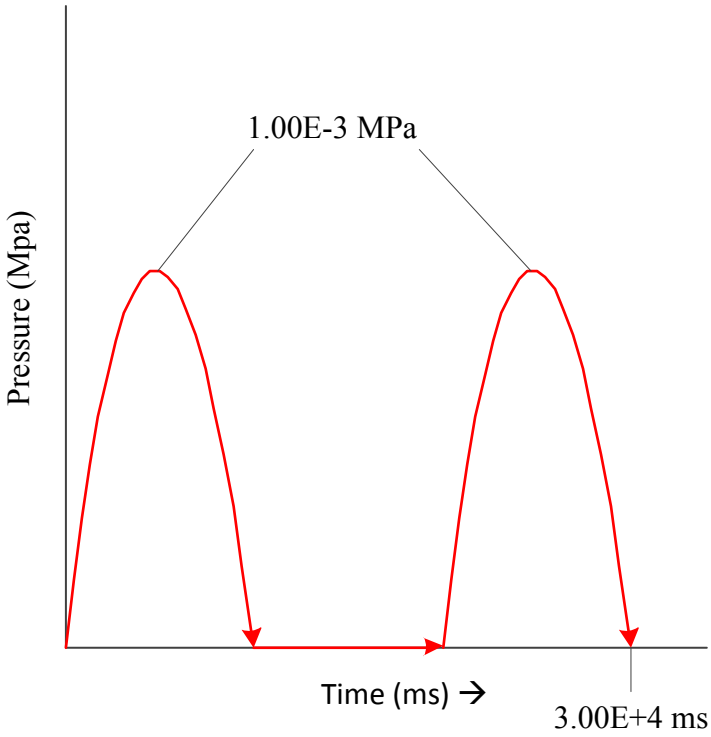

Figure 4.25 - Pulsatile Flow Graph

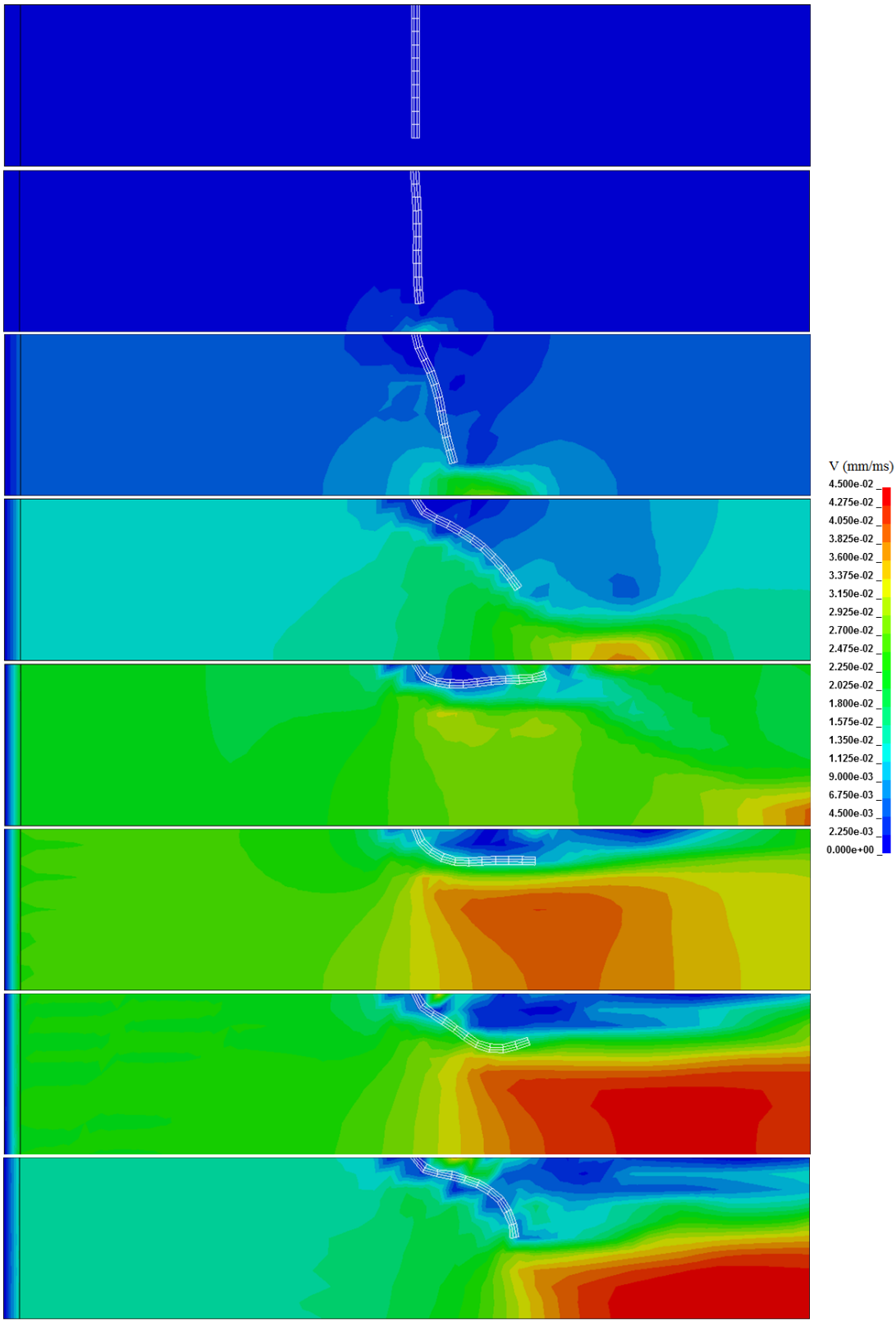

Figure 4.26 - Soft Flap (Mod), Pulsatile Flow: 0 to 8750ms, 1250ms Spacing

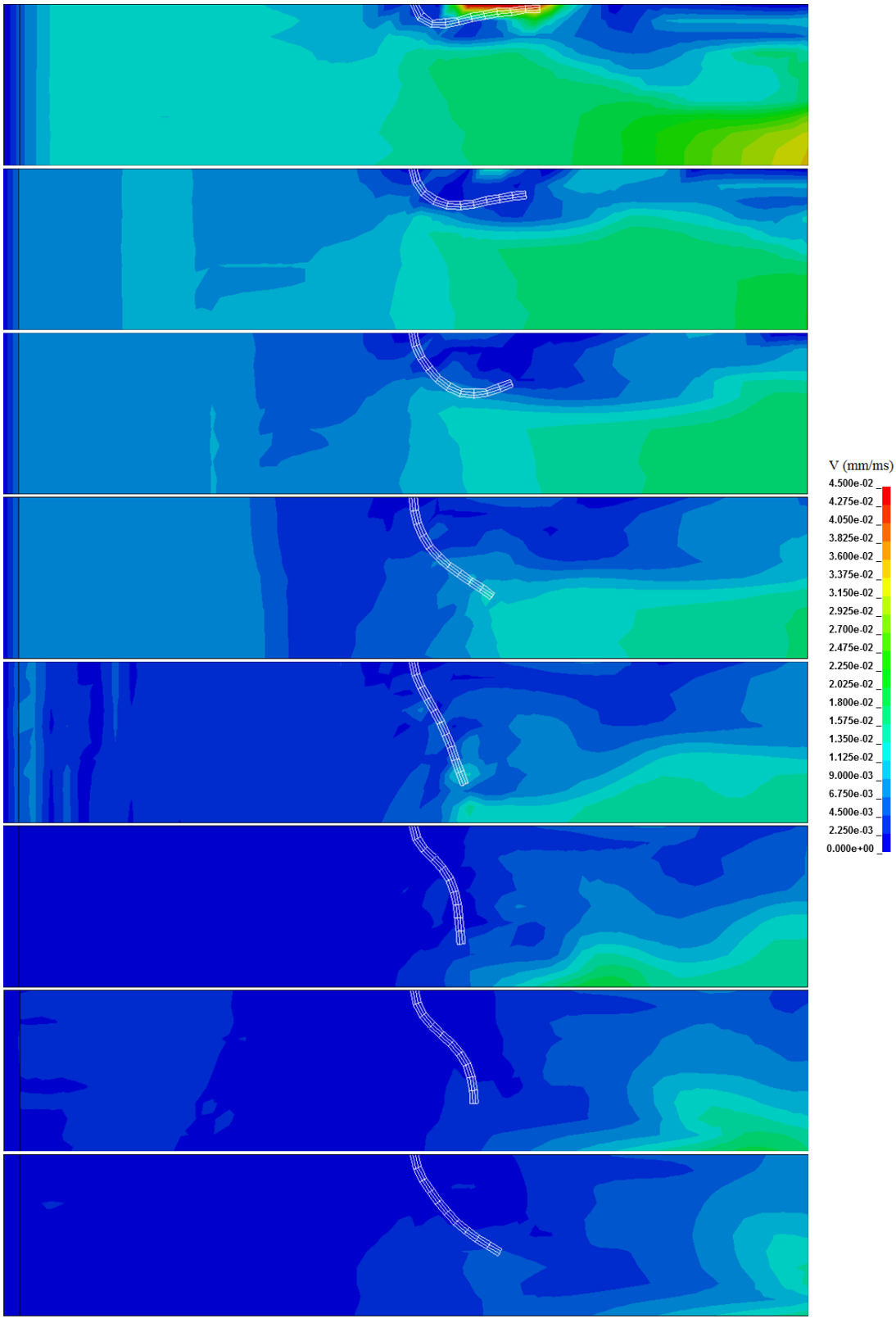

Figure 4.27 - Soft Flap (Mod), Pulsatile Flow: 10000 to 18750ms, 1250ms Spacing

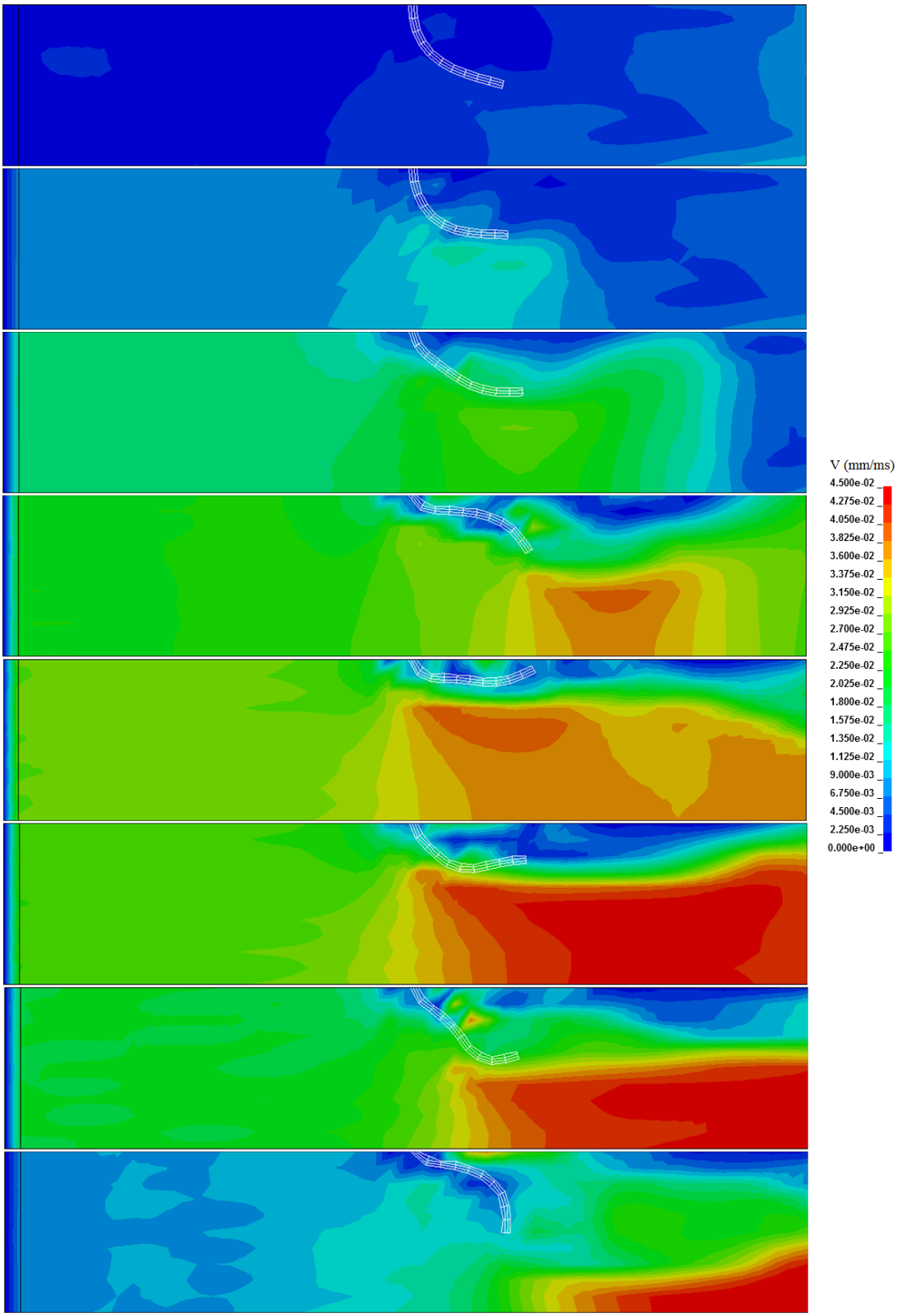

Figure 4.28 - Soft Flap (Mod), Pulsatile Flow: 20000 to 28750ms, 1250ms Spacing

Chapter 5 - Conclusion

In this thesis, a large displacement fluid-structure interaction model using LS-DYNA is developed. The model was created to be used as a preliminary study to show the feasibility of using LS-DYNA for realistic heart valve dynamics. This chapter first summarizes the key points and results found and then discusses future work.

5.1 Conclusions

Chapter 1 of the thesis provided a brief background of numerical heart valve modeling methods. Non-fixed mesh and fixed mesh methods were discussed and it was determined that a fixed mesh is more desirable for simulations with large deformations such as with a heart valve.

Chapter 2 presented the theory used by LS-DYNA relevant to the present research objective. First the core methods used in LS-DYNA were shown such as the conservation and explicit time integration equations. Next, the cards specific to the channel model were discussed in addition to the parameters of importance within them. Equation of state, materials used, fluid-structure coupling parameters, multi-material settings, mesh rezoning, advection, and boundary all played key roles in developing a functioning model.

In Chapter 3, the channel model developed was outlined in detail. First, a variety of the channel model parameters were used to present a wide variety of simulations. Simulations were tailored to test for physics, convergence, pressure drop, and then to explore the FSI capabilities of LS-DYNA in both a constant and pulsatile flow environment. The chapter concluded with details concerning each simulation talked about in Chapter 4.

Finally, Chapter 4 presented the results of the channel model simulations. The results can be summarized as follows:

1. A physical analysis for the stiff and soft flap was completed and compared to each other. Additionally, the soft flap was compared to existing work by [**23**]. The stiff and soft flap had similar characteristics in the way they moved in their initial stages, however, soon after, the stiff flap did not hold a concave shape while soft

flap did. The stiff flap did not have a resultant displacement as large as the soft flap. This also resulted in larger velocities for stiff flap example because of the smaller flow path available. When the soft flap was compared to [**23**] it showed a good qualitative agreement. The flap shape at various stages throughout the simulation were very close to each other. Additionally, the velocity magnitudes were low near the top of the flap and high underneath the flap in both cases. The only major difference between the two was the resultant displacement at the time of the flap straightening (from the concave shape). This was expected to be a result of differing flap material parameters or flap width.

2. The channel model was tested for convergence using both the stiff and soft material card. After comparing the velocity profiles, the resultant displacement over time, and the steady-state resultant displacement it was concluded that the channel model created in LS-DYNA showed strong signs of convergence. In both cases the relative error dropped from significantly as the mesh size became more refined. The stiff flap had lower error values for the coarse mesh (5% vs. 15%) and base mesh (2% vs. 6%). Both the stiff and soft flap showed very similar error values for the refined mesh (0.5%). Additionally, the velocity contours converged. Signs of leakage were seen in the coarse and base mesh of both the stiff and soft flaps and hourglass modes were seen in the coarse mesh of the soft flap.

3. The channel model was modified to perform a pressure drop test as outlined by [**18**]. In this test it is expected that the left chamber (with the pressurized source) has a pressure reading in line with what was imposed. The right chamber (blocked off from the pressure source) should have a pressure of zero. After modifications were made it was found that the pressure on the left side was the input pressure while the right side remained zero. The channel model was able to handle the large pressure drop and thus passed the test.

4. Finally, the channel model was tested for its FSI capabilities in LS-DYNA. This was accomplished by modifying the material density of the flap for the soft

material card. Both constant and pulsatile flow schemes were used. Throughout the constant simulation a repetitive pattern was found. The flap concaved, straightened, extended in direction of flow, made contact with upper channel wall and then descended before repeating. The pulsatile simulation initially followed a similar pattern, however, when reaching the zeroed pressure segment, the flap began to fall into its original position.

5.2 Future Work

There are a number of possibilities for future work in regards to the research presented in this thesis:

1. Additional comparative work can be done to further validate the channel model. Focus on mimicking lab experiments is of particular interest here.

2. In this research only Mooney-Rivlin rubber was used for the flap material because of its ability to deform greatly. Further study should be carried out to consider the use of new and existing materials used for heart valves such as smart metal alloy.

3. The main future goal is to modify the channel model into a full heart valve simulation. One of the greatest challenges for the full heart valve simulation will be the creation of a anatomically correct mesh.

Bibliography

[1] M H Yacoub and J J M Takkenberg, "Will heart valve tissue engineering change the world?," *Nature Clinical Practice Cardiovascular Medicine*, pp. 60-61, 2005.

[2] M M Black and P J Drury, "Mechanical and other problems in artificial," *Current Topics in Pathology*, vol. 84, pp. 127-159, 1994.

[3] L P Dasi, H A Simon, P Sucosky, and A P Yoganathan, "Fluid mechanics of artificial heart valves," *Clinical and Experiemental Pharmacology and Physiology*, vol. 36, no. 2, pp. 225-237, 2009.

[4] D Nordsletten et al., "Fluid-solid coupling for the investigation of diastolic and systolic human left ventricular function," *International Journal for Numerical Methods in Biomedical Engineering*, vol. 27, pp. 1017-1039, 2011.

[5] I Sazonov et al., "Modelling pipeline for subject-specific arterial blood flow - A review," *International Journal for Numerical Methods in Biomedical Engineering*, vol. 27, pp. 1868-1910, 2011.

[6] S G D Kelly, "Computational fluid dynamics insights in the design of mechanical heart valves," *Artificial Organs*, vol. 26, no. 7, pp. 608-613, 2002.

[7] P Yu, K S Yeo, D S Sundar, and S J Ang, "A three-dimensional hybrid meshfree-Cartesian scheme for fluid-body interaction," *International Journal for Numerical Methods of Engineering*, vol. 88, pp. 385-408, 2011.

[8] M Souli, LS-DYNA Advanced Course in ALE and Fluid/Structure Coupling Notes, February 2003, 2003.

[9] F Sotiropoulos and I Borazjani, "A review of state-of-the-art numerical methods for simulating flow," *Medical and Biological Engineering and Computing*, vol. 47, no. 3, pp. 245-256, 2009.

[10] T Gohil et al., "Simulation of oscillatory flow in an aortic bifurcation using FVM and FEM: A comparative study of the implementation strategies," *International Journal for Numerical Methods in Fluids*, vol. 66, pp. 1037-1067, 2011.

[11] I Borazjani, L Ge, and F Sotiropoulous, "Curvilinear immersed boundary method for simulating fluid structure interaction with complex 3d rigid bodies," *J Computat Phys*, vol. 227, no. 16, pp. 7587-7620, 2008.

[12] C S Peskin, "Flow patterns around heart valves: a numerical method," *J Comput Phys*, vol. 10, pp. 252-271, 1972.

[13] C S Peskin and D M McQueen, "A three-dimensional computational method for blood flow in the heart. 1. Immersed elastic fibers in a viscous incompressible fluid," *J Comput Phy*, vol. 81, no. 2, pp. 372-405, 1989.

[14] R Glowinski, T W Pan, T I Hesla, and D D Joseph, "A distributed Lagrange multiplier/fictitious domain method for particulate flows," *Int J Multiphase Flow*, vol. 25, no. 5, pp. 755-794, 1999.

[15] R van Loon, P D Anderson, and F N van de Vosse, "A fluid-structure interaction method with solid-rigid contact for heart valve dynamics," *J Comput Phys*, vol. 217, no. 2, pp. 806-823, 2006.

[16] M A Nicosia, R P Cochran, and K S Kunzelman, "Coupled Fluid-Structure Finite Element Modeling of the Aortic Valve and Root," in *Proceedings of the Second Joint EMBS/BMES Conference*, Houston, TX, USA, 2002, pp. 23-26.

[17] D J Benson, "Computational methods in Lagrangian and Eulerian hydrocodes," *Computer Methods in Applied Mechanics and Engineering*, vol. 99, pp. 235-394, 1992.

[18] R van Loon, P Anderson, F Baaijens, and F van de Vosse, "A three-dimensional fluid–structure interaction method for heart valve modelling," *Fluid-solid interactions: modeling, simulation, bio-mechanical applications*, vol. 333, no. 12, pp. 856-866, 2005.

[19] LSTC, *LS-DYNA Keyword User's Manual*. Livermore, California, 2007.

[20] LSTC, *LS-DYNA Theory Manual*. Livermore, California, 2006.

[21] N Aquelet, M Souli, and L Olovsson, "Euler-Lagrange coupling with damping effects: Application to slamming problems," *Comput. Methods Appl. Mech. Engrg.*, vol. 195, no. 1-3, pp. 110-132, 2005.

[22] L Olovsson, M Souli, and I Do, LS-DYNA - ALE Capabilities (Arbitrary-Lagrangian-Eurlerian) Fluid-Structure-Interaction Modeling, 2003.

[23] F Baaijens, "A fictitious domain/mortar element method for fluid-structure interaction," *Int. J. Numer. Meth Fluids*, vol. 35, pp. 743-761, 2001.

[24] D Benson, "Zero Energy Modes in One Dimension: An Introduction to "Hourglass" Modes," 2003.

[25] D Pantuso and K-J Bathe, "On the stability of mixed finite elements in large strain analysis of incompressible solids," *Finite Elements in Analysis and Design*, vol. 28, pp. 83-104, 1997.

[26] Rudolf Botticher, "Fluid Structure Interaction with *MAT_SOFT_TISSUE and EFG Elements," , Birmingham, UK, 2005.

[27] J Day, Guidelines for ALE Modeling in LS-DYNA, 2009.

[28] R van Loon, P Anderson, J de Hart, and F Baaijens, "A combined fictitious domain/adaptive meshing method for fluid-structure interaction in heart valves," *International Journal for Numerical Methods in Fluids*, vol. 46, pp. 533-544, 2004.

[29] H Wang and T Belytschko, "Fluid-structure interaction by the discontinuous-Galerkin method for large deformations," *International Journal for Numerical Methods in Engineering*, vol. 77, pp. 30-49, 2009.

[30] P Le Tallec, J-F Gerbeau, P Hauret, and M Vidrascu, "Fluid structure interaction problems in large deformation," *Fluid-solids interactions: modeling, simulation, bio-mechanical applications*, vol. 333, no. 12, pp. 910-922, 2005.

[31] S Etienne and D Pelletier, "A general approach to sensitivity analysis of fluid-structure interactions," *Journal of Fluids and Structures*, vol. 21, pp. 169-186, 2005.

[32] K-J Bathe and H Zhang, "A mesh adaptivity procedure for CFD and fluid-structure interactions," *Computers and Structures*, no. 87, pp. 604-617, 2009.

Appendix A - LS-DYNA Example Code

Example code from LS-DYNA is provided in this section. Much of the code has been excluded for length reasons (hundreds of pages). The excluded code mainly contains the element values, node values and boundary condition values. The example code follows:

```
$# LS-DYNA Keyword file created by LS-PrePost 3.1 - 07Sep2011(09:02)
$# Created on Jul-06-2012 (14:10:41)
*KEYWORD
*TITLE
$# title
LS-DYNA keyword deck by LS-PrePost

*CONTROL_ALE
$#     dct      nadv      meth      afac      bfac      cfac      dfac      efac
       2         1         2        -1.00     0.000     0.000     0.000     0.000
$#   start       end     aafac     vfact      prit       ebc      pref   nsidebc
    0.000    1.E+201.00     1.E-6         0         0     0.000         0
$#    ncpl      nbkt    imascl    checkr
       1        50         0     0.000

*CONTROL_CONTACT
$#  slsfac    rwpnal    islchk    shlthk    penopt    thkchg     orien    enmass
  0.100000     0.000         1         0         0         0         1         0
$#  usrstr    usrfrc     nsbcs    interm     xpene     ssthk      ecdt   tiedprj
       0         0         0         0  4.000000         0         0         0
$#   sfric     dfric      edc       vfc        th     th_sf    pen_sf
    0.000     0.000     0.000     0.000     0.000     0.000     0.000
$#  ignore    frceng   skiprwg    outseg   spotstp   spotdel   spothin
       0         0         0         0         0         0     0.000
$#    isym    nserod    rwgaps    rwgdth     rwksf      icov    swradf    ithoff
       0         0         0     0.000  1.000000         0     0.000         0
$#  shledg
       0

*CONTROL_CPU
$#  cputim
    0.000

*CONTROL_DAMPING
$#  nrcyck     drtol    drfctr    drterm    tssfdr    irelal     edttl    idrflg
     250  0.001000  0.995000     0.000     0.000         0  0.040000         0

*CONTROL_ENERGY
$#    hgen      rwen    slnten     rylen
       1         2         1         1

*CONTROL_OUTPUT
$#   npopt    neecho    nrefup    iaccop     opifs    ipnint    ikedit    iflush
       0         0         0         0     0.000         0       100      5000
$#   iprtf    ierode     tet10    msgmax    ipcurv
       0         0         2        50         0
```

```
*CONTROL_TERMINATION
$#  endtim    endcyc     dtmin    endeng    endmas
 10000.000         0     0.000     0.000     0.000

*CONTROL_TIMESTEP
$#  dtinit    tssfac      isdo    tslimt     dt2ms      lctm     erode     ms1st
    0.000  0.650000         0     0.000     0.000         0         0         0
$#  dt2msf   dt2mslc     imscl
    0.000         0         0

*DATABASE_ELOUT
$#      dt    binary      lcur     ioopt
 10.000000         0         0         1

*DATABASE_GLSTAT
$#      dt    binary      lcur     ioopt
 10.000000         0         0         1

*DATABASE_MATSUM
$#      dt    binary      lcur     ioopt
 10.000000         0         0         1

*DATABASE_BINARY_D3PLOT
$#      dt      lcdt      beam     npltc    psetid
 50.000000         0         0         0         0
$#   ioopt
       0

*DATABASE_BINARY_D3THDT
$#      dt      lcdt      beam     npltc    psetid
 50.000000         0         0         0         0

*DATABASE_EXTENT_BINARY
$#   neiph     neips    maxint    strflg    sigflg    epsflg    rltflg    engflg
       0         0         3         0         1         1         1         1
$#  cmpflg    ieverp    beamip     dcomp      shge     stssz    n3thdt   ialemat
       0         0         0         1         1         1         2         1
$# nintsld   pkp_sen      sclp    unused     msscl     therm    intout    nodout
       0         0  1.000000         0         0        0STRESS    STRESS

*BOUNDARY_SPC_NODE
$#     nid       cid      dofx      dofy      dofz     dofrx     dofry     dofrz
*PART
$# title
Water
$#     pid     secid       mid     eosid      hgid      grav    adpopt      tmid
       1         3         1         2         0         0         0         0

*SECTION_SOLID_ALE_TITLE
Water_section
$#   secid    elform       aet
       3        11         1
$#    afac      bfac      cfac      dfac     start       end     aafac
   0.000     0.000     0.000     0.000     0.000     0.000     0.000

*MAT_NULL_TITLE
```

```
Water
$#     mid        ro        pc        mu     terod     cerod        ym        pr
         1  0.998200     0.000 9.9820E-9     0.000     0.000     0.000     0.000

*EOS_GRUNEISEN_TITLE
Water
$#   eosid         c        s1        s2        s3     gamao         a        e0
         2  0.165000  1.920000     0.000     0.000  0.100000     0.000     0.000
$#      v0
     0.000
*PART
$# title
Flap
$#     pid     secid       mid     eosid      hgid      grav    adpopt      tmid
         2         1         2         0         0         0         0         0

*SECTION_SOLID_TITLE
Flap_section
$#   secid    elform       aet
         1         0         0

*MAT_MOONEY-RIVLIN_RUBBER_TITLE
Flap
$#     mid        ro        pr         a         b       ref
         2  1.010000  0.499000  0.013292  0.002630     0.000
$#     sgl        sw        st      lcid
     0.000     0.000     0.000         0
*PART
$# title
Source
$#     pid     secid       mid     eosid      hgid      grav    adpopt      tmid
         3         2         1         1         0         0         0         0

*SECTION_SOLID_ALE_TITLE
Source_section
$#   secid    elform       aet
         2        11         4
$#    afac      bfac      cfac      dfac     start       end     aafac
     0.000     0.000     0.000     0.000     0.000     0.000     0.000

*EOS_GRUNEISEN_TITLE
Water (source)
$#   eosid         c        s1        s2        s3     gamao         a        e0
         1  0.165000  1.920000     0.000     0.000  0.100000     0.000  0.005000
$#      v0
     0.000

*SET_PART_LIST_TITLE
water_partlist
$#     sid       da1       da2       da3       da4    solver
         1     0.000     0.000     0.000     0.000MECH
$#    pid1      pid2      pid3      pid4      pid5      pid6      pid7      pid8
         1         3         0         0         0         0         0         0

*ALE_MULTI-MATERIAL_GROUP
$#     sid    idtype    gpname
```

```
       3       1

*ALE_MULTI-MATERIAL_GROUP
$#     sid    idtype    gpname
       1       1

*CONSTRAINED_LAGRANGE_IN_SOLID_TITLE
$#  coupid                                                title
       0coupling
$#   slave    master     sstyp     mstyp     nquad     ctype     direc     mcoup
       2       1         1         0         2         4         2         0
$#   start       end      pfac      fric    frcmin      norm   normtyp      damp
   0.0001.0000E+10  0.100000     0.000  0.500000         0         0     0.000
$#      cq      hmin      hmax     ileak     pleak   lcidpor     nvent  blockage
   0.000     0.000     0.000         0  0.010000         0         0         0
$#  iboxid   ipenchk   intforc   ialesof    lagmul    pfacmm      thkf
       0         0         0         0     0.000         0     0.000

*ELEMENT_SOLID
$#   eid     pid      n1       n2       n3       n4       n5       n6       n7       n8

*NODE
$#   nid               x               y               z      tc      rc

*END
```