“Learn from yesterday, live for today, hope for tomorrow. The important thing is not to stop questioning.”

Albert Einstein

University of Alberta

DISCONTINUUM MODELLING OF VUGGY CARBONATES

by

ALMA INÉS ORNÉS

A thesis submitted to the Faculty of Graduate Studies and Research in partial fulfillment

of the requirements for the degree of

MASTER OF SCIENCE

in

GEOTECHNICAL ENGINEERING

DEPARTMENT OF CIVIL AND ENVIRONMENTAL ENGINEERING

Spring 2014

Edmonton, Alberta

To my parents, Alvaro and María, for all their love and support.

Eskerrik asko aitas!

Abstract

The Grosmont Formation in Alberta is a heterogeneous carbonate reservoir of current interest for bitumen extraction, specifically within the vuggy porosity unit. The current research addresses the effects of vugs on the strength and stiffness of carbonate samples under uniaxial compression. Particle Flow Code 3D (PFC3D), a discontinuum modelling technique, is used to evaluate vuggy carbonate samples with different vug volumes, shapes, and locations. This thesis also presents a carbonate testing workflow, which combines computed tomography, laboratory testing, and PFC3D modelling. The workflow is evaluated with laboratory experiments on six vuggy carbonate samples from the Grosmont Formation. The workflow successfully generates samples with the correct vuggy geometry; yet the PFC3D simulations over-predict the laboratory results by 24 to 163% for the uniaxial compression strength. Future research on the calibration of PFC3D material for various scales is recommended to further the understanding of up-scaling carbonate properties for heterogeneous reservoirs.

Acknowledgement

I would like to acknowledge Nathan Deisman for his help, support, and many inspiring discussions throughout the course of this research. I would also like to thank Diego Mas Ivars for his insightful comments on key issues. Special acknowledgement to Laricina Energy, National Science and Engineering Research Council (NSERC), and the Helmholtz Alberta Initiative (HAI); without whose donation and financial support this research would not have been possible. I am grateful to all the technicians and researchers at the Geomechanical Reservoir Experimental Facility (GeoREF) that assisted me during the laboratory component of my work. Finally, I would like to thank my supervisor, Dr. Chalaturnyk, for his guidance and encouragement.

Table of Contents

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1: INTRODUCTION

Carbonate reservoirs are known globally as an important source for hydrocarbon resources, and more recently as potential carbon dioxide (CO_2) storage sites. In Alberta, over 500 billion barrels of in-place bitumen resources are held in carbonate deposits (ERCB 2012). These deposits are not currently considered part of Alberta's reserves due to the absence of commercial projects producing them. The two main carbonate deposits in the province are the Athabasca Grosmont and Nisku formations, with estimated resources of approximately 400 and 100 billion barrels, respectively (ERCB 2012). The Grosmont formation was the focus of several pilot projects in the 1970s and 1980s, which obtained mixed results and no eventual commercial operation established (Edmunds et al. 2009). However, the interest in this resource has increased recently due to dwindling global hydrocarbon reserves, advancements in new technologies, and high oil prices.

The focus on the Grosmont formation has resulted in abundant literature on its geology, bitumen characteristics, and flow properties; however, geomechanical properties for this formation are not readily available in the literature. Aside from the recent work by Arseniuk et al. (2012) regarding stability for open-hole completion; there is little information on the geomechanical behaviour of these rocks and their endurance to elevated temperatures and pressures.

Currently, the most promising techniques for developing the Grosmont reservoir are thermal methods such as Steam Assisted Gravity Drainage (SAGD). This technology involves injecting steam into the reservoir to heat up and mobilize the bitumen, and it is currently used in commercial operations in the McMurray Formation (commonly known as the Alberta Oil Sands). The carbonate rocks in the Grosmont Formation will be subjected to an increase in pore pressures and temperatures caused by the steam injection. Part of a successful SAGD operation is modelling of the evolution of the steam chamber in the reservoir as well as monitoring potential changes in the formation and overlaying strata (e.g. integrity of the seal or cap rock). This requires a good characterization of the reservoir properties for a successful operation and accurate prediction.

Initial studies have focused on flow modelling for the Grosmont reservoir; but geomechanical properties have not been studied as extensively. Geomechanics was first used in the petroleum field to overcome well stability and reservoir hydraulic communication problems; but it has evolved to become a part of reservoir planning and development (Santos and Ferreira 2010). The geomechanical properties usually investigated are elastic Young's modulus (E), uniaxial compression strength (UCS), Poisson's ratio (υ), friction angle (peak and residual) and cohesion (peak and residual). Elastic moduli, both dynamic (E_d) and static (E_s), are necessary for correct interpretation of sonic surveys and strain/displacement prediction. UCS data is necessary for borehole integrity, as shown in the sensitivity analysis conducted by Arseniuk et al. (2012). The connection between geomechanical behaviour and flow processes cannot be overlooked; stress changes due to increase in temperature and pressure experienced during thermal recovery can impact both steam and bitumen flow (Freeman et al. 2009).

1.1. Objective of this Study

The current study seeks to further the knowledge on the geomechanical behaviour of vuggy specimens of the Grosmont Formation. Vugs are voids within the carbonate samples generally assumed to be caused by dissolution (Lonoy 2006); these are encountered within sections of the Grosmont and are a product of its complex diagenetic history. The objective of this research is to evaluate the effect of vugs on the geomechanical properties of rock samples. It is proposed that a discontinuum model can be used to simulate vuggy carbonate rocks at the laboratory scale, based on a calibration to small intact specimens and a geometric representation of vugs in larger samples.

According to Albahlani and Babadagli (2008), the role of geomechanics in the SAGD process had not been studied for carbonate reservoirs; and the general applicability of SAGD in carbonates has not been explored as extensively as for clastic reservoirs like the oil sands. Geomechanical factors that are important for reservoir simulations such as heterogeneity and fractures are present in the Grosmont reservoirs. The first step for reservoir simulations is a proper characterization of the formation and good knowledge of basic geomechanical properties. Understanding the mechanics of the vuggy dolomites and limestones is a step towards future analysis of the reservoir on a larger scale, and would service future modelling and production of the Grosmont reservoir.

The method chosen for this research is discontinuum numerical modelling with calibration to Grosmont UCS samples. The discontinuum numerical technique used is Itasca’s Particle Flow Code 3D (PFC3D). According to Palchik (2011), correct estimates of UCS and E are required to properly tackle rock engineering problems, which can be extended to rock mechanics application in reservoir and petroleum engineering. Thus, these two parameters are the focus of the current study.

The following chapter presents a short review on the Grosmont carbonates followed by a summary of the current knowledge on geomechanical properties for carbonate rocks. The development and state of practice for PFC3D is presented in the literature review in Chapter 2. Chapter 3 explains the methodology used for the numerical simulations of idealized vuggy laboratory scale samples. Chapter 4 presents a workflow developed for representing real carbonate samples in discontinuum simulations; this chapter covers the process of converting computed tomography (CT) images into PFC3D models that are calibrated to laboratory test results. Chapter 5 and 6 discuss the laboratory testing conducted on vuggy Grosmont samples and the comparison with PFC3D models, respectively. Finally, Chapter 7 presents the conclusions of the studies and recommendations for future research on Grosmont carbonates and PFC3D.

1.2. History of Production in the Grosmont Carbonates

The first production trial on the Grosmont formation was a cyclic steam stimulation (CSS) pilot project in 1974 (Jiang et al. 2009). During the following years and up to the late 1980’s, other demonstration programs targeted the Grosmont formation (AOSTRA 1990). Some of these projects were carried out with the involvement of the Alberta Oil Sands Technology and Research Authority (AOSTRA), which was responsible for testing and developing new technologies for Alberta’s resources production (Alvarez et al. 2008). The results of the projects were inconsistent, varying between 3 to 80 m^3 of bitumen per day (refer to Table 1.1). This led to hesitations regarding production from this carbonate reservoir due to discouraging results and the conclusion that the formation was highly heterogeneous (Edmunds et al. 2009; Dembicki and Machel 1996). Eventually, these in-situ pilots were abandoned due to unfavourable economic conditions (Buschkuehle et al. 2007).

Table 1.1: Pilot projects in the Grosmont Formation during 70s and 80s based on Alvarez et al. (2008)

Pilot Project	Year	Peak Production	Method
Chipewyan River Pilot	1975	$3m^3$/day	CSS
Old Buffalo Creek	1977-1978	$80m^3$/day	Steam Drive, CSS and three combustion tests. 14 well series
Buffalo Creek	1980-1986	$70m^3$/day	CSS (twelve cycles)
McLean	1981-1987	$100m^3$/day	CSS and foam pilot test
Saleski	1986	$117m^3$/day	CSS (three cycles)

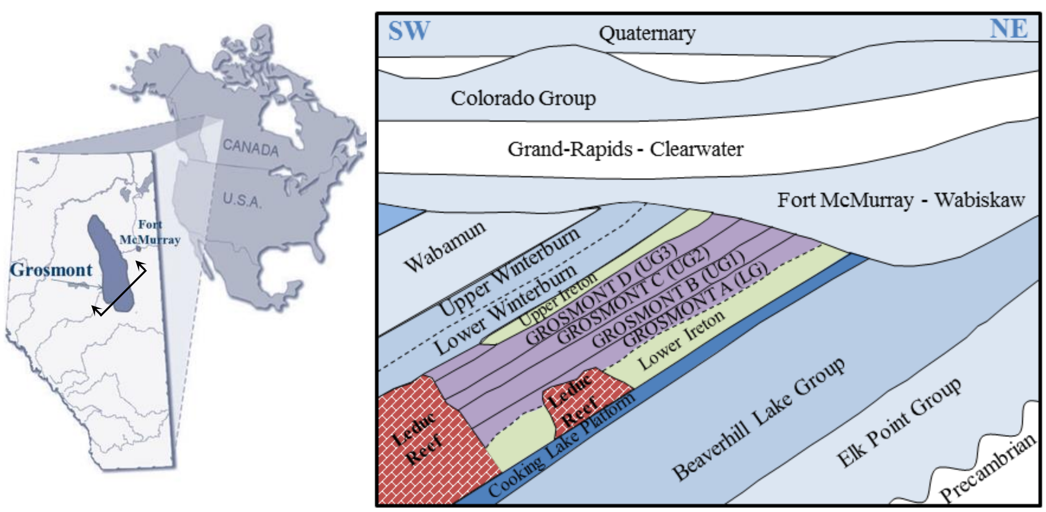
Two decades after the initial pilot projects in the bitumen carbonates, interest in this resource began to be revitalized, fueled by operational success with in-situ oil sands projects and higher oil prices. Both CSS and SAGD techniques had advanced and become proven technologies; moreover solvent and CO_2 injection for enhanced oil recovery had also improved. With the new knowledge and technology, the production of Alberta's bitumen carbonates appeared possible. Sales of Alberta Crown land for the Grosmont area increased to a total of $465 million US by early 2006 (Buschkuehle et al. 2007).

In 2007, Laricina Energy and Osum Oil Sands Corp. began a cold solvent injection test in the Grosmont formation (Jiang et al. 2009). In 2009, the Energy Resources Conservation Board (ERCB) conducted a major review of the Athabasca Grosmont deposit, resulting in a 28% increase of the potential in-place resources (ERCB 2010). Later on, new pilot projects like the Saleski Joint Venture (Osum Oil Sands Corp) and the Harper Steam Injection Scheme (Sunshine Oilsands Ltd.) began operating in 2010 and 2011, respectively. Currently, companies such as Husky Energy, Shell, and Athabasca Oil Corporation have acquired interest in the Grosmont Formation. The substantiation of the Grosmont carbonates as recoverable reserves appears possible in the near future, with the first commercial project of 10,700 bbl/day by Laricina scheduled for first steam in 2014 (Arseniuk et al. 2012). Challenges still remain in the production of Grosmont carbonates for future commercial level operations. Many drilling challenges within this formation were recently reported by Arseniuk et al. (2010).

The Grosmont is described as a dual porosity reservoir with a primary porosity in the rock matrix and secondary porosity due to networks of large vugs, fractures, channels, and caverns produced during its complex diagenesis (Luo and Machel 1995). Moreover, seal effectiveness is compromised in certain areas of the reservoir due to fractures of the intervening shales, which has been suggested as an explanation to some of the production problems in the early pilot projects of the 70s and 80s (Dembicki and Machel 1996). The reservoir is known for its high heterogeneity, multiple porosity types and paleokarst system (Luo and Machel 1995; Dembicki and Machel 1996). This provides an opportunity for large bitumen accumulation in open spaces such as caverns or fractures; but it also presents a challenge for predicting flows, recovery, cap-rock integrity, and long-term production.

1.3. Geology of the Grosmont Carbonates

The Grosmont Formation is described as a large Upper Devonian carbonate platform in Northern Alberta (refer to Figure 1.1a), which extends over 75000km^2 in area (Buschkuehle et al. 2007). The Grosmont is underlain largely by the Lower Ireton Formation, but in some sections by the Leduc reef complexes; and can be overlain conformably by either the Upper Ireton or the Nisku formation (Thériault 1993; Switzer et al. 1994). In the northeast, the Grosmont is unconformably overlain by the McMurray-Wabiskaw formation, mostly near the erosional edge of the platform (Buschkuehle et al. 2007). The formation is sub-divided into four carbonate units: Grosmont A (Lower Grosmont or LG), Grosmont B (Upper Grosmont 1 or UG1), Grosmont C (Upper Grosmont 2 or UG2), and Grosmont D (Upper Grosmont 3 or UG3). These layers are separated from each other by beds of shale or marl; traditionally referred to by the term "shale breaks" (Luo and Machel 1995). The stratigraphy of the Grosmont and near-by formations is presented in Figure 1.1b; for a more detailed review of the geology refer to Buschkuehle et al. (2007), Switzer et al. (1994), and Anfort et al. (2001).



a) b)

Figure 1.1: Location and stratigraphy of the Grosmont succession in northern Alberta, adapted from by Buschkuehle et al. (2007)

Alberta's Devonian carbonates developed as shallow marine and peritidal deposits during the Frasnian stage. The Grosmont carbonates were subjected to pervasive, yet unequally distributed dolomitization; with the upper layers UG2 and UG3 (Grosmont C and D) more heavily dolomitized (Buschkuehle et al. 2007). The platform was uplifted and sub-aerially exposed for millions of years from the Missisippian to the Cretaceous period, resulting in extensive karstification (Dembicki and Machel 1996). Other diagenetic processes that are thought to have occurred to the Grosmont Formation are fracturing, evaporite precipitation, evaporite dissolution, physical compaction, chemical compaction, and oil migration (Luo and Machel 1995). This last process is the source of the bitumen currently present in the formation, which generated from biodegradation of lighter hydrocarbons into 5 to 9 API bitumen. The complex diagenesis produced a highly heterogeneous reservoir, producing intervals with large void volume, ranging from 40 to 100% porosity in paleokarst caverns (Dembicki and Machel 1996).

Recent academic research on the Grosmont Formation has focused on more geological aspects such as the petrophysical properties of the bitumen or influence of diagenetic processes like dissolution of evaporites. Zhao (2009) conducted research on the bitumen rheology in the Grosmont Formation; the level of bitumen biodegradation and its relation to viscosity of the fluid. Borrero (2010) investigated the presence of Hondo evaporites within the Grosmont, some of which were found to have dissolved producing solution-collapse breccias. It is expected that research and knowledge on this formation will augment as interest in the resource increases and pilot projects continue to operate.

Early on, it was proposed that the top layers UG2 and UG3 would be the most promising for bitumen recovery and in-situ techniques. Thériault (1993) concluded that massive vuggy dolomite lithofacy present in UG2 would be suitable for production due to a pore system which included a network of vugs, channels, and fractures. Vuggy porosity produced by karstification is present in the two intervals which are targeted in current in-situ schemes (Arseniuk et al. 2012). Figure 1.2 presents a core example of vuggy dolomites of the UG2 unit from Osum's Laricina JV project. The presence of these vugs are not only a factor for porosity estimation and reservoir flow models, but should also be considered when analyzing the geomechanical properties of the reservoir. Therefore, the present research seeks to obtain more knowledge on the geomechanical behaviour of rock with widespread void inclusions, such as the Grosmont vuggy carbonates.

Figure 1.2: Vuggy dolomite core from Grosmont UG2 layer with varying presence of bitumen (photos of Laricina core)

1.4. Geomechanical Properties of Carbonates

One of the first detailed studies on intact properties of carbonate rocks was conducted by Miller in 1965 and later summarized in a chart by Deere in 1968 (Santos and Ferreira 2010). The chart proposed by Deere (1968) for sedimentary rocks is presented in Figure 1.3 and it is based on only two parameters: UCS and E. The range of properties for limestones and dolostones is shown in blue and encompasses rocks with low to very high strength, with a modulus ratio (E/UCS) varying from average to high (refer to Figure 1.3). The modulus ratio ranged from 250 to 700 with a mean of 420; however, more recent studies on heterogeneous carbonates established a wider range of the modulus ratio, varying from 60.9 to 1011 (Palchik 2011). Nevertheless, the average ratio was found to be a similar value of 380 (Palchik 2011). Predictions of UCS and E on reservoir scale are often done through empirical correlations. Physical properties measured through wireline logs provide information on bulk density and wave propagation, but there are limited samples to conduct thorough geomechanical testing for all the layers encountered. The estimates of strength and stiffness are thus done with empirical correlations developed for carbonate reservoirs; a summary of these are presented in Table 1.2, and the reader is referred to Santos and Ferreira (2010) and Chang et al. (2006) for further details.

Table 1.2: Empirical equations for predicting UCS and E from wireline properties (from Santos and Ferreira 2010)

Expression	Comments	Reference
$UCS = 8.374.10^{-21}\rho_b^2 Vp^4 \left(\frac{1+v}{1-v}\right)^2 (1-2v)$	Carbonate rocks CPM Schlumberger's MECHPRO™	Santos and Ferreira (2010)
$UCS = 13.8E^{0.51}$	10<UCS<300 for Limestone	Chang et al. (2006)
$UCS = 25.1E^{0.34}$	60<UCS<100 for Dolomite	Chang et al. (2006)
$UCS = 143.8e^{-6.95\varphi}$	Middle East 30<UCS<150 0.02<φ<0.2	Chang et al. (2006)
$UCS = 135.9e^{-4.8\varphi}$	10<UCS<300 0<φ<0.2	Chang et al. (2006)
$UCS = 17.9e^{-0.91\varphi}$	Carbonate rocks	Farquhar et al. (1994)
$UCS = 174.8e^{-9.3\varphi}$ $Es = 69.05e^{-6.0\varphi}$	Carbonate rocks	Farquhar et al. (1994)
$UCS = 62.567e^{-2.03\varphi}$ $Es = 92.612e^{-4.75\varphi}$	Dolomite from Ghawar field	Ameen et al. (2009)
$UCS = 89.432e^{-5.46\varphi}$ $Es = 86.094e^{-5.34\varphi}$	Limestone from Ghawar field	Ameen et al. (2009)

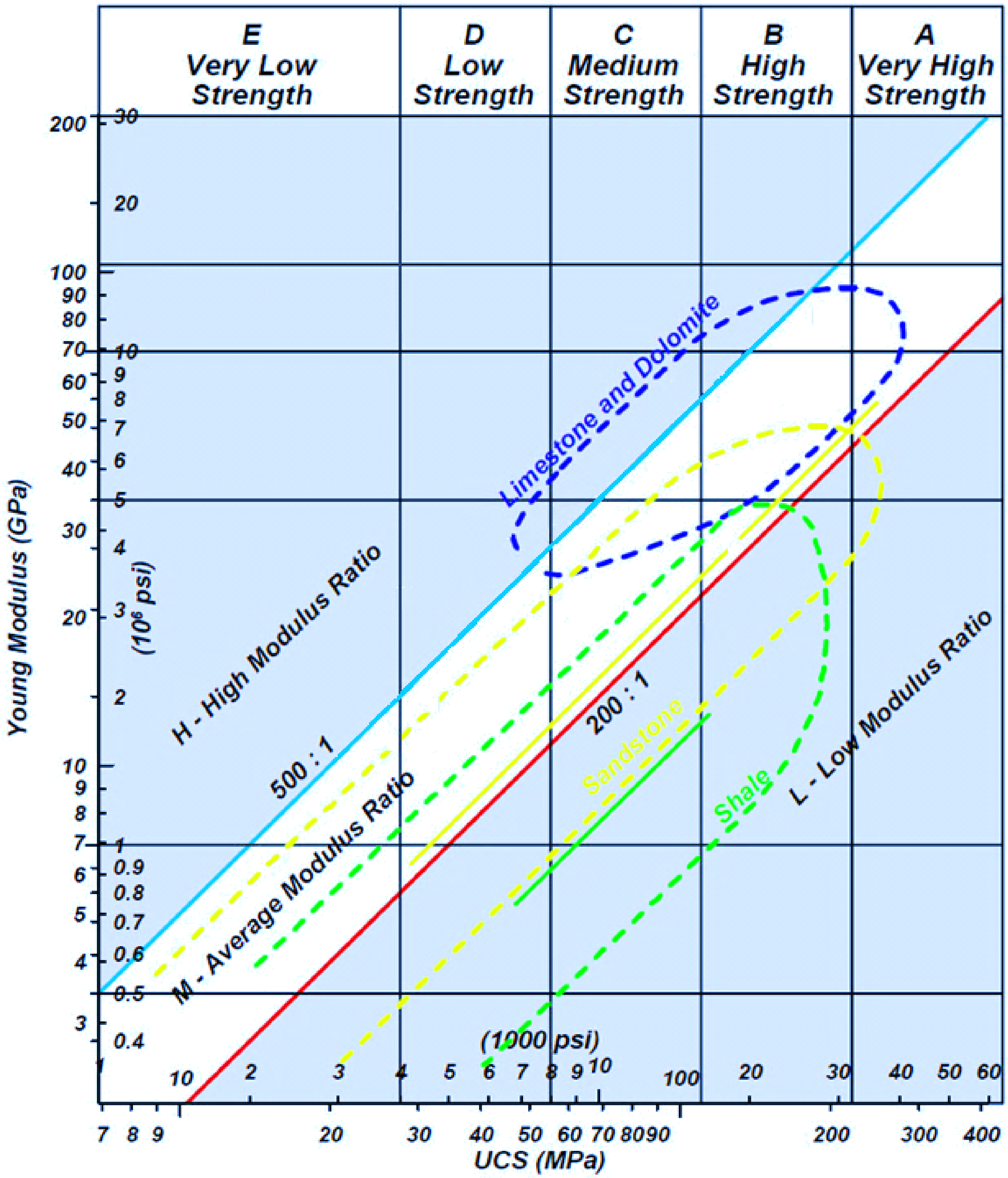


Figure 1.3: Intact properties of sedimentary rocks. Adapted from Santos and Ferreira (2010) and originally published by Deere (1968).

Initial studies on the effectiveness of UCS empirical equations showed that carbonates had a wide variation of strength, regardless of the property that was used to attempt correlation (Chang et al. 2006). Some conservative estimates could be obtained using compressional wave time parameters, but generally the predictive results were poor, and required correlations to a particular field or formation (Chang et al. 2006).

Modulus predictions have been found to be even less reliable than UCS equations. A study by Palchik (2011) further confirmed that there are no general empirical correlations between elastic modulus and UCS. In a study by Santos and Ferreira (2010) of Brazilian off-shore carbonates, the researchers tested the predictive powers of several empirical correlations, with wireline logs and laboratory sample tests. The study found that Schlumberger's MECHPRO™ CPM gave the best results, but other equations that only used porosity as a predictor (such as those from Chang et al. 2006) also produced reasonable results and were simpler to implement. Similar to other studies, equations for elastic modulus were not found to provide good estimates.

Existing research points to porosity as an adequate predictor for UCS and a possible factor for estimating E. Estimating UCS with algorithms based on well log data was used by Arseniuk et al. (2012) to provide UCS values for the Grosmont C unit (or UG2), specifically the vuggy dolomites. This information was used to evaluate the feasibility of using open-hole completions for SAGD in the Grosmont. Material strength is required for borehole integrity; and as an important parameter for determining failure parameters such as cohesion in Mohr-Coulomb failure criteria (Arseniuk et al. 2012). Moduli prediction is relevant for interpreting seismic surveys, analysing reservoir strains, and formulating displacement predictions. Future development in the Grosmont Formation will require accurate ways to predict geomechanical parameters, as well as data of tested samples. These requirements make the present study very relevant to the future production of this resource as well as contributing to the understanding of mechanical behaviour of brittle materials with inclusions.

The current study seeks to further the knowledge on the effect that inclusions (such as vugs) have on the strength and stiffness of a material, which can be measured by a change on UCS and E, respectively. Intuitively, it would be expected that inclusions would reduce the strength and stiffness of the material; however, the amount of this reduction and the effects of shape and location variations are more difficult to quantify and predict. From Griffith (1921) it has been established that the curvature or sharpness of a single defect has a quantifiable effect on the decrease in tensile strength. This is relevant for the UCS value, since rock failure can commonly be dominated by overcoming the tensile strength of the rock. Insight on whether shape is an important factor in material strength and stiffness is important for improving current predictions on mechanical behaviour of vuggy carbonate materials.

The present work also studies the effect of vugs on wave transmission properties. Sonic surveys and laboratory ultrasonic test are used to obtain dynamic stiffness parameters for the material. The presence and location of the vugs can cause dissipation of the signal and an increase of signal noise. Sonic surveys are common in the oil industry and compression and shear wave velocities are characterization parameters for the material in the smaller strain ranges. Modelling of wave propagation through vuggy material provides some insight on how these results can be interpreted, as well as the first step for future modelling in the PFC3D environment under different pressure and temperature conditions. Furthermore, the relation between E_s and E_d for vuggy samples is studied, and the calibration of PFC3D materials to represent both behaviours is investigated. The study of carbonates under smaller strain ranges is more applicable to the day-to-day operations of the petroleum industry, which mostly functions away from peak stress and failure conditions.

This study presents an initial step to testing and modelling of vuggy carbonates and attempts to further the knowledge on the geomechanical properties of these materials. The lessons learned from this work can be applied to future discontinuum modelling, for simulating different rocks with inclusions, and for other application outside of oil production, such as CO_2 capture and storage. Coupled geomechanical and flow simulations are becoming increasingly more common in the petroleum industry, but for any model to be successful the input properties must be of good quality. The present research hopes to open the path for improving characterization, with eventually advancing larger scale reservoir modelling.

CHAPTER 2: LITERATURE REVIEW

2.1. Continuum and Discontinuum for modelling rock

From the early works of Griffith in 1921, the fundamentals of rock failure have been a topic of interest. Theories and empirical formulas had been proposed, none of which have been able to fully capture all the characteristics of rock mechanical behaviour. With the advent of technology and computer processing power, the rock mechanics field turned to numerical modelling to analyze complex situations that single failure criteria or theoretical models could not resolve. Numerical models for rock are now an established tool for engineering design purposes and case study analysis (Bobet et al. 2009). Numerical techniques can be generally subdivided into two broad groups: continuum or discontinuum models.

Continuum models represent the material as continuous and failure is described implicitly through the use of constitutive laws (Potyondy and Cundall 2004). Discontinuities in the rock, such as joints or fractures, can be incorporated in the model by previously specifying them; however, formation of new discontinuities and rupture of the material cannot be represented. In essence, the material must remain a continuum for the model to proceed. Some examples of continuum methods are Finite Element Method (FEM), Finite Difference Method (FDM), and Boundary Element Method (BEM); available rock modelling software that employs continuum models are FLAC 2D/3D, ABAQUS, PENTAGON, PHASE2, PLAXIS, EXAMINE 2D/ 3D, and DEFE to name a few (Bobet et al., 2009).

Discontinuum models, on the other hand, represent the material as a collection of particles which are joined together at the contacts. These techniques model failure explicitly; with the formation, closure, or coalescence of cracks occurring naturally as an extension of the model definition (Potyondy and Cundall 2004). Primary discontinuum formulations are the Discrete Element Method (DEM) with its several variations, and Discontinuous Deformation Analysis (Bobet et al. 2009). Some of the computer codes available for discontinuum rock modelling are UDEC, 3DEC, LDEC, PFC2D/3D, and YADE.

The use of continuum or discontinuum modelling can be illustrated with the diagram shown in Figure 2.1, which is based Bobet et al. (2009). Part a) and b) are shown as straight forward discontinuum and continuum analysis, respectively. The decision is more complicated for cases c) and d). In case c) the material is continuous in each zone but there are clear large discontinuities across regions. In this example, a continuum method that considers the specific geometric discontinuities could be used; the behaviour of the rock will be controlled by structure and the joints should be explicitly included in the modelling. For case d), the joints are pervasive and in a relatively smaller scale than the problem range; in this case the rock could be modelled as homogeneous material, with some account for the reduced strength due to jointing. A pseudo-continuum model could be applied, with parameters based on failure criteria such as Hoek-Brown with the modifications for GSI (Hoek et al. 2013). These four scenarios are some basic cases and combinations of the above may occur in nature; it is up to the engineer to decide which method is appropriate on a case by case basis, prior to attempting any modelling.

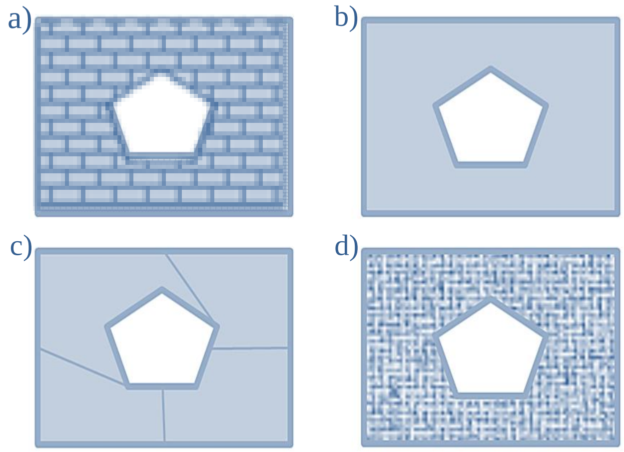


Figure 2.1: Graphical representation of different modelling situations due to the geometry of the problem and rock mass (modified from Bobet et al. 2009)

It can be difficult to determine in which case it is necessary to use discontinuum, continuum, or some combination of the two methods. The choice will depend on the geometry of the problem, the relative scale between the studied range and geological discontinuities, and the material behaviour (Bobet et al. 2009). The biggest deciding factor would be whether relative displacement between regions of the rock is sufficient for material contacts to disappear or be created; in these cases, continuum methods should not be applied (Bobet et al. 2009). It is important to note that regardless of the choice of method, numerical models are still approximations; and the engineer should understand the limitations of each model for a proper interpretation of the results.

2.2. Discrete Element Modelling (DEM)

DEM was first proposed by Cundall and Strack in 1979 for granular assemblies, further development of the formulations was presented in 1988 in two-part papers for the contact identification schemes and mechanical calculations (Cundall 1988 and Hart et al. 1988, respectively). The work by Cundall later developed into commercial codes such as UDEC, 3DEC, PFC2D, and PFC3D. Other early endeavours in DEM were carried out by Rothenburg and Bathurst, first employing disc assemblies (Bathurst and Rothenburg 1988) and later on using elliptical-shaped particles (Rothenburg and Bathurst 1991; Rothenburg and Bathurst 1993).

An important aspect of rock behaviour is to understand that failure is due to either tension or shear. The formation of compression-induced tensile cracks can be explained by schematics such as Figure 2.2; where axially loaded particles are split apart by exceeding the tensile strength of the restraining bond. A model that wishes to represent the mechanics of failure must be able to represent formation and interactions of these cracks (Potyondy and Cundall 2004). This is an advantage of DEM, where the behaviour is represented directly and micro-cracks occur naturally without the need to impose crack locations prior to modelling a material. DEM schemes do not rely on constitutive equations nor do they impose any assumptions and limitations on the macroscopic behaviour (Mas Ivars et al. 2011). Failure arises and evolves on the micro-scale based on geometric conditions and set of assigned properties to the discrete particles and the contacts between them.

Three features that distinguish DEM from other numerical modelling are the following: particles can undergo large displacement/rotations relative to each other, inter-particle forces are a results of the geometric configuration (e.g. particle positions), and the solution is explicit in time (Hart et al. 1988). Theoretically, DEM formulations can allow for any type of shape, but this will have an impact on implementation and computer processing time. The success of a DEM approach will depend on the efficiency of its contact detection system (Cundall 1988). For such, the simpler the shape the faster the program will compute. Rothenburg and Bathurst (1991) presented DEM simulation with 2D elliptical particles; although they claimed an efficient algorithm, the computational effort required for elliptical particles was double of that required for circular shapes.

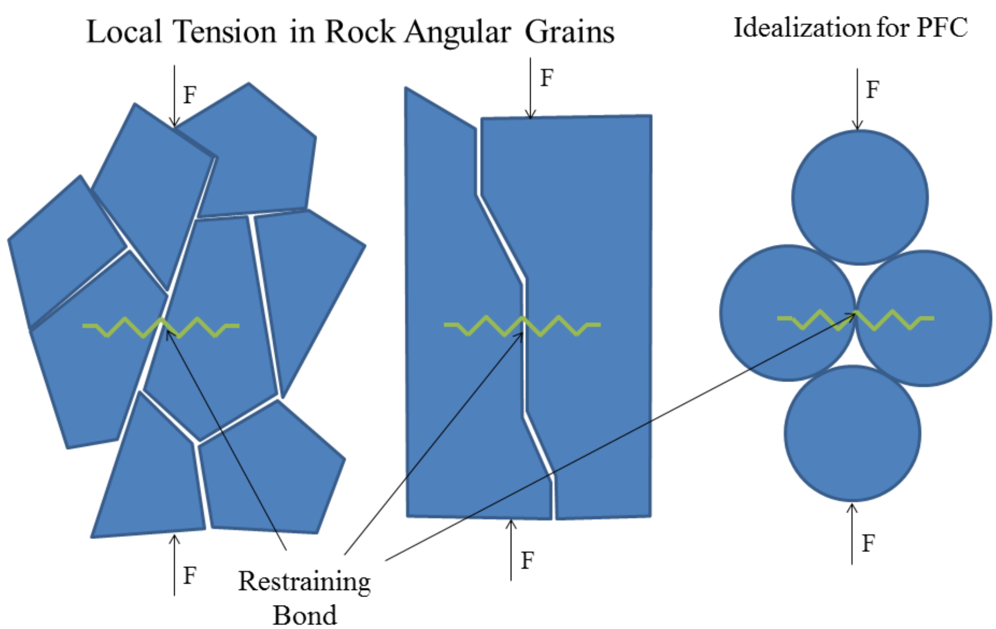


Figure 2.2: Physical mechanism for compression-induced tensile cracking in rock (a and b) and the idealized form (c) for rigid spherical particles (from Potyondy and Cundall 2004)

2.3. Particle Flow Code (PFC3D)

PFC3D is numerical software that implements DEM with rigid spherical particles. This simplifies the computations and contact identification procedures. The calculations cycles alternate between the applications of two laws: Motion and Force-Displacement (Itasca 2008a). Force displacement laws are applied to the contacts, obtaining resulting forces and moments exerted on each contact due to the relative position of the spheres. These forces are then applied to the spheres with the Motion laws (i.e. Newton’s Second Law) resulting in particle displacement (Itasca 2008a). Once the new sphere positions and contacts are updated the cycle repeats itself with a new application of Force Displacement laws.

In PFC3D particles can be bonded using contact bonds, parallel bonds, Hertzian contact law, as well as alternate contact models prescribed by the user. Contact bonds can only transfer forces (either shear or normal), while parallel bond can carry additional moment loads (Itasca 2008a). For example, the Bonded Particle Model (BPM) within PFC3D simulates rock as a spherical grains or balls cemented together through parallel bonds (Potyondy and Cundall 2004). A short summary of this model is described below, for more detailed descriptions refer to Potyondy and Cundall (2004) and Itasca (2008a).

2.4. Bonded Particle Model (BPM)

The BPM is a DEM implemented in PFC2D/3D, and it is based on a combined grain-cement behaviour which is described by a set of micro-properties. This is graphically presented in Figure 2.3. The grain behaviour of the balls is described by three parameters for each ball: normal stiffness, shear stiffness and friction coefficient (μ). Similarly, the stiffness can also be described by choosing a modulus (E_c) with an accompanying normal to shear stiffness ratio (k_n/k_s). The cement behaviour is described by normal and shear stiffness (k_n,k_s), tensile (σ_c) and shear strength (τ_c), and bond-radius factor (λ) for a total of 5 parameters (Potyondy and Cundall 2004). The tensile and shear strengths provide the limit of available strength that can be carried by that contact, if this value is exceeded the contact “breaks” or is terminated and a crack is formed. The bond-radius multipliers determines the area of the bond, but cannot be larger than 1; i.e. bond disk radius cannot be larger than the smallest ball radius in contact. For the classic BPM model described by Potyondy and Cundall (2004) the bond-radius multiplier is always given a value of 1 for modelling rocks; lower values may be used to represent material that is weakly cemented.

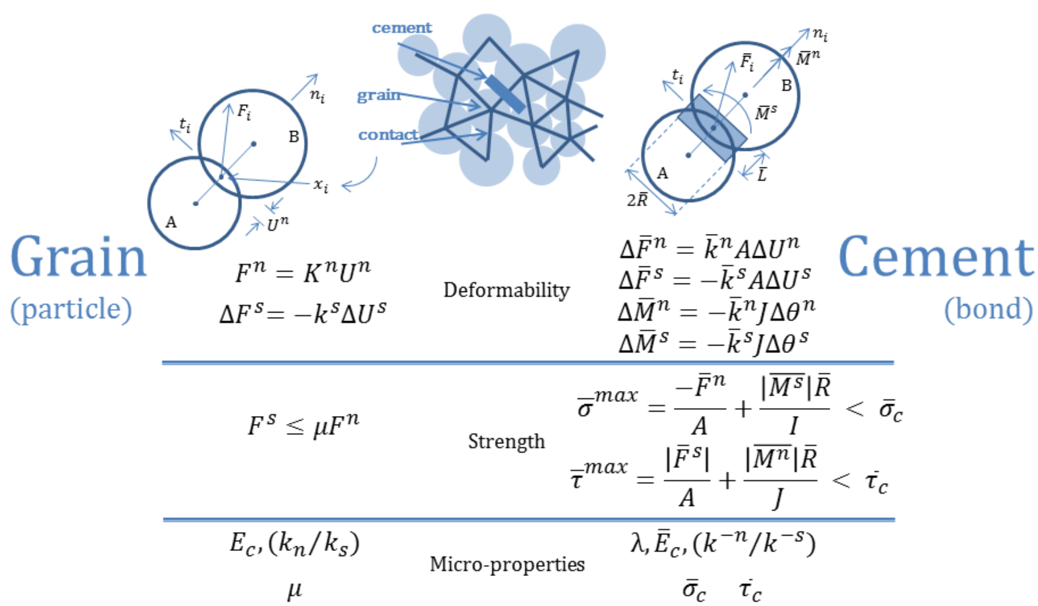


Figure 2.3: Force-displacement behaviour and corresponding micro-properties for the BPM based on a grain-cement system (modified from Potyondy and Cundall 2004)

An important parameter that needs to be prescribed is particle radius; this controls the resolution of the model but is not equivalent to altering mesh-size in a continuum model. In the BPM the ball radius is not a free parameter and changing it while keeping other properties constant will still have an effect in the macroscopic behaviour (Potyondy and Cundall 2004). Scaling micro-properties in PFC3D is complex and requires careful calibration. Both the particle modulus and the cement modulus need to be scaled with the particle radius in order to maintain the same macroscopic stiffness, refer to Figure 2.4.

Cho et al. (2007) found that decreasing particle size also decreases the ratio of tensile to compressive strength, as well as reduces the amount of dilation in the sample.

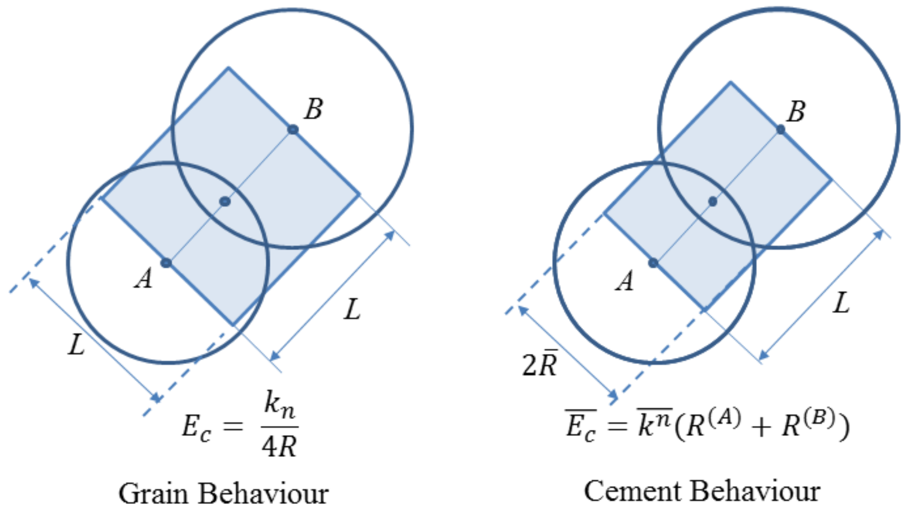


Figure 2.4: Particle size dependency on the modulus for both particle and cement behaviour in PFC3D (from Potyondy and Cundall 2004)

The process of creating a virtual rock in the BPM is accomplished in four steps: generate particles in the specified sample shape, install specified isotropic stress, ensure all particles are contacting at least three other balls (floater elimination), and introduce the parallel bonds (Potyondy and Cundall 2004). The standard sample shapes that can be created are cylinders, spheres, or square prisms; however, user code can be developed to create material in differently shaped vessels.

Once the four steps are completed, the sample can be tested in the existing PFC virtual laboratory environment for triaxial, uniaxial compression, polyaxial, or tensile tests. The tests are generally servo controlled and the walls used during confined tests remain rigid. Measurements during virtual testing can be made in three ways: based on wall measurements, gauge ball readings, or average data from measurement spheres (refer to Itasca 2008a for further details).

There is great flexibility for creating many types of materials; the difficulty is in creating the *correct* material that can model the desired behaviour. At this stage, the need for calibration becomes evident. Part of the difficulty of PFC3D (and other DEM codes) is that micro-properties cannot be directly measured in the laboratory and the choice of these properties is largely a trial and error procedure. It is during the calibration process that the first limitations of PFC became apparent, particularly when attempting calibration of a single set of micro-properties to simulate different tests or stress-paths.

2.5. Limitations of PFC3D and the BPM

The macroscopic response in PFC3D is obtained by systematically altering micro-properties until the physical laboratory data is matched. Although there are some guidelines available, this can become a tedious process. It is generally accepted that the normal to shear stiffness ratio relates directly to Poisson's ratio, and increasing the shear and tensile bond strength will increase the overall strength of the material. However, the relationships are not always quantifiable and may still affect other mechanical aspects such as stiffness, or brittle to plastic behaviour.

Ideally, a model material should be able to match the behaviour of the rock in different conditions. However, it has been found that when UCS behaviour is calibrated, the tensile strength is over-predicted. Potyondy and Cundall (2004) found that the UCS to tensile ratio of PFC3D models for granite was 7.2, whereas the ratio for real samples was 21.5. The over-prediction of tensile strength was postulated to be caused by the use of spherical shapes, and the tensile strength also increases with decreasing particle size (Cho et al. 2007).

Rigid spherical particles have the advantage of reducing calculation time and simplifying the model; however, they cannot fully represent interlocking friction and dilation, which requires more complex geometry (Cho et al. 2007). In PFC3D, there is less limitation on particle rotation, which adds additional moment load to the parallel bond and eventually contributing to bond breakage. This freedom of rotation for the spheres decreases the UCS to tensile ratios and also makes the failure enveloped unrealistic. Potyondy and Cundall (2004) found that the friction angles for PFC3D calibrated to granite were low, and the failure envelope too flat. Rothenburg and Bathurst (1991) demonstrated that by using elliptical instead of spherical particle larger friction angles could be obtained; furthermore, the elliptical assemblies had better stability allowing for wider density range and more realistic dilation behaviours were observed. Rothenburg and Bathurst (1993) found that introducing some eccentricity in the particles within the assembly improved the strength due to increasing restriction on particle rotation.

Potyondy and Cundall (2004) as well as Cho et al. (2007) successfully improved the behaviour of the BPM by grouping particles into clusters and clumps, respectively. These modifications can be considered as different ways of introducing eccentricity to the particle assembly. These and other improvements on the BPM have focused on obtaining more realistic behaviour to properly model rock failure, and are discussed later in this chapter. The following is a summary of research work, which has been conducted with PFC2D and PFC3D, with a focus on the modelling of soft rocks and rocks with inclusions or voids.

2.6. Literature review of PFC3D modelling work

The first major research with PFC3D was calibration and comparison to Lac du Bonnet granite and the rock behaviour observed within Underground Research Laboratory (URL) during the excavation of the Mine-by Experiment tunnel (Potyondy and Cundall 2004). The advantage of modelling Lac du Bonnet was that the fracture behaviour of this material was well characterized with extensive laboratory testing by Martin and Chandler (1994); providing a wealth of data for calibration and validation of PFC3D models.

Hazzard and Young (2000) provided model calibration to Westerly Granite and Ekofisk Chalk. The behaviour of chalk was markedly different from granite, presenting lower strength and stiffness. To obtain this behaviour Hazzard and Young (1994) increased the porosity of the sample allowing voids within the material. The increase in porosity produced a reduction in stiffness and changed the behaviour from brittle to more strain hardening; authors noted that PFC2D was able to match laboratory behaviour for different porosities without changing micro-parameters (Hazzard and Young 2000). This provided the first indication that PFC could be used to model softer rocks without the introduction of additional constitutive laws.

Further modelling of soft porous rocks with PFC was carried out as an investigation into potential nuclear storage in the Yucca Mountain by Bechtel SAIC Company (Bechtel 2004). This investigation involved extensive characterization and modelling of lythophysal and non-lythophysal tuff. PFC2D and PFC3D were used to investigate the impact of lythophysae (voids) on the behaviour of the material. Initial simulations were done with spherical voids to represent lithophysal porosity (refer to Figure 2.5 for summary). The results of these tests showed the E_s and the UCS decreased exponentially with increasing porosity for both PFC2D and PFC3D models (Bechtel 2004).

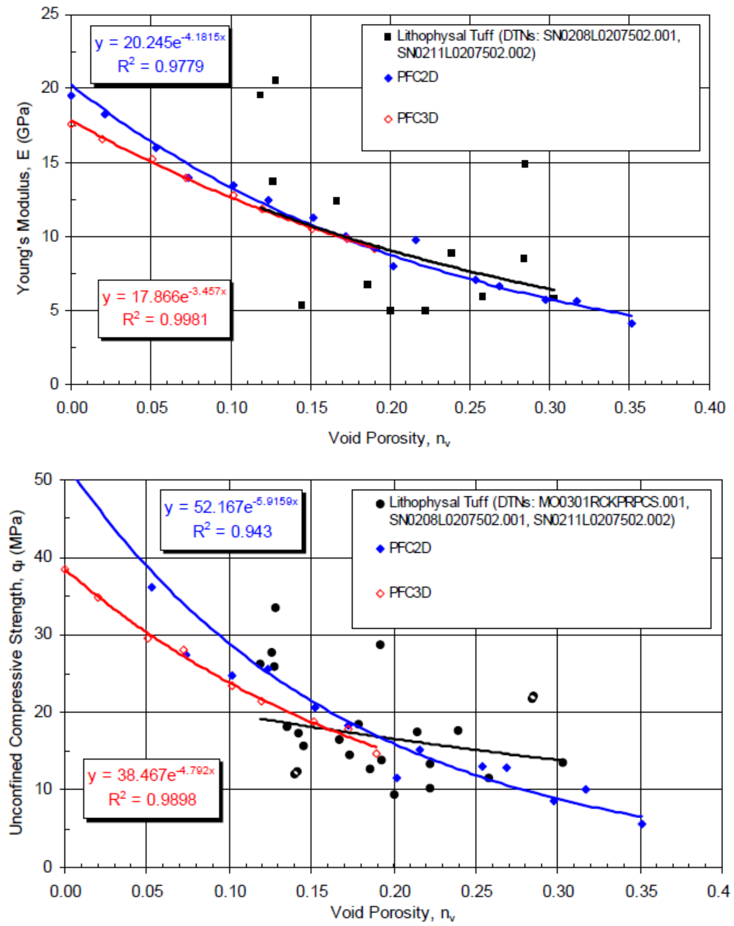


Figure 2.5: PFC2D and PFC3D simulations for varying porosity of tuff using spherical voids (from Bechtel, 2004)

Additional PFC2D modelling was carried out with the real lythophysae geometry based on panel maps from the rock in the field (refer to Figure 2.6); the simulations were limited to two dimensions but stress concentration and tensile fracture at the edges of the lythophysaes were observed during the modelling. The authors concluded that PFC was able to reproduce the behaviour of both lithophysal and non-lithophysal rocks at failure, providing some insight on the mechanism of cracking in the presence of lithophysae that caused the reduction in strength and stiffness with increasing porosity. This research provided encouraging results for the modelling of weak rocks with discontinuum techniques.

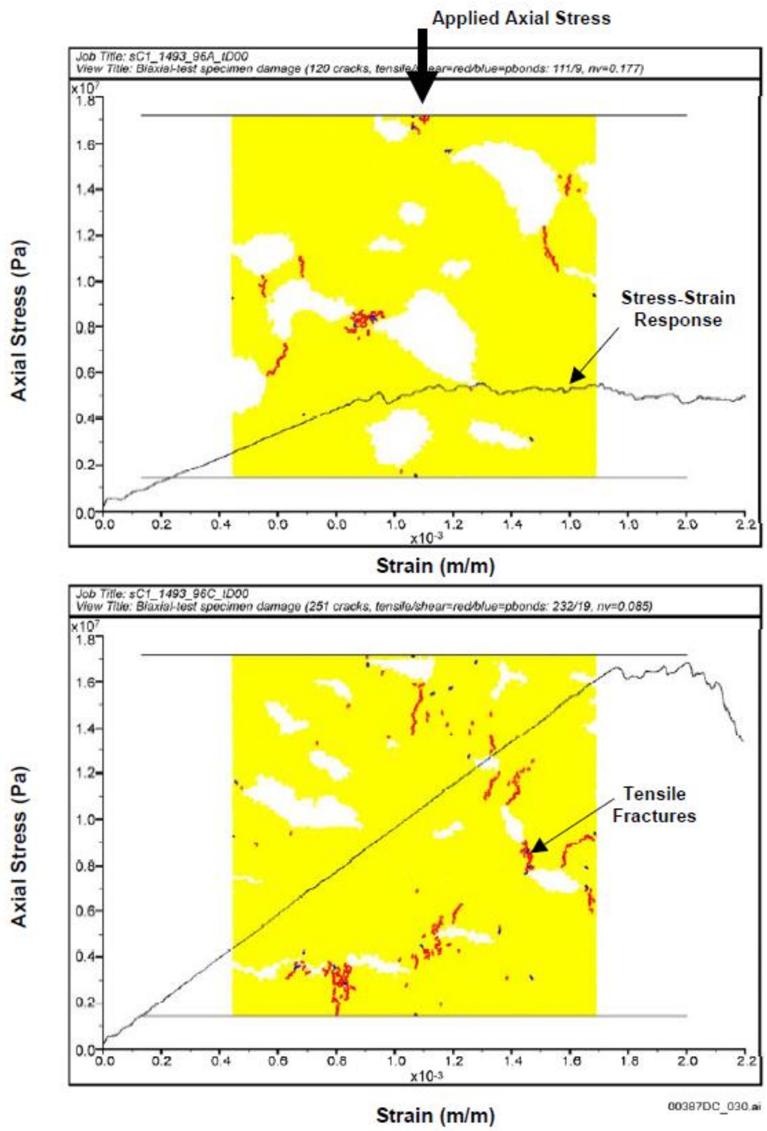


Figure 2.6: PFC2D simulations of lithophysae tuff (from Bechtel 2004)

PFC was also used to study the behaviour of fractured rock mass by Park et al. (2004). They compared UDEC and PFC2D with discrete fractures, systematically increasing the number of joint sets in the PFC2D models. The results for both PFC2D and UDEC were similar, and produced a decrease in both E_s and peak strength with increasing number of joint sets. The number of joints was observed to have an effect on the post peak behaviour, changing from brittle to perfectly plastic as the number of joints increased (Park et al.2004).

The behaviour of the micro-parameters in PFC and their effect on overall behaviour was studied by Holt et al. (2005). They compared PFC models to control experiments using glass beads and epoxy, which were compared to PFC's spherical particles and parallel bonds, respectively. The controlled experiments were carried out to provide more quantitative comparison between micro-properties and known properties of the glass and epoxy. In addition, Holt et al. (2005) created different tests environments in PFC that could provide further calibration aid. Wave velocity tests were conducted to provide correlation with dynamic stiffness parameters, and a scratch test was established to provide additional strength parameter relations (Holt et al. 2005). The wave velocity results are especially encouraging given that geomechanical properties of rocks are commonly estimated through correlations to sonic logs because direct measurements become prohibitive at reservoir scale. Moreover, laboratory rock physics tests are commonly done to obtain dynamic elastic parameters for rock samples.

Calibration of PFC2D to a weak rock also was attained by Cho et al. (2007), who used a clumped PFC2D model to simulate "Sulfaset", a synthetic weak rock used in setting anchor bolts. Cho et al. (2007) were able to reasonably match the UCS, stiffness and failure envelope; however, the initial non-linearity present in the stress vs strain curve of the laboratory data could not be modelled (see Figure 2.7). This non-linear segment of the curve is described as the crack closure region by Martin and Chandler (1994) and it is caused by the closure of existing micro-cracks within the sample. It is expected that softer rocks would have a more distinct crack closure phase; however, PFC2D was not able to simulate this.

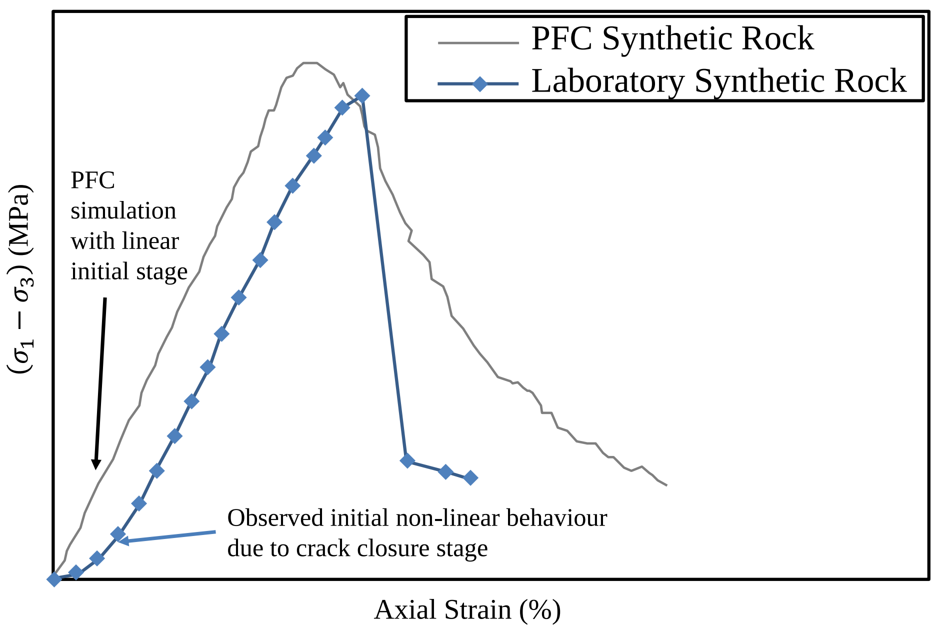


Figure 2.7: Limitations of PFC in simulating weak rock laboratory results (modified from Cho et al. 2007)

Cho et al. (2007) stipulated that the reason the model could not achieve non-linearity was due to not including flaws within the material prior to testing; further hypothesizing that the non-linearity could be achieved by randomly applying pores or cracks within the model. However, as can be observed from research by Bechtel (2004), PFC models with the inclusion of random pores do not present that initial non-linearity (refer to Figure 2.6). Furthermore, research conducted by Schöpfer et al. (2009) in which PFC3D models were generated with random pores and non-bonded contacts to represent cracks did not exhibit the initial non-linear behaviour. This issue is explored further in the remaining chapters of this thesis, where PFC3D vuggy samples are compared with physical laboratory tests.

Schöpfer et al. (2009) conducted multiple simulations with varying porosity and crack density to produce a common behaviour chart. The work included confined compression and extension tests, simulating the behaviour under different stress paths. However, their modelling of cracks was through non-bonded contacts and did not include a sliding joint model or crack aperture. The authors acknowledge that their models are more representative of lower lithified rocks (i.e. rocks with lower cement or bonding) than highly fractured rocks. The graphs produced are, however, very insightful and show the change in stiffness, strength, Poisson's ratio, and tension at different levels of porosity and crack density (refer to Figure 2.8).

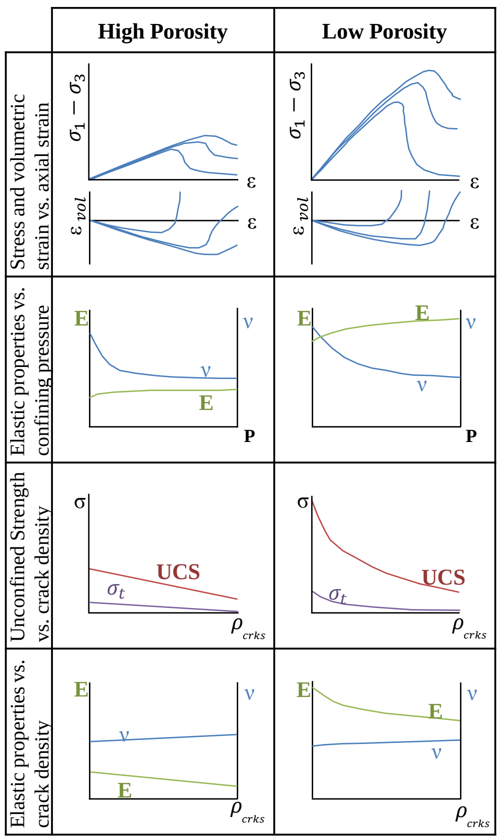


Figure 2.8: Behaviour chart for rocks with crack density and porosity (from Schöpfer et al. 2009)

A more recent study on the effect of random holes on the UCS strength of rock was presented by Xie et al. (2012). Their PFC2D models confirm previous studies that UCS decreases exponentially with porosity. Moreover, the stress vs. strain curves presented for the porous material do not exhibit any non-linearity at the start of the curves, which is in accordance with previous PFC model but contrary to what would be expected for rocks with initial flaws.

2.7. Improvements on the BPM

2.7.1. Enhanced parallel-bond behaviour

This method includes an additional moment-contribution parameter (β) as a new parallel-bond property. This parameter scales the moment in the normal and shear stress equations (see Equations 2.1 and 2.2), reducing or eliminating its influence (Potyondy 2010). The spherical shape causes the contact between particles to be small and the moment contribution on the parallel-bond increases the normal and shear stress on the bond, requiring large strengths to maintain bond contact. Increasing the strength has the consequence of over-predicting the tensile behaviour. Potyondy (2010) found that with this enhancement the tensile strength in direct-tension tests as well as the compression behaviour in UCS tests could be calibrated with a single set of micro-properties. A disadvantage of this method is that it adds an additional factor to calibrate with no physical justification for its value.

$$\sigma = \frac{-\bar{F}^n}{A} + \bar{\beta}\,\frac{|\bar{M}_i^s|}{I}\,\bar{R} \qquad \text{Eq. 2.1}$$

$$\tau = \frac{-\bar{F}_i^s}{A} + \bar{\beta}\,\frac{|\bar{M}^n|}{J}\,\bar{R} \qquad \text{Eq. 2.2}$$

2.7.2. Clusters

Clusters are particles grouped together forming a single shape that forces cracking to occur along cluster boundaries (Potyondy and Cundall 2004). This increases the interlocking between particles, which makes the strength envelope more realistic with a higher slope or friction angle. Cluster particles have a high inter-cluster strength that prevents them from breaking apart; however, rotation between the particles can still occur (Cho et al. 2007). The rotation of the inner cluster particles contributes to the force chains reducing the peak strength of a sample test (Cho et al. 2007).

2.7.3. Clumps

This concept was explored by Cho et al. (2007) in their clumped particle model. Clumps are similar to clusters with the difference that the particles behave as a single rigid body and rotation of the particles is supressed. The difference in particle rotation is presented in Figure 2.9; particle rotation is supressed but the moment loading is still simulated (Cho et al. 2007). In order to ensure the clump size was similar to the particle size in regular models, the original particle size was reduced for grouping of ball into clumps. The clump method developed by Cho et al. (2007) stamped circles of different radius and grouped all balls within that circle into a clump (refer to Figure 2.9).

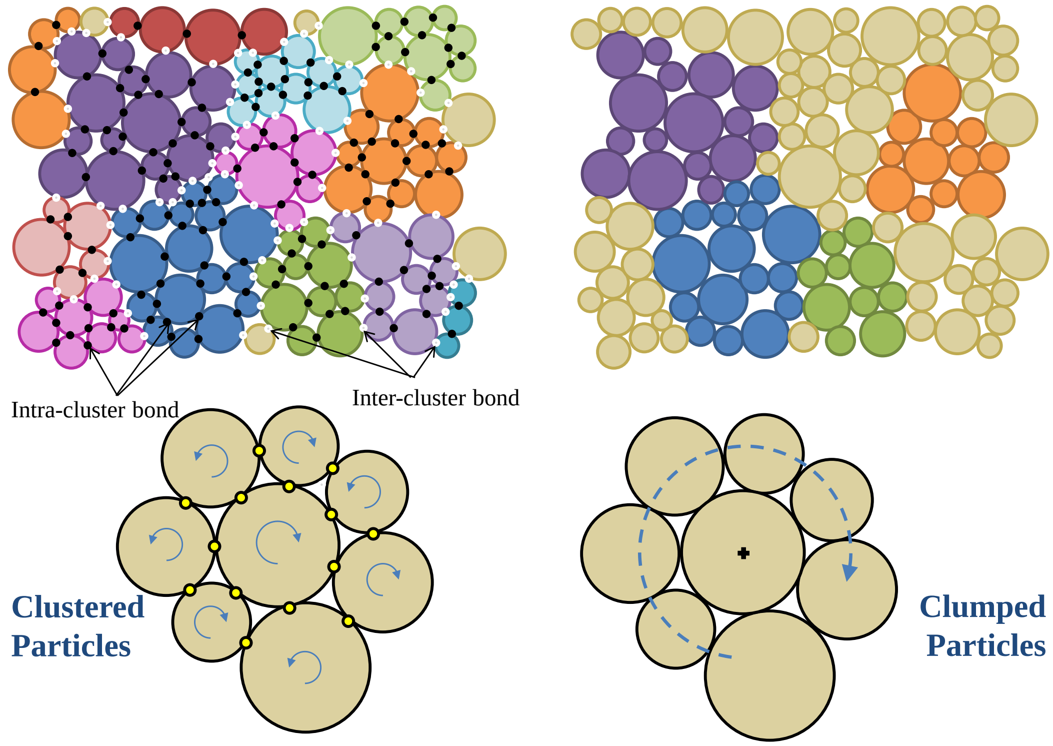


Figure 2.9: Clustered and Clumped BPM models (based on figures from Potyondy and Cundall 2004 and Cho et al. 2007)

Clump logic is now standard in PFC3D version 4.0 (Itasca 2008a) and a granular clumped BPM model can now be developed easily in the PFC3D environment. The current clump logic replaces an existing sphere with clumps of two or more balls within the same space (Itasca 2008a). The current clump shapes provided in the fish tank are peanut (two balls) and tripoid (three balls); however, additional shapes can be easily prescribed with the percentage of each shape that should be placed within the model.

Further modelling with clumps was carried out by Ding and Zhang (2011), who used PFC3D model with clumps to simulate Lac du Bonnet Granite. The clump shape used was a symmetrical five ball clump, which was chosen to be a closer representation to the Lac Du Bonnet granite fragments. Unlike the Cho et al. (2007) clumping method, which resulted in more irregular shapes, the method used by Ding and Zhang (2011) replaced existing ball with a consistent clump shape which was symmetrical and close to spherical in shape. The results were not as favourable as expected and did not produce improvement from the normal BPM model; the probable cause was using a clump shape with a shape close to a sphere (Ding and Zhang 2011). As demonstrated in early simulation work by Rothenburg and Bathurst (1993) it is the eccentricity of the particles that increase packing, coordination number, and eventually strength; thus, the clump shape used should represent eccentricity to produce an improvement in the BPM model.

2.7.4. Specimen Generation and Particle Size Distribution:

Schöpfer et al. (2009) suggested that the sample generation procedure could be adjusted to improve the behaviour of the materials generated in PFC3D. They employed power-law particle size distribution (PSD) instead of a uniform PSD when generating the samples. This produced models with lower porosity and higher radius ratio (r_{max}/r_{min}) of 10, compared to the more common radius ratio of 1.66. The samples generated with power-law PSD had higher coordination numbers, and exhibited higher strength, and stiffness. This effect was to be expected based on previous work by Rothenburg and Bathurst (1993), which showed that the strength of the material is related to the number of contacts (i.e. the coordination number). The friction angles obtained for these specimens were higher and in better accordance to those observed in natural rocks.

Work by Schöpfer et al. (2009) also included non-bonded contacts to model pre-existing 'cracks' or poorly lithified rocks. The addition of non-bonded contacts had a higher effect on the tensile than on the compression behaviour, due to non-bonded contacts capable of carrying some load in compression but none in tension. The specimens with non-bonded contacts presented higher UCS/σ_t ratio than regular PFC3D, and the values were closer to the range for real rock (Schöpfer et al. 2009).

2.7.5. Artificial Neural Network for Calibration

One of the more repetitive aspects of discontinuum modelling is the calibration of multiple micro-properties to obtain an overall behaviour that is characterized by three to four macro-properties. An improvement to this issue was the application of Artificial Neural Network (ANN) to aid in the calibration procedure. Tawadrous et al. (2009) applied different ANN architectures to predict the micro-properties required to obtain a target UCS, E_s and ν_s. Some success was obtained but the authors noted additional samples were being generated to retrain and improve the predictions (Tawadrous et al. 2009). More recently, Guo et al. (2013) developed a user interface to obtain PFC3D micro-properties based on a large number of simulations and an adaptive neural-fuzzy inference system. It is likely that with more research being conducted with PFC3D databases can be built to further facilitate the calibration process in the future.

2.8. Synthetic Rock Mass (SRM) Approach

The SRM approach models rock by representing the two main constituents that affect their behaviour: intact material represented by the BPM, and discontinuities represented by a discrete fracture network with the smooth joint model (Cundall et al. 2008). Figure 2.10 gives a graphical representation of the combination of both techniques to obtain the SRM. The smooth joint contact model allows particles along the joint to overlap and slide past each other, without having to mount over other particles (Pierce et al. 2007)

The SRM has been used in spherical samples with a DFN applied from joint data and subjected to different stress paths (Mas Ivars et al. 2007). From these simulations seven advantages of the SRM approach were described: 1) joint fabric was accounted for explicitly, 2) elastic/inelastic compliance matrix could be derived, 3) non-trivial stress paths could be applied, 4) slip on joints was observed, 5) the samples presented brittle behaviour which could be quantified, 6) break-up of previous continuous areas into discrete fragments, and 7) nature and development of fracture could be observed (Mas Ivars et al. 2007).

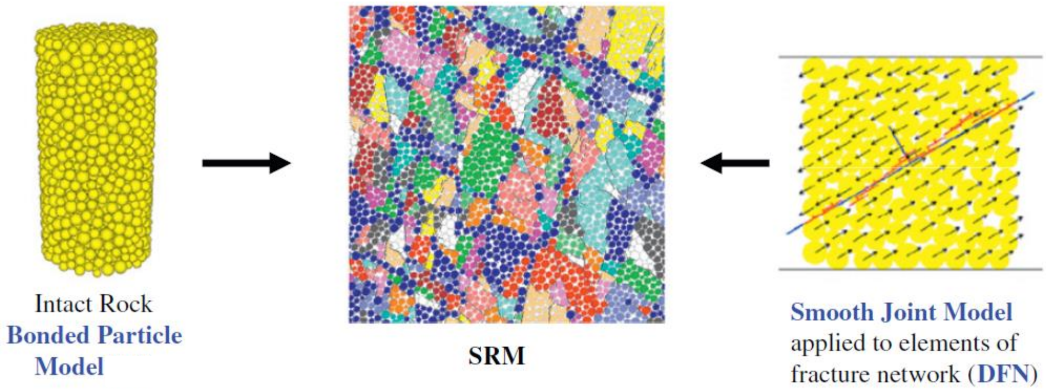


Figure 2.10: Graphical representation of the SRM components (from Mas Ivars et al. 2011)

The principal application for the SRM approach is in the mining industry where the development and progression of rock fractures are important for mining sequences. Mas Ivars et al. (2011) presented two case studies in which the SRM was applied. The first was a mine in Australia in which the ore was being extracted through block caving. The SRM was used to simulate the fracturing during mining and the slips and fracture growths could be correlated with seismic data obtained during monitoring. The second case study presented was the analysis of a slope failure in Palabora mine and the SRM was successful in matching the failure mechanism.

More fundamental utilization of the SRM was carried out by Bahaaddini et al. (2013) to explore the effect of joint configuration and geometric parameters in the behaviour of a rock mass. Bahadini et al. (2013) tested different configurations to explore parameters such as joint orientation, aperture, spacing, and persistency. Tensile wing cracks, stress concentration at the joint tips, and differences between tensile and shear cracks were observed. As expected from previous studies (Hoek and Brown 1980) strength and stiffness are reduced the most when the joints are at 45 to the axial force. Bahaadini et al (2013) further recorded that the failure mode was observed to change with different joint configurations.

2.9. Other Discontinuum Models Available for Rock Modelling

The SRM approach was adopted by Scholtès and Donzé (2012) to tests its validity on modelling mechanism of rock slope failures. Instead of PFC3D, Scholtès and Donzé (2012) used an open source DEM called YADE, which was developed, at Grenoble University (Kozicki and Donzé 2008). The formulation for YADE is similar to PFC3D, using spherical particles bonded to each other at their contacts.

Scholtès and Donzé (2012) used the SRM approach to model rock slopes with existing discontinuities dipping at different angles to the slope. The SRM was able to simulate slippage along existing fractures, stress concentrations at crack tips, and slope toppleling failure (Scholtès and Donzé 2012). Donzé et al. (2013) presented several examples of YADE modelling of jointed rock slopes with increasing complexity, from a single persistent joint dipping at 40º to two random joint sets with 40º and 80º dip angles. Finally, the decrease in strength of rock with increasing fracture intensity was studied using different statistical joint distributions.

Further modifications on the YADE model were published by Scholtès and Donzé (2013), in which the microstructure was enhanced by allowing proximate balls that were not directly in contact with a ball to be considered to be connected (refer to Figure 2.11). This increases the number of possible contact per ball (i.e. coordination number), which is an implicit enhancement to interlocking of particles and increases the strength of the material. As noted by Scholtès and Donzé (2013), this process is similar to the clump method used by Cho et al. (2007) for PFC2D, with the added advantage that complex particle shapes are not necessary. This modification allowed YADE to model material with high UCS to tensile ratios and non-linear failure envelopes; moreover, the authors report more proportional relation between the micro-properties in the model and macro-properties that are measured in the laboratory tests.

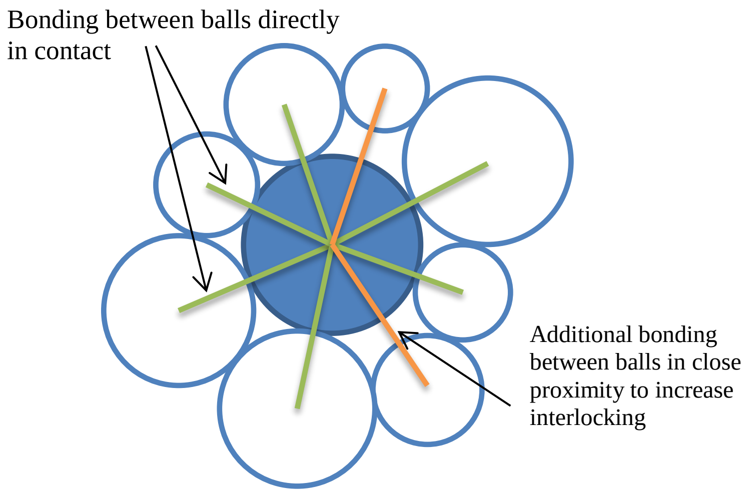


Figure 2.11: Graphical representation of enhanced bonding in YADE which allows connections between balls not directly in contact (modified from Scholtès and Donzé 2013)

2.10. Contributions of Current Research

The current program looks to expand knowledge on the effect of voids or inclusions in rocks. The context of this research is heterogeneous carbonate specimens that present vugs of various shapes and sizes. The first objective is to evaluate the effect of vug geometry on the strength and stiffness on idealized vuggy samples at the laboratory scale. The UCS test environment within PFC3D as well as analogous P- and S-wave velocity tests are used on samples with different vug configurations to determine the changes caused by vugs size, shape, and location. The second objective is to simulate real samples within PFC3D created with the help of computed tomography (CT) imaging technology. This would allow comparing the PFC3D simulations to equivalent physical laboratory test results. It is the hope that these simulations can provide an initial step towards non-destructive tests for samples in which limited core is available, as well as providing insight on up-scaling methods for heterogeneous reservoirs.

CHAPTER 3: PFC3D MODELLING OF VUGGY CARBONATES

3.1. Creations of inclusions in PFC3D BPM

The methodology used to introduce vugs into a BPM was first developed by Nathan Deisman at the University of Alberta, as an extension to the Virtual Lab Assistant excel spreadsheet (Deisman 2008). The fish (Itasca's programing language) code developed allows the introduction of inclusions into previously generated intact samples based on two shapes: a rectangular prism and an ellipsoid. The user could prescribe the length of the three rectangular sides or ellipsoid semi-axes, the location of the centre of the vug and the direction of these axes by defining three rotation angles. These options permit modelling various shapes by adjusting the length and direction of the axes. The vugs can be void or filled; and the infilling material within the vugs can be assigned different micro-properties for a separately calibrated material.

The vugs are created through an iterative process, in which the centre of each ball is examined, if the ball-centre is within the vug volume prescribed by the coordinates and axes it is flagged by the fish code. Subsequently the ball is either eliminated, for the void case, or the ball properties are changed to the fill material prescribed. Moreover, the spheres located at the boundaries of the vugs are identified; this can be used to track the shape of the vugs or to change the properties of the material to represent the interface between vugs and matrix (e.g, refer to Figure 3.1Figure 3.1).

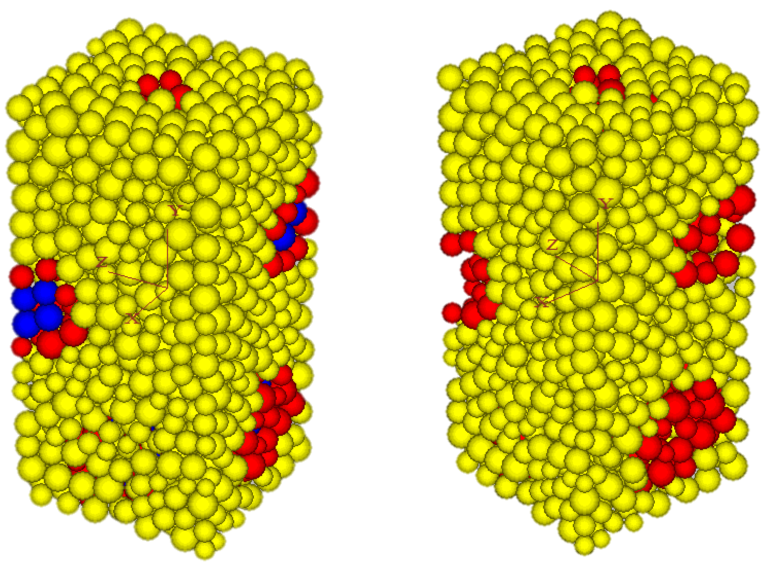


Figure 3.1: Filled and void vuggy samples in PFC3D, yellow balls represent intact materials, red balls represent vug-matrix interface, and blue balls represent the vug infilling material.

The volume of the balls assigned to the vugs is tracked in order to determine the vug to intact material volume ratio, which is calculated as the ratio between the total volume of balls assigned to vugs with the initial total volume of balls. This ratio is used as an analogous measurement for vuggy porosity, which is calculated as total volume of vugs divided by total volume of the sample. The ratio is used to establish a numerical value for comparison with the different PFC3D simulations. The vug volume ratio is the main geometric parameter used in this study to evaluate the effect of vug volume on the strength and stiffness of samples.

3.2. Single and Two Vug Simulations

UCS test simulations of vuggy specimens with different resolution and vug sizes were conducted. Cylindrical samples of 60 mm height and 30 mm diameter were created using several particle sizes with the BPM. The micro-properties used in this initial trial were those used in the verification problems for the PFC3D software (ITASCA 2008b), with minimum ball radius of 1.22 mm (resolution of approximately 9 particles across sample diameter). The other materials used ball radius of 0.8 mm (resolution of about 14 particles for width) and radius of 0.5 mm (resolution of approximately 22 particles per diameter). As has been noted in previous studies, ball radius affects the behaviour of PFC samples and would require some calibration to obtain the same overall response. For this study, this additional calibration was not carried out, and the behaviour of the vuggy material is compared to its "parent" by normalizing the results to the intact material samples.

Spherical vugs of varying radius from 2.44 mm to 9.76 mm were placed in the centre of the cylindrical sample. Although the vugs are assigned to be spherical in shape, the shape obtained was dependent on the resolution (refer to top row of graphics in Figure 3.2). The higher resolution the closer the shape will be to the assigned shape; this appears to be of lower importance for the larger vugs (see bottom row of Figure 3.2). The results of the virtual UCS test are presented in Figure 3.3; the results shown are the compressive strength of the test normalized to the intact strength. The increase of the resolution reduces variance in the results but appears to produce similar averaged results; and may indicate that a lower resolution (albeit still capable of reasonable geometry representation) can be used to represent the vugs.

A second set of simulations with vugs of 4.88 mm were located in the centre of the sample, at 5 mm above the centre (offset in y direction along the length of the sample) and 5mm offset in the z or x direct, which are perpendicular to the sample length (refer to Figure 3.4 for results). The change in location of the vug had some effect on the strength of the sample. The influence was found to be smaller than varying the size of the vugs, and some scatter in the data was observed for the vugs that were offset by 5mm in the x and z direction. The UCS for single vug simulations was observed to reduce almost linearly with increasing size of vug. However, as soon as a second vug is added this relation changed as shown in Figure 3.4. This shift in behaviour shows that interaction between the vugs may be significant.

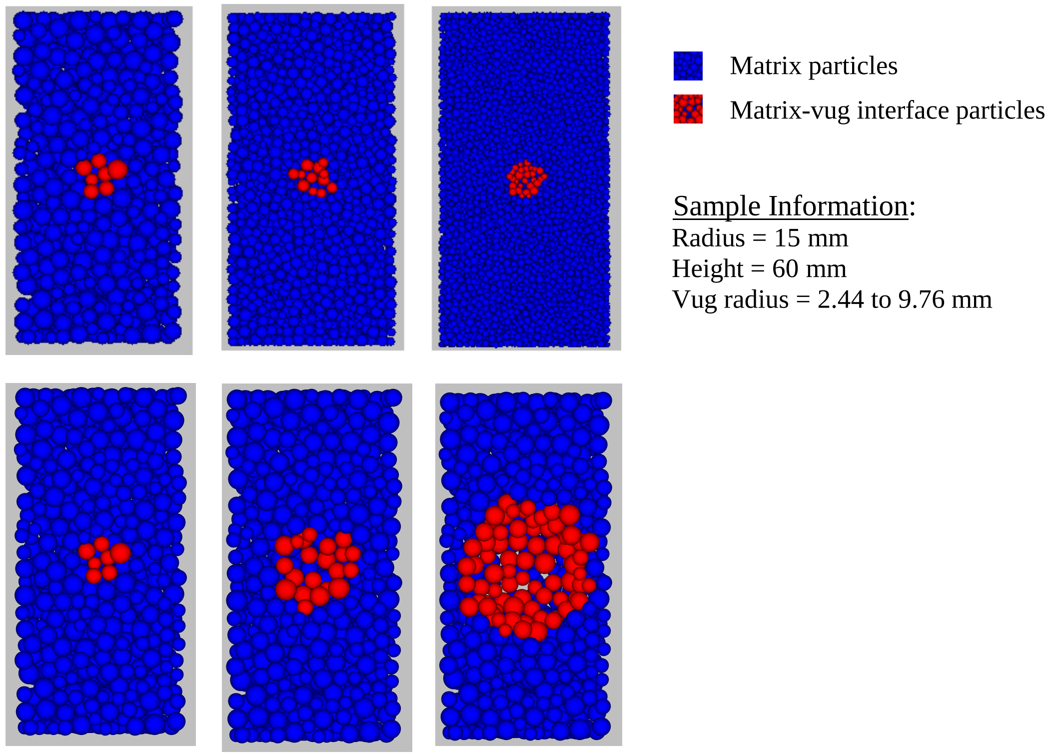


Figure 3.2: Single vug simulations. Top row: same vug size - different resolutions. Bottom row: same resolution with varying vug size

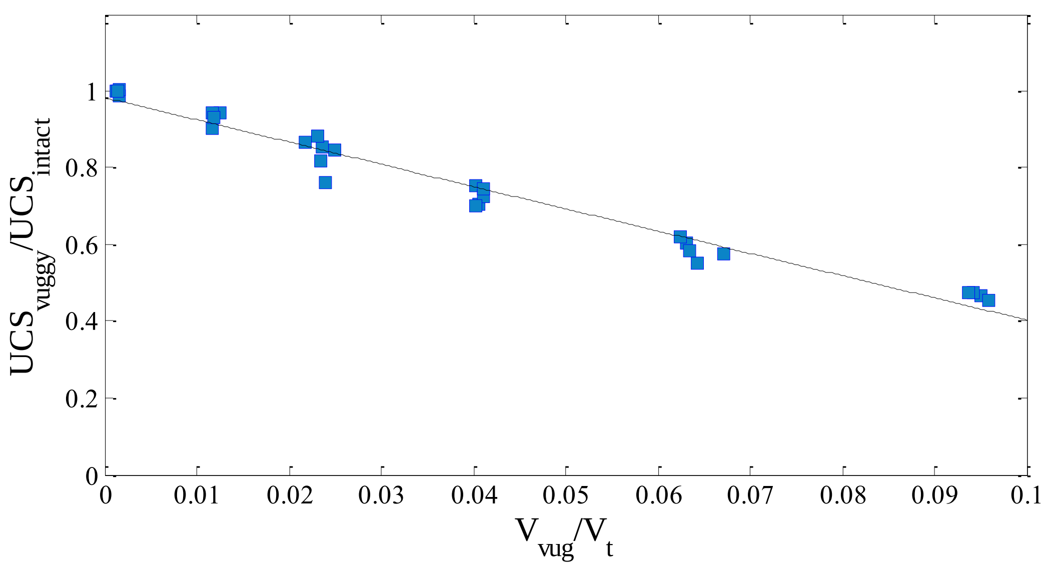


Figure 3.3: Single vug simulation results, UCS values normalized to the intact material

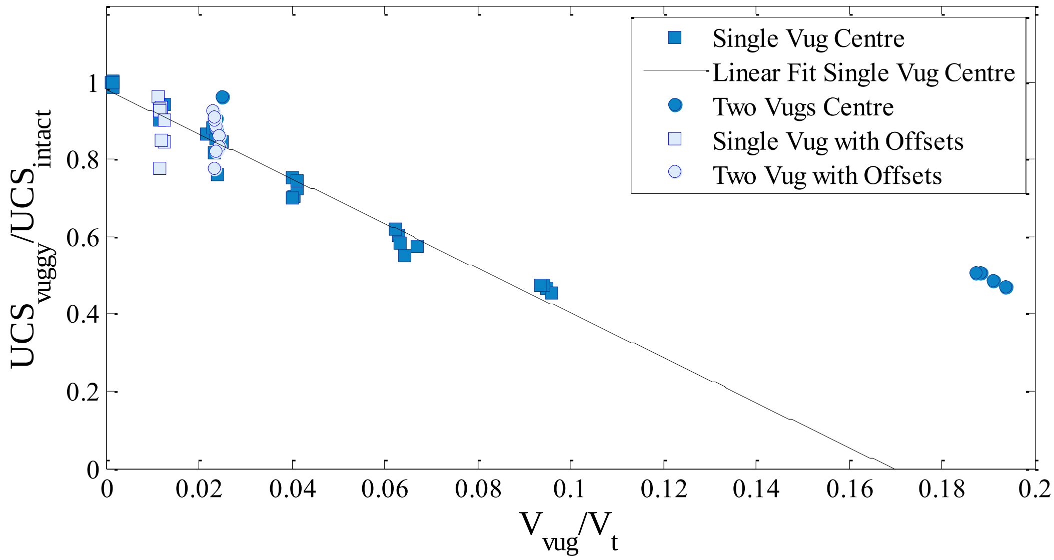


Figure 3.4: Single and two vug simulations, values are normalized to the intact material

The simulations discussed above consider only spherical vugs. The next shape that was studied was ellipsoids. Samples with a single elliptical vug were simulated with different orientation for the axis of the ellipsoid. Figure 3.5shows the change in the normalized UCS at different ellipsoid orientations. The orientation angle beta is measured as shown in Figure 3.5Figure 3.5.

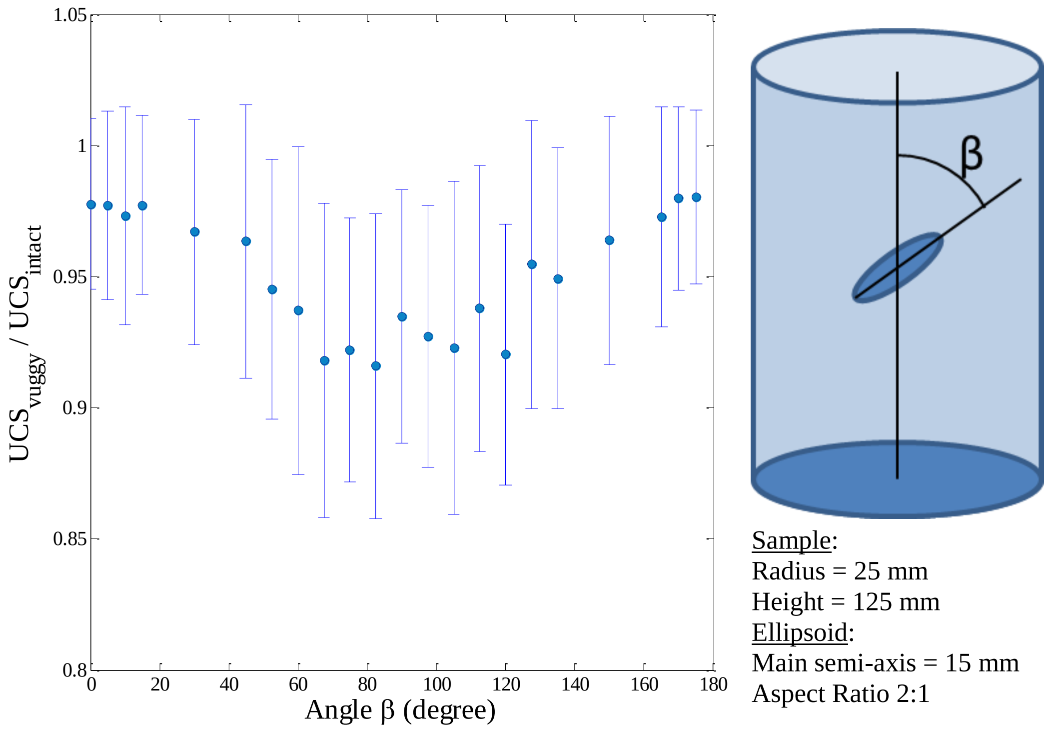


Figure 3.5: Single vug ellipsoid simulation results, variations in the UCS with changes in the ellipsoid orientation (error bars represent the standard deviation based on results from ten realizations)

Unlike similar experiments with fractures, which are continuous throughout the sample, the lowest strength did not occur at 30-45 degrees (Hoek and Brown 1980) but between 80 and 120 degrees. The slight double dip observed is similar to the graphs presented by Hoek and Brown (1980) for two discontinuities that intersect at 90 degrees. The three-dimensional effect captured in the simulations shows a promising result that vug effects can be modelled by PFC3D. The outcomes presented for the single ellipsoid vug were the average of ten sample realizations (material resolution was 7.5 particles per diameter) with equal properties but using different seed numbers. The graph is not exactly symmetric, which may be due to the discontinuous nature of the material that causes the shape to be slightly irregular. Further studies could be made reducing the particle size or increasing the number of sample realizations.

3.3. Modelling of Vuggy Carbonates

3.3.1. Parameters for Carbonates in PFC3D

The next step towards modelling of vuggy carbonates was to create idealized PFC3D samples with micro-properties that reflected strength and stiffness of a carbonate material. The change in behaviour from intact to vuggy samples was investigated using four different materials. All samples created were of 25 mm radius by 125 mm height (length to width ratio of 2.5). The BPM (as presented in Potyondy and Cundall 2004) was used for the numerical simulations by selecting parallel bond with radius multiplier of 1 for all models. The materials chosen (M1 through M4) are intended to represent carbonate rocks of low to high strength, and low to high E_s/UCS ratio.

The micro-properties used for each material are presented in Table 3.1; and the observed UCS and E for the intact specimen is presented in Table 3.2. The bond strengths shown in Table 3.1 are presented as an average plus/minus standard deviation. The UCS and E_s for most of the intact materials created are plotted in the summary graph proposed by Deere (1968) for sedimentary rock (refer to Figure 3.6). The micro-properties for materials M1 to M3 were selected so the samples behaviour would fall within the limestone-dolostone area, with different strengths and E_s/UCS ratios for comparison. The properties for the M4 materials were chosen to represent a much lower strength and stiffness, as an extreme case. Consequently they fall outside the carbonate range plotted in Figure 3.6.

Table 3.1: Micro-property parameters for the BPM simulating carbonate rocks

Material	Min Radius (mm)	Rmax/Rmin	Bulk Density (kg/m^3)	Ball Modulus (GPa)	Ball K_n/K_s	Ball Friction coefficient	Parallel Bond E (GPa)	Parallel Bond K_n/K_s	Bond Normal strength (MPa)	Bond Shear strength (MPa)
M1.1	1.5	1.66	2630	5	2.5	0.5	67	2.5	166 ± 33.2	166 ± 33.2
M1.2	1.0	1.66	2630	5	2.5	0.5	67	2.5	166 ± 33.2	166 ± 33.2
M1.3	0.85	1.66	2630	5	2.5	0.5	67	2.5	166 ± 33.2	166 ± 33.2
M1.4	0.75	1.66	2630	5	2.5	0.5	67	2.5	166 ± 33.2	166 ± 33.2
M2	1.5	1.66	2630	40	2	0.5	75	2.5	250 ± 50	240 ± 48
M3	1.5	1.66	2630	7.5	2.5	0.5	70	1.5	140 ± 28	100 ± 20
M4.1	1.5	1.66	2630	1.5	2.5	0.5	2.2	2.5	51 ± 12.75	53 ± 13.25
M4.2	1.0	1.66	2630	1.5	2.5	0.5	2.2	2.5	51 ± 12.75	53 ± 13.25

Table 3.2: PFC3D UCS test result parameters

Material	UCS (MPa)	E_s (GPa)	v_s	E_s/UCS	Sample Resolution (particle/diameter)
M1.1	118.7	33.3	0.17	280.4	12.5
M1.2	127.1	33.7	0.17	265.5	19
M1.3	130.3	34.5	0.16	265.1	22
M1.4	135.8	35.0	0.17	257.8	25
M2	194.9	66.3	0.18	340.2	12.5
M3	73.9	38.8	0.15	524.3	12.5
M4.1	50.2	1.8	0.24	36.9	12.5
M4.2	60.6	1.9	0.25	31.3	19

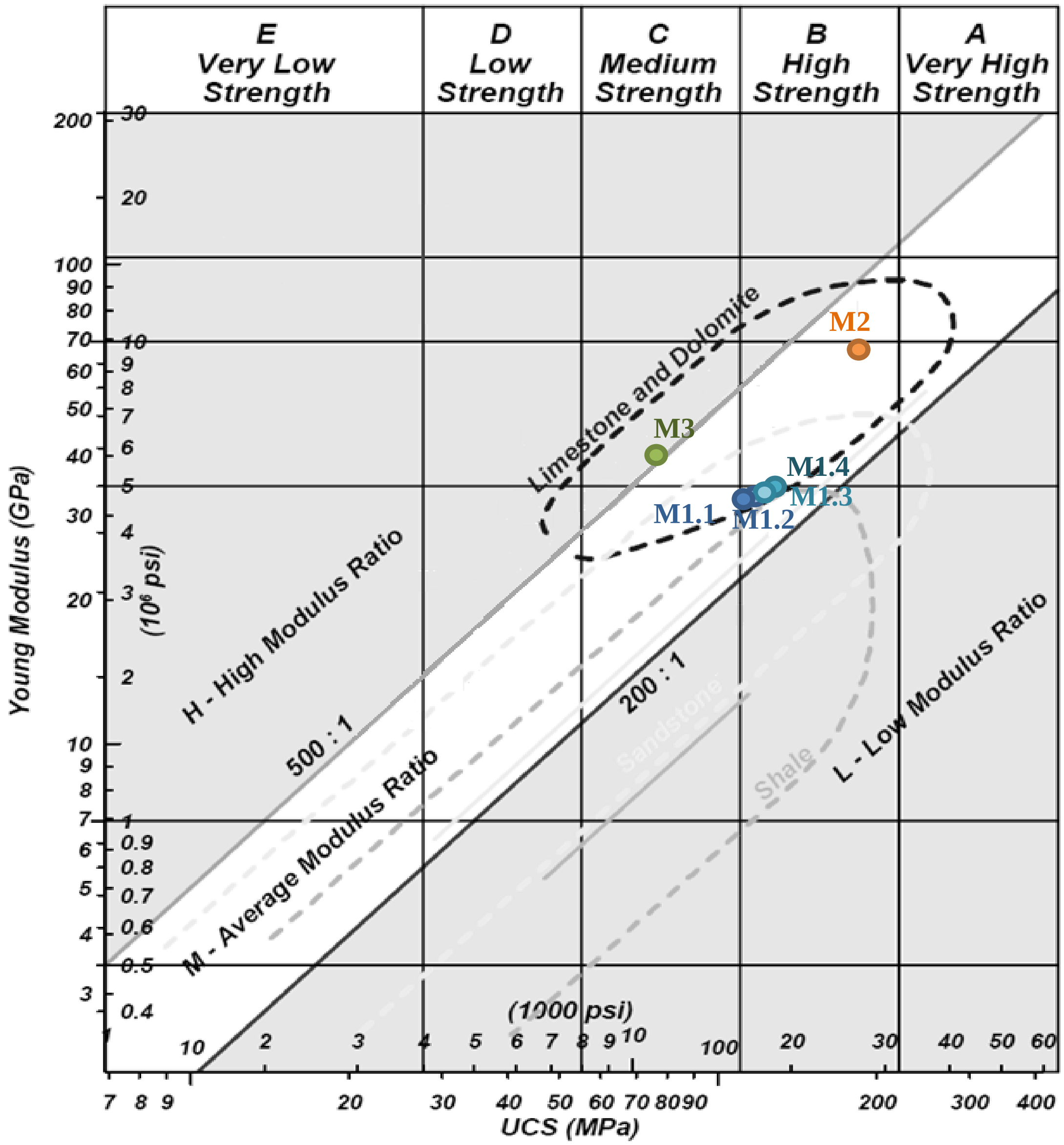


Figure 3.6: Engineering classification for intact rock proposed by Deere (1968), figure modified from Santos and Ferreira (2010)

The E_s/UCS ratios (shown in Table 3.2Table 3.2) are mostly within the range of carbonate data found by Palchik (2011) and Deere (1968). Materials M1.2, M1.3 and M1.4 use the same micro-properties as M1.1 but with different particle size. Contrary to continuum models, the particle size does not only control resolution in a discontinuum model, but will also affect the macro-response (see Table 3.2 for UCS and E_s values); other parameters remaining equal (Potyondy and Cundall 2004; Cho et al. 2007). The M1.1, M2, M3, and M4.1 materials had an average resolution of 12.5 particles across the diameter; and material M1.2 and M4.2 had a resolution of 18.8 particles in the same dimension. Materials M1.3 and M1.4 had a resolution of 22 and 25 particles per diameter, respectively. The main purpose of using a higher resolution was to determine if it affected the geometrical representation of the vugs, and its influence on strength and stiffness. The use of a variety of strength, E_s/UCS ratios and resolutions allow for the extension of these results to a wider range of carbonate materials.

3.3.2. Vug Configurations

3.3.2.1. Randomly Located Vugs

Vuggy samples with randomly placed vugs were used to evaluate the change in geomechanics properties between intact specimens and specimens with inclusions. Vug networks of 5, 10, 15, 20, and 22 voids were created, with the vugs placed randomly within the sample and with no overlapping allowed. Each network of voids was further modified with six different shapes but the vug centre coordinates were maintained a constant. A total of 30 vug networks were created for each material, generating more than 200 vuggy samples. The vugs within the same network were approximately of equal shape and size, with some variation caused by the position of the balls in PFC3D.

The shapes used in the current simulations (refer to Table 3.3 and Figure 3.7) were spherical, prolate ellipsoids (cigar shape), oblate ellipsoids (disk shaped), and scalene (all sides unequal) ellipsoids. The orientation of the ellipsoidal semi-axes was randomly chosen and remained constant for the different shapes; in the case of the spheres the axis orientation is not relevant. The vugs were placed randomly using a random-normal distribution function, which would generate more vugs in the middle of the samples and the additional condition that the vugs could not overlap each other. Some overlapping with the edges of the samples was allowed in the samples with high number of vugs. The voids were inserted after the sample was generated by eliminating balls that were within the vug volume location, as described at the beginning of this chapter. The vug volume ratio was tracked and used to evaluate the change in geomechanical property with increasing volume of vugs. The samples were not replicated with different seed numbers, but the results were normalized and compared to each other allowing some evaluation of the scatter.

Table 3.3: Shape parameters for the vugs used in the randomly ordered samples

Vug type	Semi -axis relationship	a (m)	b (m)	c (m)	Curvature at tip (1/ρ)*	$\sqrt{\frac{a}{\rho}}$ *
Spherical	a=b=c	0.01	0.01	0.01	100	1
Prolate A	a, b=c=a/2	0.01	0.005	0.005	400	2
Oblate A	a=b, c=a/2	0.01	0.01	0.005	250	1.58
Prolate B	a, b=c=a/4	0.01	0.0025	0.0025	1600	4
Scalene	a, b=a/2, c=a/4	0.01	0.005	0.0025	1000	3.16
Oblate B	a=b, c=a/4	0.01	0.01	0.0025	850	2.91

* Refer to Appendix B for calculation details

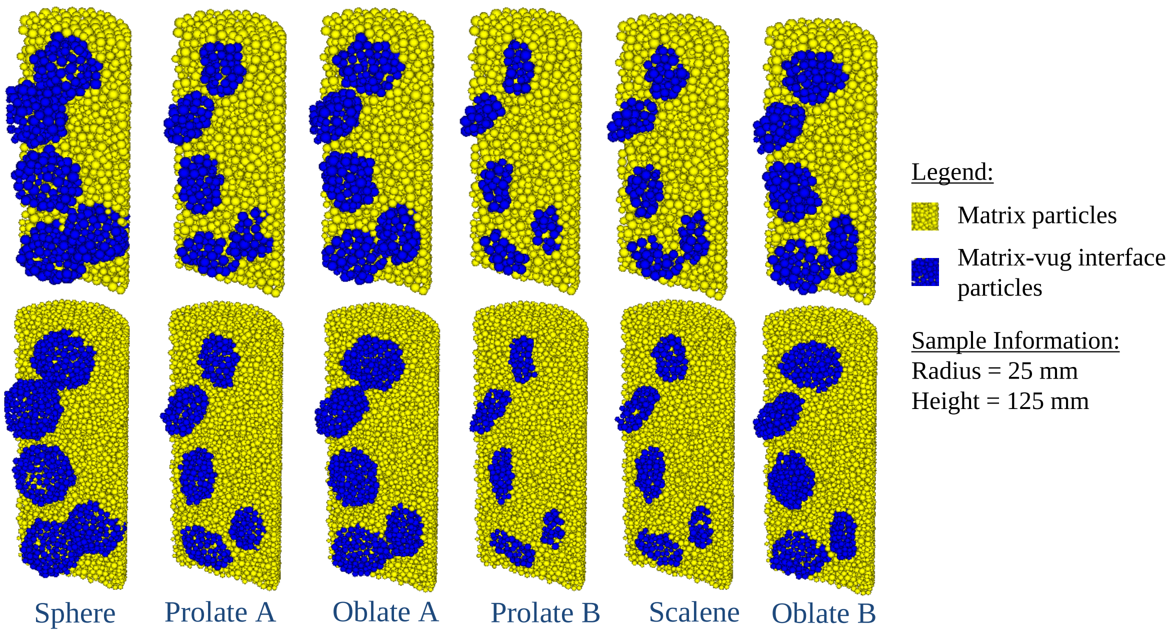


Figure 3.7: Vug networks samples with two resolutions. Vug location is maintained, but the axis length changes. Locations were random but limited to ensure non-overlapping voids

3.3.2.2. Ordered Array or Symmetric Vugs

A study of the effect of vug location on the geomechanical parameters was conducted by creating samples in which the vugs were located in an ordered or symmetric pattern. Twenty different configurations of "ordered" vugs were applied with the materials M1.1 through M4.2. These were further modified in to either spherical or ellipsoid vugs, for a total of forty vug configurations, which were symmetrical with respect to the origin. The vugs were placed in columns with the centers along the Z=0 or X=0 planes. Some examples of the configurations used are shown in Figure 3.8 for the spherical vugs. The shape of the ellipsoids had semi-axes ratio of 2:1 and a dip angle of 45º. The arrays created had equal spacing between vugs which limited the number of vugs placed within the sample. This limitation reduced the range of vug volume ratio to a maximum of 0.04.

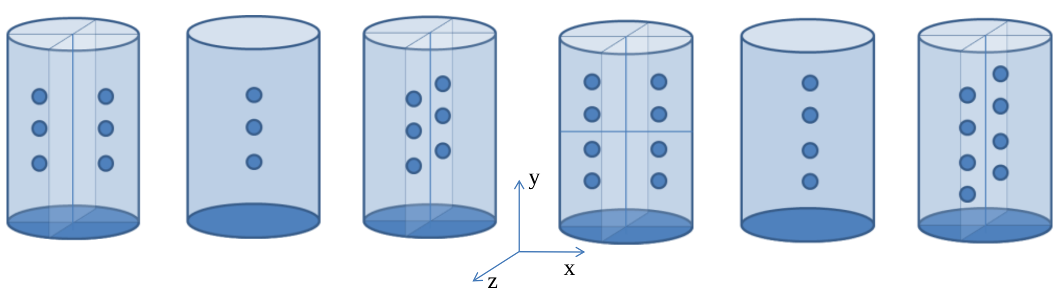


Figure 3.8: Representation of the regularly ordered vuggy samples

3.4. Static Test Results

3.4.1. Uniaxial Compression Strength Test

Intact and vuggy samples were subjected to the virtual UCS tests in PFC3D. The results for the intact tests were presented previously in Table 3.3; the UCS and E_s for the vuggy specimen were normalized to the intact modulus and strength of the same material. These normalized strength and moduli were plotted against the vug volume ratio for each sample. Figure 3.9 and Figure 3.10 present the result for regularly ordered and random vug sample set, respectively. The results for the orderly or symmetric vugs are quite spread and the correlation observed is poor, but this could be related to the smaller range of vug volume sampled. The results for the randomized vugs present a clearer trend line. This difference is partly due to sampling a wider range of vug volume ratios; however there does appear to be an effect of the location of the vugs when they are ordered in a regular pattern.

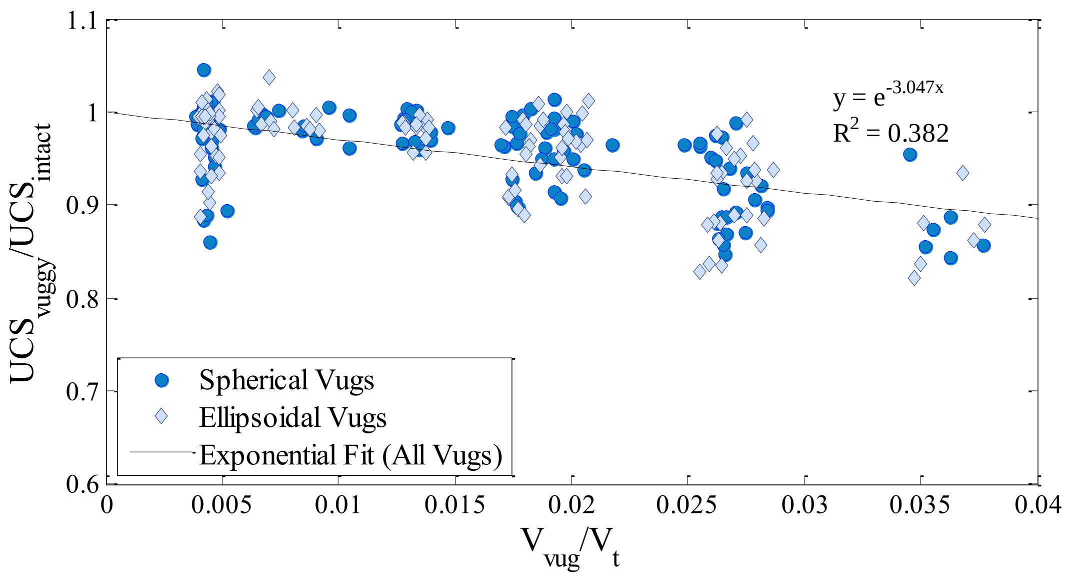


Figure 3.9: Normalized UCS vs. vug volume ratio for samples with ordered vugs; graph in bottom right corner is the same plot with the scale set to match Figure 3.10 for comparison

Figure 3.10 shows the UCS decreases exponentially with increasing vug volume. The exponential curve fitted to the data of the materials had a multiplying factor to the vug volume ratio that varied from -4.92 to – 5.54, with the intercept set to 1 (when vug volume is zero the UCS value for vug and intact should be equal). This is consistent with several relations presented by Chang et al. (2006) and Santos and Ferreira (2010), in which the UCS decreases exponentially with increasing porosity. In this case the vug volume ratio could be considered a surrogate for Vuggy porosity, and the decrease in strength of the Vuggy samples matches the trend that has been found by previous authors.

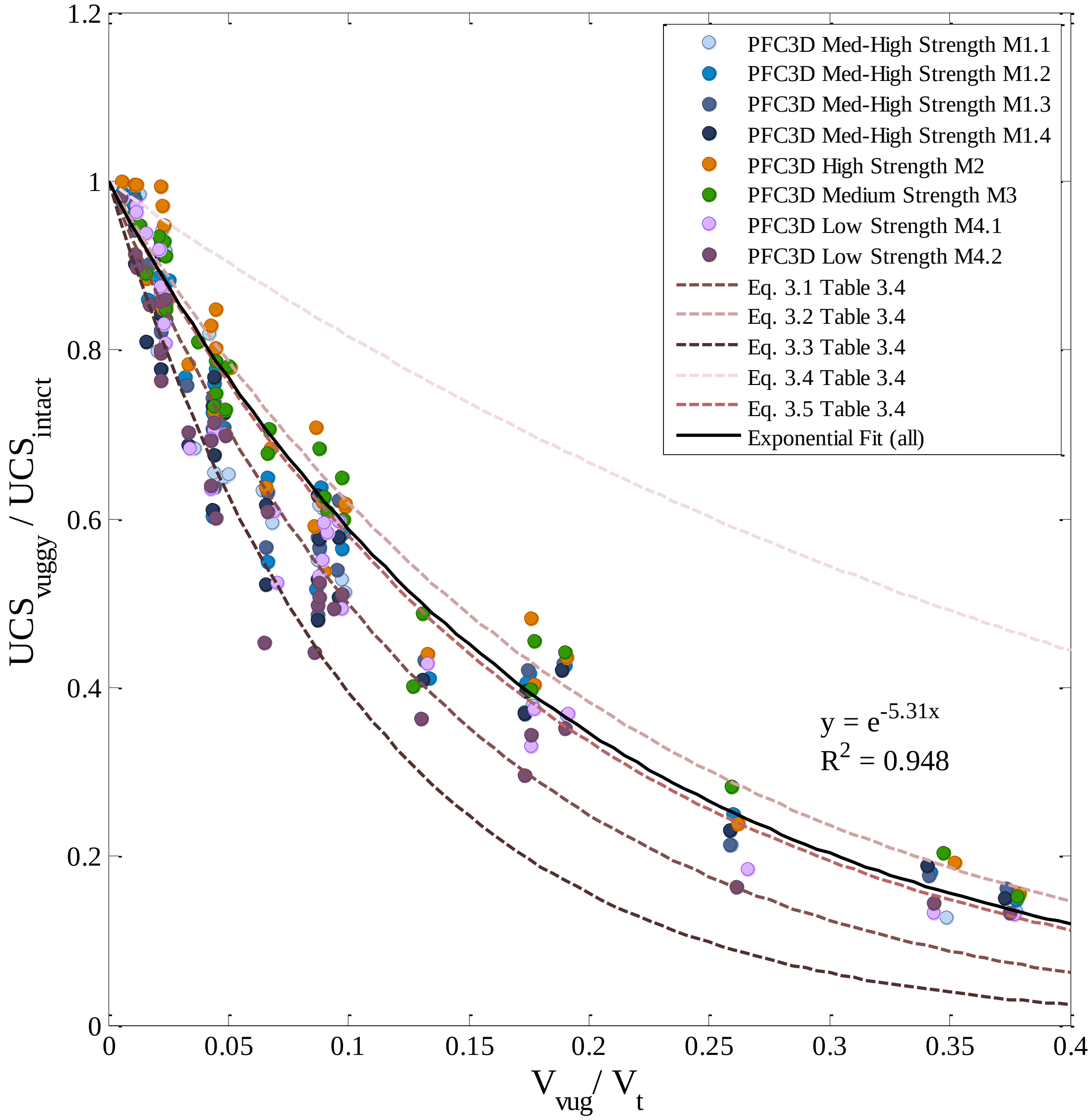


Figure 3.10: Normalized UCS vs. vug volume ratio for samples with randomized vugs

A selection of the equations provided for predicting carbonate behaviour are presented in Table 3.4. The empirical correlations were normalized to an intercept of 1 to compare with the PFC3D results, both the correlation and the PFC3D data were plotted together as seen in Figure 3.10 and show very similar trends. Moreover, a large amount of the data points fall within the boundaries of the curves. This presents evidence that PFC3D is capable of modelling the decrease in strength due to void space increase, which has been observed in real carbonate samples.

Table 3.4: Empirical relations for predicting UCS and E based on porosity measurements

Expression	Comments	Reference	Normalized Form	Eq. no.
$UCS_{(MPa)} = 143.8e^{-6.95\varphi}$	Developed in Middle East $0.05<\varphi<0.2$ 30<UCS<150MPa	Chang et al. (2006)	$e^{-6.95\varphi}$	3.1
$UCS_{(MPa)} = 135.9e^{-4.8\varphi}$	$0.05<\varphi<0.2$ 10<UCS<150MPa	Chang et al. (2006)	$e^{-4.8\varphi}$	3.2
$UCS_{(MPa)} = 174.8e^{-9.3\varphi}$	Carbonate Rocks	Farquhar et al. (1994)	$e^{-9.3\varphi}$	3.3
$UCS_{(MPa)} = 62.567e^{-2.03\varphi}$	Dolomite, Ghawar field	Ameen et al. (2009)	$e^{-2.03\varphi}$	3.4
$UCS_{(MPa)} = 89.432e^{-5.46\varphi}$	Limestone, Ghawar field	Ameen et al. (2009)	$e^{-5.46\varphi}$	3.5
$Es_{(GPa)} = 69.05e^{-6.0\varphi}$	Static modulus Carbonate rocks	Farquhar et al. (1994)	$e^{-6.0\varphi}$	3.6
$E_{(GPa)} = 92.612e^{-4.75\varphi}$	Dolomite, Ghawar field	Ameen et al. (2009)	$e^{-4.75\varphi}$	3.7
$E_{(GPa)} = 86.094e^{-5.34\varphi}$	Limestone, Ghawar field	Ameen et al. (2009)	$e^{-5.34\varphi}$	3.8
$Ed_{(GPa)} = 66.98e^{-4.2\varphi}$	Dynamic modulus Carbonate rocks	Farquhar et al. (1994)	$e^{-4.2\varphi}$	3.9

The current experiment provides further confirmation that the equations in Table 3.4 for the prediction of strength based on porosity are relevant and capture the strength decrease correctly. The results from numerical simulations indicate that the exponential decrease behaviour is independent of initial intact strength; Equation 3.10 was developed using all data points from the PFC3D simulations (R^2 = 0.95). This correlation can be adapted to various carbonate materials via multiplying the exponential function by an average intact UCS value. These results can be applied to the field of reservoir geomechanics as an approach to determine strength of carbonate formations by combining UCS test of a few intact samples with porosity wireline results to provide a strength representation of the reservoir.

$$\frac{UCS_{vuggy}}{UCS_{intact}} = e^{-5.31\frac{V_{vug}}{V_{intact}}} \qquad \textbf{Eq. 3.10}$$

A similar normalized approach was taken to study the effect of void volume on the E_s. The normalized moduli are plotted against the vug volume ratio in Figure 3.11 and Figure 3.12 for the ordered and random vug sample sets, respectively. The regularly ordered vugs show a decrease in stiffness with increasing vug volume; the relation appears linear but can also be explained by an exponential function given the limited range of vug volumes used for the ordered configuration. The vug volume evaluated for regularly ordered vugs was below 0.04, due to geometric constrains involved in creating an orderly array with equal spacing between vugs. Unlike the results obtained for the UCS of the ordered vug arrangements, the modulus does not appear to be affected by the placement of the vugs; the decrease observed for both random and ordered vugs is similar.

An exponential decrease is observed again for the numerical simulation of the randomized vug samples. Equations 3.6 to 3.8 are plotted in Figure 3.12 for comparison. These equations have been developed for real carbonate samples and have been normalized to present a comparison to the numerical simulations. The data points follow similar trends and an exponential relationship; however, the results tend to plot above the correlations from the literature. Once more, the behaviour appears to be independent of material strength and E_s/UCS ratio; thus the expression developed with all the PFC3D results is presented in equation 3.11 (R^2 = 0.97). It is noteworthy that the exponent was similar for both the random and ordered vug arrangements, leading to the conclusion that vug volume is a strong indicator of the stiffness reduction in vuggy samples, and that the relative location or arrangement of vugs does not contribute significantly to a change in stiffness.

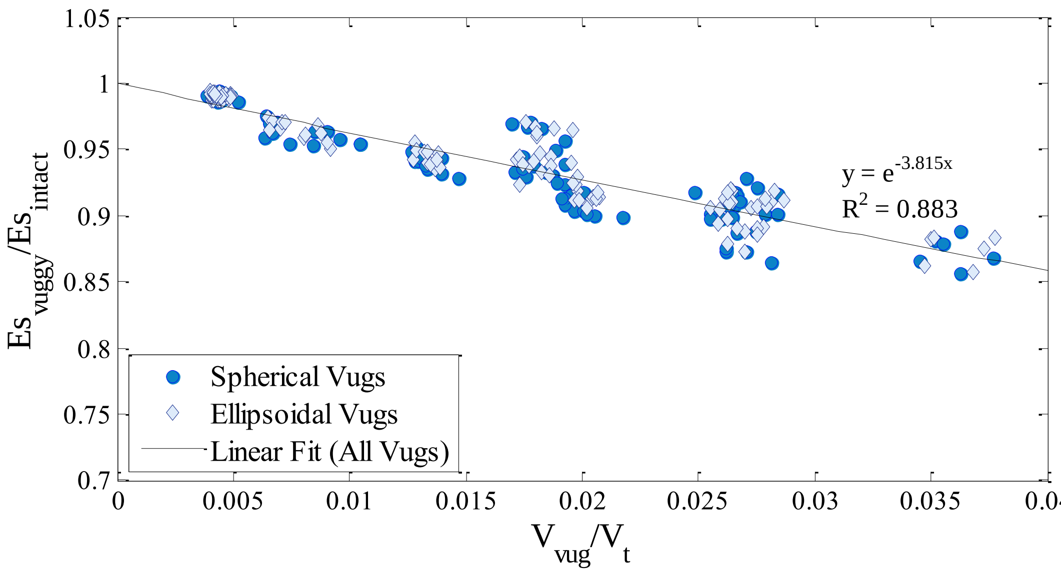


Figure 3.11: Normalized stiffness vs vug volume ratio for samples with ordered vugs

$$\frac{Es_{vuggy}}{Es_{intact}} = e^{-3.82\frac{V_{vug}}{V_{intact}}} \quad \textbf{Eq. 3.11}$$

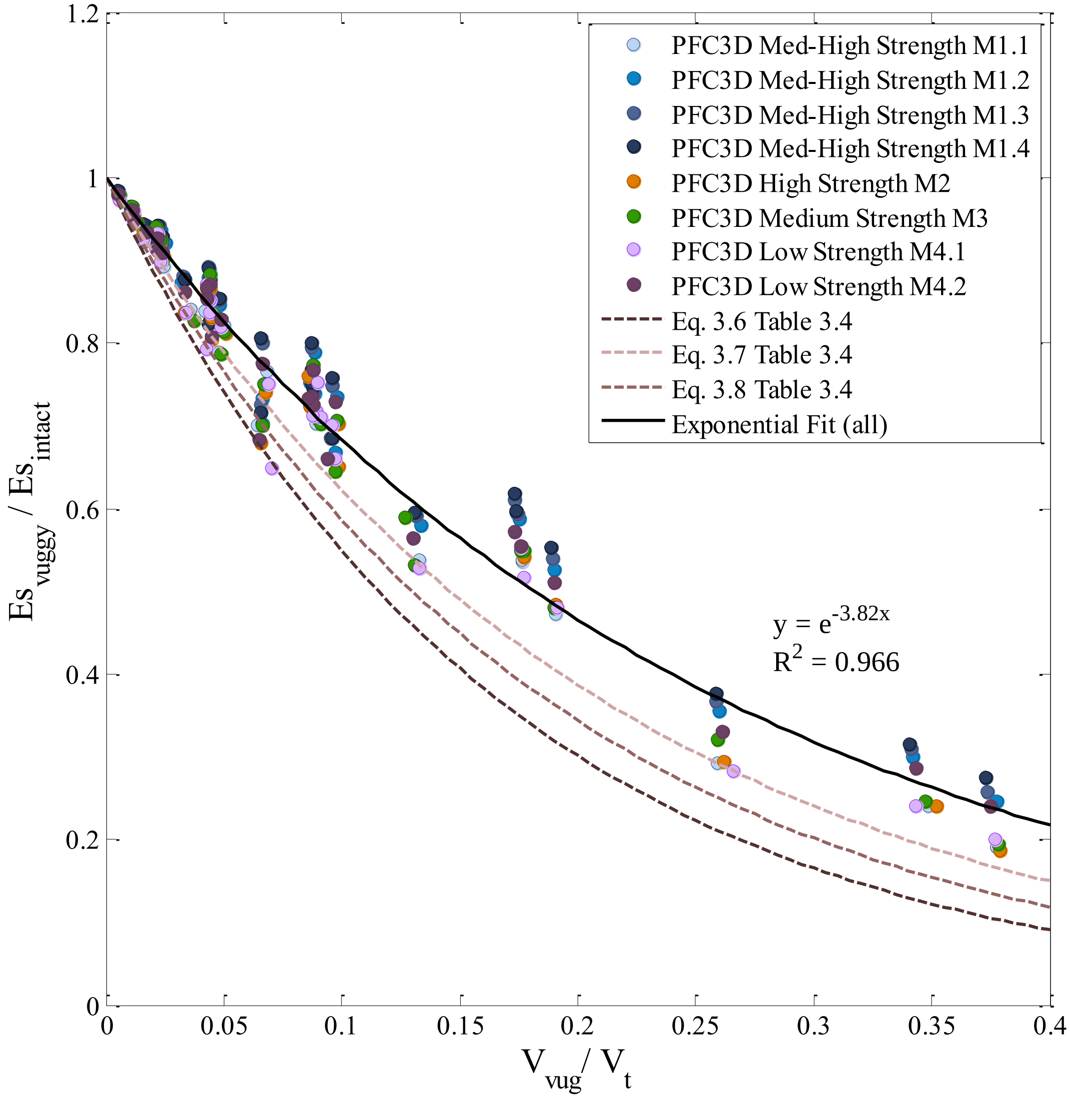


Figure 3.12: Normalized stiffness vs vug volume ratio for samples with randomly placed vugs

The stress versus strain curves for the simulated samples are presented in Figure 3.13 for material M1.1 and M1.2. The plot presents the difference in behavior between intact samples and increasing vuggy porosity for spherically shaped vugs. Similar behaviours were observed in the results for vuggy samples with other shapes, and these are included in the Appendix A. Figure 3.14 presents the stress strain curve for materials M4.1 and M4.2 for samples with oblate shaped vugs. These materials are considerably softer and weaker but the change in behavior is similar. The post behaviour of PFC3D does not necessarily match the real sample behaviour and its representation requires separate calibration; however, the focus of this study is on the change prompted by the vugs, which can be observed to change the behaviour of the sample from brittle to ductile.

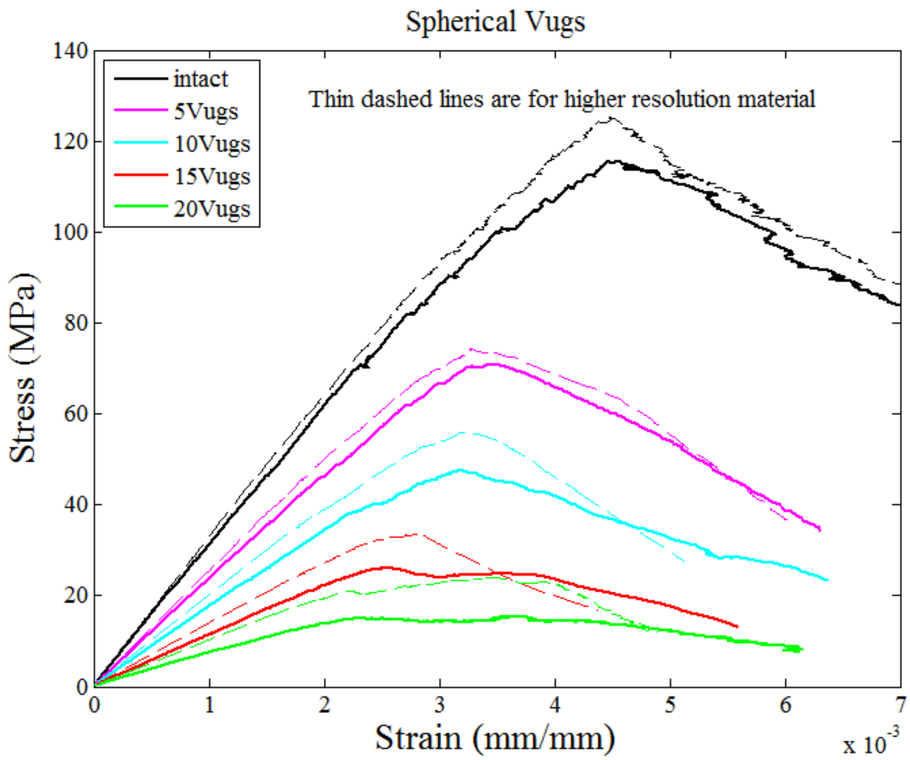


Figure 3.13: Stress-Strain curves for UCS test of vuggy samples for Materials M1.1 and M1.2, vug shape is spherical

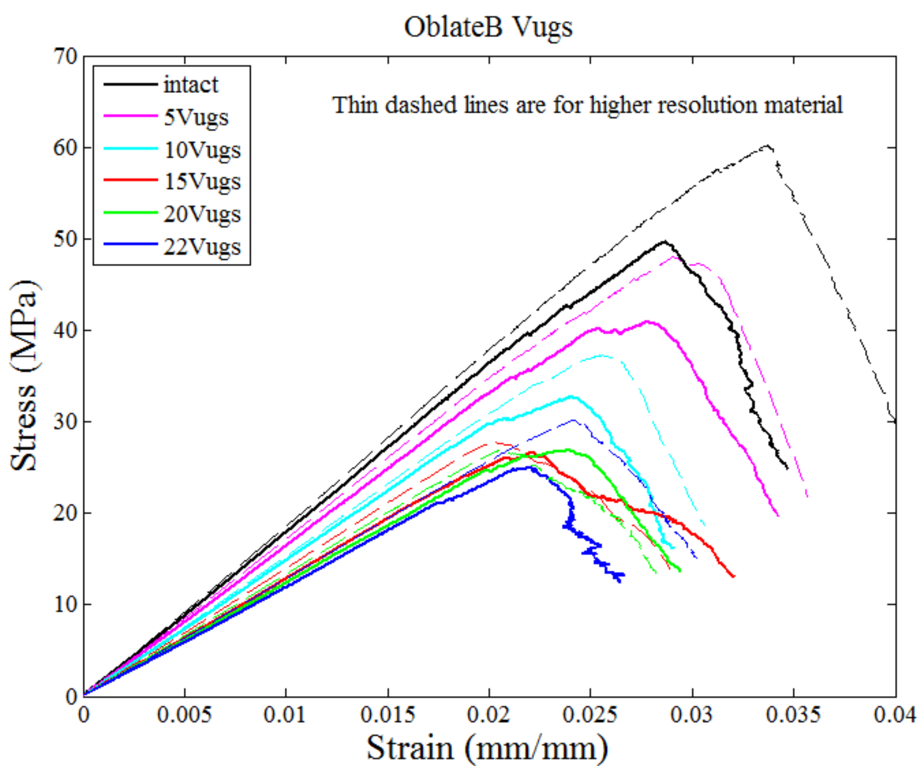


Figure 3.14: Stress-Strain curves for UCS test of vuggy samples for Materials M4.1 and M4.2, vug shape is oblate ellipsoid

3.4.2. Tension Test

The samples generated for the above simulations were also subjected to the direct tensile test environment available in PFC3D. This test in PFC3D does not utilize a dog bone shape sample. The test is carried out by assigning "grips" to the particles at the top and bottom edges of the sample and applying an outward displacement to these "grips" to induce tension. The results of the tensile tests for vuggy samples were normalized to the intact (see Table 3.5) to provide a similar analysis as conducted with the UCS and E_s.

Table 3.5: Direct tensile test results for intact materials

Material	Tensile (MPa)	UCS/σ_t Ratio
M1.1	49.3	2.41
M1.2	50.1	2.54
M2	78.0	2.50
M3	41.2	1.79
M4.1	13.9	3.62
M4.2	14.5	4.17

The normalized tensile results are presented in Figure 3.15 for the ordered vug samples and Figure 3.16 for the random vug samples. Both figures show an exponentially decreasing tensile strength with increasing vug volume ratio. The ordered vug samples also show a decrease, with a much larger exponent (faster rate). The discrepancy can be due to the lower vug volumes used for the ordered vug simulations. In comparison with the UCS results the ordered vug samples do present a clearer trend, even though the correlation coefficient is not high.

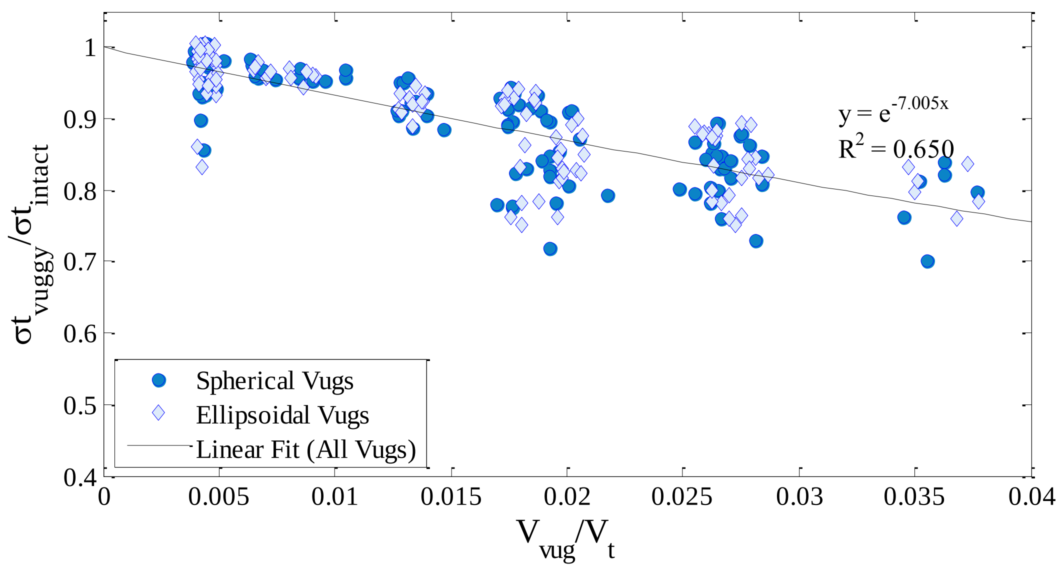


Figure 3.15: Direct tensile test results for ordered vug samples

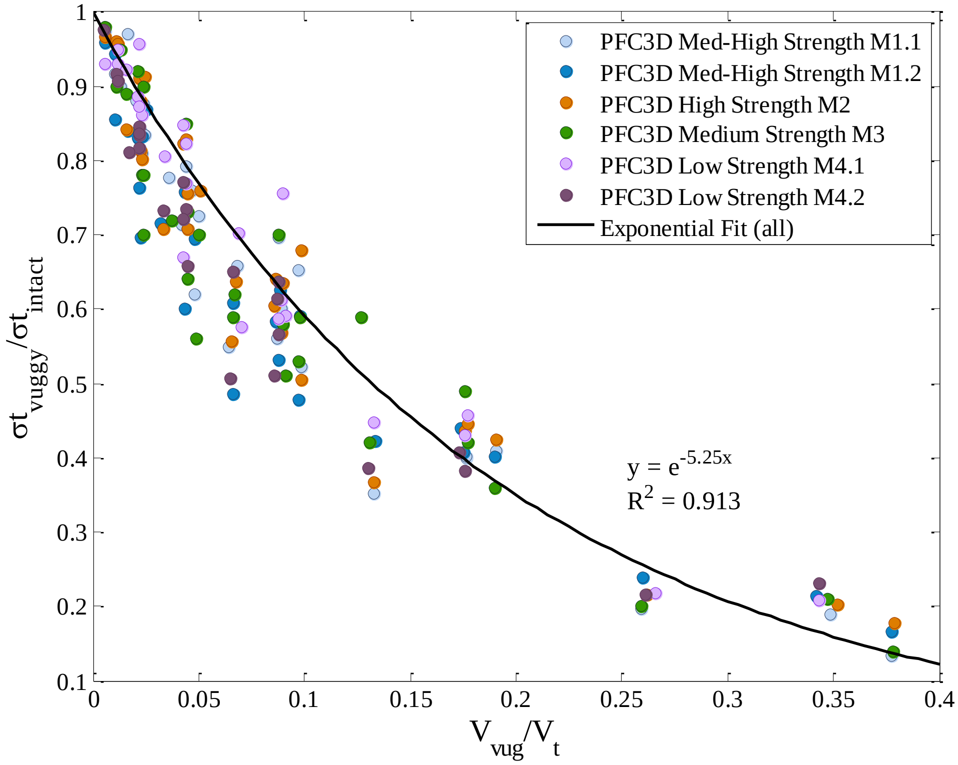


Figure 3.16: Direct tensile test results for random vuggy samples

The results from both UCS and direct tension test were combined to compare the UCS to tensile strength ratio (UCS/σ_t) of the samples (refer to Figure 3.17 and Figure 3.18). This comparison revealed the largest discrepancy between the regularly ordered and randomly placed vug samples. The random sample results were scattered but with a trend of decreasing UCS/σ_t with increasing vug volume ratio. Conversely, the symmetric samples show a clear trend towards increasing UCS/σ_t with increasing vug ratio. The reasons for this divergence are discussed in the analysis section below.

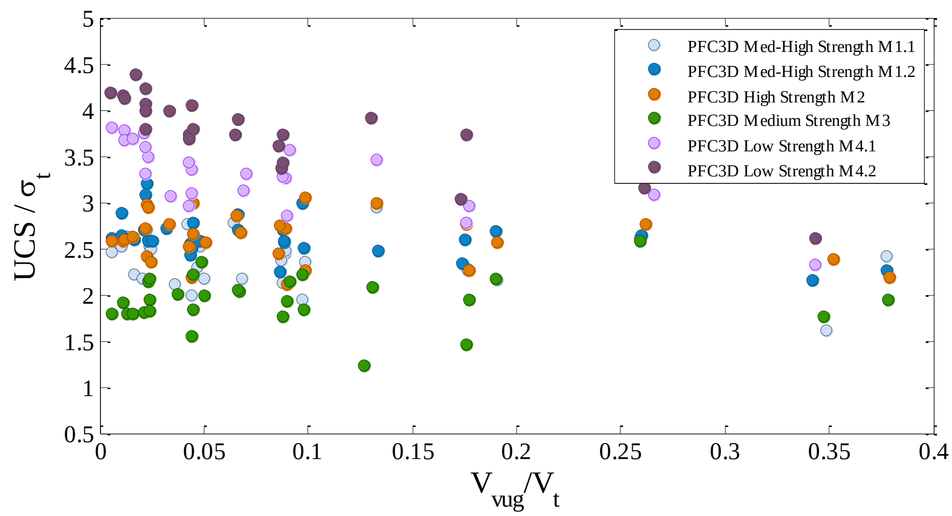


Figure 3.17: UCS to tensile ratio by material for random vuggy samples

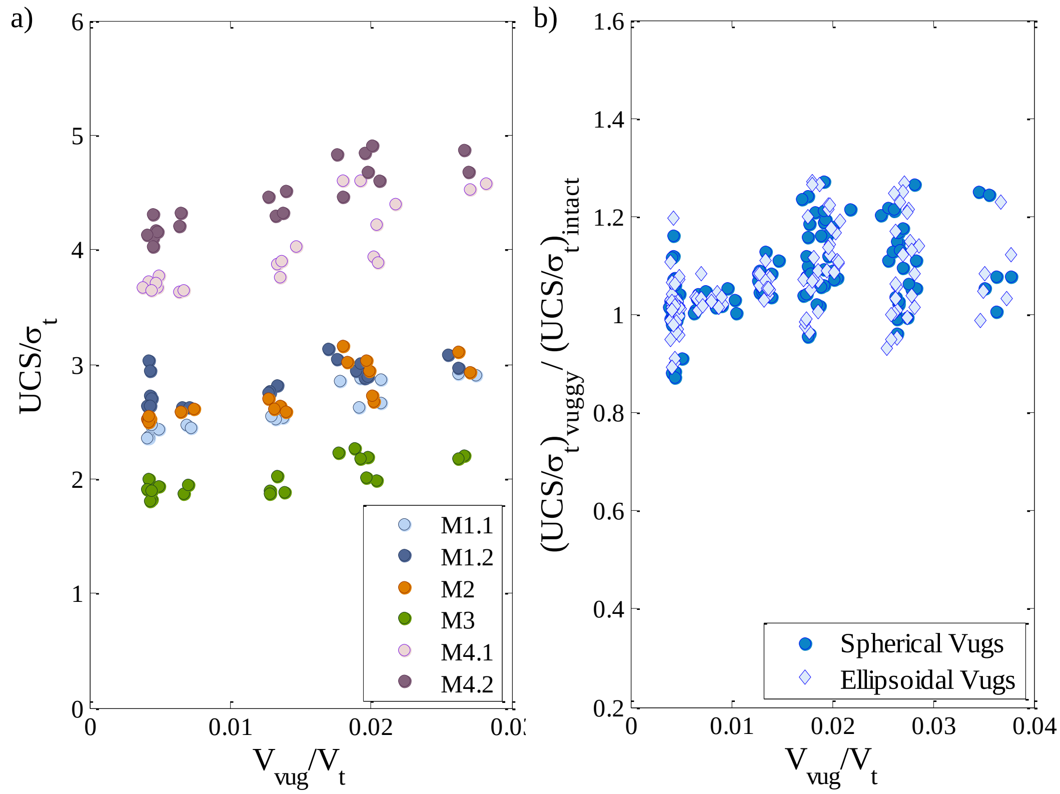


Figure 3.18: Orderly vugs, UCS/σ_t increase with increasing porosity. a) UCS/σ_t by material, b) normalized UCS/σ_t separated by vug shape

3.5. Analysis of Static Results

The prediction of the strength of carbonate material has been hindered by the wide variability of responses and the high heterogeneity present in these materials (Chang et al. 2006; Santos and Ferreira 2010). Multiple correlations have been developed which relate porosity, acoustic wave velocity, dynamic modulus, or bulk density to the strength of limestone and dolomites. The purpose of these empirical equations is to be able to minimize laboratory measurements, and extend their results over larger areas through physical properties from wellbore logs.

Santos and Ferreira (2010) tested the prediction capabilities of several correlations with wellbore data and samples from Brazilian off-shore reservoirs. They found that although the CPM equation (from Schlumberger's MECHPRO^{TM}) provided the best results, it was closely followed by porosity dependent equations such as Eq. 1 and 3 presented in Table 3.4. The authors noted that the CPM equation was complex and dependent on bulk modulus and acoustic velocities; while the other equations predicted strength based solely on porosity measurements (Santos and Ferreira, 2010). Furthermore, Chang et al. (2006) found that exponential relations to porosity provided good predictions from carbonate strength for porosities greater than 0.1. The results from the current simulations present further evidence that exponential functions between porosity and UCS can provide reliable results for carbonates with vuggy porosity.

On the numerical side, similar simulations conducted by Xie et al. (2012) using PFC2D samples with random spherical inclusions were found to have an exponential decrease in UCS with increasing porosity. Moreover, the stiffness of the sample also decreased with increasing porosity. The authors concluded that PFC2D was suitable to conduct numerical testing of uniaxial compression and could be an alternative to laboratory testing.

Figure 3.10 presents some scatter in the data, which could be a possible effect of different vug shapes. The vug shape and location may become more important at low porosities, which could explain the scatter in Figures 3.9 and 3.10 (below vuggy porosity of 0.07). Similar effects were found in micro-compression testing of brecciated dolostones and limestones by Martínez-Martínez et al. (2001). Their experiments tested small samples from multi-textural rocks, to determine the effect of texture and porosity. They noted that the pore geometry had an important effect on the stresses at low porosities. Martínez-Martínez et al. (2011) indicated that at the porosities below 7%, samples with larger three-dimensional pores had a higher strength than samples in which the pore geometry was two-dimensional (plate-like or micro-cracks). This was postulated to be caused by lower stress concentrations near the larger pores due to larger radius of curvature. Further simulations exploring the effect of vug shape at low porosity should be conducted to determine if there is a possible correlation between strength and radius of curvature of the inclusions. To properly evaluate the shape effect, multiple realizations with different seed numbers should be created.

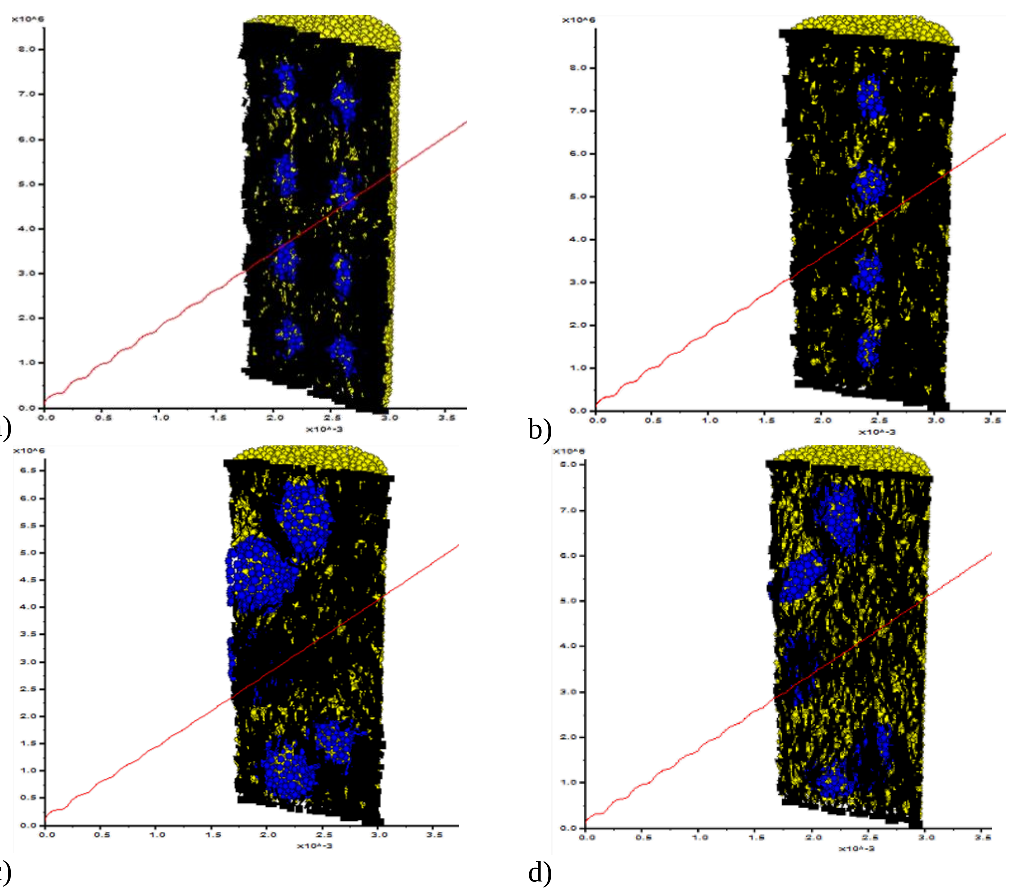
The effect of porosity on E_s was also found to be exponential, similar to the empirical relations found in literature. However, it has been noted in several studies that correlations for E are much less reliable than for UCS (Palchik 2011; Santos and Ferreira 2010). Martínez-Martínez et al. (2010) found little effect of porosity on E_s, although their tests were mostly done on low porous carbonates. Their experiments show that the stiffness was more dependent on the grain size and the amount of contacts for each grain. Santos and Ferreira (2010) establish that the empirical correlations for E_s yielded less accurate estimates for their experimental data, especially for low stiffness materials. The expressions developed in the PFC3D simulations, when compared to the existing correlation, under-estimate the reduction in stiffness due to increasing void volume. Further investigation is required into the validity of stiffness-porosity empirical equations in order to expand their use for up-scaling geomechanical properties.

The direct tensile test results show high values for the tensile strength of the intact materials. This is one of the known limitations of PFC3D BPM, in which calibration for the UCS results in high values for the tensile strength (Potyondy and Cundall 2004; Cho et al. 2007). As shown in Table 3.5 the UCS/σ_t is too low for the ratio found in natural rocks. It is not the purpose of the present research to address this limitation of PFC3D as this has been researched in other studies (Schöpfer et al. 2009; Cho 2008; Scholtés and Donzé 2013); thus, the results are normalized to provide a comparison between vuggy and intact samples.

The tensile strength reduces with increasing vug volume ratio (refer to Figure 3.15 and Figure 3.16), which is consistent with a general decrease in strength with increase in porosity (Schöpfer et al. 2009). However, UCS tends to decrease more rapidly than tensile strength (compare the exponents in Figure 3.10 and Figure 3.16) resulting in an overall decrease of the UCS/σ_t ratio in the case of the random vug samples. This is consistent with the simulation results obtained by Schöpfer et al. (2009).

The trend in the UCS/σ_t ratio for the symmetric or ordered vugs goes against what was found for the random vug case, and increases with increasing vuggy porosity. This is also contrary to what has been found in other studies, in which UCS/σ_t ratio decreases with increasing porosity (Schöpfer et al. 2009). This is most likely due to the low decrease in UCS strength observed in the ordered samples (refer to Figure 3.9). A possible reason for the higher strength presented in regularly ordered vug arrangement could be an arching effect within the sample. Figure 3.19 below shows the transfer of force within the samples components for two symmetric and random cases. In the symmetric cases the load is taken by the side of the sample in an even fashion, decreasing stress from the centre of the sample (around the vug location) and relieving load from the weakest section. Conversely in the random case, force chains must circumvent the vugs locations in a more tortuous path and do not develop arching effect in the middle of the sample.

The arching effect gives larger peak strength for the symmetric vug arrangements than a sample with an equivalent vuggy porosity but with randomly ordered - the higher UCS values increase the overall UCS/σ_t ratio. It can be concluded, based on the results presented above for the two different vug arrangements that stiffness is not affected by the relative location of the vugs but UCS strength is. Regularly ordered vugs are not commonly found in real vuggy carbonates and this effect may rarely be encountered during laboratory tests.



a) b) c) d)

Figure 3.19: Load distribution within a center plane of vuggy samples (black lines represent force chains); a) and b) are ordered arrays of vug placement, c) and d) are random vug placement.

3.5.1. On the Use of Empirical Correlations

The comparison between empirical correlations and the simulation results show a similarity in trend and good correspondence at low porosity values. For UCS, the data points plot within the boundaries of the normalized Equations 3.1 through 3.3 from Table 3.4. Equation 3.4 under-predicts the reduction in strength due to porosity, and would only be valid at vuggy porosities below 3%. The results illustrate that care must be taken when using general correlation equations, which were developed for a particular sample set, to predict UCS strength for a new carbonate formation. The correlations should not be used outside of the boundaries from which they were originally developed. For example Chang et al. (2006) equations should be limited to porosities between 0.05 and 0.2, for carbonates with strengths closer to 140MPa. A more useful technique would be to use the exponents from the correlation and an intact UCS strength tested from the material studied. The current simulations suggest that the relative effects of voids is similar regardless the original strength of the intact material. The combination of the intact values with the measure of porosity can provide a more accurate representation of the targeted carbonate material.

The results from the tensile test suggest that strength in both the tensile and unconfined compression stress paths is reduced with porosity in a similar exponential trend. The simulation results are harder to compare with laboratory samples, given that correlations between tensile strength and porosity are not common in literature. Rajabzadeh et al. (2012) showed a relation between Brazilian tensile test results and porosity; although they do not provide a direct correlation between both variables. Porosity has long been considered the best predictor of strength is sedimentary rocks (Faquhar et al. 1994). The correlations presented by Rajabzadeh et al. (2012) were a linear relationship of UCS with tensile strength (for sedimentary limestones and diagenetic dolomitic limestones) or with a tensile strength normalized by porosity (for sedimentary limestones). The author noted that including porosity in his correlations did not improve for the diagenetic dolomitic limestones tested (Rajabzadeh et al. 2012). This emphasizes that correlations produced from a restricted number of samples are limited and cannot be implemented as universal equations.

The correlations of porosity with E_s provided good agreement up to 10% vuggy porosity. The normalized correlation for stiffness (Equation 3.11) was found with both the random and symmetric vug samples, indicating that the location of the vugs does not affect the stiffness in PFC3D as was observed with the UCS tests. The use of modulus correlations presented in Table 3.4 should be used with caution, since they may over-predict the stiffness reduction at higher porosities.

The simulations of PFC3D provide confirmation on the relation between strength and porosity. Although there are some remaining limitations with the BPM that restrict good quantitative predictions, such as the tensile to UCS strength ratio, it still provides good qualitative estimates on rock behaviour (Potyondy and Cundall 2004). The previous simulations could be tested using clump particle model developed by Cho et al. (2007) to provide better correlations (in particular for tensile test environment) and expanding to confined strength parameters. Improvements in PFC3D are needed that allow for a single set of micro-properties calibrated to UCS that still provides good behaviour under different stress path.

3.5.2. Modelling Soft Rock with PFC 3D

PFC has been used to model soft rock materials with low UCS and E_s, like Chalk or Tuff as discussed in Chapter 2. Materials 4.1 and 4.2 are examples of a PFC3D calibration to low UCS value and very low stiffness to simulate a soft rock. However, on analyzing the stress vs strain curve it is evident that no initial non-linearity was observed at the start of the test, even when the samples had a high number of vugs. This non-linearity is normally referred to as the crack closure stage (Martin and Chandler 1994); and tends to be more significant in soft rocks. Cho et al. (2007) proposed that this could be accomplished with PFC3D by adding pre-existing flaws, such as a random distribution of pores or cracks were used. As observed in Figure 3.13 and Figure 3.14, as well as other stress vs strain curves presented in Appendix A; a distribution of voids within the sample did not produce this non-linearity. These results are consistent with previous PFC simulations by Bechtel (2004) for soft rock, in which detail geometric representation of voids did not produce a crack closure stage (refer to Figure 2.6 in Chapter 2).

The persistence of the initial linearity may be caused by the lack of spaces within the matrix of the sample. This is a requirement during generation of the sample and equilibrium calculations. Attempts were made to use the standard sample generation procedure with higher initial porosity values or allowing the presence of floaters, but no significant improvements were observed. Samples with high initial porosity could not be generated with the traditional fish code for BPM samples. It is also not sufficient to impose cracks or eliminate bonds, as this was done Schöpfer et al. (2009). They obtained increasingly ductile samples with increasing crack density; however the initial stage of the stress-strain curve remained linear. It is possible that a non-linearity may be observed if a DFN of micro-cracks is superimposed on the sample using the sliding joint model, and including a distribution of apertures on each micro-crack. This solution is proposed for future study of soft rock where it is important to model the initial non-linear behaviour of the material.

3.6. Dynamic Testing of Vuggy Samples

3.6.1. P- and S- Wave analysis in the Field and Laboratory

Seismic methods are commonly used for the exploration of in situ rock masses (Ambraseys and Hendron 1968). This has become a standard in resource exploration, as well as monitoring of CO_2 storage sites. In this method, the elastic properties of the rock are determined based on the speed of propagation of waves through the rock medium. The rock mass is normally assumed to be isotropic; thus, only two wave velocities are necessary to describe the elastic behaviour of the material (Thomsen 1986). The first wave to arrive is P-wave (V_p), also called longitudinal or compressional wave, and the second wave is the shear or transverse wave (V_s). The relationships between the P- and S-wave with the elastic properties, for homogeneous and isotropic materials, are presented in the equations below:

$$V_p = L/t_p, \qquad \text{Eq. 3.12}$$

$$V_s = L/t_s, \qquad \text{Eq. 3.13}$$

$$\nu_d = \frac{\frac{1}{2}\left(^{Vp}/_{Vs}\right)^2 - 1}{\left(^{Vp}/_{Vs}\right)^2 - 1}, \qquad \text{Eq. 3.14}$$

$$E_d = Vp^2\rho \frac{(1+\nu_d)(1-2\nu_d)}{(1-\nu_d)},$$ **Eq. 3.15**

$$E_d = 2Vs^2\rho(1+\nu_d),$$ **Eq. 3.16**

where L is the length of the sample or the distance between the sensors, and t_p and t_s are the arrival times determined from visual inspection of the transmitted pulse data.

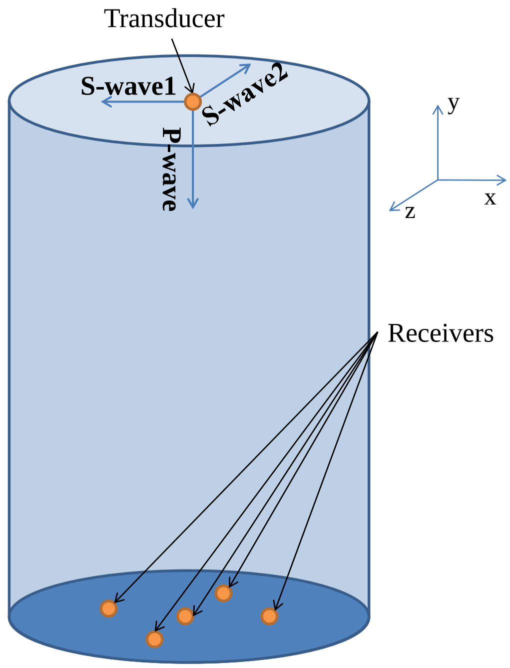
The P-waves travel in the direction of the wave and will arrive first to the detection areas. It is common to assume a dynamic Poisson's ratio (ν_d) and determine the dynamic modulus (E_d) from equations 3.15 above. The ν_d can also be directly calculated by using the S-wave velocity, but this is difficult to determine in the field. Shear waves travel slower than the P-waves; thus, it is challenging to determine the appearance of the S-waves because they arrive in conjunction with P-waves that have taken alternate (longer) travel paths (Ambraseys and Hendron 1968).

An equivalent laboratory test is conducted in rocks to determine elastic constants from P- and S-wave velocities by the use of ultrasonic pulses. The laboratory test also allows probing the elastic behaviour at different confining stresses. This is especially advantageous to predict the behaviour of the rock under changing stress that will remain within elastic range. As in the field, the first arrival of the P-waves is well defined and easy to determine, but the S-waves arrival is difficult to establish due to interference with compressional waves (ASTM D2845-08).

An additional advantage of the laboratory tests is that longitudinal and transverse waves can be induced and measured at different directions (perpendicular, parallel or at an angle to rock bedding). Thus the assumption of isotropic conditions is not necessary and the full elastic compliance matrix of the material can be determined. For a non-isotropic material nine velocity measurements are required; if some assumption of isotropy is made such as transverse isotropy, the number of velocity measurements is reduced to five (Thomsen 1968; Meléndez-Martínez and Schmitt 2013). Thomsen (1968) presented the relation between the velocities measured at different direction and the elastic compliance matrix. He also presented parameters that simplify the equation for transverse anisotropy, and these parameters also provide a measure of the anisotropy of the sample based on wave measurement.

3.6.2. PFC3D modelling of wave velocity measurements

PFC3D is intrinsically a dynamic system, which makes it ideal for the evaluation of dynamic properties and wave propagation (Holt et al. 2005). The PFC3D simulations for P- and S- waves were based on previous work done by Holt et al. (2005) to model wave propagation through glass beads. The present simulations were set up to run a displacement pulse through a ball at the top of the sample (the "transducer") and allowing the model to cycle until a velocity pulse was observed at the array of ball "receivers" placed at the bottom of the sample (refer to Figure 3.20 below). Both the "transducer" and "receiver" balls are within the sample, which reduces issues that may arise from coupling an external source/receiver (Holt, et al. 2005). The pulse was applied by using Equation 3.17 below based on an existing example in Itasca's PFC3D manual (Itasca 2008b).



$$V(t) = A/2(1 - \cos(2\pi f t)$$

Eq. 3.17

Figure 3.20: Representation of elastic wave velocity analogous test for PFC3D

The frequencies used varied between 100 kHz to 1 MHz, and the displacement pulse was applied in the longitudinal (y) and both transverse (x and z) directions. The default damping in PFC3D is local damping, which is not applicable to dynamic simulations (Itasca 2008a); thus, this was eliminated and replaced with a small amount of viscous damping (0.1 on shear and normal directions) to smooth out the velocity curves in the receivers. The viscous damping value was arrived at through trial and error, by assessing changes in arrival times and response curves from samples with 0, 0.05, 0.1, and 0.2 damping. The value chosen did not affect wave arrival time when compared to the no-damping case, but reduced the noise in the response curves.

The arrival times were based on the response curve from the center receiver; the data from the other receivers was also plotted to aid in the identification of the wave arrival as well as determining the scatter due to the presence of the vugs. The arrival of the P-wave was selected as the first trough observed in the velocity response. The arrival of the shear waves was more difficult to determine; the first peak or slight increase in velocity would always arrive at a time approximately equal to the P-wave arrival. For many samples one to two smaller peaks were observed before the first largely negative trough arrived at the receivers. The S-wave was assumed to arrive at this larger negative trough. This assumption was based on the fact the pulse was applied in the negative direction as well as observations of the progression of the wave through the sample during the simulation.

An advantage presented by PFC3D is that displacements and velocities of the particles within the sample can be monitored throughout the simulation; essentially following the progress of the waves as it travels through the sample. An important observation was that the shear waves tended to have significant components travelling in the longitudinal direction, regardless of the fact that the impulse was applied purely in a transverse direction. Thus, the first velocity pulse that arrived at the receiver travelled through part of the sample as a P-wave and should not be identified as the shear wave arrival. Moreover, the first peak observed during the S-wave simulations coincided approximately with the P-wave arrival time. Although this phenomenon has been observed in real rock (e.g. difficulty in interpreting shear wave laboratory tests) it is potentially amplified in PFC3D due to the formulation of the model. Figure 3.21 presents a graphic with a possible explanation for the shift in the direction of the wave.

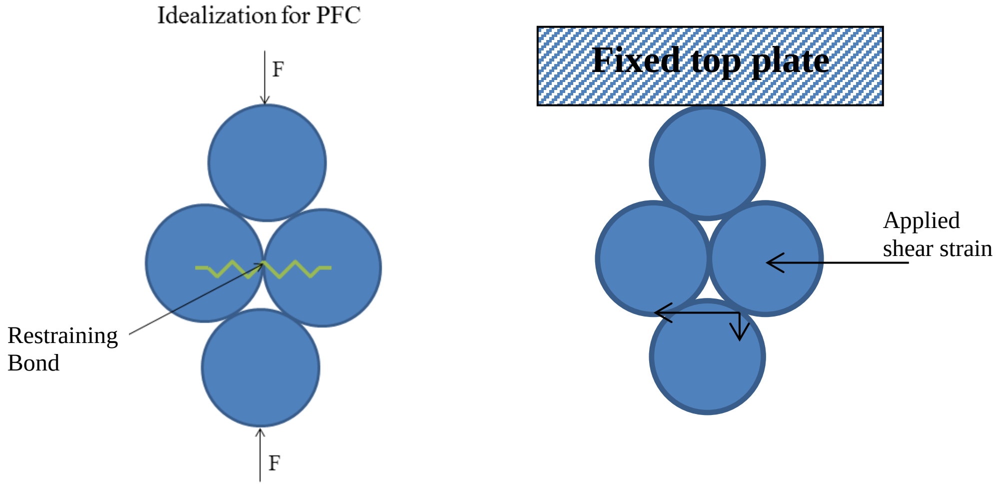


Figure 3.21: Representation of transfer of displacement pulse between the balls in PFC3D, pure transverse displacement may produce displacements in the longitudinal direction

The wave pulse in PFC3D is applied to a ball and can only be transferred to the surrounding particles through the contact plane. These contacts are in different directions and create two components during transfer; thus, a purely transverse velocity will transfer to other balls having both transverse and longitudinal components (see Figure 3.21). This is repeated several times as the wave travels through the sample, creating several P-wave impulses which will travel faster than and arrive prior to the main S-wave. This was also confirmed by plotting the displacement vectors for each ball during some dynamic simulations. Throughout S-wave simulations some vectors shifted and were perpendicular to the initial direction of the wave.

The longitudinal displacements of the center receiver peaked simultaneously with the initial peaks in the transverse measurements, indicating they were likely P-wave precursors. Conversely, the first large negative trough coincided with a larger accumulation of displacement vectors at the bottom of the sample, most in the transverse direction. Thus, it was concluded that the first and potentially second peaks observed in the response were not indicative of shear wave transmission, and the first large negative through was taken as the S-wave arrival time.

3.7. Dynamic Results

The first simulations were carried out in the intact materials to determine V_p, V_{s1}(x direction), and V_{s2} (z direction). The ν_d and E_d were calculated based on Equations 3.14 and 3.15, assuming the material was isotropic and using an average of V_{s1} and V_{s2}. A quick check to confirm the isotropic assumption was to assess whether V_{s1} and V_{s2} were approximately equal. The shear wave velocities were either equal or close to equal for intact materials and samples with very low vuggy porosity. However, the discrepancies between the velocities augmented at higher porosity, indicating the material was increasingly anisotropic. Figures 3.22, 3.23, and 3.24 show the results for the intact material simulations in V_p, V_{s1} and V_{s2} directions, respectively. These plots are for a frequency of 564 kHz, results for different frequencies are presented in Appendix A. Table 3.6 summarizes the results obtained for the intact materials.

Table 3.6: Elastic wave velocity tests results for frequency 564kHz, compared with elastic parameters from static test

Material	V_p (m/s)	V_s avg (m/s)	Dynamic Poisson's Ratio, υ_d	Dynamic Modulus, E_d (GPa)	Static Modulus*, E_s (GPa)	Static Poisson's Ratio*, υ_s
M1.1	3400	1620	0.353	18.9	33.3	0.17
M1.2	3510	2160	0.197	29.6	33.7	0.17
M1.3	3540	2150	0.208	29.7	34.5	0.16
M1.4	3610	2190	0.209	30.8	35.0	0.17
M2	4940	3062	0.188	59.0	66.3	0.18
M3	3720	1770	0.353	22.5	38.8	0.15
M4.1	860	824	0.206	1.75	1.8	0.24
M4.2	895	826	0.236	1.81	1.9	0.25

* Based on virtual UCS tests

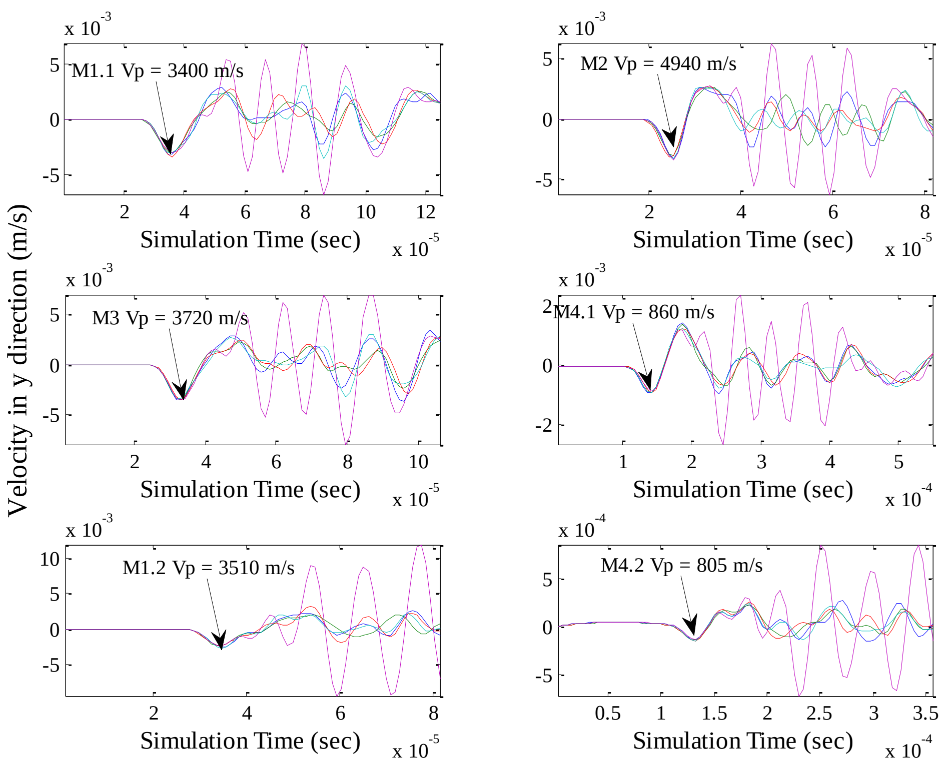


Figure 3.22: Intact longitudinal wave pulse, frequency = 564 kHz

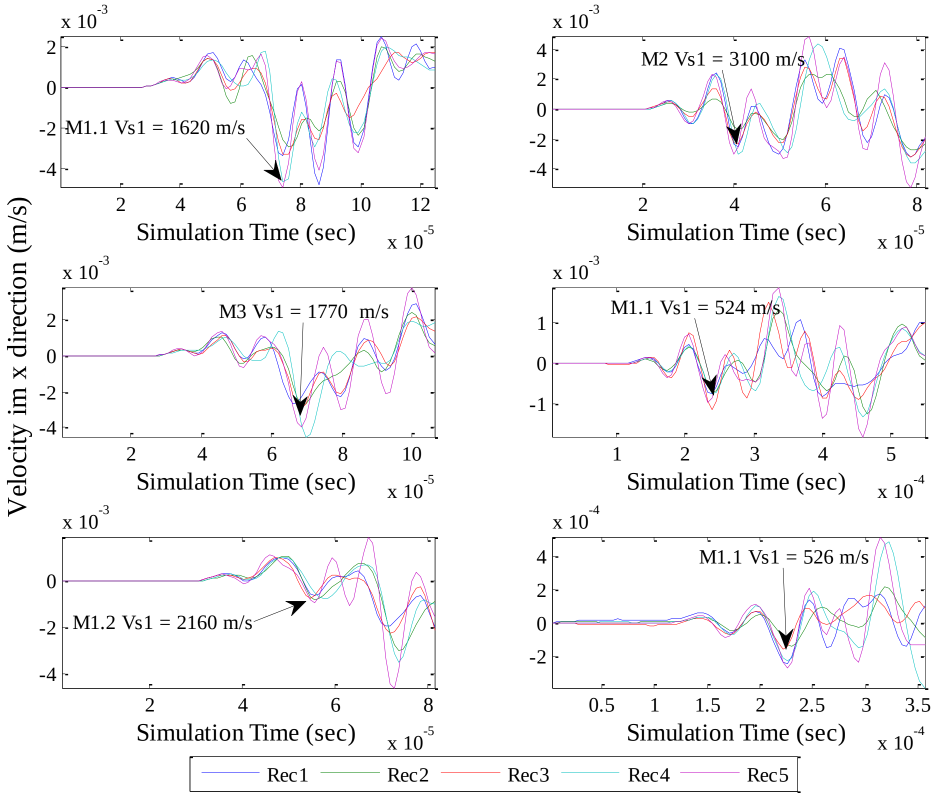


Figure 3.23: Intact transverse (x-direction) wave pulse, frequency = 564 kHz

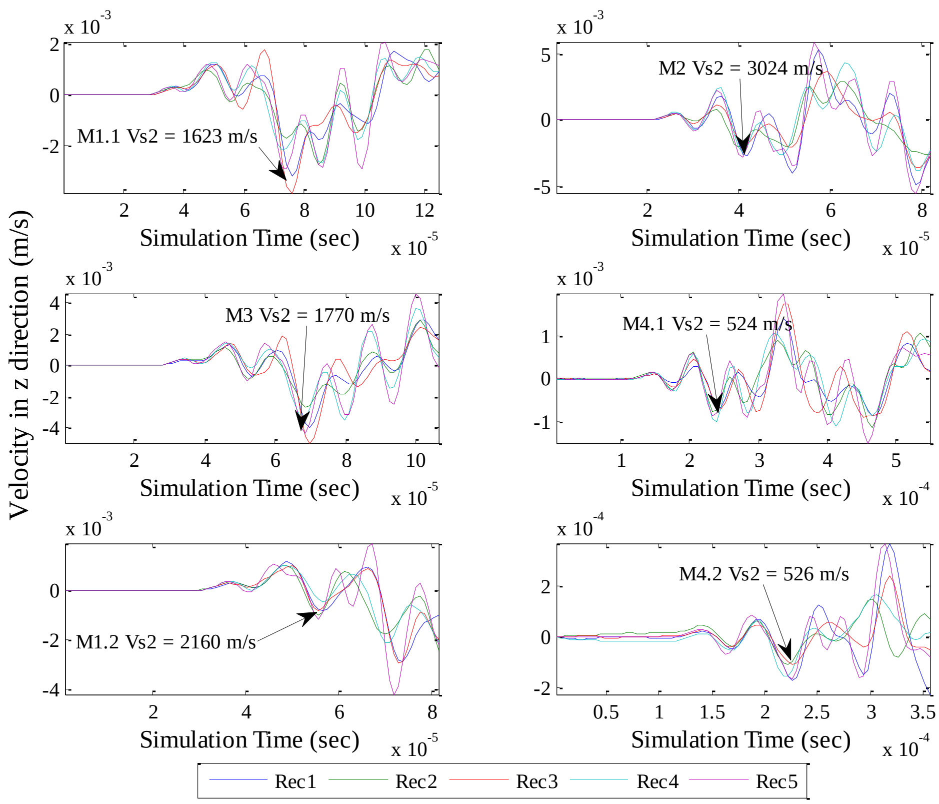


Figure 3.24: Intact transverse (z-direction) wave pulse, frequency = 564 kHz

The change in the response with different vuggy networks showed how the wave is dissipated in different samples. The wave speed is reduced with increasing vuggy porosity, reflected by longer arrival times. Also, the response curves from the different receivers become more disperse. Note in Figure 3.25 how vuggy samples with vuggy porosities of 0.089 (VUG_Net1) and 0.042 (VUG_Net3) have a larger spread than the other vuggy samples which have porosity between 0.006 and 0.02. The wave response curves for the remaining vuggy samples, different parent materials, and other frequencies are presented in Appendix A.

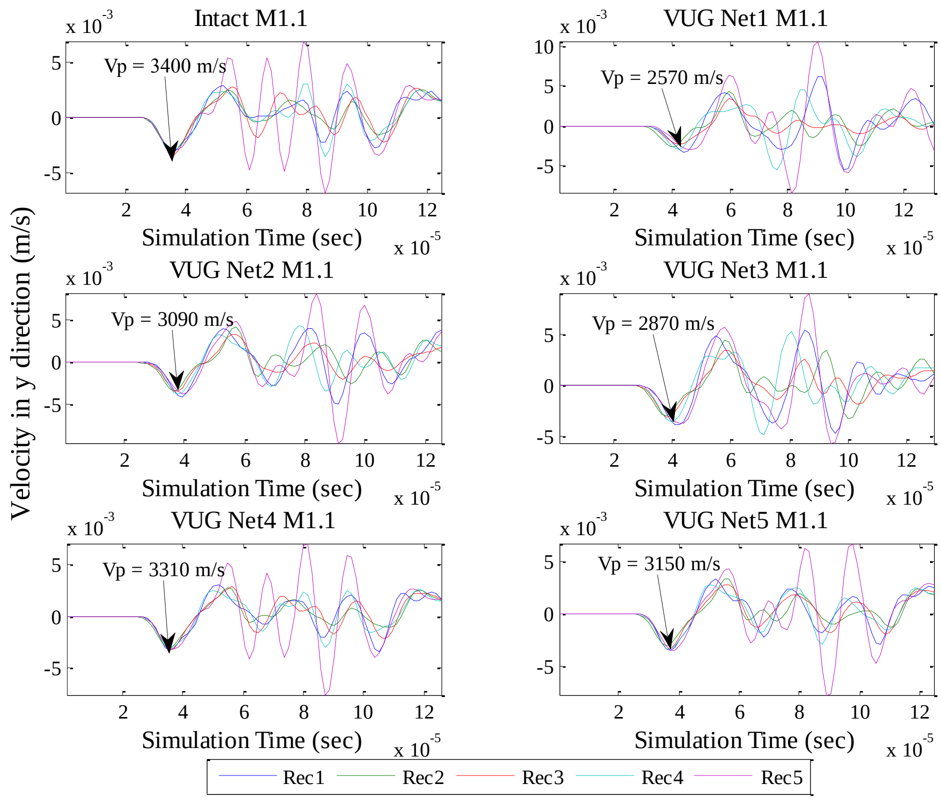


Figure 3.25: Change between intact and five different vuggy samples, response for material M1.1

The results of these dynamic tests were compiled and plotted against the vug volume ratio. The dynamic moduli obtained from the vuggy samples were normalized to the E_d from the intact samples and are presented in Figure 3.26. An empirical correlation between dynamic modulus and porosity (Equation 3.9 in Table 3.4, normalized from correlation presented by Farquar et al. 1994) is also plotted with the PFC3D results. The E_d is observed to decrease with increasing vuggy porosity, similar to the decrease observed for E_s. Moreover, the trend was independent of the original material properties. Both the normalized dynamic and static moduli are compared with the vuggy porosity in Figure 3.27. The figure shows how stiffness in both types of simulations decreases with vug volume. The scatter at higher vug volume ratio was caused by difficulties in determining S-wave arrival time, as well as the samples become increasingly anisotropic and the equations used to calculate the dynamic modulus (Eq. 3.14 to 3.16) are less applicable.

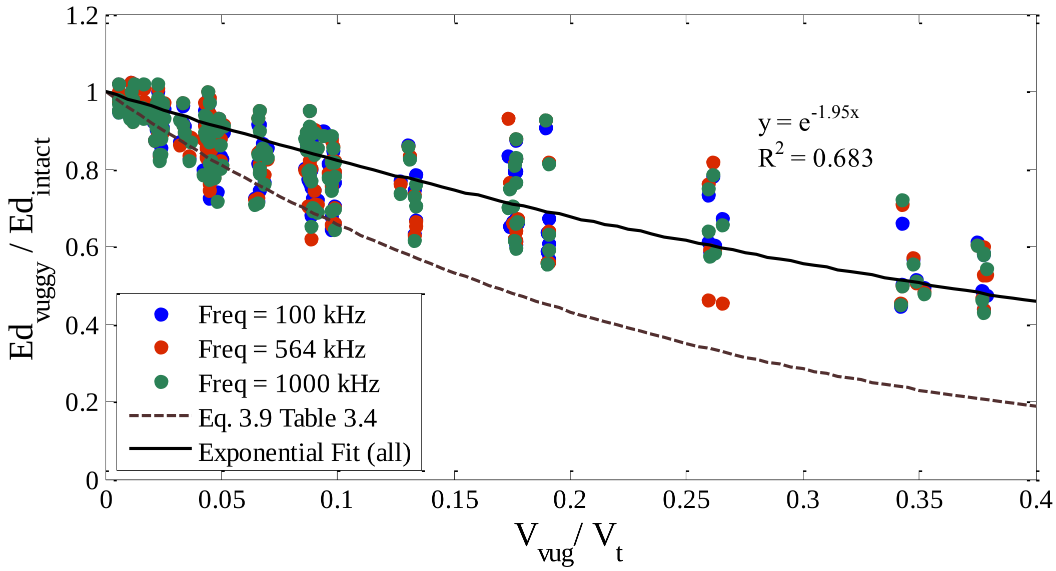


Figure 3.26: Normalized stiffness for random vuggy samples in a dynamic test with frequencies 100 kHz, 564 kHz, and 1.0 MHz

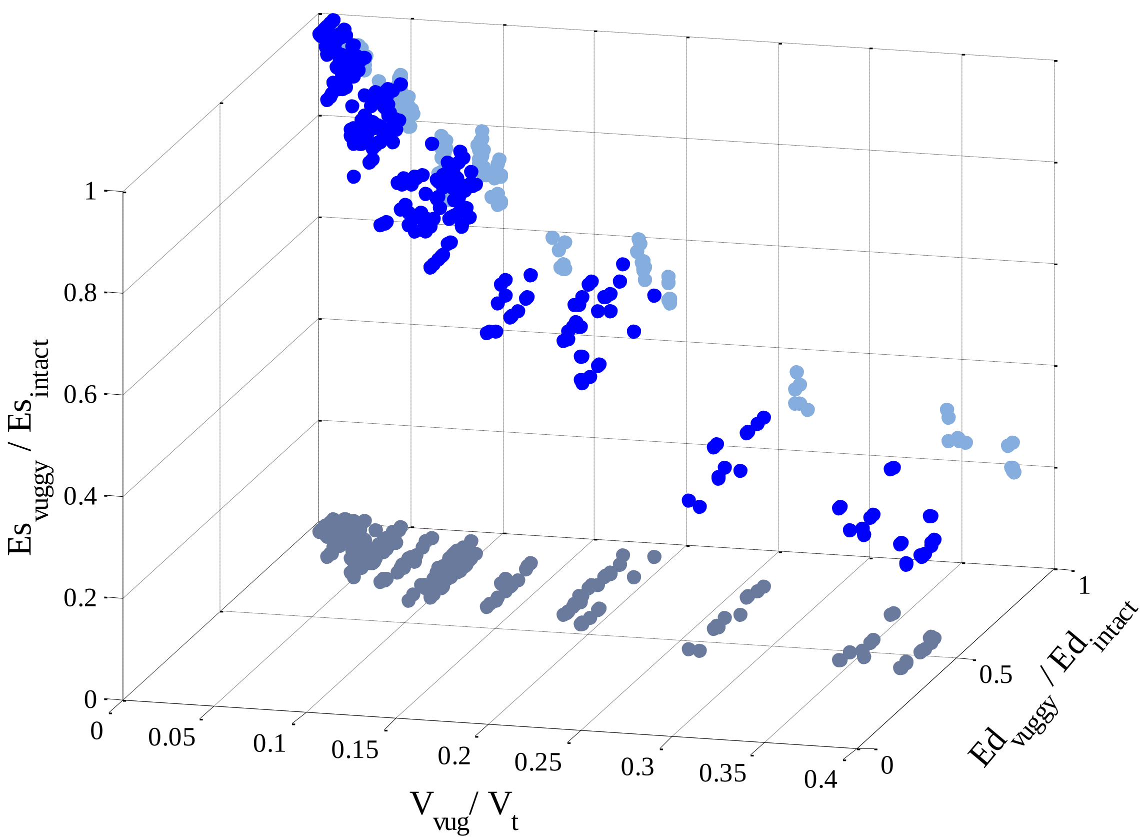


Figure 3.27: Normalized static and dynamic modulus decreasing with vuggy porosity, results include samples from various materials and vug shapes

The wave velocity simulations on the symmetric vuggy samples can be used to show the vug effect along the direction of the different shear waves. Anisotropy can first be observed when V_{s1} and V_{s2} values start to diverge. Of the 48 samples in which the vugs are placed in either the x or z planes; 29 of them showed a clear reduction in the shear wave of the same direction. The samples in which the vugs were centered or in both x and z planes the shear velocities were close to equal. Some examples of the effect of the vugs along the shear wave direction are shown in Figure 3.28. The delay caused by the interference of the in-plane vugs is shown in red in Figure 3.28. The amplitude is also reduced along the plane where the vugs are located.

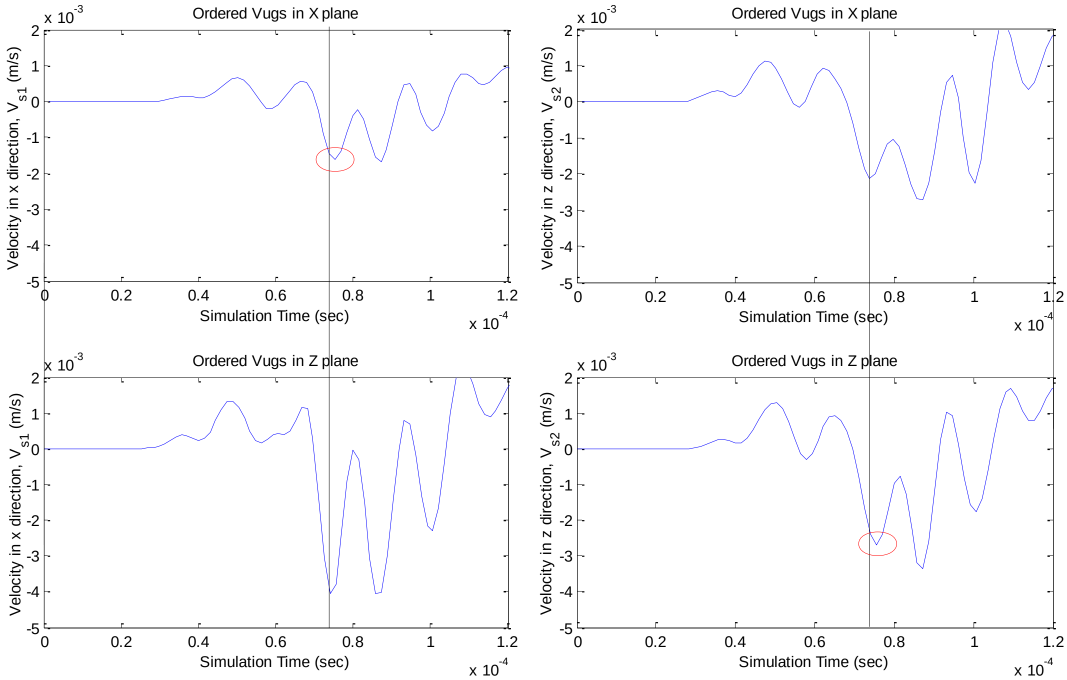


Figure 3.28: Transverse wave response for ordered vugs in the plane of the wave direction. The delay caused by the interference of the in-plane vugs is shown by the red circle.

3.8. Analysis of Dynamic Results

The velocities obtained for the intact materials M1.1, M1.2, M2 and M3 are within the velocity range presented by Stagg (1968) of 914 to 6096 m/s. The intact materials M4.1 and M4.2 are outside that range because it was originally calibrated to a much lower stiffness and UCS than the average carbonate. It was evident in the simulations that these dynamic tests were more sensitive to the ball moduli than the parallel bond moduli. The closest results between static and dynamic moduli were for the materials that had a ball modulus much higher than the parallel bond modulus (M2, M4.1 and M4.2). This observation could be explained since the transfer of the displacement pulse is through very small strains, and the stress on the bond becomes less important than on the static case. Chapter 6 further discusses the use of these dynamic tests for material calibration.

The effects of the vugs interfering with the wave pulse was observed in all samples, both for randomly and symmetrically ordered vugs. The advantage of PFC3D is that velocities can be measured in several directions, given initial indication of non-homogeneous behaviour when the shear velocities in the x and z direction start to diverge from each other. The clearest example is observed in ordered arrangements in which vugs are placed on either one of the directions (refer to Figure 3.28). The S-waves in both directions were similar only for intact and vug volume ratio below 2%; once the vug volume exceeded that threshold the velocities in the z-direction would be between 0.8 to 1.2 times the velocities encountered in x-direction.

The use of equations for isotropic homogeneous materials to calculate the dynamic parameters of vuggy samples can be considered questionable for higher vug volumes. The biggest variance or discrepancies were observed for samples that had several spherical vugs. The shear velocity was difficult to determine, and notably varied between the two transverse directions, indicating the material could no longer be considered isotropic. Future research could focus on applying velocities at different directions and developing a full elastic modulus matrix as described by Thomsen (1986); some examples of laboratory tests with these types of assessment can be found in Meléndez-Martíniz and Schmitt (2013) and Petružálek et al. (2013).

The PFC3D simulations show an adequate increase in wave velocity with increasing stiffness. The intact dynamic modulus obtained is lower than the intact static modulus for all cases, which is contrary what is found in real rocks. This could stem from bad calibration of the ball modulus. However, it is noteworthy that in vuggy samples the dynamic modulus exceeds the static in several tests. In the case of material M4.2, the dynamic stiffness exceeded the static in all samples with vuggy porosities greater than 2.2%. The normalized exponential equations for the stiffness show that the exponent for the dynamic stiffness is lower than the static stiffness. The difference in values indicates that the effect of vugs is reduced in a dynamic (small strain) environment when compared to the static case. This effect is consistent with what has been observed in the field where the modulus determined by dynamic methods is always larger than the static modulus; moreover, the ratio between static and dynamic modulus (E_s/E_d) decreases with increasing of fracturing within the rock (Ambraseys and Hendron 1968). This can be extended to the presence of vugs; increasing the vuggy porosity reduces the E_s/E_d as shown in Figure 3.29.

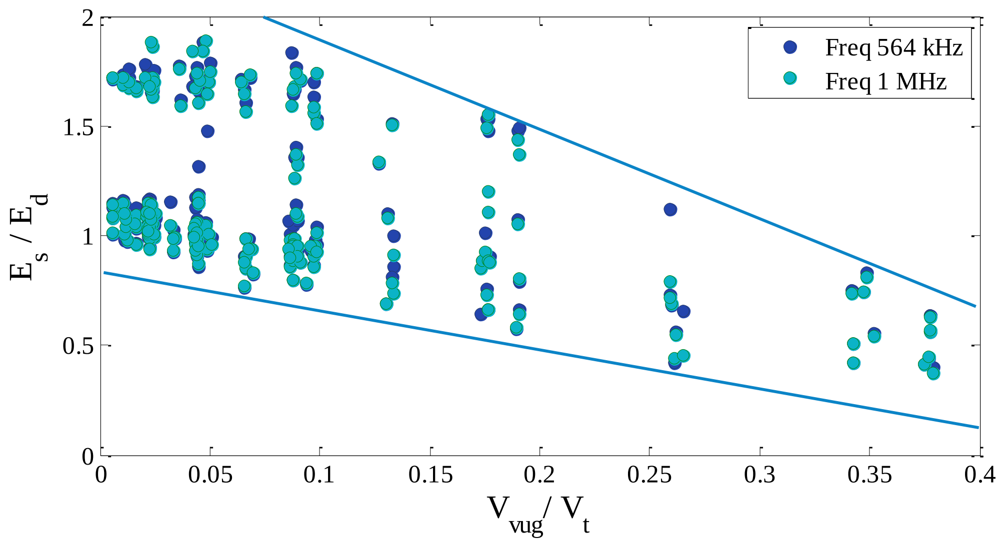


Figure 3.29: Ratio between dynamic and static modulus for various vuggy samples, the dynamic results are for frequencies 564 kHz and 1 MHz

Although in Figure 3.29 the ratio is above 1 for many samples, likely due to inadequate calibration, the overall trend is a decrease of E_s/E_d with increasing vuggy porosity. This provides the possibility of developing correlations for vuggy materials similar to the existing correlations between E_s/E_d and Rock Quality Designation (RQD). Empirical correlations between static and dynamic modulus must include a quantity that represent the rock mass quality (Ambraseys and Hendron 1968); in the case of vuggy carbonate formations instead of an RQD measurement, the rock quality could be represented by a measure of porosity or other volumetric parameter.

It is important to note that there is a difference in the dynamic modulus for rocks in the dry or saturated state, which could not be replicated in the current simulations. Dry and wet rocks transmit waves at different rates, whereas the stiffness under static loading is not greatly affected (Stagg 1968). P-waves can travel through fluids, increasing the P-wave velocity when compared to void samples; conversely S-waves do not travel through Newtonian fluids (like water) and are not affected by saturation (Ambraseys and Hendron 1968). This effect cannot be modelled with the current model without creating a PFC3D and fluid coupled simulations. Further care must be taken when the vuggy carbonates are saturated with heavy oil, which is a non-Newtonian fluid and has been found to behave as a fluid at low frequencies but as a solid at high frequencies (Gurevich et al. 2008). The high frequency behaviour of saturated heavy oil carbonates could potentially be modelled by filled vugs with a calibrated material to match the elastic behaviour of the bitumen.

3.8.1. Limitations

The most difficult part of analyzing the dynamic simulations was the identification of the S-wave arrival time. As has been found in field and laboratory tests, the determination of the shear wave is difficult and can be fraught by interference with P-waves and boundary conditions. In order to study the shear wave displacement through the sample, real time images of the simulations where done in which the vector displacements of the balls were plotted along the sample in parallel to the velocity vs time plots of the “receivers” (refer to Figure 3.30). Additionally, the vertical movement (i.e. longitudinal direction) in the receivers was monitored; initial readings on the S-wave tests coincided with increase movement in the longitudinal direction indicating that the pulse arrived from partial component rotation into a faster P-wave.

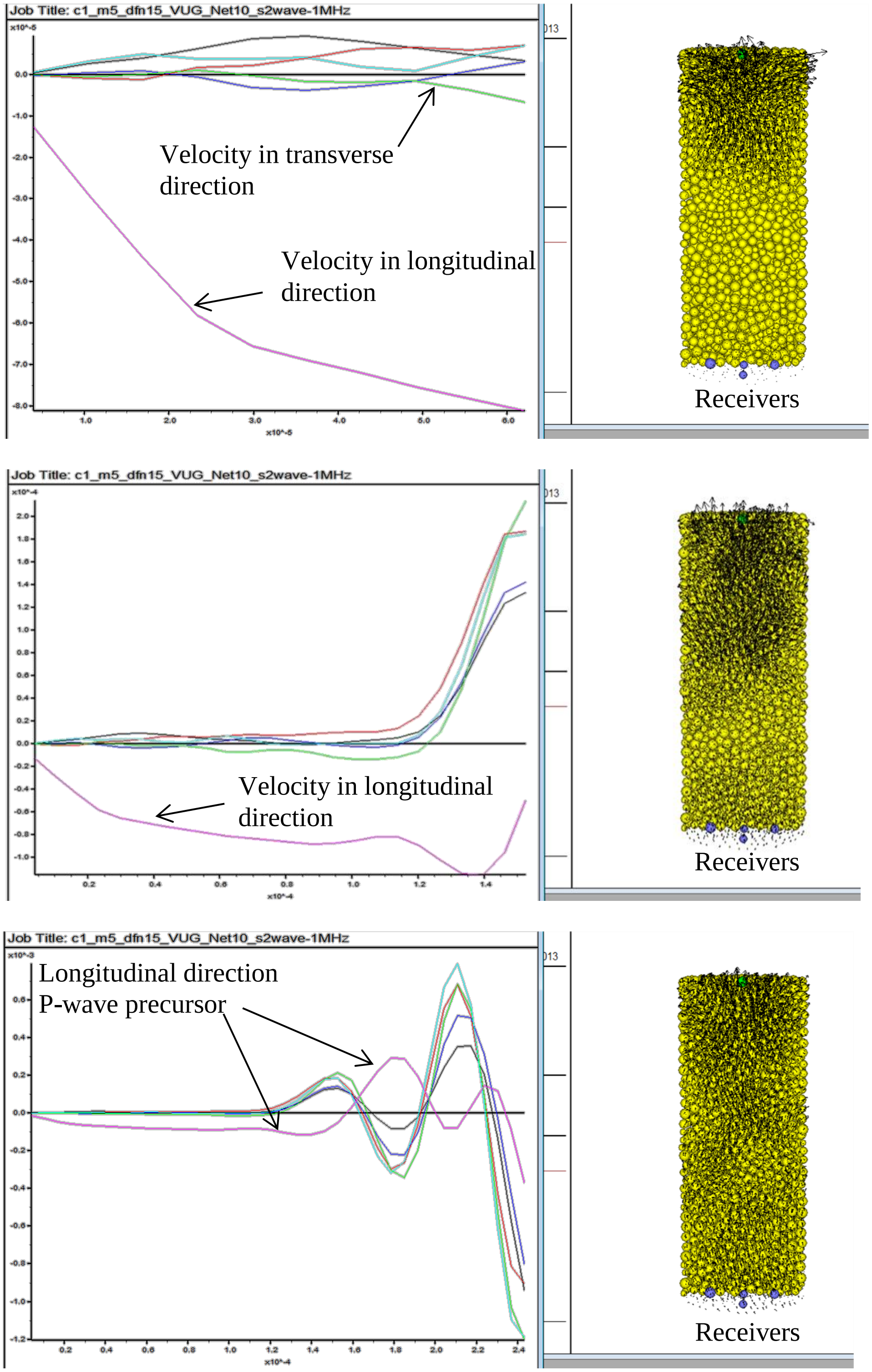


Figure 3.30: Visualization of S-wave analogous test in PFC3D

The arrival time for the S-wave was generally taken as the first large negative trough, and all attempts were made to maintain consistency within the materials. The ease in determining this point varied with material type and presence of vugs. In samples with low stiffness or with a large numbers of vugs, the assessment of the S-wave became very difficult and was subject to error. In all cases, the velocity measurements were checked to ensure the ν_d and E_d was within a reasonable range. The best indication of mistaking faster P-wave arrivals with the S-wave was obtaining a negative ν_d. With the combination of ball displacement monitoring, velocity checks in all directions, and maintaining a consistent criteria, all attempts were made to be as accurate as possible in the S-wave arrival time; however, the results at large vug volumes had large variations.

These P- and S-wave simulations could be considered an analogy for laboratory wave velocity tests. It is not part of the regular testing environment provided by Itasca and was developed during this research based on previous work by Holt et al. (2005) with Hertzian contact law. The purpose of using this dynamic environment was to provide insight on the effect of vugs under low strain or elastic applications. The fish code developed to execute these tests is presented in Appendix B. More research is encouraged on this dynamic behaviour, as it can potentially provide an avenue for calibrating ball stiffness properties for dynamic testing. Refer to Chapter 6 for further discussion on the calibration of materials with both dynamic and static test data. Hopefully, the results presented encourage future research in this area and the presence of more rock physics testing in traditional reservoir geomechanics.

CHAPTER 4: VUGGY CARBONATE TESTING WORKFLOW

Imaging technologies have progressed the characterization of rocks, with techniques such as Scanning Electron Microscopy (SEM) and X-ray computed tomography (CT) providing great amounts of information on the rock structure to the micro (and even nano) level. Some techniques, such as SEM, have been used as an analytical method for carbonate geological classification for many years. More recently, micro and nanoCT have been used by core analysts to study petrophysical and flow properties of rocks (Zhang et al., 2012). CT imaging at different resolution scales are combined with Digital Rock Physics (DRP) technology, to characterize detail pore structures and predict flow behaviour in rocks (Lopez et al. 2012).

The use of imaging technologies for predicting geomechanical properties of rocks is also on the rise. Shulakuva et al. (2013) used micro-CT images for creating finite element model (FEM) simulations to predict the elastic and bulk modulus of sandstone samples. The FEM simulations were compared with experimental data for the sandstone samples; thus providing some confirmation to the models. Moreover, X-ray tomography has been used to study damage and crack formations in samples under loading; with technology available to have real-time x-ray scans during triaxial tests (Lenoir 2007; Zinsmeister et al. 2013). CT imaging during triaxial testing combined with Digital Image Correlation (DIC) has allowed for mapping and tracking internal damage within the samples during the stress application (Zinsmeister et al. 2013).

The technology to combine CT scans with numerical models or simulations has greatly increased in recent years, with commercial software available from Xradia, VSG (Avizo) and iRock technologies, to name a few. As CT imaging becomes more widely available it is important to establish a workflow that can be followed to combine this technique with existing laboratory testing and rock modelling methods. This chapter provides such a workflow developed for the study of vuggy carbonate samples with PFC3D software and CT imaging.

4.1. Sample Preparation

Initial evaluation of the core samples either by visual inspection or x-rays should be done to identify the samples that will be used for testing. For samples saturated with bitumen, the bitumen may be extracted prior to conducting the CT scans. After the samples are cleaned they should be prepared to an appropriate samples size with a height to diameter ratio between 2:1 and 2.5:1. In cases when the bitumen cannot be extracted or the sample will be tested with bitumen, the samples should be sealed to avoid any pore fluid leakage during the CT scanning. The size of the samples should take into account the space available in the CT scanner and the resolution available for the CT. The size of the sample and the resolution requested will affect the time required for the CT scan.

4.2. Intact Material Testing and Calibration

Samples which are considered “intact” would be subjected to pulse velocity and unconfined compressive strength (UCS) tests. The intact sample should be selected from samples that were previously scanned and determined to have low to no cracks or vugs, and appear more homogeneous. If possible, micro-CT of the intact samples should be conducted to evaluate the initial state of the material and identify small flaws that could add error to the tests. The P- and S-wave velocity measurements will be used to determine the dynamic modulus of the unconfined dry samples (refer to ASTM D2845 – 08). These values can be used to calibrate the ball modulus and stiffness ratio based on the analogous P- and S-wave tests described in Chapter 3.

The UCS test should be conducted as per ASTM D 7012 – 10 with axial and radial strain measurements in order to provide UCS, E_s and ν_d. These results will be the macro-properties for calibration of the PFC3D samples in the UCS virtual testing environment. The dynamic results can be used to calibrate a separate material or attempt to calibrate both static and dynamic tests (refer to Chapter 6 for further discussion). At least two material resolutions (refer to Table 4.1) will be used initially to test the impact of model and vug resolution on the effectiveness of the sample representation.

Table 4.1: Resolution and sample realizations for calibrated materials

Resolution (Particle/diameter)	Seed number realizations
15	10
20	5

A minimum resolution that allows for several particles between free-surfaces should be used to reduce the variability of the results. However, it is difficult to know what resolution would be required without having previous knowledge of the location of the vugs within the samples. Thus, it is recommended that the micro-properties for possibly three or more materials with different resolutions be calibrated if samples with very high vuggy porosity are expected.

Results from Chapter 3 show that a resolution of at least 19 particles per diameter was required for samples with vuggy porosity above 25% to reduce stiffness variations at high porosities. The resolutions and seed number realizations shown in Table 4.1 are an initial recommendation, and should be adapted depending on the project. Seed number realizations are sample replicates in PFC3D in which all the micro-properties are equal except for the seed number. The seed number varies the locations of the balls within the sample generation stage, which causes fluctuations in the results obtained for the macroscopic behaviour. The macro-properties should be calibrated to the averaged results from all the seed number realizations. In general, the number of realizations can be reduced as the resolution increases because the variability between the results decreases.

4.3. CT scan and Image Data Processing

The data provided by a CT scan is a series of images that represent sequential slices along the length or diameter of the samples. Very small-scale resolution for the CT scan is not required unless micro-pore identification is desired. A scale of 34μm was found to be more than sufficient to identify voids in the vug size range, and lower resolution (on the mm scale) could be used for large samples with bigger vugs. The processing of the image data was executed with Matlab code using available functions from Matlab's Image Processing Toolbox. This code has been made available in Appendix B and a description of the image processing stages is provided below.

4.3.1. Image compilation to 3D image

The first step involves compiling the image sequence into a single 3D image. Note that the distance between each slice should be the same as the resolution used for the images; thus the voxels (3D version of a pixel) have equal size in all directions. Figure 4.1 is a graphical representation of the CT scan images for a trial carbonate sample. Once the 3D image is built, slices through the samples can be done at different angles from the original images. For example in Figure 4.1, the original CT scan images were taken in the horizontal plane, but slices in the vertical direction can now be produced from the 3D image.

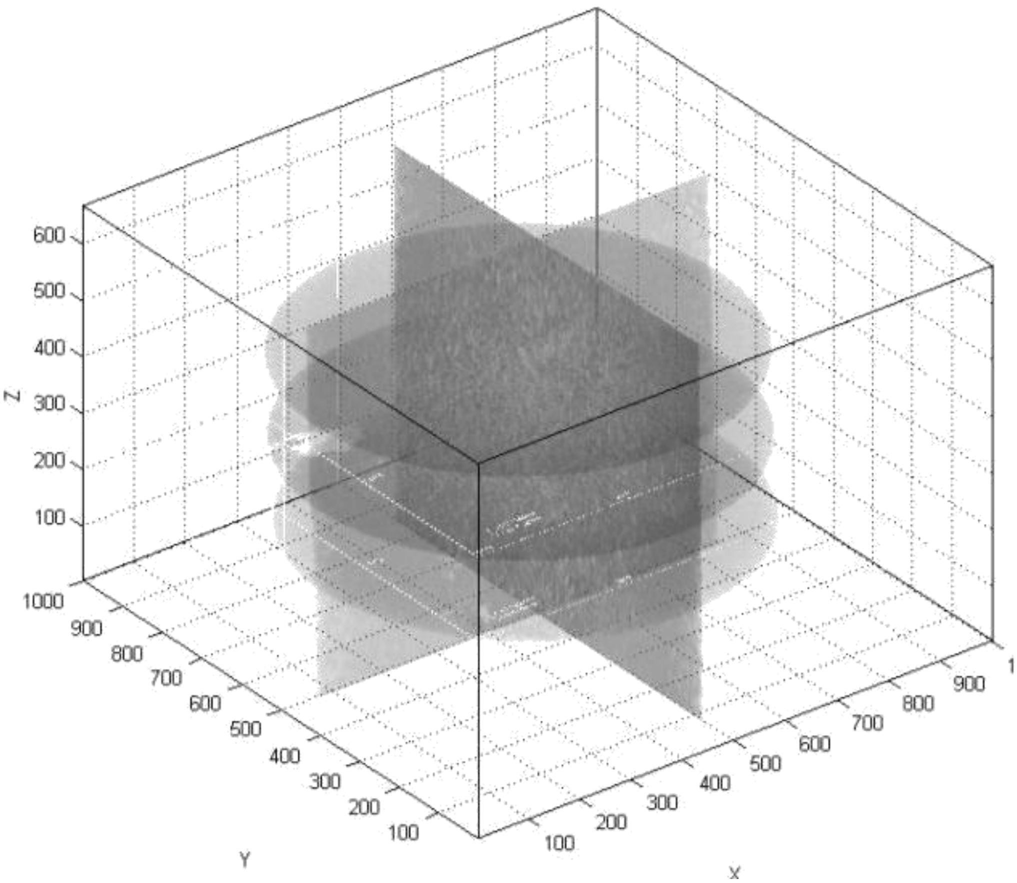


Figure 4.1: CT scan 3D image of a 1” diameter by 1” height sample

4.3.2. Thresholding and Identification of Vugs

Image thresholding identifies vugs by converting the grayscale images into a binary image. A threshold value is chosen in such a way that every voxel with a grayscale above that level is considered true or 1 (solid) and all values below are false or 0 (void). The threshold level will depend on the images, but can be identified by trial and error or by evaluating the intensity histogram of some sample images (Gonzalez 2009). An example of a binary image created from thresholding is showing in Figure 4.2b. The process can be aided by filtering the image for a Region of Interest (ROI) to only evaluate the sample area and discard other data on the edges.

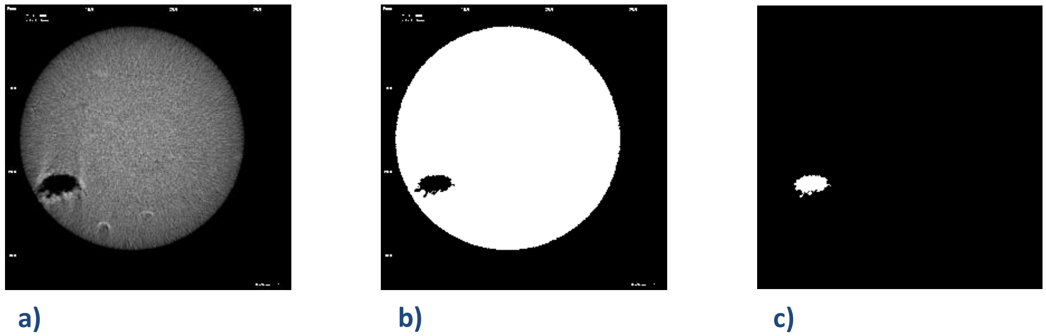


Figure 4.2: Image processing phases. a) CT scan image, b) Binary image obtained from thresholding grayscale at level of 0.08, c) image compliment in which white values are voids

From the binary image in which solid is 1 and void in 0 (Figure 4.2b); the void voxel can be obtained from the image complement. In the complement of a binary image the black and white are reversed (Mathworks 2012); and an image like Figure 4.2c is produced, which only shows the voxels within vugs. The vugs are identified by grouping together voxels from the complement image that are connected to each other. The connectivity of the voxels is evaluated using the *bwconncomp* function from Matlab's Image Processing Tool Box. For three-dimensional connectivity there is the option to consider 6, 18 or 26 connected voxel neighborhoods; in the present case the default connectivity of 26 was used. The function creates a list of regions that can be evaluated further for area/volume, centroid, and other geometric properties. Each region is identified as a vug, and a list of voxels belonging to the vug is created. Figure 4.3 shows a 3D representation of the vugs identified in a trial carbonate sample.

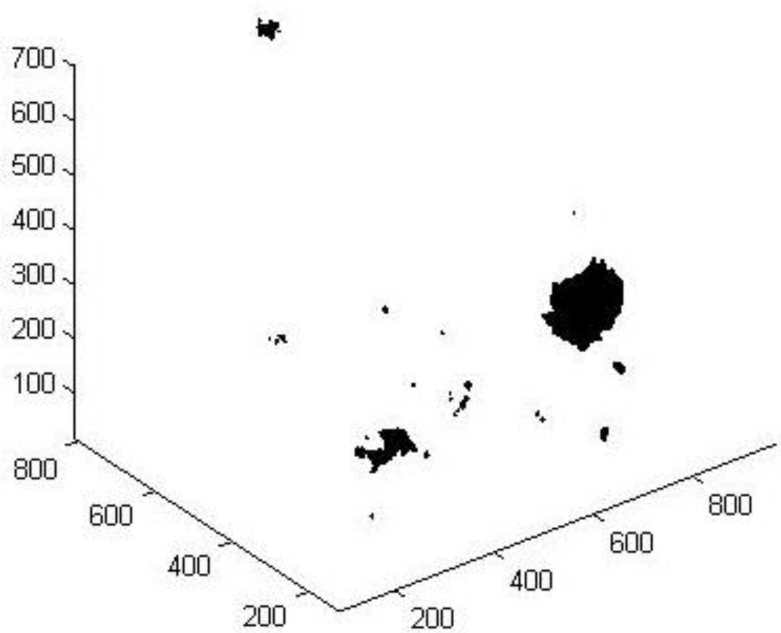


Figure 4.3: Voxels identified as vugs in 3D plot

4.3.3. Vug Parameter Outputs

The process for representing vugs in PFC3D was described in Chapter 3, and consists of either ellipsoid or prismatic shapes that can be modified by location and axis length. This is in contrast with existing methods that transfer CT images to FEM by constructing a finite-element mesh. The meshing processing uses triangular approximation to divide surfaces in triangle sets in 2D and tetrahedrals in 3D (Shulakova et al. 2013). The complexity of the mesh can be modified depending on the accuracy required for surface representation and on the computational capacity available. Figure 4.4 shows examples of FEM created based on CT imaging of sandstone, which was used by Shulakova et al. (2013) for testing sample stiffness under different stresses.

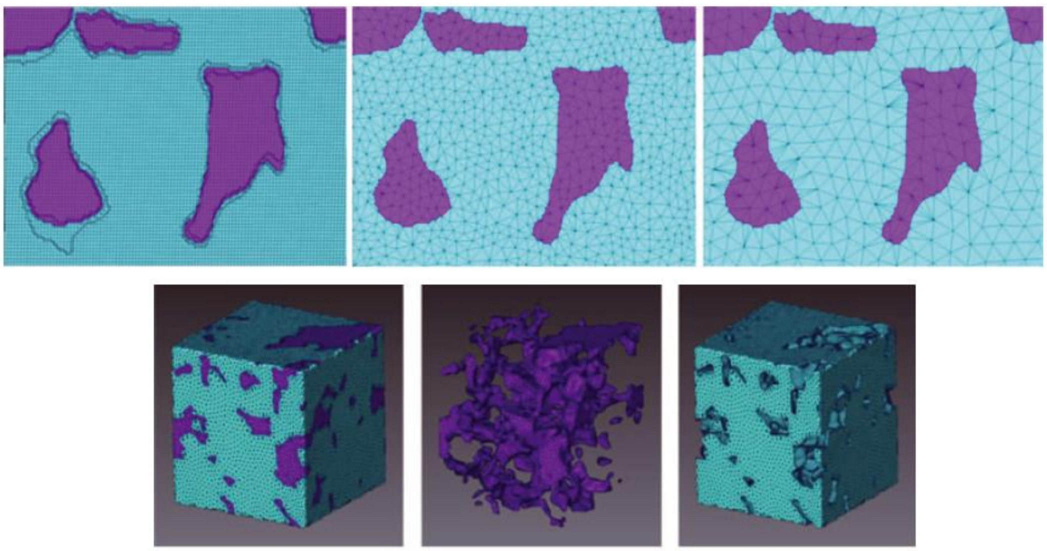


Figure 4.4: CT scan representation for FEM (blue is rock matrix, purple is pore space). Top: stages of surface simplification with changing mesh sizes. Bottom: three dimensional mesh (from Shulakova et al. 2013)

The transfer of the image data to PFC3D is not as simple as creating solid models for FEM with automated meshing techniques. The current method used for creating vuggy samples requires that an initial intact material is created and the vugs are introduced by eliminating balls within the desired shape. Accuracy of the represented shape will be dependent on resolution and may be difficult to obtain due to the spherical shapes in PFC3D, which cannot be cut or trimmed. Three methods are proposed for the vug representation and sample generation in PFC3D and are presented below. These methods try to represent the geometry of the vugs in different ways, and will be compared to determine which would provide both an accurate and relatively simple method for representing the vugs.

4.3.3.1. Voxel Method:

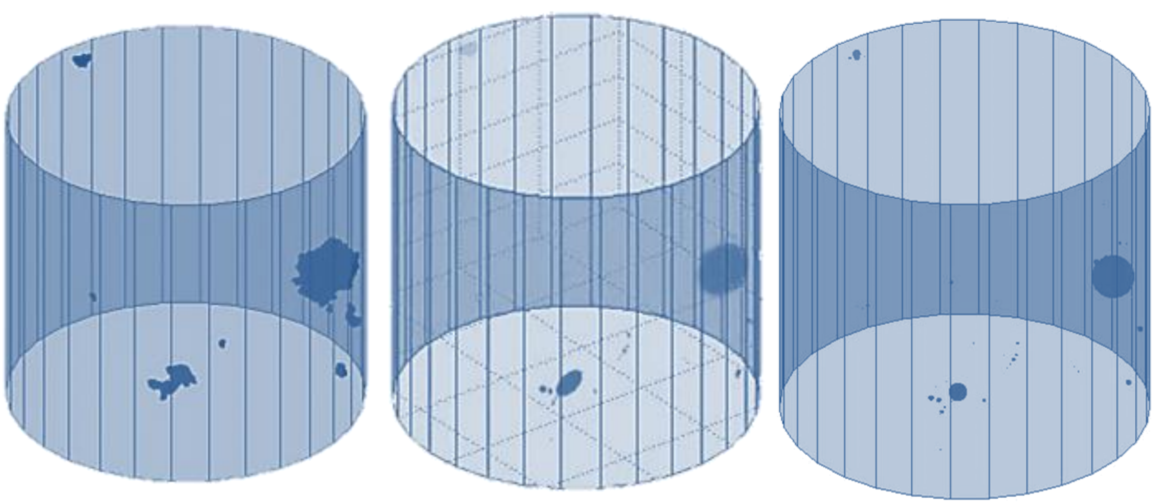
In this method each voxel identified as void is transferred to PFC3D by eliminating any ball within a cube of each voxel. The advantage of this process is that it most accurately represents the shape of the vugs. However, it requires a high resolution in PFC3D which must be close to resolution of the CT scan; otherwise vugs might be over represented by eliminating a ball centered at the void location but larger than the voxel size. This method also can take a large amount of time, since the voxel list can be several thousands of items long if the sample is very vuggy. An option to reduce the time for sample creation is to eliminate the vugs that have a number of voxels below a certain value and consider them too small to be represented. This will reduce the number of voxels to evaluate while still capturing most of the vuggy porosity.

4.3.3.2. Equivalent Ellipsoid Method

This method consists of replacing the regions or vugs identified with the connectivity evaluation by ellipsoids with equivalent inertia or 2nd moment. The Matlab file provided by David Legland (2011) entitled *Image ellipsoid 3D* was modified to create a code that generates an ellipsoid that would encompass the void voxels. The procedure is based on single value decomposition of the voxel coordinate list, in which the resulting diagonal matrix is equivalent to the main axis of the ellipsoid. The location of the ellipsoid is obtained from the centroid data generated for each vug. The advantage of this method is that it synchronizes well with the fish code currently used to represent vugs in PFC3D that employs ellipsoids. An equivalent ellipsoid allows a representation of the shape (second moment) of the vug, with a simpler shape.

4.3.3.3. Equivalent Sphere

This method represents the vug as a sphere with equivalent volume located at the centroid of the vug. The volume and center of the sphere are obtained from the region functions standard in Matlab's Image Processing Toolbox. This is the simplest representation for the vugs, only providing volume and location. All three methods are graphically represented in Figure 4.5 and illustrates that while the voxel method is the most accurate in representing shape, it may be sufficient only to evaluate the volume.



a) b) c)

Figure 4.5: Example of voxel (a), equivalent ellipsoid (b) and equivalent sphere (c) vug representations.

The three methods above present varying degree of complexity, and should be tested at different resolutions. The difference, if any, observed between the methods will help evaluate the relative importance of volume and shape in the geomechanical behaviour of the samples. The results will help determine which geometrical parameters should be considered for quantifying strength and stiffness reduction in vuggy samples.

The samples will be created with PFC3D with the proposed methods and varying resolutions. Images of the PFC3D should be evaluated to visualize how accurate the shape of the vugs is represented. The vug volume ratio for all methods will be tracked and compared to the vuggy porosity calculated with the CT scan images to ensure they do not diverge greatly from each other. Some examples on variations of vug representation (from the same CT imaging data) due to resolution and method are shown in Figure 4.6.

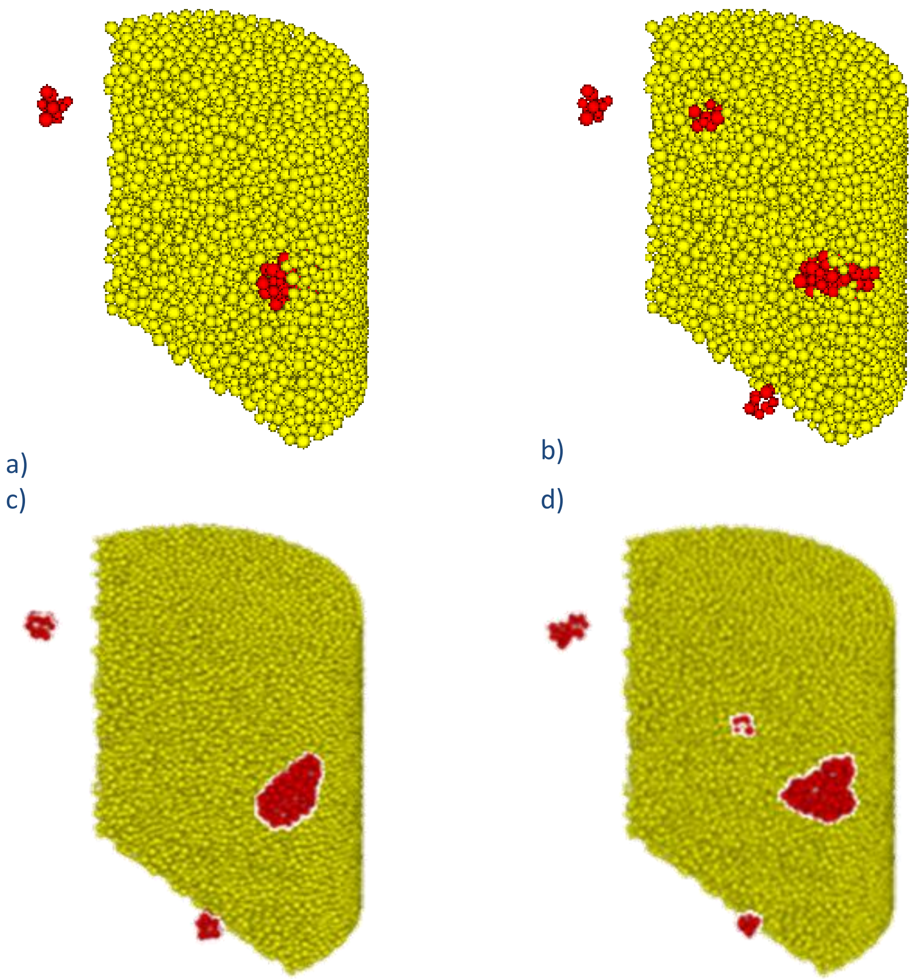


Figure 4.6: Different vug shapes obtained using two resolutions with voxel (b and d) and equivalent ellipsoid (a and c) methods

4.4. Vuggy sample Testing

4.4.1. Laboratory Testing

The vuggy samples, previously scanned, will be submitted to UCS tests in the laboratory. The sample edges must be trimmed carefully as smooth and even as possible (if not done so before the CT scan), avoiding vugs in the ends of the samples whenever possible. The

Samples should be measured, if any changes in geometry have been made after the CT scan. If possible, velocity wave measurements should be done on the samples; however, if the specimens have high porosity and are saturated with brine or bitumen, the P-wave measurement may over-predict the modulus. A low strain rate should be used, and video or pictures should be taken before, during and after the test. Radial strain should be recorded to provide a measure for lateral strain stiffness required for calculating ν_d.

4.4.2. Virtual PFC3D Testing

The virtual samples will be tested in the PFC3D triaxial test environment under UCS conditions, i.e. without the radial wall. The samples will be created under a confinement of 100 kPa or atmospheric pressure. A low strain rate shall be used during the test, especially if there are vugs within the edges of the samples. A few trial runs could be conducted at different strain rates to ensure early cracking is not occurring and particles are not separating from the main samples or moving large distances away.

Video of the virtual test can be produced in which the vugs are marked and the bond breakage in the sample is documented as the test progresses. The maximum stress, E_s, and ν_s will be recorded and compared to the laboratory results. The strain-stress curves will also be compared but it is likely that they will not match due to the difficulty in capturing the initial discontinuity in soft rocks with PFC3D. The analogous P- and S-wave velocity test for PFC3D described in Chapter 3 will also be conducted with the virtual samples at a similar frequency as the one used in the laboratory tests. All PFC3D samples will be conducted in replicate with different seed numbers, as described in the intact material calibration section above (see Table 4.1). At least two different resolutions will be used in the initial evaluation of the workflow.

4.5. Results Comparison and Model Evaluation

The UCS, E_s and ν_s from the virtual and laboratory samples will be compared; as well as the dynamic modulus measurement (if the laboratory measurements were conducted). Pictures from the laboratory test and graphics from the virtual model will be assessed to determine if PFC3D predicted the breakage pattern correctly. Special attention should be given to any cracking pattern around the vugs. Any major discrepancies between the PFC3D models and the laboratory testing will be noted. In the future CT scans during triaxial testing may allow even better comparisons by providing real-time crack formations that can be compared with bond breakages in the PFC3D models.

As part of the evaluation of the workflow, the PFC3D models from the different resolutions and vug representation methods will be compared. The crack formation and progression patterns, as well as the numerical results (UCS, E_s, and ν_s) will be considered when determining which vug representation method is most accurate. Furthermore, the difference between voxel and equivalent sphere method will be evaluated to elucidate the relative importance of volume and shape as geometric parameters to predict the effect of vugs.

4.6. Workflow Summary

The workflow described above is summarized in Figure 4.7; the CT scanning and laboratory testing section of the workflow is presented in Chapter 5. Chapter 6 presents the calibration of the PFC3D models and their validation with the physical laboratory tests. The procedure presented is hopefully the first step in further experimentation with virtual representation of real samples. Further discontinuum work should include simulations under different confinements, both for static and dynamic testing. Moreover, increasing the scale of the samples or conducting different stress path, like extension tests, can further increase the practicality of this workflow and initiate a path towards non-destructive testing of core samples.

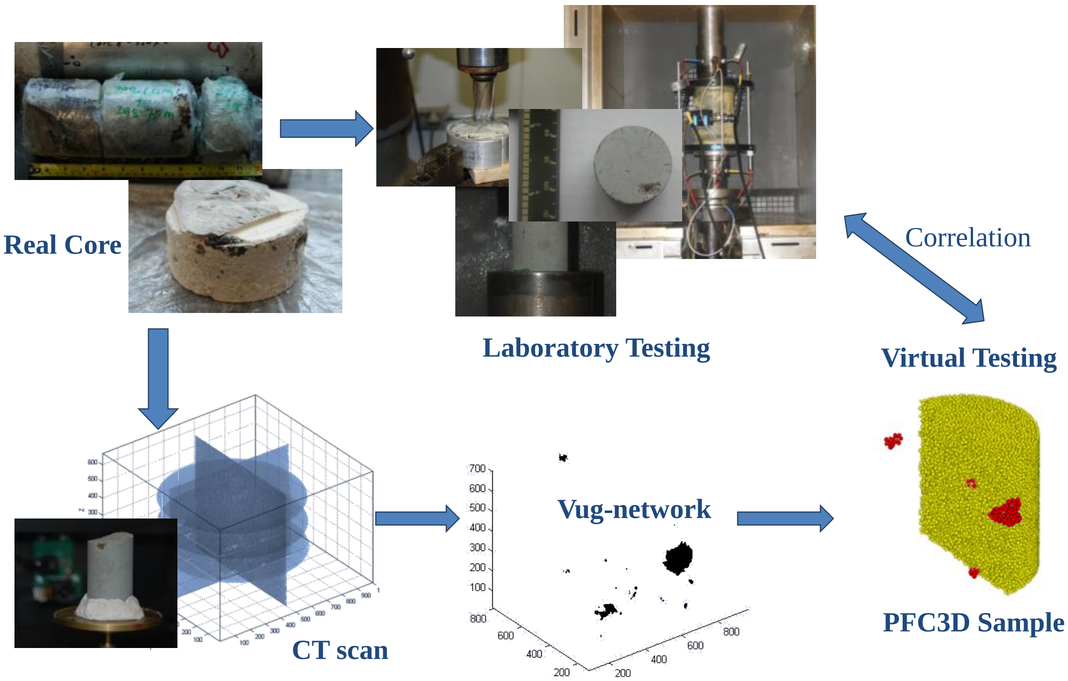


Figure 4.7: Graphical representation of the vuggy carbonate testing workflow

CHAPTER 5: LABORATORY TESTS

5.1. Sample Description

Vuggy carbonate samples from the Grosmont Formation were obtained thanks to the contribution of Laricina Energy. Six samples of varying porosity from Grosmont C Unit were acquired from the core recovered during drilling of an observation well. A summary of the sample elevation is presented in Table 5.1, and pictures of the Grosmont core are presented in Figure 5.1. The samples ranged from low amount of vugs for potential intact testing, to very high vuggy porosity (see bottom left section in Figure 5.1). The amount of vugs within the sample was verified with CT scans, as discussed in the following sections.

Table 5.1: Core information and depth of the carbonate vuggy samples

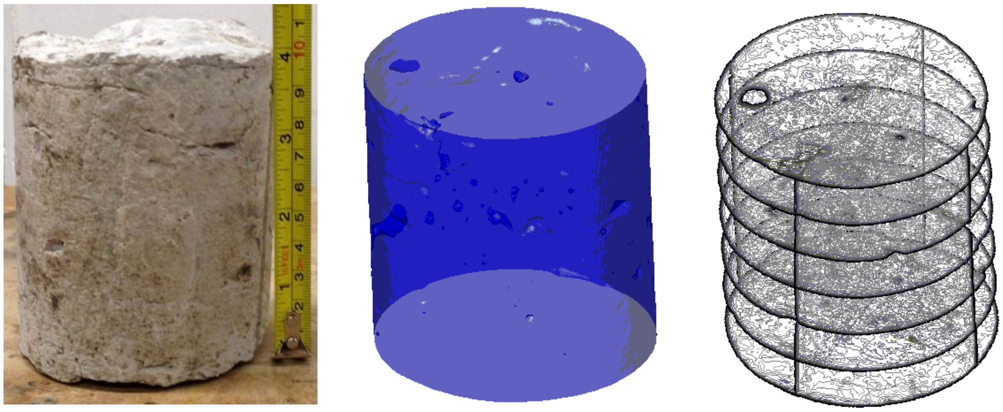
Sample #	Depth Interval (m)	Length (cm)	Core	Box
1	386.180 - 386.350	17.0	18	6 of 7
2	382.780 - 382.905	12.5	18	4 of 7
3	380.592 - 380.770	17.8	18	2 of 7
4	377.390 - 377.564	17.4	17	7 of 7
5	361.180 - 361.320	14.0	16	1 of 7
6	360.945 - 361.050	10.5	16	1 of 7

Figure 5.1: Photographs of Grosmont carbonate core and segments were samples were extracted.

5.2.Material Testing

5.2.1. Sample Coring

Samples 2 and 6 were identified as potential intact specimens due to the low number of vugs observed on the surface of the sample (see Figure 5.2a for S6). A CT scan of sample S6 prior to coring showed only a few voids, which were located near the outer diameter and edges of the samples (refer to Figure 5.2). Smaller 1.5" diameter samples were cored from Sample 2 and 6 to be used for intact sample calibration. Initial attempts were made to obtain two 1" (25.4 mm) diameter samples from S6, using a milling machine and air as the coring fluid; however, the material was too brittle and initial damage of the sample edges were observed. Thus, the samples were cored at a larger diameter of 1.5" (38.1 mm), using water and a coring machine at a rate of 0.003"/revolution (0.076 mm/ revolution). The samples were levelled and trimmed to obtain a length to diameter ratio between 2 and 2.5 as required by ASTM D2845-08 (see table 5.2). Photographs of the final cored samples S2 and S6 are presented in Figure 5.3.



a) b) c)

Figure 5.2: Vuggy carbonate Sample 6. a) Photographs, b) Digital image of sample surface, c) Contours of equal greyscale value based on CT imaging data

a) b)

Figure 5.3: Photographs of cored samples S2 (a) and S6 (b)

Based on the difficulties encountered coring S2 and S6 and the high number of vugs that were visible in the other samples, it was decided to minimize any alterations to the remaining four samples. There was a high risk of damaging the samples during coring to a smaller diameter; thus, these samples were kept in the same core diameter of 3.5” (88.9 mm). The ASTM requirement of 2:1 height to width ratio could not be obtained due to limited sample size and availability. Samples S1, S3, S4, and S5 had length to diameter ratios varying from 1.3 to 1.9 (refer to Table 5.2). The ends of the samples were trimmed and levelled to obtain as smooth surface as possible. Smooth ends, however, were not achieved for all samples due to the random location of the vugs, as can be observed in Figure 5.4. Additional photographs are presented in Appendix C.

Figure 5.4: Top and bottom face for sample S4, smooth faces could not be obtained due to the presence of the vugs

5.2.2. CT Scan

All samples were scanned in collaboration with Dr. Riauka, an Associate Professor in the Medical Physics Division of the Department of Oncology at the University of Alberta. The machine used was a Toshiba Aquilion 64 Slice CT scanner model. The image data was analyzed using the Image Processing Tool box in Matlab. The CT resolution for the samples was between 0.3 to 0.6 mm thick slices, but the data was scaled and standardized to a voxel size of 0.3 and 0.5 mm for the sample size of 1.5” and 3.5“ diameter, respectively. The data was scaled to have equal pixel and slice dimensions; visualizations examples for S3 and S4 are presented in Figures 5.5 and 5.6, respectively. The procedure for determining the number and location of vugs described in Chapter 4 was used for all six samples. Sample 6 was determined to have low vuggy porosity and was chosen for intact calibration of the PFC3D models. The vuggy porosity for each sample was calculated by determining the total volume of void voxels and dividing it by the sample volume. The CT scan resolution and vuggy porosity for each specimen is presented in Table 5.2.

Table 5.2: Vuggy porosity for carbonate samples calculated from CT-scans

Sample	Diameter (m)	Height (m)	H:D ratio	Resolution (mm)	Void voxels	Vuggy Porosity
S1	0.0876	0.1624	1.9	0.6	123879	2.73%
S2	0.0377	0.0848	2.2	0.3	78354	2.24%
S3	0.0872	0.1361	1.6	0.6	362792	9.64%
S4	0.0874	0.1096	1.3	0.6	483788	15.89%
S5	0.0874	0.1096	1.3	0.6	233204	7.66%
S6	0.0377	0.0829	2.2	0.3	6388	0.19%

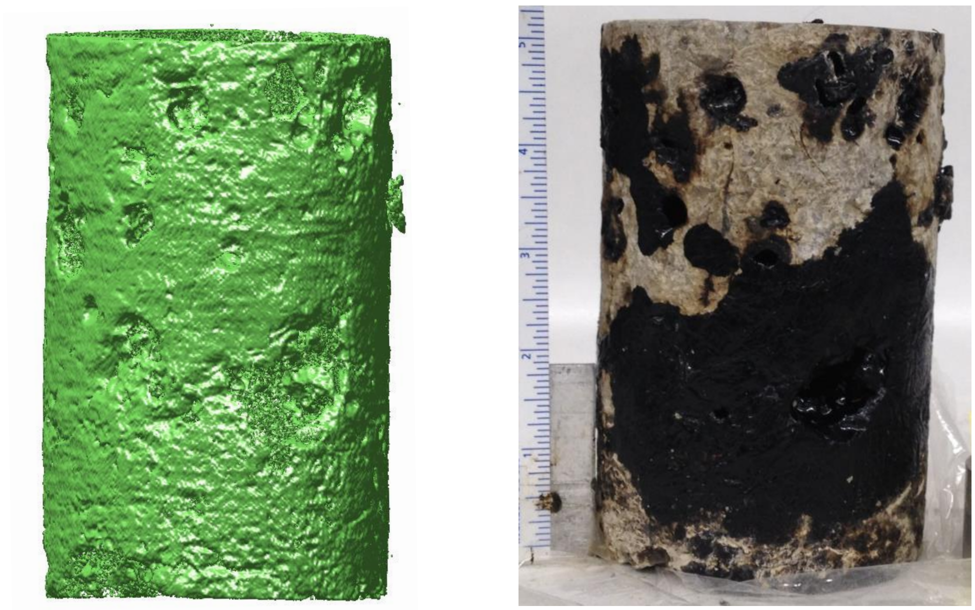


Figure 5.5: Digital image and photograph of Sample S3

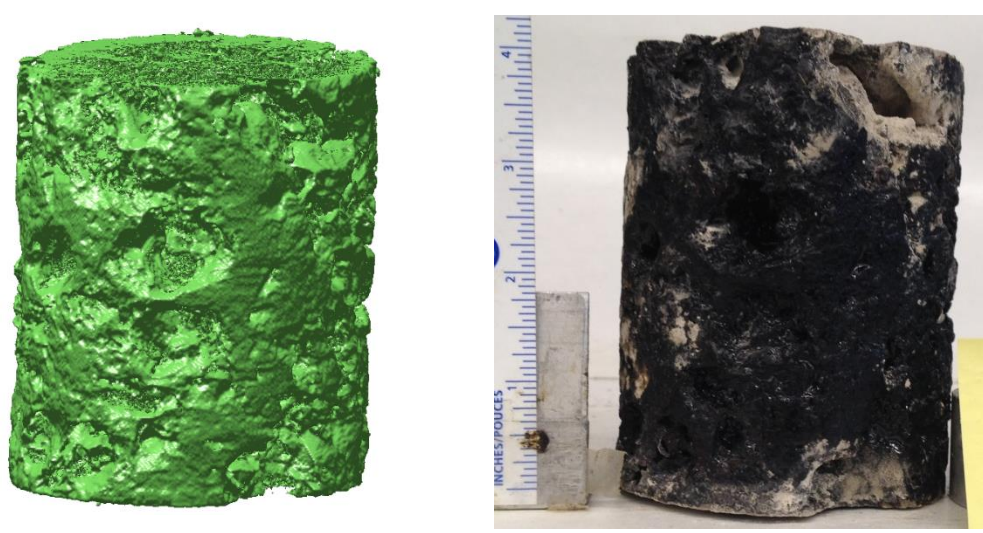


Figure 5.6: Digital image and photograph of Sample S4

5.2.3. Laboratory Testing

All samples were tested within a uniaxial compression test set-up that included sonic wave velocities. The uniaxial compression tests were carried out at a strain rate of 0.2%/min. The experimental set-up consisted in two axial LPs, a radial chain with lateral LP and ultrasonic wave transducers (see Figure 5.7). The calibration curves for the LPs are presented in Appendix C. Wavelength® Multi-Purpose Ultrasound Gel was placed on the ends of the samples in contact with the transducers. The direct and transmitted pulse was captured by an oscilloscope from Agilent Technologies. The data from the transmitted pulse was recorded in Excel files for post-processing. The P- and S- waves were captured prior to start of the UCS test and at different stages during the loading process.

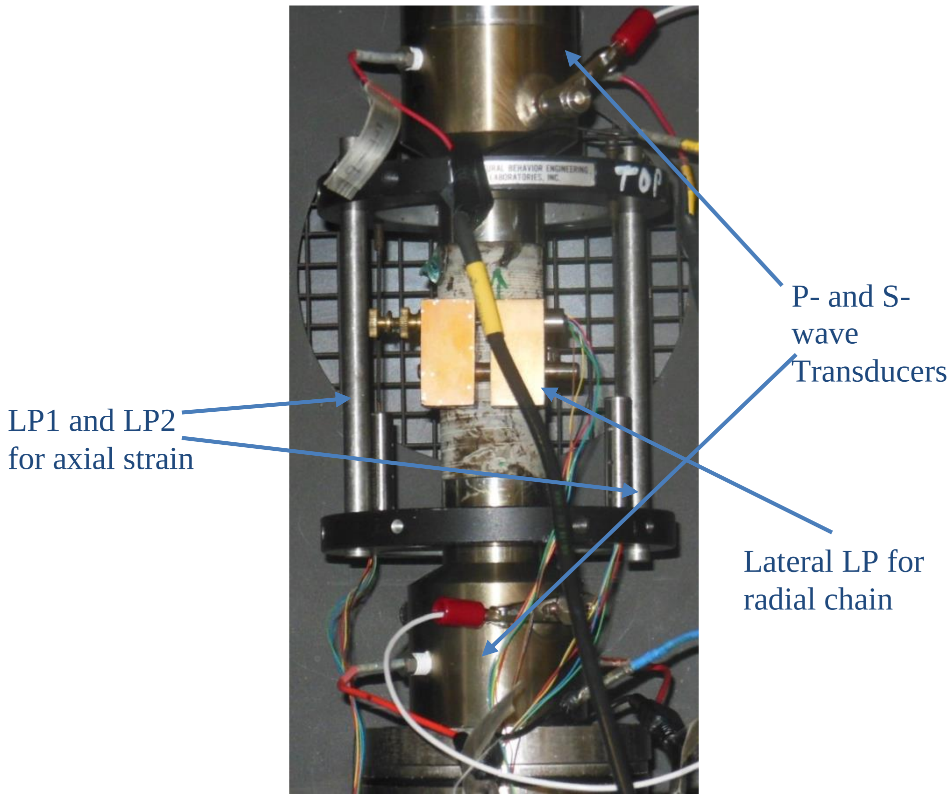


Figure 5.7: Experimental set-up for UCS test and sonic wave velocities

The propagation velocities and elastic parameters were calculated using Equations 3.14 to 3.16 (refer to ASTM D2845-08 and Abraseys and Hendron 1968 for further details). These equations assume an isotropic material, which was not the case for the vuggy samples. Thus, there was great uncertainty on the elastic properties calculated with this method for the high porosity samples. The results for the measured P- and S-wave velocities are presented in Table 5.3; the captured waveforms are presented in Appendix C.

Table 5. 3: Longitudinal (Vp) and transverse (Vs) waves for carbonate samples

Sample	V_p (m/s)	V_s (m/s)
S1	3960	2388
S2	4621	2303
S3	N/A	N/A
S4	1858	1054
S5	2192	1006
S6	3892	2228

The UCS data was analyzed based on ASTM D7012-10, and the basic equations used are presented below. An additional relationship was required for calculating the change in diameter based on the change in chord length (Δl) captured by the lateral LP and the radial chain (see Equation 5.7 obtained from SBEL 2000). This correction required measurement of the initial angle subtended within the ends of the chain (θ_o).

$$\sigma = \frac{P}{A} \qquad \textbf{Eq. 5.1}$$

$$\varepsilon_a = \frac{\Delta L}{L} \qquad \textbf{Eq. 5.2}$$

$$\varepsilon_l = \frac{\Delta D}{D} \qquad \textbf{Eq. 5.3}$$

$$\varepsilon_{vol} = \varepsilon_a + 2\varepsilon_l \qquad \textbf{Eq. 5.4}$$

$$\Delta D = {\Delta l}\Big/{\pi\left\{\left(1-\frac{\theta_o}{2\pi}\right)\cos\frac{\theta_o}{2}+\frac{1}{\pi}\sin\frac{\theta_o}{2}\right\}} \qquad \textbf{Eq. 5.5}$$

$$E = \frac{\Delta\sigma}{\Delta\varepsilon_a} \qquad \textbf{Eq. 5.6}$$

$$v = -\frac{E}{slope\ of\ lateral\ curve} \qquad \textbf{Eq. 5.7}$$

The change in sample length was taken as the average of two LP measurements, except during the tests where one of the LPs failed. The elastic modulus was calculated based on Method b in ASTM D7012-10, which uses the average modulus of the linear portion of the axial stress-strain curve. The slope of the lateral curve was taken within the same linear range as the axial curve. A summary of the static and dynamic parameters tested is presented in Table 5.4; the stress versus strain curves for the UCS tests are presented in Figure 5.8.

Table 5.4: Static and Dynamic parameters for vuggy carbonates tested

Sample	UCS (MPa)	E_s (GPa)	ν_s	$E_{d\ \sigma=0}$ (GPa)	$\nu_{d\ \sigma=0}$	$E_{d\ peak}$† (GPa)	$\nu_{d\ peak}$†
S1	44.26	8.45	0.171	35.39	0.214	46.98	0.094
S2	27.33	8.53	0.199	37.25	0.336	n/a	n/s
S3	12.96	2.04	0.102	n/a	n/a	n/a	n/a
S4	15.26	1.97	0.109	6.42	0.263	7.51	0.366
S5	15.02	4.89	0.170	6.93	0.367	7.12	0.353
S6	55.39	15.02	0.110	32.82	0.256	38.41	0.128

† Values obtained from maximum P- and S- velocities measured during the UCS test

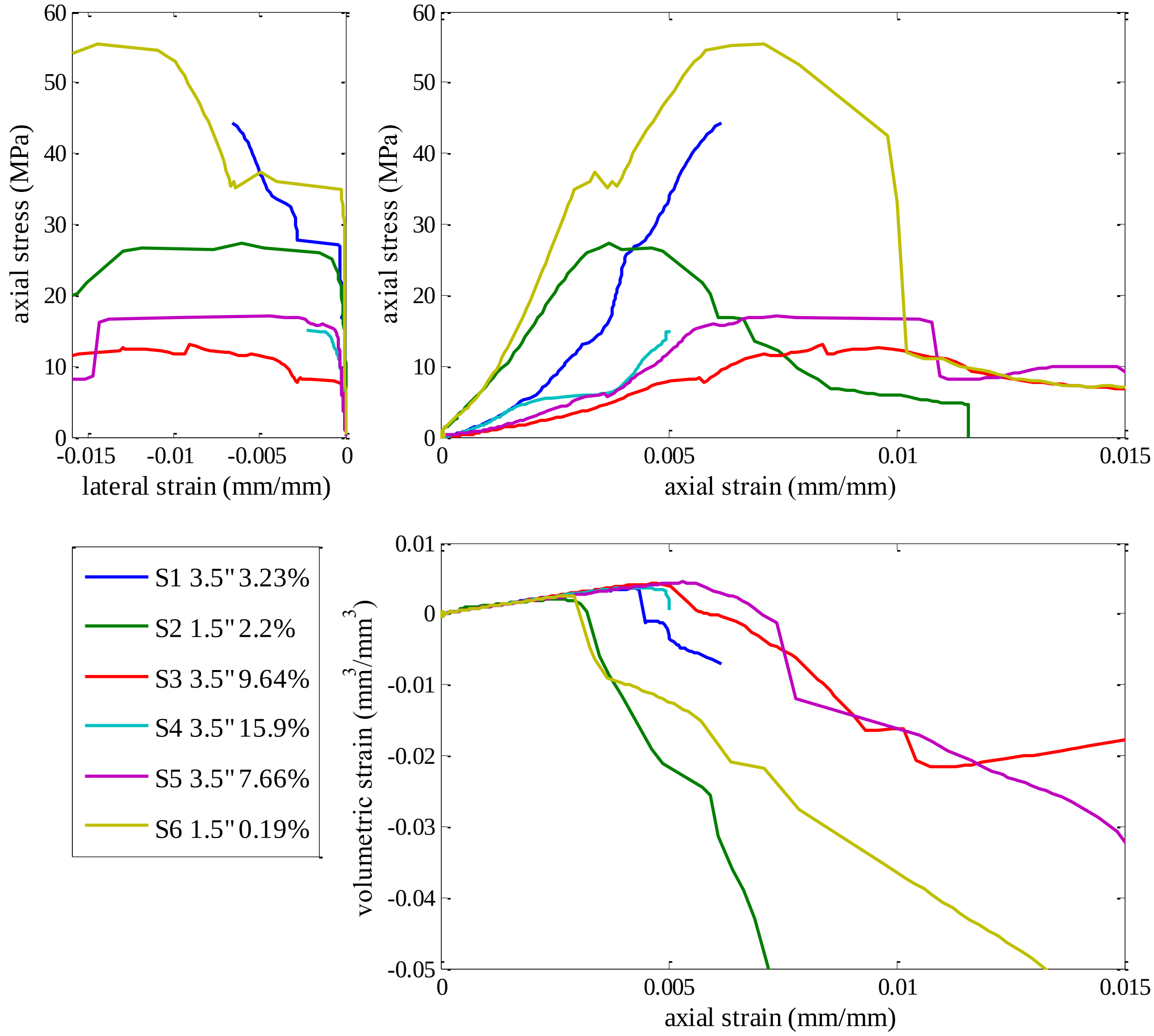


Figure 5 8: Axial stress vs lateral strain, Axial Stress vs axial strain, Volumetric strain vs axial strain

5.3.Result Analysis

The velocities measured are within the range for sedimentary rock velocities of 880 – 6150 m/s, but slightly below limestone range of 500 – 6150 m/s (Stagg 1968). The lower values could be a reflection of weaker properties due to weathering and complex diagenetic history of the Grosmont Formation. Sonic velocities were difficult to determine, in particular during low uniaxial stress. Additionally, the arrival of the shear wave velocity was obscured due to a persistent noise in the signal from the oscilloscope. Arrival times for S3 were not distinguishable at any stage of the test and the velocities could not be determined.

The difficulties with the sonic wave velocities were caused both by issues with the equipment and by the heterogeneity of the samples. Sample S1 had the lowest vuggy porosity and also presented the clearest arrival times. Accurate determination of P- and S-wave velocities will remain a further challenge as the heterogeneity of the medium increases. For future work, it is recommended that the ends of the samples are free from vugs (or as few vugs as possible) to ensure there is good contact between the sample and the crystals. This condition could not be followed in the current experiment due to limited number of samples and the high vuggy porosities studied.

The E_d obtained for the low vuggy porosity samples are lower but similar to the value estimated by Arseniuk et al. (2012) of 41.0 GPa. The E_d measured during loading of the samples (refer to Table 5.4 for $E_{d\ peak}$) is closer to the value presented by Arseniuk et al. (2012). The E_d was between 1.5 to 4.3 times higher than the measured Es. There was no clear trend between the vuggy porosity and the static to dynamic modulus ratio (Es/Ed); however, the original sample size was limited and further reduced by the problems encountered in determining arrival time for some samples.

There were several difficulties during uniaxial load testing which resulted in poor or incomplete data. One of the issues for the static test was failure of one or both of the LPs during shearing. The curves for S1 and S4 are terminated at peak stress (see Figure 5.8) due to failure of both LPs by that point. Stress and strain curves for S3, S5 and S6 show a localized drop in stress followed by an increase of stress. Some of these peaks also coincide with large changes in the lateral strain (e.g. S6 curve). It is hypothesized that this is caused due to the collapse of a void vug (producing the drop in stress) followed by an increase in stress once the vug's inner surfaces form a new contact. This type of behaviour and jagged stress-strain curves was observed in some of the idealized vug models discussed in Chapter 3 (also refer to Appendix A). Meanwhile, S5 showed some strain hardening after the post peak drop in stress, which could indicate that the failed rock was acting as a granular material post-peak. These observations further support the premise that these materials require discontinuum modelling that can handle contacts to break and form throughout the test.

The biggest challenge during UCS tests was the interpretation of the lateral strain; since the location of existing vugs and cracks affected the measurements obtained with the radial chain. In several cases the lateral strain measurements increased rapidly without significant increase in stress; this was probably cause by local shearing of cracks or vugs collapse within the zone of the radial chain. For further research, it is recommended that several lateral LPs are placed around the sample. This would allow capturing lateral changes at different points in the sample, and also provide some measure of strain heterogeneity caused by the vugs. Additionally, digital image correlation could be considered to track localized strains around the vug locations. This technique has been successfully used by Zinsmeister et al. (2013) for studies of local strains during testing.

The UCS value obtained for S6 is within the expected range for a limestone (Hoek and Brown 1980); however, it is below the average value of 90 MPa ± 37 MPa estimated by Arseniuk et al. (2012) for the Middle Grosmont C Unit. The UCS was found to further decrease with increasing vuggy porosity. The E_s also followed this trend. The υ_s calculated are at the lower limit of the typical range for limestones and dolomite of 0.1-0.35 (Gercek 2007). Unlike UCS and static modulus, there was no trend between the Poisson's ratio and the vuggy porosity. More laboratory data is required to confirm the trends observed, as well as additional lateral strain data information needs to be analyzed to determine the effect of vugs and dilation within the carbonate material.

The volumetric versus axial strain curves presented in Figure 5.8 appear to follow a similar pattern during the initial axial strains. The samples with lower vuggy porosity deviated first; however, a clear pattern could not be observed with the limited number of tests. The UCS and static modulus are plotted against the vuggy porosity calculated from the CT scans in Figures 5.9 and 5.10, respectively. The figures also include the UCS and modulus predictive curve obtained with PFC3D with synthetically generated vuggy samples in PFC3D (refer to Chapter 3). The reduction in UCS and modulus is much greater than anticipated by PFC3D, but it does appear to follow an exponential decrease with increasing porosity. However, the number of samples is limited and the conditions could not be standardized (sample size and length-to-diameter ratio) for all the samples, so the results may not be directly comparable. The following chapter (Chapter 6) summarizes the PFC3D simulation based on the models created from the CT scan data of the 3.5" samples and provides a more direct comparison between the laboratory results and PFC3D.

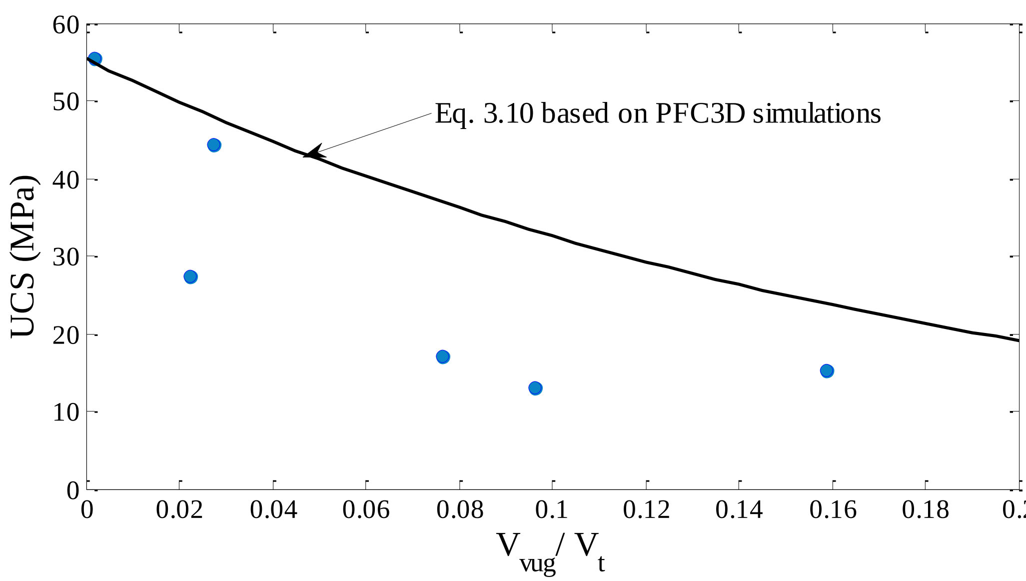


Figure 5.9: UCS versus vuggy porosity (V_{vug}/V_t) for carbonate samples tested, compared with the correlation developed in Chapter 3 based on PFC3D simulations with idealized vuggy samples

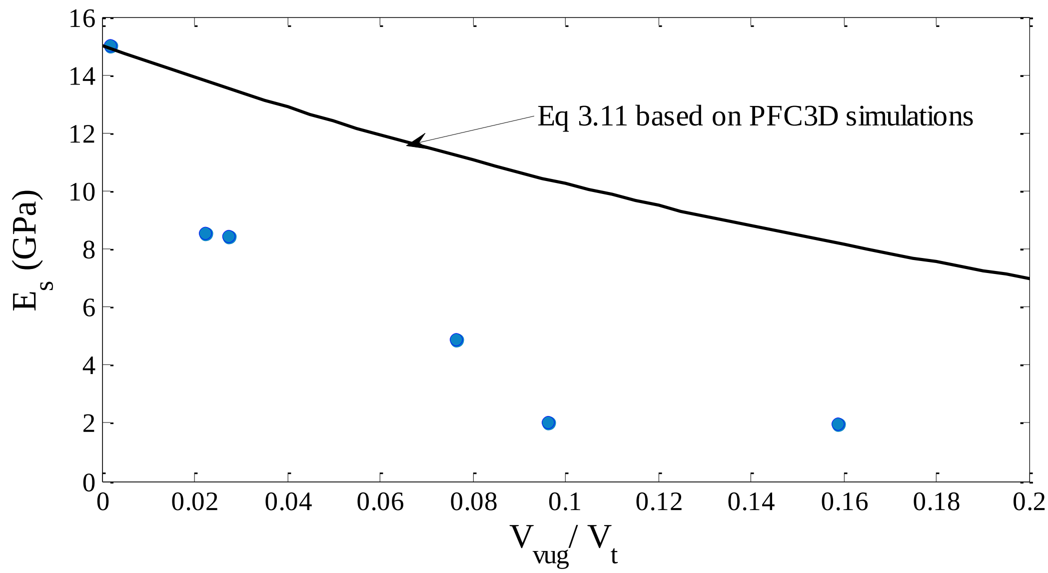


Figure 5.10: Static Young's Modulus (E_s) versus vuggy porosity (V_{vug}/V_t) for carbonate samples tested, compared with the correlation developed in Chapter 3 based on PFC3D simulations with idealized vuggy samples

CHAPTER 6: PFC3D CALIBRATION AND VALIDATION

6.1. Model Calibration

The UCS results for S6 (refer to Table 5.4) were used as the intact macro-properties to calibrate the PFC3D model. An initial calibration was attempted to model both static and dynamic laboratory tests. As discussed in Chapter 3, dynamic modulus is generally higher than static modulus. This could be modelled in the BPM by setting the ball modulus to a higher value than the parallel bond modulus; however, there was a limit on how different these two properties could be. It was found that to calibrate the BPM for a macro-behaviour with a dynamic modulus twice the value of the static modulus required a combination of micro-properties that resulted in erroneous ν_s. Setting the ball modulus significantly higher than the parallel bond modulus created problems during the static test simulations and produced an abnormally high ν_s (over 0.5 in some cases, refer to Table 6.1 for some examples). Although ν_d, E_d and E_s as well as UCS values could be calibrated, the lateral strain observed during static tests was unrealistic.

Table 6.1: Effect of changing the ball and parallel bond modulus on static and dynamic macro-properties

Material	E_d (GPa)	ν_d	UCS (MPa)	E_s (GPa)	ν_s	E_d/E_s	E_{ball} (GPa)	E_{pbond} (GPa)	E_{ball}/E_{pbond}
c1_s550	31.26	0.248	52.57	33.48	0.244	0.934	31.50	37.80	0.833
c1_s580	31.83	0.237	62.48	32.67	0.275	0.974	36.50	32.85	1.111
c1_sp4	36.99	0.256	57.57	34.08	0.344	1.085	60.09	33.57	1.790
c1_sp5	33.44	0.308	56.24	31.04	0.404	1.077	60.67	26.47	2.292
c1_sp8	42.89	0.239	55.28	47.89	0.227	0.896	37.37	62.62	0.597
c1_sp11	35.22	0.300	58.55	32.64	0.384	1.079	61.34	29.18	2.102
c1_sp12	34.38	0.311	60.90	29.46	0.417	1.167	61.78	23.86	2.590
c1_sp15	42.63	0.252	57.85	45.65	0.243	0.934	37.75	58.36	0.647
c1_sp22	40.15	0.276	54.23	37.08	0.347	1.083	61.21	38.19	1.603
c1_sp24	35.82	0.284	56.82	31.35	0.402	1.142	62.13	27.26	2.279
c1_sp31	35.24	0.288	55.28	32.65	0.342	1.079	60.51	31.98	1.892
c1_sp33	34.18	0.298	59.76	28.72	0.419	1.190	61.36	23.12	2.654
c1_sp43	15.14	0.234	52.68	15.58	0.212	0.972	15.00	12.50	1.200
c1_sp45	16.67	0.224	62.68	16.20	0.243	1.029	17.00	12.50	1.360
c1_sp46	11.12	0.217	52.32	11.32	0.219	0.983	10.00	10.00	1.0
c1_sp53	19.57	0.213	56.74	19.82	0.240	0.988	21.00	13.00	1.615
c1_sp55	15.87	0.237	53.32	16.81	0.188	0.944	15.00	12.50	1.2
c1_sp58	11.67	0.210	55.00	10.62	0.202	1.099	10.00	10.00	1.0
S6_sc5	20.90	0.309	63.93	16.59	0.537	1.260	45.00	5.00	9.0
S6_sc7	28.45	0.304	66.44	17.84	0.633	1.595	60.00	5.00	12.0
S6_sc21	31.79	0.309	51.96	15.41	0.686	2.063	60.00	3.00	20.0
S6_sc30	43.51	0.285	45.47	21.42	0.672	2.031	62.00	2.80	22.1
S6_sc33	33.03	0.279	42.67	18.94	0.616	1.745	50.00	2.80	17.9
S6_sc36	31.38	0.295	51.14	14.39	0.673	2.180	50.00	3.00	16.7
S6_sc54	50.21	0.205	44.42	55.88	0.136	0.899	50.00	20.00	2.5
S6_sc61	24.96	0.256	53.49	12.22	0.552	2.042	35.00	5.00	7.0
S6_sc63	28.55	0.243	50.50	17.87	0.416	1.597	35.00	10.00	3.500
S6_sc64	29.63	0.214	48.46	20.04	0.378	1.478	35.00	12.50	2.800
S6_sc65	32.78	0.255	52.14	18.85	0.440	1.739	40.00	10.00	4.000
S6_sc66	36.31	0.266	54.71	20.55	0.445	1.767	45.00	10.00	4.500

The difficulties encountered suggest that it is not possible to calibrate both dynamic and static behaviour using the traditional BPM. Holt et al. (2005) showed that it was possible to obtain dynamic behaviours using a Hertz contact model; however their study did not attempt to use the same calibration to model a static behaviour. Therefore, calibration of the PFC3D material may have to be dependent on the application (dynamic vs. static) and the contact model would have to vary accordingly.

For the model validation presented in this chapter, only calibration to the static laboratory test was conducted, since these laboratory results provided more information than the dynamic tests. PFC3D BPM was calibrated for two resolutions with minimum ball radius of 0.9 mm and 0.7 mm. The comparison between the calibrated behaviour and the laboratory results is presented in Table 6.2. The behaviour of the BPM is calculated based on different realization with equal properties but varying seed numbers. The macro-properties are presented as an average plus/minus a standard deviation. As noted in Chapter 4, at higher resolutions fewer realizations are required since the variations is reduced by having more particles along the length scale. The calibrated micro-parameters for the two different resolutions are presented in Table 6.3; note that changing the resolution requires a slight change in the micro-properties to maintain similar macroscopic behaviour.

Table 6.2: Comparison of static results between laboratory tests and PFC3D

Test	Nº of Realizations	UCS (MPa)	E_s (GPa)	ν_s
Sample S6		55.39	15.02	0.110
G1 BPM (Res ~ 15)	10	55.36 ± 1.59 diff = 0.05%	15.06 ± 0.11 diff = -0.27%	0.110 ±0.005 diff = -0.25%
G2 BPM (Res ~ 20)	5	55.43 ± 1.11 diff = -0.07%	15.04 ± 0.04 diff = -0.15%	0.110 ±0.005 diff = -0.16%

Table 6.3: Calibrated micro-properties to S6 for two different resolutions

Parameter	G1	G2
Resolution (average nº of particles/diameter)	15.8	20.3
Minimum Radius (m)	$9\text{x}10^{-4}$	$7\text{x}10^{-4}$
R_{max}/R_{min}	1.66	1.66
Ball Density (kg/m^3)	2630	2630
Ball Modulus (GPa)	10.2	10.1
Ball Stiffness Ratio	0.670	0.671
Friction Coefficient	0.5	0.5
P-bond Modulus (GPa)	9.70	9.60
P-bond stiffness ratio	0.695	0.710
P-bond normal Strength (MPa)	55.4 ± 11.1	54.5 ± 10.9
P-bond shear strength (MPa)	65.8 ± 13.2	62.2 ± 12.4

These micro-properties were used as intact materials for samples S1, S3, S4 and S5. With the larger size of these samples the average number of particles per diameter was approximately 36.5 and 47 for G1 and G2 materials, respectively. This resulted in very large number of particles per sample. Once the samples were generated with the intact material, the vugs were superimposed using the procedure described in Chapters 3 and 4.

6.2. Vuggy Samples based on CT Data

A large number of void voxels were identified, especially for samples S3. Many of these voxels were a single instance and had no connection to other void voxels. This increased the processing time required to insert the vugs into the samples. Several trials were carried out with minimum voxel per vug varied from three to ten. A threshold below three voxel per vug would be too small to be represented accurately with the ball radius used for the PFC3D models. The minimum equivalent spherical radius for different voxel thresholds is shown in Table 6.4.

Table 6.4: Minimum spherical radius for varying threshold of voxels per vugs

Minimum Number of Voxels	Equivalent spherical radius	Number of Vugs identified S1	Number of Vugs identified S3	Number of Vugs identified S4	Number of Vugs identified S5
3	0.537	3387	2277	703	803
4	0.591	2712	1776	575	621
5	0.636	2258	1446	467	507
6	0.676	1988	1244	396	434
7	0.712	1768	1077	352	361
8	0.744	1584	962	322	326
9	0.774	1427	888	288	293
10	0.802	1292	810	259	275

The variations of the calculated vuggy porosity with increasing voxel threshold are presented in Table 6.5 for samples S4 and S5; each result presented is the average of ten sample realizations conducted with different seed numbers. Setting a higher threshold number of voxels reduces the number of inputs into PFC3D and speeds up the sample generation process. This threshold is more important for the voxel method, but can also reduce computing time for equivalent ellipsoid or sphere. As shown in Table 6.5, the PFC3D porosity can be both above and below the CT scan porosity. A conservative approach of using three voxels per vug was used for this preliminary validation. Table 6.6 presents the average and standard deviation of the vuggy porosity obtain for the different PFC3D sample representations

Table 6.5: Variations in PFC3D sample porosity for the three proposed methods. Results are for samples with minimum radius of 0.9 mm

Minimum Voxel per Vug	S4 ellipsoid method	S4 sphere method	S4 voxel method	S4 CT-Scan Vuggy Porosity	S5 ellipsoid method	S5 sphere method	S5 Voxel method	S5 CT-Scan Vuggy Porosity
3	73.32%	16.05%	15.98%	15.89%	35.71%	7.64%	7.70%	7.66%
4	73.32%	16.04%	15.94%		35.71%	7.63%	7.62%	
5	73.32%	16.02%	15.92%		35.70%	7.61%	7.61%	
6	73.32%	16.03%	15.91%		35.70%	7.60%	7.60%	
7	73.32%	16.01%	15.90%		35.61%	7.59%	7.59%	
8	73.31%	16.01%	15.89%		35.69%	7.58%	7.58%	
9	73.31%	16.00%	15.89%		35.68%	7.57%	7.57%	
10	73.31%	16.00%	15.88%		35.68%	7.56%	7.57%	

Table 6.6: Vuggy porosity for PFC3D samples with different methods compared to the CT scan calculated value

Sample	Equivalent Ellipsoid	Equivalent Sphere	Voxel	CT scan
S1	6.40% ± 0.04%	2.61% ± 0.04%	2. 72% ± 0.05%	2.73%
S3	45.86% ± 0.04%	9.43% ± 0.04%	9.76% ± 0.05%	9.64%
S4	73.32% ± 0.04%	16.05% ± 0.04%	15.98% ± 0.09%	15.89%
S5	35.71% ± 0.05%	7.64% ± 0.03%	7.70% ± 0.06%	7.66%

The samples with voxel representation took between 12 to 25 hrs to generate for each realization (based on a 24 GB RAM computer with 3.2GHz processor); the higher porosity models (such as sample S4) required the longest time. In contrast, samples generated using the ellipsoid and spherical method took between 10 and 30 minutes. The samples generated using equivalent ellipsoids had excessive porosity that was caused by large semi-axis calculated using the image processing procedure. These large axes resulted from misinterpretation of several vugs that were slightly connected but identified as a large single vug. The procedure would calculate semi-axis for an ellipsoid that would encompass all the void voxels, but at the same time include several additional voxels (see Figure 6.1).

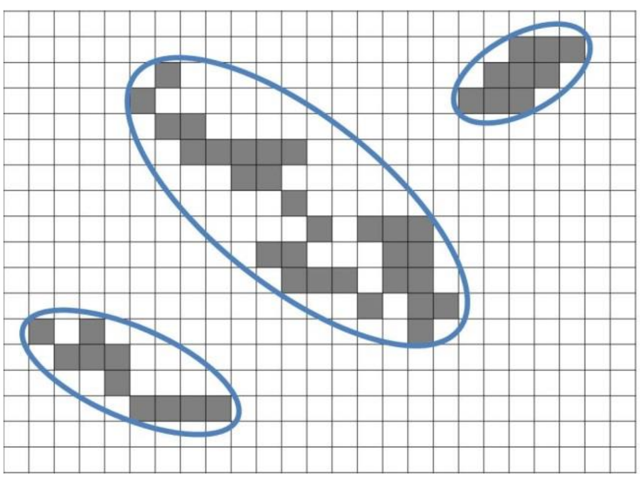


Figure 6.1: Issues with the equivalent ellipsoid representation

In future research, it is proposed that further image processing may be required to better define the vugs in order to create improved equivalent ellipsoid samples. For the current validation, only the voxel and the spherical methods were compared. The results contrast samples in which vug volume is represented against samples in which a complete shape representation is made. As shown in Tables 6.5 and 6.6, the vuggy porosity between samples generated using the spherical and voxel method are very similar. Thus, it provides a direct comparison to measure the importance of shape and volume of voids for the geomechanical behaviour.

6.3. Virtual PFC3D Testing

Each laboratory sample was simulated with spherical and voxel methods of vug representation using ten realizations, for a total of twenty simulations per sample. The stress vs strain curves are presented in Figures 6.2 to 6.5 for both the equivalent sphere (a) and voxel (b) representation. The spherical vug samples show little variation between realizations in the stress strain curves; some divergence starts at approximately 80% of the peak stress value. The stress strain curve for the spherical representation was very smooth and did not show the jagged peaks or plateau observed during the laboratory testing. The samples with voxel representation had more divergence between the realizations and there appears to be more vug crushing near 90% of the peak value. However, the detailed representation achieved by the voxel method was still not able to accurately represent the laboratory behaviour.

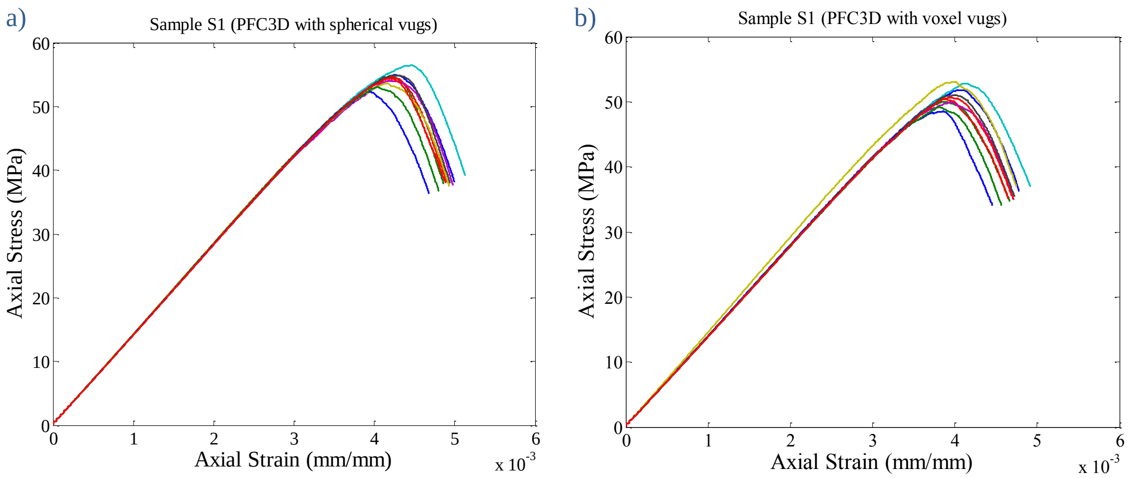


Figure 6.2: PFC3D results for realizations of Sample S1 (material G1 resolution 36.5 particles/diameter): a) equivalent sphere method, b) voxel method.

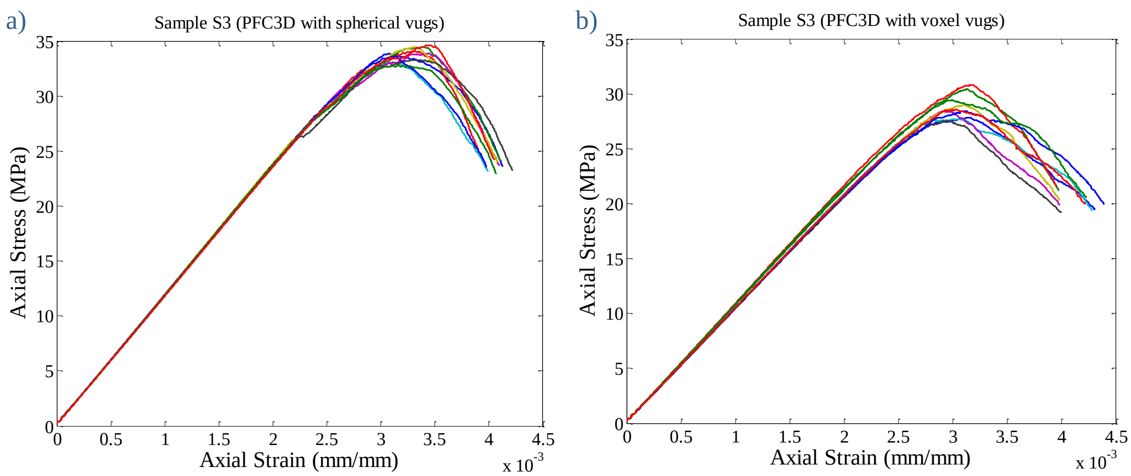


Figure 6.3: PFC3D results for realizations of Sample S3 (material G1 resolution 36.5 particles/diameter): a) equivalent sphere method, b) voxel method.

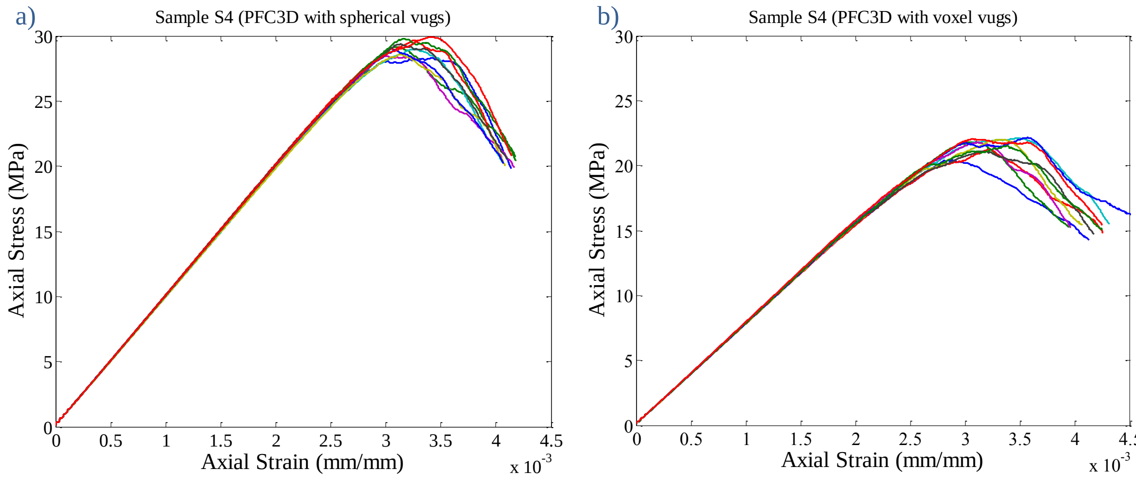


Figure 6.4: PFC3D results for realizations of Sample S4 (material G1 resolution 36.5 particles/diameter): a) equivalent sphere method, b) voxel method.

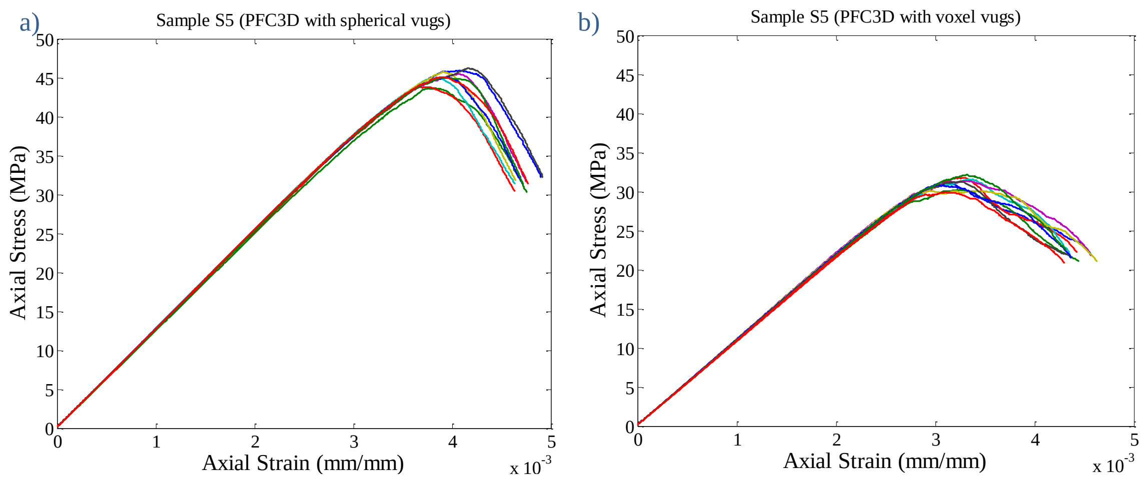


Figure 6.5: PFC3D results for realizations of Sample S5 (material G1 resolution 36.5 particles/diameter): a) equivalent sphere method, b) voxel method.

6.4. Results Comparison and Model Evaluation

Tables 6.7, 6.8 and 6.9 compare the laboratory test with the PFC3D simulations for the UCS, E_s and υ_s, respectively. The results presented in the tables are the mean (m) and standard deviation (s) of the realizations performed. The spherical vug method shows some reduction in strength and stiffness but the simulations results have much higher values than the laboratory tests. The voxel representation has a more pronounced decrease in strength, yet not sufficient to match the laboratory results. The stiffness was relatively similar between the equivalent sphere and voxel methods; S4 voxel simulations even presented an increase in the E_s from the spherical models.

The υ_s from all the PFC3D simulations were not a good match with the laboratory results (see Table 6.9). However, this could be accounted for by problems with the lateral strain measurements during the laboratory tests, which may have made the laboratory results not representative. The PFC3D simulations consistently over predicted the value of the static Young's Modulus, regardless of the method. The difference in the stiffness of the samples is especially evident in the graphical comparison of the stress versus strain curves shown in Figure 6.6; these figures show the laboratory curves plotted with one of the PFC3D realizations for the voxel and spherical methods.

Table 6.7: Comparison of UCS results between laboratory tests and PFC3D models

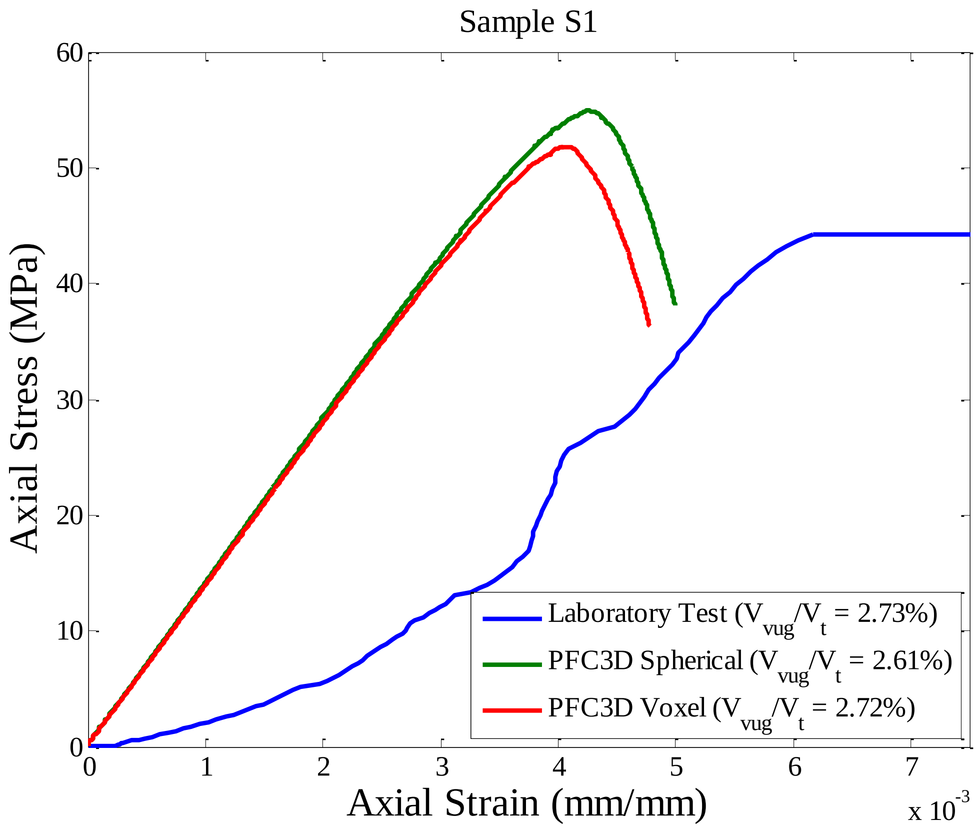
Sample	Vuggy Porosity	Laboratory UCS (MPa)	PFC3D Spherical Vugs UCS (MPa)		PFC3D Voxel Vugs UCS (MPa)	
			m	s	m	s
S1	2.73%	44.26	54.99	0.80	50.72	1.26
S3	9.64%	12.96	32.90	0.55	28.74	0.92
S4	15.89%	15.26	27.24	0.51	21.83	0.54
S5	7.66%	15.02	44.70	0.77	30.90	0.69

Table 6.8: Comparison of Young's Modulus (Es) results between laboratory tests and PFC3D models

Sample	Vuggy Porosity	Laboratory E_s (GPa)	PFC3D Spherical Vugs E_s (GPa)		PFC3D Voxel Vugs E_s (GPa)	
			m	s	m	s
S1	2.73%	8.45	14.13	0.08	13.90	0.16
S3	9.64%	2.04	10.82	0.13	10.49	0.24
S4	15.89%	1.97	8.19	0.04	8.29	0.02
S5	7.66%	4.89	12.29	0.08	11.07	0.10

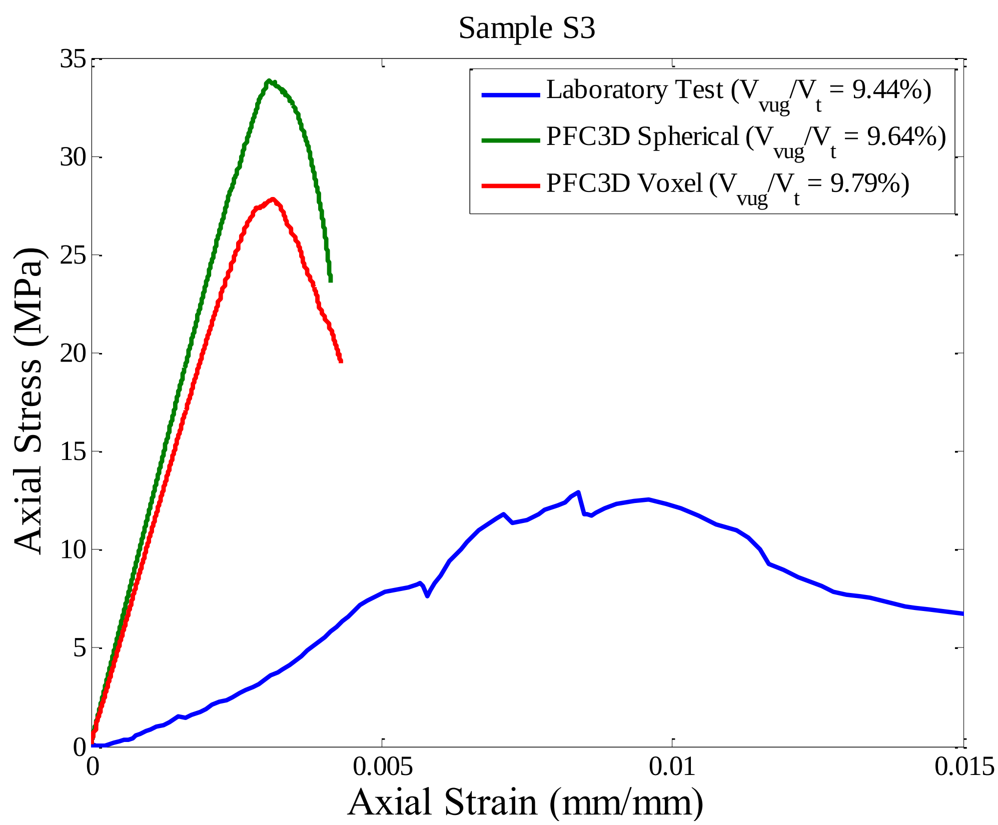
Table 6.9: Comparison of static Poisson's ratio (vs) results between laboratory tests and PFC3D models

Sample	Vuggy Porosity	Laboratory v_s	PFC3D Spherical Vugs v_s		PFC3D Voxel Vugs v_s	
			m	s	m	s
S1	2.73%	0.171	0.122	0.004	0.122	0.003
S3	9.64%	0.102	0.198	0.003	0.143	0.007
S4	15.89%	0.109	0.351	0.004	0.151	0.002
S5	7.66%	0.170	0.104	0.028	0.134	0.004



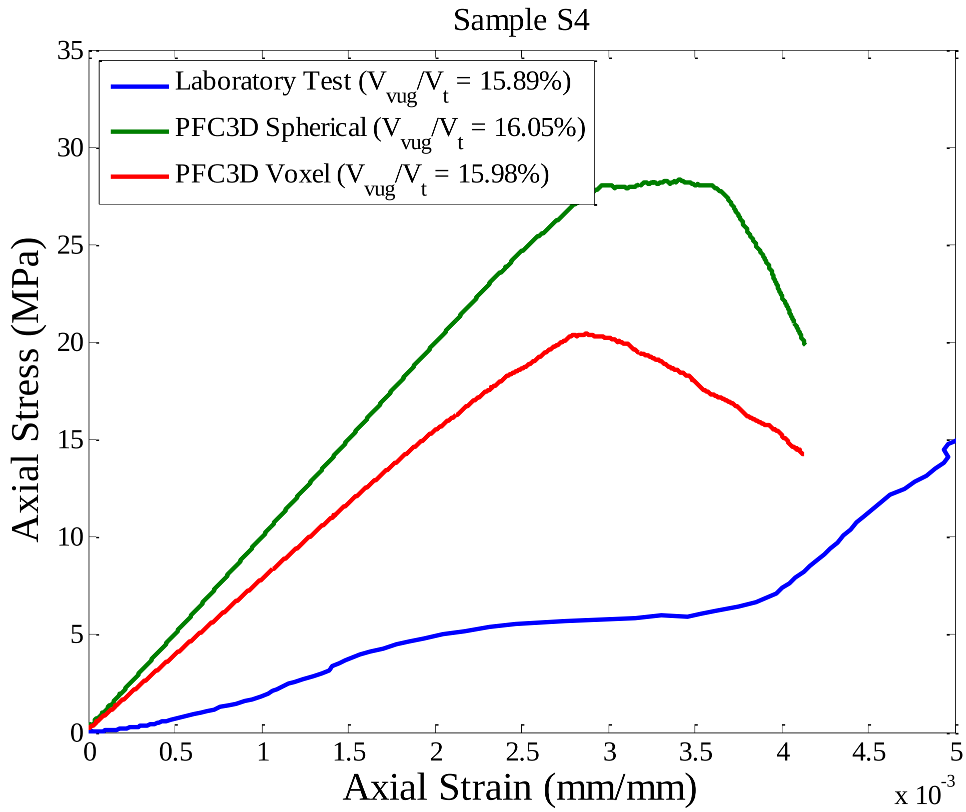
Sample S1
Axial Stress (MPa)
Axial Strain (mm/mm)
x 10^-3
0
10
20
30
40
50
60
0
1
2
3
4
5
6
7
Laboratory Test (Vvug/Vt = 2.73%)
PFC3D Spherical (Vvug/Vt = 2.61%)
PFC3D Voxel (Vvug/Vt = 2.72%)

a)

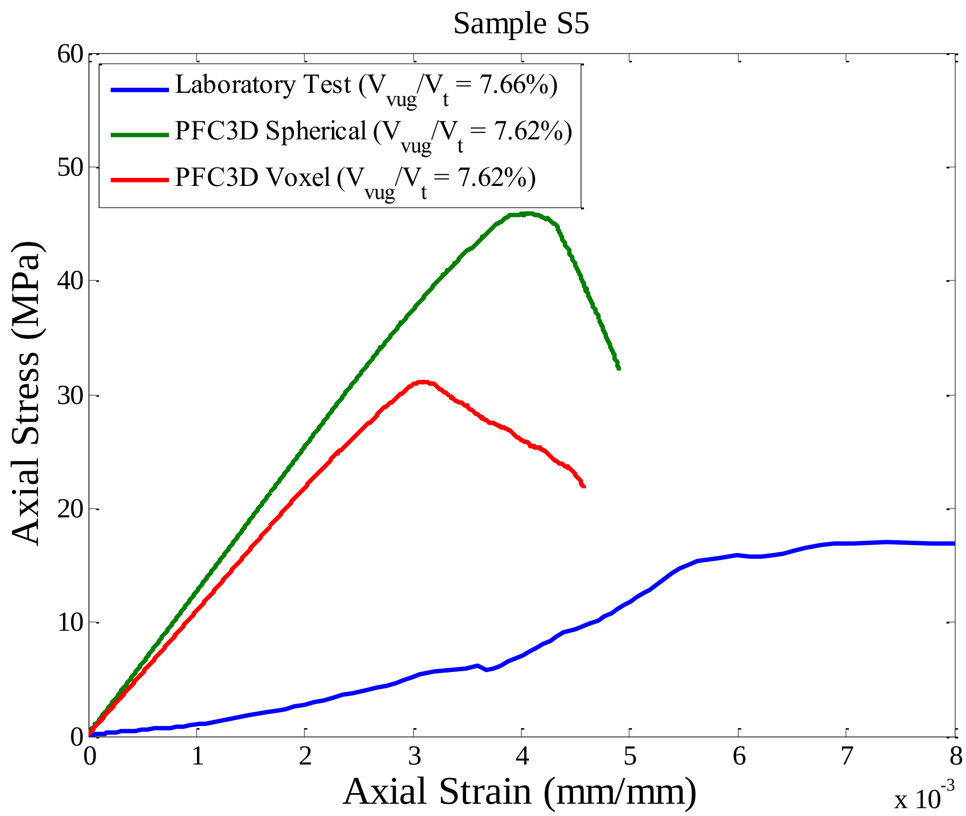


Sample S3
Axial Stress (MPa)
Axial Strain (mm/mm)
0
5
10
15
20
25
30
35
0
0.005
0.01
0.015
Laboratory Test (Vvug/Vt = 9.44%)
PFC3D Spherical (Vvug/Vt = 9.64%)
PFC3D Voxel (Vvug/Vt = 9.79%)

b)



c)



d)

Figure 6.6: Comparison between laboratory and PFC3D simulations: a) Sample S1, b) Sample S3, c) Sample S4 and d) Sample S5

These results show that the volume of vugs is a factor in the strength and stiffness reduction, but that the shape and location of the vug has an effect in the behaviour. The effect of the vugs is also observed in the stress vs strain curves. For example in Figure 6.6d, the voxel and the spherical sample have the same vuggy porosity (7.62%); but, the behaviour of the specimens is quite different. Although it was not able to replicate the laboratory curve, the voxel method provided a better representation. One of the main differences observed between the spherical and voxel representation was the change in failure mode of the parallel bonds. PFC3D is able to record whether a failure of the parallel bonds is caused by exceeding the normal bond strength (tensile failure) or the shear bond strength (shear failure). The relative percentage of tensile bond failures was higher for the voxel method than for the samples with spherical vugs (refer to Table 6.10). This is an indication that the shape of the vugs causes changes in the stress distribution within the sample, leading to more tensile failures.

Table 6.10: Relative percentages of tensile and shear failure between particle bonds

Sample	Equivalent Sphere Method			Voxel Method		
	Average Peak Crack #	% Tension Failure	% Shear Failure	Average Peak Crack #	% Tension Failure	% Shear Failure
S1	5542	23%	77%	4584	27%	73%
S3	2730	39%	61%	1985	46%	54%
S4	2019	33%	67%	1883	52%	48%
S5	3850	29%	71%	2229	46%	54%

The true geometric representation becomes more important if features like vug collapse prior to peak stress need to be represented. This can be shown in Figure 6.7 where some of the vug collapse observed during the laboratory test was captured in PFC3D. Note that this vug collapse behaviour was only observed in the stress-strain data captured by the gauge ball measurements, in which the strains are computed based on the movement of gauge particles within the sample. The stress-strain data obtained through PFC3D's wall measurements (shown in Figures 6.2 to 6.6) tended to be smoother because the values are averaged over a larger area.

The initial crack-closure stage of the laboratory test was not accurately represented in PFC3D. As was discussed in Chapters 2 and 3, vuggy PFC3D samples do not show the initial non-linearity observed in soft rock. All stress vs strain curves (see Figures 6.2 to 6.7) start with a linear relationship regardless of the type of vug representation. Placement of voids within the PFC3D material does not model this soft rock behaviour, which had been suggested by Cho et al. (2007). The initial non-linearity is attributed to the closure of small cracks or flaws within the rock; thus modelling this behaviour may require superimposing a discrete fracture network with varying apertures and stiffness values.

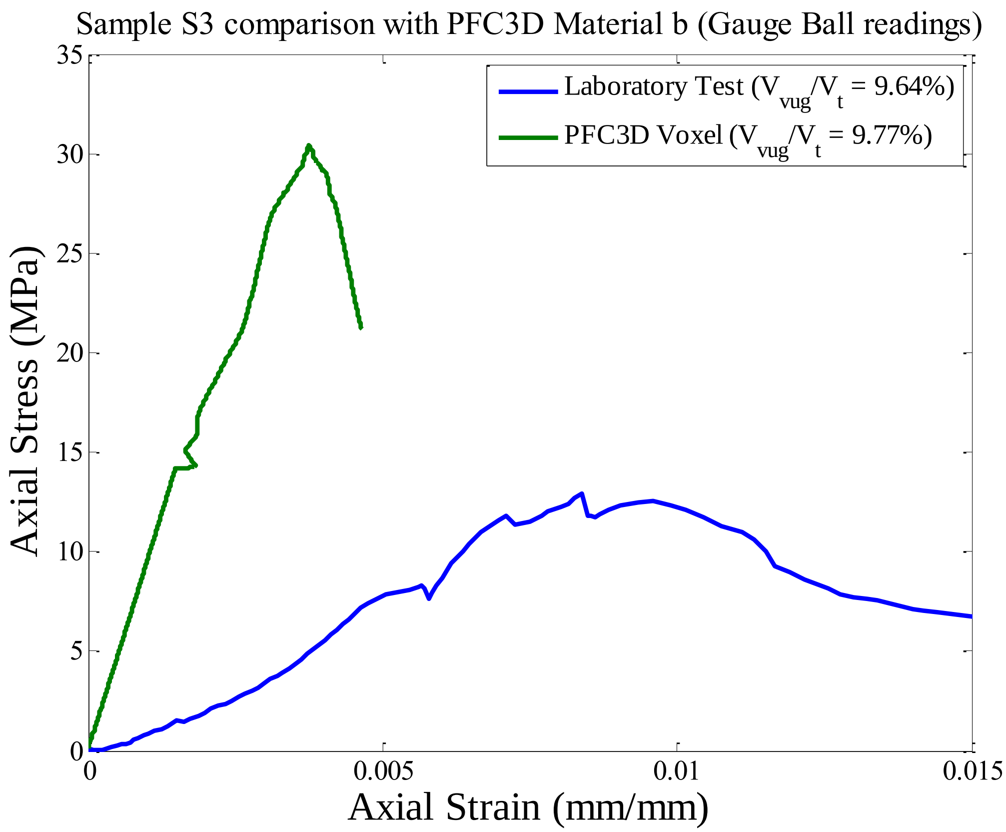


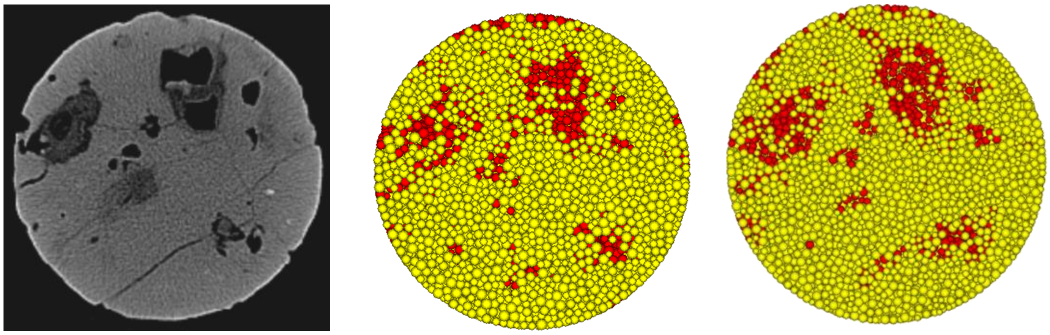
Figure 6.7: Comparison between laboratory sample S3 and the PFC3D voxel vugs sample, based on gauge ball measurements.

6.4.1. Effect of Increasing Resolution

The effect of changing the particle resolution was briefly tested by creating simulations of Sample S3 with the calibrated material G2; refer to Table 6.2 and 6.3 for G2 material static test results and micro-properties, respectively. The resolution increased from 36.5 to 47 particles across the diameter; and only five realizations were completed. The average vuggy porosity for the higher resolution samples was 9.74%, which is close to the calculated CT vuggy porosity of 9.64% and similar to the realizations for material G1 (9.76%). Figure 6.8 shows a comparison of the vug representation between different resolutions. The results of the UCS simulations are summarized in Table 6.11; the stress-strain curves are shown in Figure 6.9, which compares the lab samples and the PFC3D simulations at different resolutions. As can be observed in Figure 6.9, there was a slight decrease in both the strength and stiffness in the higher resolution samples but still not comparable to the laboratory results.

Table 6.11: Effect of changing the resolution on PFC3D samples

Parameter	Laboratory Test	PFC3D G1 voxel (res 36.5/diameter)	PFC3D G2 voxel (res 47/diameter)
UCS (MPa)	12.96	28.74 ± 0.92	26.75 ± 0.75
E_s (GPa)	2.04	10.49 ± 0.24	10.23 ± 0.21
v_s	0.102	0.143 ± 0.007	0.146 ± 0.007



a) b) c)

Figure 6. 8: Comparison through mid-plane of sample S3: a) CT scan slice, b) PFC3D voxel vugs with 36.5 particles/diameter, c) PFC3D voxel vugs with 47.0 particles/diameter.

The strength of the samples was not greatly affected and over predicted the strength observed from the laboratory experiments (refer to Figure 6.9). However, one of the improvements observed in the higher resolution samples was better representation of the collapse of discontinuities within the stress-strain curves. Figure 6.10 shows some examples of the vug collapse events captured through the gauge ball measurements.

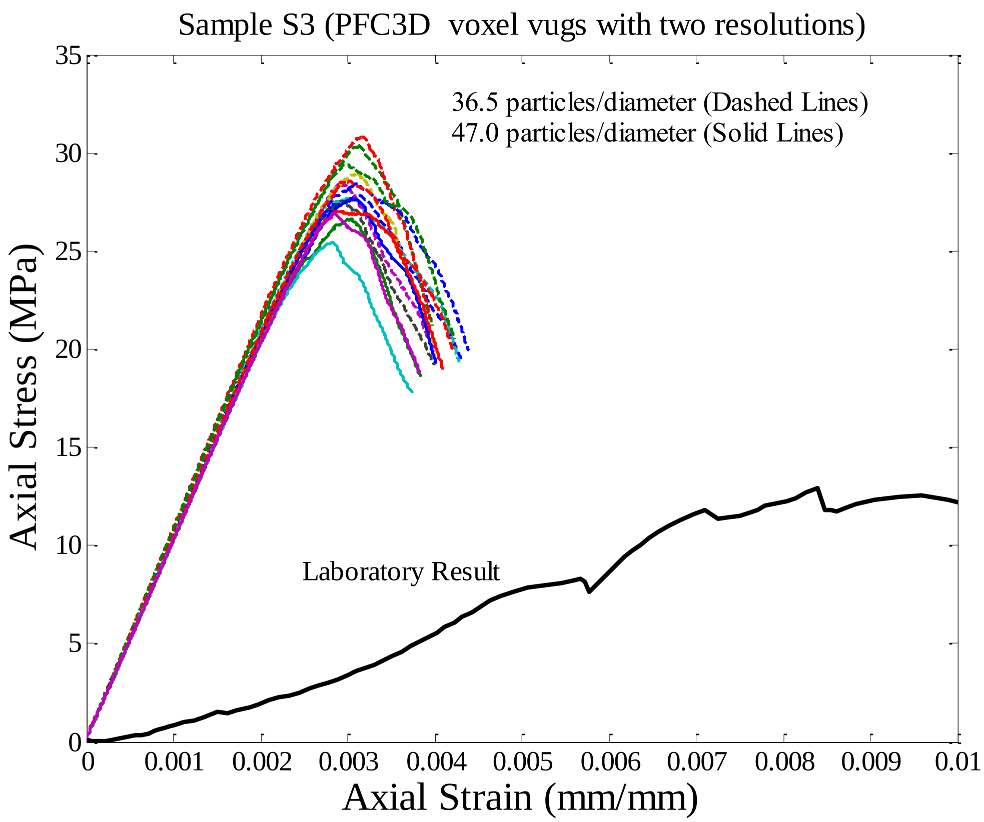


Figure 6.9: Stress vs strain curve for PFC3D samples of different resolution and the laboratory tests.

PFC3D appears to be representing the geometric effect of the vugs; however, the behaviour of the matrix or "intact" exceeds what was observed in the laboratory samples.

The percentage of tensile cracks was approximately 44% for G2 material, which is a similar to the value obtained for the lower resolution G1 material (refer to Table 6.10). The higher resolution did not increase the amount of tensile failure, which could indicate that material G1 was sufficient in representing the change in stresses caused by the geometry of the vugs. The limitations in the PFC3D model appear to be dominated by the matrix behaviour and the high strength assigned to the parallel bonds.

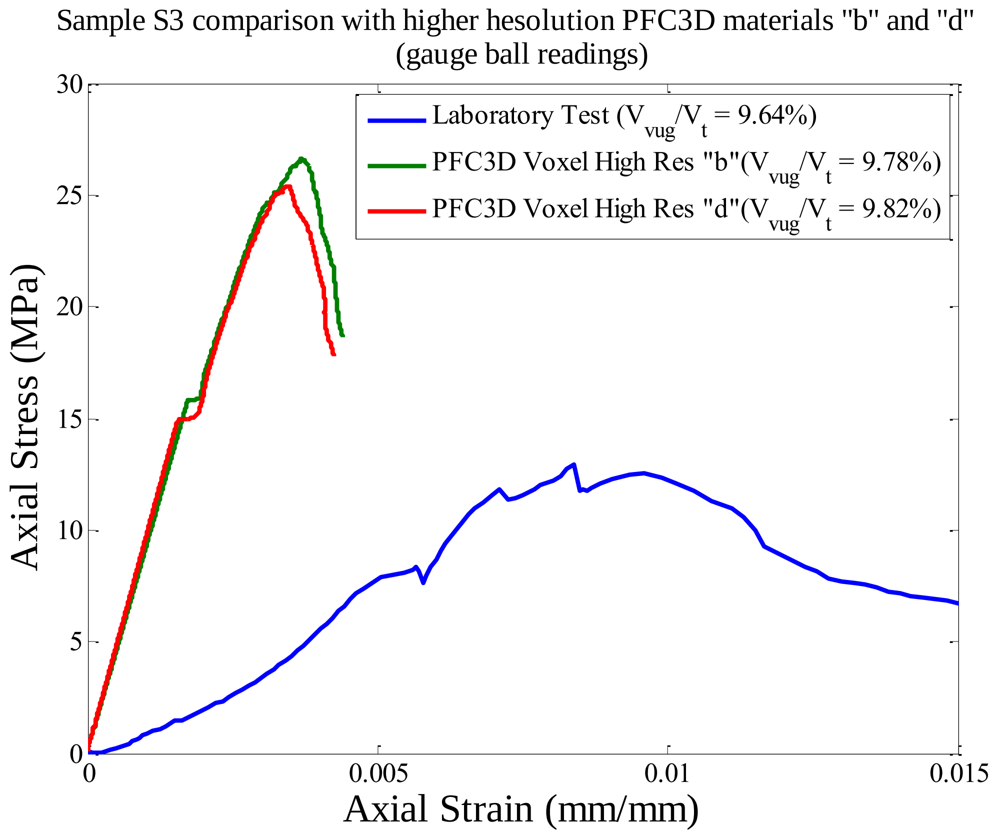


Figure 6.10: Vug collapse observed for higher resolution S3 samples

6.4.2. Issues with the spherical method

The connectivity routine used to identify vugs in Matlab groups voxels that may be slightly connected into the same vug (i.e. two vugs connected at an edge) and would thus be considered a single vug. In samples with high porosity, the interconnection between vugs results in the identification of one large single vug with a centroid near the middle of the sample. The vug is located near the centre because it is formed from voids connected throughout the sample, and has a centroid similar to the sample itself. Figures 6.11 and 6.12 show examples of the problems encountered using the spherical method in high vuggy porosity samples.

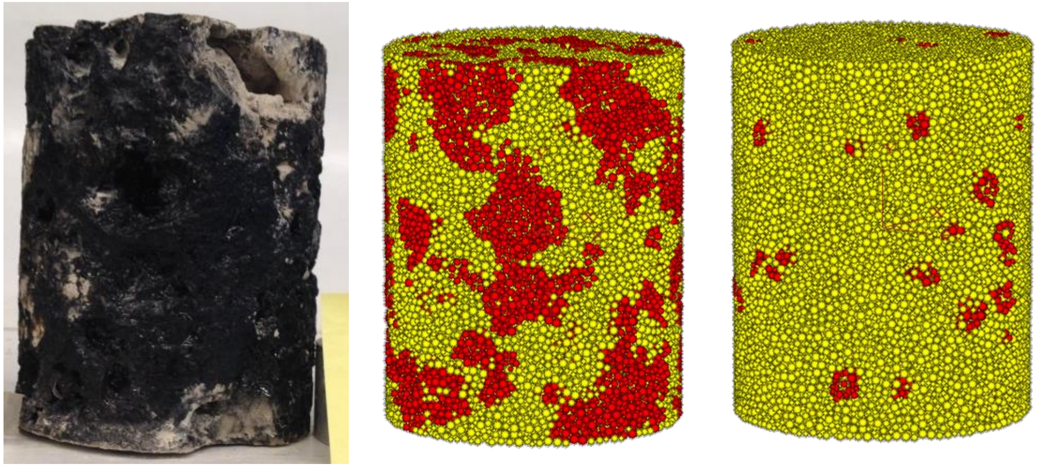


Figure 6.11: Comparison between PFC3D voxel (middle) and equivalent sphere (right) models for Sample S4

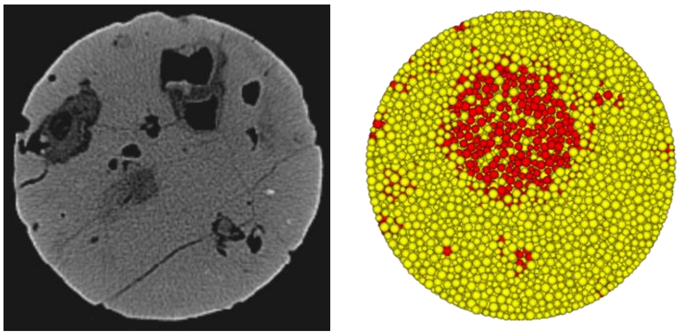


Figure 6.12: Comparison between CT scan slice of S3 (left) with cut-plane through PFC3D sample generated with equivalent sphere method (right)

In Sample S4, shown in Figure 6.11, the vugs at the edge of the sample are not represented in the spherical method because their volume is included in a large middle vug. Thus, in reproducing similar sample with the spherical method, the vuggy porosity volume is represented but it is mostly concentrated in a single vug in the centre of the specimen (see Figure 6.12 for S3 example). This greatly reduces the effect of the vugs that are at the boundaries, and could explain the higher values of the UCS and modulus results for the spherical method when compared to the voxel method. At lower porosity, with less interconnection between the vugs, the spherical methods would produce results more similar to the voxel method (for example Sample S1 in Figures 6.1 and 6.5a). Future research could focus on refining the delineation of the vugs during the image processing to improve the spherical method representation.

6.4.3. Fracture Patterns

The crack locations and final fracture patterns of the laboratory sample (when compared with the PFC3D realizations) and PFC3D samples were compared. The damage pattern for the voxel methods was consistent with the damage observed in the laboratory samples. Some examples are presented in Figure 6.13, with one area highlighted for comparison. The location of the failed parallel bonds varied slightly between the different realizations. However, the simulations tended to show the majority of the damage in the same area, which also was consistent with the damage observed in the laboratory samples. The voxel method accurately represented the increased likelihood of failure near areas with large void volume.

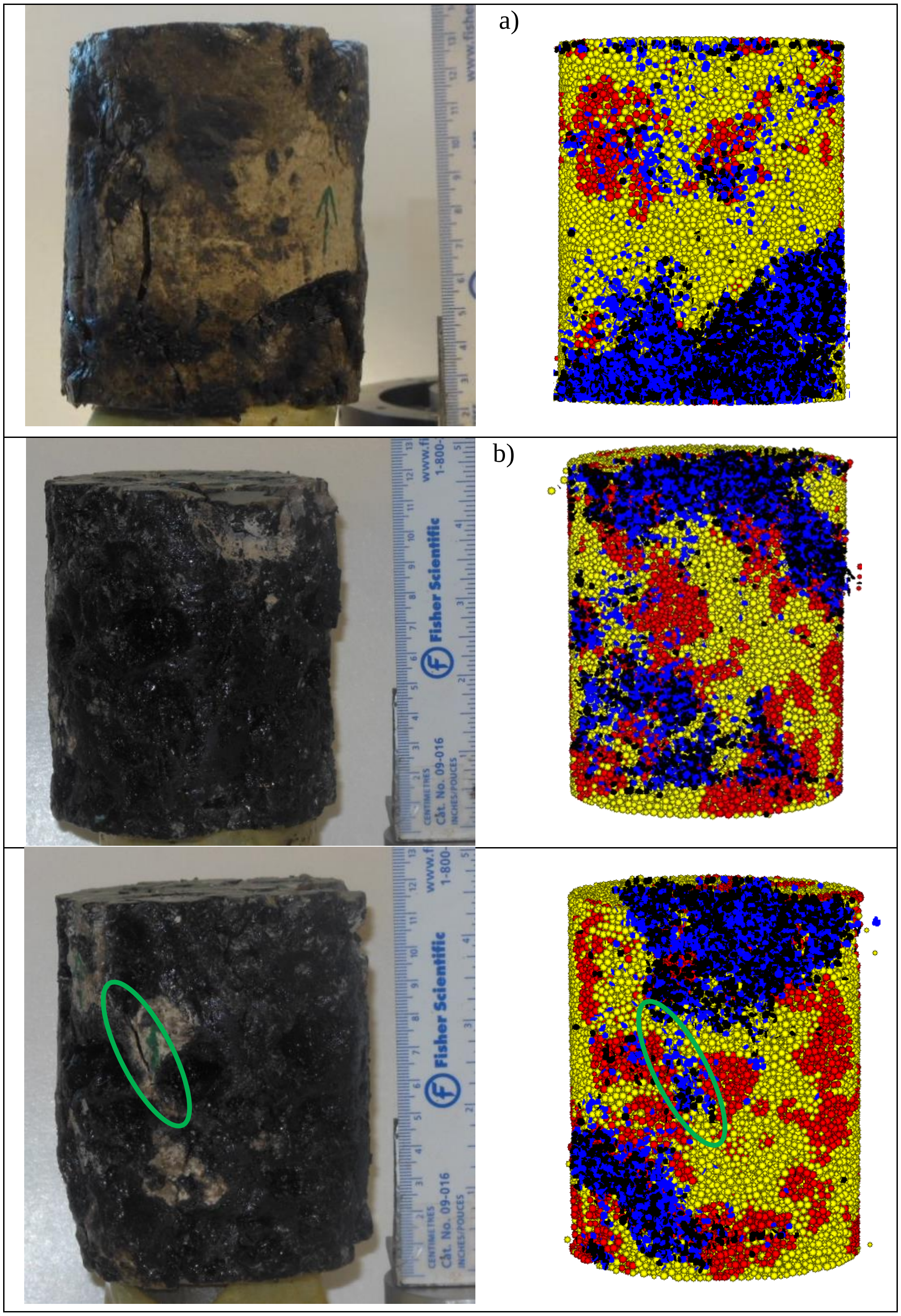


Figure 6.13: Comparison of post-test failure of the laboratory samples and PFC3D voxel vug samples: Sample S5 (a) and S4 (b and c). Yellow balls are matrix particles, red balls are matrix-vug boundary, blue disks are shear failures and black disks are tensile failures.

6.4.4. Scale Effect

Previous research with PFC3D and the BPM have found difficulties in replicating field scale results based on the calibration to small laboratory samples. The first instance of this is presented by Potyondy and Cundall (2004) in their presentation of the BPM. They attempted to replicate the failure observed at the URL's Mine-by Experiment tunnel, based on PFC2D material calibrated to intact Lac du Bonnet granite samples. The results were not encouraging, and the strength of the PFC2D required a reduction to reproduce the field results. The authors accomplished this by presenting a stress corrosion model, in which the strength of the bond was diminished over time due to the exposure to a change in stress caused by excavation of the tunnel (Potyondy and Cundall 2004; Potyondy 2007). There has been no confirmation of this stress corrosion phenomenon in granite, and it is more likely that the PFC model was not fully representative.

The synthetic rock model presented by Mas Ivars et al. (2011) used a different approach when scaling their laboratory results to the field problem. The PFC3D materials where calibrated to a scaled UCS value, which was approximately 80% of the laboratory results. The scale value was based on the Hoek and Brown (1980) scaling formula for intact laboratory samples (shown in Equation 6.1) and on the work of Yoshinaka et al. (2008) for weathered samples (see Equation 6.2). The authors justified their approach by arguing that there existed a scale effect between laboratory samples of different sizes; and thus the material needed to be calibrated to the scale of the blocks in the field problem. The scale effect is thought to be caused by the higher number of micro-flaws or weathering variability that can exist in larger samples (Pierce et al. 2009). This scale effect could be considered a factor within the larger samples tested in this research, since it is likely that the diagenetic processes that formed the vugs could have also created more micro-cracks and a weaker matrix by dissolution.

$$\frac{\sigma_d}{\sigma_{50}} = \left(\frac{50}{d}\right)^{0.18} \qquad \textbf{Eq. 6.1}$$

$$\frac{\sigma_c}{\sigma_{c0}} = \left(\frac{d_e}{d_{e0}}\right)^{-k} \quad \text{; k = 0.3- 0.9 for weathered and/or micro-flawed rock} \qquad \textbf{Eq. 6.2}$$

The scale effect in UCS specimens with PFC3D was also studied by Zhang et al. (2011). The authors simulated different samples sizes while maintaining the particle size and properties equal. The authors only discussed the scale effect as it pertained to the strength, and any variations in the stiffness were not shown. Zhang et al. (2011) considered several samples of increasing size with fracture of either equal size across all samples or proportional to the sample size. Neither of those fracture networks was able to capture the scale effect observed in the laboratory with the trend going in the opposite direction (i.e. increasing strength with increasing sample size). The authors were only able to match laboratory data when they considered a fracture network that increased at a higher rate than the sample size (Zhang et al. 2011). Thus, concluding that the increase in flaws is not linearly proportional to the sample volume.

A possible improvement to the current simulations would be to calibrate the "intact" sample to a scaled UCS based on the diameter of the larger samples. An initial attempt was made using the Equation 6.2 above with an exponent of 0.36. This was equivalent to scaling the UCS to approximately 75% of the laboratory result. The E_s and υ_s were not scaled because the available data in literature suggest that there is no scale effect on these elastic parameters (Yoshinaka et al. 2008). A replicate of S3 was generated with the micro-properties calibrated to the scaled UCS presented in Table 6.12. The results of these trials are presented in Table 6.13 and Figure 6.14. As can be observed in Figure 6.12, the peak strength of the PFC3D sample is lower but still above the laboratory observations. Moreover, there was very little change in the stiffness behaviour.

Table 6.12: Micro- and Macro-properties for material G3 based on scaled UCS

	Properties	G3
Micro-properties	Resolution (average nº of particles per diameter)	15.8
	Minimum Radius (m)	$9x10^{-4}$
	R_{max}/R_{min}	1.66
	Ball Density (kg/m^3)	2630
	Ball Modulus (GPa)	10.2
	Ball Stiffness Ratio	0.670
	Friction Coefficient	0.5
	P-bond Modulus (GPa)	9.70
	P-bond stiffness ratio	0.695
	P-bond normal Strength (MPa)	41.6 ± 8.3
	P-bond shear strength (MPa)	49.4 ± 9.9
Macro-properties	UCS (MPa)	40.98 ± 1.25
	E_s (GPa)	15.03 ± 0.12
	ν_s	0.110 ± 0.005

Table 6.13: Comparison between laboratory test and PFC3D materials calibrated to intact and "scaled intact" strength

Parameter	Laboratory Test	PFC3D Material G1	PFC3D Material G3
UCS (MPa)	12.96	28.74 ± 0.92	21.44±0.68
E_s (GPa)	2.04	10.49 ± 0.24	10.41±0.17
ν_s	0.102	0.143 ± 0.007	0.143±0.005

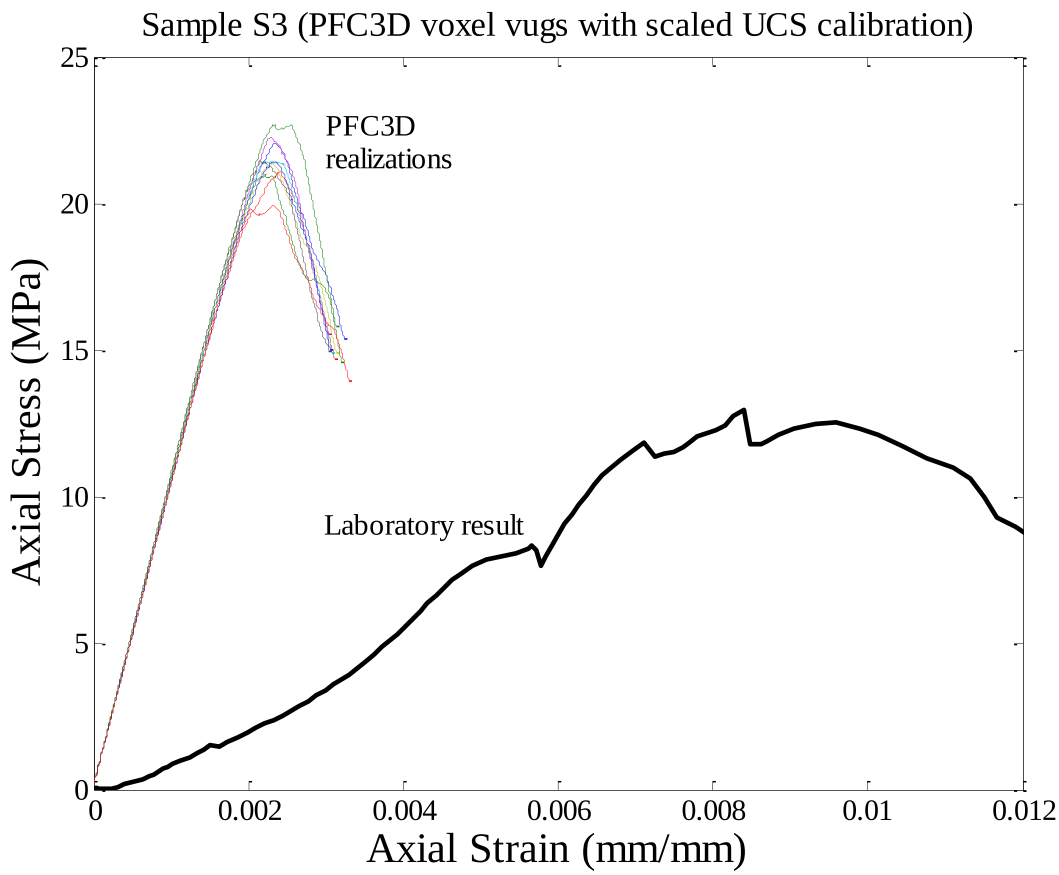


Figure 6.14: Stress versus strain curves for PFC3D voxel vug samples for scaled intact material, compared with the laboratory test results.

The above results indicate that scale effect may not be the only factor limiting proper representation of the laboratory results. Other trials could be attempted to scale the UCS to lower values; however, it would be difficult to justify lowering the strengths at the calibration stage for the purpose of matching the larger sample data. Moreover as mentioned above, current reviews of laboratory data have found no significant scale effect on the Young's modulus (Yoshinaka et al. 2008). The results presented above show that the reduction in stiffness at the larger samples is not well represented, regardless of resolution and parallel bond strengths. Finally, the initial non-linearity observed in soft rock cannot be represented with the BPM as it is currently formulated.

6.5. Possible Sources of Discrepancies

A possible reason for the difference between the PFC3D models and the laboratory results is the disproportionate tensile strength given to the bonds in the BPM. It has been shown that BPM samples have a low UCS/σ_t ratio (Potyondy and Cundall 2004; Cho et al. 2007) and samples calibrated to a compression test will over-predict the tension. The normal bond strength calibrated for BPM samples has to be increased in order to compensate for the excessive freedom of rotation, which augments the stress caused by the moment on the particle bonds (Potyondy 2010). This large normal bond strength causes over-prediction of strength in direct tension. This effect can be reduced by using clumped particle models, which add eccentricity and additional interlocking to reduce particle rotation (refer to Chapter 2).

As was discussed earlier, the detailed vug representation increases the amount of tensile bond failure. It is possible that these bonds would fail at a lower stress if the bond strength were lower. For future research, it is recommended that a clumped particle model be implemented to test these vuggy samples. A clumped model would add more interlocking and reduce the need to calibrate the normal bond strength to high values, which may result in a more realistic representation of the sample behaviour.

CHAPTER 7: CONCLUSIONS AND RECOMMENDATIONS

7.1. Summary

Vuggy specimens were simulated using the BPM in PFC3D. Initially synthetic samples with various vug configurations were studied; from simple single vug models to ordered arrays and random vug networks. The vug volume was found to be an important factor in reduction of compressive strength, tensile strength, and stiffness. Exponential relations for UCS and E_s with vuggy porosity were presented based on normalizing the vuggy results to the original intact UCS and E_s, respectively. The synthetic samples were compared based on a vug volume ratio, which was used as a geometric parameter analogous to vuggy porosity.

The behaviour observed with the synthetic samples was similar to relationships previously described in literature between strength/stiffness and porosity (Chang et al. 2006; Farquhar et al. 1994; Santos and Ferreira 2010). The location of the vugs within the sample was observed to have an effect on the maximum compressive stress obtained. Sample with ordered or symmetric vug placement had a lower reduction in strength than samples with similar vuggy porosity but with randomly situated vugs. This effect of vug arrangement on the compressive strength was not observed on the stiffness, which produced similar correlations for both ordered and random vug placements.

A vuggy carbonate testing workflow was developed to create virtual PFC3D samples based on the CT imaging data from real laboratory specimens. Matlab's Image Processing Tool box was used to create a procedure for identifying vugs and generating necessary input data for the PFC3D simulations. Three methods were proposed to represent the vugs within the sample: voxel, equivalent ellipsoid, and equivalent sphere. The voxel method gave the most detailed representation but required more computing time to generate the vuggy samples. The equivalent sphere replaced the vug with a sphere of equivalent volume at the centroid of the vug. This method was found to be more suitable for low porosity samples in which there was little interconnection between vugs. The equivalent ellipsoid over-predicted the vuggy porosity because it created ellipsoids that would encompass void voxel but also included neighbouring voxels of solid material. This equivalent ellipsoid method requires further refining and may only be applicable to low porosity specimens.

Laboratory tests on six Grosmont Carbonate samples were conducted. Samples S2 and S6 were cored to a 1.5" diameter; while samples S1, S3, S4, and S5 were maintained at their original 3.5" diameter. The minimum 2:1 length to diameter ratio was only achieved for the 1.5" diameter samples. Prior to testing, all samples were CT scanned to obtain the image data for identifying the vugs. Ultrasonic measurements and UCS tests were then carried out on all samples. There were several difficulties in the interpretation of the dynamic tests, and measurements could not be obtained for S3. The laboratory results yielded values for strength and stiffness within a reasonable range for carbonates. Both strength and stiffness were observed to decrease with increasing vuggy porosity in an exponential trend. The decrease in both strength and stiffness was greater than what was expected based on the correlations developed from idealized vuggy simulations with PFC3D. However, the limited sample data precludes making general conclusions.

The laboratory results for samples S6 were used to calibrate the micro-properties for PFC3D's BPM. The attempt to calibrate both dynamic and static elastic parameters was not successful. Representation of the dynamic behaviour is possible in PFC3D; however, it would require a separate calibration for the BPM or a different contact model. Virtual PFC3D representations of the 3.5" samples were completed using the calibrated micro-properties and two different methods for vug representation: voxel and equivalent sphere. Both methods tended to produce very similar vuggy porosity but generated different shapes in PFC3D. The equivalent sphere did not represent the void volume in the correct locations, especially for the specimens with higher vuggy porosity. The samples were tested in PFC3D's UCS environment and compared with the physical laboratory results.

The BPM was not able to reproduce the laboratory results, over-predicting both strength and stiffness. The models did not improve significantly after increasing the resolution or scaling the UCS to be approximately 75% of the calibrated value for S6. The PFC3D simulations were able to show the importance of vug shape for the behaviour of the sample. The results from voxel representation generally had lower strength and stiffness than the equivalent sphere methods, even though the specimens had very similar (or equal) vuggy porosity. Moreover, the detailed geometric representation created an increase in the number of cracks caused by tensile instead of shear failure. Features such as vug collapse and fracture patterns were captured by PFC3D with the voxel method representation; yet the overall strength predicted exceeded the laboratory measurements. Thus, it was concluded that the BPM, as it is currently implemented, is not adequate for the representation of vuggy carbonate specimens based on calibration to small intact samples.

7.2. Limitations

The limited number of samples available was a hindrance to this research program. Ideally several intact samples should be tested to determine the variability of the intact material strength. It was difficult to obtain a fully intact sample, even a very small sample (a trial of 1" diameter by 1" height) was found to contain vugs after examination through CT scans. Thus, it is recommended that several samples with vuggy porosities below 0.5% be tested to establish a range for the matrix strength. Also, multiple vuggy samples should be tested in order to determine clearer trends between strength, stiffness, and vuggy porosity. The sample geometry used for the larger vuggy specimens was not ideal and did not comply with the ISRM or ASTM standards. Furthermore, the vugs and other irregular features in the faces and edges of the samples increased the difficulty of conducting some of the laboratory tests.

The carbonate specimens were tested only under uniaxial compression due to the small number of samples. Vuggy carbonate testing under different stress paths should be considered in future studies if the behaviour at reservoir stresses is required. Testing at different confining stresses is necessary in order to determine the failure envelope of the material for different vuggy porosities; this would require a large sample set with sufficient samples of similar porosity to produce proper failure criteria. The current PFC3D virtual testing environment utilizes rigid walls for confinement that would not be representative of the laboratory tests, especially if vugs are located at the outer edges of the sample. If confined tests of vuggy specimens are required it may be necessary to develop a flexible wall routine for confining the virtual samples.

7.3. Conclusions

The workflow was successful in creating vuggy samples in PFC3D based on CT imaging data. The voxel method was found to produce the most accurate geometry in all vuggy porosities tested. The other methods proposed for vug representation were not as accurate, especially for specimens with high vuggy porosity. The traditional BPM was not able to reproduce the laboratory results when the matrix material was calibrated to a small intact sample. Vug volume was found to be an important factor in strength and stiffness reduction. However, location and shape are significant in controlling fracture patterns and stress-strain behaviour. Vug collapse events could be represented in PFC3D similar to observations in the laboratory.

The BPM could not be simultaneously calibrated to both dynamic and static test results. It is possible to represent each behaviour separately by a different calibration, and this should be considered in future research when either a dynamic or static test study is desired. For the validation in this research program, only calibration to the static tests was conducted. The micro-properties obtained still appear to over-predict the strength of the larger specimens. This could be attributed to a scale effect; however, an attempt was made to scale the parallel bond strength to a lower UCS value using established literature correlations. Even with this strength reduction the simulation results did not correspond well with the laboratory tests.

The stiffness behaviour of the BPM was the principal inconsistency with the physical tests. The initial non-linearity observed in the laboratory stress vs strain curves was not reproduced in any of the simulations. Moreover, the stiffness of the vuggy samples was between 1.6 to 5 times larger than measured in the laboratory. The representation of the vugs may not be sufficient to represent the reduction in stiffness observed within the real specimens; this may require a vug and crack network interaction which should be evaluated in future work.

7.4. Recommendations for Future Work

- The representation of the intact material should be attempted with a clumped particle model. The clumped model would add interlocking between the particles, reducing freedom of rotations and the overcompensation of the parallel bond strengths. If a clumped particle model is used, it will require a lower particle resolution to ensure the clump particles are not significantly larger than the smaller sized vugs. The clump model developed by Cho et al. (2008) is recommended over the standard clump model available with PFC3D version 4.0. The latter method was attempted during the early stages of this research and found to cause computational difficulties when void vugs were included.

- A new version of PFC3D has become available in 2014. Aside from computational improvements that allow for threading (increasing processing speed) the new version includes other developments such as the flat jointed model (Potyondy 2013), which may improve intact material representation.

- Further investigation with CT of different resolution is recommended. Micro-CT could be conducted to compare porosity of the matrix and to determine if lower strengths are caused by higher intra-particle porosities.

- Introduction of micro-cracks with varying apertures and stiffness could be attempted to calibrate the non-linearity observed in soft rock samples. The behaviour of the Grosmont carbonates is likely a complex interaction between matrix, micro-cracks, cracks, and vugs. It is recommended that the effects of discrete fracture networks on vuggy samples be evaluated and compared to the laboratory observations.

- Laboratory tests should include more detailed measurements of the radial strains to improve the estimates on υ_s. Furthermore, addition of imaging techniques such as digital image correlation could improve the analysis of local strains caused by the vugs.

- If confined tests are conducted to develop failure envelopes for vuggy specimen, it is recommended that coupling of PFC3D with FLAC is used to create confining pressures that are flexible and avoid using the current stiff walls implemented for triaxial testing in PFC3D.

- The incorporation of filling material within vugs could also be evaluated, as well as distinguishing between dolomite and limestone matrix material. However, the calibration of the micro-properties for infilling solids would require intact samples of the different materials.

- Correlations between static and dynamic data are common in the field of petroleum geomechanics. Based on the simulations conducted, simultaneous calibration for both static and dynamic stiffness is not attainable using the traditional BPM for E_s/E_d less than 0.5. Thus, new contact models should be evaluated that can represent the rock behaviour under both conditions.

- Temperature induced stresses on vuggy carbonates could be evaluated with the thermal option included in PFC3D. This would be especially significant for the Grosmont reservoir because the extraction methods are likely to include thermal schemes.

- Any future work should include a more comprehensive laboratory component with a larger sample range.

REFERENCES

Albahlani, A.M. and Babadagli, T. 2008. A critical review of the status of SAGD: Where are we and what is next? *In* Proceedings of the 2008 SPE Western Regional and Pacific Section AAPG Joint Meeting, Bakersfield, Calif., 31 March- 2 April 2008. Society of Petroleum Engineers, SPE 113283.

Alberta Oil Sands Technology and Research Authority (AOSTRA). 1990. AOSTRA: A 15 years POrtfolio of achievement. AOSTRA, Edmonton, Canada.

Alvarez, J.M., Sawatzky, R.P., Forster, L.M., and Coates, R.M. 2008. Alberta's bitumen carbonate reservoirs - moving forward with advanced R&D. *In* Proceedings of the World Heavy Oil Congress 2008, Edmonton, AB, 10 - 12 March 2008, pp. 1-14.

Ambraseys, N.N. and Hendron, A.J. 1968. Dynamic behaviour of rock massess. *In* Rock Mechanics in Engineering Practice *Edited by* K.G. Stagg and O.C. Zienkiewicz. John Wiley and Sons, London, U.K., pp. 203-236.

Ameen, M.S., Smart, B.G.D., Somerville, J.M., Hammilton, S., and Naji, N.A. 2009. Predicting rock mechanical properties of carbonates from wireline logs (A case study: Arab-D reservoir, ghawar field, saudi arabia), Marine and Petroleum Geology, **26**(4): 430-444.

Anfort, S.J., Bachu, S., and Bentley, L.R. 2001. Regional-scale hydrogeology of the upper devoinan-lower cretacous sedimentary succesion, south-central alberta basin, canada, AAPG Bulletin, **85**(4): 637-660.

Arseniuk, S.E., Becker, D.L., Barret, K.R., and Keller, D. 2010. Drilling challenges in the bitumen-saturated grosmont formation, Journal of Canadian Petroleum Technology, **49**(12): 19-27.

Arseniuk, S.E., Peterson, J., Barret, K., and Chou, Q. 2012. The evaluation of open-hole completions for steam-assisted gravity drainage wells in a heavily karsted and fractured carbonate formation. *In* Proceedings from SPE Heavy Oil Conference Canada, Calgary, AB, 12-14 June 2012. Society of Petroleum Engineers, pp. 1-11.

ASTM Standard D2845. 2008. Standard test method for laboratory determination of pulse velocities and ultrasonic elastic constants of rock. ASTM International, West Conshohocken, PA. DOI: 10.1520/ D2845-08.

ASTM Standard D7012. 2010. Standard test method for compressive strength and elastic moduli of intact ROck core specimens under varyinh states of stress and temperatures. ASTM International, West Conshohocken, PA. DOI: 10.1520/D7012-10.

Bahaaddini, M., Sharrock, G., and Hebblewhite, B.K. 2013. Numerical investigation of the effect of joint geometrical parameters on the mechanical properties of a non-persistent jointed rock mass under uniaxial compression, Computers and Geotechnics, **49**(0): 206-225.

Bathurst, R.J. and Rothenburg, L. 1988. Micromechanical aspects of isotropic granular assemblies with linear contact interactions, Journal of Applied Mathematics, **55**(1): 17-24.

Bechtel SAIC Company (Bechtel). 2004. Drift degradation analysis. DOC.200409115.0010, U.S. Department of Energy, Las Vegas, Nevada.

Bobet, A., Frakhimi, A., Johnson, S., Morris, J., Tonon, F., and Ronald Yeung, M. 2009. Numerical models in discontinuous media: Review of advances for rock mechanics applications, Journal of Geotechnical and Geoenvironmental Engineering, **135**(11): 1547-1561.

Borrero, M.L. 2010. Hondo evaporites within the grosmont heavy oil carbonate platform, alberta, canada. M.Sc. thesis, Department of Earth and Atmospheric Sciences, University of Alberta, Edmonton, Alberta, Canada.

Buschkuehle, B.E., Hein, F.J., and Grobe, M. 2007. An overview of the geology of the upper devonian grosmont carbonate deposit, northern alberta, canada, Natural Resources Research, **14**(1): 3-15.

Chang, C., Zoback, M.D., and Khaksar, A. 2006. Empirical relations between rock strength and physical properies in sedimentary rocks, Journal of Petroleum Science and Engineering, **51**(3-4): 223-237.

Cho, N. 2008. Discrete element modeling of rock: Pre-peak fracturing and dilation. Ph.D. thesis, Department of Civil and Environmental Engineering, University of Alberta, Edmonton, Alberta, Canada.

Cho, N., Martin, C.D., and Sego, D.C. 2007. A clumped particle model for rock, International Journal of Rock Mechanics and Mining Sciences, **44**(7): 997-1010.

Cundall, P.A. 1988. Formulation of a three-dimensional distinct element model - part I. A scheme to detect and represent contact in a system composed of many polyhedral blocks, International Journal of Rock Mechanics and Mining Sciences & Geomechanics Abstracts, **25**(3): 107-116.

Cundall, P.A., Pierce, M.E., and Mas Ivars, D. 2008. Quantifying the size effect of rock mass strength. *In* Proceedings from the 1st Southern Hemisphere International Rock Mechanics Symposium, Perth, Australia, 16-19 Septembre 2009, pp. 3-15.

Deere, D.U. 1968. Geological considerations. *In* Rock Mechanics in Engineering Practice *Edited by* K.G. Stagg and O.C. Zienkiewicz. John Wiley and Sons, London, U.K., pp. 1-20.

Deisman, N. 2008. PFC3D virtual lab assistant (VLA). Nathan Deisman, University of Alberta, Edmonton, AB.

Dembicki, E.A. and Machel, H.G. 1996. Recognition and delineation of paleokarst zones by the use of wireline logs in the bitumen-saturated upper devonian grosmont formation of northeastern alberta, canada, AAPG Bulletin, **80**(5): 695-712.

Ding, X. and Zhang, L. 2011. Simulation of rock fracturing using particle flow modeling: Phase I - model development and calibration. *In* Proceedings from the 45th U.S. Rock Mechanics / Geomechanics Symposium, San Francisco, Calif. 26-29 June 2011. American Rock Mechanics Association, ARMA-11-510.

Donzé, F.V., Harthong, B., and Scholtés, L. 2013. DEM modeling of jointed rock slopes. *In* Proceedings from the 47th US Rock Mechanics / Geomechanics Symposium, San Francisco, Calif., 23-26 June 2013. American Rock Mechanics Association, ARMA-13-744

Edmunds, N., Barret, K., Solanki, S., Cimolai, M., and Wong, A. 2009. Prospects for commercial bitumen recovery from the grosmont carbonate, alberta, Journal of Canadian Petroleum Technology, **48**(9): 26-32.

Energy Resources Conservation Board (ERCB). 2010. ST98-2010: Alberta's energy reserves 2009 and Supply/Demand outlook 2010-2019. Energy Resources Conservation Board, Calgary, AB, Canada.

Energy Resources Conservation Board (ERCB). 2012. ST98-2012: Alberta's energy reserves 2011 and Supply/Demand outlook 2012-2021. Energy Resources Conservation Board, Calgary, AB, Canada.

Farquhar, R.A., Somerville, J.M., and Smart, B.G.D. 1994. Porosity as a geomechanical indicator: An application of core and log data and rock mechanics. *In* Proceedings of the European Petroleum Conference, London. U.K., 25-27 October 1994, pp. 481-489.

Freeman, T.T., Chalaturnyk, R.J., and Bogdanov, I.I. 2009. Geomechanics of heterogeneous bitumen carbonates. *In* Proceedings of the SPE Reservoir Simulation Symposium, The Woodlands, Texas, 2-4 February, 2009. Society of Petroleum Engineers, SPE-119151-MS.

Gercek, H. 2007. Poisson's ratio values for rocks, International Journal of Rock Mechanics and Mining Sciences, **44**(1): 1-13.

Gonzalez, R.C., Woods, R.E., and Eddins, S.L. 2009. Digital image processing using MATLAB, 2nd ed. Gatesmark Publishing, .

Griffith, A.A. 1921. The phenomena of rupture and flow in solids, Philosophical Transactions of the Royal Society of London., **221**(A): 163-198.

Guo, J., Xu, G., Jing, H., and Kuang, T. 2013. Fast determination of meso-level mechanical parameters of PFC models, International Journal of Mining Science and Technology, **23**(1): 157-162.

Gurevich, B., Osypov, K., Ciz, R., and Makarynska, D. 2008. Modeling elastic wave velocities and attenuation in rocks saturated with heavy oil, Geophysics, **73**(4): E115-E122.

Hart, R., Cundall, P.A., and Lemos, J. 1988. Formulation of a three-dimensional distinct element model - part II. mechanical calculations for motion and interaction of a system composed of many polyhedral blocks, International Journal of Rock Mechanics and Mining Sciences & Geomechanics Abstracts, **25**(3): 117-125.

Hazzard, J.F., and Young, R.P. 2000. Micromechanical modeling of cracking and failure in brittle rocks, Journal of Geophysical Research, **105**(B7): 16683-16697.

Hoek, E. and Brown, T. 1980. Underground excavations in rock. Institution of Mining and Metallurgy, London.

Hoek, E., Carter, T.G., and Diederichs, M. 2013. Quantification of the geological strength index chart. *In* Proceedings from the 47th US Rock Mechanics/Geomechanics Symposium, San Francisco, U.S., 23-26 June 2013. American Rock Mechanics Association, pp. ARMA12-672.

Holt, R.M., Kjolaas, J., Larsem, I., Li, L., Gotusso Pillitteri, A., and Sonstebo, E.F. 2005. Comparison between controlled laboratory experiments and discrete particle simulations of the mechanical behaviour of rock, International Journal of Rock Mechanics and Mining Sciences, **42**(7-8): 985-995.

Itasca. 2008a. PFC3D particle flow code in 3 dimensions, theory and background. Itasca Consulting Group, Inc., Minneapolis, Minnesota, U.S.

Itasca. 2008b. PFC3D particle flow code in 3 dimensions, verification problems and example applications. Itasca Consulting Group, Inc., Minneapolis, Minnesota, U.S.

Jiang, Q., Yuan, J., Russel-Houston, J., Thornton, B., and Squires, A. 2009. Evaluation of recovery technologies for the grosmont carbonate reservoirs. *In* Proceedings of the Canadian Internation Petroleum Conference, Calgary, Alberta, 16-18 June 2009, pp. 1-11.

Kozicki, J. and Donzé, F.V. 2008. A new open-source software developed for numerical simulations using discrete modeling methods, Computer Methods in Applied Mechanics and Engineering, **197**(49–50): 4429-4443.

Legland, D. 2011. Image ellipsoid 3D. INRA - Cepia Software Platform, MathWorks File Exchange.

Lenoir, N., Bornert, M., Desrues, J., Bésuelle, P., and Viggiani, G. 2007. Volumetric digital image correlation applied to X-ray microtomography images from triaxial compression tests on argillaceous rock, Strain, **43**(3): 193-205.

Lonoy, A. 2006. Making sense of carbonate pore systems, AAPG Bulletin, **90**(9): 1381-1405.

Lopez, O., Mock, A., Øren, P.E., Long, H., Kalam, Z., Vahrenka, V., Gibrata, M., Seraj, S., Chacko, S., Al Hammadi, M., Al Hosni, H., Sahn, H., and Vizamora, A. 2012. Validation of fundamental carbonate reservoir core properties using digital rock physics. *In* Proceedings from the INternational Symposium of the Society of Core Analysts, Aberdeen, U.K., 27-30 August 2012. Society of Core Analysts, SCA2012-19.

Luo, P. and Machel, H.G. 1995. Pore size and pore throat types in a heterogeneous dolostone reservoir, devonian grosmont formation, western canada sedimentary basin, AAPG Bulletin, **79**(11): 1695-1720.

Martin, C.D. and Chandler, N.A. 1994. The progressive fracture of lac du bonnet granite, International Journal of Rock Mechanics and Mining Sciences & Geomechanics Abstracts, **31**(6): 643-659.

Martínez-Martínez, J., Benavente, D., and García-del-Cura, M.A. 2011. Mechanical analysis of multi-textural rocks (brecciated dolostones and limestones): A new micro-compression test for rocks, Key Engineering Materials, **465**(0): 479-482.

Mas Ivars, D., Deisman, N., Pierce, M., and Fairhurst, C. 2007. The synthetic rock mass approach - A step forward in the characterication of jointed rock masses. *In* 11th Congress of the International Society for ROck Mechanics, pp. 485-490.

Mas Ivars, D., Pierce, M.E., Darcel, C., Reyes-Montes, J., Potyondy, D.O., Young, R.P., and Cundall, P.A. 2011. The synthetic rock mass approach for jointer rock mass modelling, International Journal of Rock Mechanics and Mining Sciences, **48**(2): 219-244.

MathWorks. 2012. Image processing Toolbox^{TM} user's guide R2012a MATLAB. MathWorks Inc., Natick, MA.

Meléndez Martínez, J. and Schmitt, D.R. 2013. Anisotropic elastic moduli of carbonates and evaporites from the weyburn-midale reservoir and seal rocks, Geophysical Prospecting, **61**(2): 363-379.

Miller, R.P. 1965. Engineering classification and index properties for intact rock. Ph. D., University of Illinois, Urbana, Illinois, USA.

Palchik, V. 2011. On the ratios between elastic modulus and uniaxial compressive strength of heterogeneous carbonate rocks, Rock Mech Rock Eng, **44**(1): 121-128.

Park, E.S., Martin, C.D., and Christiansson, R. 2004. Simulation of the mechanical behaviour of discontinuous rock masses using a bonded-particle model. *In* Proceedings from Gulf Rocks 2004, the 6th North America Rock Mechanics Symposium (NARMS), Houton, Texas, 5-9 June 2004. American Rock Mechanics Association, ARMA\NARMS 04-480.

Petružálek, M., Lokajícek, T., and Svitek , T. 2013. The anisotropy of ultrasonic waves velocity and attenuation of migmatite samples under uniaxial loading. *In* Proceedings from the 47th US Rock Mechanics/Geomechanics Symposium, San Francisco, U.S., 23-26 June 2013. American Rock Mechanics Association, ARMA13-541.

Pierce, M., Cundall, P., Potyondy, D., and Ivars, D. 2007. A synthetic rock mass model for jointed rock. *In* Proceedings of the 1st Canada-US Rock Mechanics Symposium, Vancouver, BC, 27–31 May 2007. Taylor & Francis, London, pp. 341-349.

Pierce, M., Gaida, M., and DeGagne, D. 2009. Estimation of rock block strength. *In* Proceedings of the 3rd CANUS Rock Mechanics Symposium (RockEng09), Toronto, ON, 9-15 May 2009. Canadian Association of Rock Mechanics, Paper 4360.

Potyondy, D.O. 2007. Simulating stress corrosion with a bonded-particle model for rock, International Journal of Rock Mechanics and Mining Sciences, **44**(5): 677-691.

Potyondy, D. 2010. Technical memorandum: PFC2D Flat-Jointed Material Creation and Testing. Intasca Consulting Group, Inc., Mineapolis, Minnesota, U.S.

Potyondy, D. 2013. Technical memorandum: BPM improvements to match macroproperties of hard rock. Intasca Consulting Group, Inc., Mineapolis, Minnesota, U.S.

Potyondy, D.O. and Cundall, P.A. 2004. A bonded-particle model for rock, International Journal of Rock Mechanics and Mining Sciences, **41**(8): 1329-1364.

Rajabzadeh, M.A., Moosavinasab, Z., and Rakhshandehroo, G. 2012. Effects of rock classes and porosity on the relation between uniaxial compressive strength and some rock properties for carbonate rocks, Rock Mechanics and Rock Engineering, **45**(1): 113-122.

Rothenburg, L. and Bathurst, R.J. 1991. Numerical simulation of idealized granular aseemblies with elliptical particles, Computer and Geotechnics, **11**(4): 315-329.

Rothenburg, L. and Bathurst, R.J. 1993. Influence of particle eccentricity on micromechanical behavior of granular materials, Mechanics of Material, **16**(1-2): 141-152.

Santos, E.S.R. and Ferreira, F.H. 2010. Mechanical behavior of a brazilian off-shore carbonate reservoir. *In* Proceedings of the 44th US Rock Mechanics Symposium and 5th U.S.-Canada Rock Mechanics Symposium, Salt Lake City, UT, 27-30 June 2010. American Rock Mechanics Association, ARMA 10-199.

SBEL. 2000. Triaxial Cell Manual. Structural Behavior Engineering Laboratories, Inc. Phoenix, Arizona.

Scholtés, L. and Donzé, F.V. 2012. Modelling progressive failure in fractured rock masses using a 3D discrete element method, International Journal of Rock Mechanics and Mining Sciences, **52**(0): 18-30.

Scholtés, L. and Donzé, F.V. 2013. A DEM model for soft and hard rocks: Role of grain interlocking on strength, Journal of the Mechanics of Physics and Solids, **61**(2): 352-369.

Schöpfer, M., Abe, P.J., Childs, C., and Walsh, J.J.,. 2009. The impact of porosity and crack density on the elasticity, strength and friction of cohesive granular materials: Insight from DEM modeling, International Journal of Rock Mechanics and Mining Sciences, **46**(2): 250-261.

Shulakova, V., Pervukhina, M., Müller, T.M., Lebedev, M., Mayo, S., Schmid, S., Golodoniuc, P., De Paula, O.B., Clennell, M.B., and Gurevich, B. 2013. Computational elastic up-scaling of sandstone on the basis of X-ray micro-tomographic images, Geophysical Prospecting, **61**(2): 287-301.

Switzer, S.B., Holland, W.G., Christie, D.S., Graf, G.C., Hedinger, A.S., McAuley, R.J., Wierzbicki, R.A., and Packard, J.J. 1994. Devonian woodbend-winterburn strata of the western canada sedimentary basin. *In* Geological Atlas of the Western Canada Sedimentary Basin *Edited by* G.D. Mossop and I. Shetsen. Canadian Society of Petroleum Geologist and Alberta Research Council, pp. 165-202.

Tawadrous, A.S., DeGagné, D., Pierce, M., and Mas Ivars, D. 2009. Prediction of uniaxial compression PFC3D model micro-properties using artificial neural networks, International Journal for Numerical and Analytical Methods in Geomechanics, **33**(18): 1953-1962.

Thériault, F. 1988. Lithofacies, diagenesis and related reservoir properties of the upper devonian grosmont formation, northern alberta, Bulletin of Canadian Petroleum Geology, **36**(1): 52-69.

Thomsen, L. 1986. Weak elastic anisotropy, Geophysics, **51**(10): 1954-1966.

Weinsstein, E.W. 2012. "Ellipsoid" from MarhWorld--A wolfram web resource [online]. Available from http://mathworld.wolfram.com/Ellipsoid.html [cited July/27 2012].

Xie, B., Guo, L., and Xia, X. 2012. Numerical study on uniaxial compression test of rock specimen with random holes, Advanced Materials Research, **418-420**: 848-850.

Yoshinaka, R., Osada, M., Park, H., Sasaki, T., and Sasaki, K. 2008. Practical determination of mechanical design parameters of intact rock considering scale effect, Engineering Geology, **96**(3–4): 173-186.

Zhang, Q., Zhu, H., Zhang, L., and Ding, X. 2011. Study of scale effect on intact rock strength using particle flow modeling, International Journal of Rock Mechanics and Mining Sciences, **48**(8): 1320-1328.

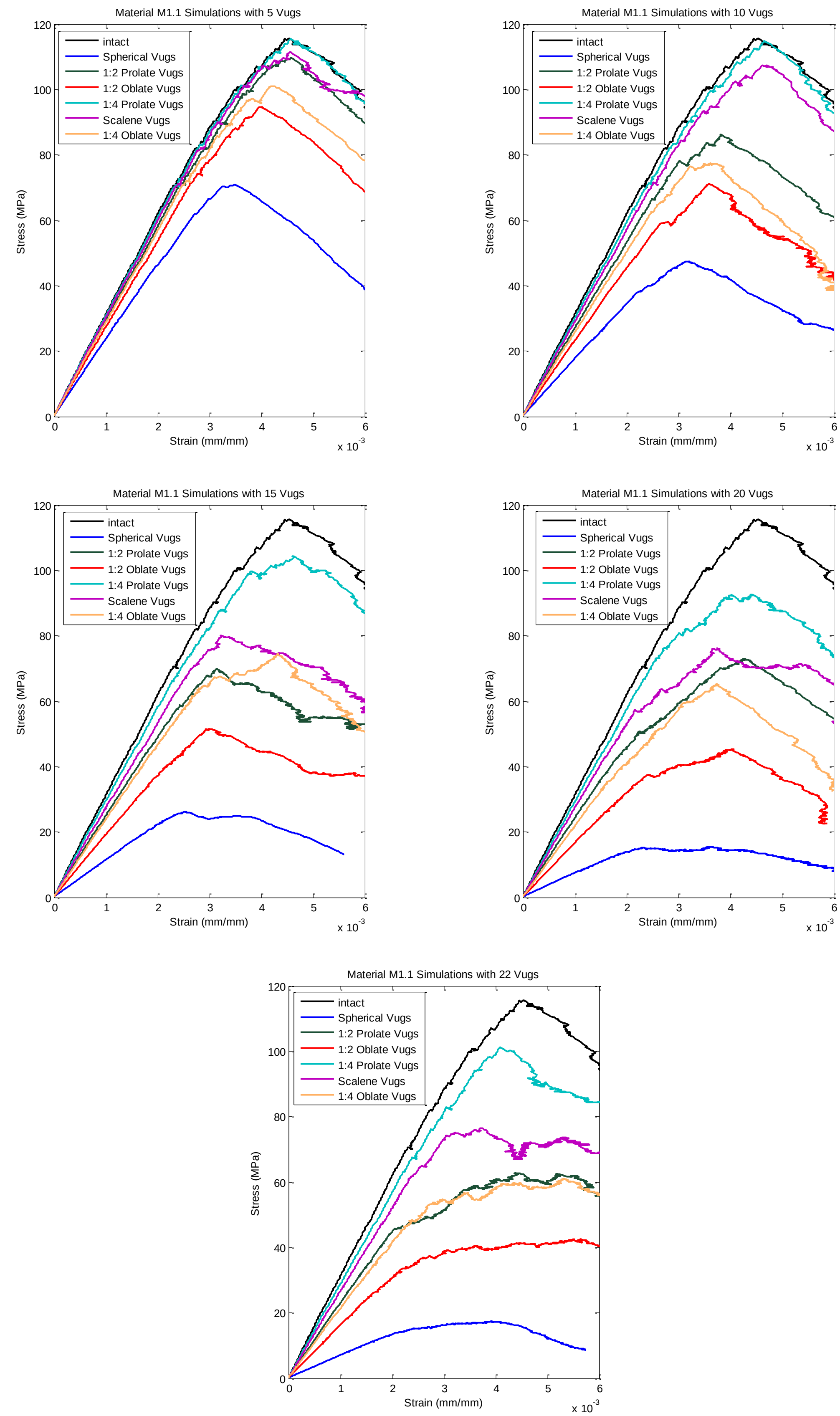
Zhang, S., Klimentidis, R.E., and Barthelemy, P. 2012. Micron to millimeter upscaling of shale rock properties based on 3D imaging and modeling. *In*

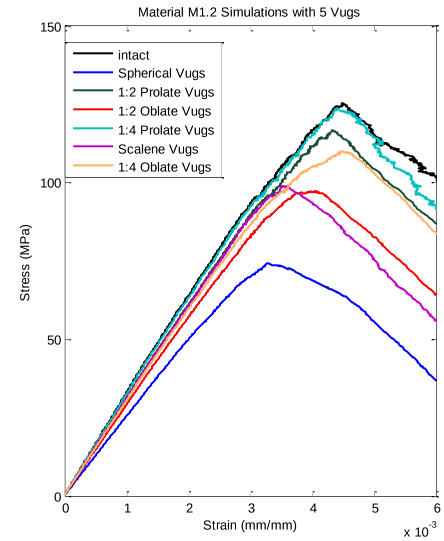
Proceedings from the International Symposium Society of Core Analyst, Aberdeen, U.K., 27-30 August 2012. Society of Core Analyst, SCA2012-20.

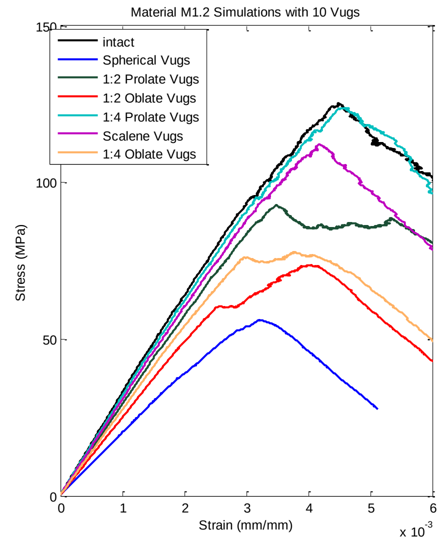
Zhao, Y. 2009. Pretrophysical properties of bitumen from the upper devonian grosmont reservoir, alberta, canada. M.Sc., Department of Earth and Atmospheric Sciences, University of Alberta, Edmonton, Alberta, Canada.

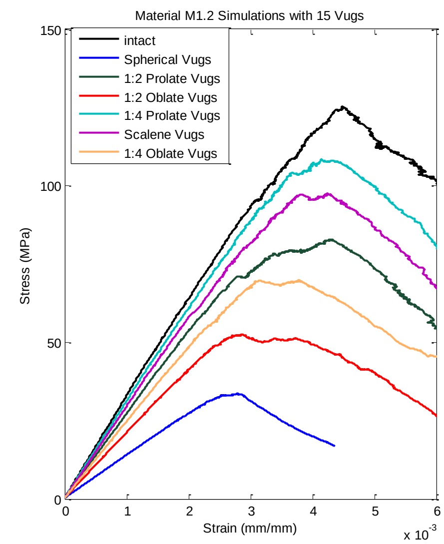
Zinsmeister, L., Dautriat, J., Dimanov, A., Raphanel, J., and Bornert, M. 2013. Mechanical evolution of an altered limestone using 2D and 3D digital image correlation (DIC). *In* Proceedings from the 47th US Rock Mechanics/Geomechanics Symposium, San Francisco, U.S., 23-26 June 2013. American Rock Mechanics Association, ARMA13-393.

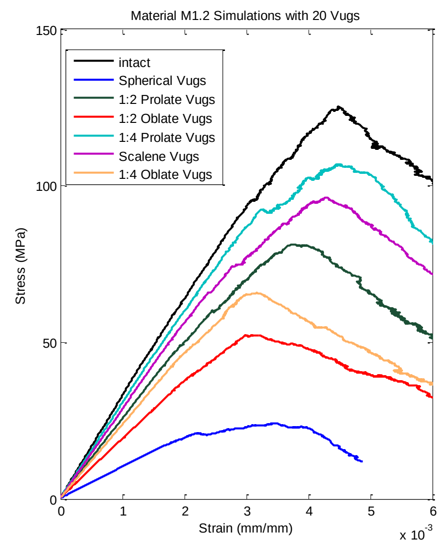
APPENDIX A: Idealized Vuggy Sample Simulations

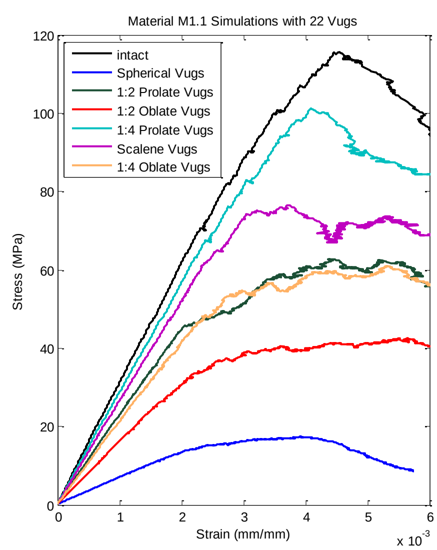

Material M1.1 Simulations with 5 Vugs
Material M1.1 Simulations with 10 Vugs
Material M1.1 Simulations with 15 Vugs
Material M1.1 Simulations with 20 Vugs
Material M1.1 Simulations with 22 Vugs
intact
Spherical Vugs
1:2 Prolate Vugs
1:2 Oblate Vugs
1:4 Prolate Vugs
Scalene Vugs
1:4 Oblate Vugs
Stress (MPa)
Strain (mm/mm)
x 10^-3

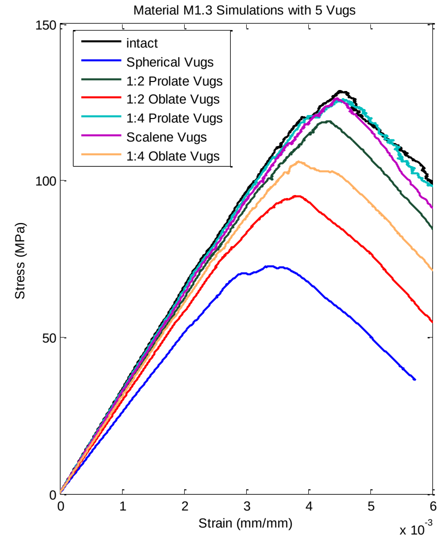

Material M1.2 Simulations with 5 Vugs
intact
Spherical Vugs
1:2 Prolate Vugs
1:2 Oblate Vugs
1:4 Prolate Vugs
Scalene Vugs
1:4 Oblate Vugs
Stress (MPa)
Strain (mm/mm)
x 10^-3
0
50
100
150
1
2
3
4
5
6

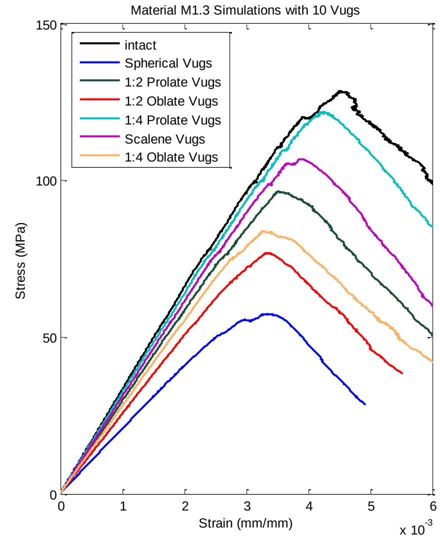

Material M1.2 Simulations with 10 Vugs
intact
Spherical Vugs
1:2 Prolate Vugs
1:2 Oblate Vugs
1:4 Prolate Vugs
Scalene Vugs
1:4 Oblate Vugs
Stress (MPa)
Strain (mm/mm)
x 10^-3
0
50
100
150
1
2
3
4
5
6

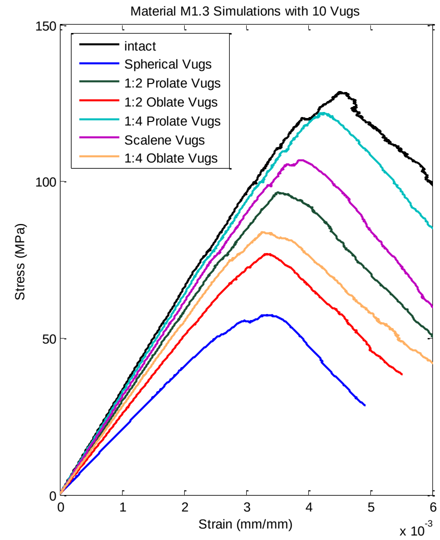

Material M1.2 Simulations with 15 Vugs
intact
Spherical Vugs
1:2 Prolate Vugs
1:2 Oblate Vugs
1:4 Prolate Vugs
Scalene Vugs
1:4 Oblate Vugs
Stress (MPa)
Strain (mm/mm)
x 10^-3
0
50
100
150
1
2
3
4
5
6

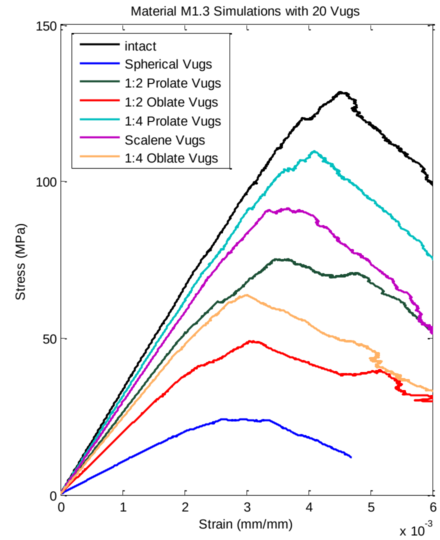

Material M1.2 Simulations with 20 Vugs
intact
Spherical Vugs
1:2 Prolate Vugs
1:2 Oblate Vugs
1:4 Prolate Vugs
Scalene Vugs
1:4 Oblate Vugs
Stress (MPa)
Strain (mm/mm)
x 10^-3
0
50
100
150
1
2
3
4
5
6

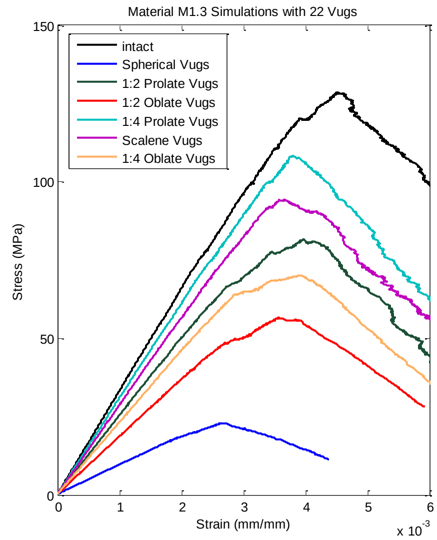

Material M1.1 Simulations with 22 Vugs
intact
Spherical Vugs
1:2 Prolate Vugs
1:2 Oblate Vugs
1:4 Prolate Vugs
Scalene Vugs
1:4 Oblate Vugs
Stress (MPa)
Strain (mm/mm)
x 10^-3
0
20
40
60
80
100
120
1
2
3
4
5
6

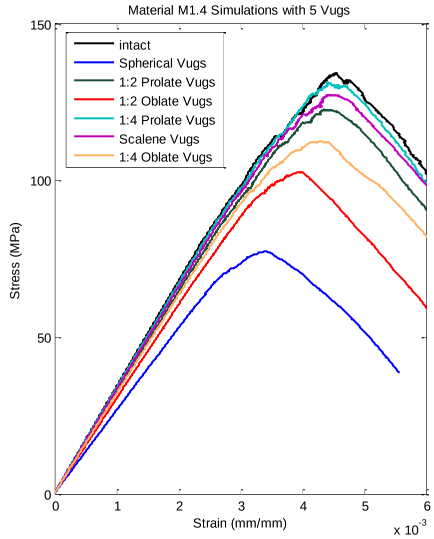

Material M1.3 Simulations with 5 Vugs
intact
Spherical Vugs
1:2 Prolate Vugs
1:2 Oblate Vugs
1:4 Prolate Vugs
Scalene Vugs
1:4 Oblate Vugs
Stress (MPa)
Strain (mm/mm)
x 10^-3
0
50
100
150
0
1
2
3
4
5
6

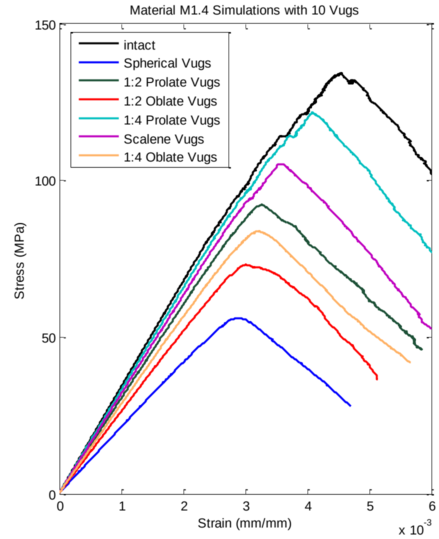

Material M1.3 Simulations with 10 Vugs
intact
Spherical Vugs
1:2 Prolate Vugs
1:2 Oblate Vugs
1:4 Prolate Vugs
Scalene Vugs
1:4 Oblate Vugs
Stress (MPa)
Strain (mm/mm)
x 10^-3
0
50
100
150
0
1
2
3
4
5
6

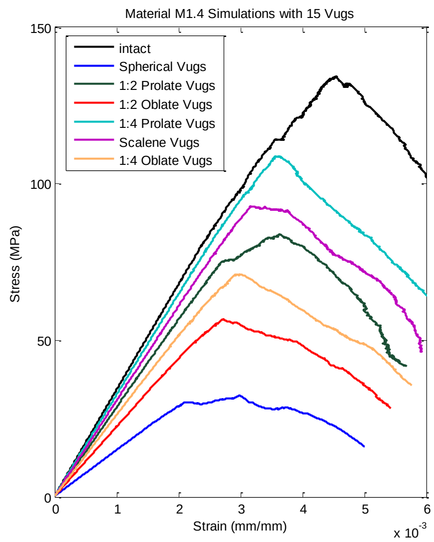

Material M1.3 Simulations with 10 Vugs
intact
Spherical Vugs
1:2 Prolate Vugs
1:2 Oblate Vugs
1:4 Prolate Vugs
Scalene Vugs
1:4 Oblate Vugs
Stress (MPa)
Strain (mm/mm)
x 10^-3
0
50
100
150
0
1
2
3
4
5
6

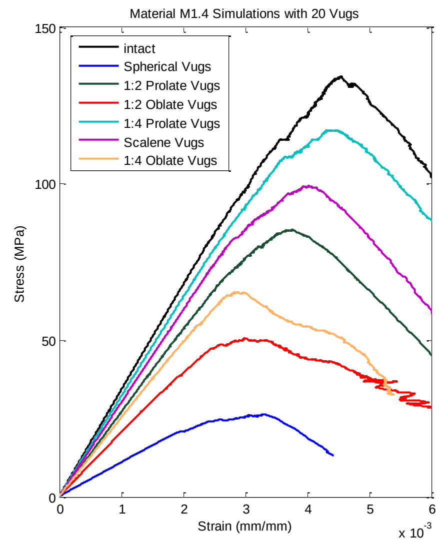

Material M1.3 Simulations with 20 Vugs
intact
Spherical Vugs
1:2 Prolate Vugs
1:2 Oblate Vugs
1:4 Prolate Vugs
Scalene Vugs
1:4 Oblate Vugs
Stress (MPa)
Strain (mm/mm)
x 10^-3
0
50
100
150
0
1
2
3
4
5
6

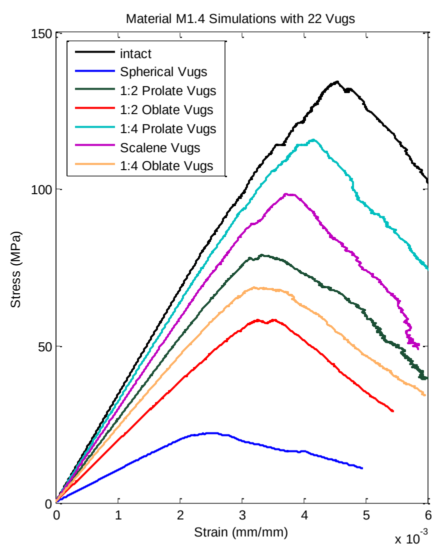

Material M1.3 Simulations with 22 Vugs
intact
Spherical Vugs
1:2 Prolate Vugs
1:2 Oblate Vugs
1:4 Prolate Vugs
Scalene Vugs
1:4 Oblate Vugs
Stress (MPa)
Strain (mm/mm)
x 10^-3
0
50
100
150
0
1
2
3
4
5
6

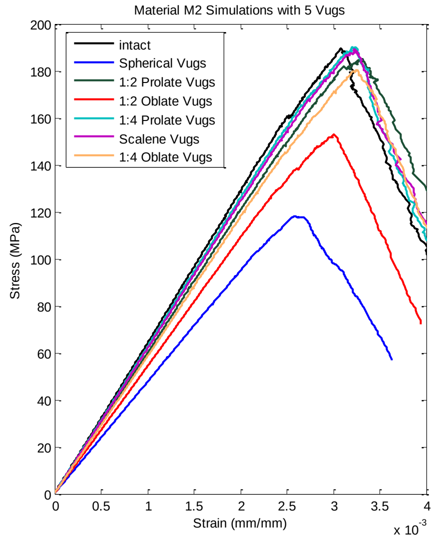

Material M1.4 Simulations with 5 Vugs
intact
Spherical Vugs
1:2 Prolate Vugs
1:2 Oblate Vugs
1:4 Prolate Vugs
Scalene Vugs
1:4 Oblate Vugs
Stress (MPa)
Strain (mm/mm)
x 10^-3
0
50
100
150
0
1
2
3
4
5
6

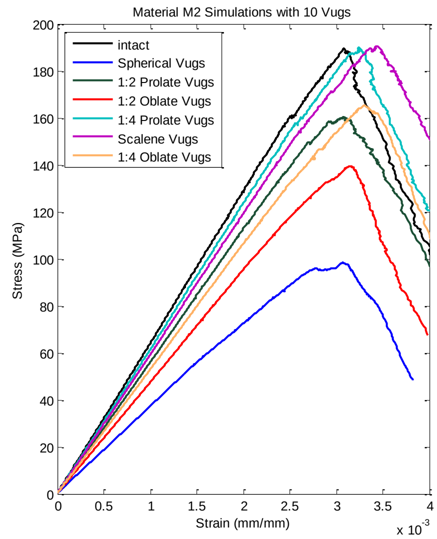

Material M1.4 Simulations with 10 Vugs
intact
Spherical Vugs
1:2 Prolate Vugs
1:2 Oblate Vugs
1:4 Prolate Vugs
Scalene Vugs
1:4 Oblate Vugs
Stress (MPa)
Strain (mm/mm)
x 10^-3
0
50
100
150
0
1
2
3
4
5
6

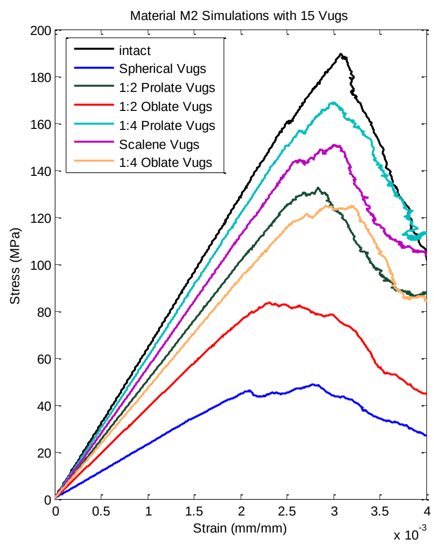

Material M1.4 Simulations with 15 Vugs
intact
Spherical Vugs
1:2 Prolate Vugs
1:2 Oblate Vugs
1:4 Prolate Vugs
Scalene Vugs
1:4 Oblate Vugs
Stress (MPa)
Strain (mm/mm)
x 10^-3
0
50
100
150
0
1
2
3
4
5
6

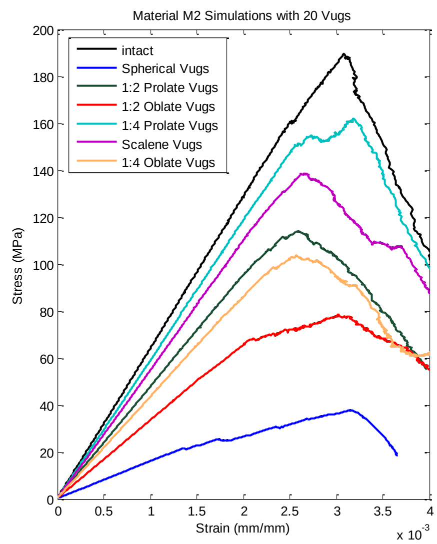

Material M1.4 Simulations with 20 Vugs
intact
Spherical Vugs
1:2 Prolate Vugs
1:2 Oblate Vugs
1:4 Prolate Vugs
Scalene Vugs
1:4 Oblate Vugs
Stress (MPa)
Strain (mm/mm)
x 10^-3
0
50
100
150
0
1
2
3
4
5
6

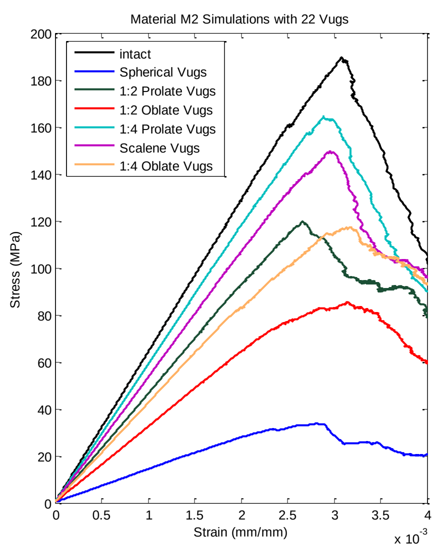

Material M1.4 Simulations with 22 Vugs
intact
Spherical Vugs
1:2 Prolate Vugs
1:2 Oblate Vugs
1:4 Prolate Vugs
Scalene Vugs
1:4 Oblate Vugs
Stress (MPa)
Strain (mm/mm)
x 10^-3
0
50
100
150
0
1
2
3
4
5
6

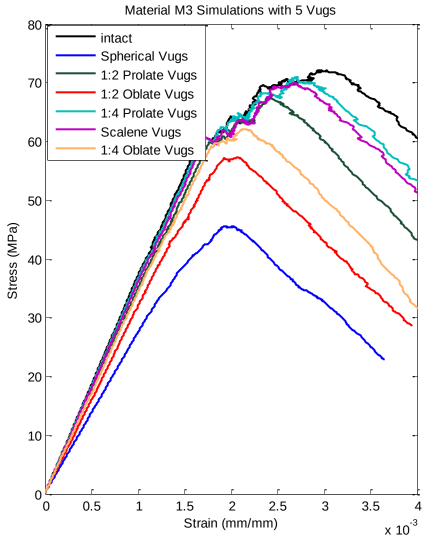

Material M2 Simulations with 5 Vugs
intact
Spherical Vugs
1:2 Prolate Vugs
1:2 Oblate Vugs
1:4 Prolate Vugs
Scalene Vugs
1:4 Oblate Vugs
Stress (MPa)
Strain (mm/mm)
x 10^-3

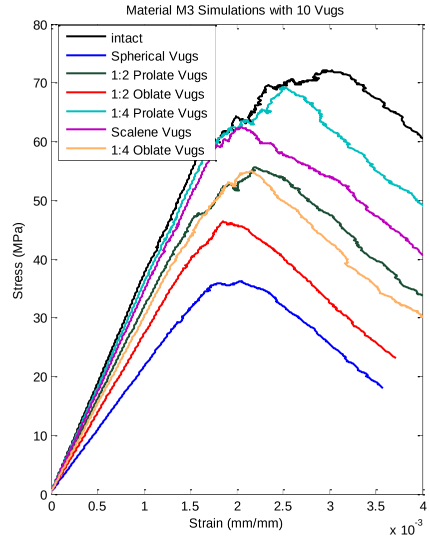

Material M2 Simulations with 10 Vugs
intact
Spherical Vugs
1:2 Prolate Vugs
1:2 Oblate Vugs
1:4 Prolate Vugs
Scalene Vugs
1:4 Oblate Vugs
Stress (MPa)
Strain (mm/mm)
x 10^-3

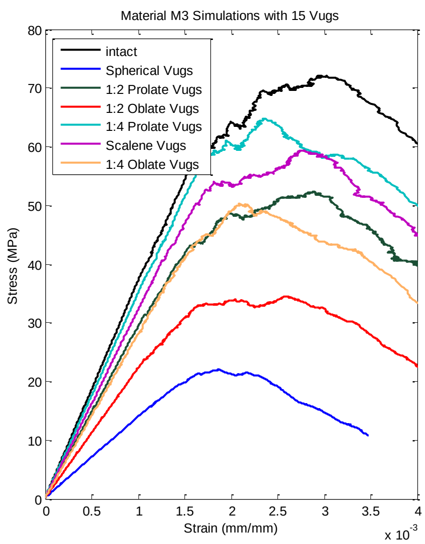

Material M2 Simulations with 15 Vugs
intact
Spherical Vugs
1:2 Prolate Vugs
1:2 Oblate Vugs
1:4 Prolate Vugs
Scalene Vugs
1:4 Oblate Vugs
Stress (MPa)
Strain (mm/mm)
x 10^-3

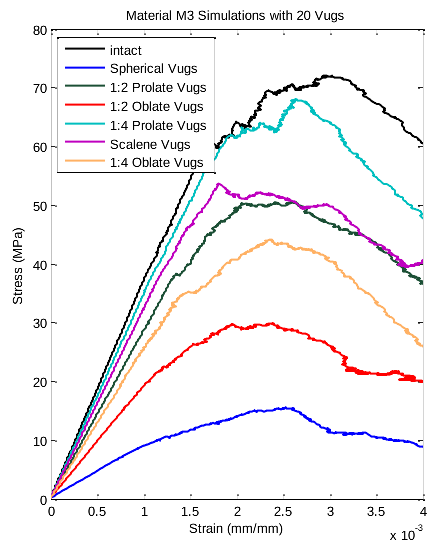

Material M2 Simulations with 20 Vugs
intact
Spherical Vugs
1:2 Prolate Vugs
1:2 Oblate Vugs
1:4 Prolate Vugs
Scalene Vugs
1:4 Oblate Vugs
Stress (MPa)
Strain (mm/mm)
x 10^-3

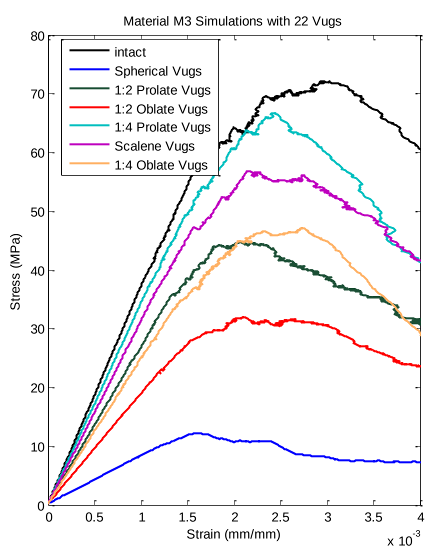

Material M2 Simulations with 22 Vugs
intact
Spherical Vugs
1:2 Prolate Vugs
1:2 Oblate Vugs
1:4 Prolate Vugs
Scalene Vugs
1:4 Oblate Vugs
Stress (MPa)
Strain (mm/mm)
x 10^-3

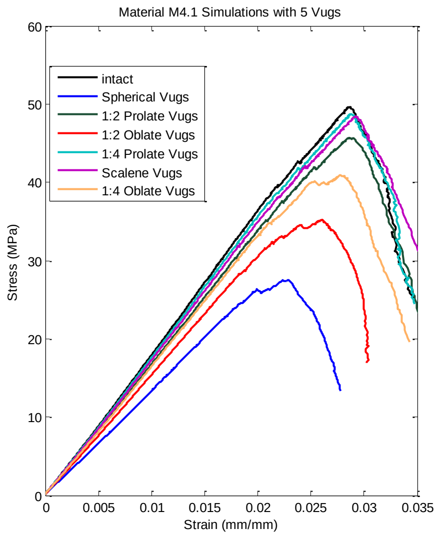

Material M3 Simulations with 5 Vugs
intact
Spherical Vugs
1:2 Prolate Vugs
1:2 Oblate Vugs
1:4 Prolate Vugs
Scalene Vugs
1:4 Oblate Vugs
Stress (MPa)
Strain (mm/mm)
x 10^-3

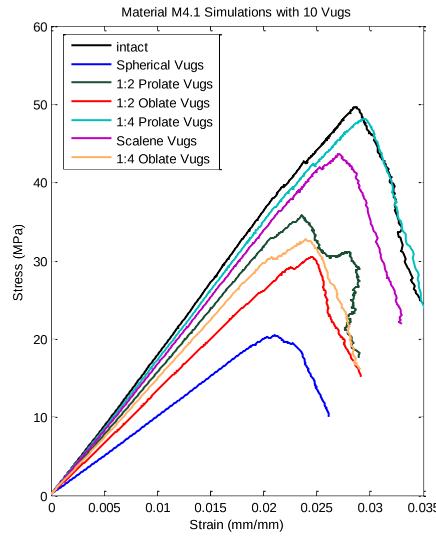

Material M3 Simulations with 10 Vugs
intact
Spherical Vugs
1:2 Prolate Vugs
1:2 Oblate Vugs
1:4 Prolate Vugs
Scalene Vugs
1:4 Oblate Vugs
Stress (MPa)
Strain (mm/mm)
x 10^-3

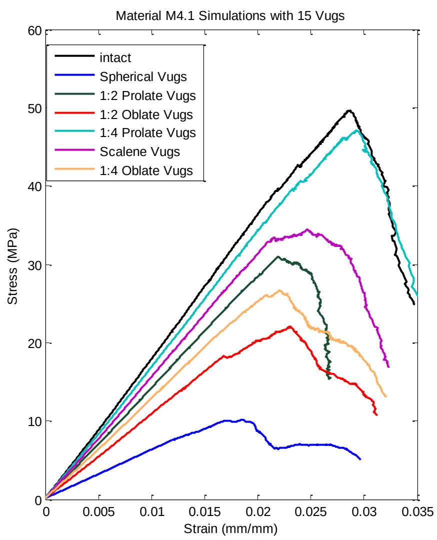

Material M3 Simulations with 15 Vugs
intact
Spherical Vugs
1:2 Prolate Vugs
1:2 Oblate Vugs
1:4 Prolate Vugs
Scalene Vugs
1:4 Oblate Vugs
Stress (MPa)
Strain (mm/mm)
x 10^-3

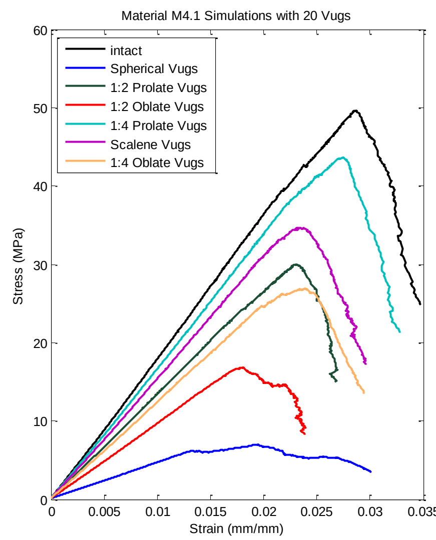

Material M3 Simulations with 20 Vugs
intact
Spherical Vugs
1:2 Prolate Vugs
1:2 Oblate Vugs
1:4 Prolate Vugs
Scalene Vugs
1:4 Oblate Vugs
Stress (MPa)
Strain (mm/mm)
x 10^-3

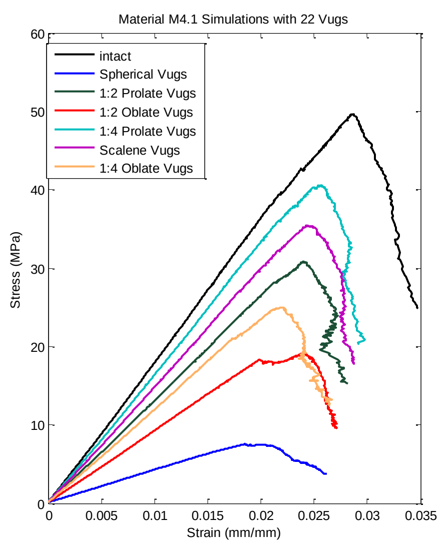

Material M3 Simulations with 22 Vugs
intact
Spherical Vugs
1:2 Prolate Vugs
1:2 Oblate Vugs
1:4 Prolate Vugs
Scalene Vugs
1:4 Oblate Vugs
Stress (MPa)
Strain (mm/mm)
x 10^-3

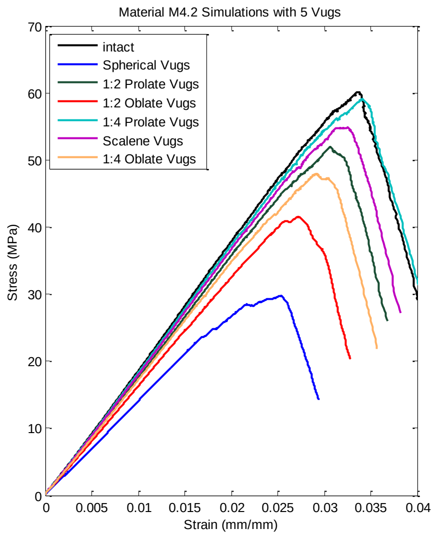

Material M4.1 Simulations with 5 Vugs
intact
Spherical Vugs
1:2 Prolate Vugs
1:2 Oblate Vugs
1:4 Prolate Vugs
Scalene Vugs
1:4 Oblate Vugs
Stress (MPa)
Strain (mm/mm)

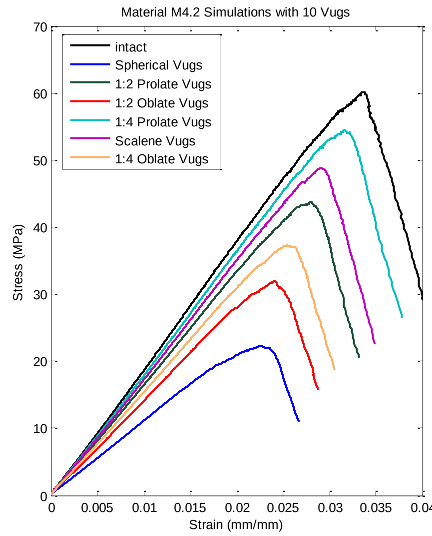

Material M4.1 Simulations with 10 Vugs
intact
Spherical Vugs
1:2 Prolate Vugs
1:2 Oblate Vugs
1:4 Prolate Vugs
Scalene Vugs
1:4 Oblate Vugs
Stress (MPa)
Strain (mm/mm)

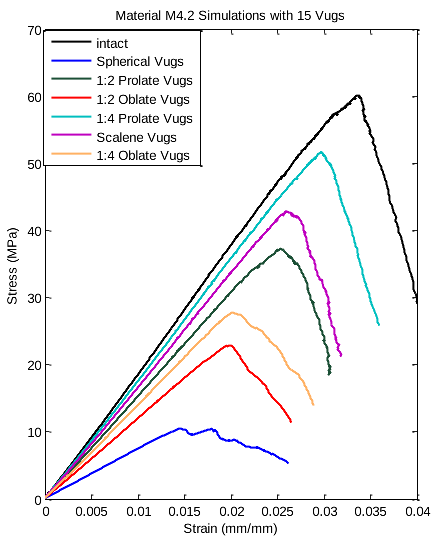

Material M4.1 Simulations with 15 Vugs
intact
Spherical Vugs
1:2 Prolate Vugs
1:2 Oblate Vugs
1:4 Prolate Vugs
Scalene Vugs
1:4 Oblate Vugs
Stress (MPa)
Strain (mm/mm)

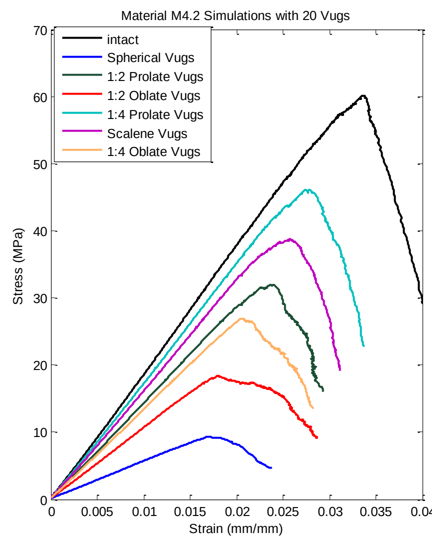

Material M4.1 Simulations with 20 Vugs
intact
Spherical Vugs
1:2 Prolate Vugs
1:2 Oblate Vugs
1:4 Prolate Vugs
Scalene Vugs
1:4 Oblate Vugs
Stress (MPa)
Strain (mm/mm)

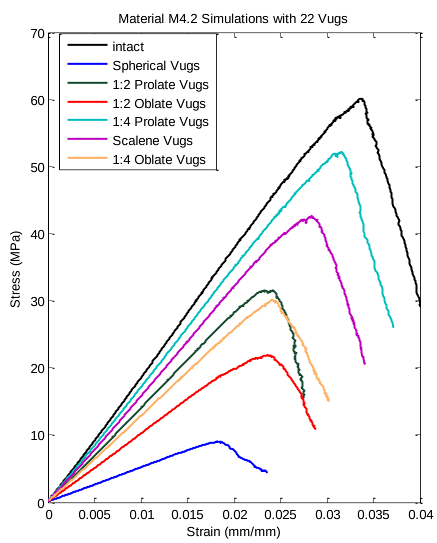

Material M4.1 Simulations with 22 Vugs
intact
Spherical Vugs
1:2 Prolate Vugs
1:2 Oblate Vugs
1:4 Prolate Vugs
Scalene Vugs
1:4 Oblate Vugs
Stress (MPa)
Strain (mm/mm)

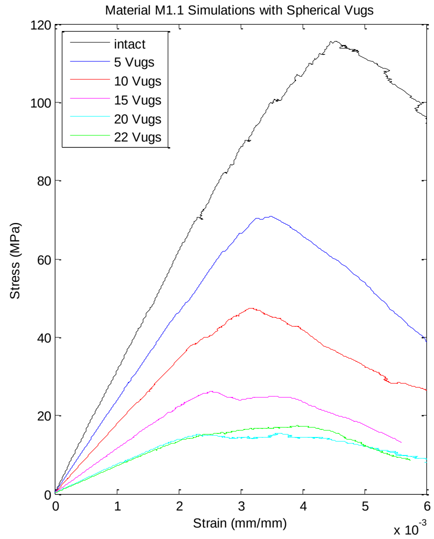

Material M4.2 Simulations with 5 Vugs
intact
Spherical Vugs
1:2 Prolate Vugs
1:2 Oblate Vugs
1:4 Prolate Vugs
Scalene Vugs
1:4 Oblate Vugs
Stress (MPa)
Strain (mm/mm)

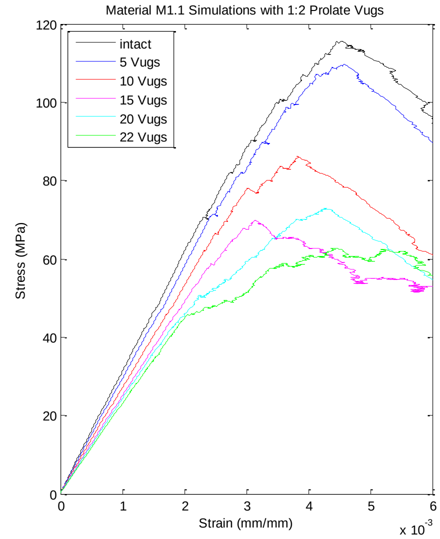

Material M4.2 Simulations with 10 Vugs
intact
Spherical Vugs
1:2 Prolate Vugs
1:2 Oblate Vugs
1:4 Prolate Vugs
Scalene Vugs
1:4 Oblate Vugs
Stress (MPa)
Strain (mm/mm)

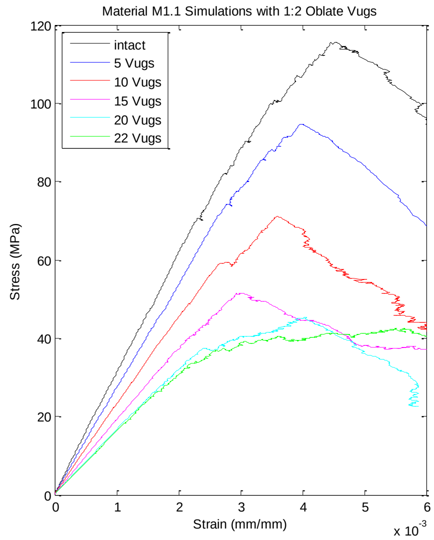

Material M4.2 Simulations with 15 Vugs
intact
Spherical Vugs
1:2 Prolate Vugs
1:2 Oblate Vugs
1:4 Prolate Vugs
Scalene Vugs
1:4 Oblate Vugs
Stress (MPa)
Strain (mm/mm)

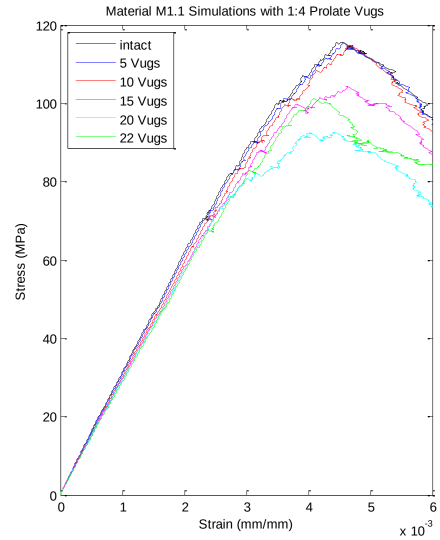

Material M4.2 Simulations with 20 Vugs
intact
Spherical Vugs
1:2 Prolate Vugs
1:2 Oblate Vugs
1:4 Prolate Vugs
Scalene Vugs
1:4 Oblate Vugs
Stress (MPa)
Strain (mm/mm)

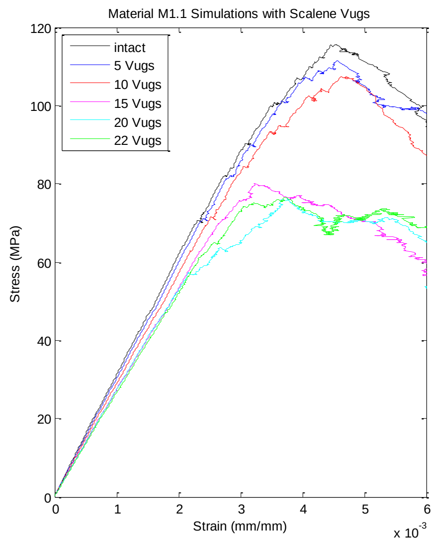

Material M4.2 Simulations with 22 Vugs
intact
Spherical Vugs
1:2 Prolate Vugs
1:2 Oblate Vugs
1:4 Prolate Vugs
Scalene Vugs
1:4 Oblate Vugs
Stress (MPa)
Strain (mm/mm)

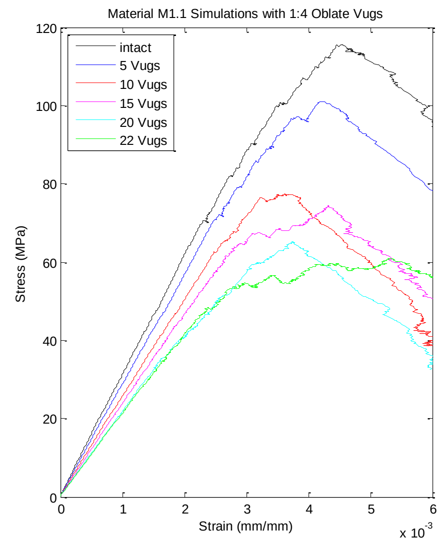

Material M1.1 Simulations with Spherical Vugs
intact
5 Vugs
10 Vugs
15 Vugs
20 Vugs
22 Vugs
Stress (MPa)
Strain (mm/mm)
x 10^-3

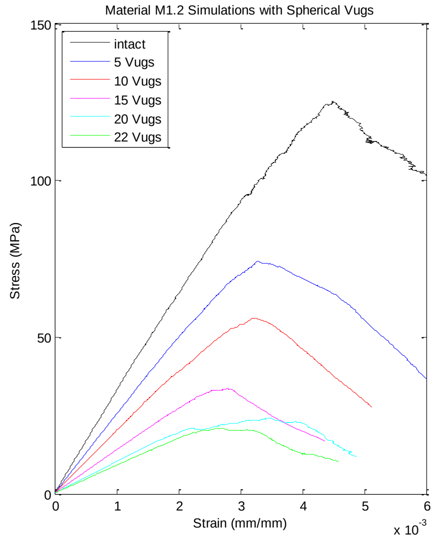

Material M1.1 Simulations with 1:2 Prolate Vugs
intact
5 Vugs
10 Vugs
15 Vugs
20 Vugs
22 Vugs
Stress (MPa)
Strain (mm/mm)
x 10^-3

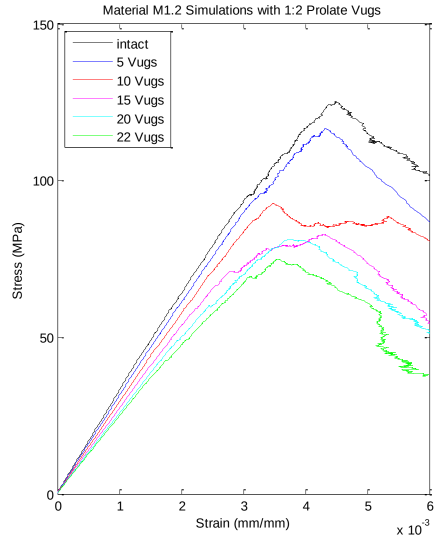

Material M1.1 Simulations with 1:2 Oblate Vugs
intact
5 Vugs
10 Vugs
15 Vugs
20 Vugs
22 Vugs
Stress (MPa)
Strain (mm/mm)
x 10^-3

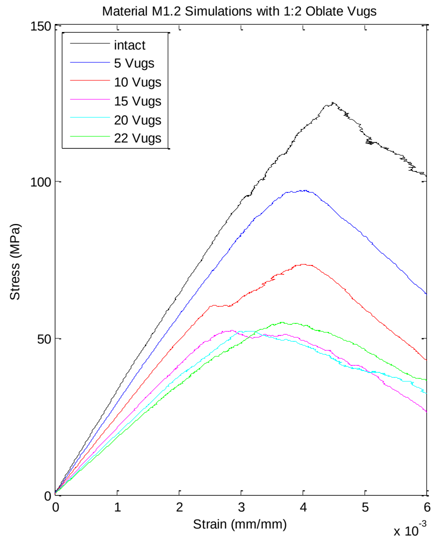

Material M1.1 Simulations with 1:4 Prolate Vugs
intact
5 Vugs
10 Vugs
15 Vugs
20 Vugs
22 Vugs
Stress (MPa)
Strain (mm/mm)
x 10^-3

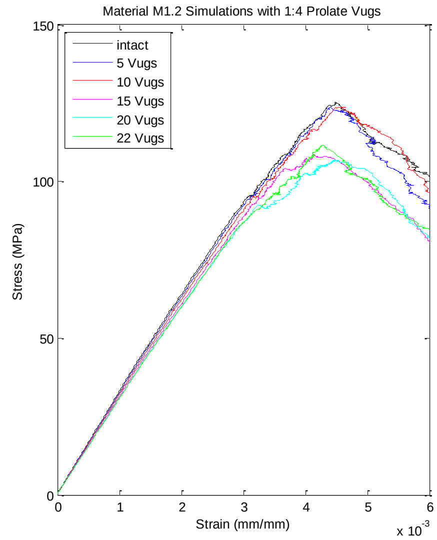

Material M1.1 Simulations with Scalene Vugs
intact
5 Vugs
10 Vugs
15 Vugs
20 Vugs
22 Vugs
Stress (MPa)
Strain (mm/mm)
x 10^-3

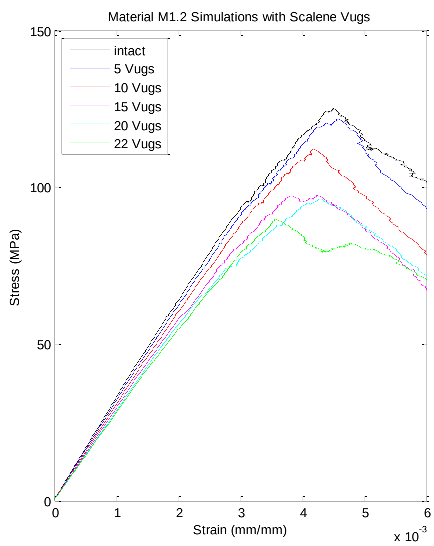

Material M1.1 Simulations with 1:4 Oblate Vugs
intact
5 Vugs
10 Vugs
15 Vugs
20 Vugs
22 Vugs
Stress (MPa)
Strain (mm/mm)
x 10^-3

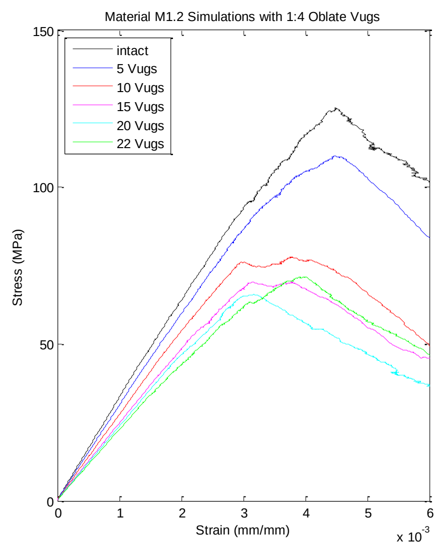

Material M1.2 Simulations with Spherical Vugs
Stress (MPa)
Strain (mm/mm)
x 10^-3
intact
5 Vugs
10 Vugs
15 Vugs
20 Vugs
22 Vugs

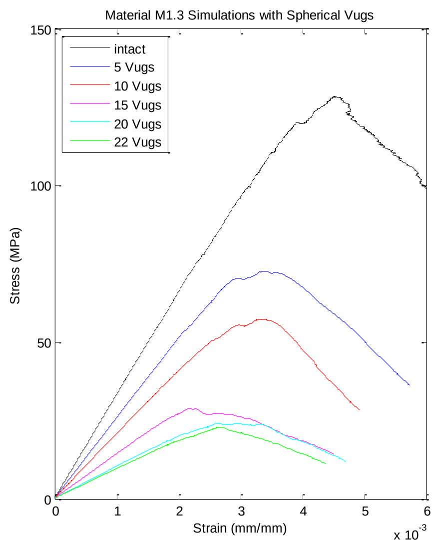

Material M1.2 Simulations with 1:2 Prolate Vugs
Stress (MPa)
Strain (mm/mm)
x 10^-3
intact
5 Vugs
10 Vugs
15 Vugs
20 Vugs
22 Vugs

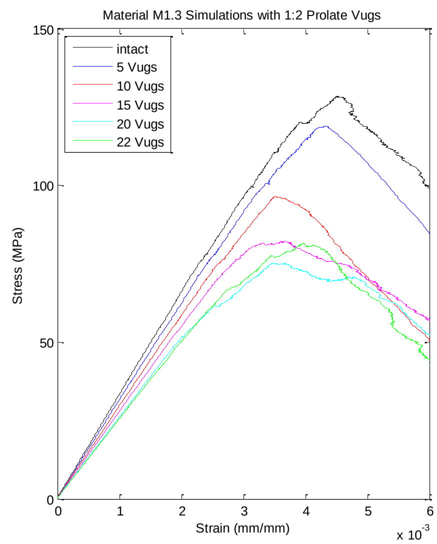

Material M1.2 Simulations with 1:2 Oblate Vugs
Stress (MPa)
Strain (mm/mm)
x 10^-3
intact
5 Vugs
10 Vugs
15 Vugs
20 Vugs
22 Vugs

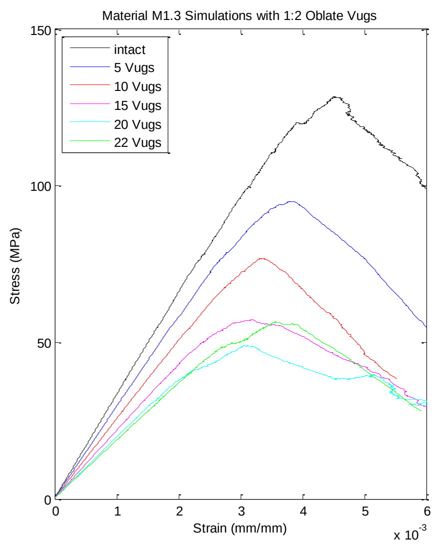

Material M1.2 Simulations with 1:4 Prolate Vugs
Stress (MPa)
Strain (mm/mm)
x 10^-3
intact
5 Vugs
10 Vugs
15 Vugs
20 Vugs
22 Vugs

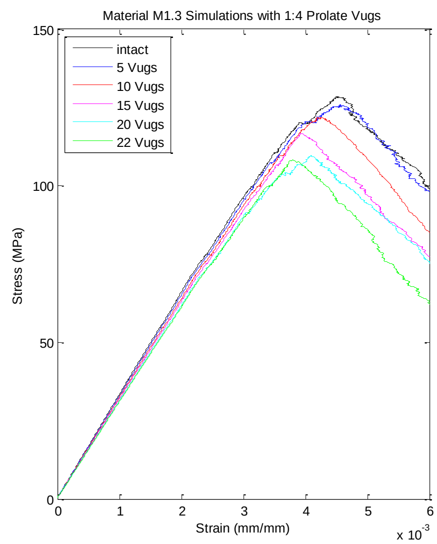

Material M1.2 Simulations with Scalene Vugs
Stress (MPa)
Strain (mm/mm)
x 10^-3
intact
5 Vugs
10 Vugs
15 Vugs
20 Vugs
22 Vugs

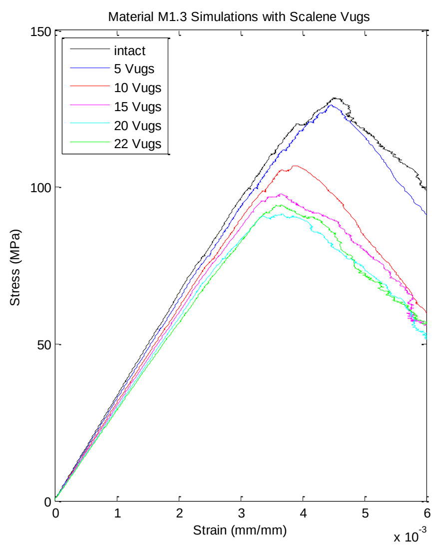

Material M1.2 Simulations with 1:4 Oblate Vugs
Stress (MPa)
Strain (mm/mm)
x 10^-3
intact
5 Vugs
10 Vugs
15 Vugs
20 Vugs
22 Vugs

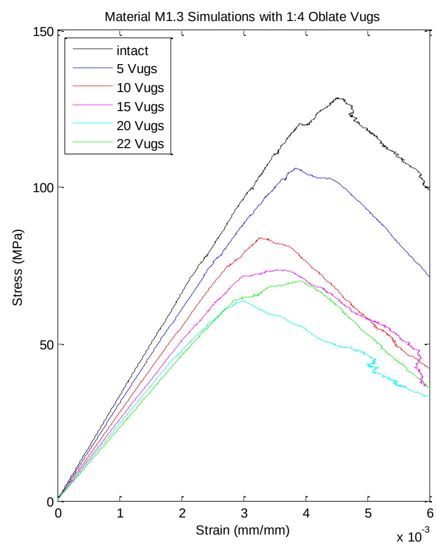

Material M1.3 Simulations with Spherical Vugs
intact
5 Vugs
10 Vugs
15 Vugs
20 Vugs
22 Vugs
Stress (MPa)
Strain (mm/mm)
x 10^-3

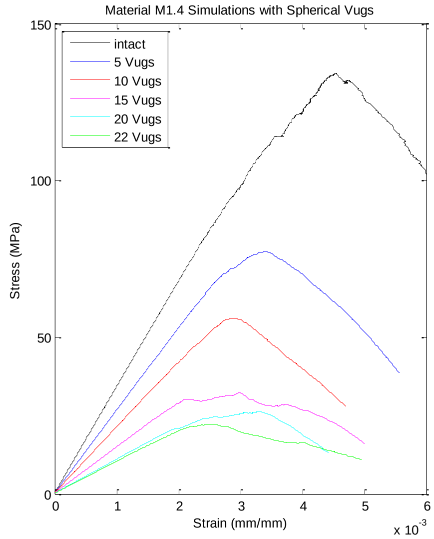

Material M1.3 Simulations with 1:2 Prolate Vugs
intact
5 Vugs
10 Vugs
15 Vugs
20 Vugs
22 Vugs
Stress (MPa)
Strain (mm/mm)
x 10^-3

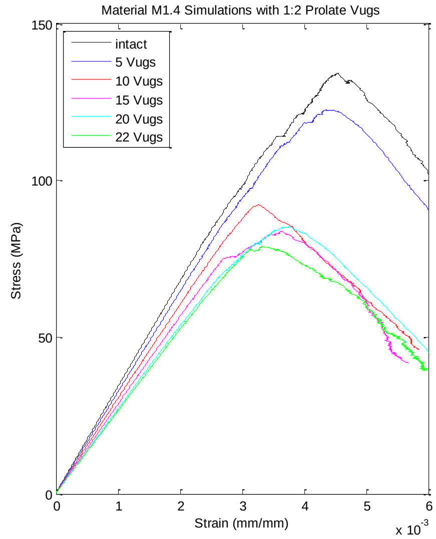

Material M1.3 Simulations with 1:2 Oblate Vugs
intact
5 Vugs
10 Vugs
15 Vugs
20 Vugs
22 Vugs
Stress (MPa)
Strain (mm/mm)
x 10^-3

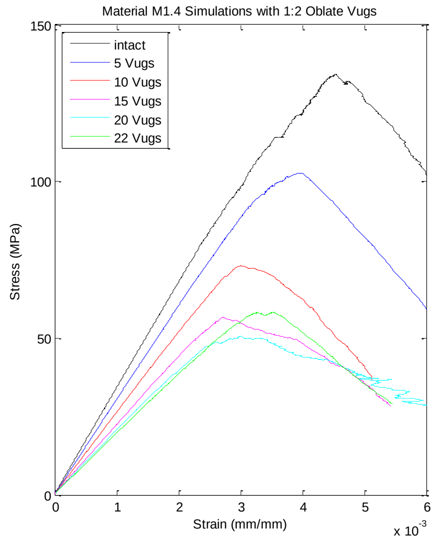

Material M1.3 Simulations with 1:4 Prolate Vugs
intact
5 Vugs
10 Vugs
15 Vugs
20 Vugs
22 Vugs
Stress (MPa)
Strain (mm/mm)
x 10^-3

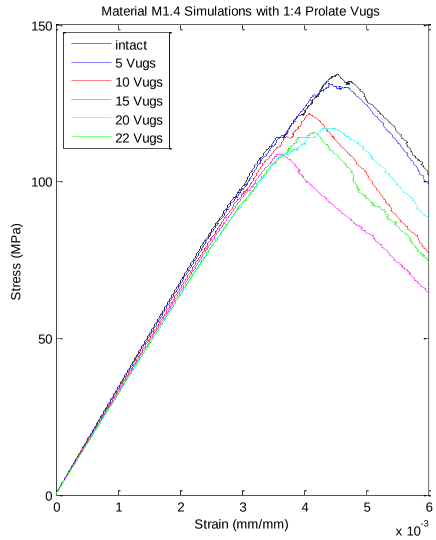

Material M1.3 Simulations with Scalene Vugs
intact
5 Vugs
10 Vugs
15 Vugs
20 Vugs
22 Vugs
Stress (MPa)
Strain (mm/mm)
x 10^-3

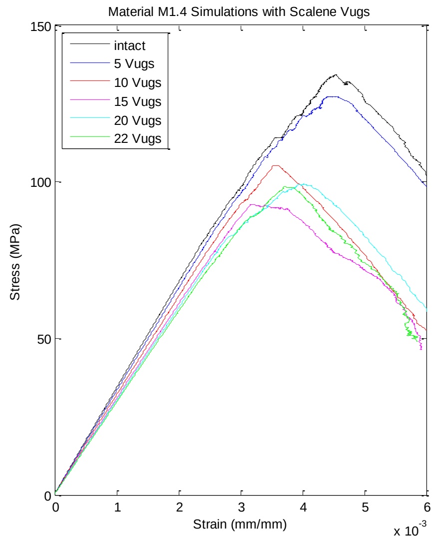

Material M1.3 Simulations with 1:4 Oblate Vugs
intact
5 Vugs
10 Vugs
15 Vugs
20 Vugs
22 Vugs
Stress (MPa)
Strain (mm/mm)
x 10^-3

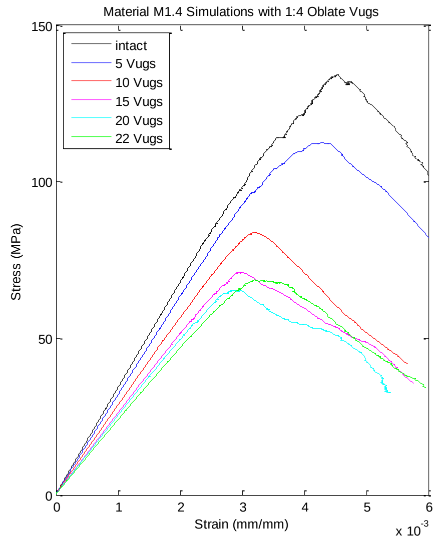

Material M1.4 Simulations with Spherical Vugs
Stress (MPa)
Strain (mm/mm)
x 10^-3
intact
5 Vugs
10 Vugs
15 Vugs
20 Vugs
22 Vugs

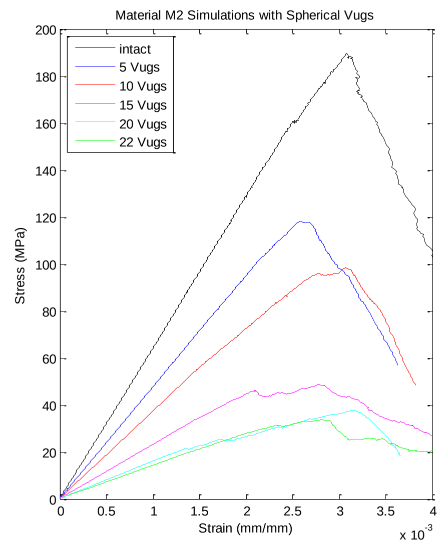

Material M1.4 Simulations with 1:2 Prolate Vugs
Stress (MPa)
Strain (mm/mm)
x 10^-3
intact
5 Vugs
10 Vugs
15 Vugs
20 Vugs
22 Vugs

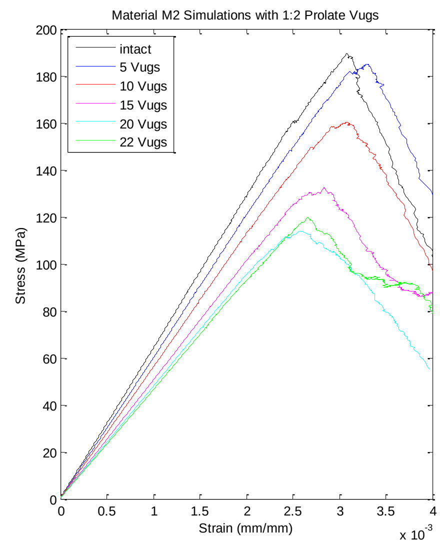

Material M1.4 Simulations with 1:2 Oblate Vugs
Stress (MPa)
Strain (mm/mm)
x 10^-3
intact
5 Vugs
10 Vugs
15 Vugs
20 Vugs
22 Vugs

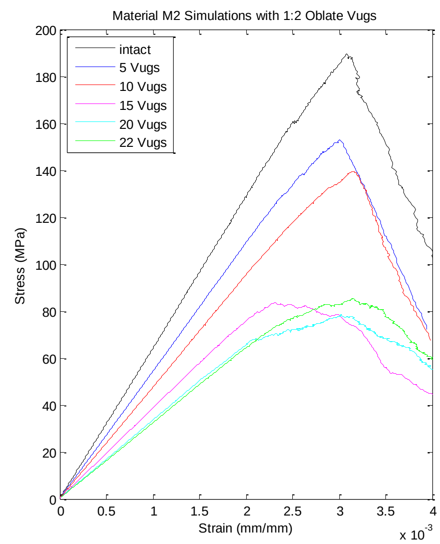

Material M1.4 Simulations with 1:4 Prolate Vugs
Stress (MPa)
Strain (mm/mm)
x 10^-3
intact
5 Vugs
10 Vugs
15 Vugs
20 Vugs
22 Vugs

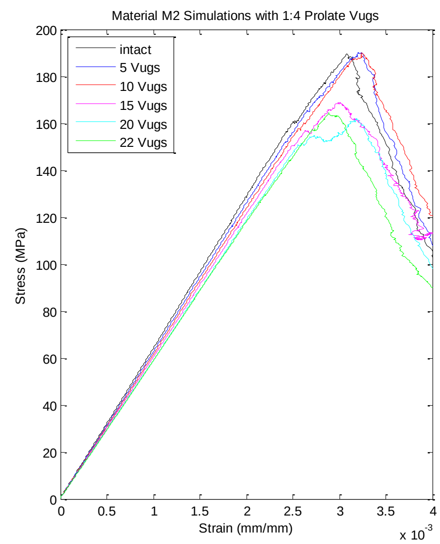

Material M1.4 Simulations with Scalene Vugs
Stress (MPa)
Strain (mm/mm)
x 10^-3
intact
5 Vugs
10 Vugs
15 Vugs
20 Vugs
22 Vugs

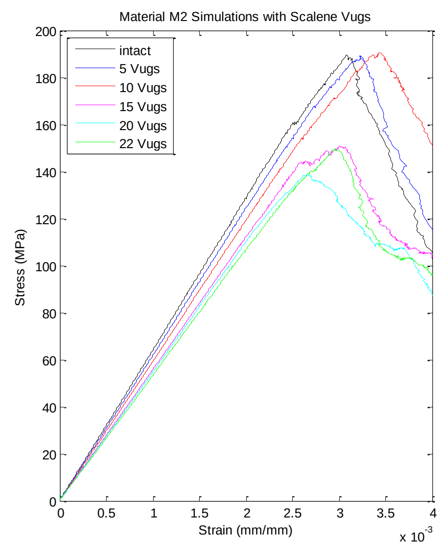

Material M1.4 Simulations with 1:4 Oblate Vugs
Stress (MPa)
Strain (mm/mm)
x 10^-3
intact
5 Vugs
10 Vugs
15 Vugs
20 Vugs
22 Vugs

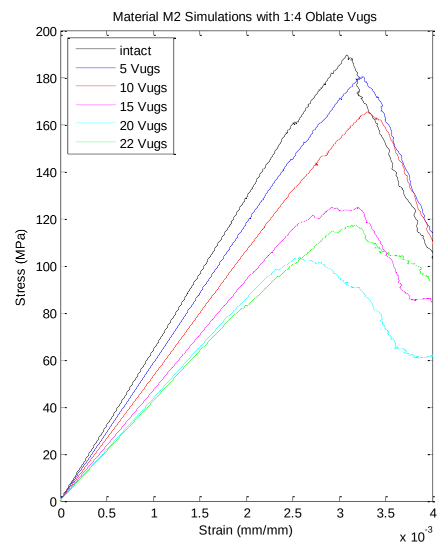

Material M2 Simulations with Spherical Vugs
Stress (MPa)
Strain (mm/mm)
x 10-3
intact
5 Vugs
10 Vugs
15 Vugs
20 Vugs
22 Vugs
0
20
40
60
80
100
120
140
160
180
200
0
0.5
1
1.5
2
2.5
3
3.5
4

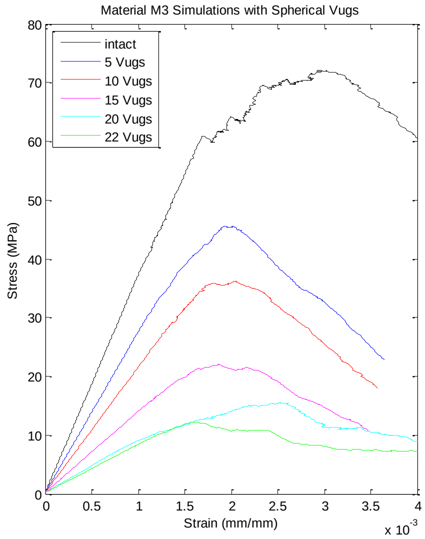

Material M2 Simulations with 1:2 Prolate Vugs
Stress (MPa)
Strain (mm/mm)
x 10-3
intact
5 Vugs
10 Vugs
15 Vugs
20 Vugs
22 Vugs
0
20
40
60
80
100
120
140
160
180
200
0
0.5
1
1.5
2
2.5
3
3.5
4

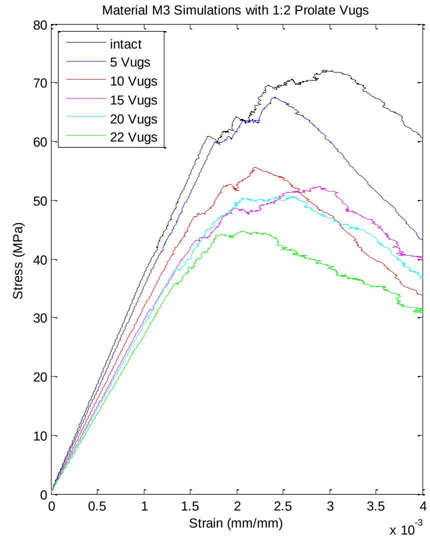

Material M2 Simulations with 1:2 Oblate Vugs
Stress (MPa)
Strain (mm/mm)
x 10-3
intact
5 Vugs
10 Vugs
15 Vugs
20 Vugs
22 Vugs
0
20
40
60
80
100
120
140
160
180
200
0
0.5
1
1.5
2
2.5
3
3.5
4

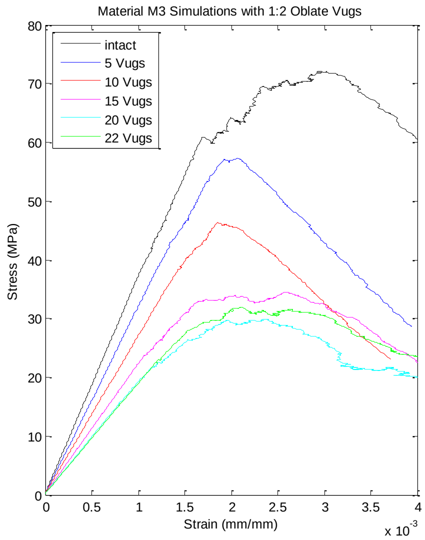

Material M2 Simulations with 1:4 Prolate Vugs
Stress (MPa)
Strain (mm/mm)
x 10-3
intact
5 Vugs
10 Vugs
15 Vugs
20 Vugs
22 Vugs
0
20
40
60
80
100
120
140
160
180
200
0
0.5
1
1.5
2
2.5
3
3.5
4

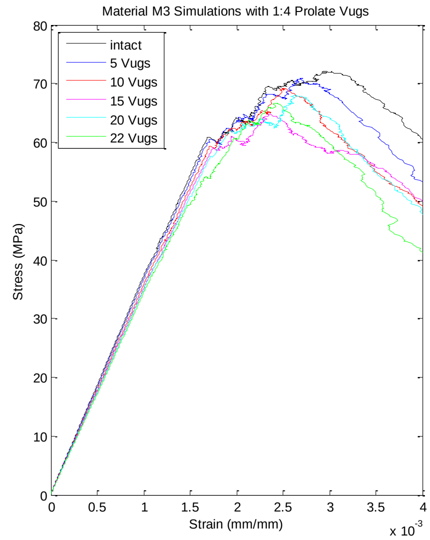

Material M2 Simulations with Scalene Vugs
Stress (MPa)
Strain (mm/mm)
x 10-3
intact
5 Vugs
10 Vugs
15 Vugs
20 Vugs
22 Vugs
0
20
40
60
80
100
120
140
160
180
200
0
0.5
1
1.5
2
2.5
3
3.5
4

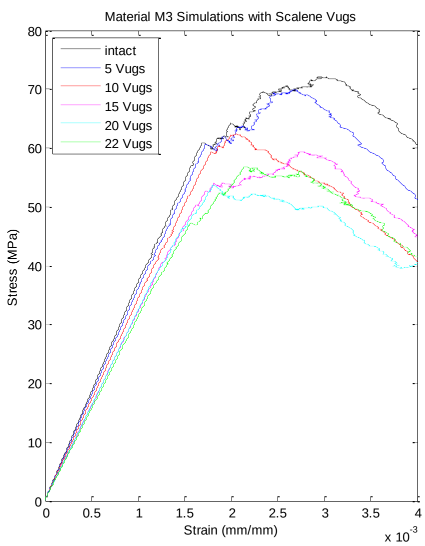

Material M2 Simulations with 1:4 Oblate Vugs
Stress (MPa)
Strain (mm/mm)
x 10-3
intact
5 Vugs
10 Vugs
15 Vugs
20 Vugs
22 Vugs
0
20
40
60
80
100
120
140
160
180
200
0
0.5
1
1.5
2
2.5
3
3.5
4

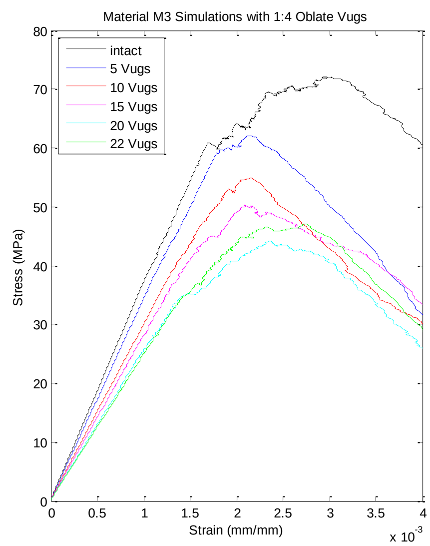

Material M3 Simulations with Spherical Vugs
intact
5 Vugs
10 Vugs
15 Vugs
20 Vugs
22 Vugs
Stress (MPa)
Strain (mm/mm)
x 10^-3

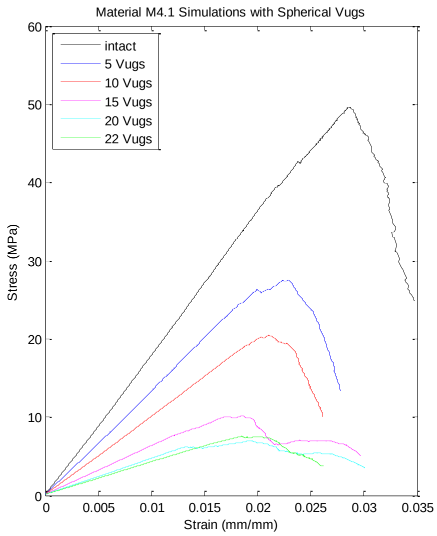

Material M3 Simulations with 1:2 Prolate Vugs
intact
5 Vugs
10 Vugs
15 Vugs
20 Vugs
22 Vugs
Stress (MPa)
Strain (mm/mm)
x 10^-3

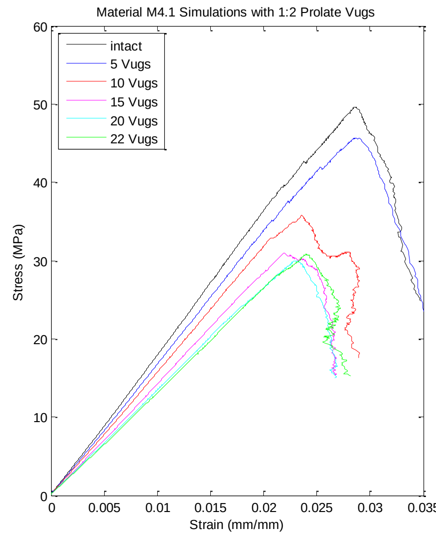

Material M3 Simulations with 1:2 Oblate Vugs
intact
5 Vugs
10 Vugs
15 Vugs
20 Vugs
22 Vugs
Stress (MPa)
Strain (mm/mm)
x 10^-3

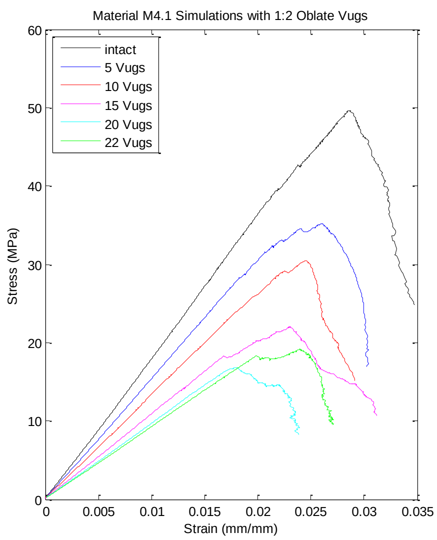

Material M3 Simulations with 1:4 Prolate Vugs
intact
5 Vugs
10 Vugs
15 Vugs
20 Vugs
22 Vugs
Stress (MPa)
Strain (mm/mm)
x 10^-3

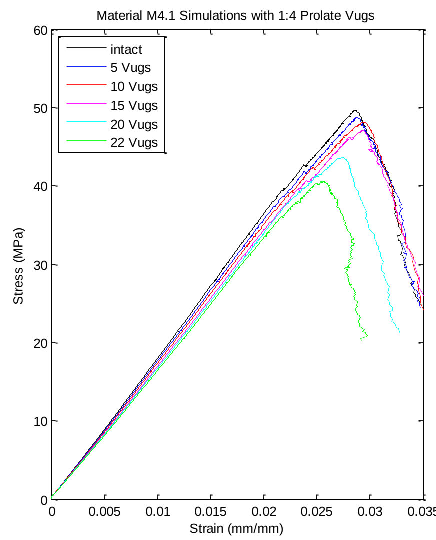

Material M3 Simulations with Scalene Vugs
intact
5 Vugs
10 Vugs
15 Vugs
20 Vugs
22 Vugs
Stress (MPa)
Strain (mm/mm)
x 10^-3

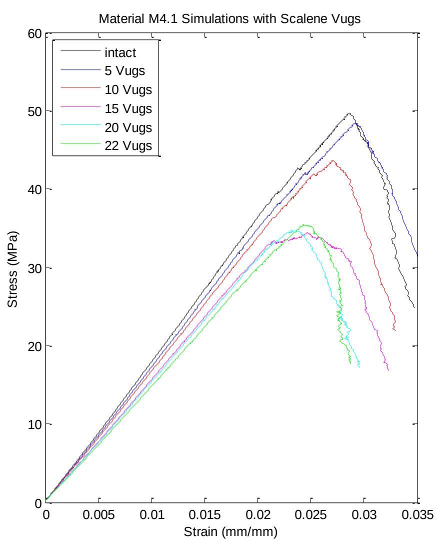

Material M3 Simulations with 1:4 Oblate Vugs
intact
5 Vugs
10 Vugs
15 Vugs
20 Vugs
22 Vugs
Stress (MPa)
Strain (mm/mm)
x 10^-3

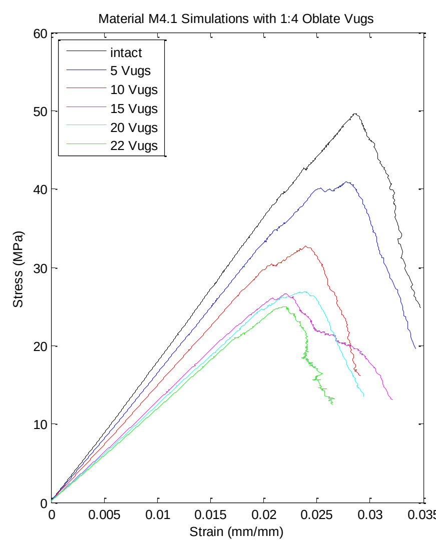

Material M4.1 Simulations with Spherical Vugs
intact
5 Vugs
10 Vugs
15 Vugs
20 Vugs
22 Vugs
Stress (MPa)
Strain (mm/mm)

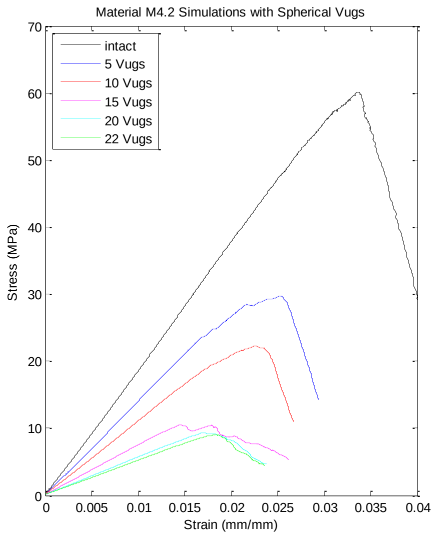

Material M4.1 Simulations with 1:2 Prolate Vugs
intact
5 Vugs
10 Vugs
15 Vugs
20 Vugs
22 Vugs
Stress (MPa)
Strain (mm/mm)

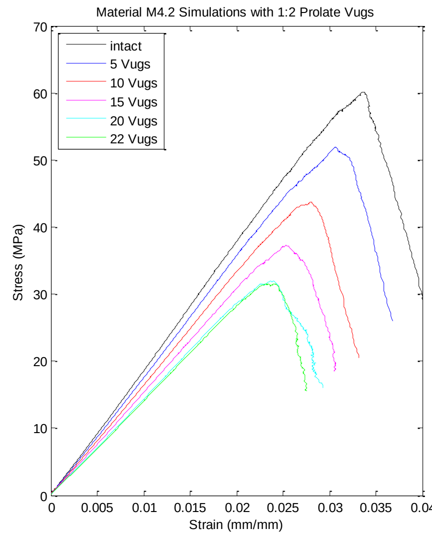

Material M4.1 Simulations with 1:2 Oblate Vugs
intact
5 Vugs
10 Vugs
15 Vugs
20 Vugs
22 Vugs
Stress (MPa)
Strain (mm/mm)

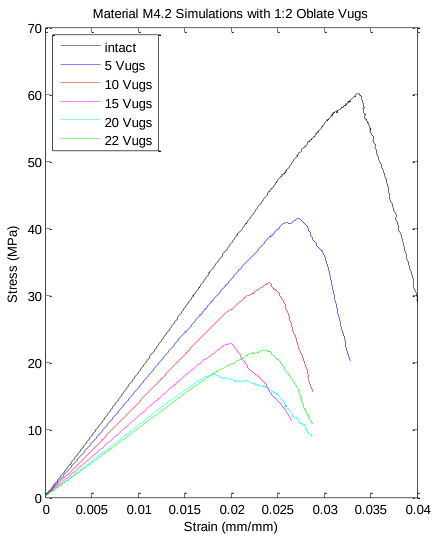

Material M4.1 Simulations with 1:4 Prolate Vugs
intact
5 Vugs
10 Vugs
15 Vugs
20 Vugs
22 Vugs
Stress (MPa)
Strain (mm/mm)

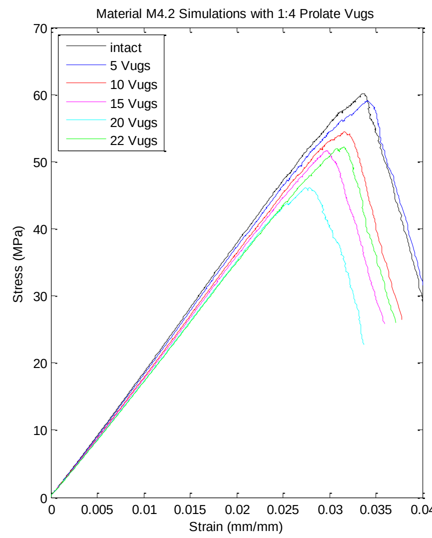

Material M4.1 Simulations with Scalene Vugs
intact
5 Vugs
10 Vugs
15 Vugs
20 Vugs
22 Vugs
Stress (MPa)
Strain (mm/mm)

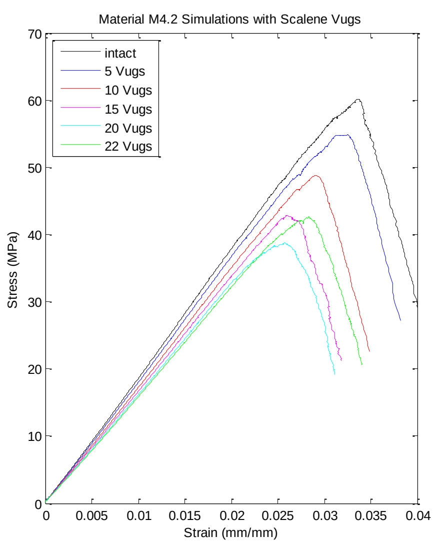

Material M4.1 Simulations with 1:4 Oblate Vugs
intact
5 Vugs
10 Vugs
15 Vugs
20 Vugs
22 Vugs
Stress (MPa)
Strain (mm/mm)

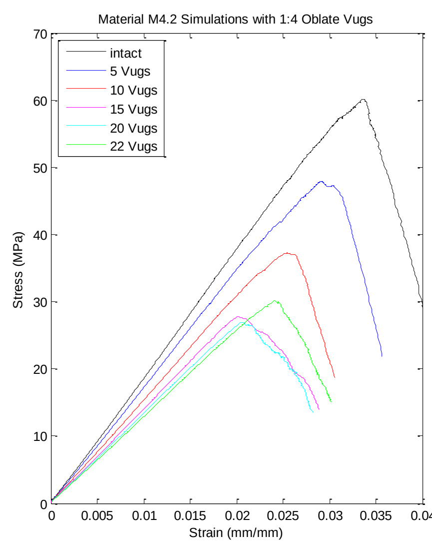

Material M4.2 Simulations with Spherical Vugs
intact
5 Vugs
10 Vugs
15 Vugs
20 Vugs
22 Vugs
Stress (MPa)
Strain (mm/mm)

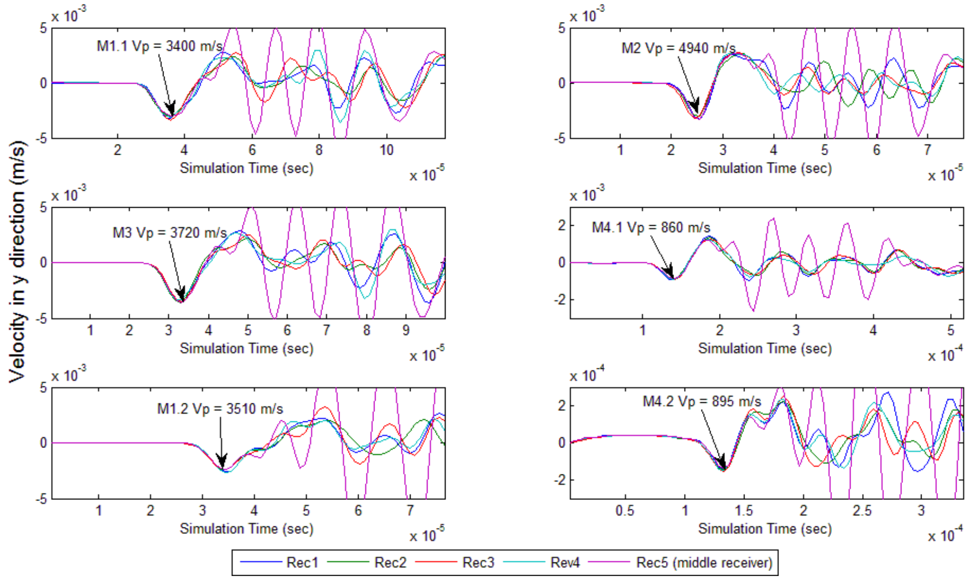

Material M4.2 Simulations with 1:2 Prolate Vugs
intact
5 Vugs
10 Vugs
15 Vugs
20 Vugs
22 Vugs
Stress (MPa)
Strain (mm/mm)


Material M4.2 Simulations with 1:2 Oblate Vugs
intact
5 Vugs
10 Vugs
15 Vugs
20 Vugs
22 Vugs
Stress (MPa)
Strain (mm/mm)

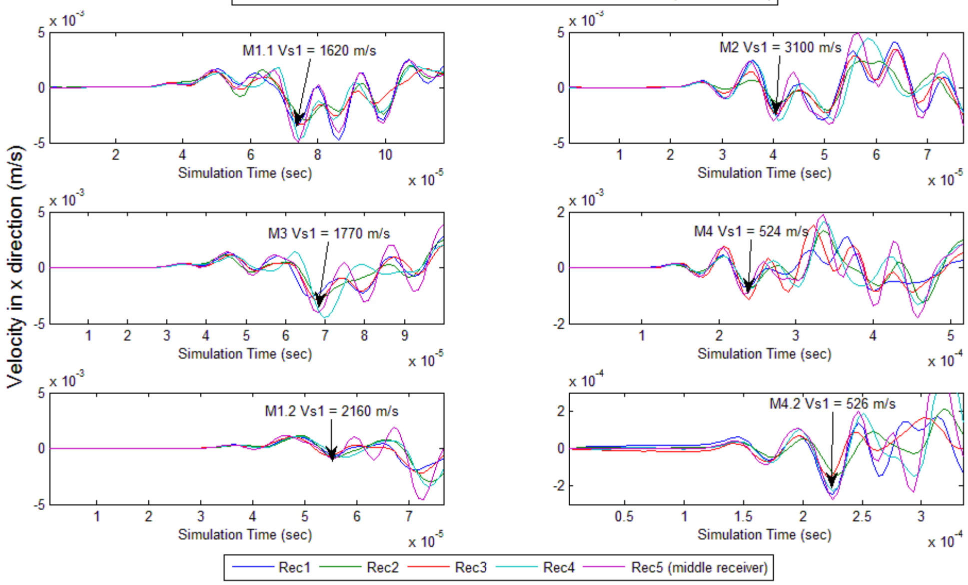

Material M4.2 Simulations with 1:4 Prolate Vugs
intact
5 Vugs
10 Vugs
15 Vugs
20 Vugs
22 Vugs
Stress (MPa)
Strain (mm/mm)


Material M4.2 Simulations with Scalene Vugs
intact
5 Vugs
10 Vugs
15 Vugs
20 Vugs
22 Vugs
Stress (MPa)
Strain (mm/mm)

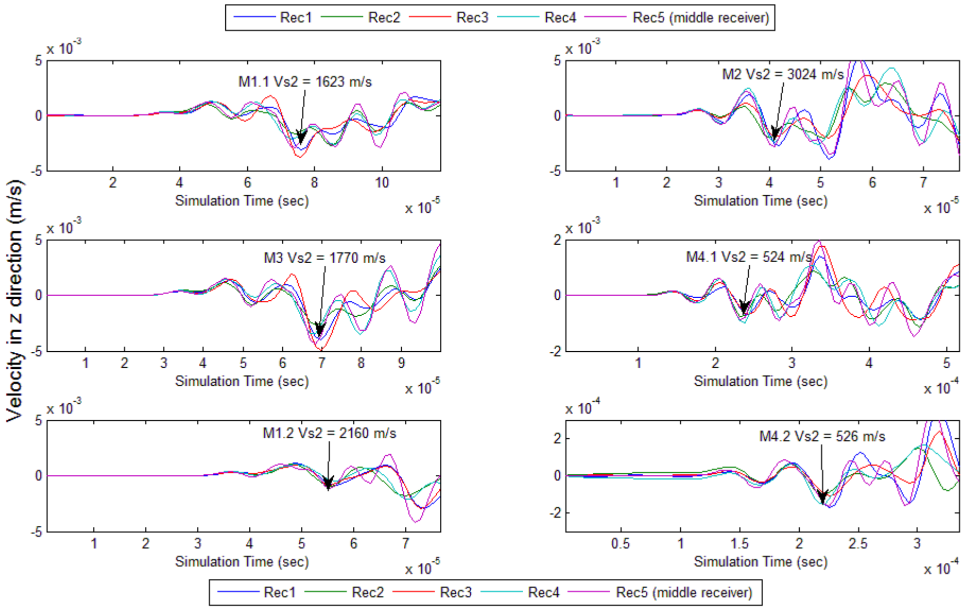

Material M4.2 Simulations with 1:4 Oblate Vugs
intact
5 Vugs
10 Vugs
15 Vugs
20 Vugs
22 Vugs
Stress (MPa)
Strain (mm/mm)



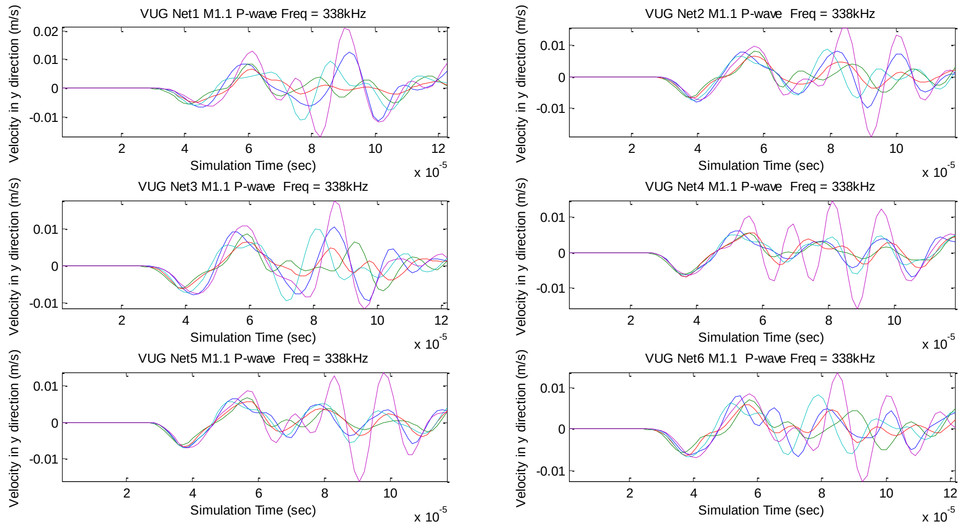
564kHz

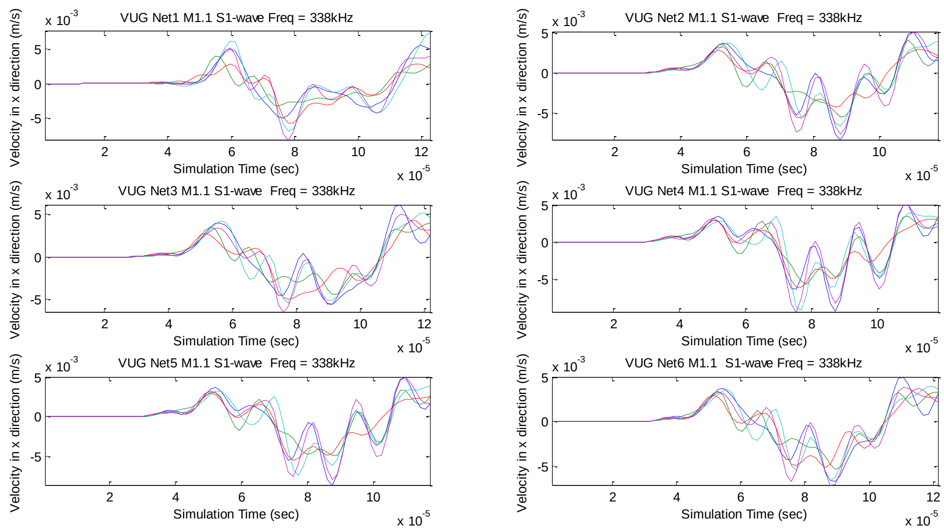


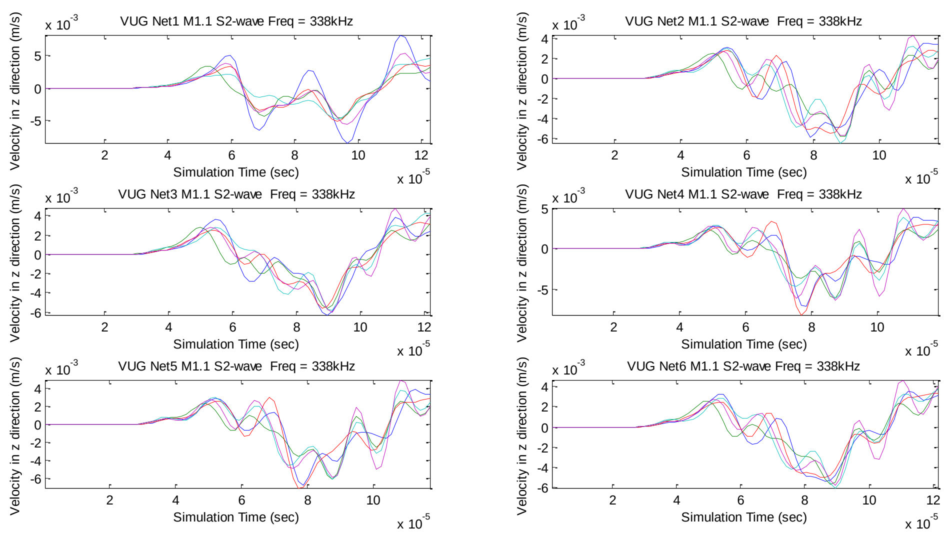
564kHz

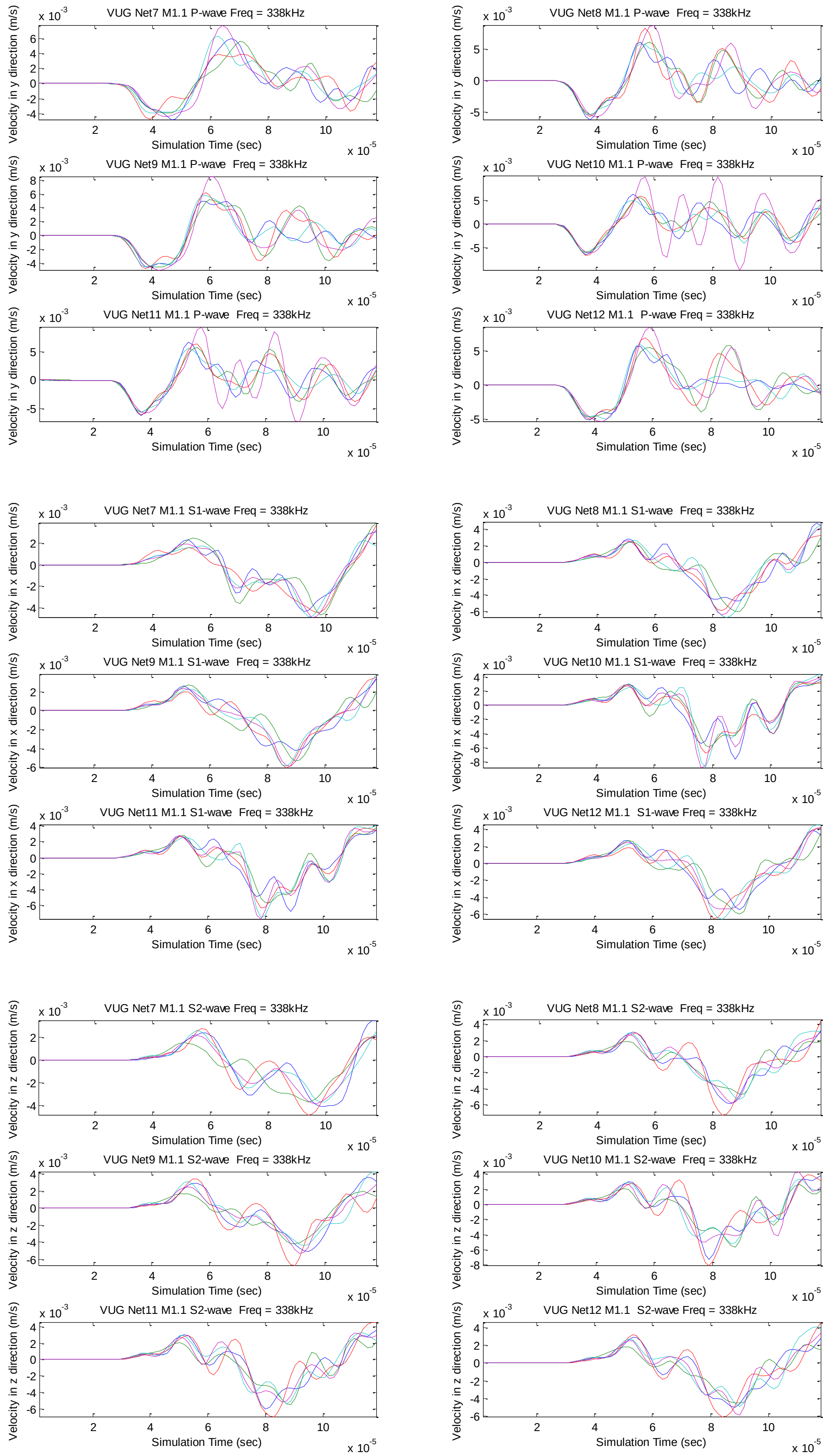


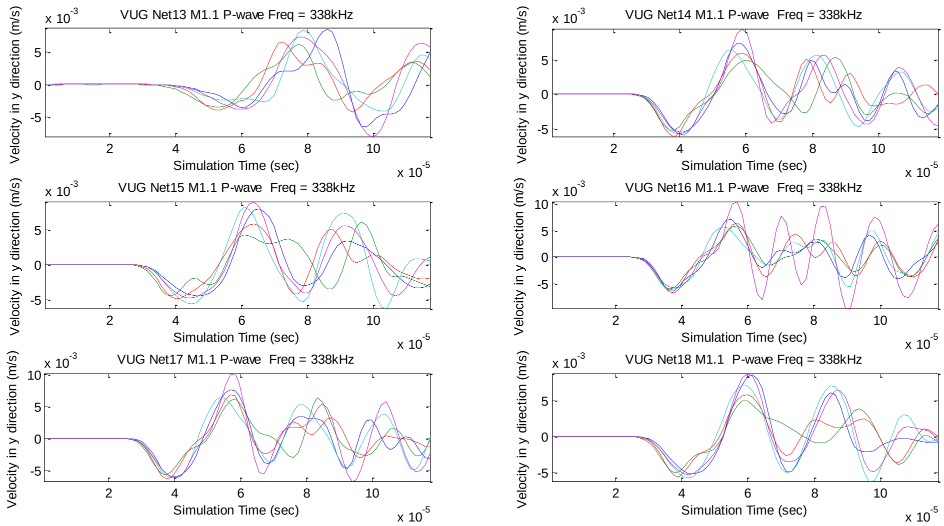
564kHz

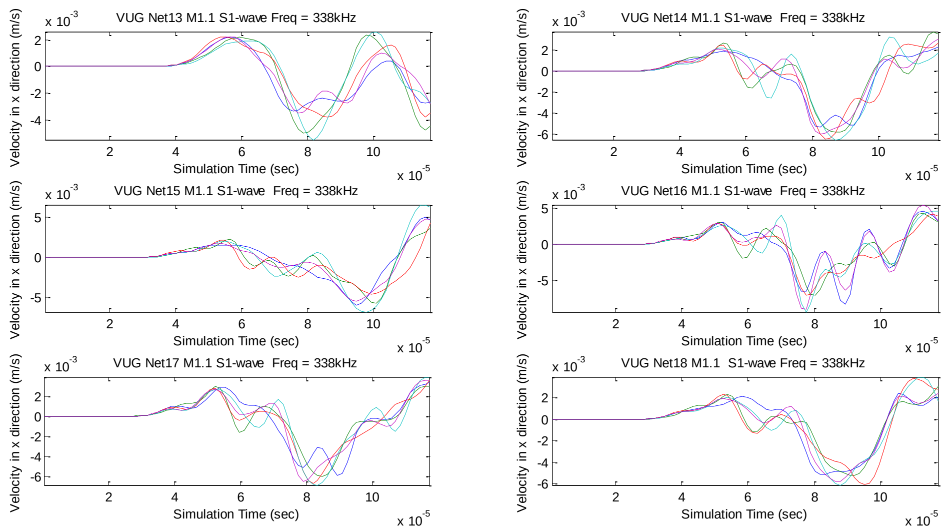

VUG Net1 M1.1 P-wave Freq = 338kHz
VUG Net2 M1.1 P-wave Freq = 338kHz
VUG Net3 M1.1 P-wave Freq = 338kHz
VUG Net4 M1.1 P-wave Freq = 338kHz
VUG Net5 M1.1 P-wave Freq = 338kHz
VUG Net6 M1.1 P-wave Freq = 338kHz
Velocity in y direction (m/s)
Simulation Time (sec)
x 10^-5

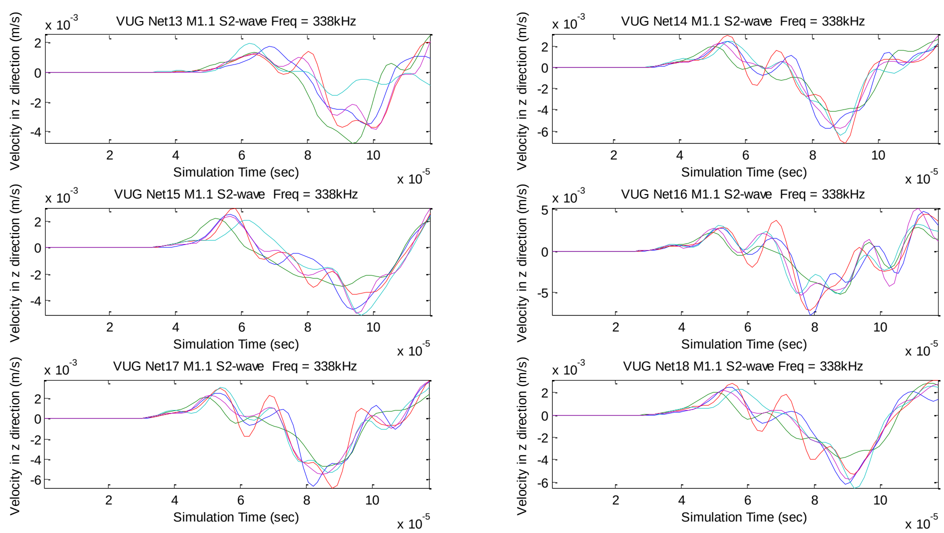

VUG Net1 M1.1 S1-wave Freq = 338kHz
VUG Net2 M1.1 S1-wave Freq = 338kHz
VUG Net3 M1.1 S1-wave Freq = 338kHz
VUG Net4 M1.1 S1-wave Freq = 338kHz
VUG Net5 M1.1 S1-wave Freq = 338kHz
VUG Net6 M1.1 S1-wave Freq = 338kHz
Velocity in x direction (m/s)
x 10^-3
Simulation Time (sec)
x 10^-5

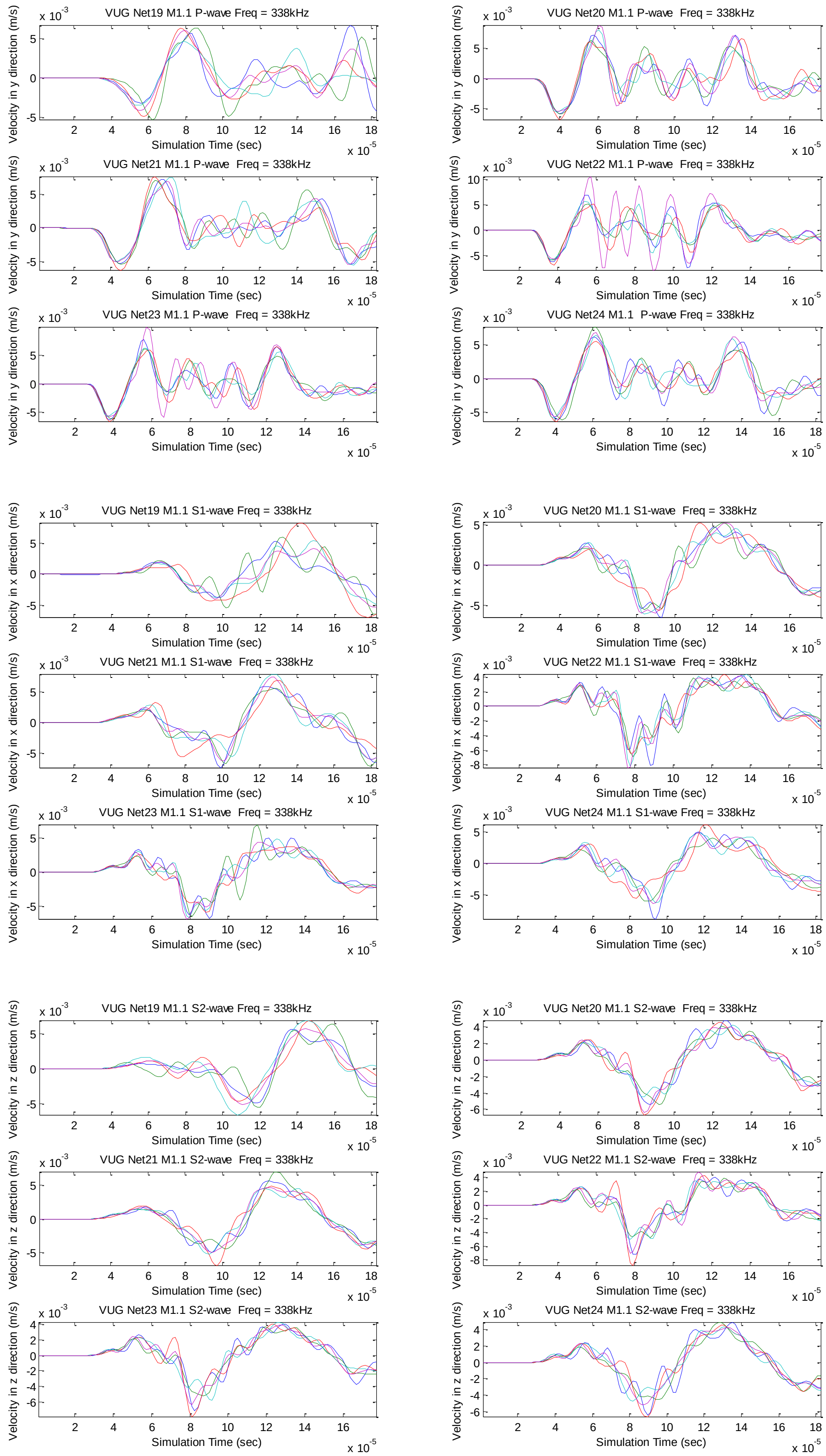

VUG Net1 M1.1 S2-wave Freq = 338kHz
VUG Net2 M1.1 S2-wave Freq = 338kHz
VUG Net3 M1.1 S2-wave Freq = 338kHz
VUG Net4 M1.1 S2-wave Freq = 338kHz
VUG Net5 M1.1 S2-wave Freq = 338kHz
VUG Net6 M1.1 S2-wave Freq = 338kHz
Velocity in z direction (m/s)
x 10^-3
Simulation Time (sec)
x 10^-5

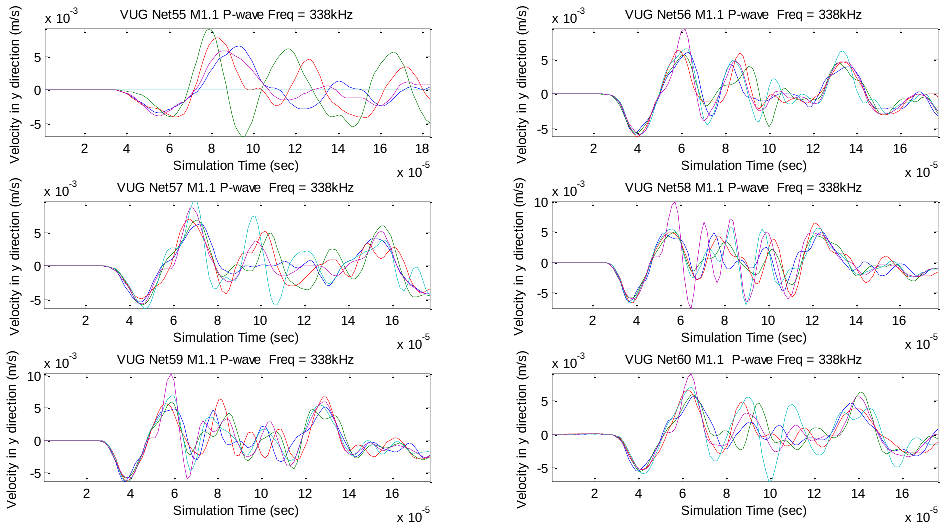

VUG Net7 M1.1 P-wave Freq = 338kHz
VUG Net8 M1.1 P-wave Freq = 338kHz
VUG Net9 M1.1 P-wave Freq = 338kHz
VUG Net10 M1.1 P-wave Freq = 338kHz
VUG Net11 M1.1 P-wave Freq = 338kHz
VUG Net12 M1.1 P-wave Freq = 338kHz
Velocity in y direction (m/s)
VUG Net7 M1.1 S1-wave Freq = 338kHz
VUG Net8 M1.1 S1-wave Freq = 338kHz
VUG Net9 M1.1 S1-wave Freq = 338kHz
VUG Net10 M1.1 S1-wave Freq = 338kHz
VUG Net11 M1.1 S1-wave Freq = 338kHz
VUG Net12 M1.1 S1-wave Freq = 338kHz
Velocity in x direction (m/s)
VUG Net7 M1.1 S2-wave Freq = 338kHz
VUG Net8 M1.1 S2-wave Freq = 338kHz
VUG Net9 M1.1 S2-wave Freq = 338kHz
VUG Net10 M1.1 S2-wave Freq = 338kHz
VUG Net11 M1.1 S2-wave Freq = 338kHz
VUG Net12 M1.1 S2-wave Freq = 338kHz
Velocity in z direction (m/s)
Simulation Time (sec)
x 10^-3
x 10^-5

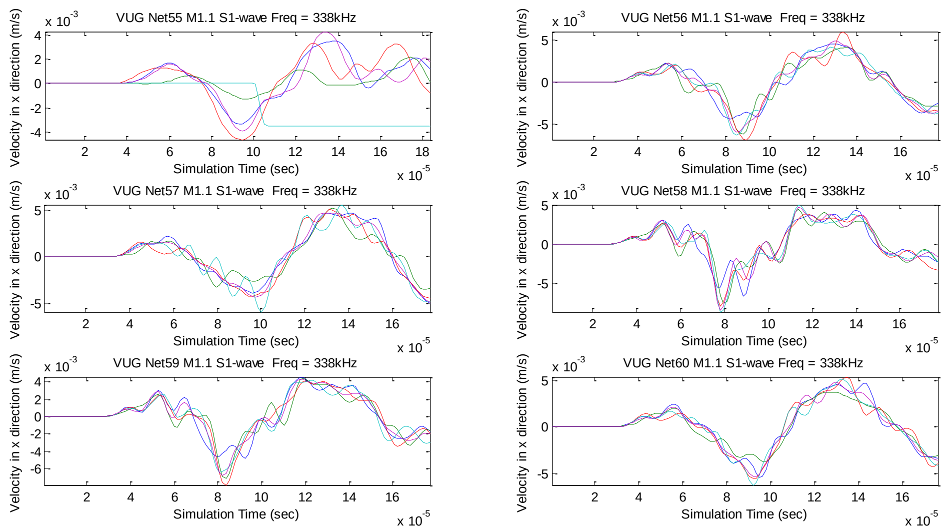

VUG Net13 M1.1 P-wave Freq = 338kHz
VUG Net14 M1.1 P-wave Freq = 338kHz
VUG Net15 M1.1 P-wave Freq = 338kHz
VUG Net16 M1.1 P-wave Freq = 338kHz
VUG Net17 M1.1 P-wave Freq = 338kHz
VUG Net18 M1.1 P-wave Freq = 338kHz
Velocity in y direction (m/s)
x 10^-3
Simulation Time (sec)
x 10^-5

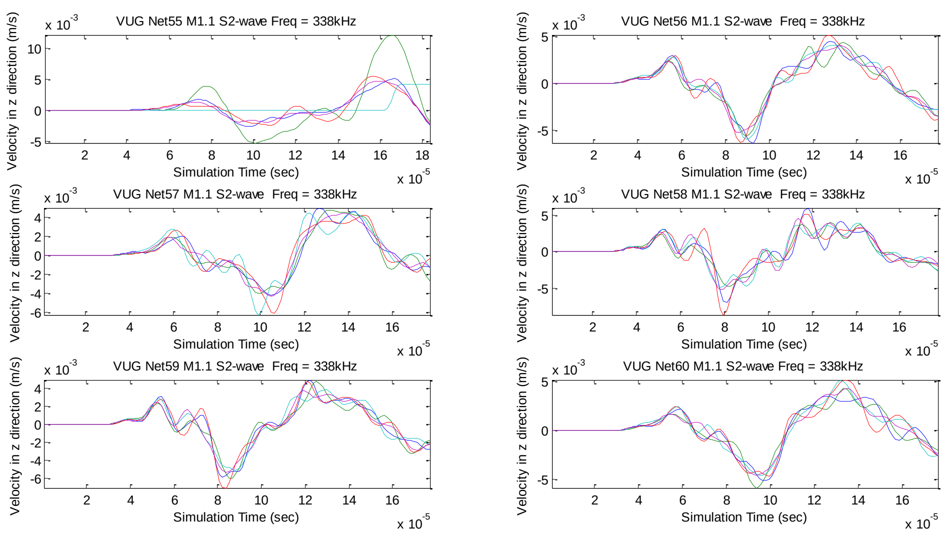

VUG Net13 M1.1 S1-wave Freq = 338kHz
VUG Net14 M1.1 S1-wave Freq = 338kHz
VUG Net15 M1.1 S1-wave Freq = 338kHz
VUG Net16 M1.1 S1-wave Freq = 338kHz
VUG Net17 M1.1 S1-wave Freq = 338kHz
VUG Net18 M1.1 S1-wave Freq = 338kHz
Velocity in x direction (m/s)
x 10^-3
Simulation Time (sec)
x 10^-5

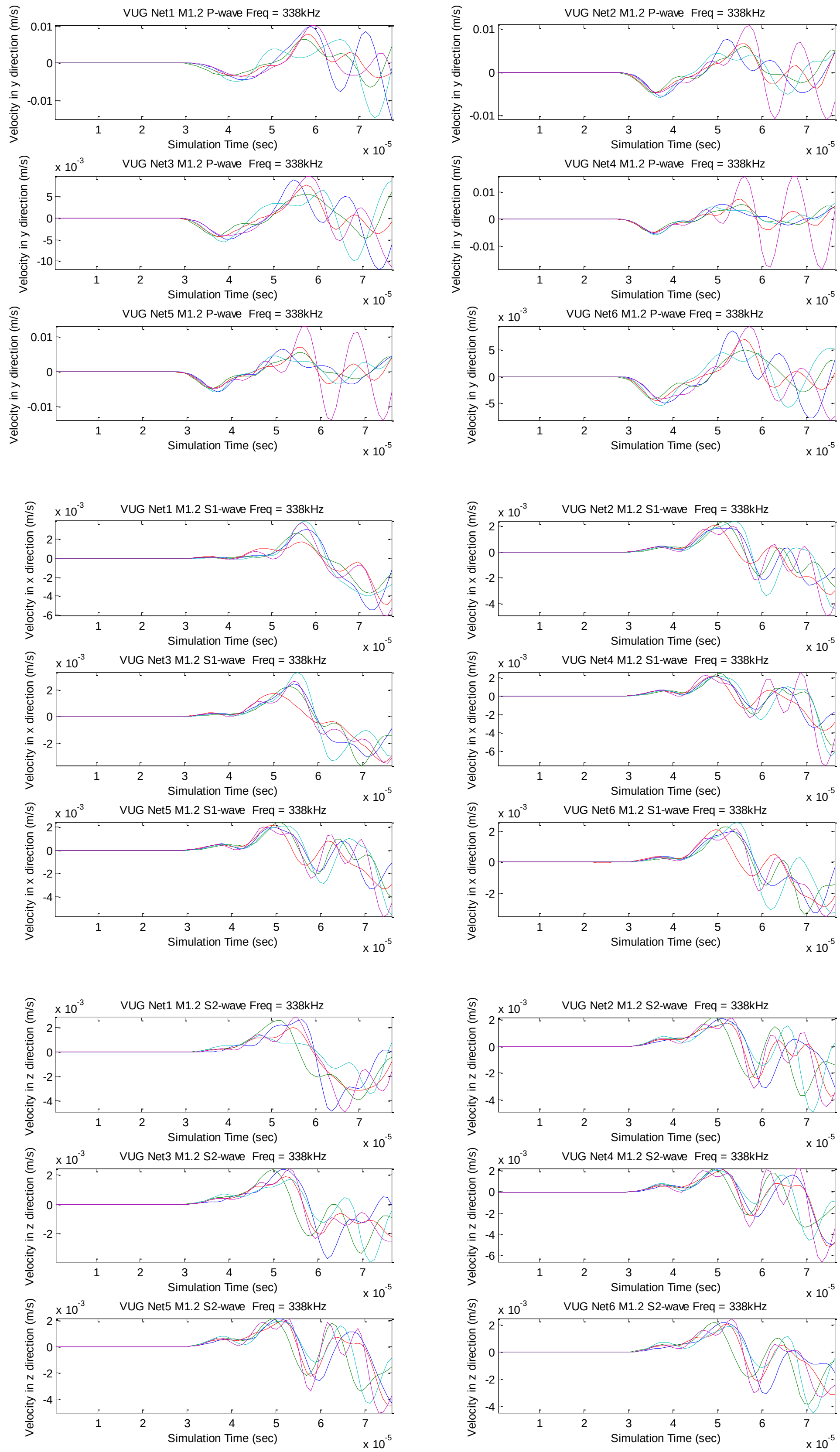

VUG Net13 M1.1 S2-wave Freq = 338kHz
VUG Net14 M1.1 S2-wave Freq = 338kHz
VUG Net15 M1.1 S2-wave Freq = 338kHz
VUG Net16 M1.1 S2-wave Freq = 338kHz
VUG Net17 M1.1 S2-wave Freq = 338kHz
VUG Net18 M1.1 S2-wave Freq = 338kHz
Velocity in z direction (m/s)
x 10^-3
Simulation Time (sec)
x 10^-5

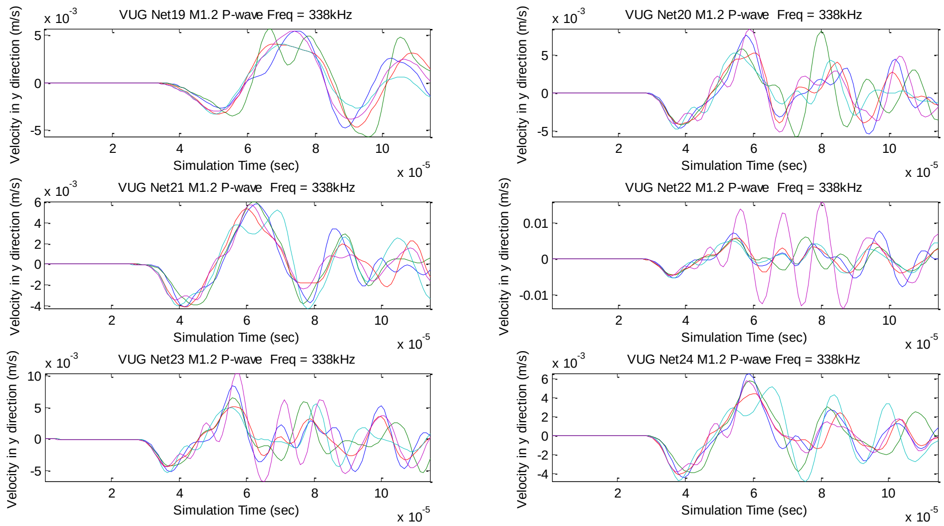

VUG Net19 M1.1 P-wave Freq = 338kHz
VUG Net20 M1.1 P-wave Freq = 338kHz
VUG Net21 M1.1 P-wave Freq = 338kHz
VUG Net22 M1.1 P-wave Freq = 338kHz
VUG Net23 M1.1 P-wave Freq = 338kHz
VUG Net24 M1.1 P-wave Freq = 338kHz
Velocity in y direction (m/s)
VUG Net19 M1.1 S1-wave Freq = 338kHz
VUG Net20 M1.1 S1-wave Freq = 338kHz
VUG Net21 M1.1 S1-wave Freq = 338kHz
VUG Net22 M1.1 S1-wave Freq = 338kHz
VUG Net23 M1.1 S1-wave Freq = 338kHz
VUG Net24 M1.1 S1-wave Freq = 338kHz
Velocity in x direction (m/s)
VUG Net19 M1.1 S2-wave Freq = 338kHz
VUG Net20 M1.1 S2-wave Freq = 338kHz
VUG Net21 M1.1 S2-wave Freq = 338kHz
VUG Net22 M1.1 S2-wave Freq = 338kHz
VUG Net23 M1.1 S2-wave Freq = 338kHz
VUG Net24 M1.1 S2-wave Freq = 338kHz
Velocity in z direction (m/s)
Simulation Time (sec)
x 10^-3
x 10^-5

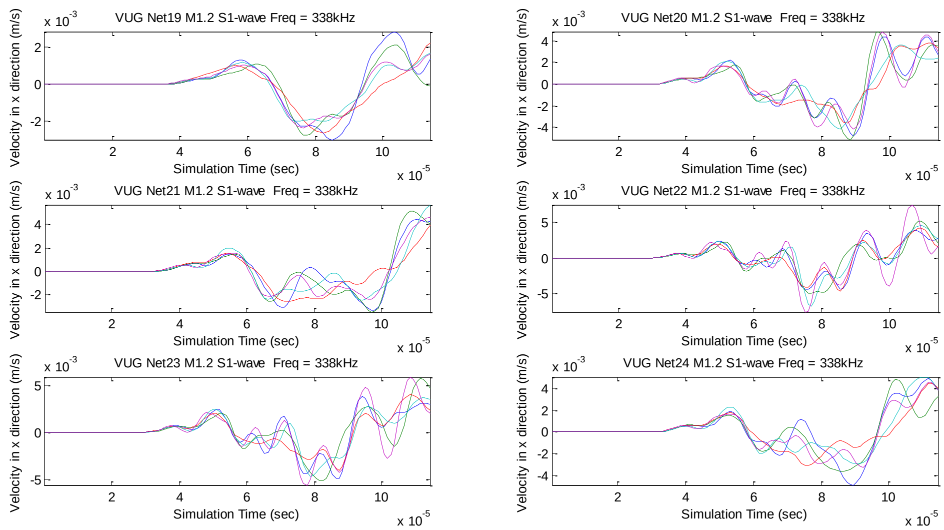

VUG Net55 M1.1 P-wave Freq = 338kHz
VUG Net56 M1.1 P-wave Freq = 338kHz
VUG Net57 M1.1 P-wave Freq = 338kHz
VUG Net58 M1.1 P-wave Freq = 338kHz
VUG Net59 M1.1 P-wave Freq = 338kHz
VUG Net60 M1.1 P-wave Freq = 338kHz
Velocity in y direction (m/s)
x 10-3
Simulation Time (sec)
x 10-5

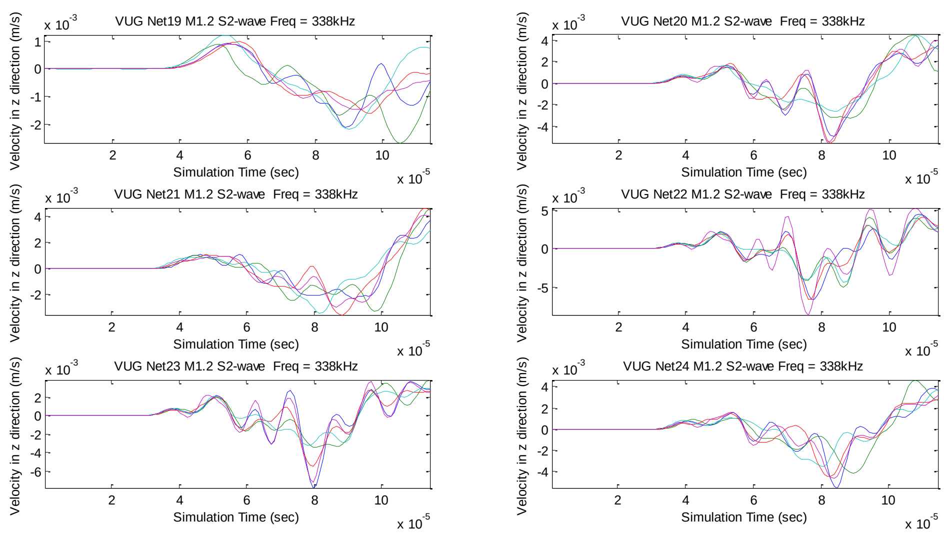

VUG Net55 M1.1 S1-wave Freq = 338kHz
VUG Net56 M1.1 S1-wave Freq = 338kHz
VUG Net57 M1.1 S1-wave Freq = 338kHz
VUG Net58 M1.1 S1-wave Freq = 338kHz
VUG Net59 M1.1 S1-wave Freq = 338kHz
VUG Net60 M1.1 S1-wave Freq = 338kHz
Velocity in x direction (m/s)
x 10-3
Simulation Time (sec)
x 10-5

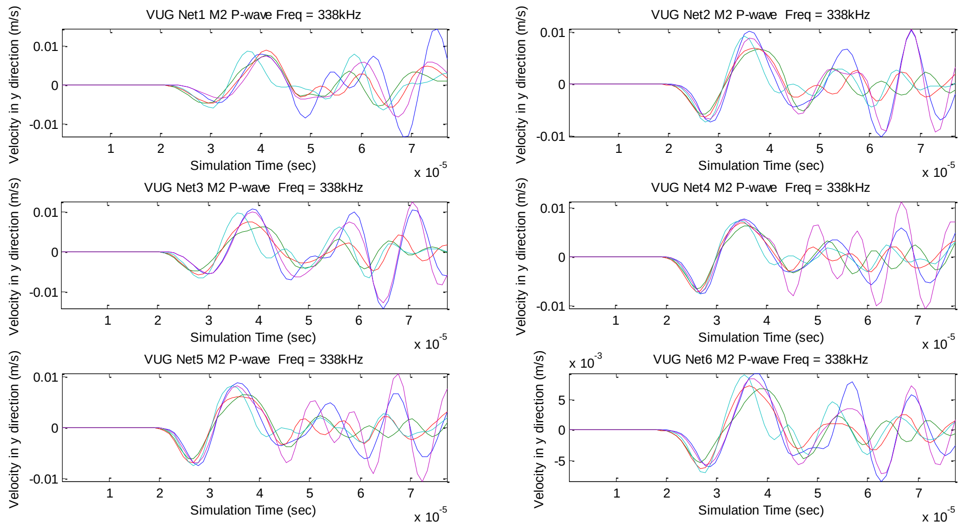

VUG Net55 M1.1 S2-wave Freq = 338kHz
VUG Net56 M1.1 S2-wave Freq = 338kHz
VUG Net57 M1.1 S2-wave Freq = 338kHz
VUG Net58 M1.1 S2-wave Freq = 338kHz
VUG Net59 M1.1 S2-wave Freq = 338kHz
VUG Net60 M1.1 S2-wave Freq = 338kHz
Velocity in z direction (m/s)
x 10-3
Simulation Time (sec)
x 10-5

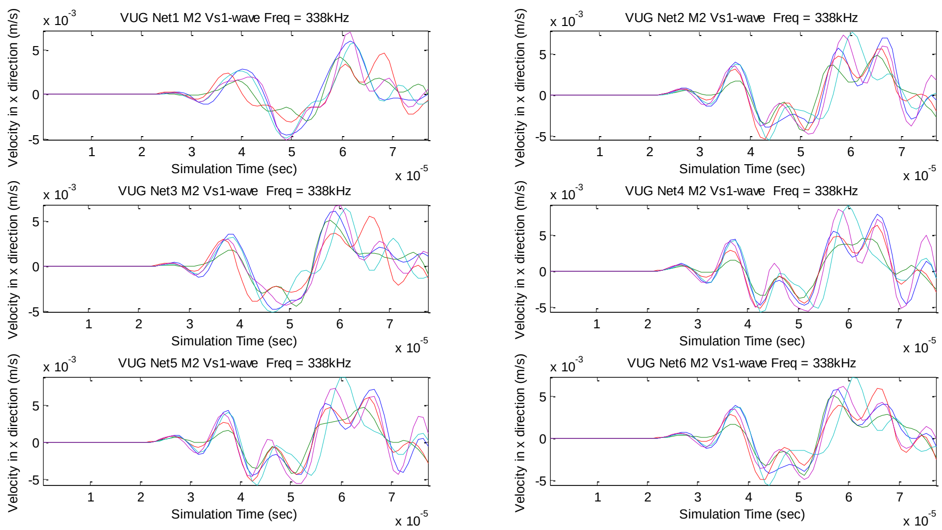

VUG Net1 M1.2 P-wave Freq = 338kHz
VUG Net2 M1.2 P-wave Freq = 338kHz
VUG Net3 M1.2 P-wave Freq = 338kHz
VUG Net4 M1.2 P-wave Freq = 338kHz
VUG Net5 M1.2 P-wave Freq = 338kHz
VUG Net6 M1.2 P-wave Freq = 338kHz
Velocity in y direction (m/s)
VUG Net1 M1.2 S1-wave Freq = 338kHz
VUG Net2 M1.2 S1-wave Freq = 338kHz
VUG Net3 M1.2 S1-wave Freq = 338kHz
VUG Net4 M1.2 S1-wave Freq = 338kHz
VUG Net5 M1.2 S1-wave Freq = 338kHz
VUG Net6 M1.2 S1-wave Freq = 338kHz
Velocity in x direction (m/s)
VUG Net1 M1.2 S2-wave Freq = 338kHz
VUG Net2 M1.2 S2-wave Freq = 338kHz
VUG Net3 M1.2 S2-wave Freq = 338kHz
VUG Net4 M1.2 S2-wave Freq = 338kHz
VUG Net5 M1.2 S2-wave Freq = 338kHz
VUG Net6 M1.2 S2-wave Freq = 338kHz
Velocity in z direction (m/s)
Simulation Time (sec)
x 10^-5
x 10^-3

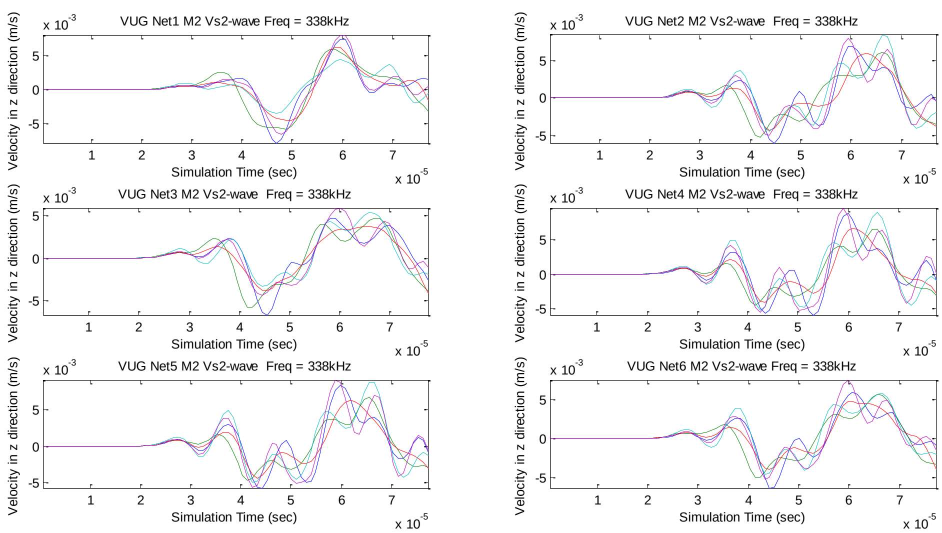

VUG Net19 M1.2 P-wave Freq = 338kHz
VUG Net20 M1.2 P-wave Freq = 338kHz
VUG Net21 M1.2 P-wave Freq = 338kHz
VUG Net22 M1.2 P-wave Freq = 338kHz
VUG Net23 M1.2 P-wave Freq = 338kHz
VUG Net24 M1.2 P-wave Freq = 338kHz
Velocity in y direction (m/s)
Simulation Time (sec)
x 10^-3
x 10^-5

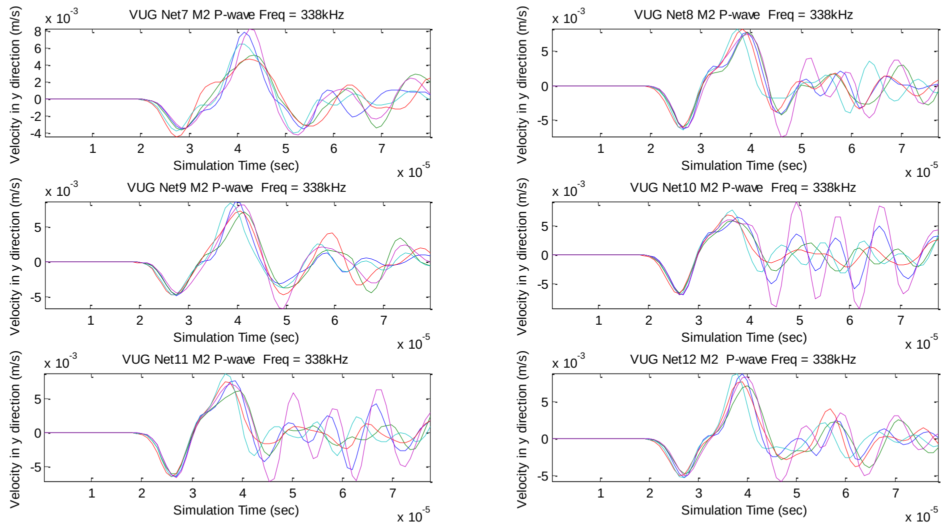

VUG Net19 M1.2 S1-wave Freq = 338kHz
VUG Net20 M1.2 S1-wave Freq = 338kHz
VUG Net21 M1.2 S1-wave Freq = 338kHz
VUG Net22 M1.2 S1-wave Freq = 338kHz
VUG Net23 M1.2 S1-wave Freq = 338kHz
VUG Net24 M1.2 S1-wave Freq = 338kHz
Velocity in x direction (m/s)
Simulation Time (sec)
x 10^-3
x 10^-5

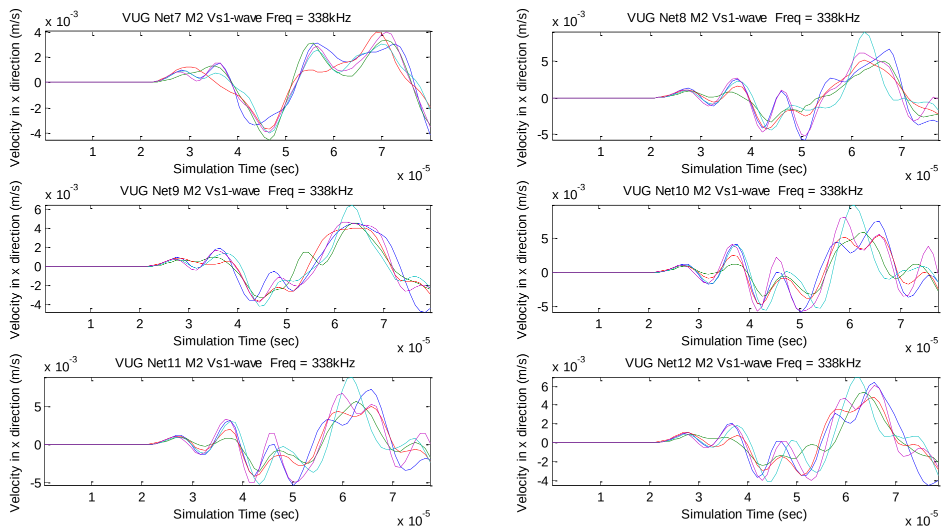

VUG Net19 M1.2 S2-wave Freq = 338kHz
VUG Net20 M1.2 S2-wave Freq = 338kHz
VUG Net21 M1.2 S2-wave Freq = 338kHz
VUG Net22 M1.2 S2-wave Freq = 338kHz
VUG Net23 M1.2 S2-wave Freq = 338kHz
VUG Net24 M1.2 S2-wave Freq = 338kHz
Velocity in z direction (m/s)
Simulation Time (sec)
x 10^-3
x 10^-5

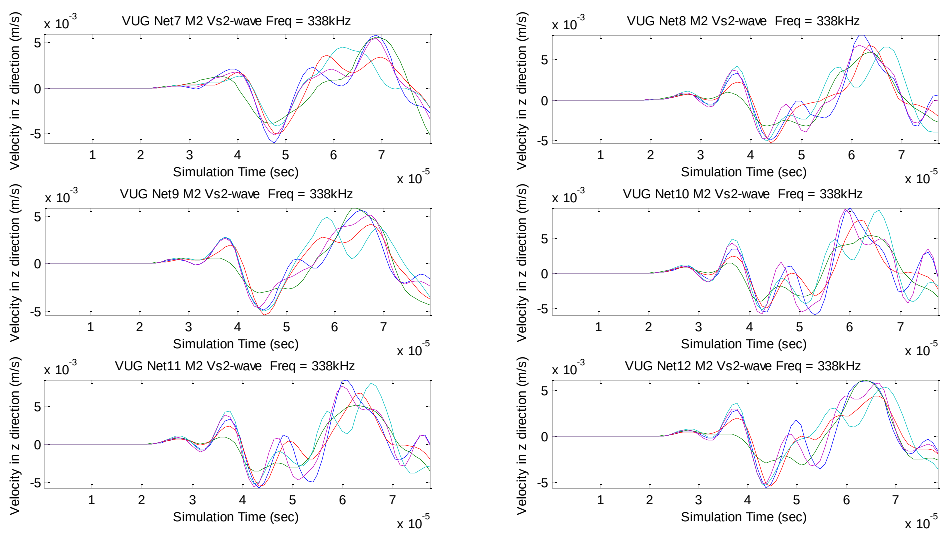

VUG Net1 M2 P-wave Freq = 338kHz
VUG Net2 M2 P-wave Freq = 338kHz
VUG Net3 M2 P-wave Freq = 338kHz
VUG Net4 M2 P-wave Freq = 338kHz
VUG Net5 M2 P-wave Freq = 338kHz
VUG Net6 M2 P-wave Freq = 338kHz
Velocity in y direction (m/s)
Simulation Time (sec)
x 10^-5
x 10^-3

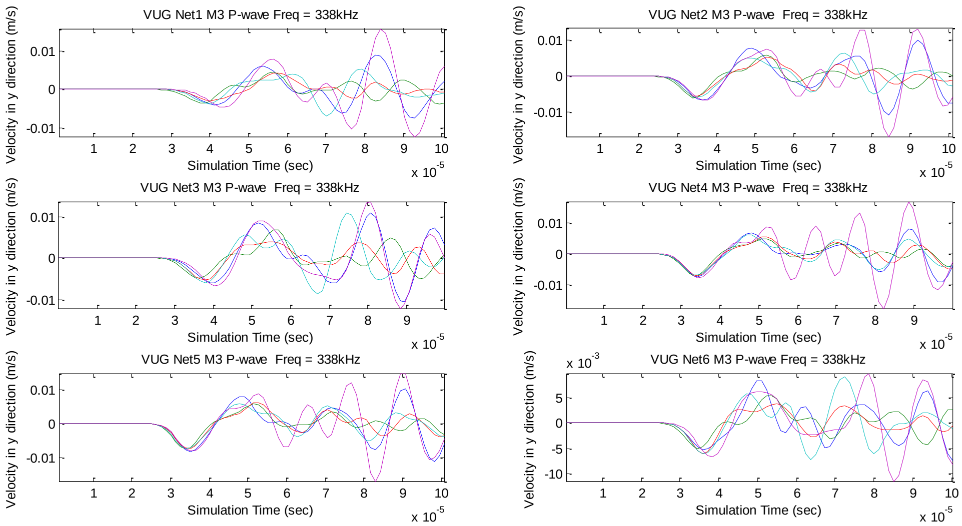

VUG Net1 M2 Vs1-wave Freq = 338kHz
VUG Net2 M2 Vs1-wave Freq = 338kHz
VUG Net3 M2 Vs1-wave Freq = 338kHz
VUG Net4 M2 Vs1-wave Freq = 338kHz
VUG Net5 M2 Vs1-wave Freq = 338kHz
VUG Net6 M2 Vs1-wave Freq = 338kHz
Velocity in x direction (m/s)
Simulation Time (sec)
x 10^-5
x 10^-3

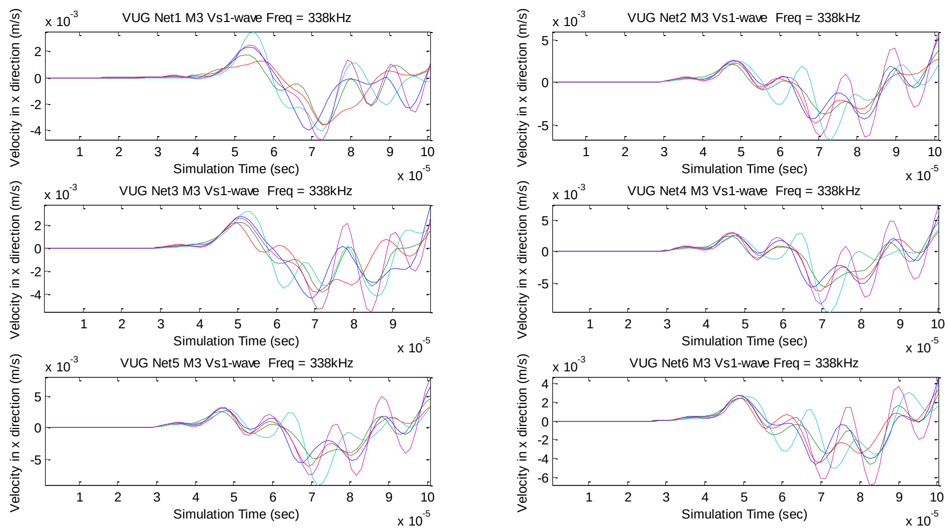

VUG Net1 M2 Vs2-wave Freq = 338kHz
VUG Net2 M2 Vs2-wave Freq = 338kHz
VUG Net3 M2 Vs2-wave Freq = 338kHz
VUG Net4 M2 Vs2-wave Freq = 338kHz
VUG Net5 M2 Vs2-wave Freq = 338kHz
VUG Net6 M2 Vs2-wave Freq = 338kHz
Velocity in z direction (m/s)
Simulation Time (sec)
x 10^-5
x 10^-3

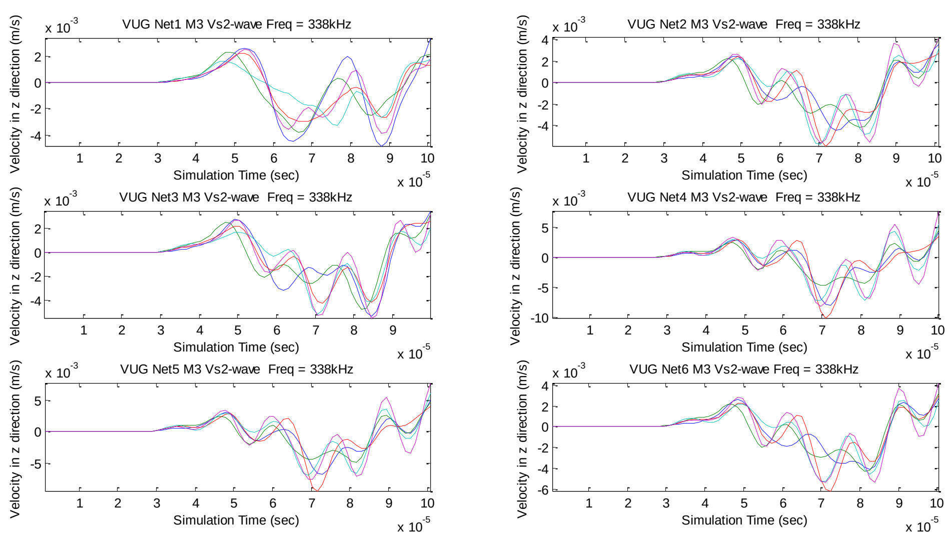

VUG Net7 M2 P-wave Freq = 338kHz
VUG Net8 M2 P-wave Freq = 338kHz
VUG Net9 M2 P-wave Freq = 338kHz
VUG Net10 M2 P-wave Freq = 338kHz
VUG Net11 M2 P-wave Freq = 338kHz
VUG Net12 M2 P-wave Freq = 338kHz
Velocity in y direction (m/s)
x 10-3
Simulation Time (sec)
x 10-5

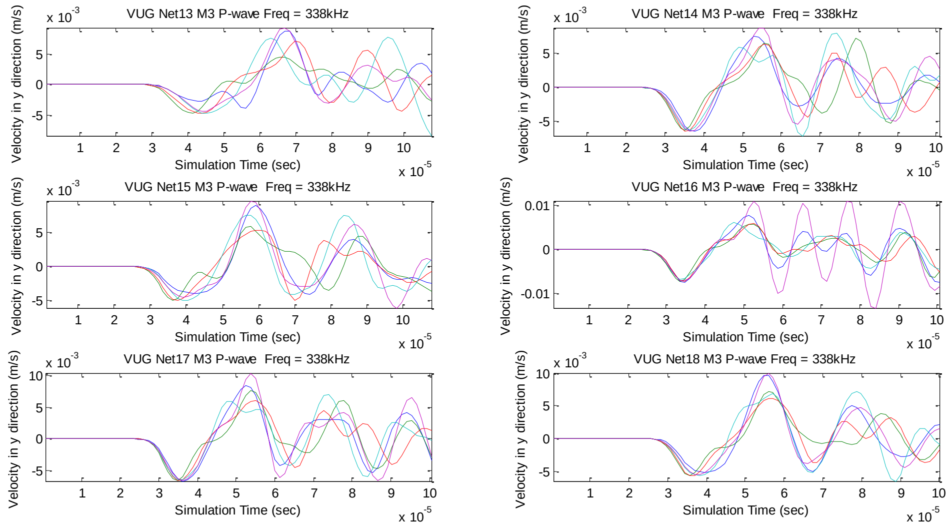

VUG Net7 M2 Vs1-wave Freq = 338kHz
VUG Net8 M2 Vs1-wave Freq = 338kHz
VUG Net9 M2 Vs1-wave Freq = 338kHz
VUG Net10 M2 Vs1-wave Freq = 338kHz
VUG Net11 M2 Vs1-wave Freq = 338kHz
VUG Net12 M2 Vs1-wave Freq = 338kHz
Velocity in x direction (m/s)
x 10-3
Simulation Time (sec)
x 10-5

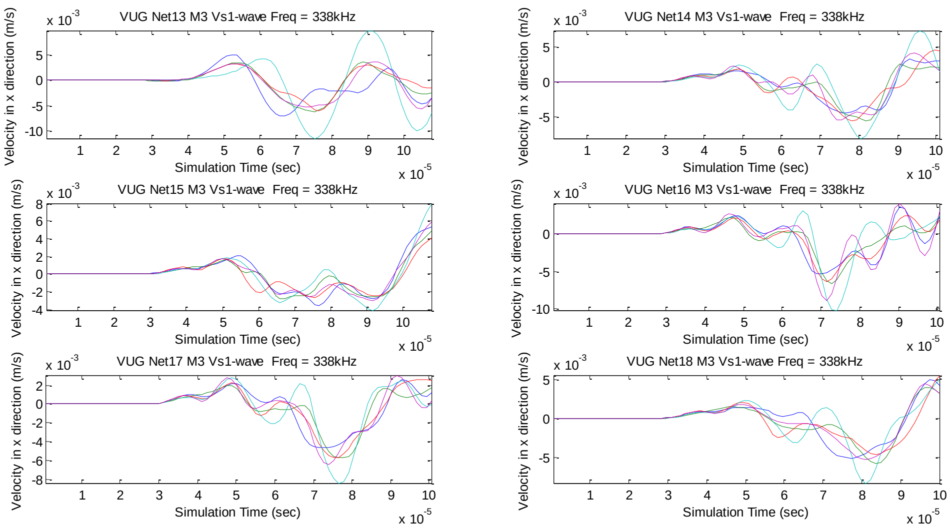

VUG Net7 M2 Vs2-wave Freq = 338kHz
VUG Net8 M2 Vs2-wave Freq = 338kHz
VUG Net9 M2 Vs2-wave Freq = 338kHz
VUG Net10 M2 Vs2-wave Freq = 338kHz
VUG Net11 M2 Vs2-wave Freq = 338kHz
VUG Net12 M2 Vs2-wave Freq = 338kHz
Velocity in z direction (m/s)
x 10-3
Simulation Time (sec)
x 10-5

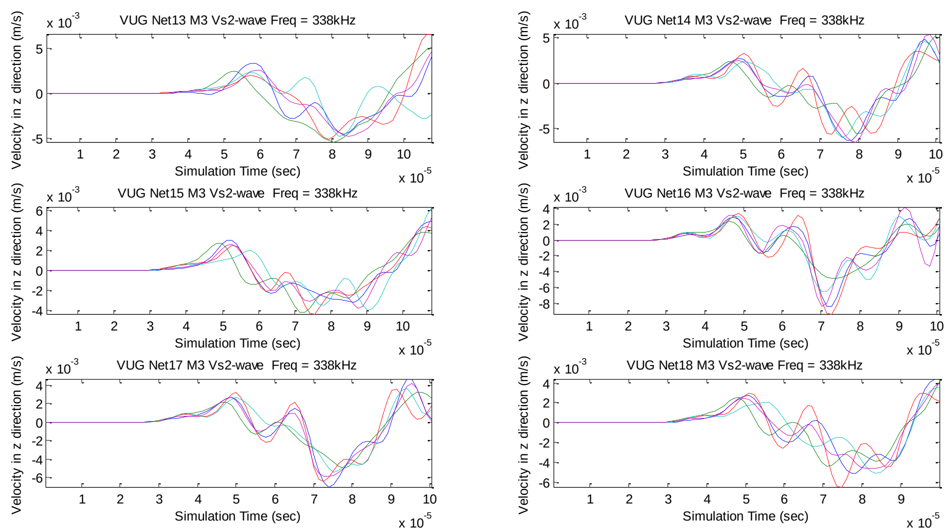

VUG Net1 M3 P-wave Freq = 338kHz
VUG Net2 M3 P-wave Freq = 338kHz
VUG Net3 M3 P-wave Freq = 338kHz
VUG Net4 M3 P-wave Freq = 338kHz
VUG Net5 M3 P-wave Freq = 338kHz
VUG Net6 M3 P-wave Freq = 338kHz
Velocity in y direction (m/s)
Simulation Time (sec)

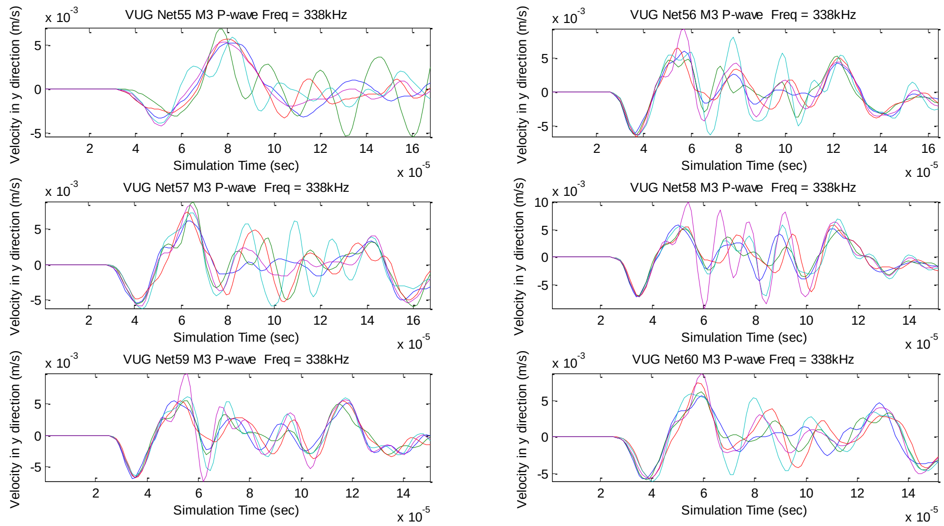

VUG Net1 M3 Vs1-wave Freq = 338kHz
VUG Net2 M3 Vs1-wave Freq = 338kHz
VUG Net3 M3 Vs1-wave Freq = 338kHz
VUG Net4 M3 Vs1-wave Freq = 338kHz
VUG Net5 M3 Vs1-wave Freq = 338kHz
VUG Net6 M3 Vs1-wave Freq = 338kHz
Velocity in x direction (m/s)
Simulation Time (sec)

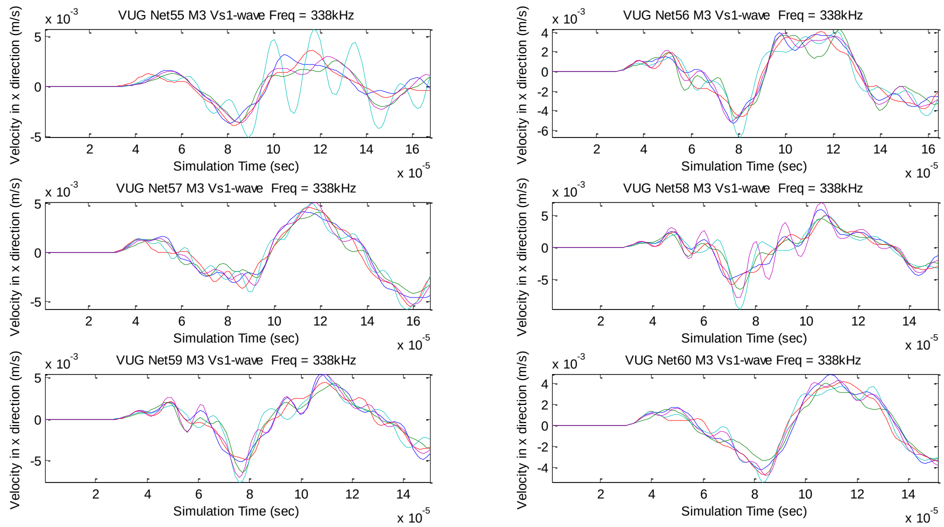

VUG Net1 M3 Vs2-wave Freq = 338kHz
VUG Net2 M3 Vs2-wave Freq = 338kHz
VUG Net3 M3 Vs2-wave Freq = 338kHz
VUG Net4 M3 Vs2-wave Freq = 338kHz
VUG Net5 M3 Vs2-wave Freq = 338kHz
VUG Net6 M3 Vs2-wave Freq = 338kHz
Velocity in z direction (m/s)
Simulation Time (sec)

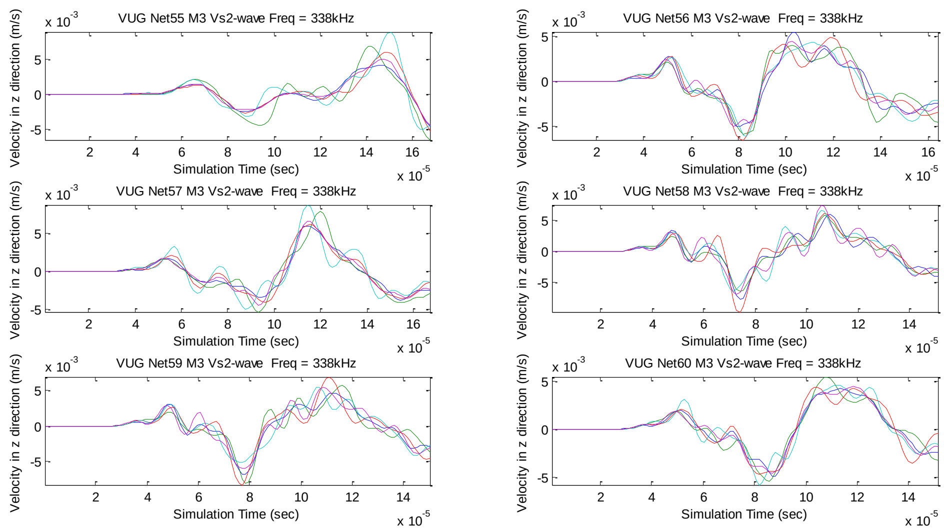

VUG Net13 M3 P-wave Freq = 338kHz
VUG Net14 M3 P-wave Freq = 338kHz
VUG Net15 M3 P-wave Freq = 338kHz
VUG Net16 M3 P-wave Freq = 338kHz
VUG Net17 M3 P-wave Freq = 338kHz
VUG Net18 M3 P-wave Freq = 338kHz
Velocity in y direction (m/s)
Simulation Time (sec)
x 10^-3
x 10^-5

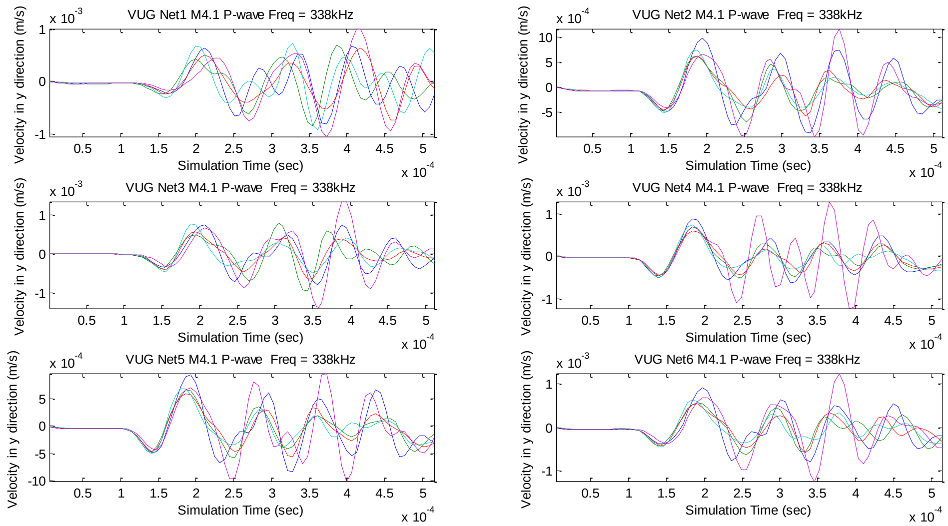

VUG Net13 M3 Vs1-wave Freq = 338kHz
VUG Net14 M3 Vs1-wave Freq = 338kHz
VUG Net15 M3 Vs1-wave Freq = 338kHz
VUG Net16 M3 Vs1-wave Freq = 338kHz
VUG Net17 M3 Vs1-wave Freq = 338kHz
VUG Net18 M3 Vs1-wave Freq = 338kHz
Velocity in x direction (m/s)
Simulation Time (sec)
x 10^-3
x 10^-5

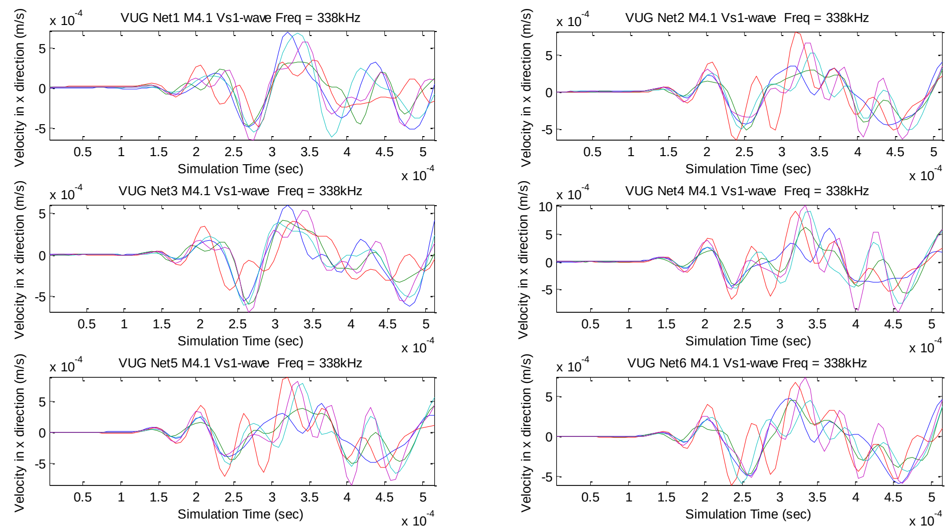

VUG Net13 M3 Vs2-wave Freq = 338kHz
VUG Net14 M3 Vs2-wave Freq = 338kHz
VUG Net15 M3 Vs2-wave Freq = 338kHz
VUG Net16 M3 Vs2-wave Freq = 338kHz
VUG Net17 M3 Vs2-wave Freq = 338kHz
VUG Net18 M3 Vs2-wave Freq = 338kHz
Velocity in z direction (m/s)
Simulation Time (sec)
x 10^-3
x 10^-5

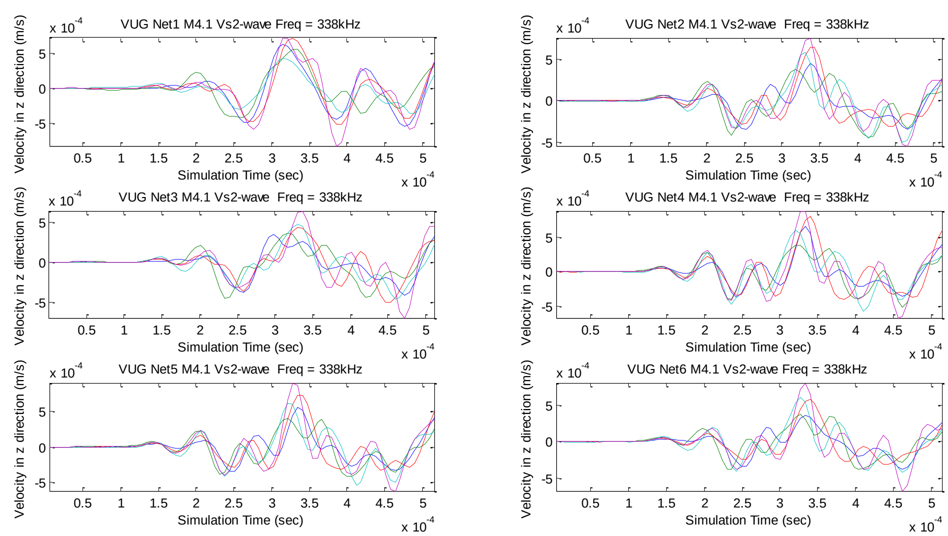

VUG Net55 M3 P-wave Freq = 338kHz
VUG Net56 M3 P-wave Freq = 338kHz
VUG Net57 M3 P-wave Freq = 338kHz
VUG Net58 M3 P-wave Freq = 338kHz
VUG Net59 M3 P-wave Freq = 338kHz
VUG Net60 M3 P-wave Freq = 338kHz
Velocity in y direction (m/s)
Simulation Time (sec)
x 10^-3
x 10^-5

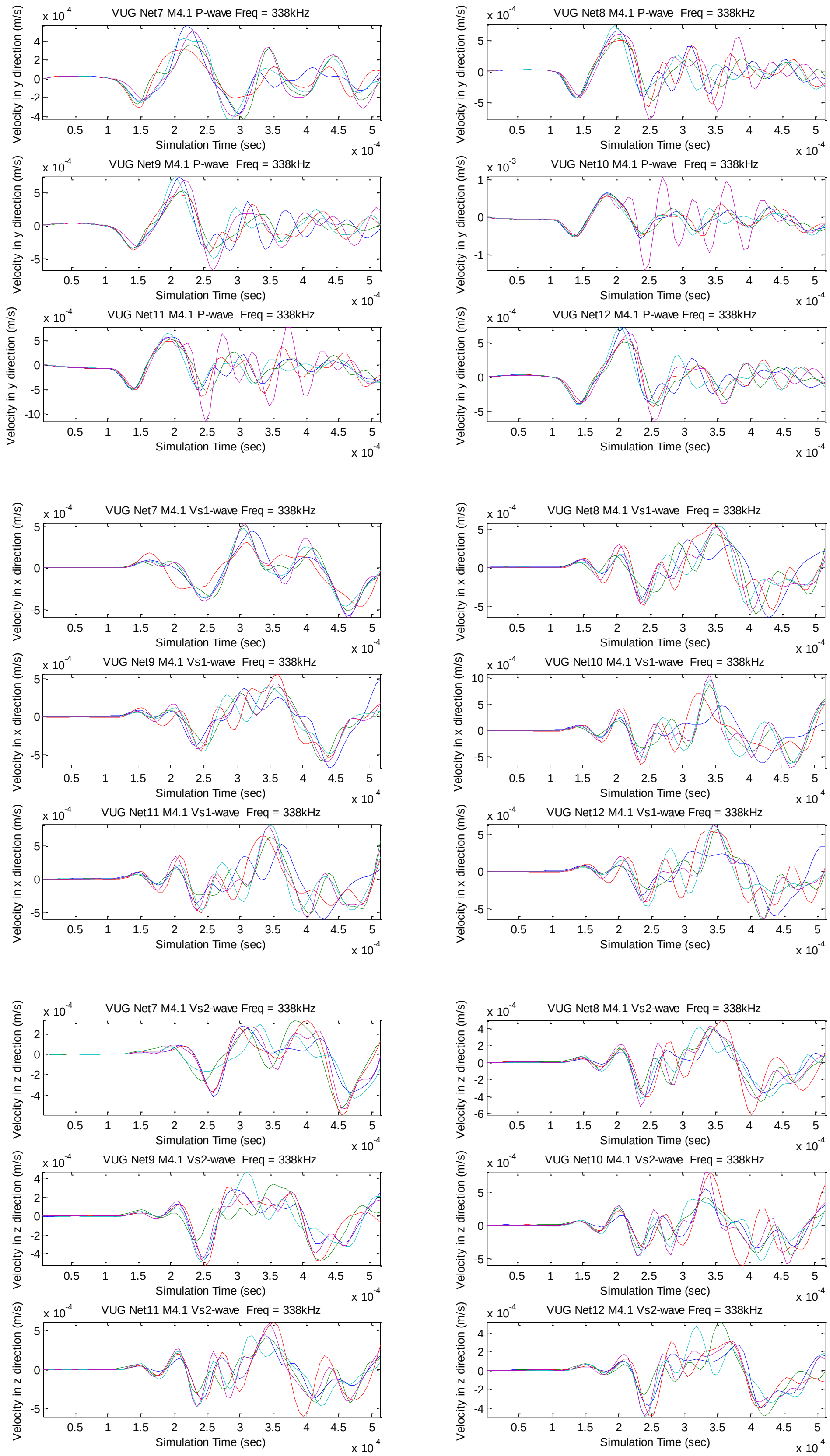

VUG Net55 M3 Vs1-wave Freq = 338kHz
VUG Net56 M3 Vs1-wave Freq = 338kHz
VUG Net57 M3 Vs1-wave Freq = 338kHz
VUG Net58 M3 Vs1-wave Freq = 338kHz
VUG Net59 M3 Vs1-wave Freq = 338kHz
VUG Net60 M3 Vs1-wave Freq = 338kHz
Velocity in x direction (m/s)
Simulation Time (sec)
x 10^-3
x 10^-5

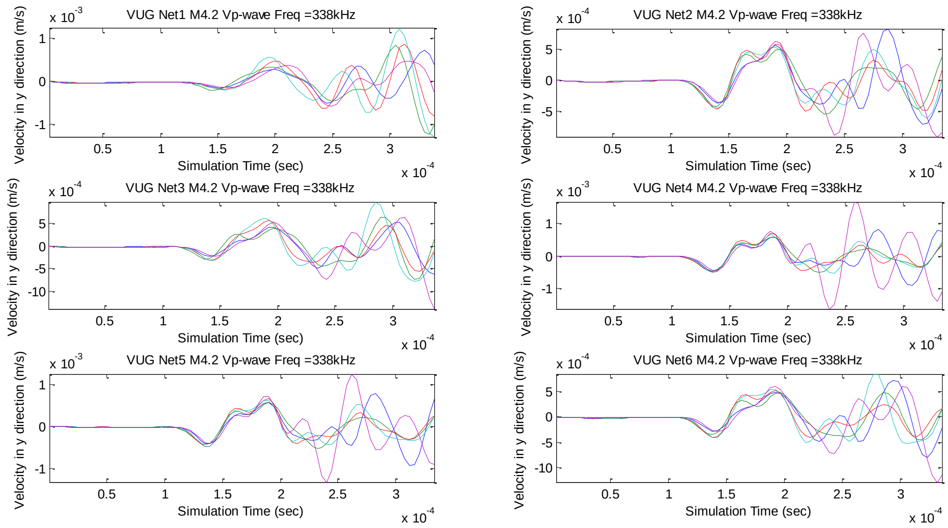

VUG Net55 M3 Vs2-wave Freq = 338kHz
VUG Net56 M3 Vs2-wave Freq = 338kHz
VUG Net57 M3 Vs2-wave Freq = 338kHz
VUG Net58 M3 Vs2-wave Freq = 338kHz
VUG Net59 M3 Vs2-wave Freq = 338kHz
VUG Net60 M3 Vs2-wave Freq = 338kHz
Velocity in z direction (m/s)
Simulation Time (sec)
x 10^-3
x 10^-5

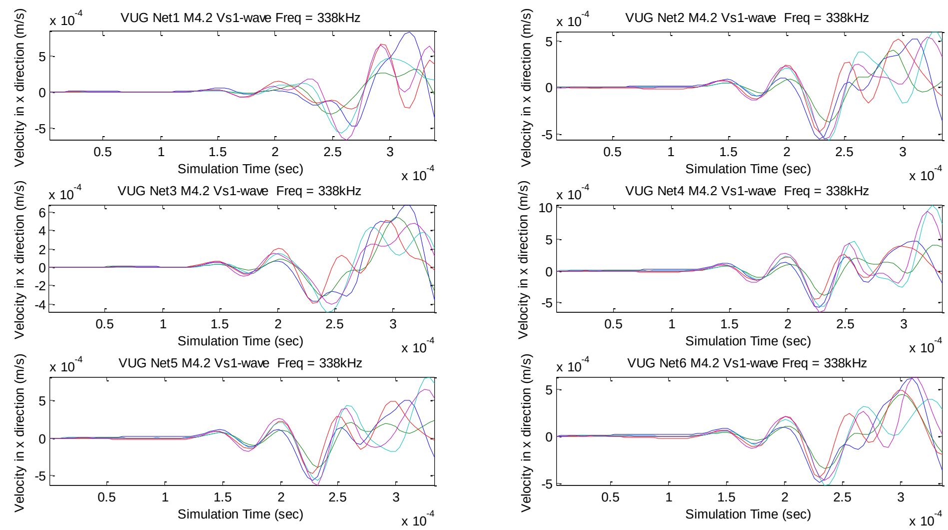

VUG Net1 M4.1 P-wave Freq = 338kHz
VUG Net2 M4.1 P-wave Freq = 338kHz
VUG Net3 M4.1 P-wave Freq = 338kHz
VUG Net4 M4.1 P-wave Freq = 338kHz
VUG Net5 M4.1 P-wave Freq = 338kHz
VUG Net6 M4.1 P-wave Freq = 338kHz
Velocity in y direction (m/s)
Simulation Time (sec)
x 10^-4

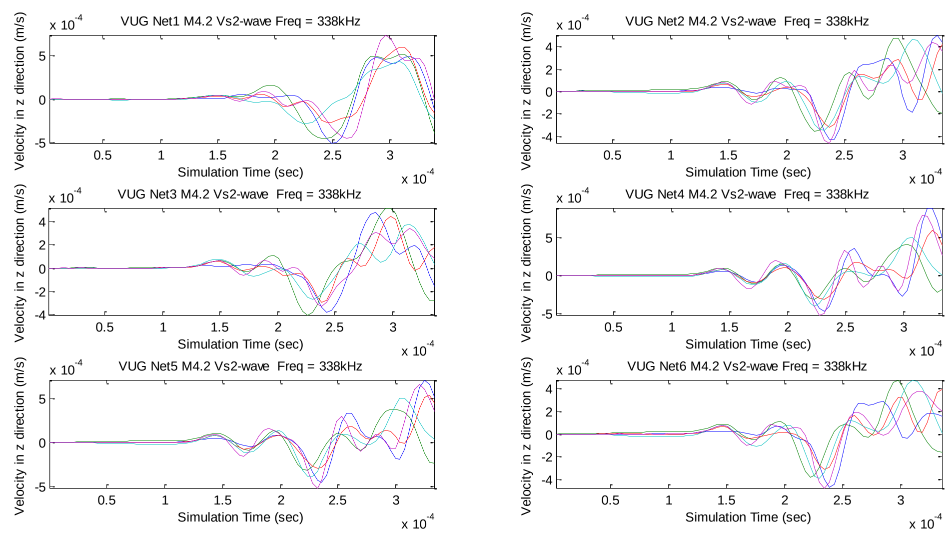

VUG Net1 M4.1 Vs1-wave Freq = 338kHz
VUG Net2 M4.1 Vs1-wave Freq = 338kHz
VUG Net3 M4.1 Vs1-wave Freq = 338kHz
VUG Net4 M4.1 Vs1-wave Freq = 338kHz
VUG Net5 M4.1 Vs1-wave Freq = 338kHz
VUG Net6 M4.1 Vs1-wave Freq = 338kHz
Velocity in x direction (m/s)
Simulation Time (sec)
x 10^-4

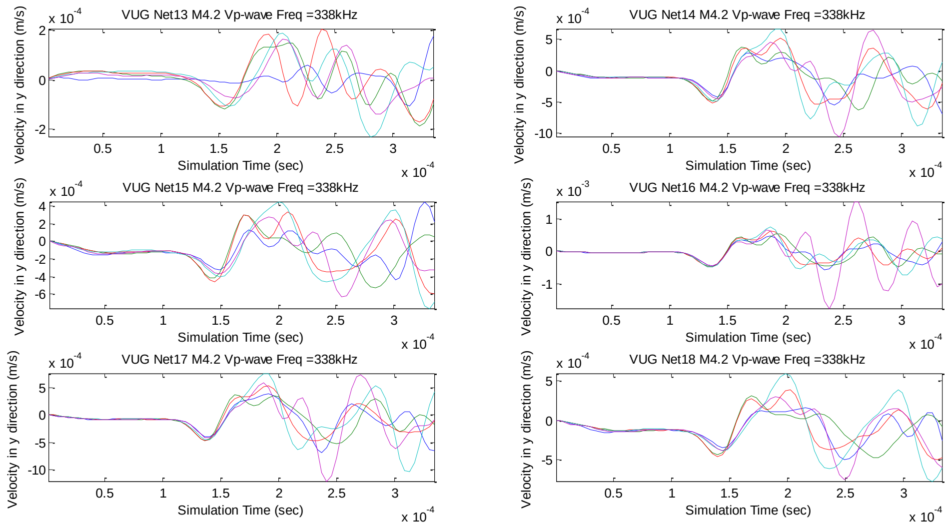

VUG Net1 M4.1 Vs2-wave Freq = 338kHz
VUG Net2 M4.1 Vs2-wave Freq = 338kHz
VUG Net3 M4.1 Vs2-wave Freq = 338kHz
VUG Net4 M4.1 Vs2-wave Freq = 338kHz
VUG Net5 M4.1 Vs2-wave Freq = 338kHz
VUG Net6 M4.1 Vs2-wave Freq = 338kHz
Velocity in z direction (m/s)
Simulation Time (sec)
x 10^-4

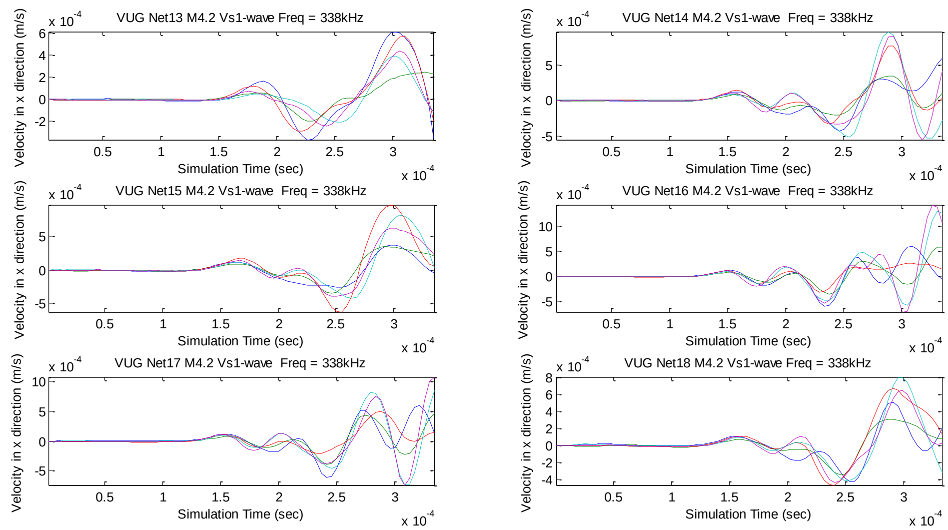

VUG Net7 M4.1 P-wave Freq = 338kHz
VUG Net8 M4.1 P-wave Freq = 338kHz
VUG Net9 M4.1 P-wave Freq = 338kHz
VUG Net10 M4.1 P-wave Freq = 338kHz
VUG Net11 M4.1 P-wave Freq = 338kHz
VUG Net12 M4.1 P-wave Freq = 338kHz
Velocity in y direction (m/s)
VUG Net7 M4.1 Vs1-wave Freq = 338kHz
VUG Net8 M4.1 Vs1-wave Freq = 338kHz
VUG Net9 M4.1 Vs1-wave Freq = 338kHz
VUG Net10 M4.1 Vs1-wave Freq = 338kHz
VUG Net11 M4.1 Vs1-wave Freq = 338kHz
VUG Net12 M4.1 Vs1-wave Freq = 338kHz
Velocity in x direction (m/s)
VUG Net7 M4.1 Vs2-wave Freq = 338kHz
VUG Net8 M4.1 Vs2-wave Freq = 338kHz
VUG Net9 M4.1 Vs2-wave Freq = 338kHz
VUG Net10 M4.1 Vs2-wave Freq = 338kHz
VUG Net11 M4.1 Vs2-wave Freq = 338kHz
VUG Net12 M4.1 Vs2-wave Freq = 338kHz
Velocity in z direction (m/s)
Simulation Time (sec)
x 10^-4

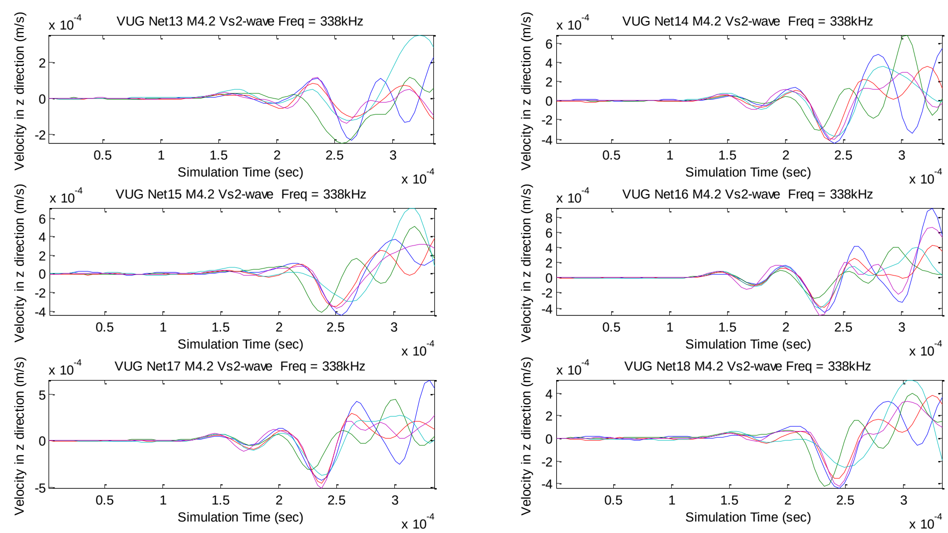

VUG Net1 M4.2 Vp-wave Freq =338kHz
VUG Net2 M4.2 Vp-wave Freq =338kHz
VUG Net3 M4.2 Vp-wave Freq =338kHz
VUG Net4 M4.2 Vp-wave Freq =338kHz
VUG Net5 M4.2 Vp-wave Freq =338kHz
VUG Net6 M4.2 Vp-wave Freq =338kHz
Velocity in y direction (m/s)
Simulation Time (sec)

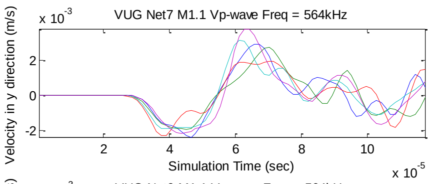

VUG Net1 M4.2 Vs1-wave Freq = 338kHz
VUG Net2 M4.2 Vs1-wave Freq = 338kHz
VUG Net3 M4.2 Vs1-wave Freq = 338kHz
VUG Net4 M4.2 Vs1-wave Freq = 338kHz
VUG Net5 M4.2 Vs1-wave Freq = 338kHz
VUG Net6 M4.2 Vs1-wave Freq = 338kHz
Velocity in x direction (m/s)
Simulation Time (sec)

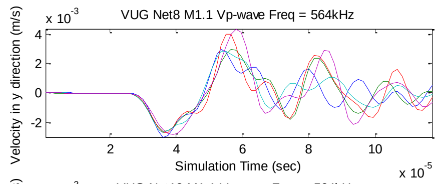

VUG Net1 M4.2 Vs2-wave Freq = 338kHz
VUG Net2 M4.2 Vs2-wave Freq = 338kHz
VUG Net3 M4.2 Vs2-wave Freq = 338kHz
VUG Net4 M4.2 Vs2-wave Freq = 338kHz
VUG Net5 M4.2 Vs2-wave Freq = 338kHz
VUG Net6 M4.2 Vs2-wave Freq = 338kHz
Velocity in z direction (m/s)
Simulation Time (sec)

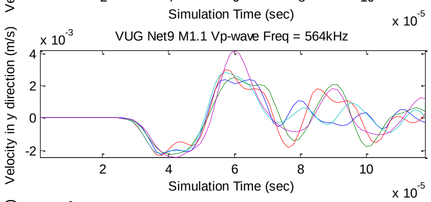

VUG Net13 M4.2 Vp-wave Freq =338kHz
VUG Net14 M4.2 Vp-wave Freq =338kHz
VUG Net15 M4.2 Vp-wave Freq =338kHz
VUG Net16 M4.2 Vp-wave Freq =338kHz
VUG Net17 M4.2 Vp-wave Freq =338kHz
VUG Net18 M4.2 Vp-wave Freq =338kHz
Velocity in y direction (m/s)
Simulation Time (sec)

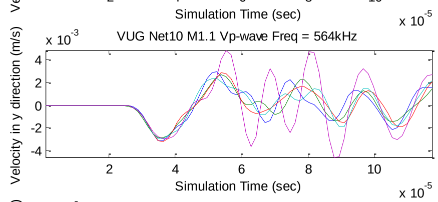

VUG Net13 M4.2 Vs1-wave Freq = 338kHz
VUG Net14 M4.2 Vs1-wave Freq = 338kHz
VUG Net15 M4.2 Vs1-wave Freq = 338kHz
VUG Net16 M4.2 Vs1-wave Freq = 338kHz
VUG Net17 M4.2 Vs1-wave Freq = 338kHz
VUG Net18 M4.2 Vs1-wave Freq = 338kHz
Velocity in x direction (m/s)
Simulation Time (sec)

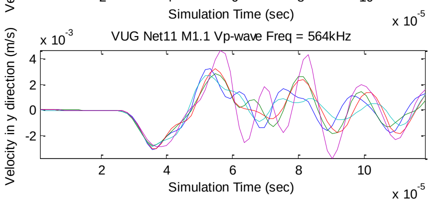

VUG Net13 M4.2 Vs2-wave Freq = 338kHz
VUG Net14 M4.2 Vs2-wave Freq = 338kHz
VUG Net15 M4.2 Vs2-wave Freq = 338kHz
VUG Net16 M4.2 Vs2-wave Freq = 338kHz
VUG Net17 M4.2 Vs2-wave Freq = 338kHz
VUG Net18 M4.2 Vs2-wave Freq = 338kHz
Velocity in z direction (m/s)
Simulation Time (sec)

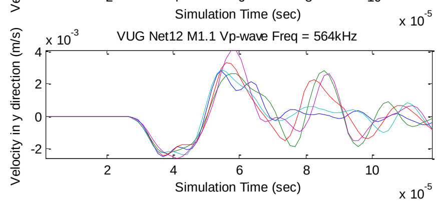

x 10^-3
VUG Net7 M1.1 Vp-wave Freq = 564kHz
Velocity in y direction (m/s)
Simulation Time (sec)
x 10^-5

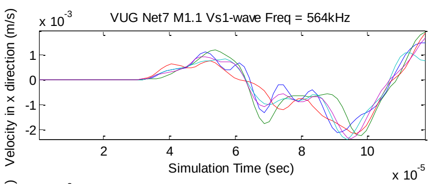

x 10^-3
VUG Net8 M1.1 Vp-wave Freq = 564kHz
Velocity in y direction (m/s)
Simulation Time (sec)
x 10^-5

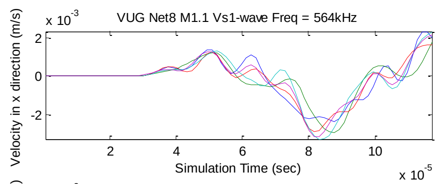

x 10^-3
VUG Net9 M1.1 Vp-wave Freq = 564kHz
Velocity in y direction (m/s)
Simulation Time (sec)
x 10^-5

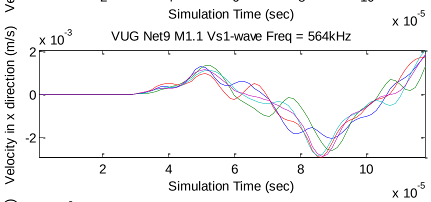

x 10^-3
VUG Net10 M1.1 Vp-wave Freq = 564kHz
Velocity in y direction (m/s)
Simulation Time (sec)
x 10^-5

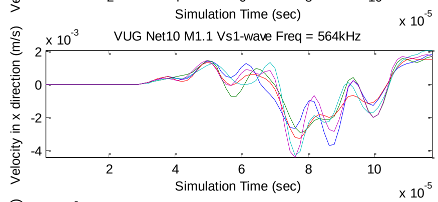

x 10^-3
VUG Net11 M1.1 Vp-wave Freq = 564kHz
Velocity in y direction (m/s)
Simulation Time (sec)
x 10^-5

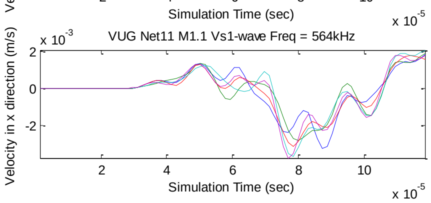

x 10^-3
VUG Net12 M1.1 Vp-wave Freq = 564kHz
Velocity in y direction (m/s)
Simulation Time (sec)
x 10^-5

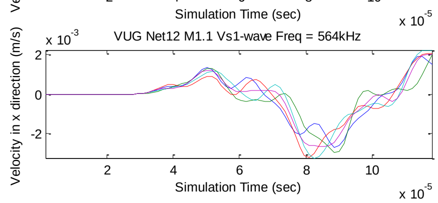

x 10^-3
VUG Net7 M1.1 Vs1-wave Freq = 564kHz
Velocity in x direction (m/s)
Simulation Time (sec)
x 10^-5

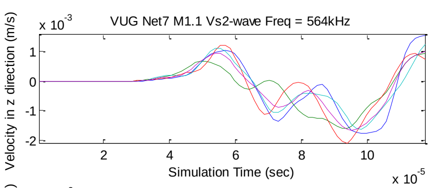

x 10^-3
VUG Net8 M1.1 Vs1-wave Freq = 564kHz
Velocity in x direction (m/s)
Simulation Time (sec)
x 10^-5

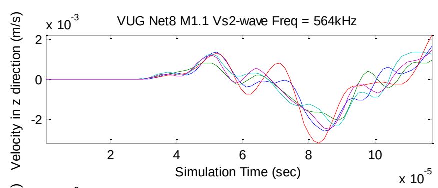

x 10^-3
VUG Net9 M1.1 Vs1-wave Freq = 564kHz
Velocity in x direction (m/s)
Simulation Time (sec)
x 10^-5

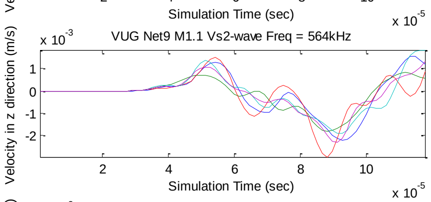

x 10^-3
VUG Net10 M1.1 Vs1-wave Freq = 564kHz
Velocity in x direction (m/s)
Simulation Time (sec)
x 10^-5

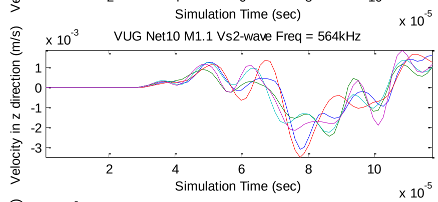

x 10^-3
VUG Net11 M1.1 Vs1-wave Freq = 564kHz
Velocity in x direction (m/s)
Simulation Time (sec)
x 10^-5

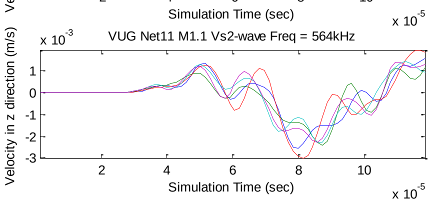

x 10^-3
VUG Net12 M1.1 Vs1-wave Freq = 564kHz
Velocity in x direction (m/s)
Simulation Time (sec)
x 10^-5

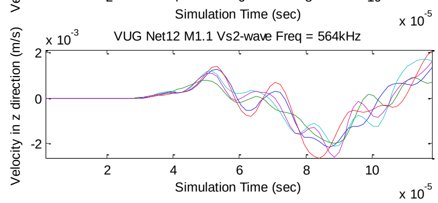

x 10^-3
VUG Net7 M1.1 Vs2-wave Freq = 564kHz
Velocity in z direction (m/s)
Simulation Time (sec)
x 10^-5

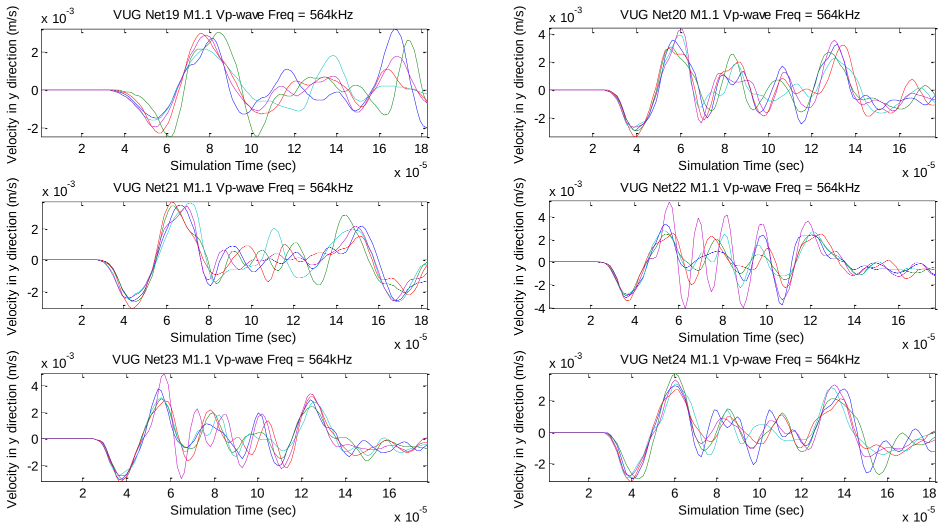

x 10^-3
VUG Net8 M1.1 Vs2-wave Freq = 564kHz
Velocity in z direction (m/s)
Simulation Time (sec)
x 10^-5

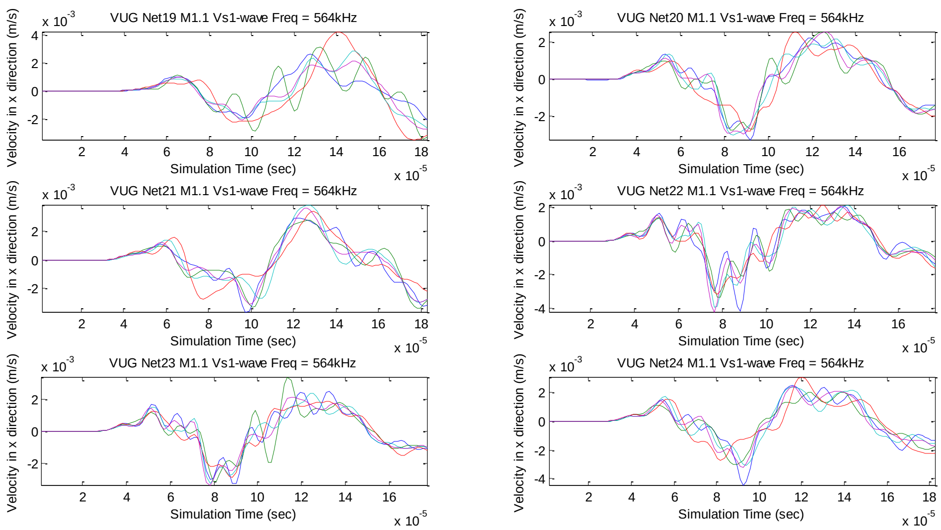

x 10^-3
VUG Net9 M1.1 Vs2-wave Freq = 564kHz
Velocity in z direction (m/s)
Simulation Time (sec)
x 10^-5

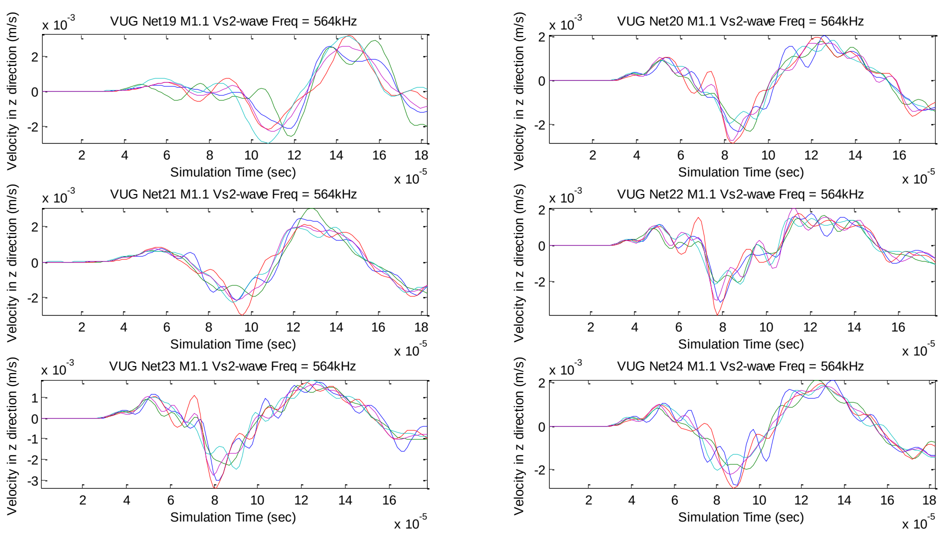

x 10^-3
VUG Net10 M1.1 Vs2-wave Freq = 564kHz
Velocity in z direction (m/s)
Simulation Time (sec)
x 10^-5

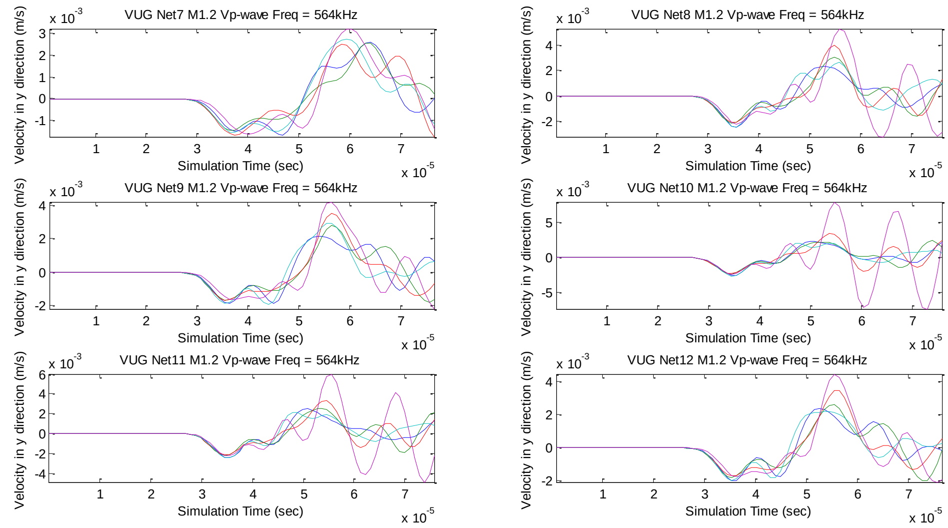

x 10^-3
VUG Net11 M1.1 Vs2-wave Freq = 564kHz
Velocity in z direction (m/s)
Simulation Time (sec)
x 10^-5

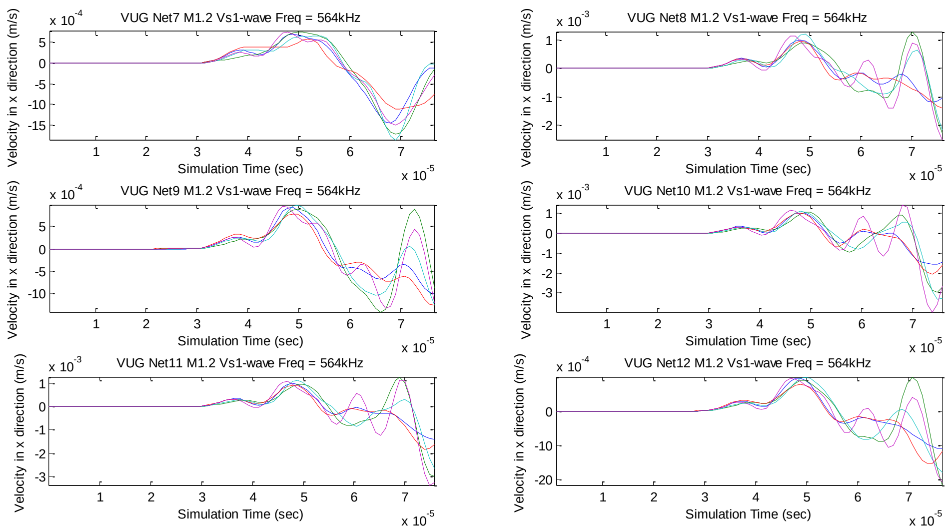

x 10^-3
VUG Net12 M1.1 Vs2-wave Freq = 564kHz
Velocity in z direction (m/s)
Simulation Time (sec)
x 10^-5

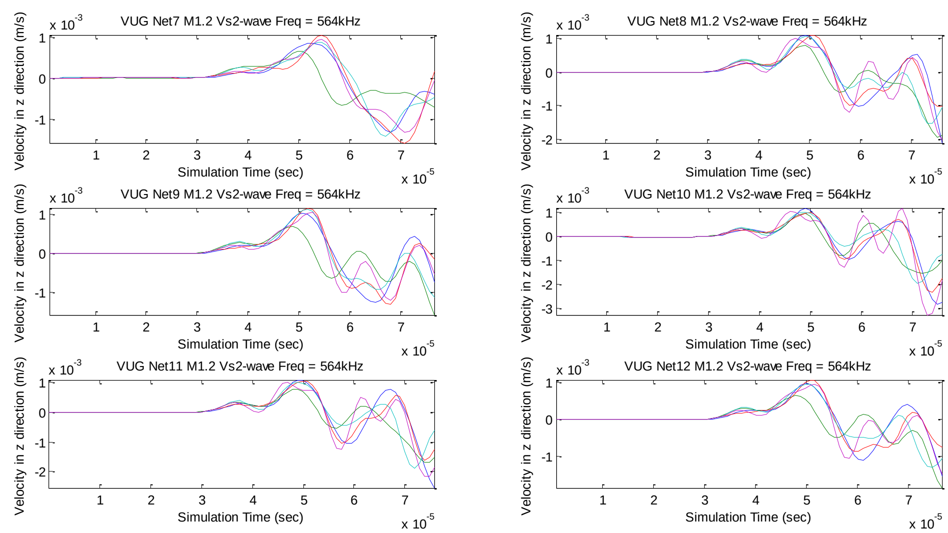

VUG Net19 M1.1 Vp-wave Freq = 564kHz
VUG Net20 M1.1 Vp-wave Freq = 564kHz
VUG Net21 M1.1 Vp-wave Freq = 564kHz
VUG Net22 M1.1 Vp-wave Freq = 564kHz
VUG Net23 M1.1 Vp-wave Freq = 564kHz
VUG Net24 M1.1 Vp-wave Freq = 564kHz
Velocity in y direction (m/s)
Simulation Time (sec)
x 10-3
x 10-5

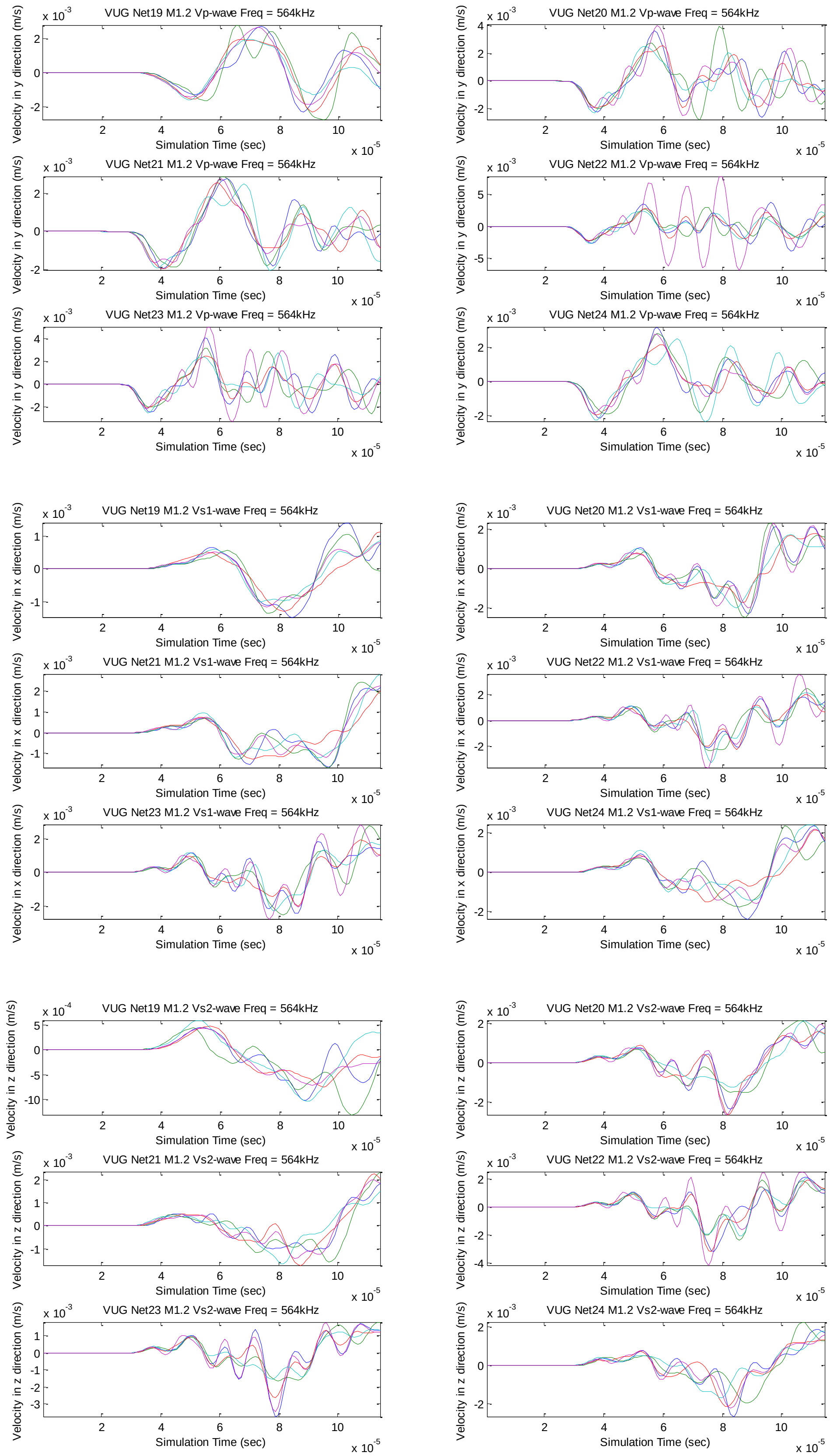

VUG Net19 M1.1 Vs1-wave Freq = 564kHz
VUG Net20 M1.1 Vs1-wave Freq = 564kHz
VUG Net21 M1.1 Vs1-wave Freq = 564kHz
VUG Net22 M1.1 Vs1-wave Freq = 564kHz
VUG Net23 M1.1 Vs1-wave Freq = 564kHz
VUG Net24 M1.1 Vs1-wave Freq = 564kHz
Velocity in x direction (m/s)
Simulation Time (sec)
x 10-3
x 10-5

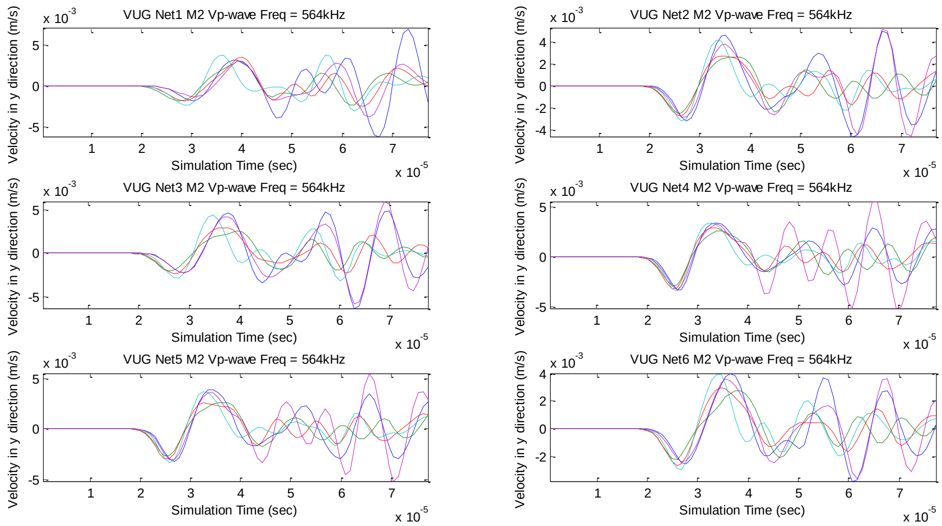

VUG Net19 M1.1 Vs2-wave Freq = 564kHz
VUG Net20 M1.1 Vs2-wave Freq = 564kHz
VUG Net21 M1.1 Vs2-wave Freq = 564kHz
VUG Net22 M1.1 Vs2-wave Freq = 564kHz
VUG Net23 M1.1 Vs2-wave Freq = 564kHz
VUG Net24 M1.1 Vs2-wave Freq = 564kHz
Velocity in z direction (m/s)
Simulation Time (sec)
x 10-3
x 10-5

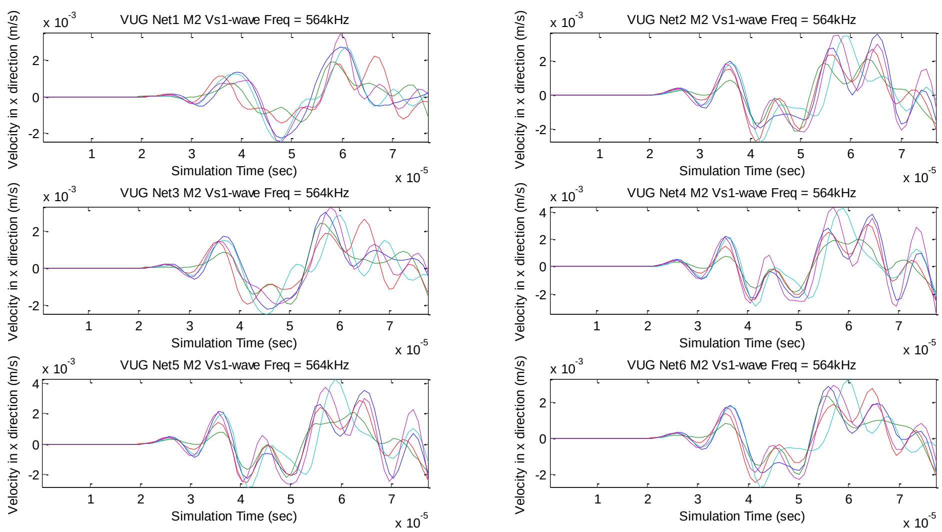

VUG Net7 M1.2 Vp-wave Freq = 564kHz
VUG Net8 M1.2 Vp-wave Freq = 564kHz
VUG Net9 M1.2 Vp-wave Freq = 564kHz
VUG Net10 M1.2 Vp-wave Freq = 564kHz
VUG Net11 M1.2 Vp-wave Freq = 564kHz
VUG Net12 M1.2 Vp-wave Freq = 564kHz
Velocity in y direction (m/s)
Simulation Time (sec)
x 10^-3
x 10^-5

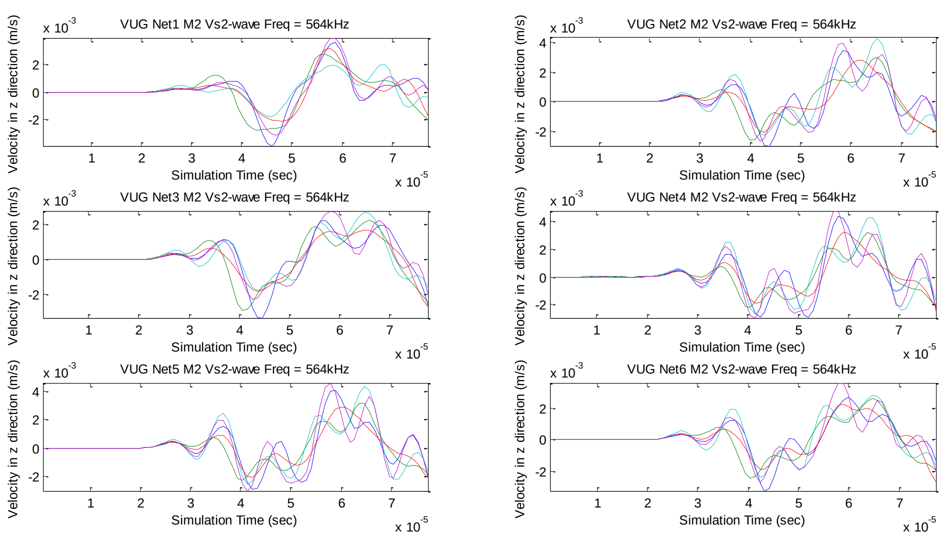

VUG Net7 M1.2 Vs1-wave Freq = 564kHz
VUG Net8 M1.2 Vs1-wave Freq = 564kHz
VUG Net9 M1.2 Vs1-wave Freq = 564kHz
VUG Net10 M1.2 Vs1-wave Freq = 564kHz
VUG Net11 M1.2 Vs1-wave Freq = 564kHz
VUG Net12 M1.2 Vs1-wave Freq = 564kHz
Velocity in x direction (m/s)
Simulation Time (sec)
x 10^-4
x 10^-3
x 10^-5

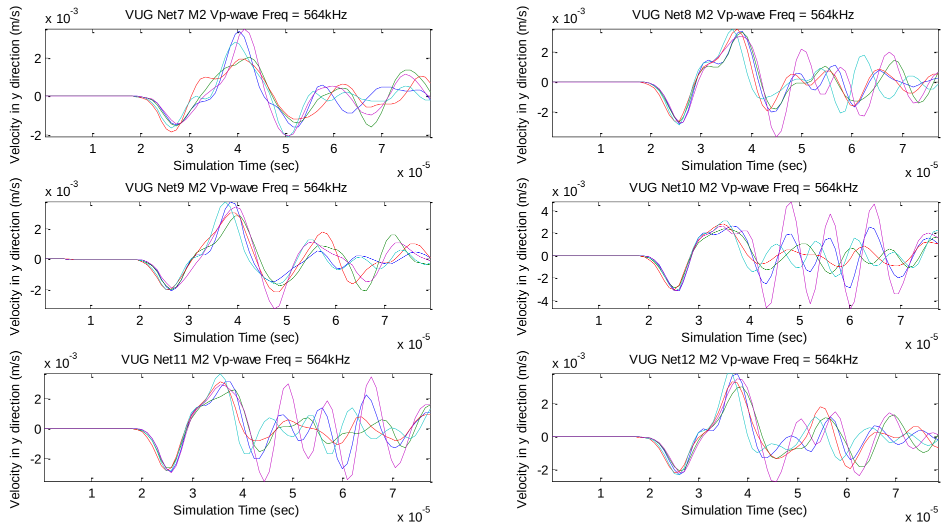

VUG Net7 M1.2 Vs2-wave Freq = 564kHz
VUG Net8 M1.2 Vs2-wave Freq = 564kHz
VUG Net9 M1.2 Vs2-wave Freq = 564kHz
VUG Net10 M1.2 Vs2-wave Freq = 564kHz
VUG Net11 M1.2 Vs2-wave Freq = 564kHz
VUG Net12 M1.2 Vs2-wave Freq = 564kHz
Velocity in z direction (m/s)
Simulation Time (sec)
x 10^-3
x 10^-5

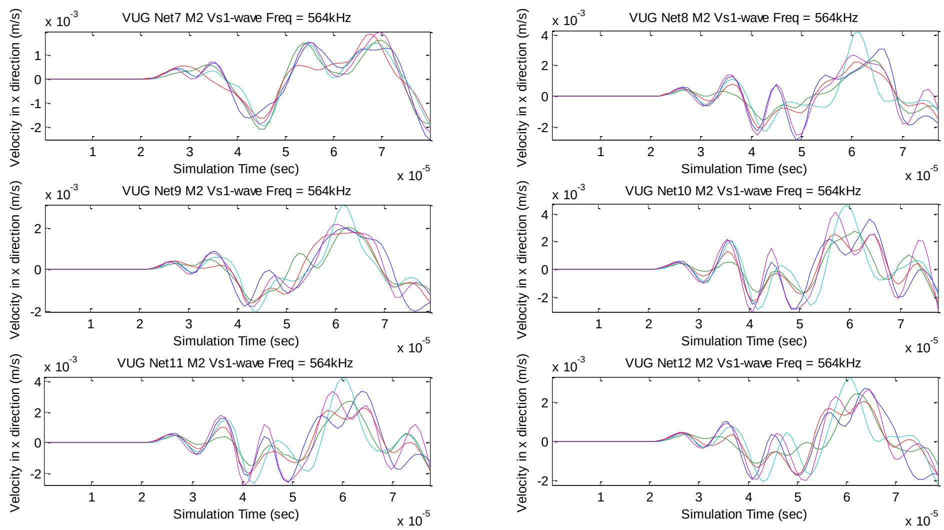

VUG Net19 M1.2 Vp-wave Freq = 564kHz
VUG Net20 M1.2 Vp-wave Freq = 564kHz
VUG Net21 M1.2 Vp-wave Freq = 564kHz
VUG Net22 M1.2 Vp-wave Freq = 564kHz
VUG Net23 M1.2 Vp-wave Freq = 564kHz
VUG Net24 M1.2 Vp-wave Freq = 564kHz
Velocity in y direction (m/s)
VUG Net19 M1.2 Vs1-wave Freq = 564kHz
VUG Net20 M1.2 Vs1-wave Freq = 564kHz
VUG Net21 M1.2 Vs1-wave Freq = 564kHz
VUG Net22 M1.2 Vs1-wave Freq = 564kHz
VUG Net23 M1.2 Vs1-wave Freq = 564kHz
VUG Net24 M1.2 Vs1-wave Freq = 564kHz
Velocity in x direction (m/s)
VUG Net19 M1.2 Vs2-wave Freq = 564kHz
VUG Net20 M1.2 Vs2-wave Freq = 564kHz
VUG Net21 M1.2 Vs2-wave Freq = 564kHz
VUG Net22 M1.2 Vs2-wave Freq = 564kHz
VUG Net23 M1.2 Vs2-wave Freq = 564kHz
VUG Net24 M1.2 Vs2-wave Freq = 564kHz
Velocity in z direction (m/s)
Simulation Time (sec)
x 10-5
x 10-3
x 10-4

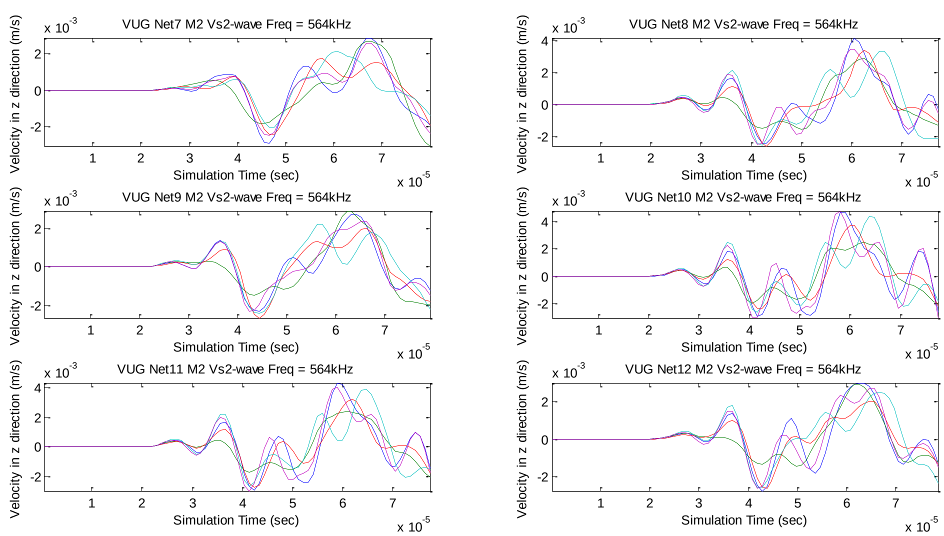

VUG Net1 M2 Vp-wave Freq = 564kHz
VUG Net2 M2 Vp-wave Freq = 564kHz
VUG Net3 M2 Vp-wave Freq = 564kHz
VUG Net4 M2 Vp-wave Freq = 564kHz
VUG Net5 M2 Vp-wave Freq = 564kHz
VUG Net6 M2 Vp-wave Freq = 564kHz
Velocity in y direction (m/s)
Simulation Time (sec)
x 10^-3
x 10^-5

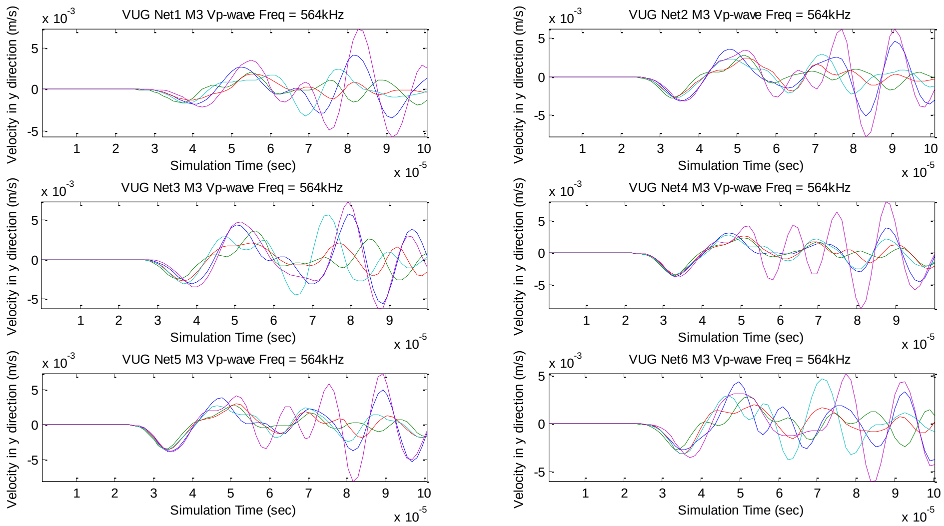

VUG Net1 M2 Vs1-wave Freq = 564kHz
VUG Net2 M2 Vs1-wave Freq = 564kHz
VUG Net3 M2 Vs1-wave Freq = 564kHz
VUG Net4 M2 Vs1-wave Freq = 564kHz
VUG Net5 M2 Vs1-wave Freq = 564kHz
VUG Net6 M2 Vs1-wave Freq = 564kHz
Velocity in x direction (m/s)
Simulation Time (sec)
x 10^-3
x 10^-5

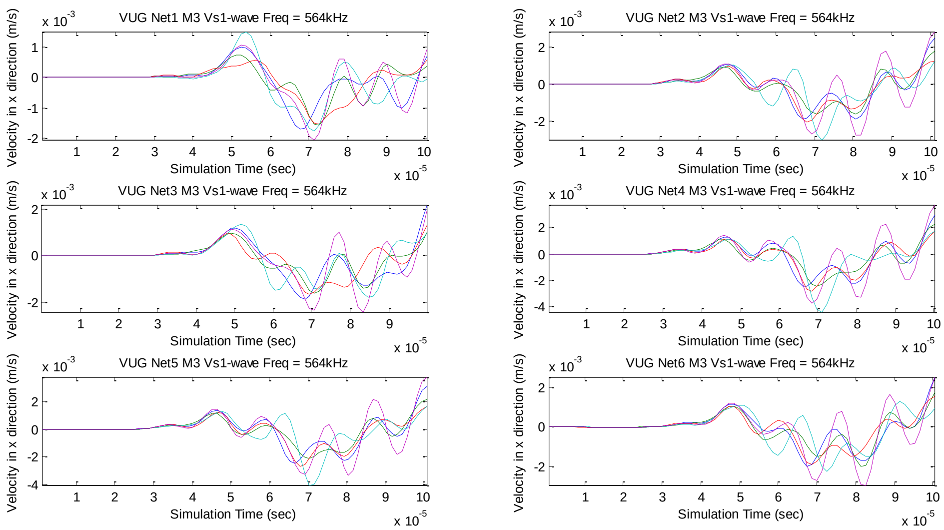

VUG Net1 M2 Vs2-wave Freq = 564kHz
VUG Net2 M2 Vs2-wave Freq = 564kHz
VUG Net3 M2 Vs2-wave Freq = 564kHz
VUG Net4 M2 Vs2-wave Freq = 564kHz
VUG Net5 M2 Vs2-wave Freq = 564kHz
VUG Net6 M2 Vs2-wave Freq = 564kHz
Velocity in z direction (m/s)
Simulation Time (sec)
x 10^-3
x 10^-5

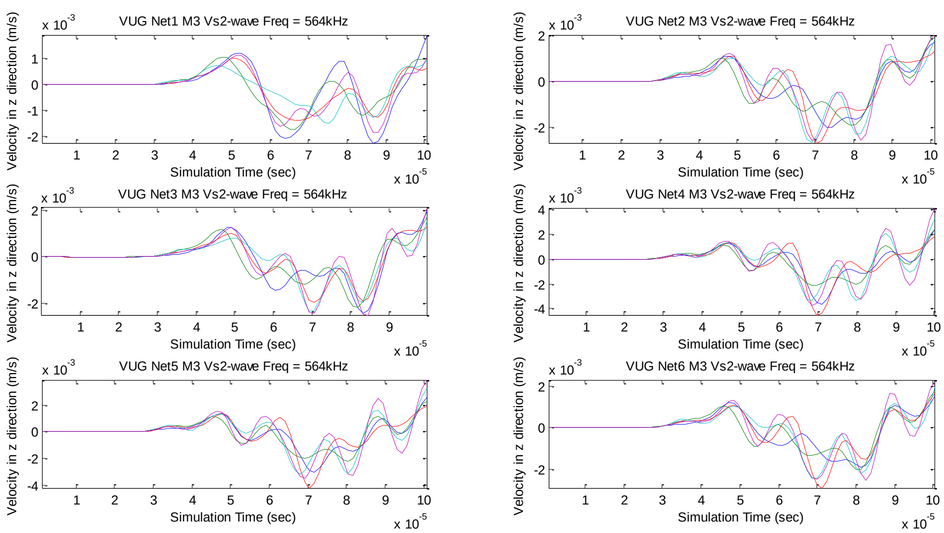

VUG Net7 M2 Vp-wave Freq = 564kHz
VUG Net8 M2 Vp-wave Freq = 564kHz
VUG Net9 M2 Vp-wave Freq = 564kHz
VUG Net10 M2 Vp-wave Freq = 564kHz
VUG Net11 M2 Vp-wave Freq = 564kHz
VUG Net12 M2 Vp-wave Freq = 564kHz
Velocity in y direction (m/s)
Simulation Time (sec)
x 10-3
x 10-5

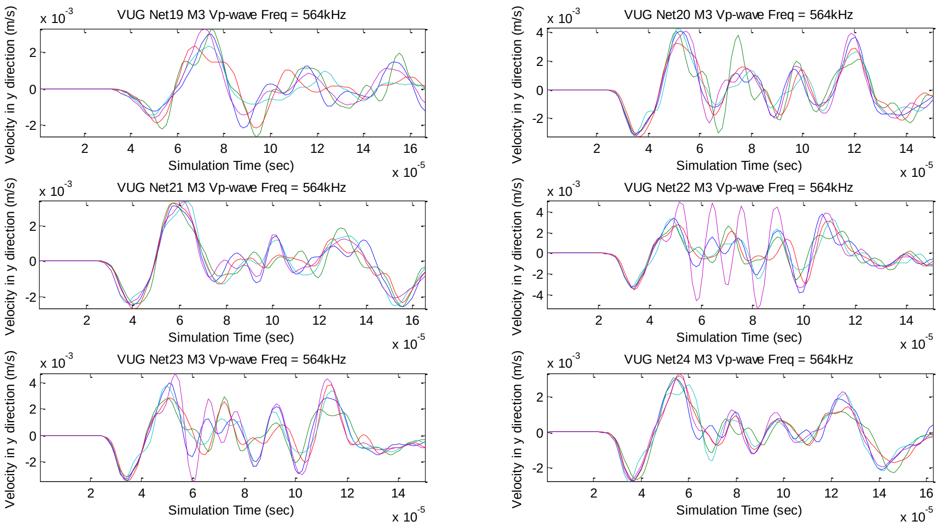

VUG Net7 M2 Vs1-wave Freq = 564kHz
VUG Net8 M2 Vs1-wave Freq = 564kHz
VUG Net9 M2 Vs1-wave Freq = 564kHz
VUG Net10 M2 Vs1-wave Freq = 564kHz
VUG Net11 M2 Vs1-wave Freq = 564kHz
VUG Net12 M2 Vs1-wave Freq = 564kHz
Velocity in x direction (m/s)
Simulation Time (sec)
x 10-3
x 10-5

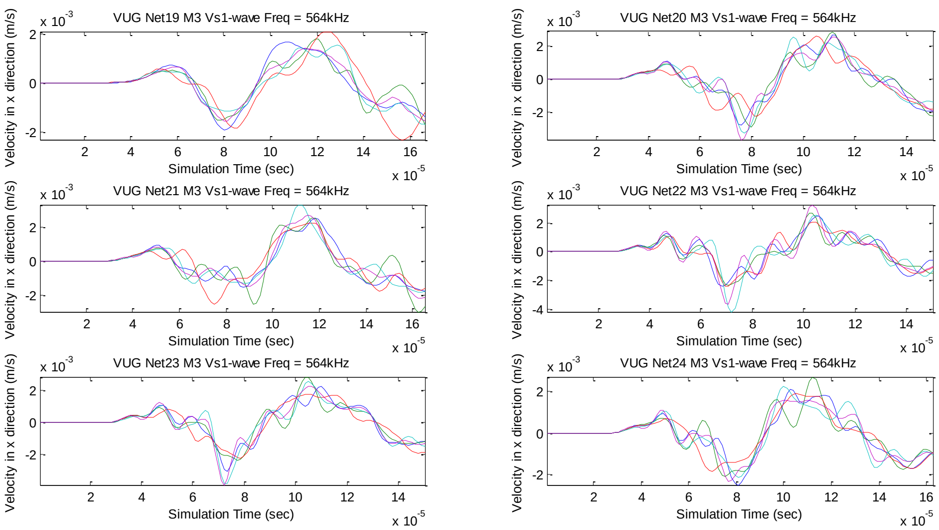

VUG Net7 M2 Vs2-wave Freq = 564kHz
VUG Net8 M2 Vs2-wave Freq = 564kHz
VUG Net9 M2 Vs2-wave Freq = 564kHz
VUG Net10 M2 Vs2-wave Freq = 564kHz
VUG Net11 M2 Vs2-wave Freq = 564kHz
VUG Net12 M2 Vs2-wave Freq = 564kHz
Velocity in z direction (m/s)
Simulation Time (sec)
x 10-3
x 10-5

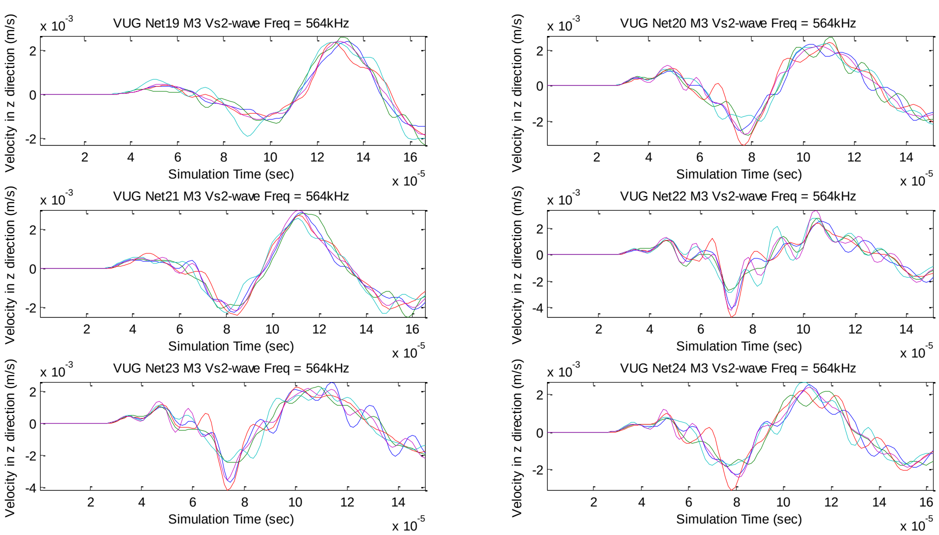

VUG Net1 M3 Vp-wave Freq = 564kHz
VUG Net2 M3 Vp-wave Freq = 564kHz
VUG Net3 M3 Vp-wave Freq = 564kHz
VUG Net4 M3 Vp-wave Freq = 564kHz
VUG Net5 M3 Vp-wave Freq = 564kHz
VUG Net6 M3 Vp-wave Freq = 564kHz
Velocity in y direction (m/s)
x 10^-3
Simulation Time (sec)
x 10^-5

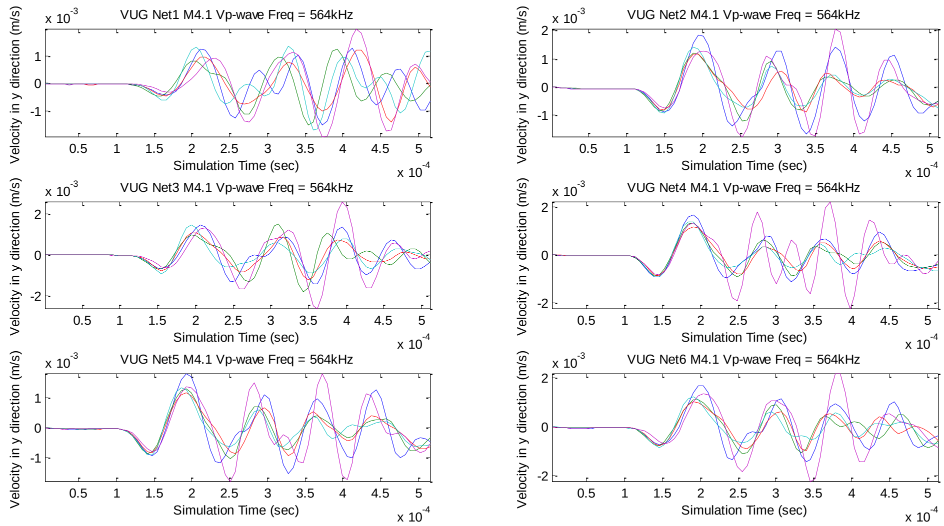

VUG Net1 M3 Vs1-wave Freq = 564kHz
VUG Net2 M3 Vs1-wave Freq = 564kHz
VUG Net3 M3 Vs1-wave Freq = 564kHz
VUG Net4 M3 Vs1-wave Freq = 564kHz
VUG Net5 M3 Vs1-wave Freq = 564kHz
VUG Net6 M3 Vs1-wave Freq = 564kHz
Velocity in x direction (m/s)
x 10^-3
Simulation Time (sec)
x 10^-5

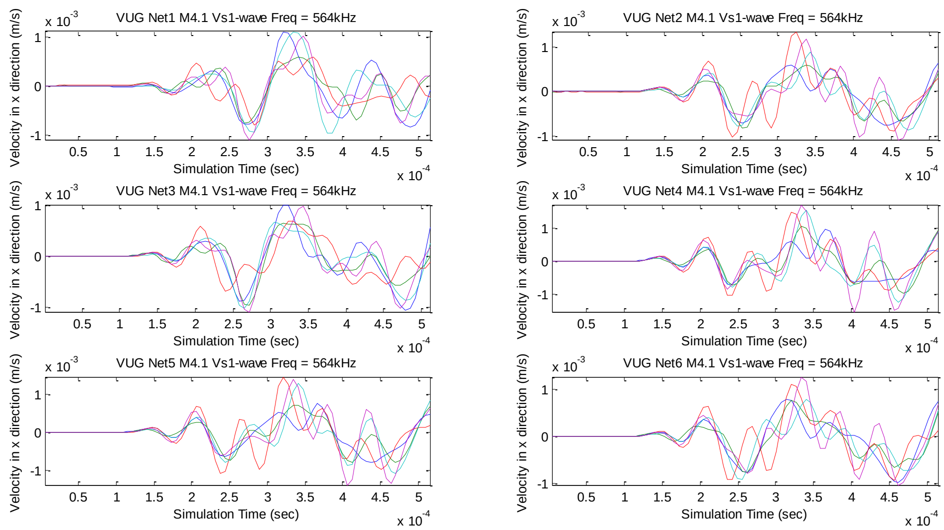

VUG Net1 M3 Vs2-wave Freq = 564kHz
VUG Net2 M3 Vs2-wave Freq = 564kHz
VUG Net3 M3 Vs2-wave Freq = 564kHz
VUG Net4 M3 Vs2-wave Freq = 564kHz
VUG Net5 M3 Vs2-wave Freq = 564kHz
VUG Net6 M3 Vs2-wave Freq = 564kHz
Velocity in z direction (m/s)
x 10^-3
Simulation Time (sec)
x 10^-5

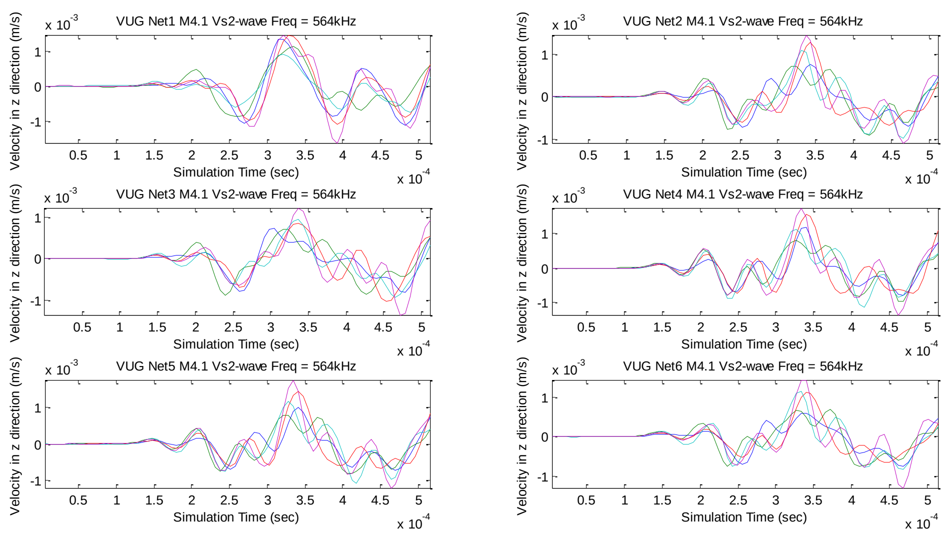

VUG Net19 M3 Vp-wave Freq = 564kHz
VUG Net20 M3 Vp-wave Freq = 564kHz
VUG Net21 M3 Vp-wave Freq = 564kHz
VUG Net22 M3 Vp-wave Freq = 564kHz
VUG Net23 M3 Vp-wave Freq = 564kHz
VUG Net24 M3 Vp-wave Freq = 564kHz
Velocity in y direction (m/s)
Simulation Time (sec)
x 10^-3
x 10^-5

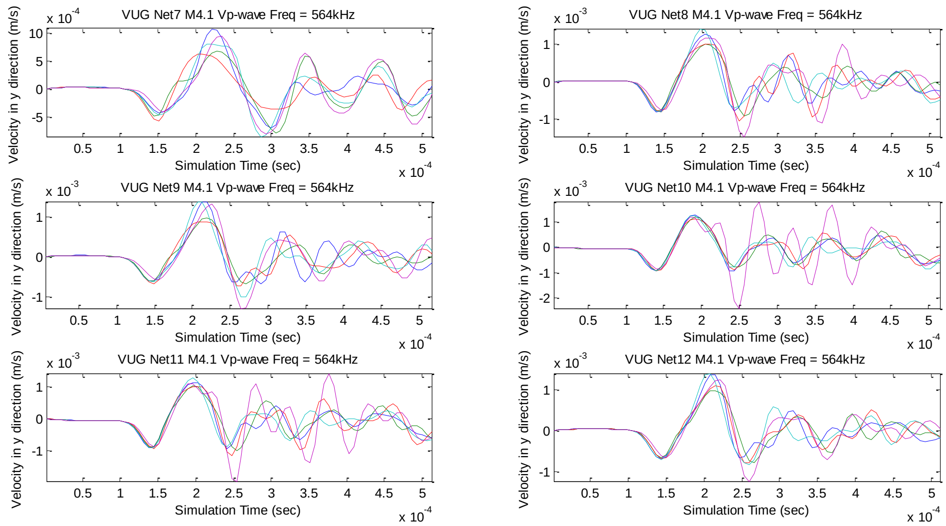

VUG Net19 M3 Vs1-wave Freq = 564kHz
VUG Net20 M3 Vs1-wave Freq = 564kHz
VUG Net21 M3 Vs1-wave Freq = 564kHz
VUG Net22 M3 Vs1-wave Freq = 564kHz
VUG Net23 M3 Vs1-wave Freq = 564kHz
VUG Net24 M3 Vs1-wave Freq = 564kHz
Velocity in x direction (m/s)
Simulation Time (sec)
x 10^-3
x 10^-5

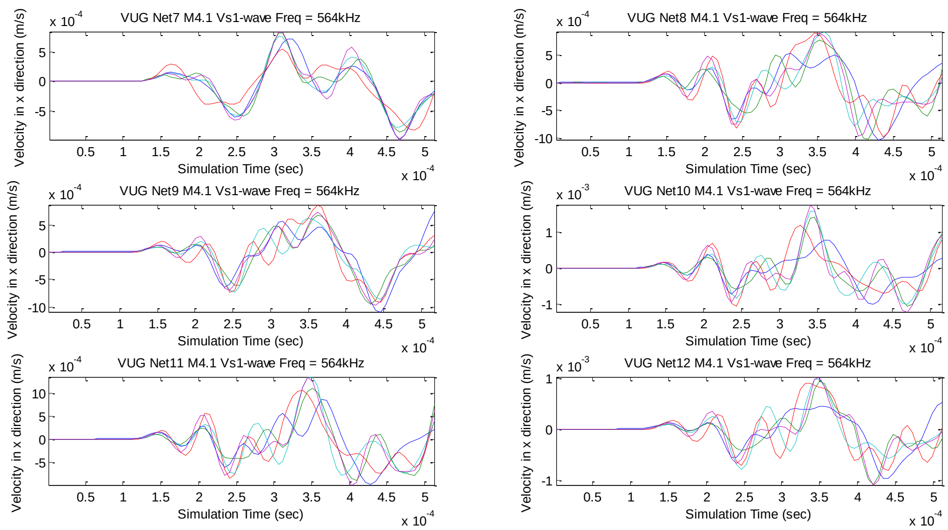

VUG Net19 M3 Vs2-wave Freq = 564kHz
VUG Net20 M3 Vs2-wave Freq = 564kHz
VUG Net21 M3 Vs2-wave Freq = 564kHz
VUG Net22 M3 Vs2-wave Freq = 564kHz
VUG Net23 M3 Vs2-wave Freq = 564kHz
VUG Net24 M3 Vs2-wave Freq = 564kHz
Velocity in z direction (m/s)
Simulation Time (sec)
x 10^-3
x 10^-5

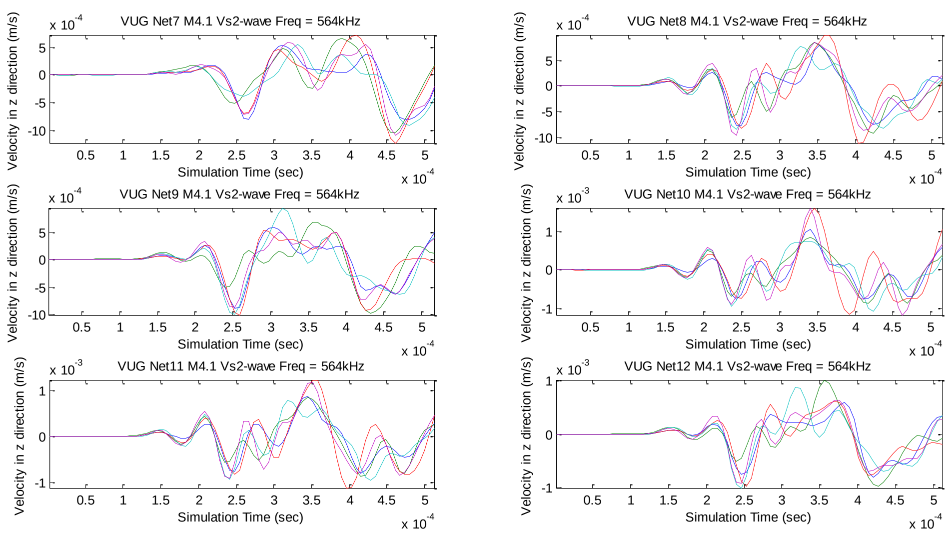

VUG Net1 M4.1 Vp-wave Freq = 564kHz
VUG Net2 M4.1 Vp-wave Freq = 564kHz
VUG Net3 M4.1 Vp-wave Freq = 564kHz
VUG Net4 M4.1 Vp-wave Freq = 564kHz
VUG Net5 M4.1 Vp-wave Freq = 564kHz
VUG Net6 M4.1 Vp-wave Freq = 564kHz
Velocity in y direction (m/s)
Simulation Time (sec)
x 10^-3
x 10^-4

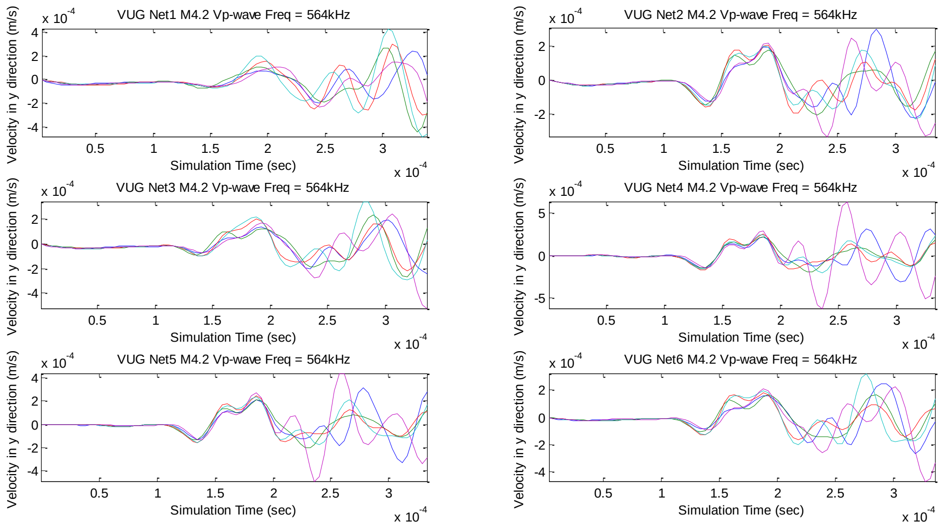

VUG Net1 M4.1 Vs1-wave Freq = 564kHz
VUG Net2 M4.1 Vs1-wave Freq = 564kHz
VUG Net3 M4.1 Vs1-wave Freq = 564kHz
VUG Net4 M4.1 Vs1-wave Freq = 564kHz
VUG Net5 M4.1 Vs1-wave Freq = 564kHz
VUG Net6 M4.1 Vs1-wave Freq = 564kHz
Velocity in x direction (m/s)
Simulation Time (sec)
x 10^-3
x 10^-4

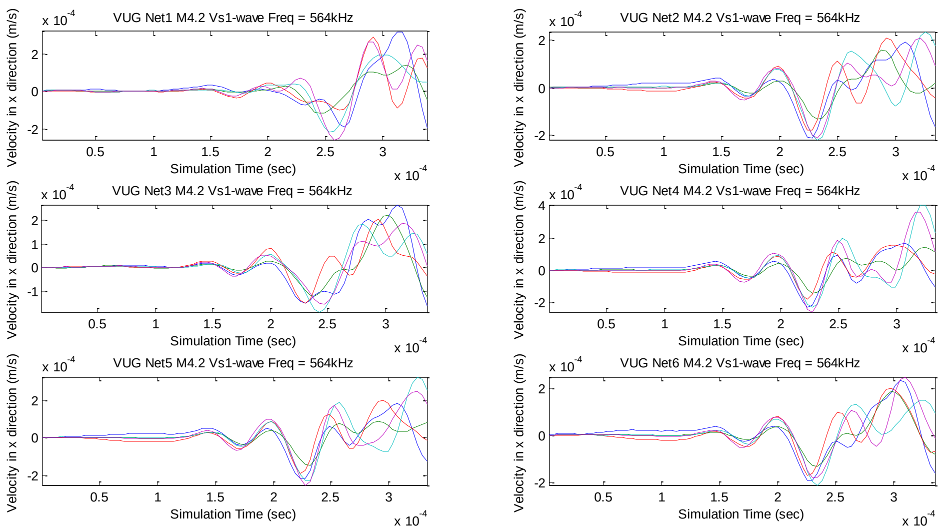

VUG Net1 M4.1 Vs2-wave Freq = 564kHz
VUG Net2 M4.1 Vs2-wave Freq = 564kHz
VUG Net3 M4.1 Vs2-wave Freq = 564kHz
VUG Net4 M4.1 Vs2-wave Freq = 564kHz
VUG Net5 M4.1 Vs2-wave Freq = 564kHz
VUG Net6 M4.1 Vs2-wave Freq = 564kHz
Velocity in z direction (m/s)
Simulation Time (sec)
x 10^-3
x 10^-4

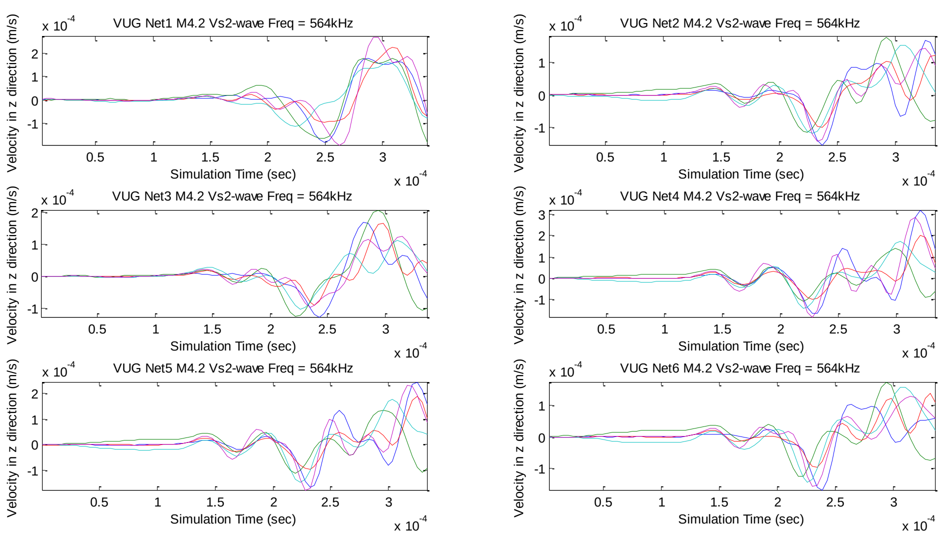

VUG Net7 M4.1 Vp-wave Freq = 564kHz
VUG Net8 M4.1 Vp-wave Freq = 564kHz
VUG Net9 M4.1 Vp-wave Freq = 564kHz
VUG Net10 M4.1 Vp-wave Freq = 564kHz
VUG Net11 M4.1 Vp-wave Freq = 564kHz
VUG Net12 M4.1 Vp-wave Freq = 564kHz
Velocity in y direction (m/s)
Simulation Time (sec)
x 10^-4
x 10^-3

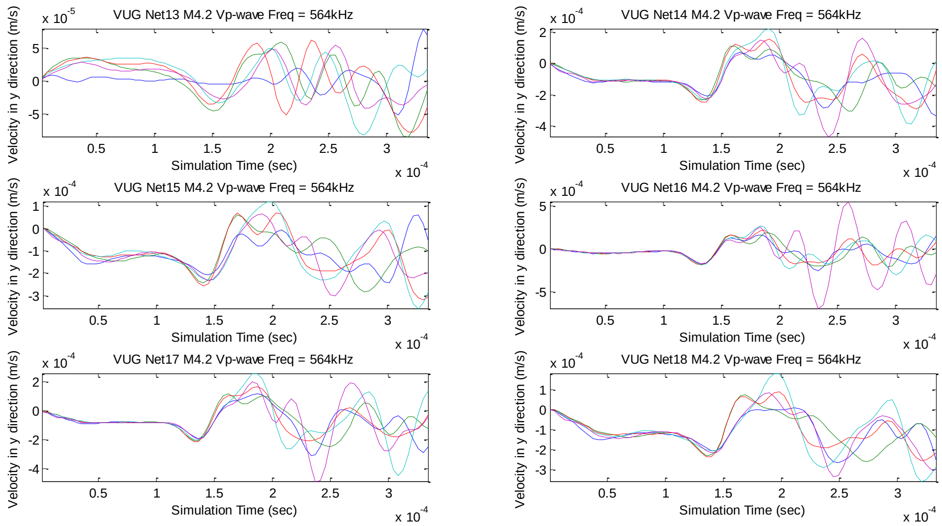

VUG Net7 M4.1 Vs1-wave Freq = 564kHz
VUG Net8 M4.1 Vs1-wave Freq = 564kHz
VUG Net9 M4.1 Vs1-wave Freq = 564kHz
VUG Net10 M4.1 Vs1-wave Freq = 564kHz
VUG Net11 M4.1 Vs1-wave Freq = 564kHz
VUG Net12 M4.1 Vs1-wave Freq = 564kHz
Velocity in x direction (m/s)
Simulation Time (sec)
x 10^-4
x 10^-3

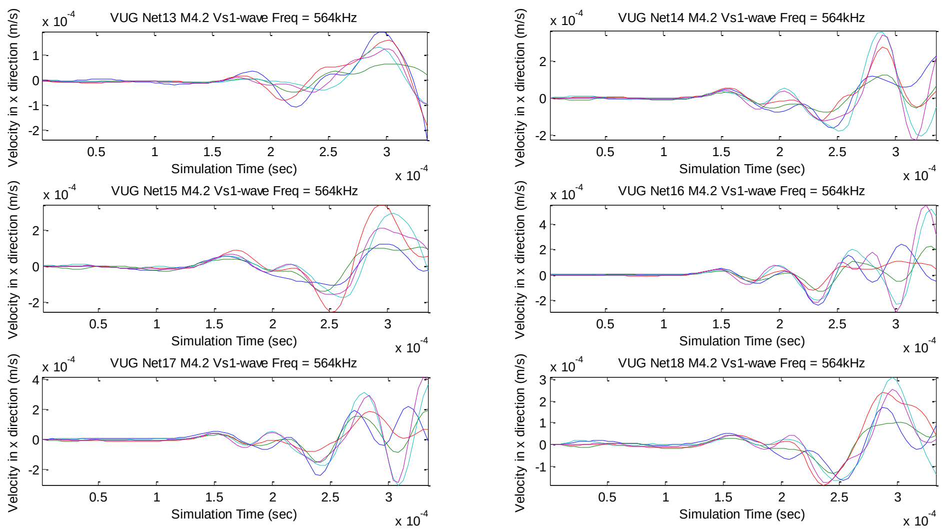

VUG Net7 M4.1 Vs2-wave Freq = 564kHz
VUG Net8 M4.1 Vs2-wave Freq = 564kHz
VUG Net9 M4.1 Vs2-wave Freq = 564kHz
VUG Net10 M4.1 Vs2-wave Freq = 564kHz
VUG Net11 M4.1 Vs2-wave Freq = 564kHz
VUG Net12 M4.1 Vs2-wave Freq = 564kHz
Velocity in z direction (m/s)
Simulation Time (sec)
x 10^-4
x 10^-3

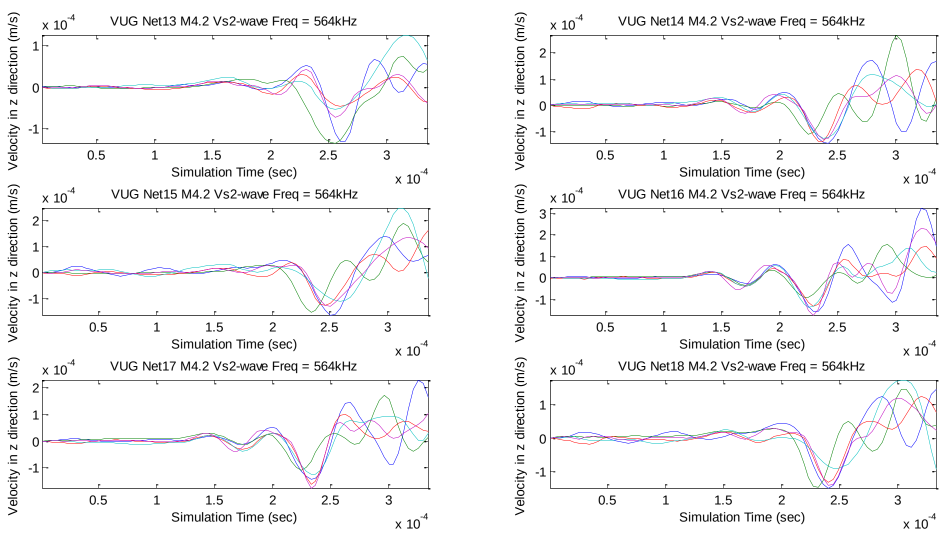

VUG Net1 M4.2 Vp-wave Freq = 564kHz
VUG Net2 M4.2 Vp-wave Freq = 564kHz
VUG Net3 M4.2 Vp-wave Freq = 564kHz
VUG Net4 M4.2 Vp-wave Freq = 564kHz
VUG Net5 M4.2 Vp-wave Freq = 564kHz
VUG Net6 M4.2 Vp-wave Freq = 564kHz
Velocity in y direction (m/s)
Simulation Time (sec)
x 10^-4

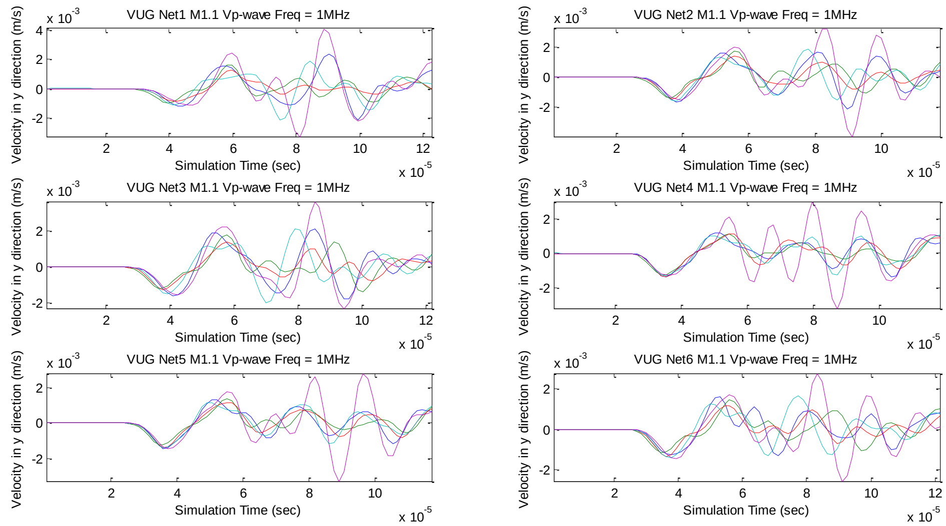

VUG Net1 M4.2 Vs1-wave Freq = 564kHz
VUG Net2 M4.2 Vs1-wave Freq = 564kHz
VUG Net3 M4.2 Vs1-wave Freq = 564kHz
VUG Net4 M4.2 Vs1-wave Freq = 564kHz
VUG Net5 M4.2 Vs1-wave Freq = 564kHz
VUG Net6 M4.2 Vs1-wave Freq = 564kHz
Velocity in x direction (m/s)
Simulation Time (sec)
x 10^-4

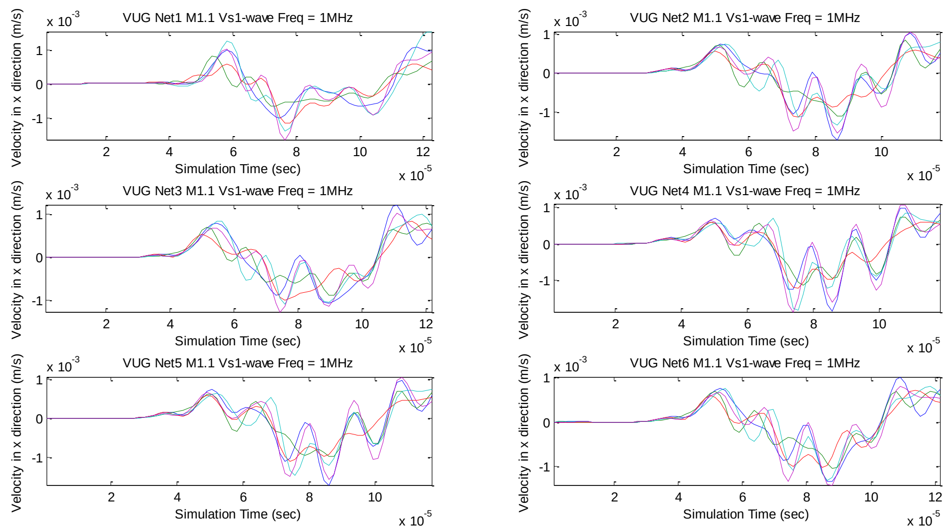

VUG Net1 M4.2 Vs2-wave Freq = 564kHz
VUG Net2 M4.2 Vs2-wave Freq = 564kHz
VUG Net3 M4.2 Vs2-wave Freq = 564kHz
VUG Net4 M4.2 Vs2-wave Freq = 564kHz
VUG Net5 M4.2 Vs2-wave Freq = 564kHz
VUG Net6 M4.2 Vs2-wave Freq = 564kHz
Velocity in z direction (m/s)
Simulation Time (sec)
x 10^-4

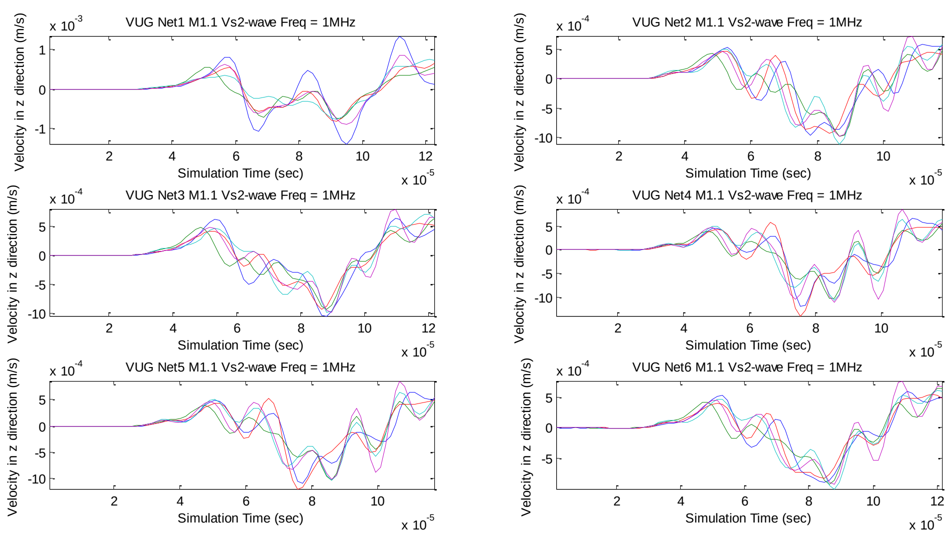

VUG Net13 M4.2 Vp-wave Freq = 564kHz
VUG Net14 M4.2 Vp-wave Freq = 564kHz
VUG Net15 M4.2 Vp-wave Freq = 564kHz
VUG Net16 M4.2 Vp-wave Freq = 564kHz
VUG Net17 M4.2 Vp-wave Freq = 564kHz
VUG Net18 M4.2 Vp-wave Freq = 564kHz
Velocity in y direction (m/s)
Simulation Time (sec)
x 10^-4

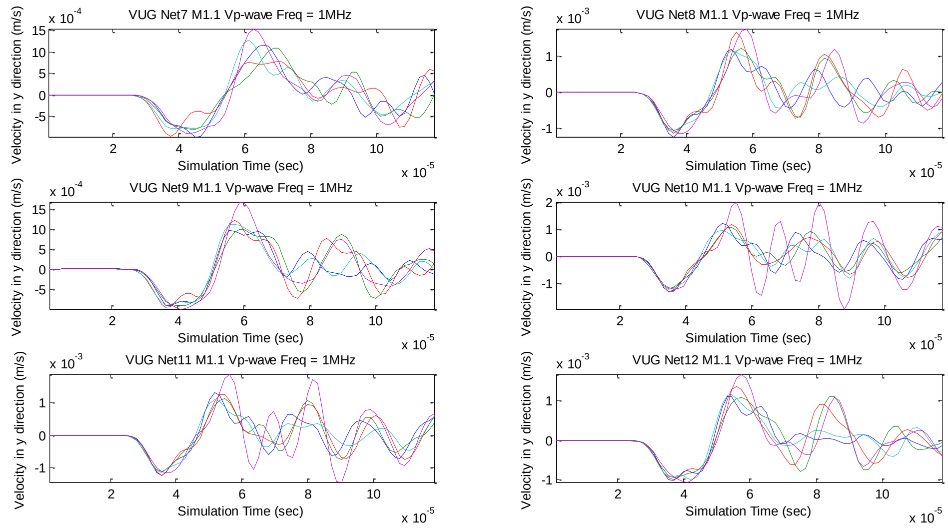

VUG Net13 M4.2 Vs1-wave Freq = 564kHz
VUG Net14 M4.2 Vs1-wave Freq = 564kHz
VUG Net15 M4.2 Vs1-wave Freq = 564kHz
VUG Net16 M4.2 Vs1-wave Freq = 564kHz
VUG Net17 M4.2 Vs1-wave Freq = 564kHz
VUG Net18 M4.2 Vs1-wave Freq = 564kHz
Velocity in x direction (m/s)
Simulation Time (sec)
x 10^-4

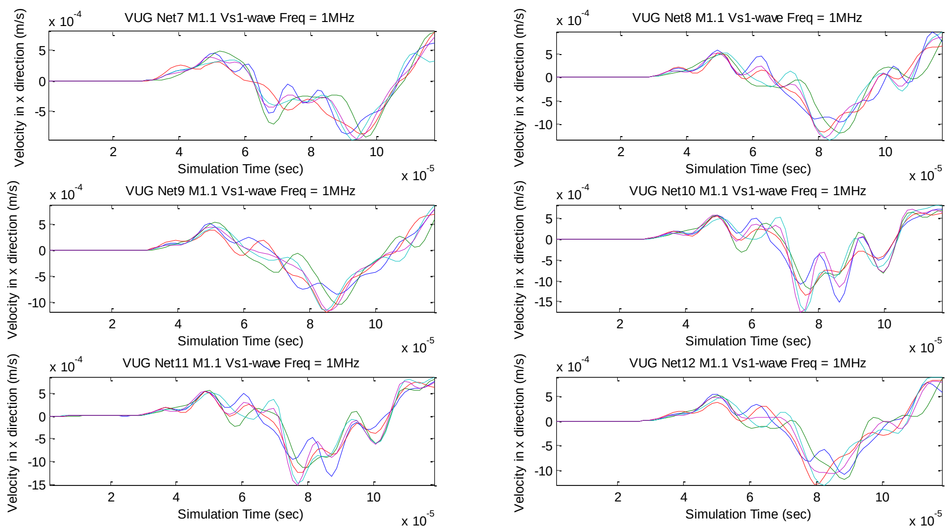

VUG Net13 M4.2 Vs2-wave Freq = 564kHz
VUG Net14 M4.2 Vs2-wave Freq = 564kHz
VUG Net15 M4.2 Vs2-wave Freq = 564kHz
VUG Net16 M4.2 Vs2-wave Freq = 564kHz
VUG Net17 M4.2 Vs2-wave Freq = 564kHz
VUG Net18 M4.2 Vs2-wave Freq = 564kHz
Velocity in z direction (m/s)
Simulation Time (sec)
x 10^-4

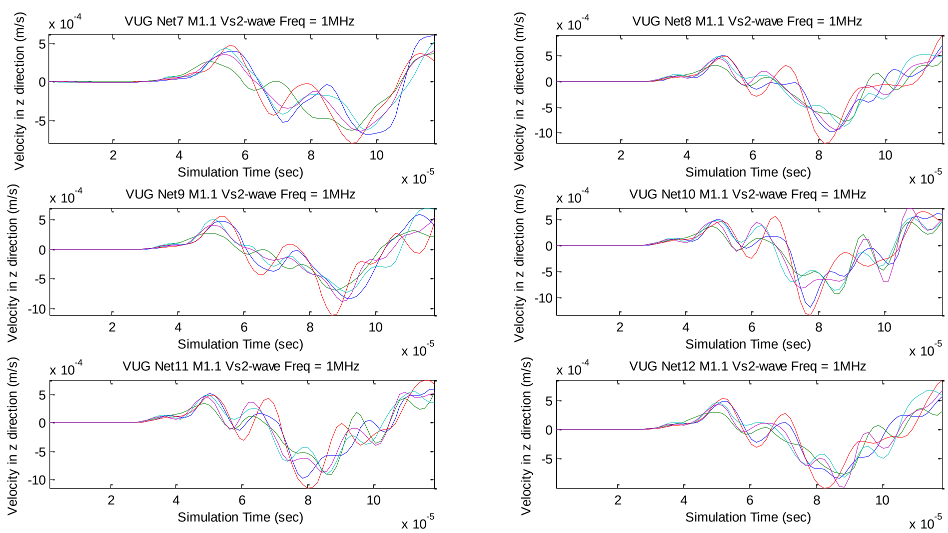

VUG Net1 M1.1 Vp-wave Freq = 1MHz
VUG Net2 M1.1 Vp-wave Freq = 1MHz
VUG Net3 M1.1 Vp-wave Freq = 1MHz
VUG Net4 M1.1 Vp-wave Freq = 1MHz
VUG Net5 M1.1 Vp-wave Freq = 1MHz
VUG Net6 M1.1 Vp-wave Freq = 1MHz
Velocity in y direction (m/s)
Simulation Time (sec)
x 10^-3
x 10^-5

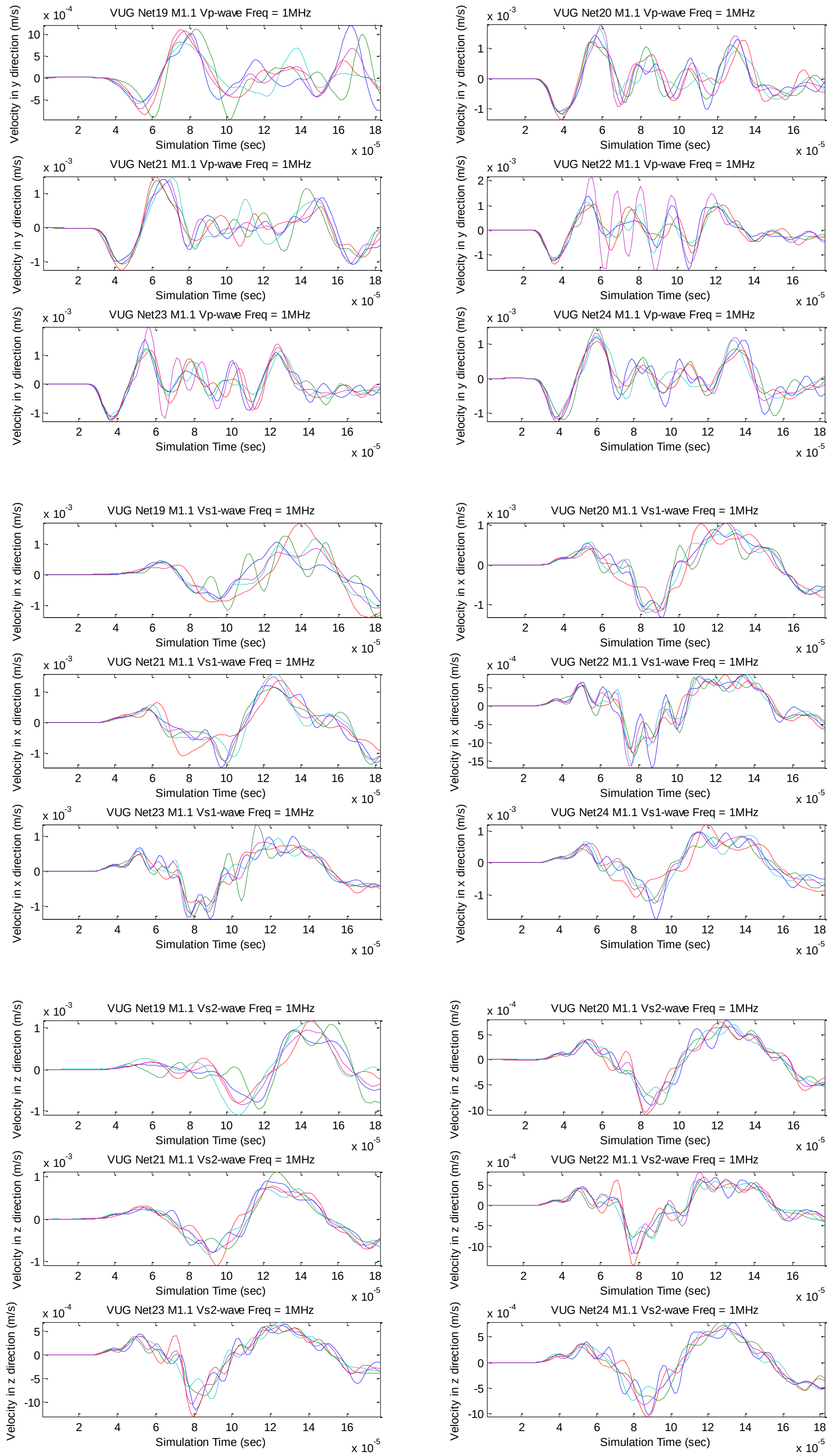

VUG Net1 M1.1 Vs1-wave Freq = 1MHz
VUG Net2 M1.1 Vs1-wave Freq = 1MHz
VUG Net3 M1.1 Vs1-wave Freq = 1MHz
VUG Net4 M1.1 Vs1-wave Freq = 1MHz
VUG Net5 M1.1 Vs1-wave Freq = 1MHz
VUG Net6 M1.1 Vs1-wave Freq = 1MHz
Velocity in x direction (m/s)
Simulation Time (sec)
x 10^-3
x 10^-5

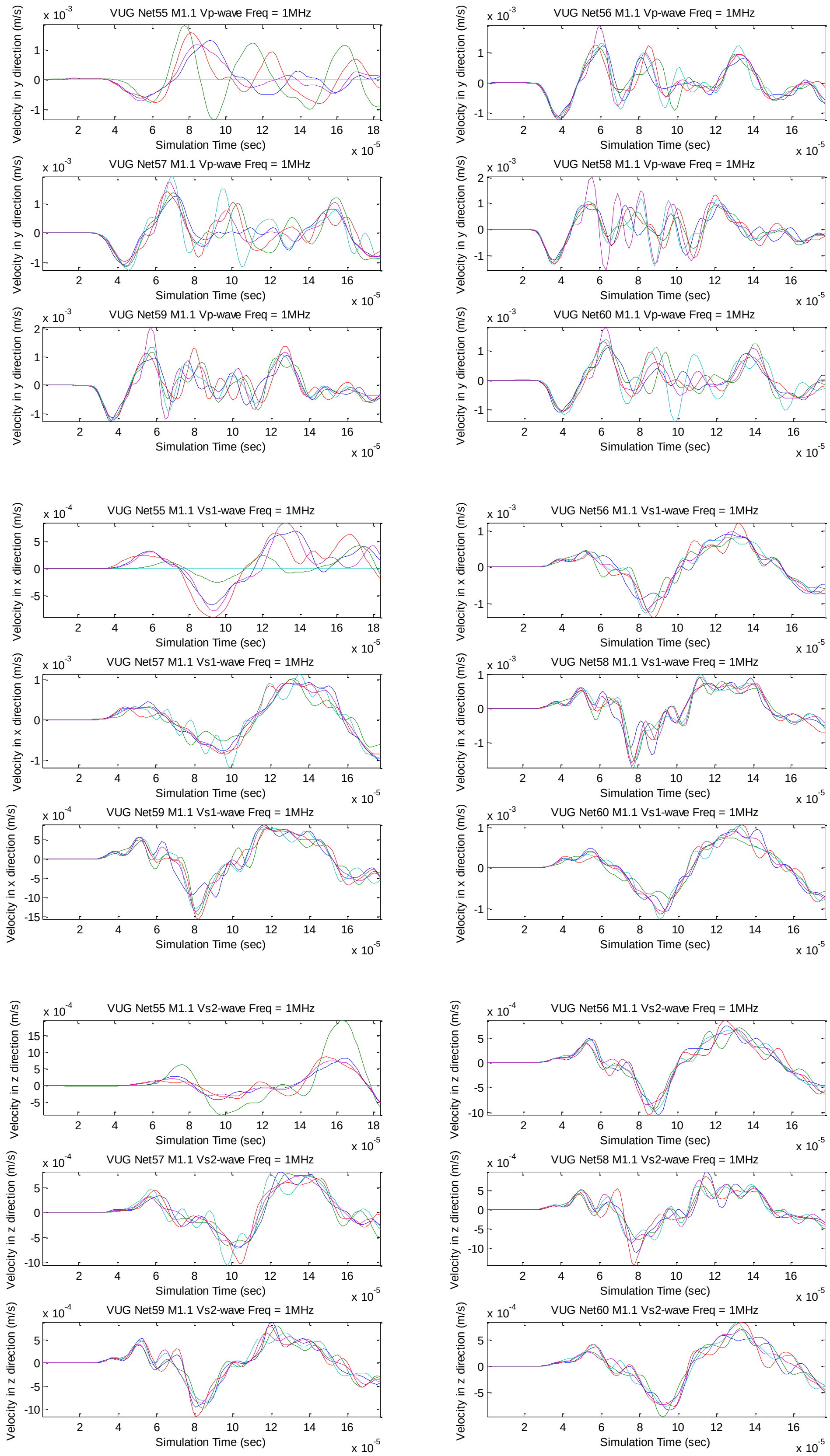

VUG Net1 M1.1 Vs2-wave Freq = 1MHz
VUG Net2 M1.1 Vs2-wave Freq = 1MHz
VUG Net3 M1.1 Vs2-wave Freq = 1MHz
VUG Net4 M1.1 Vs2-wave Freq = 1MHz
VUG Net5 M1.1 Vs2-wave Freq = 1MHz
VUG Net6 M1.1 Vs2-wave Freq = 1MHz
Velocity in z direction (m/s)
Simulation Time (sec)
x 10^-3
x 10^-4
x 10^-5

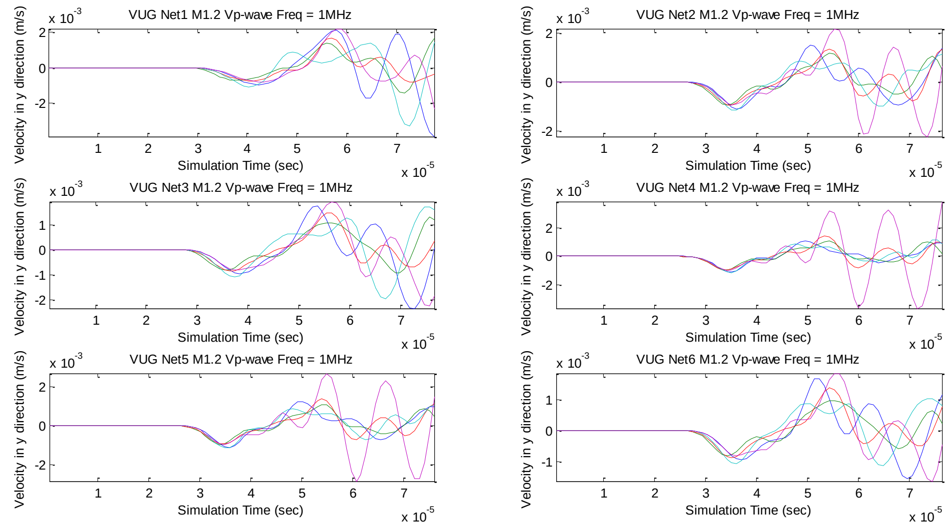

VUG Net7 M1.1 Vp-wave Freq = 1MHz
VUG Net8 M1.1 Vp-wave Freq = 1MHz
VUG Net9 M1.1 Vp-wave Freq = 1MHz
VUG Net10 M1.1 Vp-wave Freq = 1MHz
VUG Net11 M1.1 Vp-wave Freq = 1MHz
VUG Net12 M1.1 Vp-wave Freq = 1MHz
Velocity in y direction (m/s)
Simulation Time (sec)
x 10^-4
x 10^-3
x 10^-5

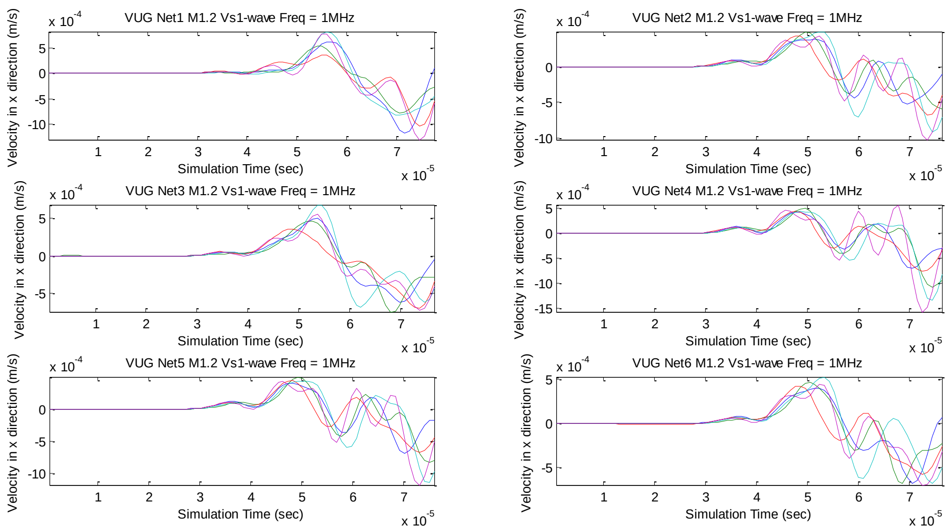

VUG Net7 M1.1 Vs1-wave Freq = 1MHz
VUG Net8 M1.1 Vs1-wave Freq = 1MHz
VUG Net9 M1.1 Vs1-wave Freq = 1MHz
VUG Net10 M1.1 Vs1-wave Freq = 1MHz
VUG Net11 M1.1 Vs1-wave Freq = 1MHz
VUG Net12 M1.1 Vs1-wave Freq = 1MHz
Velocity in x direction (m/s)
Simulation Time (sec)
x 10^-4
x 10^-5

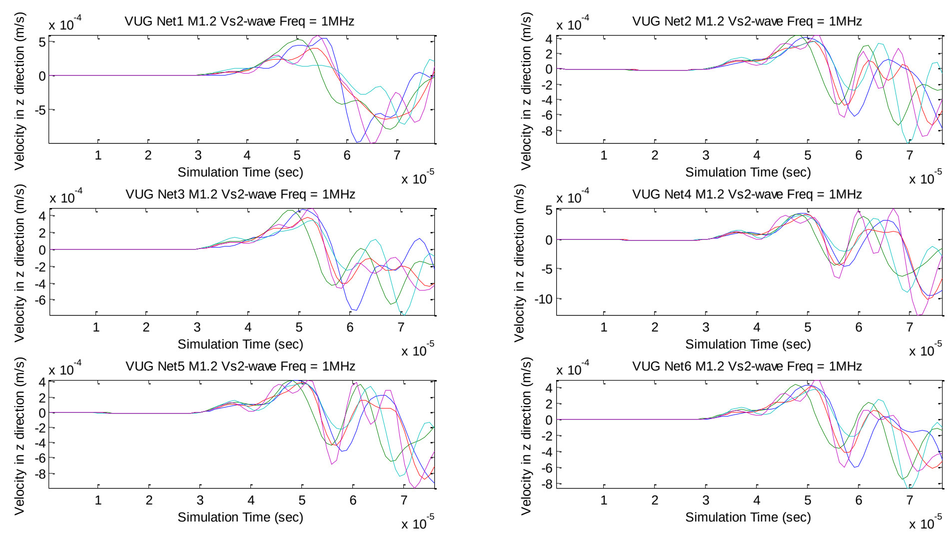

VUG Net7 M1.1 Vs2-wave Freq = 1MHz
VUG Net8 M1.1 Vs2-wave Freq = 1MHz
VUG Net9 M1.1 Vs2-wave Freq = 1MHz
VUG Net10 M1.1 Vs2-wave Freq = 1MHz
VUG Net11 M1.1 Vs2-wave Freq = 1MHz
VUG Net12 M1.1 Vs2-wave Freq = 1MHz
Velocity in z direction (m/s)
Simulation Time (sec)
x 10^-4
x 10^-5

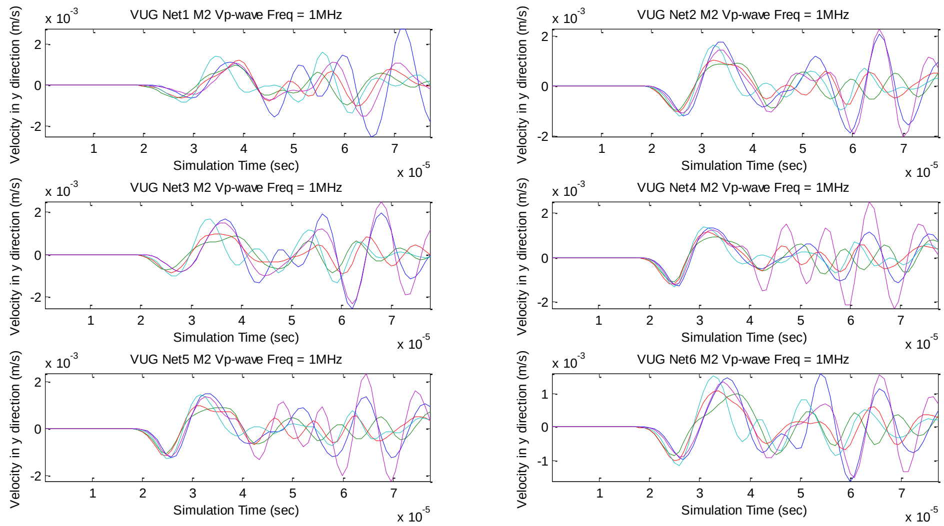

VUG Net19 M1.1 Vp-wave Freq = 1MHz
VUG Net20 M1.1 Vp-wave Freq = 1MHz
VUG Net21 M1.1 Vp-wave Freq = 1MHz
VUG Net22 M1.1 Vp-wave Freq = 1MHz
VUG Net23 M1.1 Vp-wave Freq = 1MHz
VUG Net24 M1.1 Vp-wave Freq = 1MHz
Velocity in y direction (m/s)
VUG Net19 M1.1 Vs1-wave Freq = 1MHz
VUG Net20 M1.1 Vs1-wave Freq = 1MHz
VUG Net21 M1.1 Vs1-wave Freq = 1MHz
VUG Net22 M1.1 Vs1-wave Freq = 1MHz
VUG Net23 M1.1 Vs1-wave Freq = 1MHz
VUG Net24 M1.1 Vs1-wave Freq = 1MHz
Velocity in x direction (m/s)
VUG Net19 M1.1 Vs2-wave Freq = 1MHz
VUG Net20 M1.1 Vs2-wave Freq = 1MHz
VUG Net21 M1.1 Vs2-wave Freq = 1MHz
VUG Net22 M1.1 Vs2-wave Freq = 1MHz
VUG Net23 M1.1 Vs2-wave Freq = 1MHz
VUG Net24 M1.1 Vs2-wave Freq = 1MHz
Velocity in z direction (m/s)
Simulation Time (sec)
x 10^-5
x 10^-3
x 10^-4

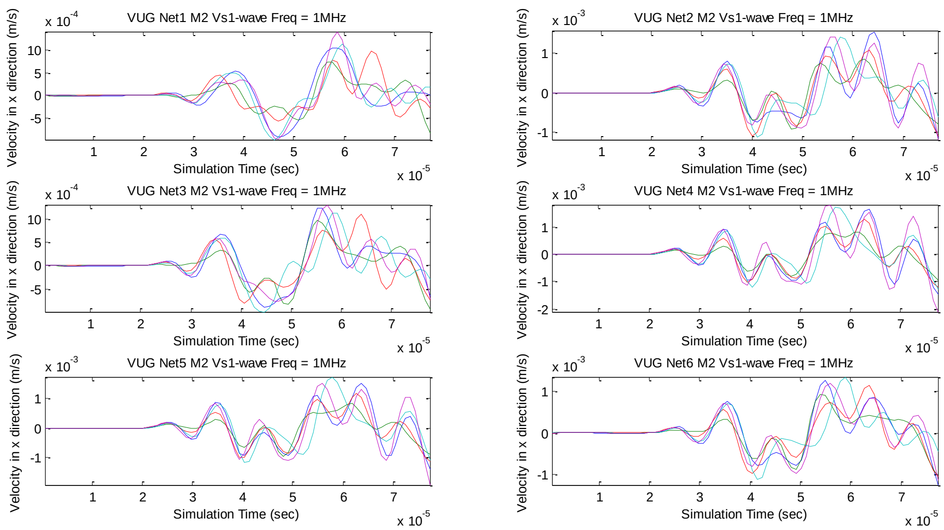

VUG Net55 M1.1 Vp-wave Freq = 1MHz
VUG Net56 M1.1 Vp-wave Freq = 1MHz
VUG Net57 M1.1 Vp-wave Freq = 1MHz
VUG Net58 M1.1 Vp-wave Freq = 1MHz
VUG Net59 M1.1 Vp-wave Freq = 1MHz
VUG Net60 M1.1 Vp-wave Freq = 1MHz
Velocity in y direction (m/s)
VUG Net55 M1.1 Vs1-wave Freq = 1MHz
VUG Net56 M1.1 Vs1-wave Freq = 1MHz
VUG Net57 M1.1 Vs1-wave Freq = 1MHz
VUG Net58 M1.1 Vs1-wave Freq = 1MHz
VUG Net59 M1.1 Vs1-wave Freq = 1MHz
VUG Net60 M1.1 Vs1-wave Freq = 1MHz
Velocity in x direction (m/s)
VUG Net55 M1.1 Vs2-wave Freq = 1MHz
VUG Net56 M1.1 Vs2-wave Freq = 1MHz
VUG Net57 M1.1 Vs2-wave Freq = 1MHz
VUG Net58 M1.1 Vs2-wave Freq = 1MHz
VUG Net59 M1.1 Vs2-wave Freq = 1MHz
VUG Net60 M1.1 Vs2-wave Freq = 1MHz
Velocity in z direction (m/s)
Simulation Time (sec)
x 10^-5
x 10^-3
x 10^-4

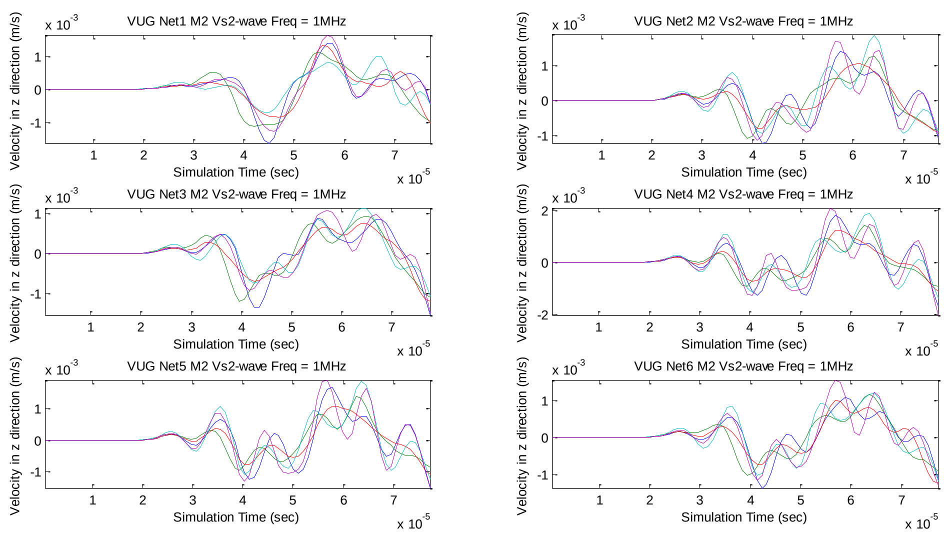

VUG Net1 M1.2 Vp-wave Freq = 1MHz
VUG Net2 M1.2 Vp-wave Freq = 1MHz
VUG Net3 M1.2 Vp-wave Freq = 1MHz
VUG Net4 M1.2 Vp-wave Freq = 1MHz
VUG Net5 M1.2 Vp-wave Freq = 1MHz
VUG Net6 M1.2 Vp-wave Freq = 1MHz
Velocity in y direction (m/s)
x 10-3
Simulation Time (sec)
x 10-5

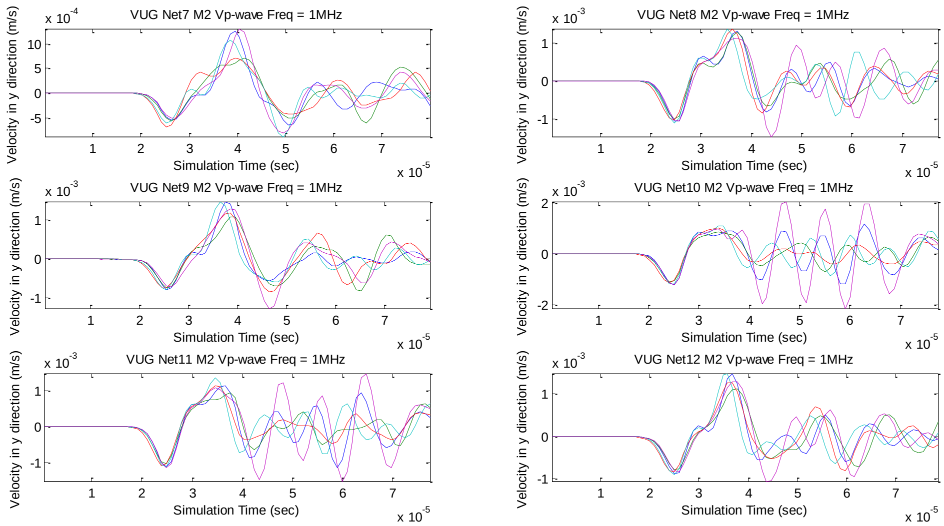

VUG Net1 M1.2 Vs1-wave Freq = 1MHz
VUG Net2 M1.2 Vs1-wave Freq = 1MHz
VUG Net3 M1.2 Vs1-wave Freq = 1MHz
VUG Net4 M1.2 Vs1-wave Freq = 1MHz
VUG Net5 M1.2 Vs1-wave Freq = 1MHz
VUG Net6 M1.2 Vs1-wave Freq = 1MHz
Velocity in x direction (m/s)
x 10-4
Simulation Time (sec)
x 10-5

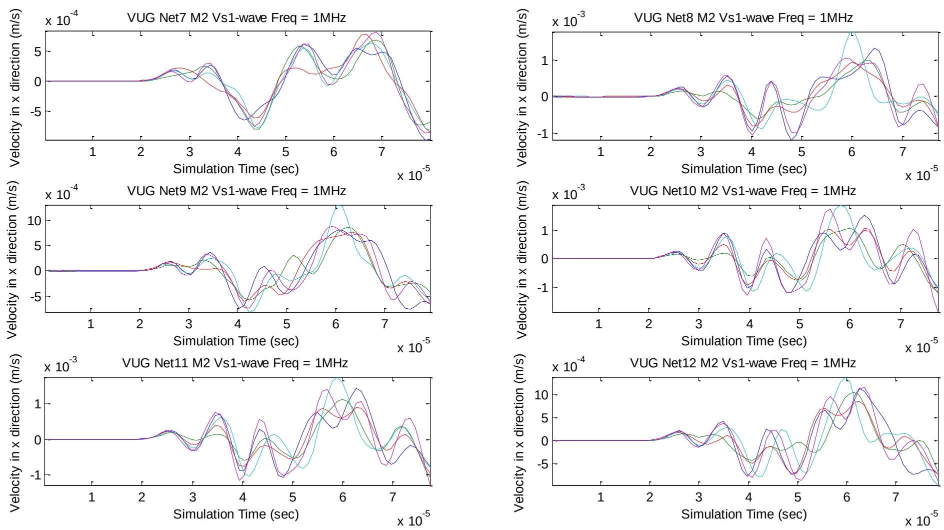

VUG Net1 M1.2 Vs2-wave Freq = 1MHz
VUG Net2 M1.2 Vs2-wave Freq = 1MHz
VUG Net3 M1.2 Vs2-wave Freq = 1MHz
VUG Net4 M1.2 Vs2-wave Freq = 1MHz
VUG Net5 M1.2 Vs2-wave Freq = 1MHz
VUG Net6 M1.2 Vs2-wave Freq = 1MHz
Velocity in z direction (m/s)
x 10-4
Simulation Time (sec)
x 10-5

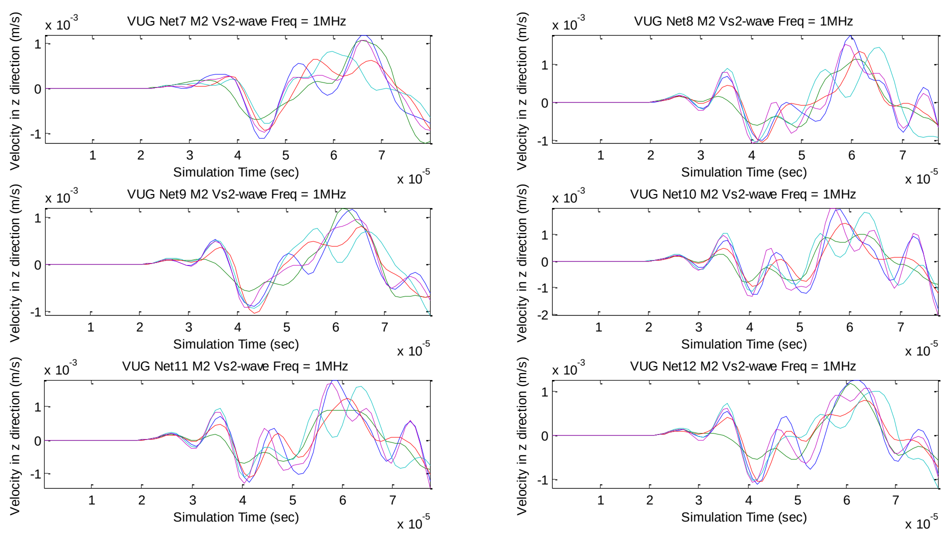

VUG Net1 M2 Vp-wave Freq = 1MHz
VUG Net2 M2 Vp-wave Freq = 1MHz
VUG Net3 M2 Vp-wave Freq = 1MHz
VUG Net4 M2 Vp-wave Freq = 1MHz
VUG Net5 M2 Vp-wave Freq = 1MHz
VUG Net6 M2 Vp-wave Freq = 1MHz
Velocity in y direction (m/s)
x 10^{-3}
Simulation Time (sec)
x 10^{-5}

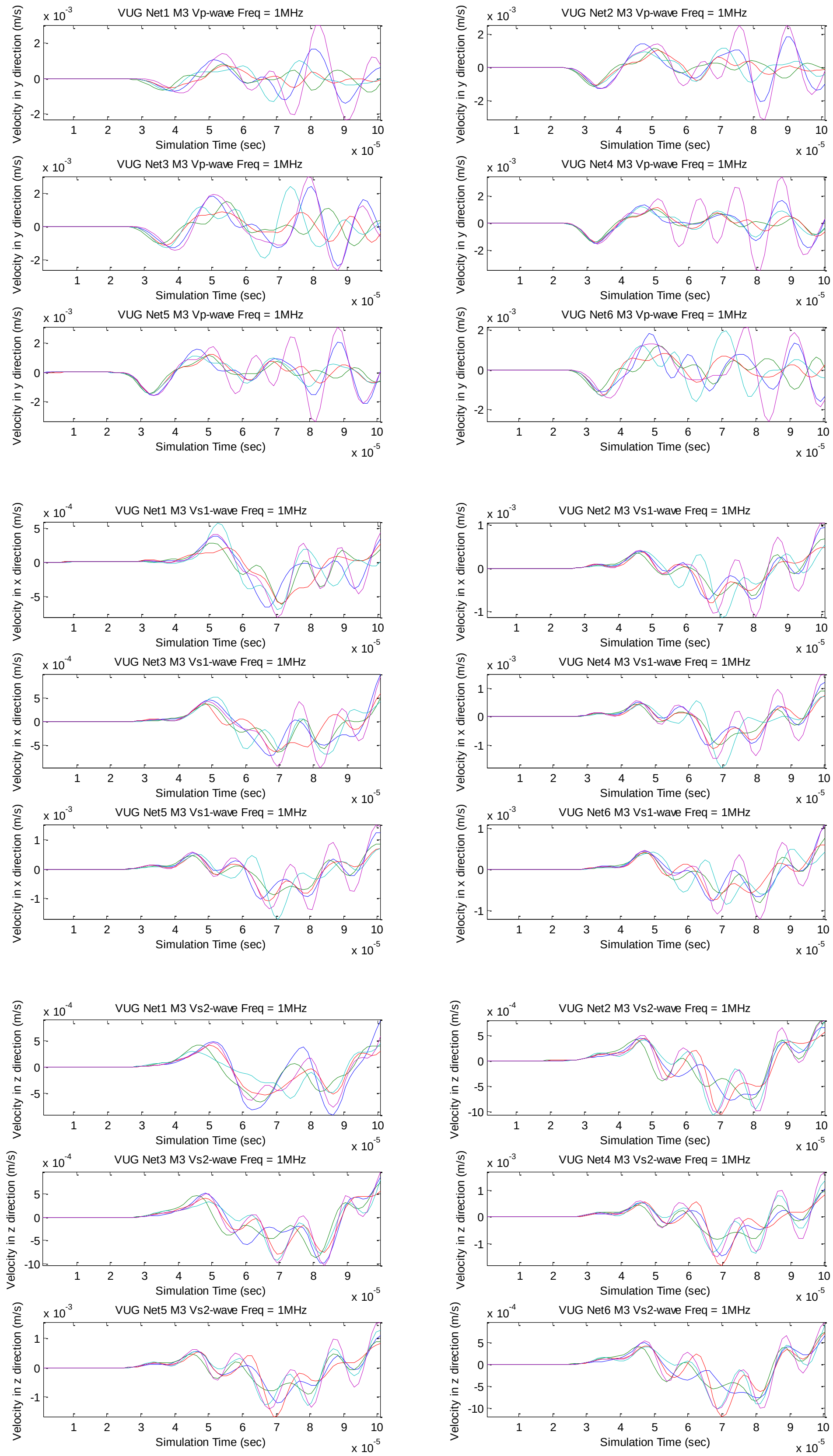

VUG Net1 M2 Vs1-wave Freq = 1MHz
VUG Net2 M2 Vs1-wave Freq = 1MHz
VUG Net3 M2 Vs1-wave Freq = 1MHz
VUG Net4 M2 Vs1-wave Freq = 1MHz
VUG Net5 M2 Vs1-wave Freq = 1MHz
VUG Net6 M2 Vs1-wave Freq = 1MHz
Velocity in x direction (m/s)
x 10^{-4}
x 10^{-3}
Simulation Time (sec)
x 10^{-5}

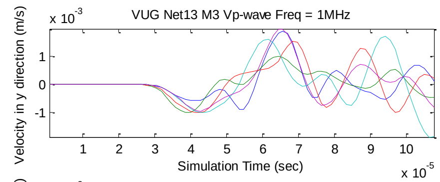
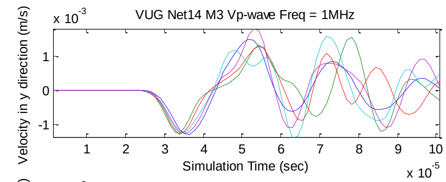
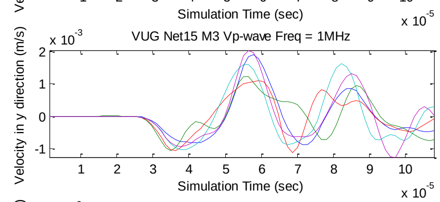
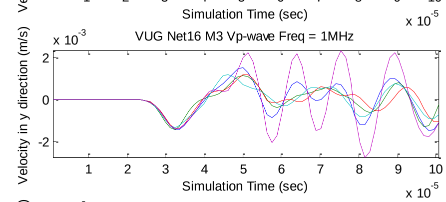
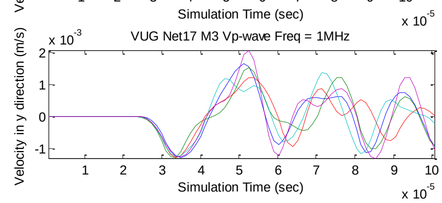
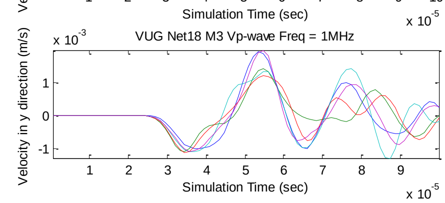
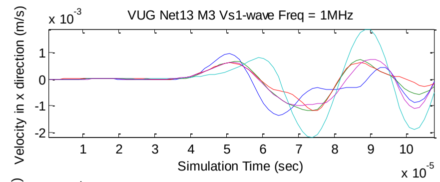
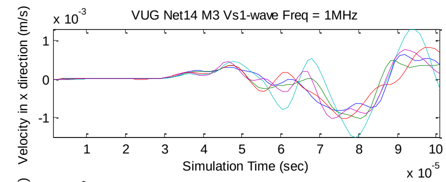
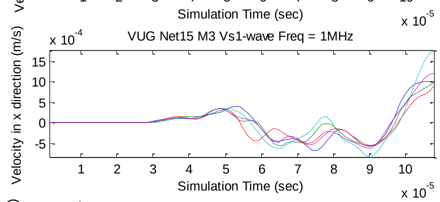
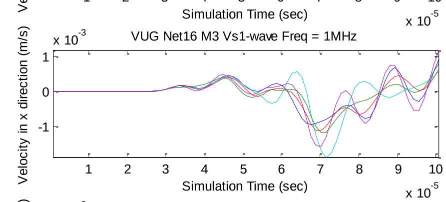
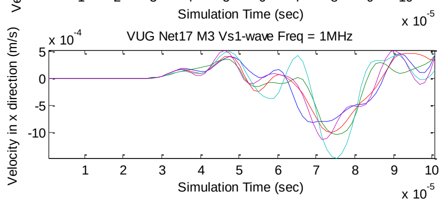
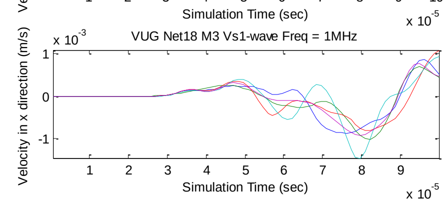
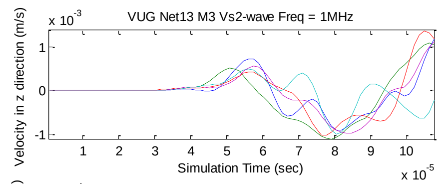
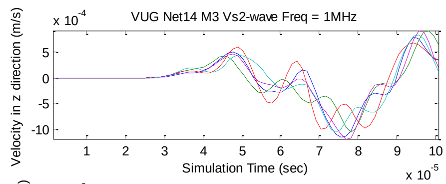
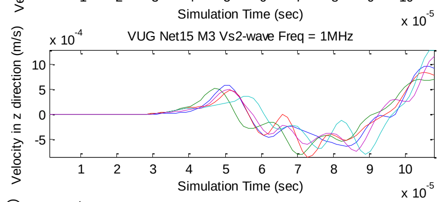
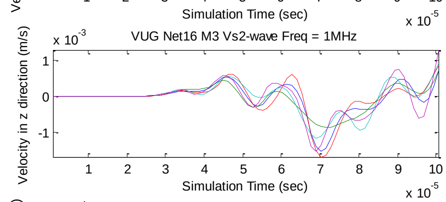
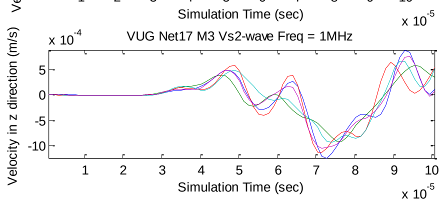
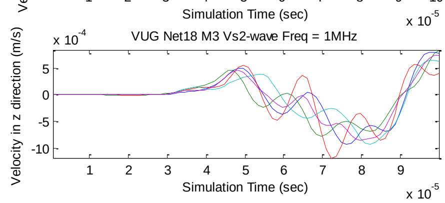

VUG Net1 M2 Vs2-wave Freq = 1MHz
VUG Net2 M2 Vs2-wave Freq = 1MHz
VUG Net3 M2 Vs2-wave Freq = 1MHz
VUG Net4 M2 Vs2-wave Freq = 1MHz
VUG Net5 M2 Vs2-wave Freq = 1MHz
VUG Net6 M2 Vs2-wave Freq = 1MHz
Velocity in z direction (m/s)
x 10^{-3}
Simulation Time (sec)
x 10^{-5}

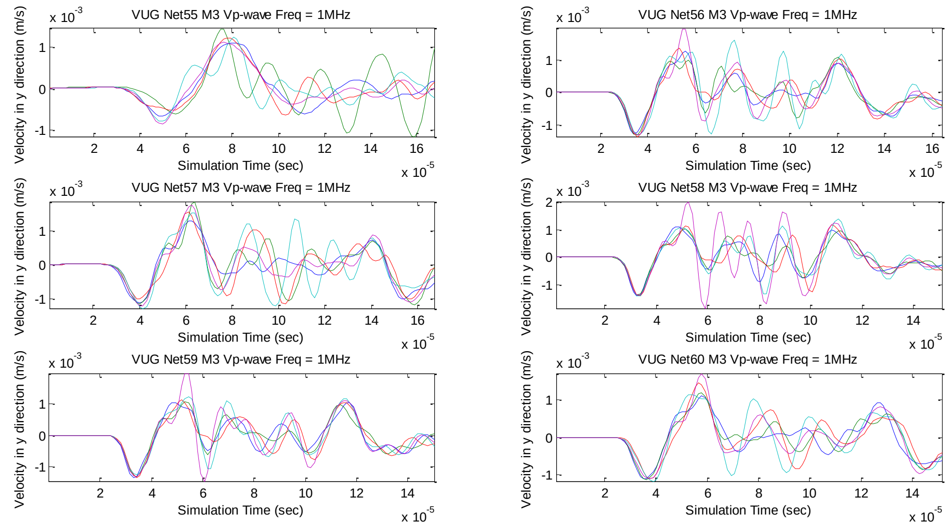

VUG Net7 M2 Vp-wave Freq = 1MHz
VUG Net8 M2 Vp-wave Freq = 1MHz
VUG Net9 M2 Vp-wave Freq = 1MHz
VUG Net10 M2 Vp-wave Freq = 1MHz
VUG Net11 M2 Vp-wave Freq = 1MHz
VUG Net12 M2 Vp-wave Freq = 1MHz
Velocity in y direction (m/s)
Simulation Time (sec)
x 10^-4
x 10^-3
x 10^-5

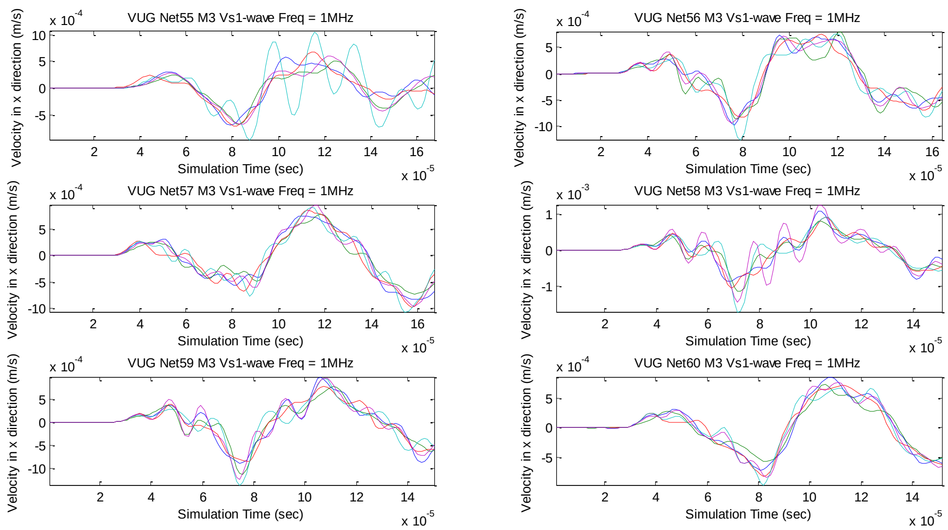

VUG Net7 M2 Vs1-wave Freq = 1MHz
VUG Net8 M2 Vs1-wave Freq = 1MHz
VUG Net9 M2 Vs1-wave Freq = 1MHz
VUG Net10 M2 Vs1-wave Freq = 1MHz
VUG Net11 M2 Vs1-wave Freq = 1MHz
VUG Net12 M2 Vs1-wave Freq = 1MHz
Velocity in x direction (m/s)
Simulation Time (sec)
x 10^-4
x 10^-3
x 10^-5

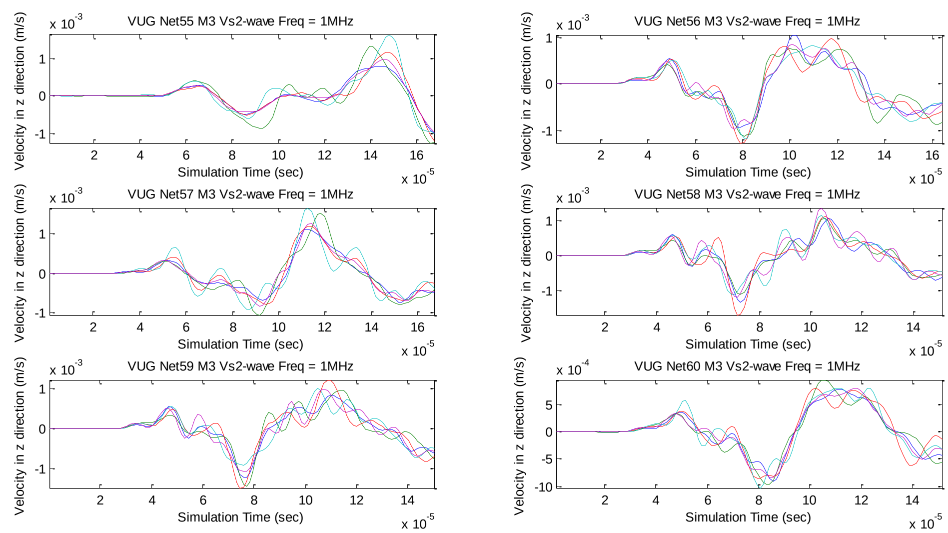

VUG Net7 M2 Vs2-wave Freq = 1MHz
VUG Net8 M2 Vs2-wave Freq = 1MHz
VUG Net9 M2 Vs2-wave Freq = 1MHz
VUG Net10 M2 Vs2-wave Freq = 1MHz
VUG Net11 M2 Vs2-wave Freq = 1MHz
VUG Net12 M2 Vs2-wave Freq = 1MHz
Velocity in z direction (m/s)
Simulation Time (sec)
x 10^-3
x 10^-5

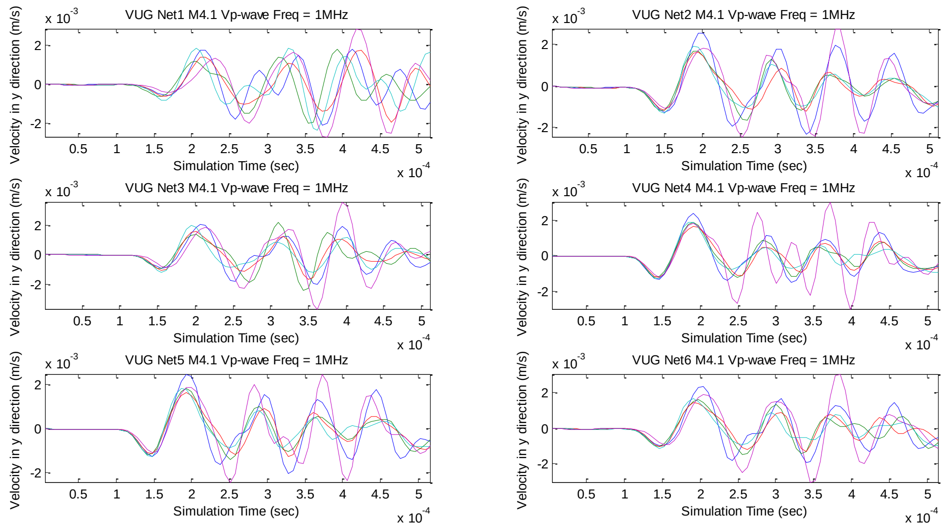

VUG Net1 M3 Vp-wave Freq = 1MHz
VUG Net2 M3 Vp-wave Freq = 1MHz
VUG Net3 M3 Vp-wave Freq = 1MHz
VUG Net4 M3 Vp-wave Freq = 1MHz
VUG Net5 M3 Vp-wave Freq = 1MHz
VUG Net6 M3 Vp-wave Freq = 1MHz
Velocity in y direction (m/s)
VUG Net1 M3 Vs1-wave Freq = 1MHz
VUG Net2 M3 Vs1-wave Freq = 1MHz
VUG Net3 M3 Vs1-wave Freq = 1MHz
VUG Net4 M3 Vs1-wave Freq = 1MHz
VUG Net5 M3 Vs1-wave Freq = 1MHz
VUG Net6 M3 Vs1-wave Freq = 1MHz
Velocity in x direction (m/s)
VUG Net1 M3 Vs2-wave Freq = 1MHz
VUG Net2 M3 Vs2-wave Freq = 1MHz
VUG Net3 M3 Vs2-wave Freq = 1MHz
VUG Net4 M3 Vs2-wave Freq = 1MHz
VUG Net5 M3 Vs2-wave Freq = 1MHz
VUG Net6 M3 Vs2-wave Freq = 1MHz
Velocity in z direction (m/s)
Simulation Time (sec)

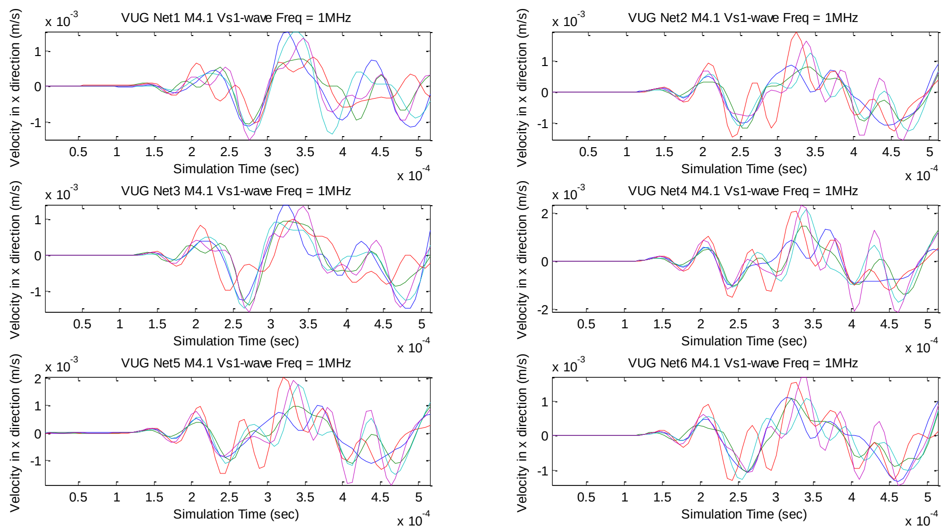

VUG Net13 M3 Vp-wave Freq = 1MHz
Velocity in y direction (m/s)
x 10^-3
Simulation Time (sec)
x 10^-5

VUG Net14 M3 Vp-wave Freq = 1MHz
Velocity in y direction (m/s)
x 10^-3
Simulation Time (sec)
x 10^-5

VUG Net15 M3 Vp-wave Freq = 1MHz
Velocity in y direction (m/s)
x 10^-3
Simulation Time (sec)
x 10^-5

VUG Net16 M3 Vp-wave Freq = 1MHz
Velocity in y direction (m/s)
x 10^-3
Simulation Time (sec)
x 10^-5

VUG Net17 M3 Vp-wave Freq = 1MHz
Velocity in y direction (m/s)
x 10^-3
Simulation Time (sec)
x 10^-5

VUG Net18 M3 Vp-wave Freq = 1MHz
Velocity in y direction (m/s)
x 10^-3
Simulation Time (sec)
x 10^-5

VUG Net13 M3 Vs1-wave Freq = 1MHz
Velocity in x direction (m/s)
x 10^-3
Simulation Time (sec)
x 10^-5

VUG Net14 M3 Vs1-wave Freq = 1MHz
Velocity in x direction (m/s)
x 10^-3
Simulation Time (sec)
x 10^-5

VUG Net15 M3 Vs1-wave Freq = 1MHz
Velocity in x direction (m/s)
x 10^-4
Simulation Time (sec)
x 10^-5

VUG Net16 M3 Vs1-wave Freq = 1MHz
Velocity in x direction (m/s)
x 10^-3
Simulation Time (sec)
x 10^-5

VUG Net17 M3 Vs1-wave Freq = 1MHz
Velocity in x direction (m/s)
x 10^-4
Simulation Time (sec)
x 10^-5

VUG Net18 M3 Vs1-wave Freq = 1MHz
Velocity in x direction (m/s)
x 10^-3
Simulation Time (sec)
x 10^-5

VUG Net13 M3 Vs2-wave Freq = 1MHz
Velocity in z direction (m/s)
x 10^-3
Simulation Time (sec)
x 10^-5

VUG Net14 M3 Vs2-wave Freq = 1MHz
Velocity in z direction (m/s)
x 10^-4
Simulation Time (sec)
x 10^-5

VUG Net15 M3 Vs2-wave Freq = 1MHz
Velocity in z direction (m/s)
x 10^-4
Simulation Time (sec)
x 10^-5

VUG Net16 M3 Vs2-wave Freq = 1MHz
Velocity in z direction (m/s)
x 10^-3
Simulation Time (sec)
x 10^-5

VUG Net17 M3 Vs2-wave Freq = 1MHz
Velocity in z direction (m/s)
x 10^-4
Simulation Time (sec)
x 10^-5

VUG Net18 M3 Vs2-wave Freq = 1MHz
Velocity in z direction (m/s)
x 10^-4
Simulation Time (sec)
x 10^-5

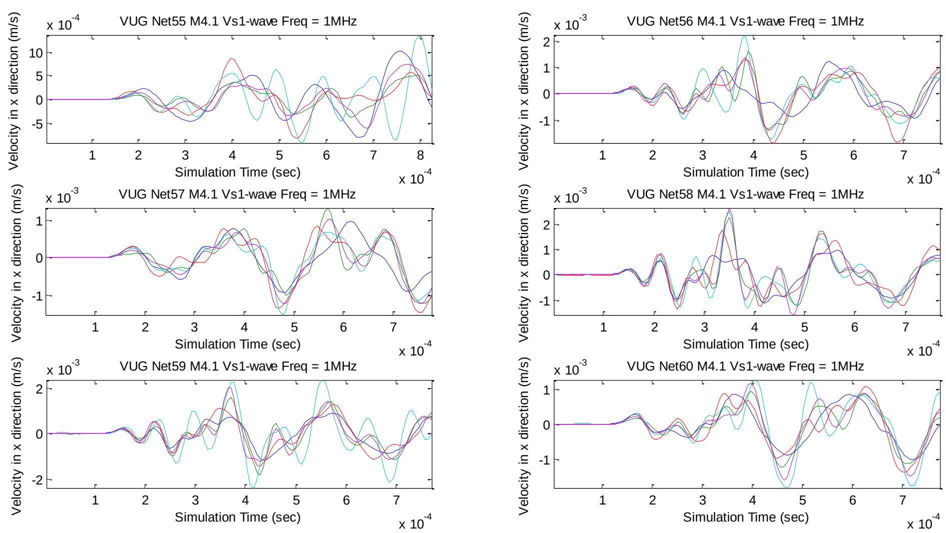

VUG Net55 M3 Vp-wave Freq = 1MHz
VUG Net56 M3 Vp-wave Freq = 1MHz
VUG Net57 M3 Vp-wave Freq = 1MHz
VUG Net58 M3 Vp-wave Freq = 1MHz
VUG Net59 M3 Vp-wave Freq = 1MHz
VUG Net60 M3 Vp-wave Freq = 1MHz
Velocity in y direction (m/s)
Simulation Time (sec)
x 10^-3
x 10^-5

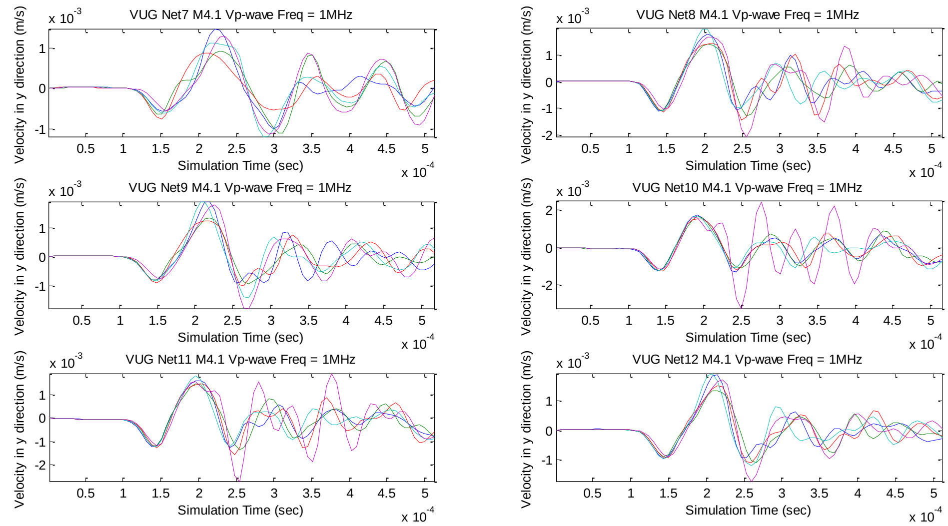

VUG Net55 M3 Vs1-wave Freq = 1MHz
VUG Net56 M3 Vs1-wave Freq = 1MHz
VUG Net57 M3 Vs1-wave Freq = 1MHz
VUG Net58 M3 Vs1-wave Freq = 1MHz
VUG Net59 M3 Vs1-wave Freq = 1MHz
VUG Net60 M3 Vs1-wave Freq = 1MHz
Velocity in x direction (m/s)
Simulation Time (sec)
x 10^-4
x 10^-3
x 10^-5

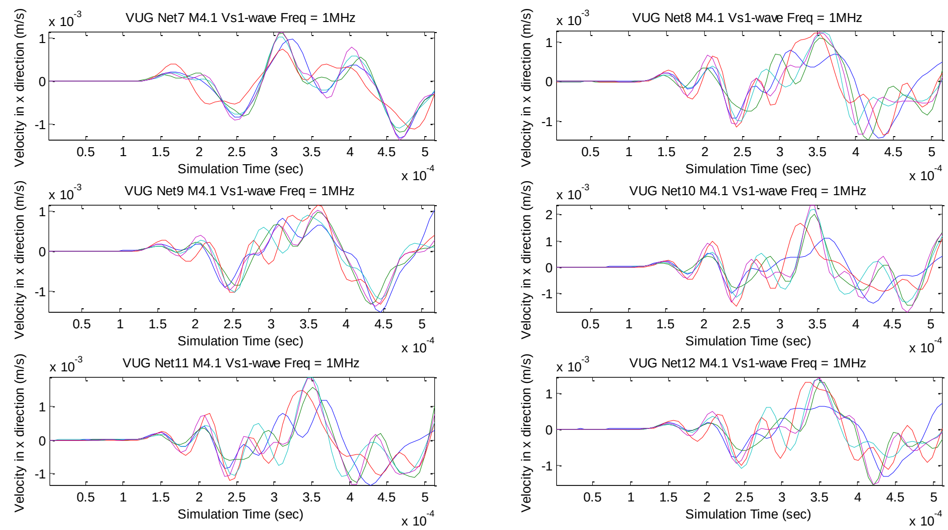

VUG Net55 M3 Vs2-wave Freq = 1MHz
VUG Net56 M3 Vs2-wave Freq = 1MHz
VUG Net57 M3 Vs2-wave Freq = 1MHz
VUG Net58 M3 Vs2-wave Freq = 1MHz
VUG Net59 M3 Vs2-wave Freq = 1MHz
VUG Net60 M3 Vs2-wave Freq = 1MHz
Velocity in z direction (m/s)
Simulation Time (sec)
x 10^-3
x 10^-4
x 10^-5

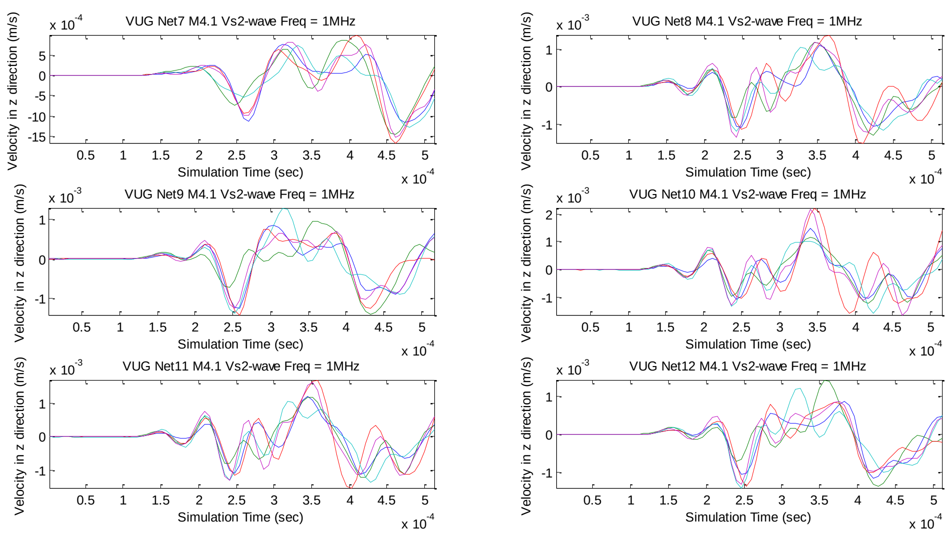

VUG Net1 M4.1 Vp-wave Freq = 1MHz
VUG Net2 M4.1 Vp-wave Freq = 1MHz
VUG Net3 M4.1 Vp-wave Freq = 1MHz
VUG Net4 M4.1 Vp-wave Freq = 1MHz
VUG Net5 M4.1 Vp-wave Freq = 1MHz
VUG Net6 M4.1 Vp-wave Freq = 1MHz
Velocity in y direction (m/s)
x 10^-3
Simulation Time (sec)
x 10^-4

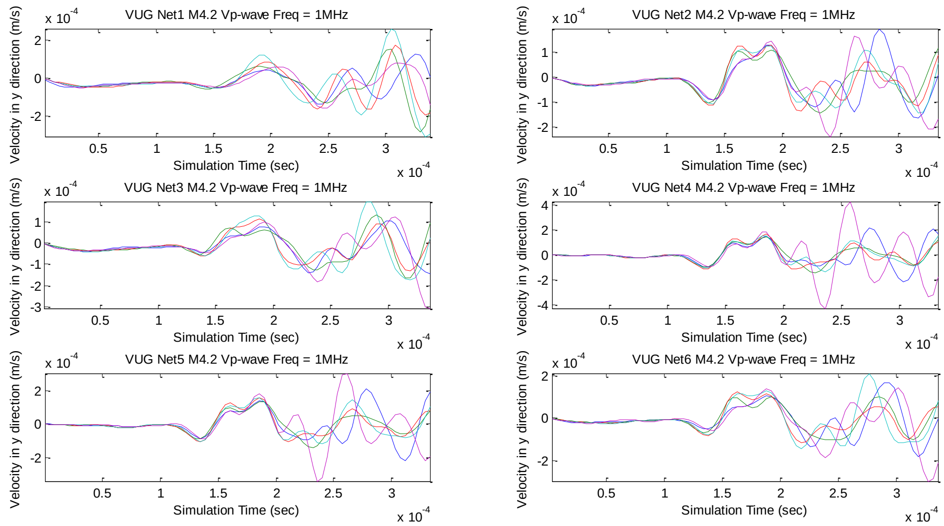

VUG Net1 M4.1 Vs1-wave Freq = 1MHz
VUG Net2 M4.1 Vs1-wave Freq = 1MHz
VUG Net3 M4.1 Vs1-wave Freq = 1MHz
VUG Net4 M4.1 Vs1-wave Freq = 1MHz
VUG Net5 M4.1 Vs1-wave Freq = 1MHz
VUG Net6 M4.1 Vs1-wave Freq = 1MHz
Velocity in x direction (m/s)
x 10^-3
Simulation Time (sec)
x 10^-4

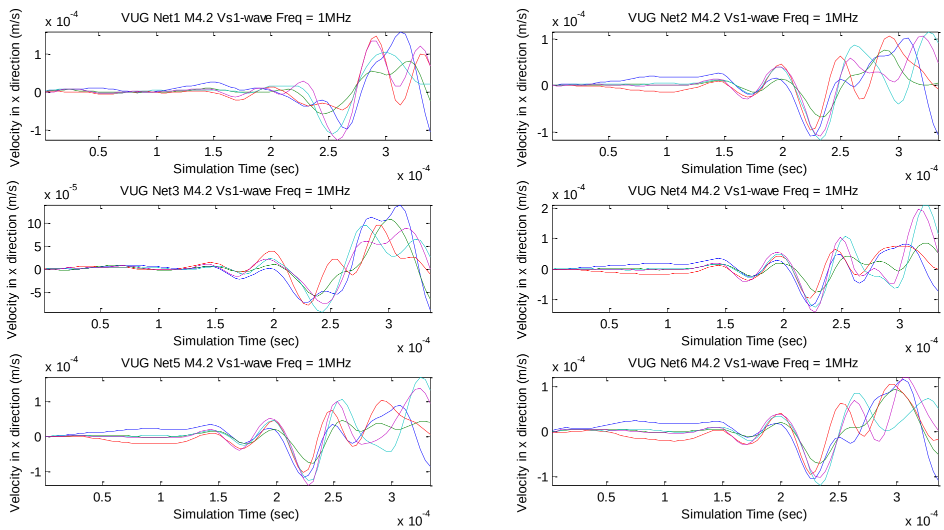

VUG Net55 M4.1 Vs1-wave Freq = 1MHz
VUG Net56 M4.1 Vs1-wave Freq = 1MHz
VUG Net57 M4.1 Vs1-wave Freq = 1MHz
VUG Net58 M4.1 Vs1-wave Freq = 1MHz
VUG Net59 M4.1 Vs1-wave Freq = 1MHz
VUG Net60 M4.1 Vs1-wave Freq = 1MHz
Velocity in x direction (m/s)
x 10^-4
x 10^-3
Simulation Time (sec)
x 10^-4

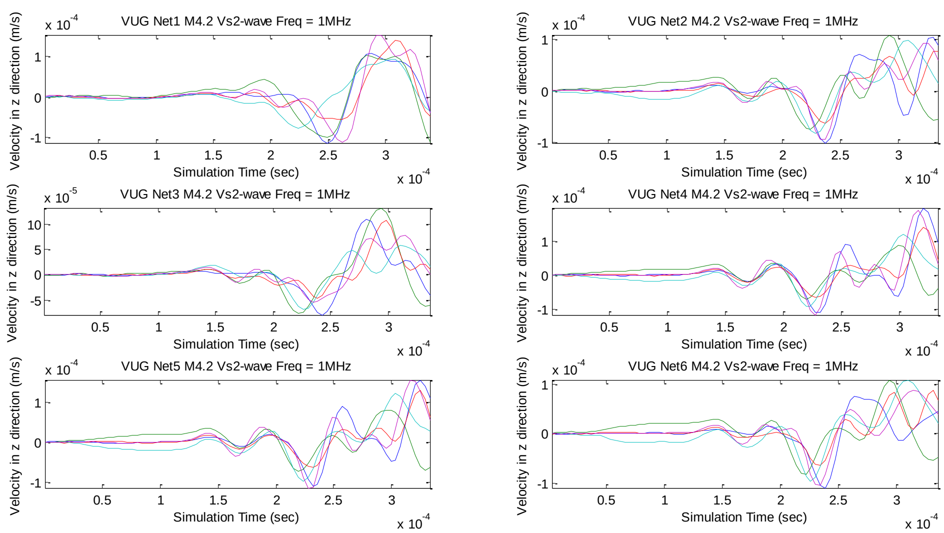

VUG Net7 M4.1 Vp-wave Freq = 1MHz
VUG Net8 M4.1 Vp-wave Freq = 1MHz
VUG Net9 M4.1 Vp-wave Freq = 1MHz
VUG Net10 M4.1 Vp-wave Freq = 1MHz
VUG Net11 M4.1 Vp-wave Freq = 1MHz
VUG Net12 M4.1 Vp-wave Freq = 1MHz
Velocity in y direction (m/s)
Simulation Time (sec)
x 10^-3
x 10^-4

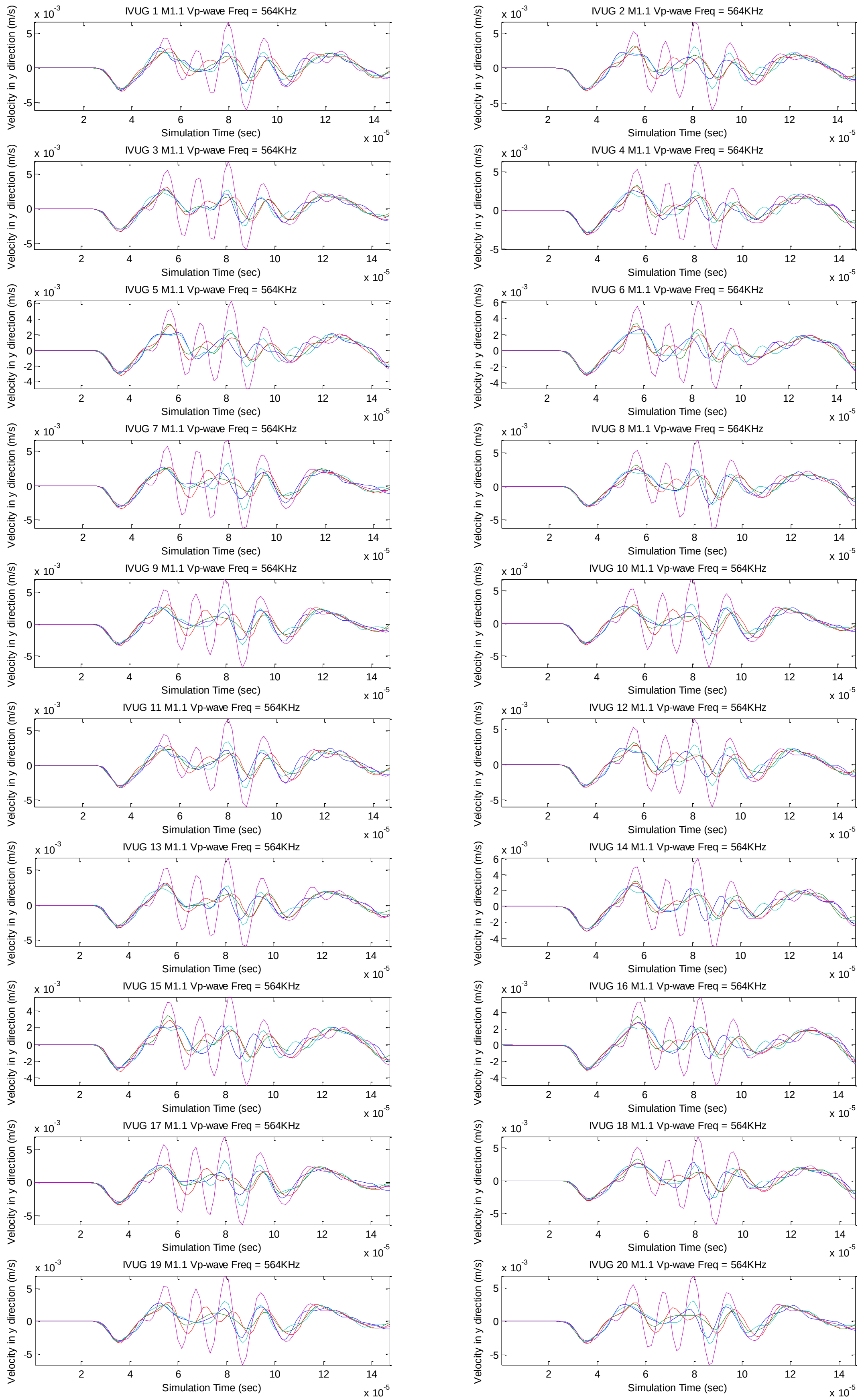

VUG Net7 M4.1 Vs1-wave Freq = 1MHz
VUG Net8 M4.1 Vs1-wave Freq = 1MHz
VUG Net9 M4.1 Vs1-wave Freq = 1MHz
VUG Net10 M4.1 Vs1-wave Freq = 1MHz
VUG Net11 M4.1 Vs1-wave Freq = 1MHz
VUG Net12 M4.1 Vs1-wave Freq = 1MHz
Velocity in x direction (m/s)
Simulation Time (sec)
x 10^-3
x 10^-4

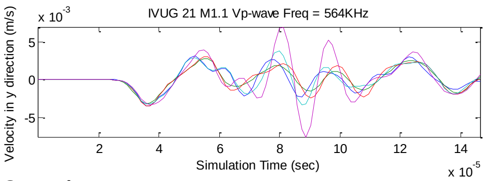

VUG Net7 M4.1 Vs2-wave Freq = 1MHz
VUG Net8 M4.1 Vs2-wave Freq = 1MHz
VUG Net9 M4.1 Vs2-wave Freq = 1MHz
VUG Net10 M4.1 Vs2-wave Freq = 1MHz
VUG Net11 M4.1 Vs2-wave Freq = 1MHz
VUG Net12 M4.1 Vs2-wave Freq = 1MHz
Velocity in z direction (m/s)
Simulation Time (sec)
x 10^-3
x 10^-4

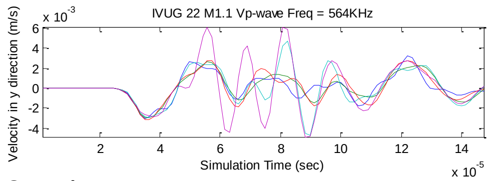

VUG Net1 M4.2 Vp-wave Freq = 1MHz
VUG Net2 M4.2 Vp-wave Freq = 1MHz
VUG Net3 M4.2 Vp-wave Freq = 1MHz
VUG Net4 M4.2 Vp-wave Freq = 1MHz
VUG Net5 M4.2 Vp-wave Freq = 1MHz
VUG Net6 M4.2 Vp-wave Freq = 1MHz
Velocity in y direction (m/s)
Simulation Time (sec)
x 10^-4

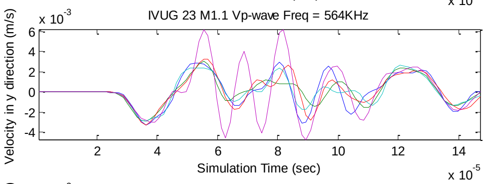

VUG Net1 M4.2 Vs1-wave Freq = 1MHz
VUG Net2 M4.2 Vs1-wave Freq = 1MHz
VUG Net3 M4.2 Vs1-wave Freq = 1MHz
VUG Net4 M4.2 Vs1-wave Freq = 1MHz
VUG Net5 M4.2 Vs1-wave Freq = 1MHz
VUG Net6 M4.2 Vs1-wave Freq = 1MHz
Velocity in x direction (m/s)
Simulation Time (sec)
x 10^-4
x 10^-5

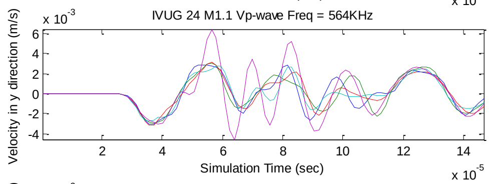

VUG Net1 M4.2 Vs2-wave Freq = 1MHz
VUG Net2 M4.2 Vs2-wave Freq = 1MHz
VUG Net3 M4.2 Vs2-wave Freq = 1MHz
VUG Net4 M4.2 Vs2-wave Freq = 1MHz
VUG Net5 M4.2 Vs2-wave Freq = 1MHz
VUG Net6 M4.2 Vs2-wave Freq = 1MHz
Velocity in z direction (m/s)
Simulation Time (sec)
x 10^-4
x 10^-5

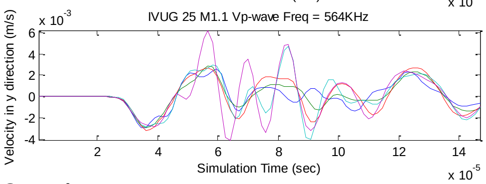

IVUG 1 M1.1 Vp-wave Freq = 564KHz
IVUG 2 M1.1 Vp-wave Freq = 564KHz
IVUG 3 M1.1 Vp-wave Freq = 564KHz
IVUG 4 M1.1 Vp-wave Freq = 564KHz
IVUG 5 M1.1 Vp-wave Freq = 564KHz
IVUG 6 M1.1 Vp-wave Freq = 564KHz
IVUG 7 M1.1 Vp-wave Freq = 564KHz
IVUG 8 M1.1 Vp-wave Freq = 564KHz
IVUG 9 M1.1 Vp-wave Freq = 564KHz
IVUG 10 M1.1 Vp-wave Freq = 564KHz
IVUG 11 M1.1 Vp-wave Freq = 564KHz
IVUG 12 M1.1 Vp-wave Freq = 564KHz
IVUG 13 M1.1 Vp-wave Freq = 564KHz
IVUG 14 M1.1 Vp-wave Freq = 564KHz
IVUG 15 M1.1 Vp-wave Freq = 564KHz
IVUG 16 M1.1 Vp-wave Freq = 564KHz
IVUG 17 M1.1 Vp-wave Freq = 564KHz
IVUG 18 M1.1 Vp-wave Freq = 564KHz
IVUG 19 M1.1 Vp-wave Freq = 564KHz
IVUG 20 M1.1 Vp-wave Freq = 564KHz
Velocity in y direction (m/s)
x 10^-3
Simulation Time (sec)
x 10^-5

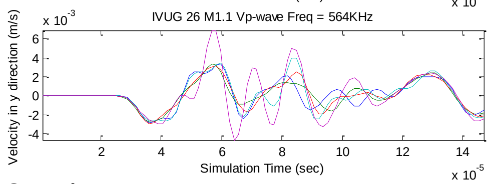

IVUG 21 M1.1 Vp-wave Freq = 564KHz
Velocity in y direction (m/s)
x 10-3
Simulation Time (sec)
x 10-5

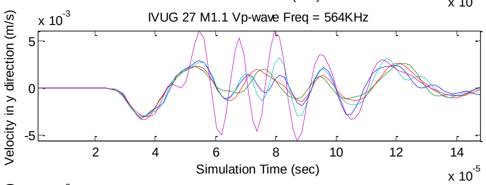

IVUG 22 M1.1 Vp-wave Freq = 564KHz
Velocity in y direction (m/s)
x 10-3
Simulation Time (sec)
x 10-5

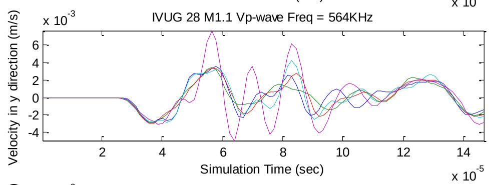

IVUG 23 M1.1 Vp-wave Freq = 564KHz
Velocity in y direction (m/s)
x 10-3
Simulation Time (sec)
x 10-5

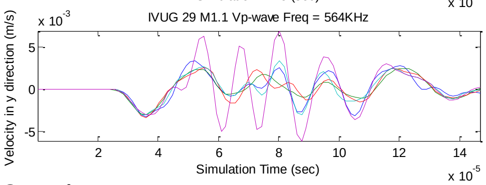

IVUG 24 M1.1 Vp-wave Freq = 564KHz
Velocity in y direction (m/s)
x 10-3
Simulation Time (sec)
x 10-5

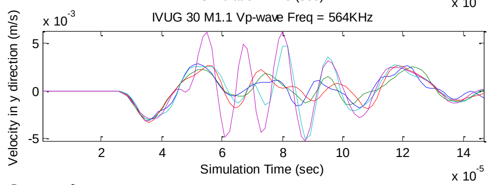

IVUG 25 M1.1 Vp-wave Freq = 564KHz
Velocity in y direction (m/s)
x 10-3
Simulation Time (sec)
x 10-5

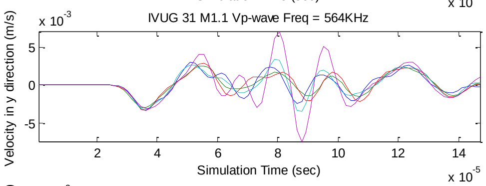

IVUG 26 M1.1 Vp-wave Freq = 564KHz
Velocity in y direction (m/s)
x 10-3
Simulation Time (sec)
x 10-5

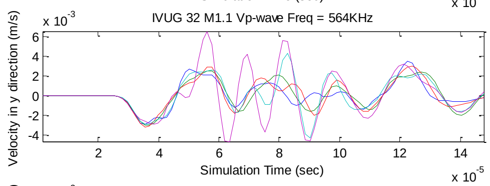

IVUG 27 M1.1 Vp-wave Freq = 564KHz
Velocity in y direction (m/s)
x 10-3
Simulation Time (sec)
x 10-5

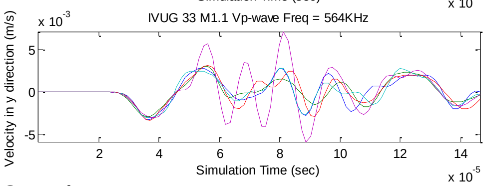

IVUG 28 M1.1 Vp-wave Freq = 564KHz
Velocity in y direction (m/s)
x 10-3
Simulation Time (sec)
x 10-5

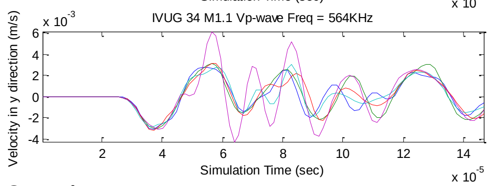

IVUG 29 M1.1 Vp-wave Freq = 564KHz
Velocity in y direction (m/s)
x 10-3
Simulation Time (sec)
x 10-5

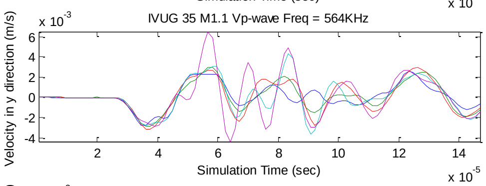

IVUG 30 M1.1 Vp-wave Freq = 564KHz
Velocity in y direction (m/s)
x 10-3
Simulation Time (sec)
x 10-5

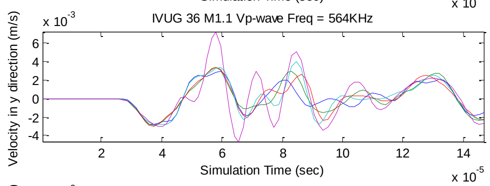

IVUG 31 M1.1 Vp-wave Freq = 564KHz
Velocity in y direction (m/s)
x 10-3
Simulation Time (sec)
x 10-5

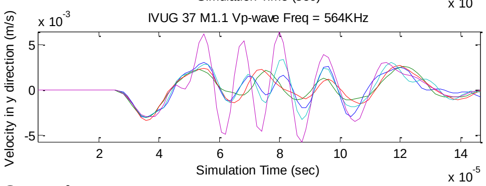

IVUG 32 M1.1 Vp-wave Freq = 564KHz
Velocity in y direction (m/s)
x 10-3
Simulation Time (sec)
x 10-5

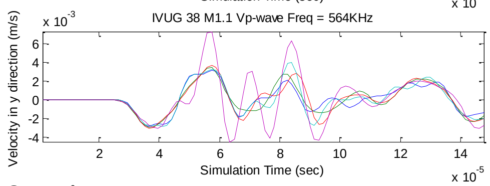

IVUG 33 M1.1 Vp-wave Freq = 564KHz
Velocity in y direction (m/s)
x 10-3
Simulation Time (sec)
x 10-5

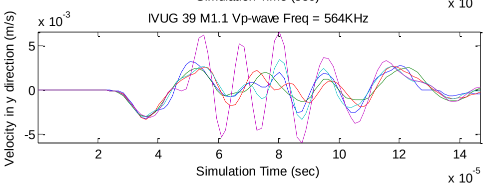

IVUG 34 M1.1 Vp-wave Freq = 564KHz
Velocity in y direction (m/s)
x 10-3
Simulation Time (sec)
x 10-5

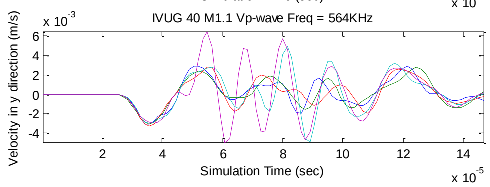

IVUG 35 M1.1 Vp-wave Freq = 564KHz
Velocity in y direction (m/s)
x 10-3
Simulation Time (sec)
x 10-5

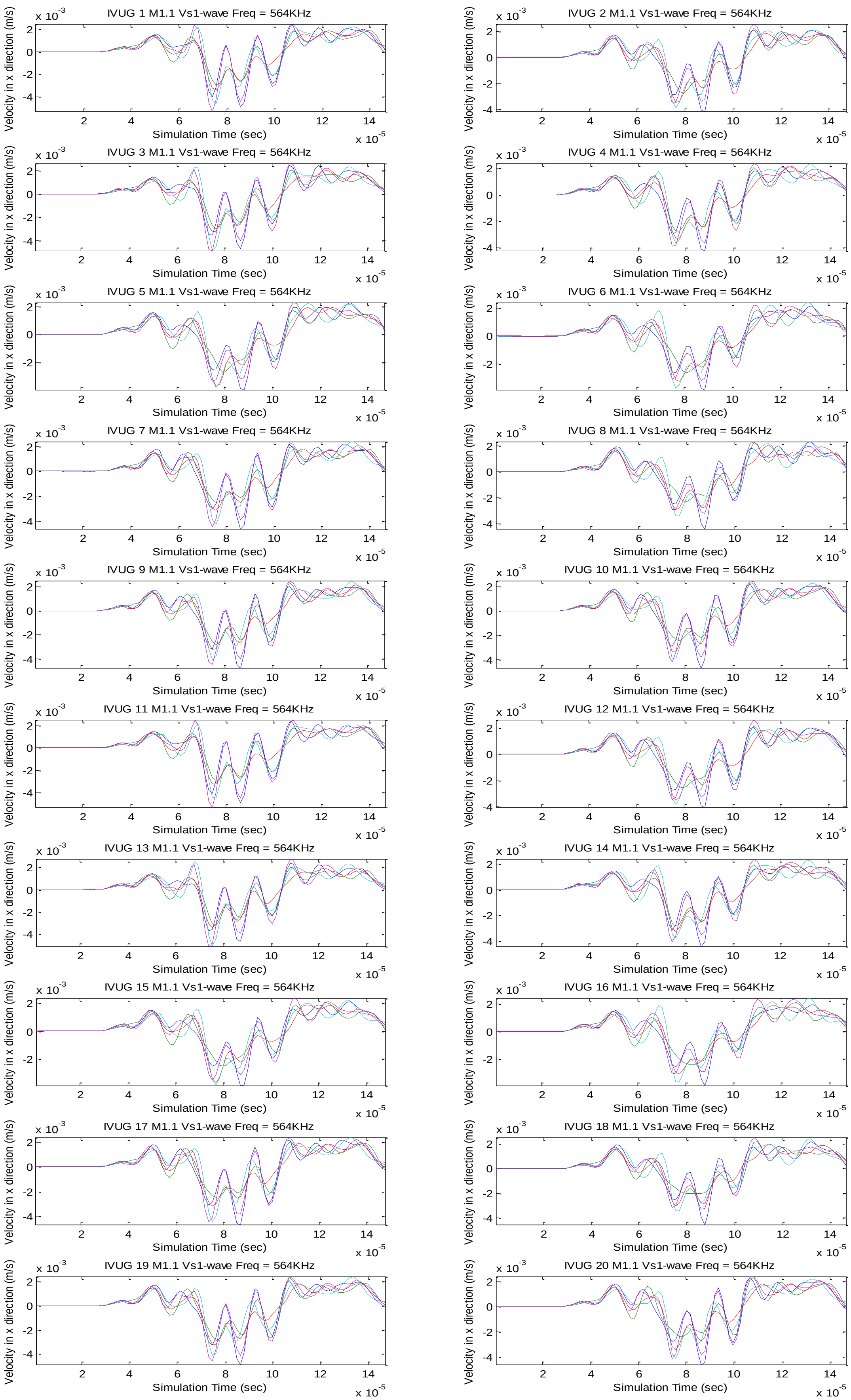

IVUG 36 M1.1 Vp-wave Freq = 564KHz
Velocity in y direction (m/s)
x 10-3
Simulation Time (sec)
x 10-5

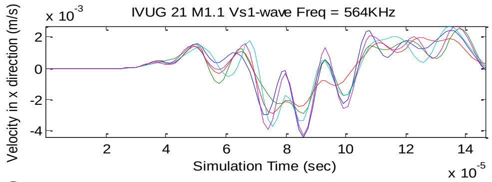

IVUG 37 M1.1 Vp-wave Freq = 564KHz
Velocity in y direction (m/s)
x 10-3
Simulation Time (sec)
x 10-5

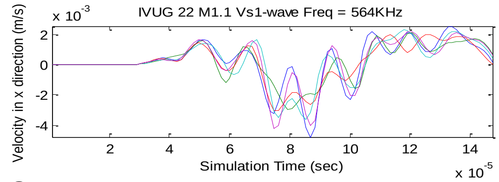

IVUG 38 M1.1 Vp-wave Freq = 564KHz
Velocity in y direction (m/s)
x 10-3
Simulation Time (sec)
x 10-5

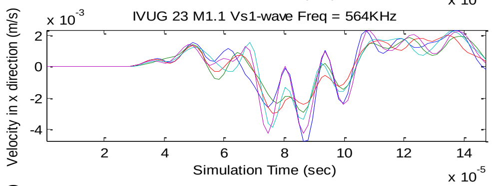

IVUG 39 M1.1 Vp-wave Freq = 564KHz
Velocity in y direction (m/s)
x 10-3
Simulation Time (sec)
x 10-5

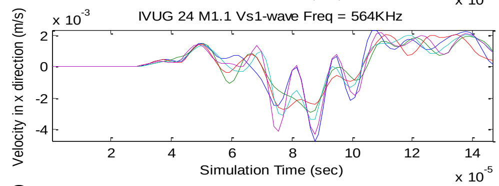

IVUG 40 M1.1 Vp-wave Freq = 564KHz
Velocity in y direction (m/s)
x 10-3
Simulation Time (sec)
x 10-5

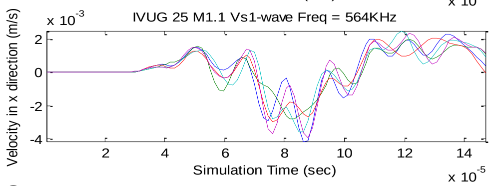

IVUG 1 M1.1 Vs1-wave Freq = 564KHz
IVUG 2 M1.1 Vs1-wave Freq = 564KHz
IVUG 3 M1.1 Vs1-wave Freq = 564KHz
IVUG 4 M1.1 Vs1-wave Freq = 564KHz
IVUG 5 M1.1 Vs1-wave Freq = 564KHz
IVUG 6 M1.1 Vs1-wave Freq = 564KHz
IVUG 7 M1.1 Vs1-wave Freq = 564KHz
IVUG 8 M1.1 Vs1-wave Freq = 564KHz
IVUG 9 M1.1 Vs1-wave Freq = 564KHz
IVUG 10 M1.1 Vs1-wave Freq = 564KHz
IVUG 11 M1.1 Vs1-wave Freq = 564KHz
IVUG 12 M1.1 Vs1-wave Freq = 564KHz
IVUG 13 M1.1 Vs1-wave Freq = 564KHz
IVUG 14 M1.1 Vs1-wave Freq = 564KHz
IVUG 15 M1.1 Vs1-wave Freq = 564KHz
IVUG 16 M1.1 Vs1-wave Freq = 564KHz
IVUG 17 M1.1 Vs1-wave Freq = 564KHz
IVUG 18 M1.1 Vs1-wave Freq = 564KHz
IVUG 19 M1.1 Vs1-wave Freq = 564KHz
IVUG 20 M1.1 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
x 10-3
Simulation Time (sec)
x 10-5

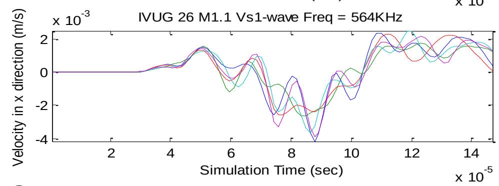

IVUG 21 M1.1 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
x 10^-3
Simulation Time (sec)
x 10^-5

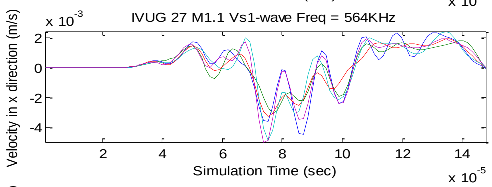

IVUG 22 M1.1 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
x 10^-3
Simulation Time (sec)
x 10^-5

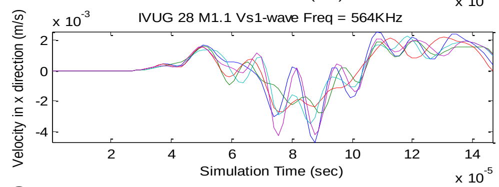

IVUG 23 M1.1 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
x 10^-3
Simulation Time (sec)
x 10^-5

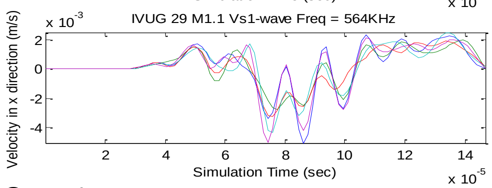

IVUG 24 M1.1 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
x 10^-3
Simulation Time (sec)
x 10^-5

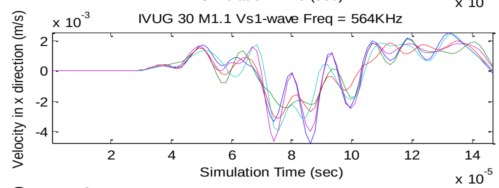

IVUG 25 M1.1 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
x 10^-3
Simulation Time (sec)
x 10^-5

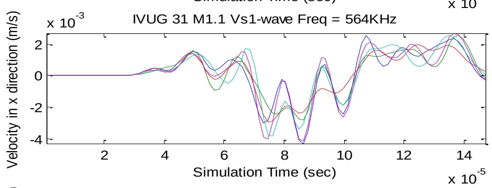

IVUG 26 M1.1 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
x 10^-3
Simulation Time (sec)
x 10^-5

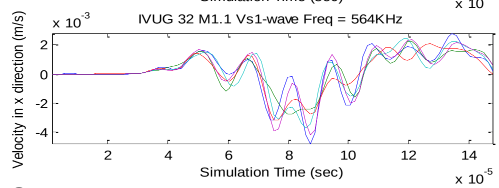

IVUG 27 M1.1 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
x 10^-3
Simulation Time (sec)
x 10^-5

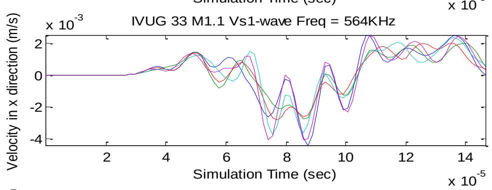

IVUG 28 M1.1 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
x 10^-3
Simulation Time (sec)
x 10^-5

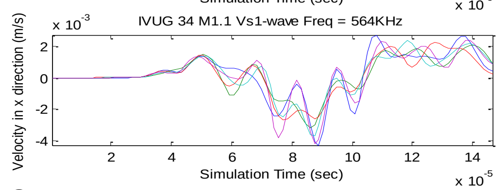

IVUG 29 M1.1 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
x 10^-3
Simulation Time (sec)
x 10^-5

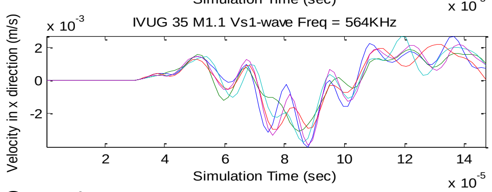

IVUG 30 M1.1 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
x 10^-3
Simulation Time (sec)
x 10^-5

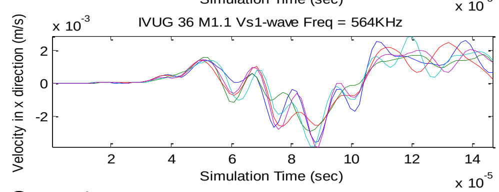

IVUG 31 M1.1 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
x 10^-3
Simulation Time (sec)
x 10^-5

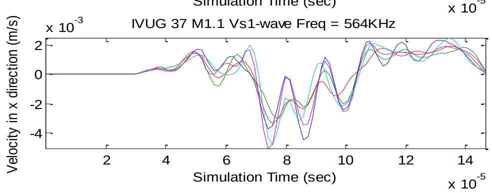

IVUG 32 M1.1 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
x 10^-3
Simulation Time (sec)
x 10^-5

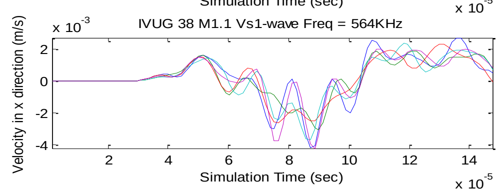

IVUG 33 M1.1 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
x 10^-3
Simulation Time (sec)
x 10^-5

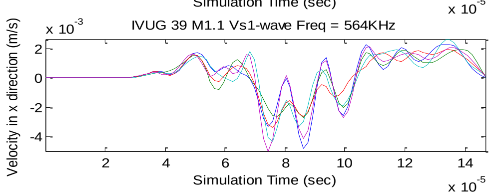

IVUG 34 M1.1 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
x 10^-3
Simulation Time (sec)
x 10^-5

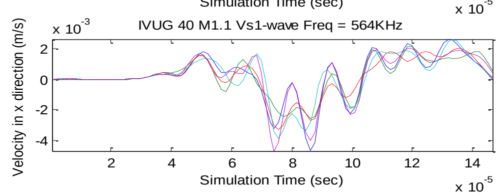

IVUG 35 M1.1 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
x 10^-3
Simulation Time (sec)
x 10^-5

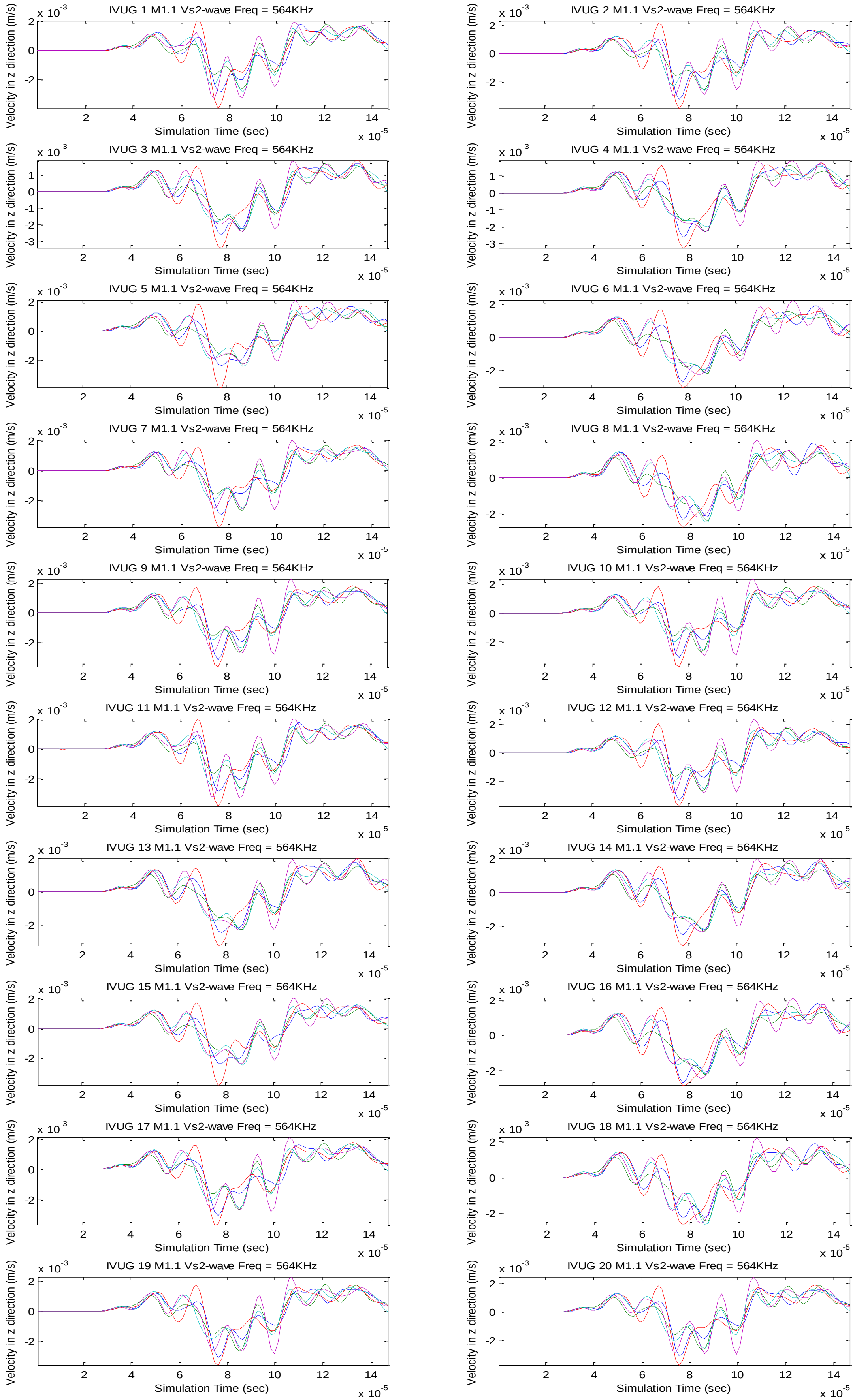

IVUG 36 M1.1 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
x 10^-3
Simulation Time (sec)
x 10^-5

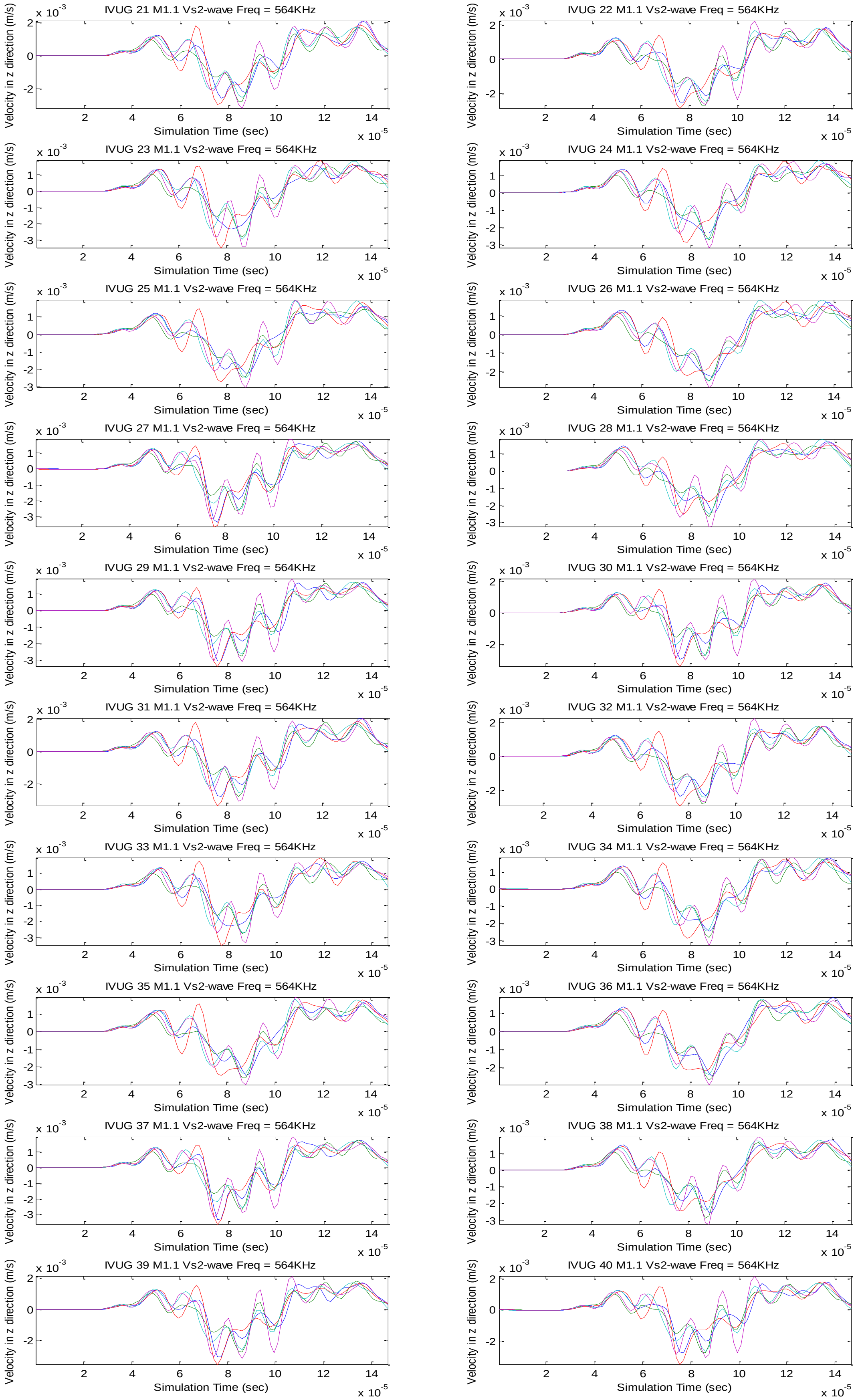

IVUG 37 M1.1 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
x 10^-3
Simulation Time (sec)
x 10^-5

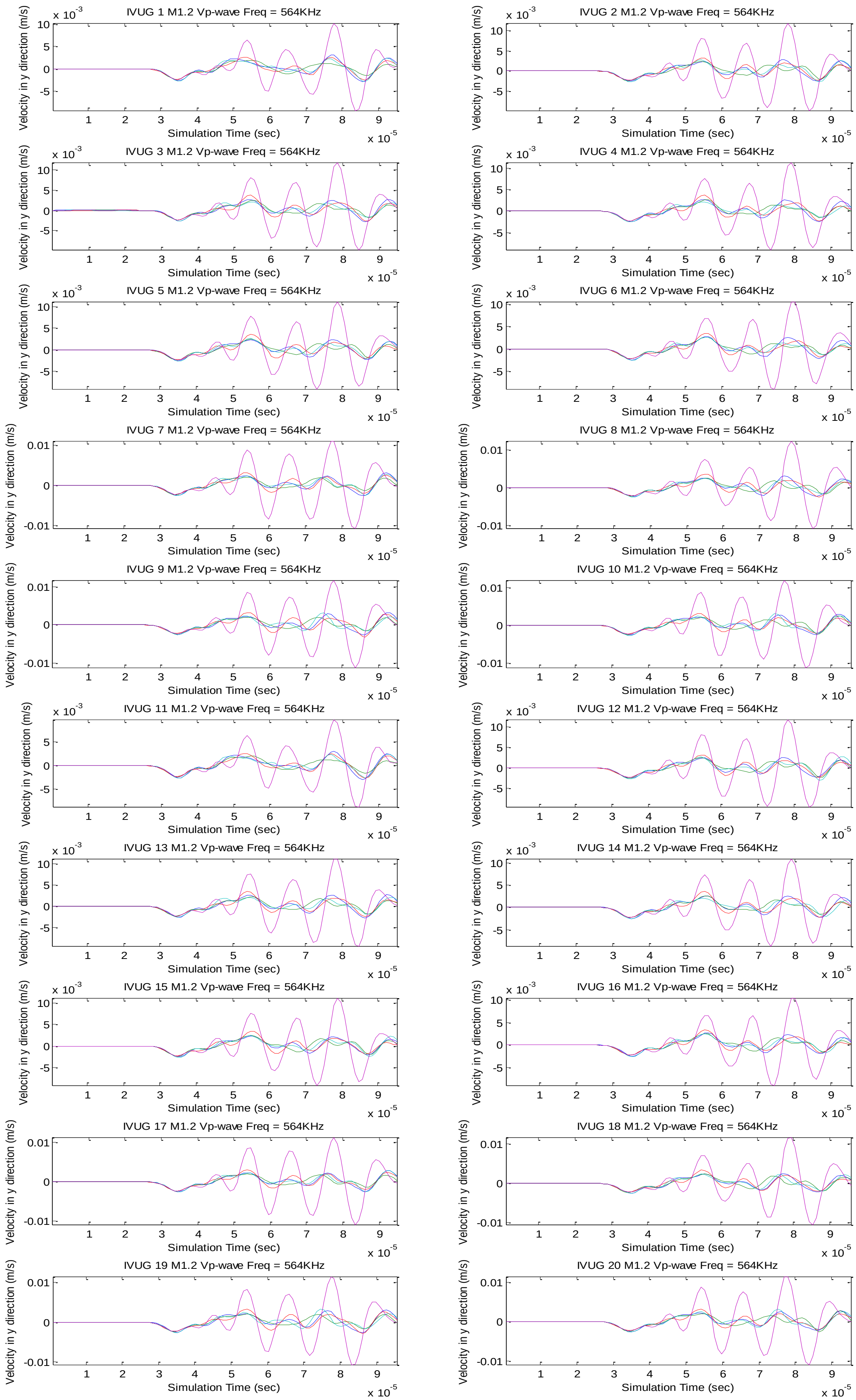

IVUG 38 M1.1 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
x 10^-3
Simulation Time (sec)
x 10^-5

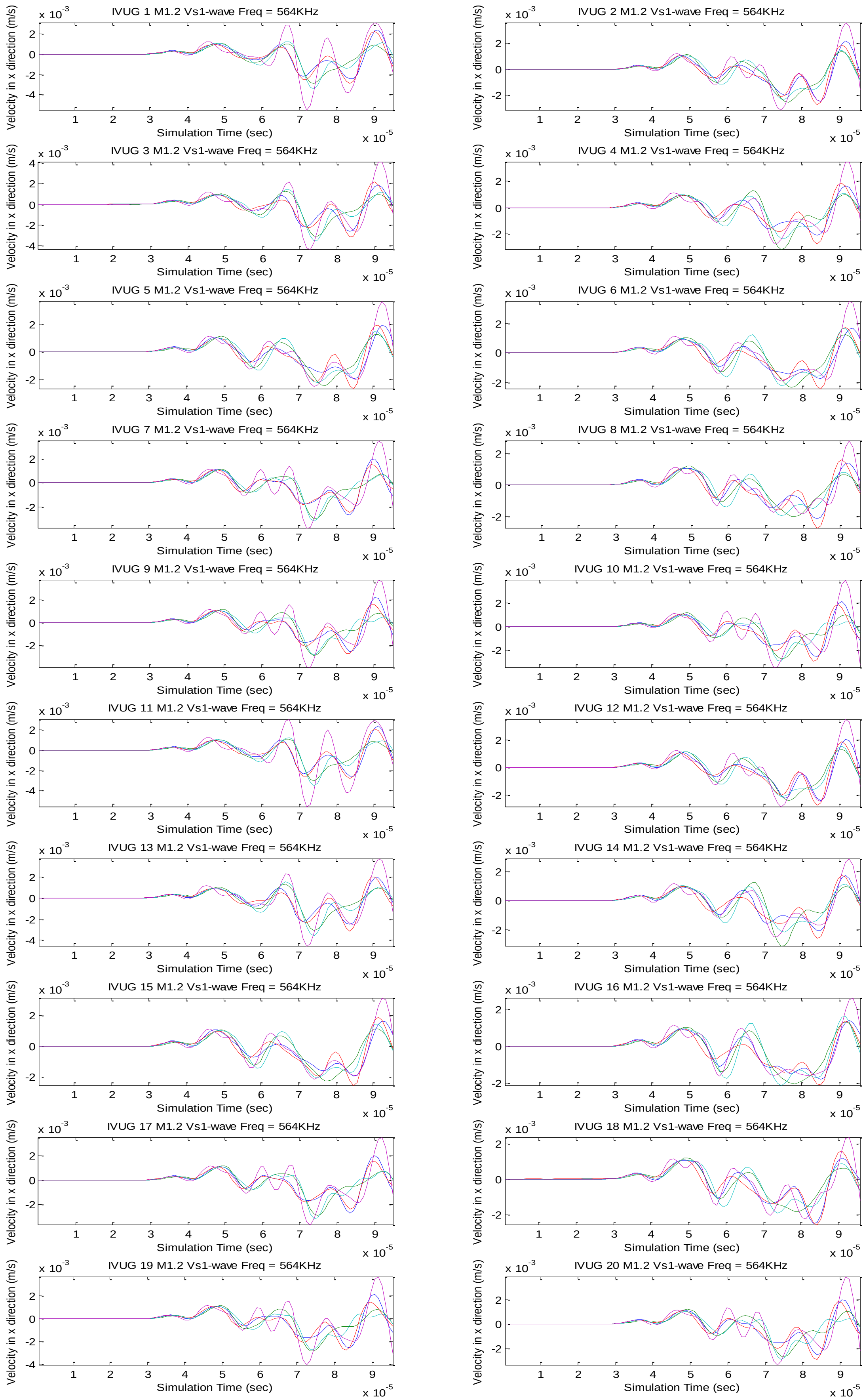

IVUG 39 M1.1 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
x 10^-3
Simulation Time (sec)
x 10^-5

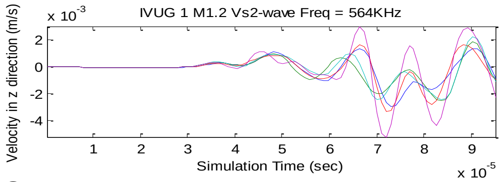
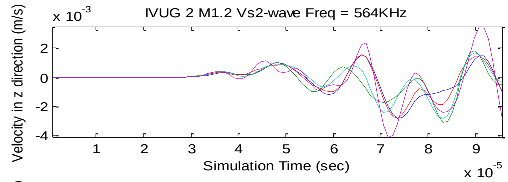
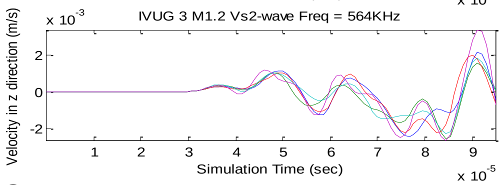
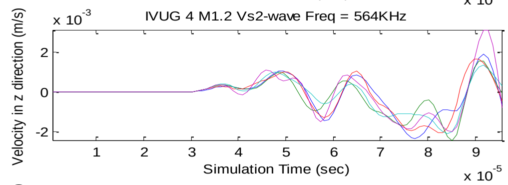
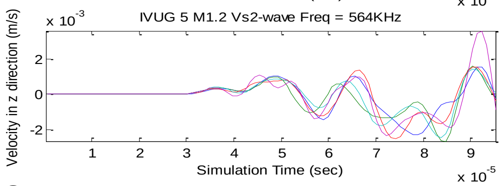
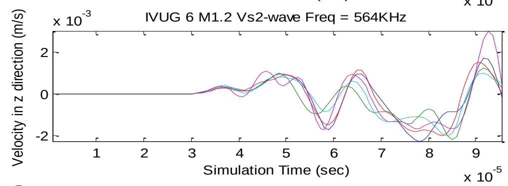
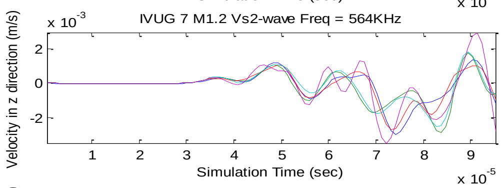
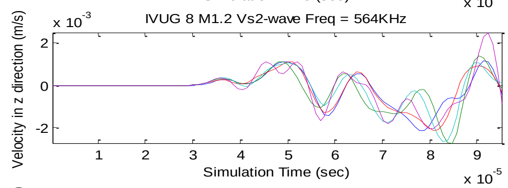
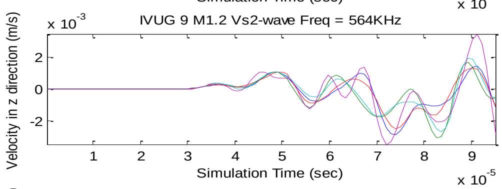
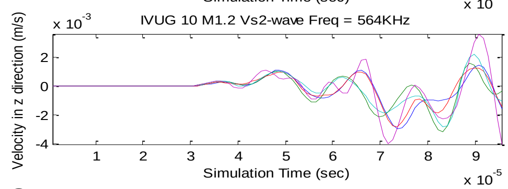
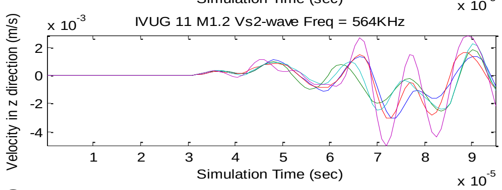
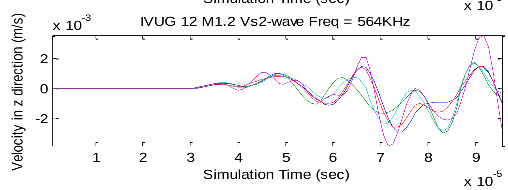
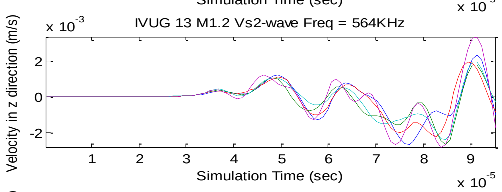
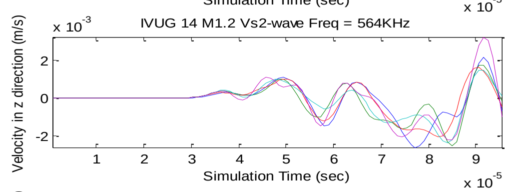
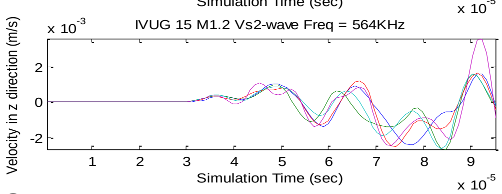
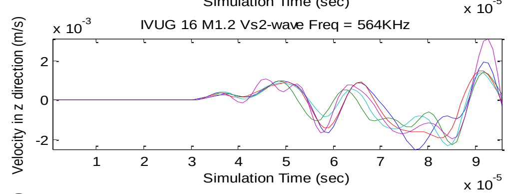
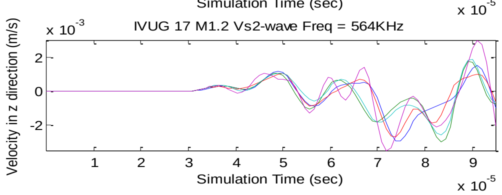
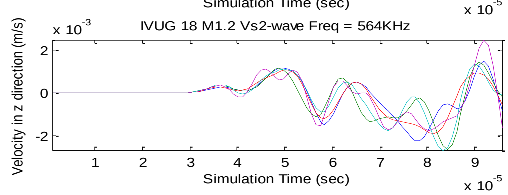
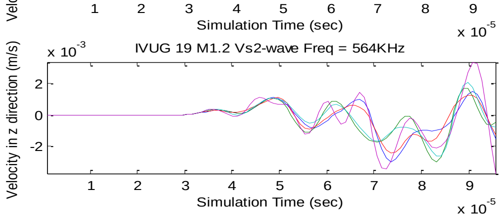
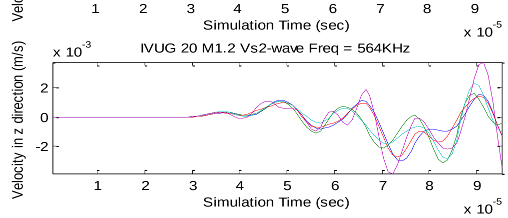

IVUG 40 M1.1 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
x 10^-3
Simulation Time (sec)
x 10^-5

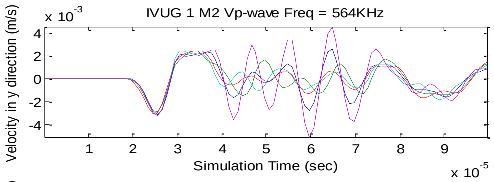
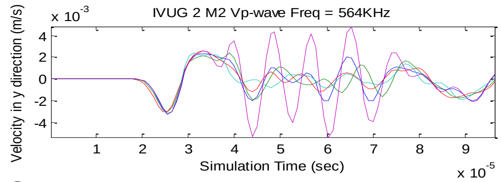
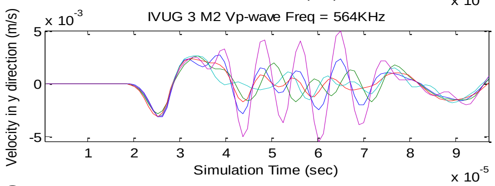
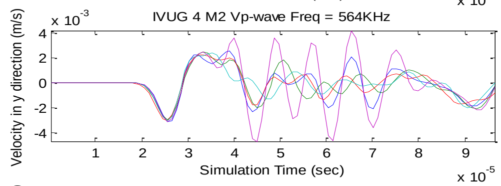
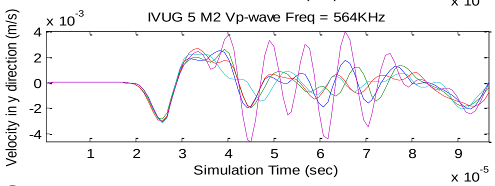
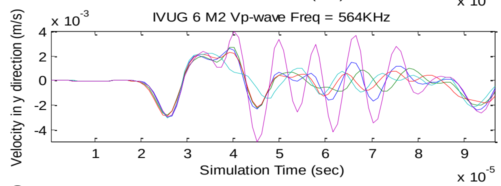
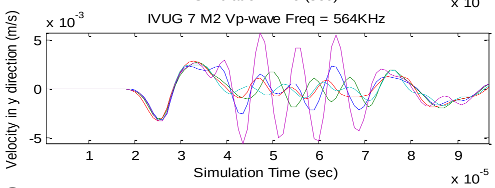
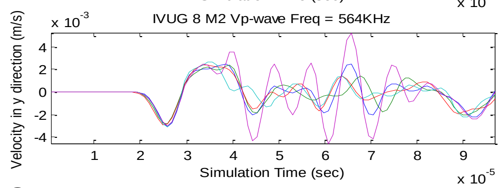
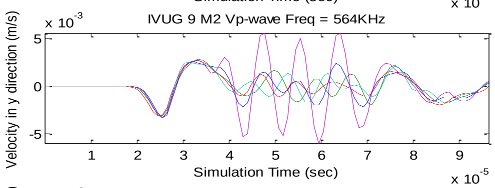
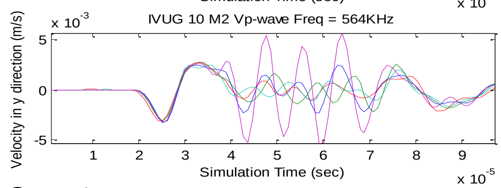
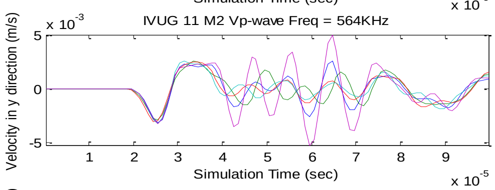
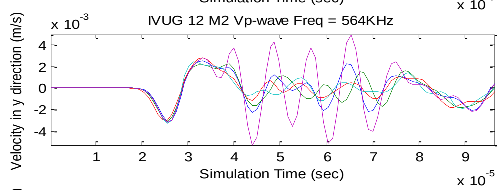
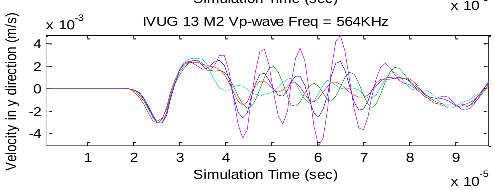
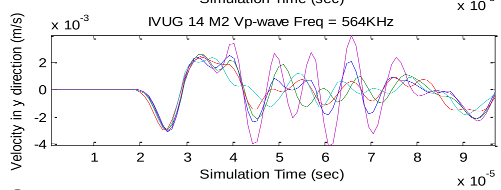
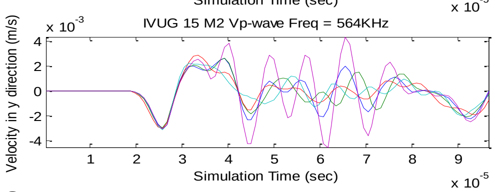
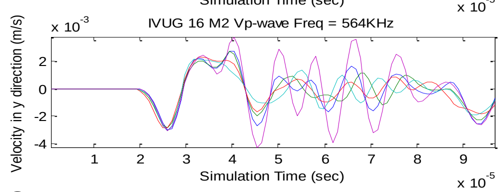
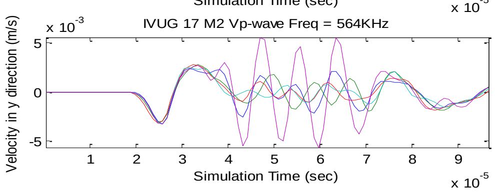
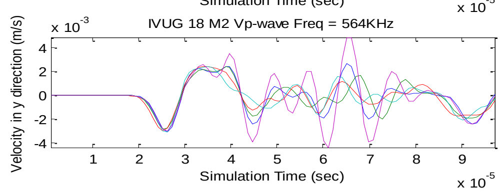
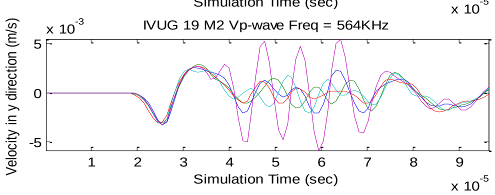
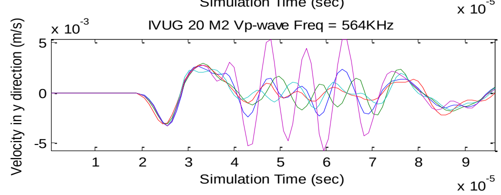

IVUG 1 M1.1 Vs2-wave Freq = 564KHz
IVUG 2 M1.1 Vs2-wave Freq = 564KHz
IVUG 3 M1.1 Vs2-wave Freq = 564KHz
IVUG 4 M1.1 Vs2-wave Freq = 564KHz
IVUG 5 M1.1 Vs2-wave Freq = 564KHz
IVUG 6 M1.1 Vs2-wave Freq = 564KHz
IVUG 7 M1.1 Vs2-wave Freq = 564KHz
IVUG 8 M1.1 Vs2-wave Freq = 564KHz
IVUG 9 M1.1 Vs2-wave Freq = 564KHz
IVUG 10 M1.1 Vs2-wave Freq = 564KHz
IVUG 11 M1.1 Vs2-wave Freq = 564KHz
IVUG 12 M1.1 Vs2-wave Freq = 564KHz
IVUG 13 M1.1 Vs2-wave Freq = 564KHz
IVUG 14 M1.1 Vs2-wave Freq = 564KHz
IVUG 15 M1.1 Vs2-wave Freq = 564KHz
IVUG 16 M1.1 Vs2-wave Freq = 564KHz
IVUG 17 M1.1 Vs2-wave Freq = 564KHz
IVUG 18 M1.1 Vs2-wave Freq = 564KHz
IVUG 19 M1.1 Vs2-wave Freq = 564KHz
IVUG 20 M1.1 Vs2-wave Freq = 564KHz
Velocity in z direction (m/s)
x 10^-3
Simulation Time (sec)
x 10^-5

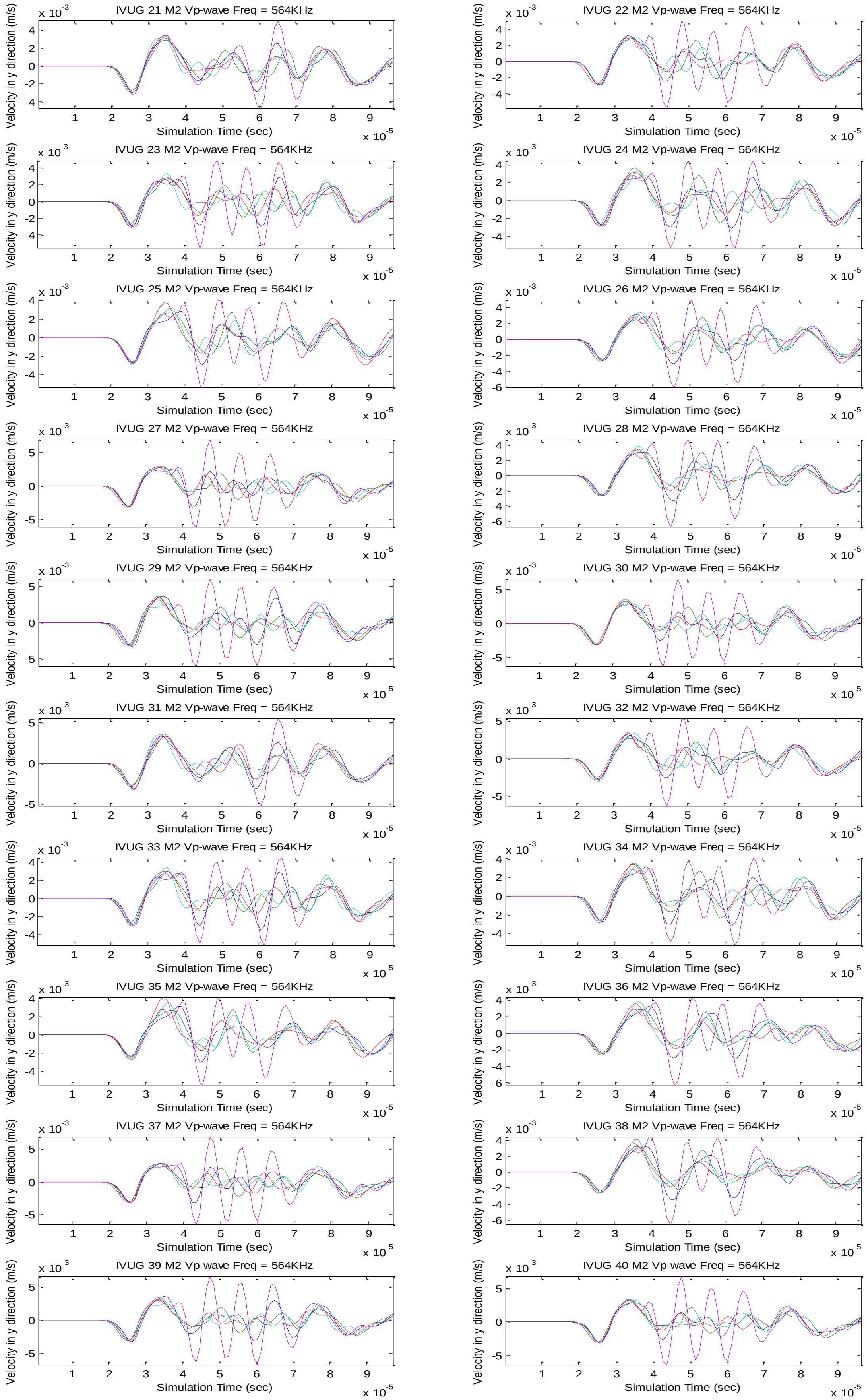

IVUG 21 M1.1 Vs2-wave Freq = 564KHz
IVUG 22 M1.1 Vs2-wave Freq = 564KHz
IVUG 23 M1.1 Vs2-wave Freq = 564KHz
IVUG 24 M1.1 Vs2-wave Freq = 564KHz
IVUG 25 M1.1 Vs2-wave Freq = 564KHz
IVUG 26 M1.1 Vs2-wave Freq = 564KHz
IVUG 27 M1.1 Vs2-wave Freq = 564KHz
IVUG 28 M1.1 Vs2-wave Freq = 564KHz
IVUG 29 M1.1 Vs2-wave Freq = 564KHz
IVUG 30 M1.1 Vs2-wave Freq = 564KHz
IVUG 31 M1.1 Vs2-wave Freq = 564KHz
IVUG 32 M1.1 Vs2-wave Freq = 564KHz
IVUG 33 M1.1 Vs2-wave Freq = 564KHz
IVUG 34 M1.1 Vs2-wave Freq = 564KHz
IVUG 35 M1.1 Vs2-wave Freq = 564KHz
IVUG 36 M1.1 Vs2-wave Freq = 564KHz
IVUG 37 M1.1 Vs2-wave Freq = 564KHz
IVUG 38 M1.1 Vs2-wave Freq = 564KHz
IVUG 39 M1.1 Vs2-wave Freq = 564KHz
IVUG 40 M1.1 Vs2-wave Freq = 564KHz
Velocity in z direction (m/s)
x 10^-3
Simulation Time (sec)
x 10^-5

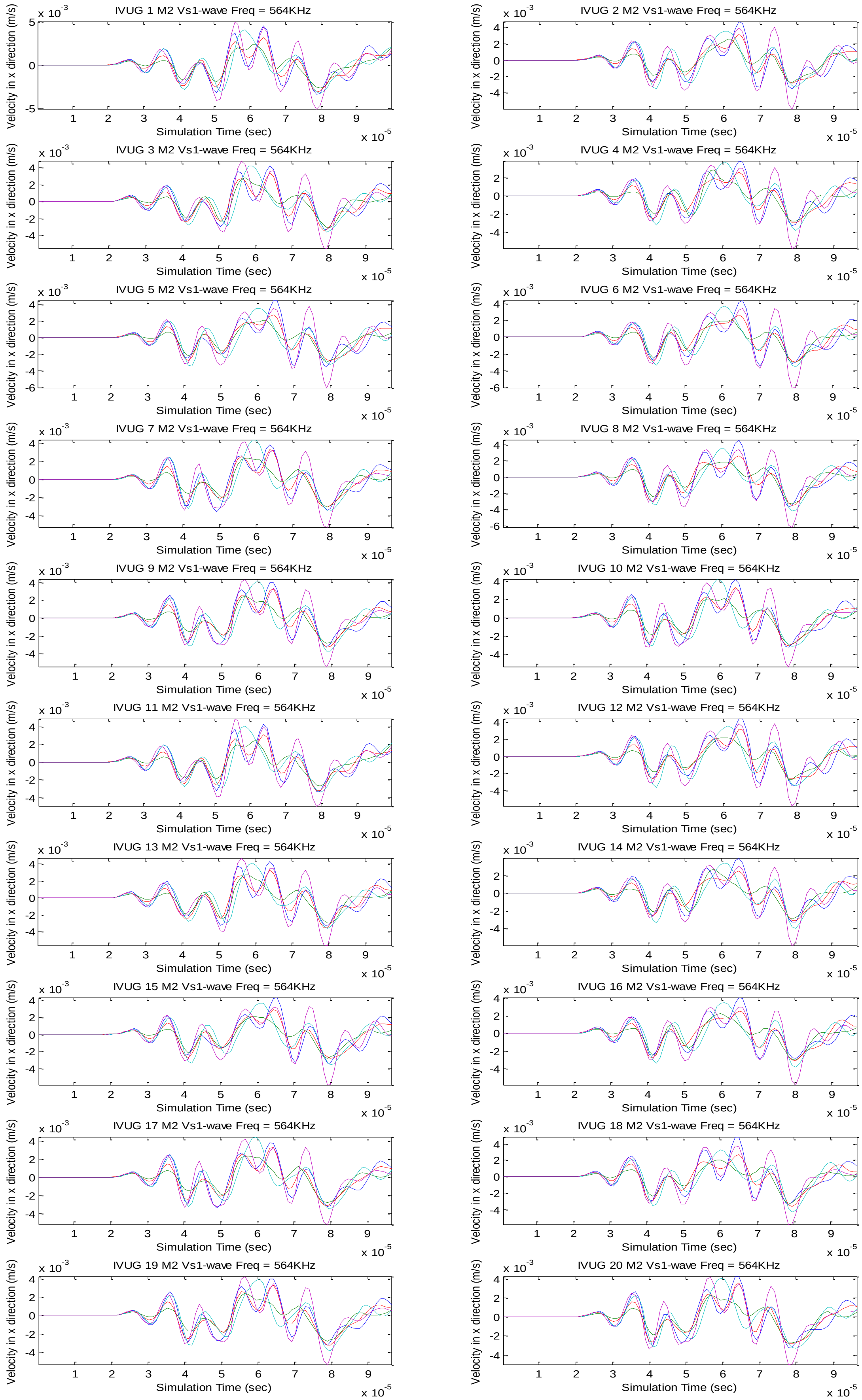

IVUG 1 M1.2 Vp-wave Freq = 564KHz
IVUG 2 M1.2 Vp-wave Freq = 564KHz
IVUG 3 M1.2 Vp-wave Freq = 564KHz
IVUG 4 M1.2 Vp-wave Freq = 564KHz
IVUG 5 M1.2 Vp-wave Freq = 564KHz
IVUG 6 M1.2 Vp-wave Freq = 564KHz
IVUG 7 M1.2 Vp-wave Freq = 564KHz
IVUG 8 M1.2 Vp-wave Freq = 564KHz
IVUG 9 M1.2 Vp-wave Freq = 564KHz
IVUG 10 M1.2 Vp-wave Freq = 564KHz
IVUG 11 M1.2 Vp-wave Freq = 564KHz
IVUG 12 M1.2 Vp-wave Freq = 564KHz
IVUG 13 M1.2 Vp-wave Freq = 564KHz
IVUG 14 M1.2 Vp-wave Freq = 564KHz
IVUG 15 M1.2 Vp-wave Freq = 564KHz
IVUG 16 M1.2 Vp-wave Freq = 564KHz
IVUG 17 M1.2 Vp-wave Freq = 564KHz
IVUG 18 M1.2 Vp-wave Freq = 564KHz
IVUG 19 M1.2 Vp-wave Freq = 564KHz
IVUG 20 M1.2 Vp-wave Freq = 564KHz
Velocity in y direction (m/s)
Simulation Time (sec)
x 10^-3
x 10^-5

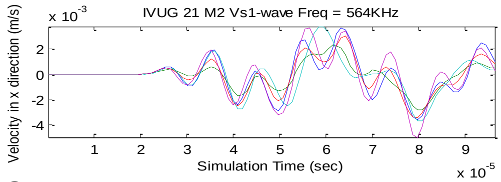
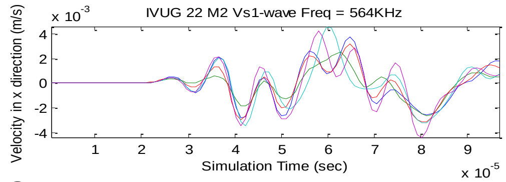
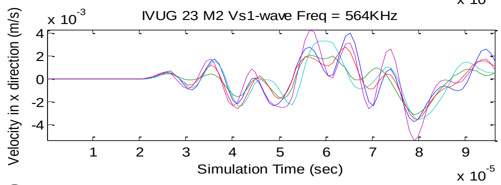
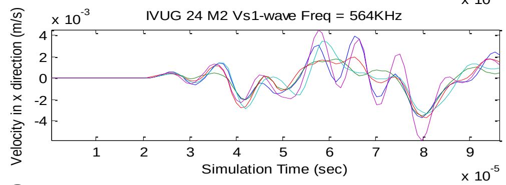
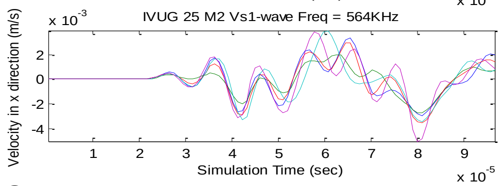
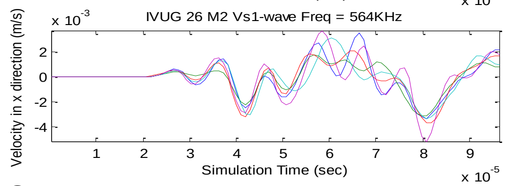
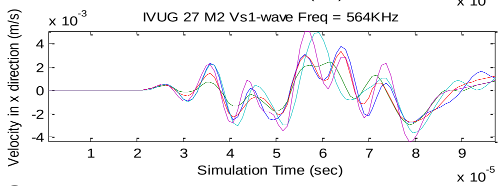
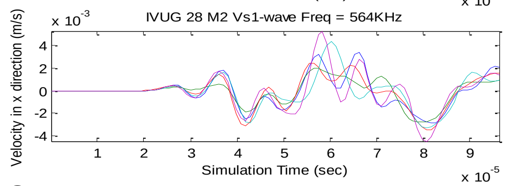
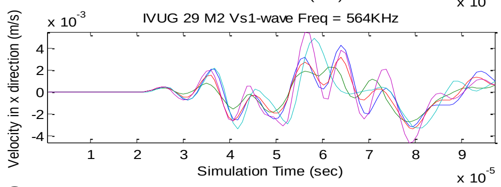
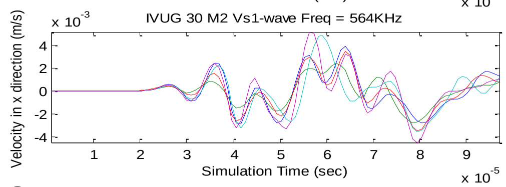
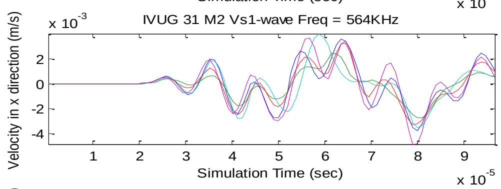
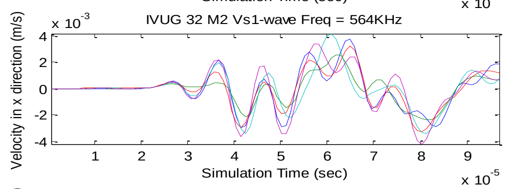
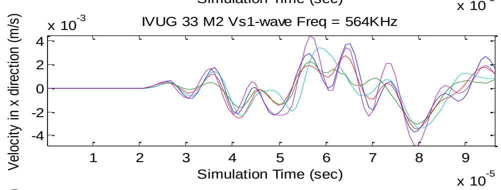
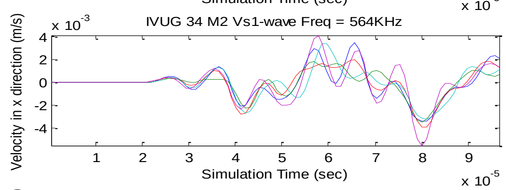
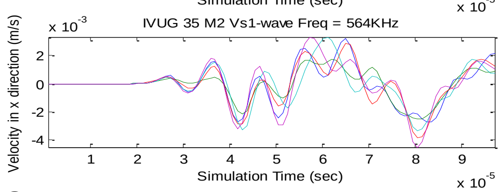
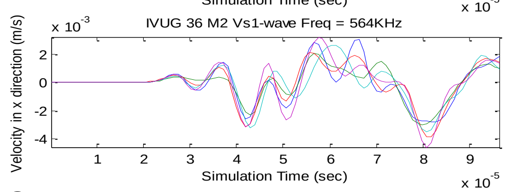
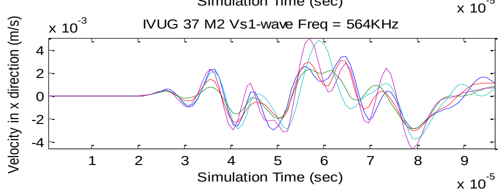
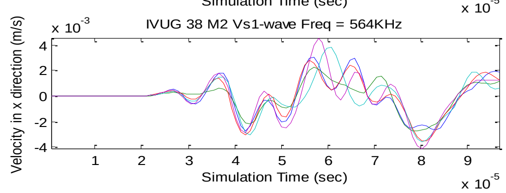
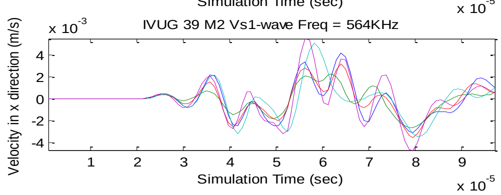
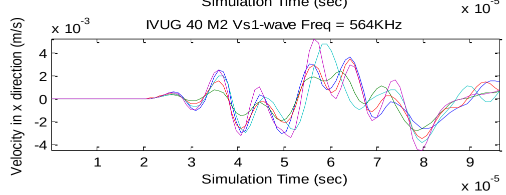

IVUG 1 M1.2 Vs1-wave Freq = 564KHz
IVUG 2 M1.2 Vs1-wave Freq = 564KHz
IVUG 3 M1.2 Vs1-wave Freq = 564KHz
IVUG 4 M1.2 Vs1-wave Freq = 564KHz
IVUG 5 M1.2 Vs1-wave Freq = 564KHz
IVUG 6 M1.2 Vs1-wave Freq = 564KHz
IVUG 7 M1.2 Vs1-wave Freq = 564KHz
IVUG 8 M1.2 Vs1-wave Freq = 564KHz
IVUG 9 M1.2 Vs1-wave Freq = 564KHz
IVUG 10 M1.2 Vs1-wave Freq = 564KHz
IVUG 11 M1.2 Vs1-wave Freq = 564KHz
IVUG 12 M1.2 Vs1-wave Freq = 564KHz
IVUG 13 M1.2 Vs1-wave Freq = 564KHz
IVUG 14 M1.2 Vs1-wave Freq = 564KHz
IVUG 15 M1.2 Vs1-wave Freq = 564KHz
IVUG 16 M1.2 Vs1-wave Freq = 564KHz
IVUG 17 M1.2 Vs1-wave Freq = 564KHz
IVUG 18 M1.2 Vs1-wave Freq = 564KHz
IVUG 19 M1.2 Vs1-wave Freq = 564KHz
IVUG 20 M1.2 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
Simulation Time (sec)

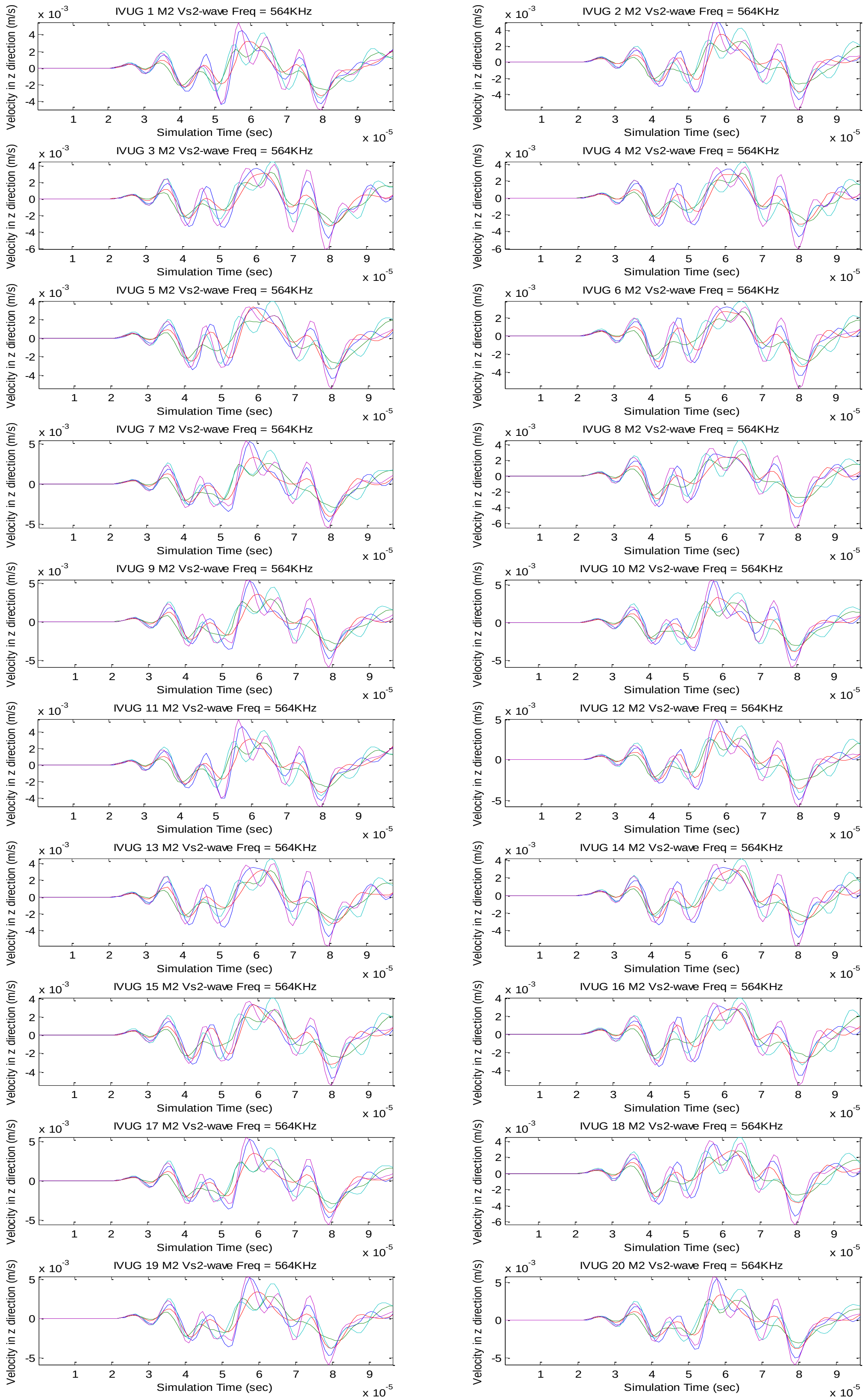

IVUG 1 M1.2 Vs2-wave Freq = 564KHz
Velocity in z direction (m/s)
Simulation Time (sec)

IVUG 2 M1.2 Vs2-wave Freq = 564KHz
Velocity in z direction (m/s)
Simulation Time (sec)

IVUG 3 M1.2 Vs2-wave Freq = 564KHz
Velocity in z direction (m/s)
Simulation Time (sec)

IVUG 4 M1.2 Vs2-wave Freq = 564KHz
Velocity in z direction (m/s)
Simulation Time (sec)

IVUG 5 M1.2 Vs2-wave Freq = 564KHz
Velocity in z direction (m/s)
Simulation Time (sec)

IVUG 6 M1.2 Vs2-wave Freq = 564KHz
Velocity in z direction (m/s)
Simulation Time (sec)

IVUG 7 M1.2 Vs2-wave Freq = 564KHz
Velocity in z direction (m/s)
Simulation Time (sec)

IVUG 8 M1.2 Vs2-wave Freq = 564KHz
Velocity in z direction (m/s)
Simulation Time (sec)

IVUG 9 M1.2 Vs2-wave Freq = 564KHz
Velocity in z direction (m/s)
Simulation Time (sec)

IVUG 10 M1.2 Vs2-wave Freq = 564KHz
Velocity in z direction (m/s)
Simulation Time (sec)

IVUG 11 M1.2 Vs2-wave Freq = 564KHz
Velocity in z direction (m/s)
Simulation Time (sec)

IVUG 12 M1.2 Vs2-wave Freq = 564KHz
Velocity in z direction (m/s)
Simulation Time (sec)

IVUG 13 M1.2 Vs2-wave Freq = 564KHz
Velocity in z direction (m/s)
Simulation Time (sec)

IVUG 14 M1.2 Vs2-wave Freq = 564KHz
Velocity in z direction (m/s)
Simulation Time (sec)

IVUG 15 M1.2 Vs2-wave Freq = 564KHz
Velocity in z direction (m/s)
Simulation Time (sec)

IVUG 16 M1.2 Vs2-wave Freq = 564KHz
Velocity in z direction (m/s)
Simulation Time (sec)

IVUG 17 M1.2 Vs2-wave Freq = 564KHz
Velocity in z direction (m/s)
Simulation Time (sec)

IVUG 18 M1.2 Vs2-wave Freq = 564KHz
Velocity in z direction (m/s)
Simulation Time (sec)

IVUG 19 M1.2 Vs2-wave Freq = 564KHz
Velocity in z direction (m/s)
Simulation Time (sec)

IVUG 20 M1.2 Vs2-wave Freq = 564KHz
Velocity in z direction (m/s)
Simulation Time (sec)

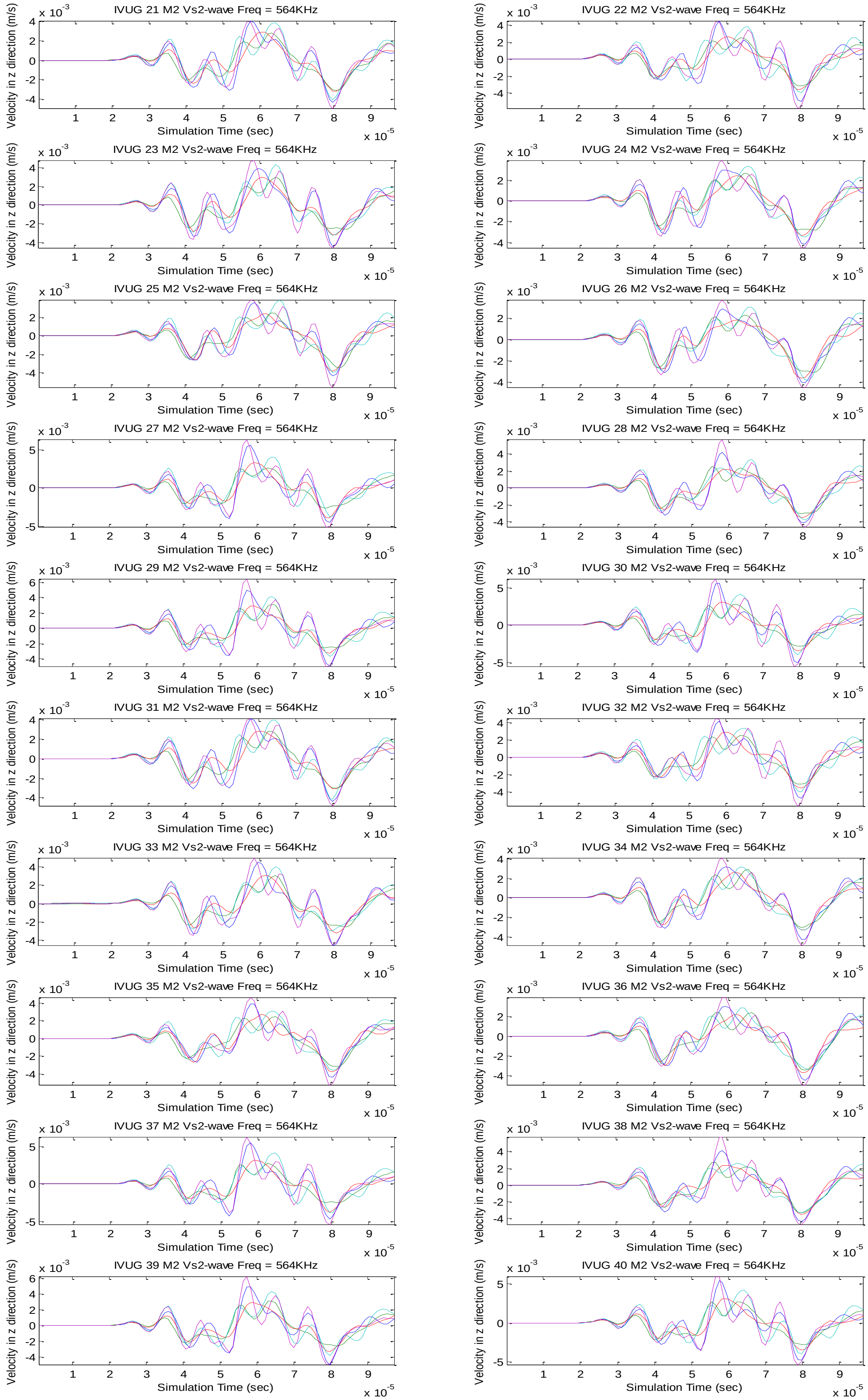

IVUG 1 M2 Vp-wave Freq = 564KHz
Velocity in y direction (m/s)
x 10^-3
Simulation Time (sec)
x 10^-5

IVUG 2 M2 Vp-wave Freq = 564KHz
Velocity in y direction (m/s)
x 10^-3
Simulation Time (sec)
x 10^-5

IVUG 3 M2 Vp-wave Freq = 564KHz
Velocity in y direction (m/s)
x 10^-3
Simulation Time (sec)
x 10^-5

IVUG 4 M2 Vp-wave Freq = 564KHz
Velocity in y direction (m/s)
x 10^-3
Simulation Time (sec)
x 10^-5

IVUG 5 M2 Vp-wave Freq = 564KHz
Velocity in y direction (m/s)
x 10^-3
Simulation Time (sec)
x 10^-5

IVUG 6 M2 Vp-wave Freq = 564KHz
Velocity in y direction (m/s)
x 10^-3
Simulation Time (sec)
x 10^-5

IVUG 7 M2 Vp-wave Freq = 564KHz
Velocity in y direction (m/s)
x 10^-3
Simulation Time (sec)
x 10^-5

IVUG 8 M2 Vp-wave Freq = 564KHz
Velocity in y direction (m/s)
x 10^-3
Simulation Time (sec)
x 10^-5

IVUG 9 M2 Vp-wave Freq = 564KHz
Velocity in y direction (m/s)
x 10^-3
Simulation Time (sec)
x 10^-5

IVUG 10 M2 Vp-wave Freq = 564KHz
Velocity in y direction (m/s)
x 10^-3
Simulation Time (sec)
x 10^-5

IVUG 11 M2 Vp-wave Freq = 564KHz
Velocity in y direction (m/s)
x 10^-3
Simulation Time (sec)
x 10^-5

IVUG 12 M2 Vp-wave Freq = 564KHz
Velocity in y direction (m/s)
x 10^-3
Simulation Time (sec)
x 10^-5

IVUG 13 M2 Vp-wave Freq = 564KHz
Velocity in y direction (m/s)
x 10^-3
Simulation Time (sec)
x 10^-5

IVUG 14 M2 Vp-wave Freq = 564KHz
Velocity in y direction (m/s)
x 10^-3
Simulation Time (sec)
x 10^-5

IVUG 15 M2 Vp-wave Freq = 564KHz
Velocity in y direction (m/s)
x 10^-3
Simulation Time (sec)
x 10^-5

IVUG 16 M2 Vp-wave Freq = 564KHz
Velocity in y direction (m/s)
x 10^-3
Simulation Time (sec)
x 10^-5

IVUG 17 M2 Vp-wave Freq = 564KHz
Velocity in y direction (m/s)
x 10^-3
Simulation Time (sec)
x 10^-5

IVUG 18 M2 Vp-wave Freq = 564KHz
Velocity in y direction (m/s)
x 10^-3
Simulation Time (sec)
x 10^-5

IVUG 19 M2 Vp-wave Freq = 564KHz
Velocity in y direction (m/s)
x 10^-3
Simulation Time (sec)
x 10^-5

IVUG 20 M2 Vp-wave Freq = 564KHz
Velocity in y direction (m/s)
x 10^-3
Simulation Time (sec)
x 10^-5

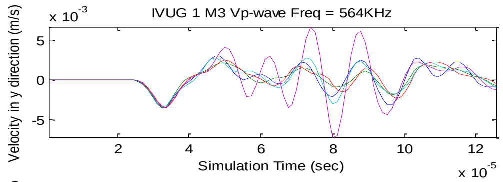
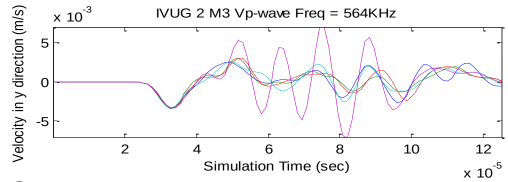
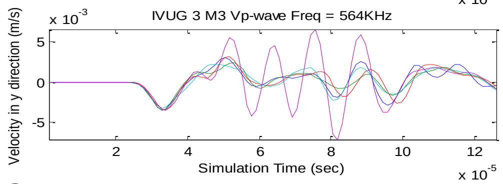
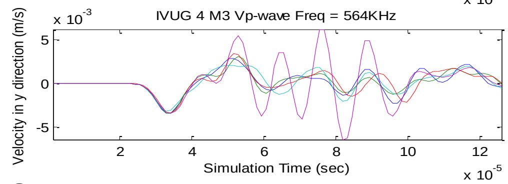
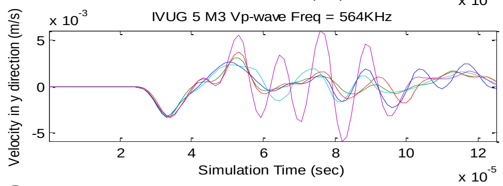
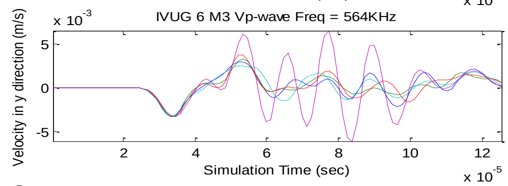
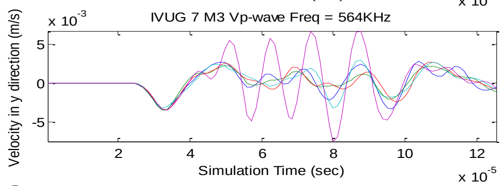
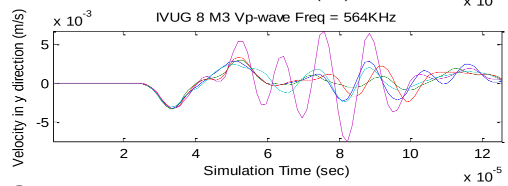
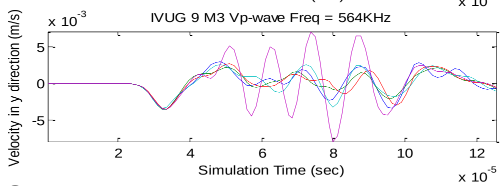
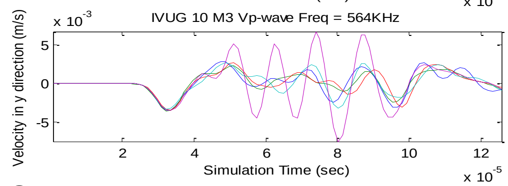
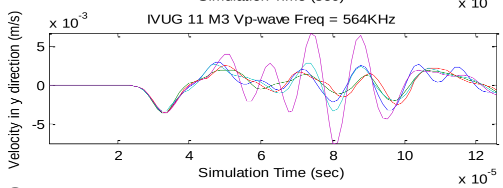
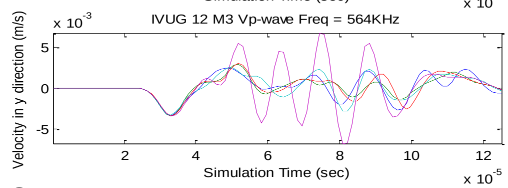
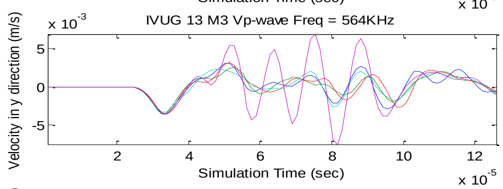
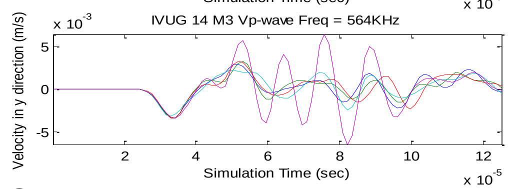
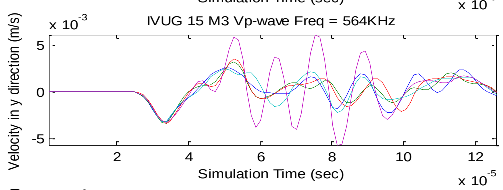
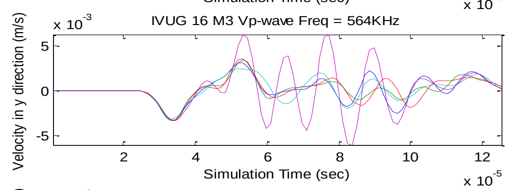
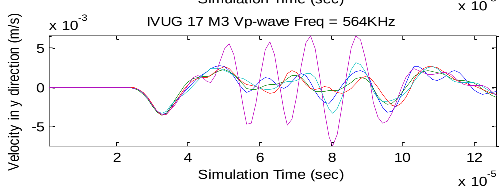
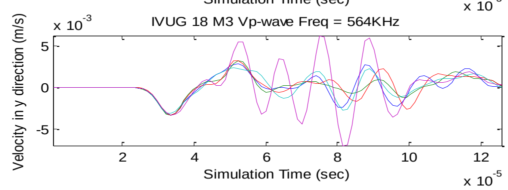
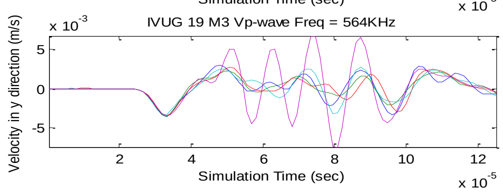
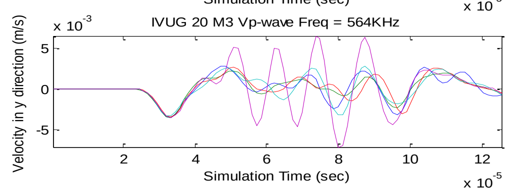

IVUG 21 M2 Vp-wave Freq = 564KHz
IVUG 22 M2 Vp-wave Freq = 564KHz
IVUG 23 M2 Vp-wave Freq = 564KHz
IVUG 24 M2 Vp-wave Freq = 564KHz
IVUG 25 M2 Vp-wave Freq = 564KHz
IVUG 26 M2 Vp-wave Freq = 564KHz
IVUG 27 M2 Vp-wave Freq = 564KHz
IVUG 28 M2 Vp-wave Freq = 564KHz
IVUG 29 M2 Vp-wave Freq = 564KHz
IVUG 30 M2 Vp-wave Freq = 564KHz
IVUG 31 M2 Vp-wave Freq = 564KHz
IVUG 32 M2 Vp-wave Freq = 564KHz
IVUG 33 M2 Vp-wave Freq = 564KHz
IVUG 34 M2 Vp-wave Freq = 564KHz
IVUG 35 M2 Vp-wave Freq = 564KHz
IVUG 36 M2 Vp-wave Freq = 564KHz
IVUG 37 M2 Vp-wave Freq = 564KHz
IVUG 38 M2 Vp-wave Freq = 564KHz
IVUG 39 M2 Vp-wave Freq = 564KHz
IVUG 40 M2 Vp-wave Freq = 564KHz
Velocity in y direction (m/s)
Simulation Time (sec)
x 10-3
x 10-5

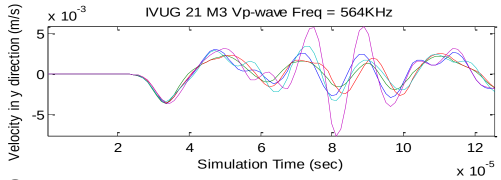

IVUG 1 M2 Vs1-wave Freq = 564KHz
IVUG 2 M2 Vs1-wave Freq = 564KHz
IVUG 3 M2 Vs1-wave Freq = 564KHz
IVUG 4 M2 Vs1-wave Freq = 564KHz
IVUG 5 M2 Vs1-wave Freq = 564KHz
IVUG 6 M2 Vs1-wave Freq = 564KHz
IVUG 7 M2 Vs1-wave Freq = 564KHz
IVUG 8 M2 Vs1-wave Freq = 564KHz
IVUG 9 M2 Vs1-wave Freq = 564KHz
IVUG 10 M2 Vs1-wave Freq = 564KHz
IVUG 11 M2 Vs1-wave Freq = 564KHz
IVUG 12 M2 Vs1-wave Freq = 564KHz
IVUG 13 M2 Vs1-wave Freq = 564KHz
IVUG 14 M2 Vs1-wave Freq = 564KHz
IVUG 15 M2 Vs1-wave Freq = 564KHz
IVUG 16 M2 Vs1-wave Freq = 564KHz
IVUG 17 M2 Vs1-wave Freq = 564KHz
IVUG 18 M2 Vs1-wave Freq = 564KHz
IVUG 19 M2 Vs1-wave Freq = 564KHz
IVUG 20 M2 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
x 10^-3
Simulation Time (sec)
x 10^-5

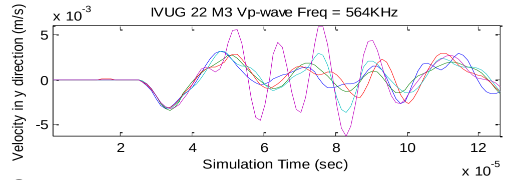

IVUG 21 M2 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
x 10-3
Simulation Time (sec)
x 10-5

IVUG 22 M2 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
x 10-3
Simulation Time (sec)
x 10-5

IVUG 23 M2 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
x 10-3
Simulation Time (sec)
x 10-5

IVUG 24 M2 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
x 10-3
Simulation Time (sec)
x 10-5

IVUG 25 M2 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
x 10-3
Simulation Time (sec)
x 10-5

IVUG 26 M2 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
x 10-3
Simulation Time (sec)
x 10-5

IVUG 27 M2 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
x 10-3
Simulation Time (sec)
x 10-5

IVUG 28 M2 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
x 10-3
Simulation Time (sec)
x 10-5

IVUG 29 M2 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
x 10-3
Simulation Time (sec)
x 10-5

IVUG 30 M2 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
x 10-3
Simulation Time (sec)
x 10-5

IVUG 31 M2 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
x 10-3
Simulation Time (sec)
x 10-5

IVUG 32 M2 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
x 10-3
Simulation Time (sec)
x 10-5

IVUG 33 M2 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
x 10-3
Simulation Time (sec)
x 10-5

IVUG 34 M2 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
x 10-3
Simulation Time (sec)
x 10-5

IVUG 35 M2 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
x 10-3
Simulation Time (sec)
x 10-5

IVUG 36 M2 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
x 10-3
Simulation Time (sec)
x 10-5

IVUG 37 M2 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
x 10-3
Simulation Time (sec)
x 10-5

IVUG 38 M2 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
x 10-3
Simulation Time (sec)
x 10-5

IVUG 39 M2 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
x 10-3
Simulation Time (sec)
x 10-5

IVUG 40 M2 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
x 10-3
Simulation Time (sec)
x 10-5

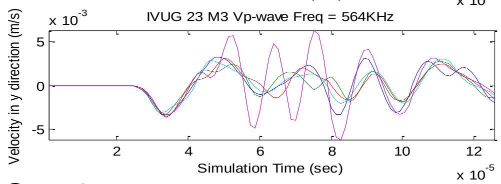

IVUG 1 M2 Vs2-wave Freq = 564KHz
IVUG 2 M2 Vs2-wave Freq = 564KHz
IVUG 3 M2 Vs2-wave Freq = 564KHz
IVUG 4 M2 Vs2-wave Freq = 564KHz
IVUG 5 M2 Vs2-wave Freq = 564KHz
IVUG 6 M2 Vs2-wave Freq = 564KHz
IVUG 7 M2 Vs2-wave Freq = 564KHz
IVUG 8 M2 Vs2-wave Freq = 564KHz
IVUG 9 M2 Vs2-wave Freq = 564KHz
IVUG 10 M2 Vs2-wave Freq = 564KHz
IVUG 11 M2 Vs2-wave Freq = 564KHz
IVUG 12 M2 Vs2-wave Freq = 564KHz
IVUG 13 M2 Vs2-wave Freq = 564KHz
IVUG 14 M2 Vs2-wave Freq = 564KHz
IVUG 15 M2 Vs2-wave Freq = 564KHz
IVUG 16 M2 Vs2-wave Freq = 564KHz
IVUG 17 M2 Vs2-wave Freq = 564KHz
IVUG 18 M2 Vs2-wave Freq = 564KHz
IVUG 19 M2 Vs2-wave Freq = 564KHz
IVUG 20 M2 Vs2-wave Freq = 564KHz
Velocity in z direction (m/s)
Simulation Time (sec)
x 10^-3
x 10^-5

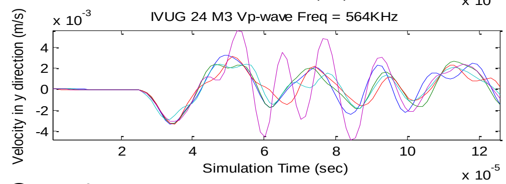

IVUG 21 M2 Vs2-wave Freq = 564KHz
IVUG 22 M2 Vs2-wave Freq = 564KHz
IVUG 23 M2 Vs2-wave Freq = 564KHz
IVUG 24 M2 Vs2-wave Freq = 564KHz
IVUG 25 M2 Vs2-wave Freq = 564KHz
IVUG 26 M2 Vs2-wave Freq = 564KHz
IVUG 27 M2 Vs2-wave Freq = 564KHz
IVUG 28 M2 Vs2-wave Freq = 564KHz
IVUG 29 M2 Vs2-wave Freq = 564KHz
IVUG 30 M2 Vs2-wave Freq = 564KHz
IVUG 31 M2 Vs2-wave Freq = 564KHz
IVUG 32 M2 Vs2-wave Freq = 564KHz
IVUG 33 M2 Vs2-wave Freq = 564KHz
IVUG 34 M2 Vs2-wave Freq = 564KHz
IVUG 35 M2 Vs2-wave Freq = 564KHz
IVUG 36 M2 Vs2-wave Freq = 564KHz
IVUG 37 M2 Vs2-wave Freq = 564KHz
IVUG 38 M2 Vs2-wave Freq = 564KHz
IVUG 39 M2 Vs2-wave Freq = 564KHz
IVUG 40 M2 Vs2-wave Freq = 564KHz
Velocity in z direction (m/s)
Simulation Time (sec)
x 10-3
x 10-5

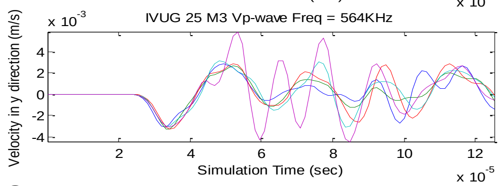

IVUG 1 M3 Vp-wave Freq = 564KHz
Velocity in y direction (m/s)
Simulation Time (sec)

IVUG 2 M3 Vp-wave Freq = 564KHz
Velocity in y direction (m/s)
Simulation Time (sec)

IVUG 3 M3 Vp-wave Freq = 564KHz
Velocity in y direction (m/s)
Simulation Time (sec)

IVUG 4 M3 Vp-wave Freq = 564KHz
Velocity in y direction (m/s)
Simulation Time (sec)

IVUG 5 M3 Vp-wave Freq = 564KHz
Velocity in y direction (m/s)
Simulation Time (sec)

IVUG 6 M3 Vp-wave Freq = 564KHz
Velocity in y direction (m/s)
Simulation Time (sec)

IVUG 7 M3 Vp-wave Freq = 564KHz
Velocity in y direction (m/s)
Simulation Time (sec)

IVUG 8 M3 Vp-wave Freq = 564KHz
Velocity in y direction (m/s)
Simulation Time (sec)

IVUG 9 M3 Vp-wave Freq = 564KHz
Velocity in y direction (m/s)
Simulation Time (sec)

IVUG 10 M3 Vp-wave Freq = 564KHz
Velocity in y direction (m/s)
Simulation Time (sec)

IVUG 11 M3 Vp-wave Freq = 564KHz
Velocity in y direction (m/s)
Simulation Time (sec)

IVUG 12 M3 Vp-wave Freq = 564KHz
Velocity in y direction (m/s)
Simulation Time (sec)

IVUG 13 M3 Vp-wave Freq = 564KHz
Velocity in y direction (m/s)
Simulation Time (sec)

IVUG 14 M3 Vp-wave Freq = 564KHz
Velocity in y direction (m/s)
Simulation Time (sec)

IVUG 15 M3 Vp-wave Freq = 564KHz
Velocity in y direction (m/s)
Simulation Time (sec)

IVUG 16 M3 Vp-wave Freq = 564KHz
Velocity in y direction (m/s)
Simulation Time (sec)

IVUG 17 M3 Vp-wave Freq = 564KHz
Velocity in y direction (m/s)
Simulation Time (sec)

IVUG 18 M3 Vp-wave Freq = 564KHz
Velocity in y direction (m/s)
Simulation Time (sec)

IVUG 19 M3 Vp-wave Freq = 564KHz
Velocity in y direction (m/s)
Simulation Time (sec)

IVUG 20 M3 Vp-wave Freq = 564KHz
Velocity in y direction (m/s)
Simulation Time (sec)

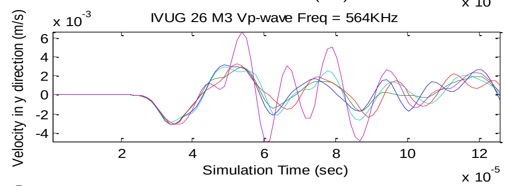

IVUG 21 M3 Vp-wave Freq = 564KHz
Velocity in y direction (m/s)
Simulation Time (sec)

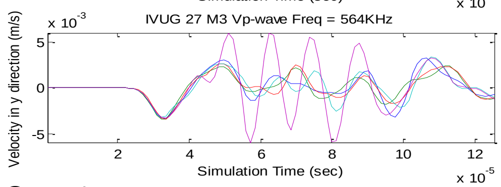

IVUG 22 M3 Vp-wave Freq = 564KHz
Velocity in y direction (m/s)
Simulation Time (sec)

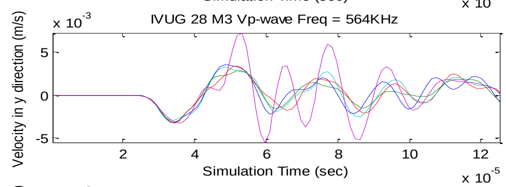

IVUG 23 M3 Vp-wave Freq = 564KHz
Velocity in y direction (m/s)
Simulation Time (sec)

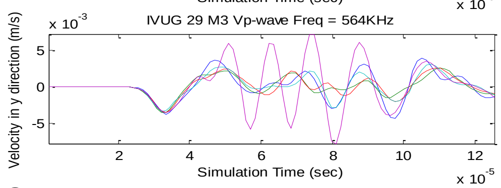

IVUG 24 M3 Vp-wave Freq = 564KHz
Velocity in y direction (m/s)
Simulation Time (sec)

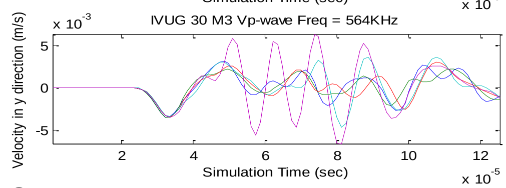

IVUG 25 M3 Vp-wave Freq = 564KHz
Velocity in y direction (m/s)
Simulation Time (sec)

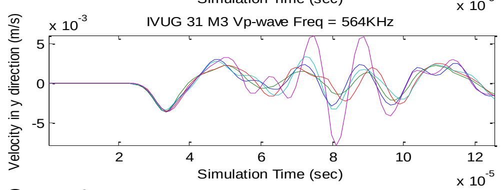

IVUG 26 M3 Vp-wave Freq = 564KHz
Velocity in y direction (m/s)
Simulation Time (sec)

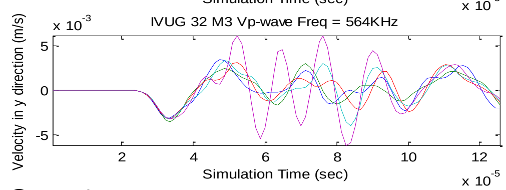

IVUG 27 M3 Vp-wave Freq = 564KHz
Velocity in y direction (m/s)
Simulation Time (sec)

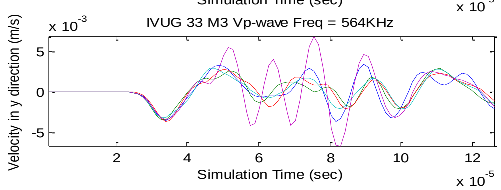

IVUG 28 M3 Vp-wave Freq = 564KHz
Velocity in y direction (m/s)
Simulation Time (sec)

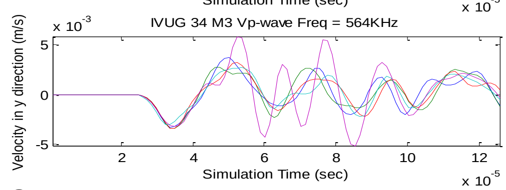

IVUG 29 M3 Vp-wave Freq = 564KHz
Velocity in y direction (m/s)
Simulation Time (sec)

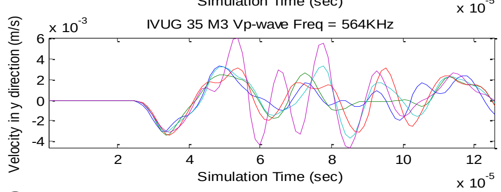

IVUG 30 M3 Vp-wave Freq = 564KHz
Velocity in y direction (m/s)
Simulation Time (sec)

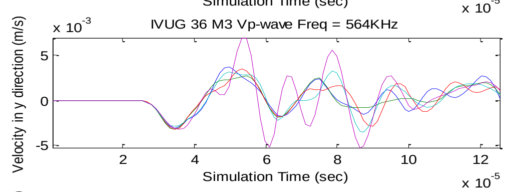

IVUG 31 M3 Vp-wave Freq = 564KHz
Velocity in y direction (m/s)
Simulation Time (sec)

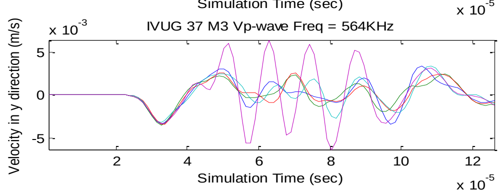

IVUG 32 M3 Vp-wave Freq = 564KHz
Velocity in y direction (m/s)
Simulation Time (sec)

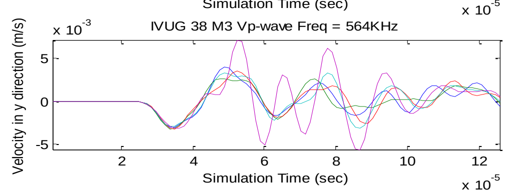

IVUG 33 M3 Vp-wave Freq = 564KHz
Velocity in y direction (m/s)
Simulation Time (sec)

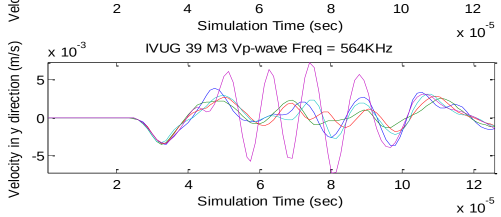

IVUG 34 M3 Vp-wave Freq = 564KHz
Velocity in y direction (m/s)
Simulation Time (sec)

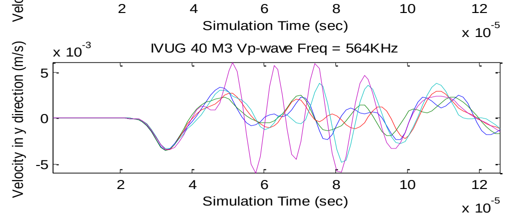

IVUG 35 M3 Vp-wave Freq = 564KHz
Velocity in y direction (m/s)
Simulation Time (sec)

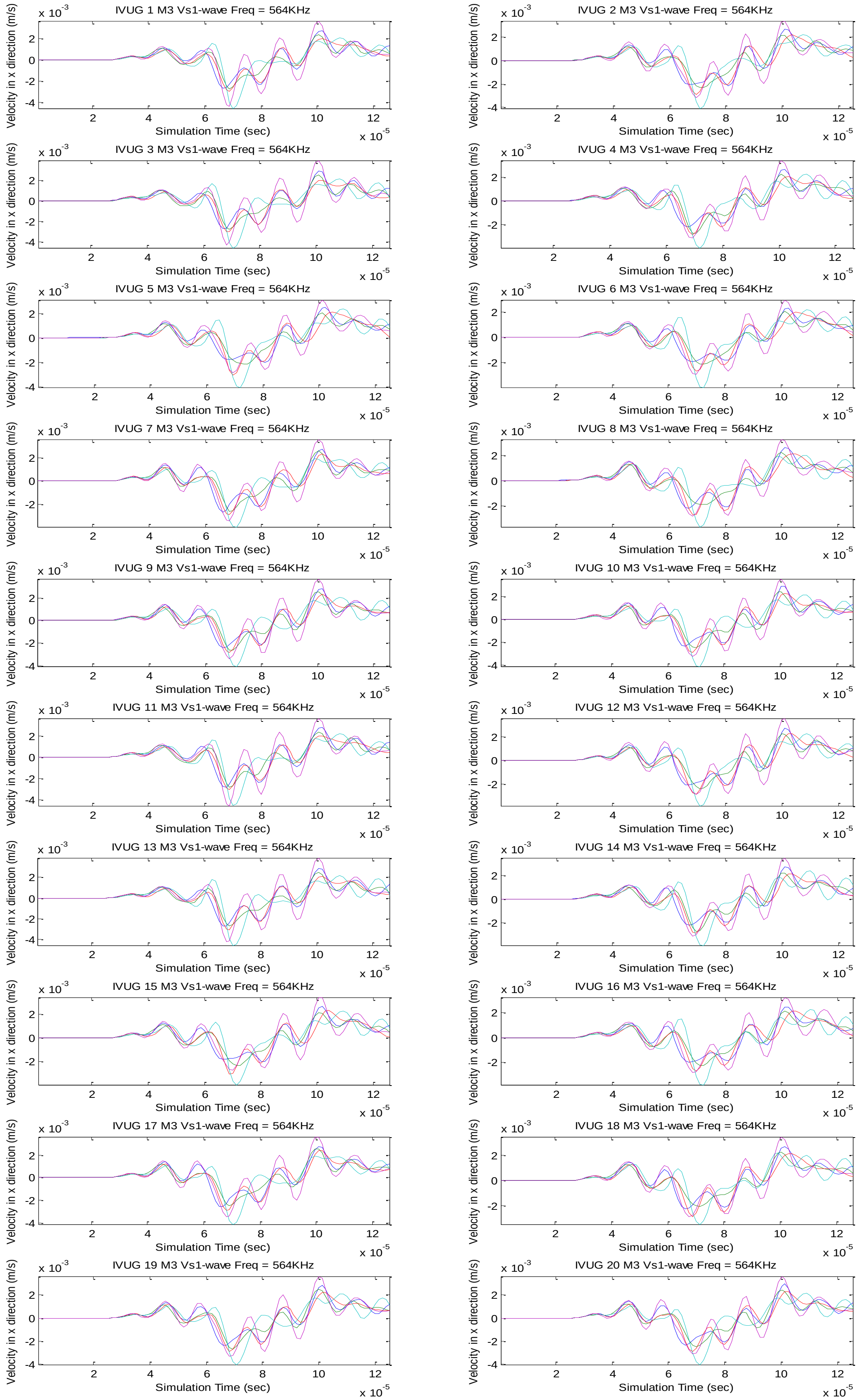

IVUG 36 M3 Vp-wave Freq = 564KHz
Velocity in y direction (m/s)
Simulation Time (sec)

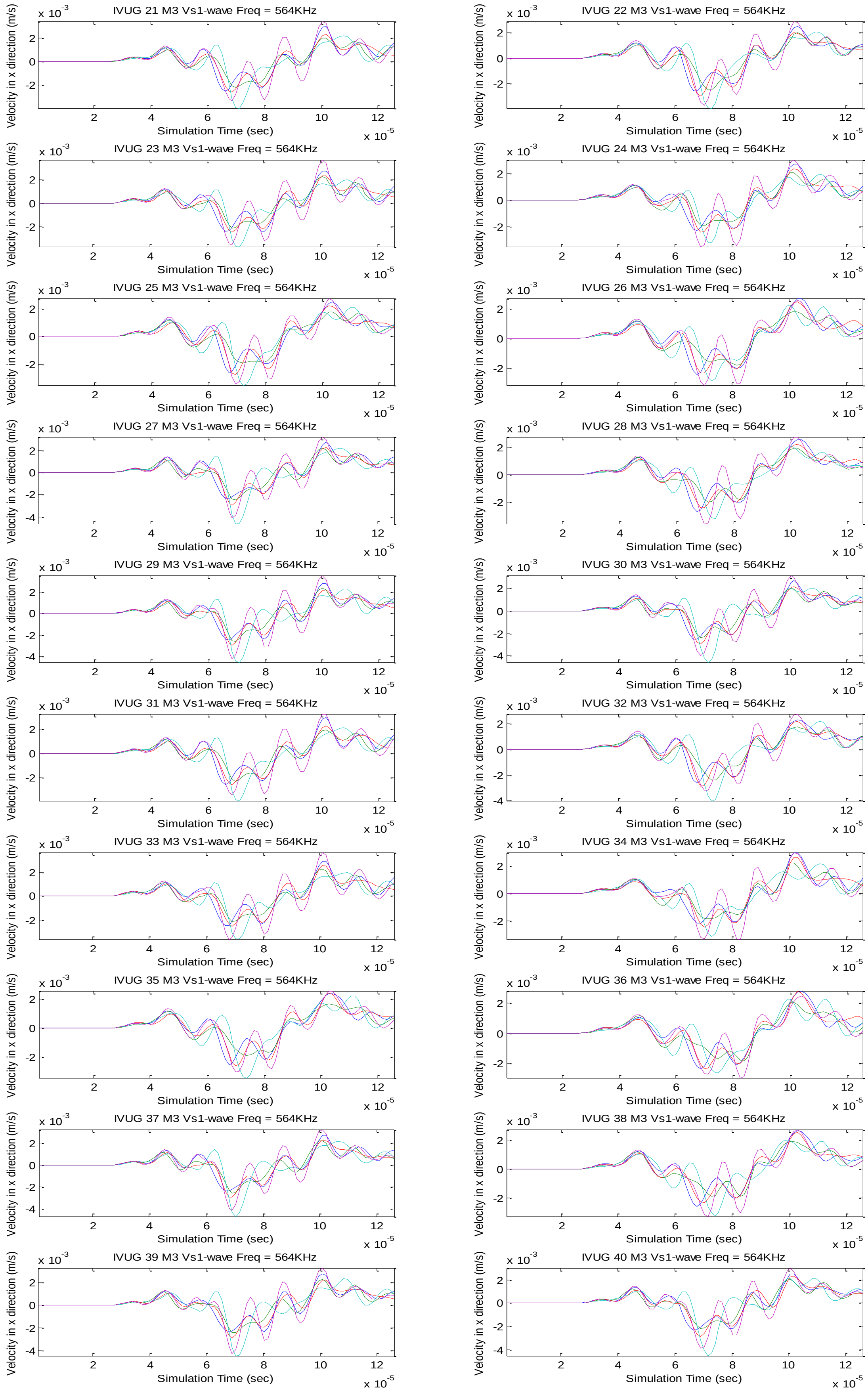

IVUG 37 M3 Vp-wave Freq = 564KHz
Velocity in y direction (m/s)
Simulation Time (sec)

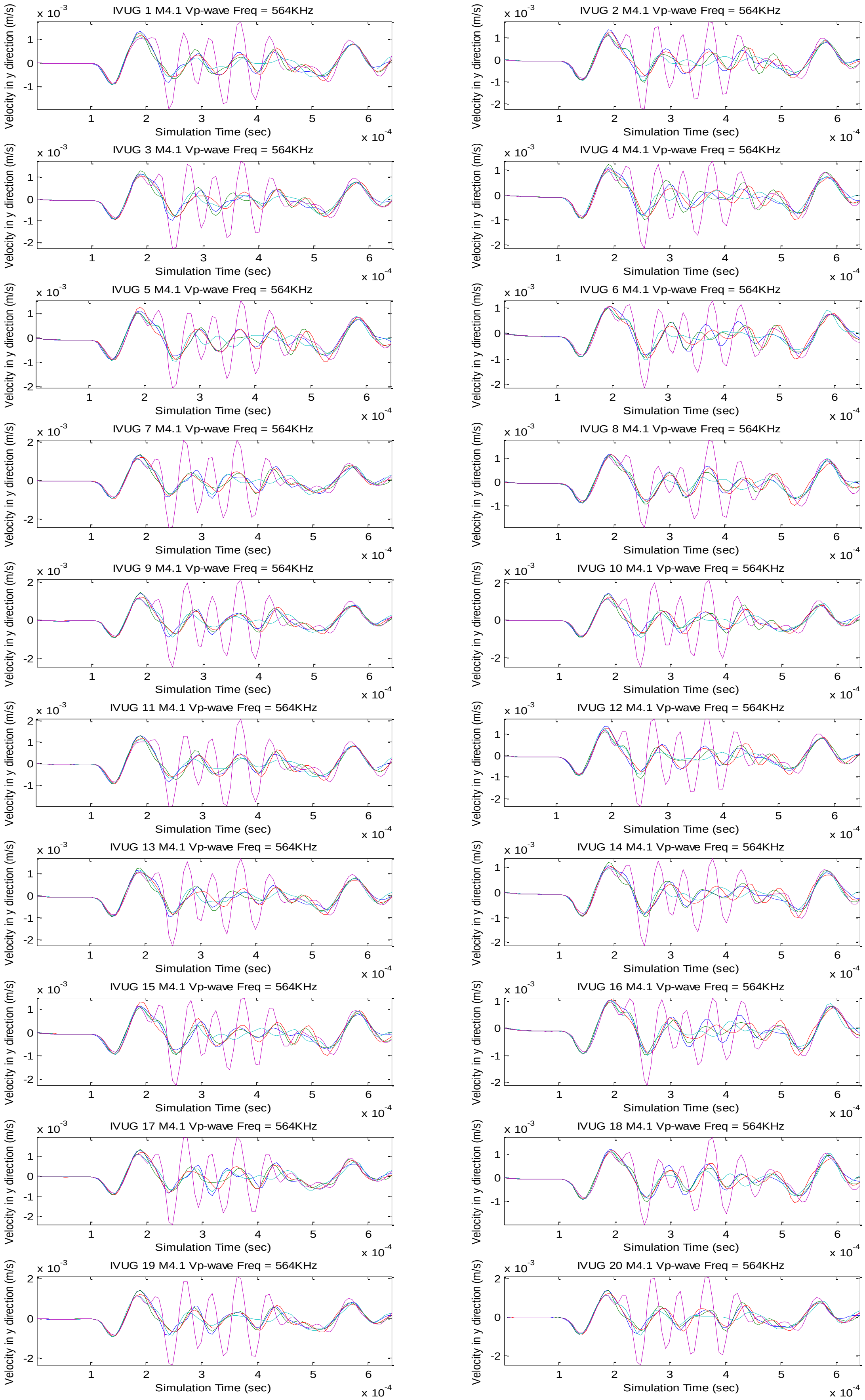

IVUG 38 M3 Vp-wave Freq = 564KHz
Velocity in y direction (m/s)
Simulation Time (sec)

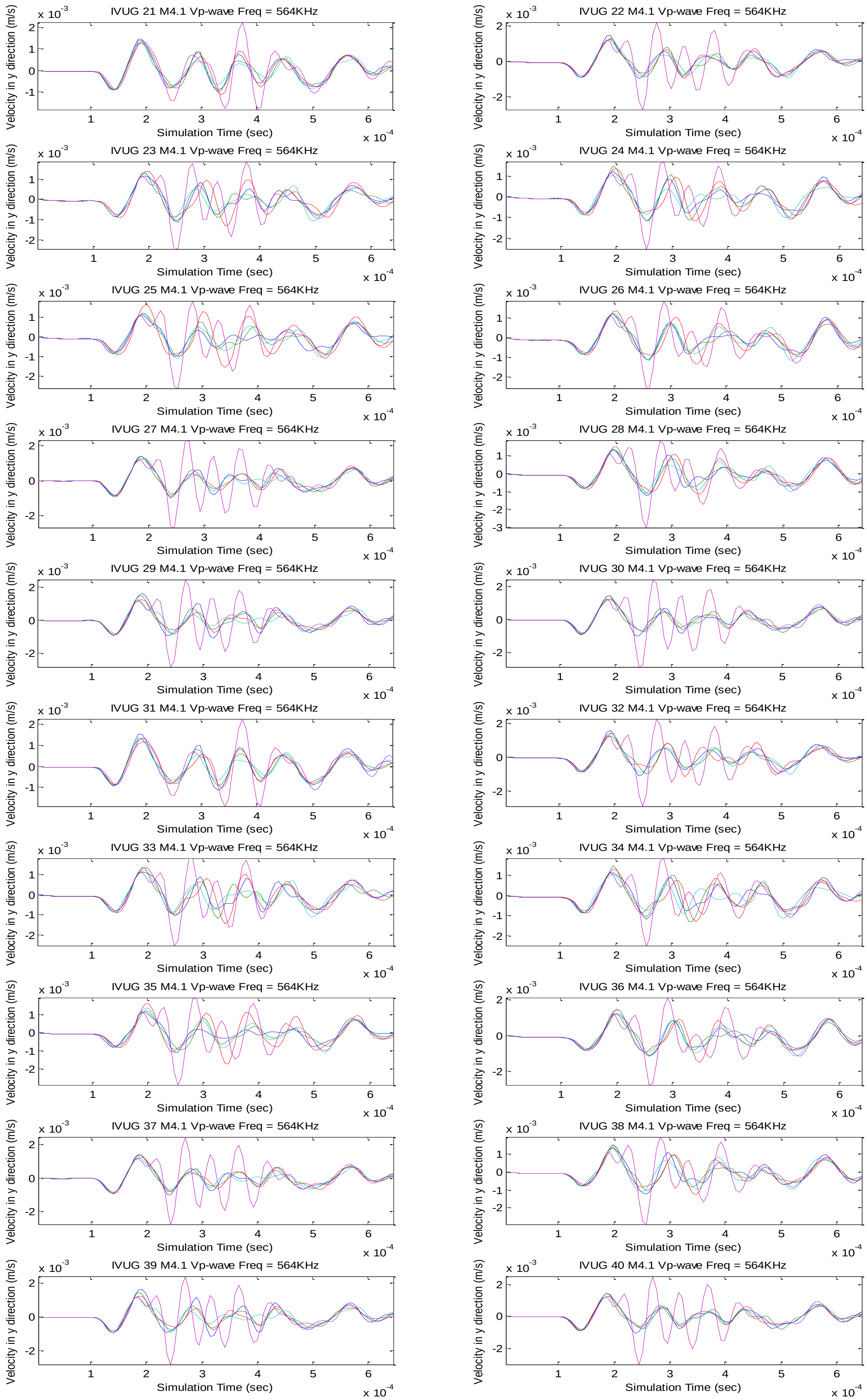

IVUG 39 M3 Vp-wave Freq = 564KHz
Velocity in y direction (m/s)
Simulation Time (sec)

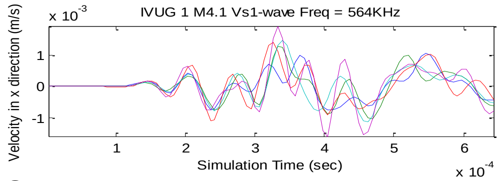
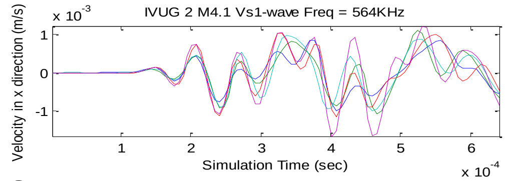
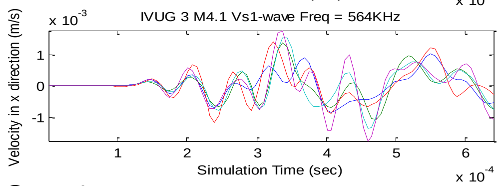
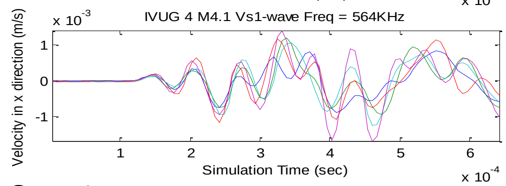
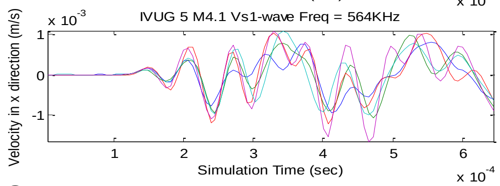
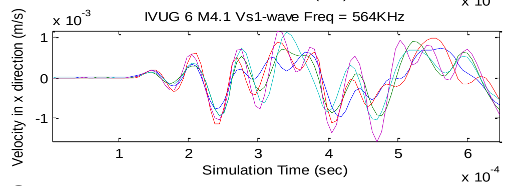
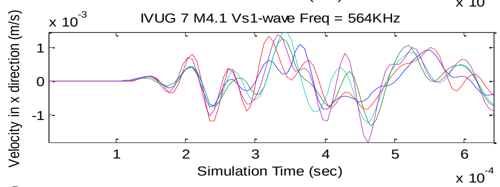
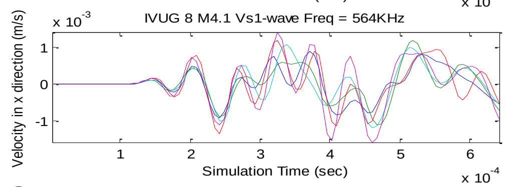
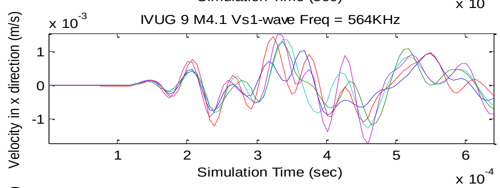
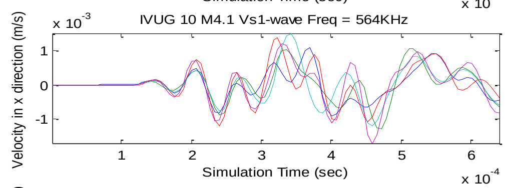
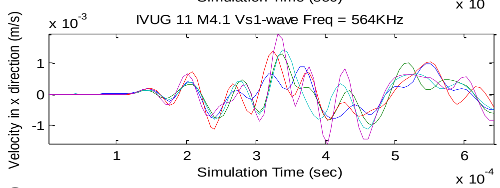
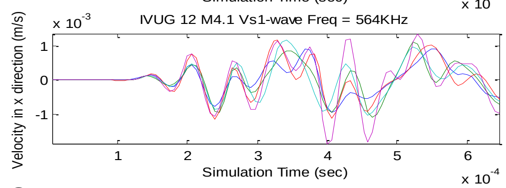
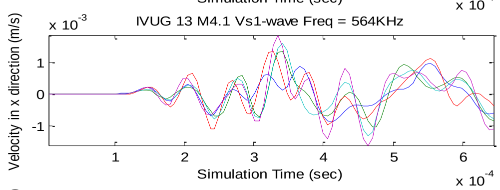
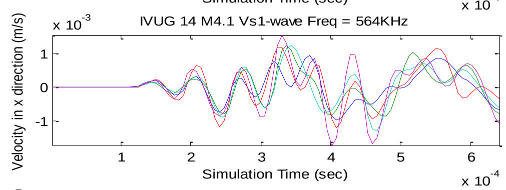
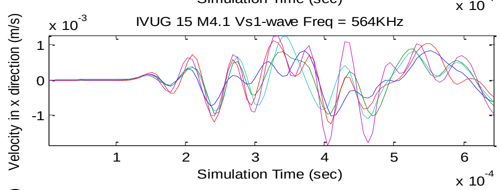
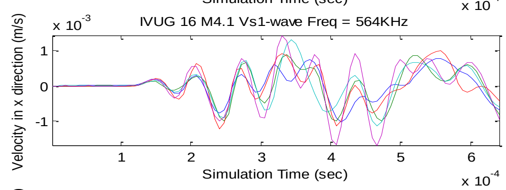
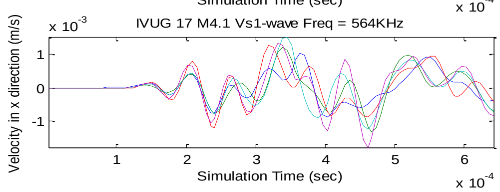
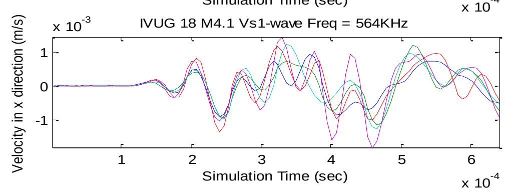
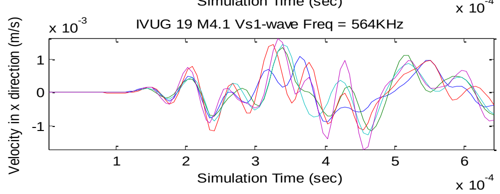
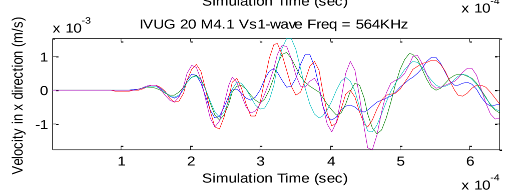

IVUG 40 M3 Vp-wave Freq = 564KHz
Velocity in y direction (m/s)
Simulation Time (sec)

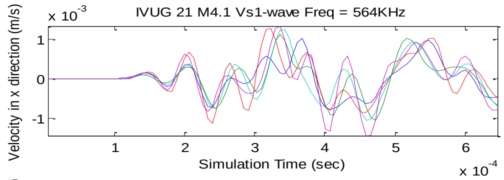
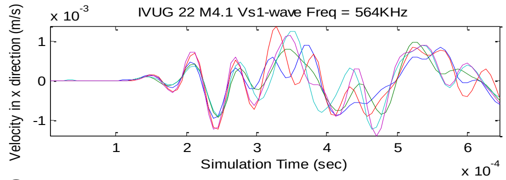
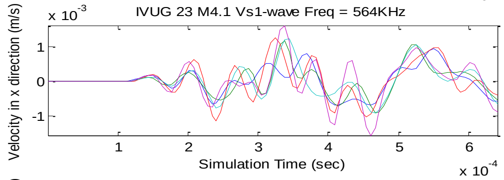
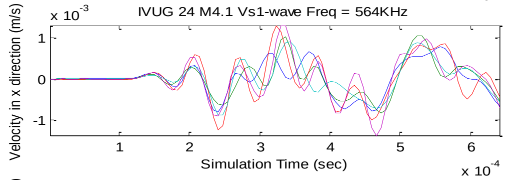
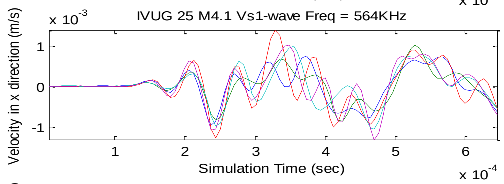
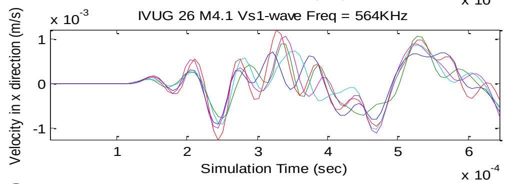
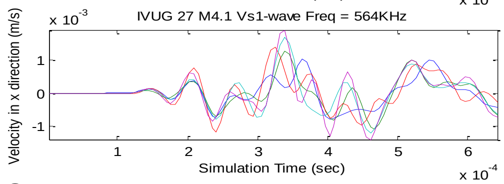
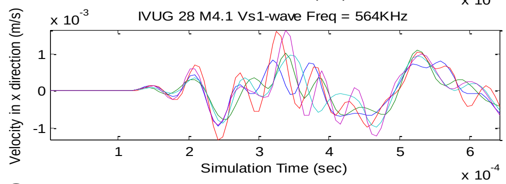
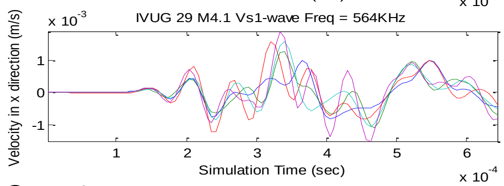
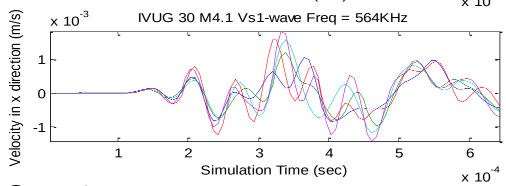
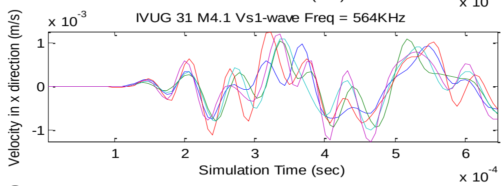
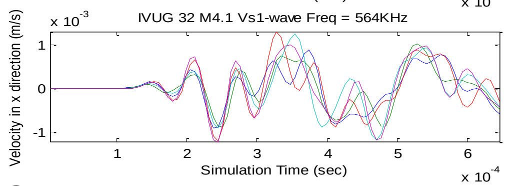
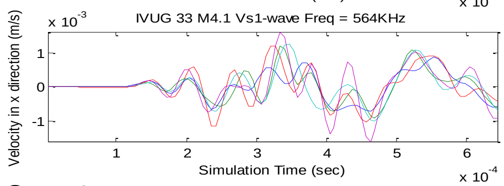
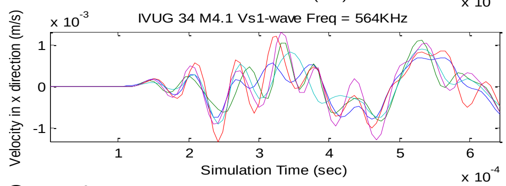
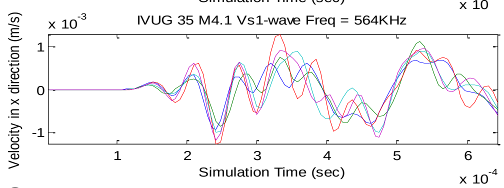
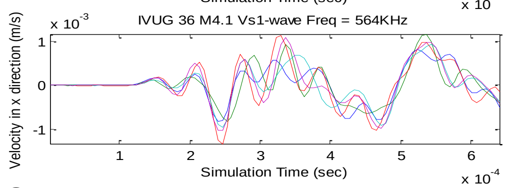
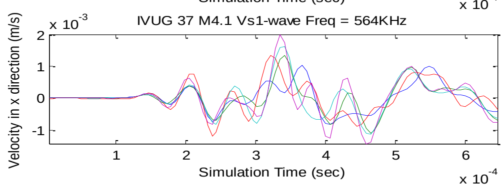
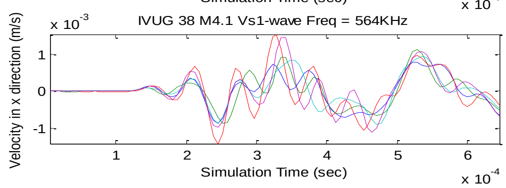
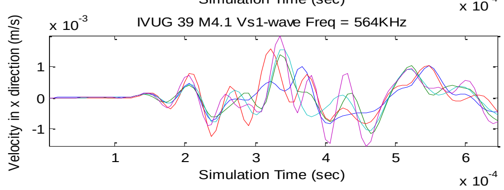
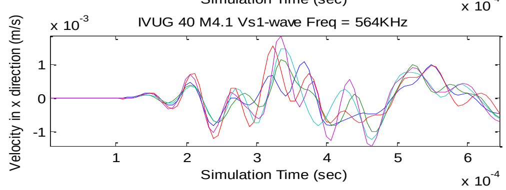

IVUG 1 M3 Vs1-wave Freq = 564KHz
IVUG 2 M3 Vs1-wave Freq = 564KHz
IVUG 3 M3 Vs1-wave Freq = 564KHz
IVUG 4 M3 Vs1-wave Freq = 564KHz
IVUG 5 M3 Vs1-wave Freq = 564KHz
IVUG 6 M3 Vs1-wave Freq = 564KHz
IVUG 7 M3 Vs1-wave Freq = 564KHz
IVUG 8 M3 Vs1-wave Freq = 564KHz
IVUG 9 M3 Vs1-wave Freq = 564KHz
IVUG 10 M3 Vs1-wave Freq = 564KHz
IVUG 11 M3 Vs1-wave Freq = 564KHz
IVUG 12 M3 Vs1-wave Freq = 564KHz
IVUG 13 M3 Vs1-wave Freq = 564KHz
IVUG 14 M3 Vs1-wave Freq = 564KHz
IVUG 15 M3 Vs1-wave Freq = 564KHz
IVUG 16 M3 Vs1-wave Freq = 564KHz
IVUG 17 M3 Vs1-wave Freq = 564KHz
IVUG 18 M3 Vs1-wave Freq = 564KHz
IVUG 19 M3 Vs1-wave Freq = 564KHz
IVUG 20 M3 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
x 10^-3
Simulation Time (sec)
x 10^-5

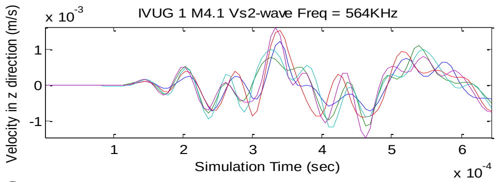

IVUG 21 M3 Vs1-wave Freq = 564KHz
IVUG 22 M3 Vs1-wave Freq = 564KHz
IVUG 23 M3 Vs1-wave Freq = 564KHz
IVUG 24 M3 Vs1-wave Freq = 564KHz
IVUG 25 M3 Vs1-wave Freq = 564KHz
IVUG 26 M3 Vs1-wave Freq = 564KHz
IVUG 27 M3 Vs1-wave Freq = 564KHz
IVUG 28 M3 Vs1-wave Freq = 564KHz
IVUG 29 M3 Vs1-wave Freq = 564KHz
IVUG 30 M3 Vs1-wave Freq = 564KHz
IVUG 31 M3 Vs1-wave Freq = 564KHz
IVUG 32 M3 Vs1-wave Freq = 564KHz
IVUG 33 M3 Vs1-wave Freq = 564KHz
IVUG 34 M3 Vs1-wave Freq = 564KHz
IVUG 35 M3 Vs1-wave Freq = 564KHz
IVUG 36 M3 Vs1-wave Freq = 564KHz
IVUG 37 M3 Vs1-wave Freq = 564KHz
IVUG 38 M3 Vs1-wave Freq = 564KHz
IVUG 39 M3 Vs1-wave Freq = 564KHz
IVUG 40 M3 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
x 10^-3
Simulation Time (sec)
x 10^-5

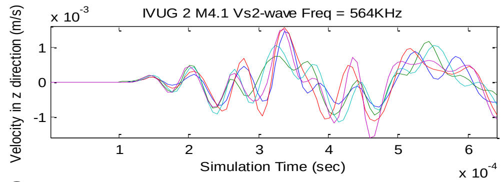

IVUG 1 M4.1 Vp-wave Freq = 564KHz
IVUG 2 M4.1 Vp-wave Freq = 564KHz
IVUG 3 M4.1 Vp-wave Freq = 564KHz
IVUG 4 M4.1 Vp-wave Freq = 564KHz
IVUG 5 M4.1 Vp-wave Freq = 564KHz
IVUG 6 M4.1 Vp-wave Freq = 564KHz
IVUG 7 M4.1 Vp-wave Freq = 564KHz
IVUG 8 M4.1 Vp-wave Freq = 564KHz
IVUG 9 M4.1 Vp-wave Freq = 564KHz
IVUG 10 M4.1 Vp-wave Freq = 564KHz
IVUG 11 M4.1 Vp-wave Freq = 564KHz
IVUG 12 M4.1 Vp-wave Freq = 564KHz
IVUG 13 M4.1 Vp-wave Freq = 564KHz
IVUG 14 M4.1 Vp-wave Freq = 564KHz
IVUG 15 M4.1 Vp-wave Freq = 564KHz
IVUG 16 M4.1 Vp-wave Freq = 564KHz
IVUG 17 M4.1 Vp-wave Freq = 564KHz
IVUG 18 M4.1 Vp-wave Freq = 564KHz
IVUG 19 M4.1 Vp-wave Freq = 564KHz
IVUG 20 M4.1 Vp-wave Freq = 564KHz
Velocity in y direction (m/s)
Simulation Time (sec)
x 10^-3
x 10^-4

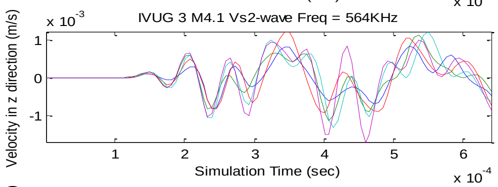

IVUG 21 M4.1 Vp-wave Freq = 564KHz
IVUG 22 M4.1 Vp-wave Freq = 564KHz
IVUG 23 M4.1 Vp-wave Freq = 564KHz
IVUG 24 M4.1 Vp-wave Freq = 564KHz
IVUG 25 M4.1 Vp-wave Freq = 564KHz
IVUG 26 M4.1 Vp-wave Freq = 564KHz
IVUG 27 M4.1 Vp-wave Freq = 564KHz
IVUG 28 M4.1 Vp-wave Freq = 564KHz
IVUG 29 M4.1 Vp-wave Freq = 564KHz
IVUG 30 M4.1 Vp-wave Freq = 564KHz
IVUG 31 M4.1 Vp-wave Freq = 564KHz
IVUG 32 M4.1 Vp-wave Freq = 564KHz
IVUG 33 M4.1 Vp-wave Freq = 564KHz
IVUG 34 M4.1 Vp-wave Freq = 564KHz
IVUG 35 M4.1 Vp-wave Freq = 564KHz
IVUG 36 M4.1 Vp-wave Freq = 564KHz
IVUG 37 M4.1 Vp-wave Freq = 564KHz
IVUG 38 M4.1 Vp-wave Freq = 564KHz
IVUG 39 M4.1 Vp-wave Freq = 564KHz
IVUG 40 M4.1 Vp-wave Freq = 564KHz
Velocity in y direction (m/s)
Simulation Time (sec)
x 10-3
x 10-4

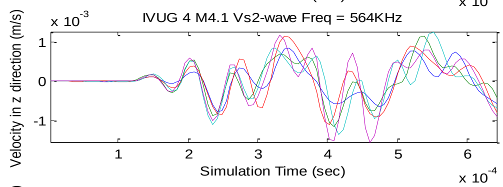

IVUG 1 M4.1 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
x 10-3
Simulation Time (sec)
x 10-4

IVUG 2 M4.1 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
x 10-3
Simulation Time (sec)
x 10-4

IVUG 3 M4.1 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
x 10-3
Simulation Time (sec)
x 10-4

IVUG 4 M4.1 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
x 10-3
Simulation Time (sec)
x 10-4

IVUG 5 M4.1 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
x 10-3
Simulation Time (sec)
x 10-4

IVUG 6 M4.1 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
x 10-3
Simulation Time (sec)
x 10-4

IVUG 7 M4.1 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
x 10-3
Simulation Time (sec)
x 10-4

IVUG 8 M4.1 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
x 10-3
Simulation Time (sec)
x 10-4

IVUG 9 M4.1 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
x 10-3
Simulation Time (sec)
x 10-4

IVUG 10 M4.1 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
x 10-3
Simulation Time (sec)
x 10-4

IVUG 11 M4.1 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
x 10-3
Simulation Time (sec)
x 10-4

IVUG 12 M4.1 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
x 10-3
Simulation Time (sec)
x 10-4

IVUG 13 M4.1 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
x 10-3
Simulation Time (sec)
x 10-4

IVUG 14 M4.1 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
x 10-3
Simulation Time (sec)
x 10-4

IVUG 15 M4.1 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
x 10-3
Simulation Time (sec)
x 10-4

IVUG 16 M4.1 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
x 10-3
Simulation Time (sec)
x 10-4

IVUG 17 M4.1 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
x 10-3
Simulation Time (sec)
x 10-4

IVUG 18 M4.1 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
x 10-3
Simulation Time (sec)
x 10-4

IVUG 19 M4.1 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
x 10-3
Simulation Time (sec)
x 10-4

IVUG 20 M4.1 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
x 10-3
Simulation Time (sec)
x 10-4

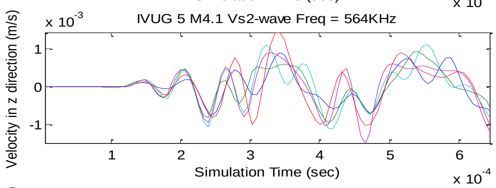

IVUG 21 M4.1 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
x 10-3
Simulation Time (sec)
x 10-4

IVUG 22 M4.1 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
x 10-3
Simulation Time (sec)
x 10-4

IVUG 23 M4.1 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
x 10-3
Simulation Time (sec)
x 10-4

IVUG 24 M4.1 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
x 10-3
Simulation Time (sec)
x 10-4

IVUG 25 M4.1 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
x 10-3
Simulation Time (sec)
x 10-4

IVUG 26 M4.1 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
x 10-3
Simulation Time (sec)
x 10-4

IVUG 27 M4.1 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
x 10-3
Simulation Time (sec)
x 10-4

IVUG 28 M4.1 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
x 10-3
Simulation Time (sec)
x 10-4

IVUG 29 M4.1 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
x 10-3
Simulation Time (sec)
x 10-4

IVUG 30 M4.1 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
x 10-3
Simulation Time (sec)
x 10-4

IVUG 31 M4.1 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
x 10-3
Simulation Time (sec)
x 10-4

IVUG 32 M4.1 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
x 10-3
Simulation Time (sec)
x 10-4

IVUG 33 M4.1 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
x 10-3
Simulation Time (sec)
x 10-4

IVUG 34 M4.1 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
x 10-3
Simulation Time (sec)
x 10-4

IVUG 35 M4.1 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
x 10-3
Simulation Time (sec)
x 10-4

IVUG 36 M4.1 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
x 10-3
Simulation Time (sec)
x 10-4

IVUG 37 M4.1 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
x 10-3
Simulation Time (sec)
x 10-4

IVUG 38 M4.1 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
x 10-3
Simulation Time (sec)
x 10-4

IVUG 39 M4.1 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
x 10-3
Simulation Time (sec)
x 10-4

IVUG 40 M4.1 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
x 10-3
Simulation Time (sec)
x 10-4

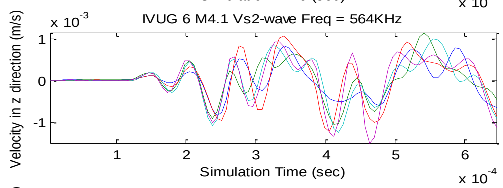

IVUG 1 M4.1 Vs2-wave Freq = 564KHz
Velocity in z direction (m/s)
Simulation Time (sec)

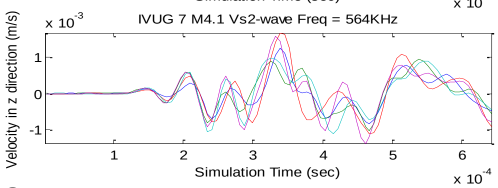

IVUG 2 M4.1 Vs2-wave Freq = 564KHz
Velocity in z direction (m/s)
Simulation Time (sec)

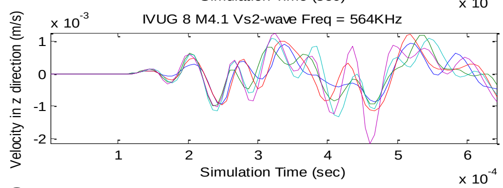

IVUG 3 M4.1 Vs2-wave Freq = 564KHz
Velocity in z direction (m/s)
Simulation Time (sec)

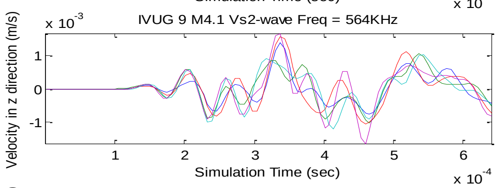

IVUG 4 M4.1 Vs2-wave Freq = 564KHz
Velocity in z direction (m/s)
Simulation Time (sec)

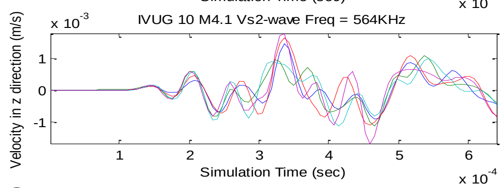

IVUG 5 M4.1 Vs2-wave Freq = 564KHz
Velocity in z direction (m/s)
Simulation Time (sec)

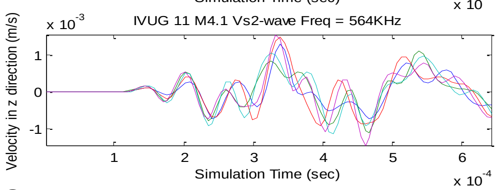

IVUG 6 M4.1 Vs2-wave Freq = 564KHz
Velocity in z direction (m/s)
Simulation Time (sec)

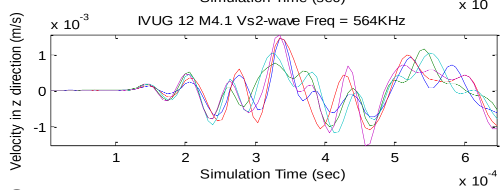

IVUG 7 M4.1 Vs2-wave Freq = 564KHz
Velocity in z direction (m/s)
Simulation Time (sec)

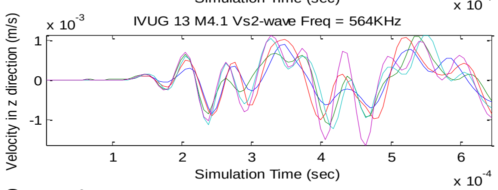

IVUG 8 M4.1 Vs2-wave Freq = 564KHz
Velocity in z direction (m/s)
Simulation Time (sec)

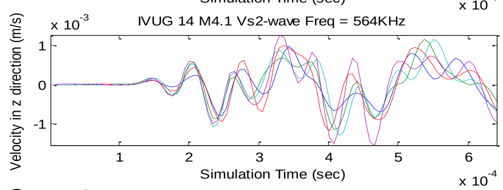

IVUG 9 M4.1 Vs2-wave Freq = 564KHz
Velocity in z direction (m/s)
Simulation Time (sec)

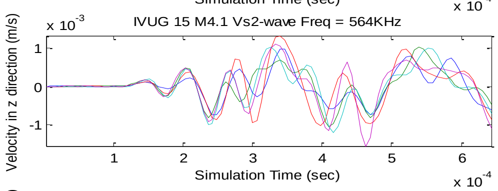

IVUG 10 M4.1 Vs2-wave Freq = 564KHz
Velocity in z direction (m/s)
Simulation Time (sec)

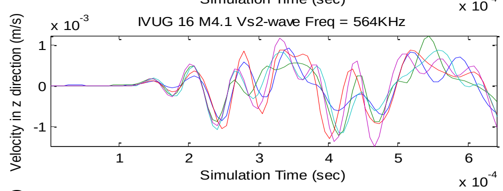

IVUG 11 M4.1 Vs2-wave Freq = 564KHz
Velocity in z direction (m/s)
Simulation Time (sec)

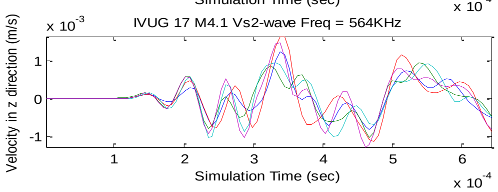

IVUG 12 M4.1 Vs2-wave Freq = 564KHz
Velocity in z direction (m/s)
Simulation Time (sec)

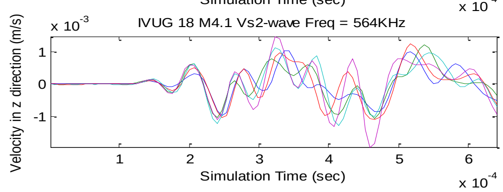

IVUG 13 M4.1 Vs2-wave Freq = 564KHz
Velocity in z direction (m/s)
Simulation Time (sec)

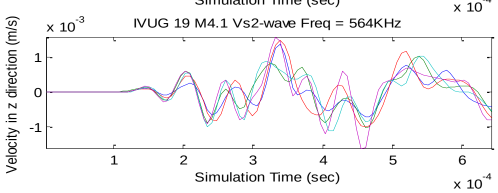

IVUG 14 M4.1 Vs2-wave Freq = 564KHz
Velocity in z direction (m/s)
Simulation Time (sec)

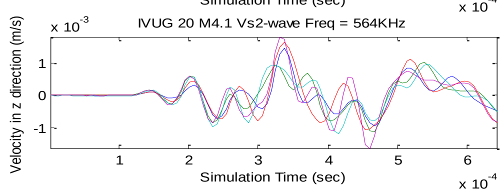

IVUG 15 M4.1 Vs2-wave Freq = 564KHz
Velocity in z direction (m/s)
Simulation Time (sec)

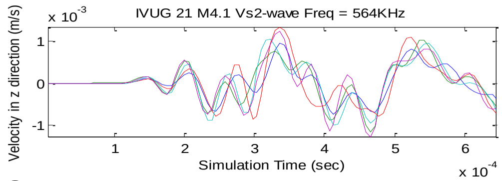

IVUG 16 M4.1 Vs2-wave Freq = 564KHz
Velocity in z direction (m/s)
Simulation Time (sec)

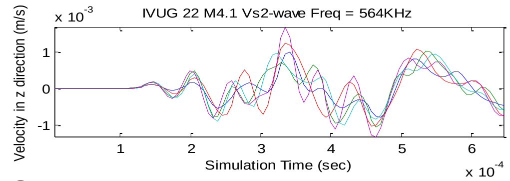

IVUG 17 M4.1 Vs2-wave Freq = 564KHz
Velocity in z direction (m/s)
Simulation Time (sec)

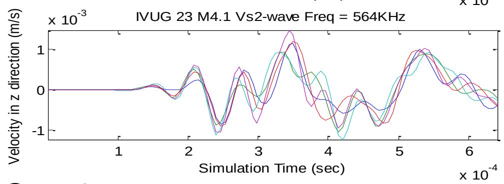

IVUG 18 M4.1 Vs2-wave Freq = 564KHz
Velocity in z direction (m/s)
Simulation Time (sec)

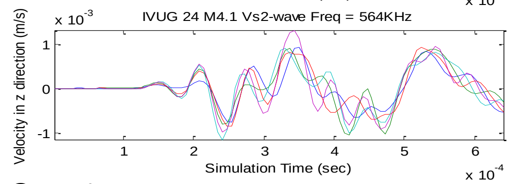

IVUG 19 M4.1 Vs2-wave Freq = 564KHz
Velocity in z direction (m/s)
Simulation Time (sec)

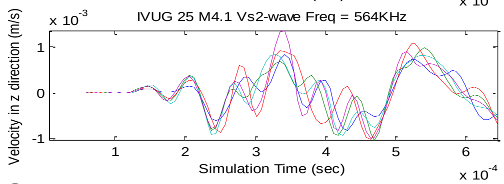

IVUG 20 M4.1 Vs2-wave Freq = 564KHz
Velocity in z direction (m/s)
Simulation Time (sec)

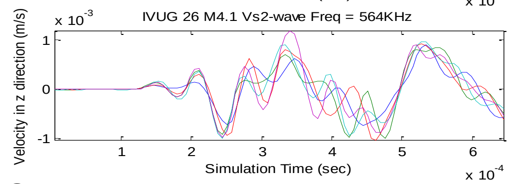

IVUG 21 M4.1 Vs2-wave Freq = 564KHz
Velocity in z direction (m/s)
x 10^-3
Simulation Time (sec)
x 10^-4

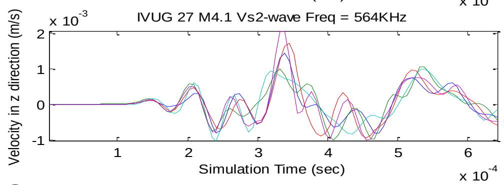

IVUG 22 M4.1 Vs2-wave Freq = 564KHz
Velocity in z direction (m/s)
x 10^-3
Simulation Time (sec)
x 10^-4

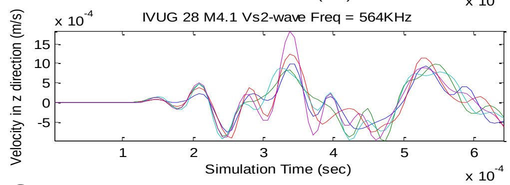

IVUG 23 M4.1 Vs2-wave Freq = 564KHz
Velocity in z direction (m/s)
x 10^-3
Simulation Time (sec)
x 10^-4

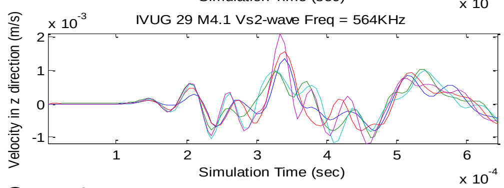

IVUG 24 M4.1 Vs2-wave Freq = 564KHz
Velocity in z direction (m/s)
x 10^-3
Simulation Time (sec)
x 10^-4

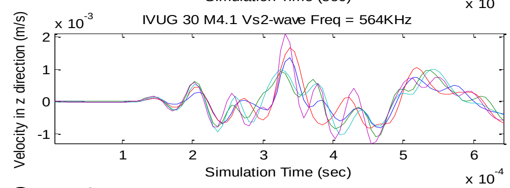

IVUG 25 M4.1 Vs2-wave Freq = 564KHz
Velocity in z direction (m/s)
x 10^-3
Simulation Time (sec)
x 10^-4

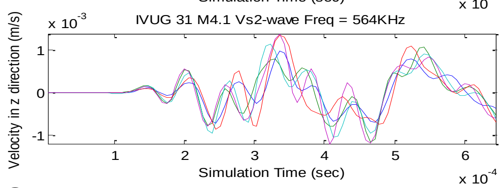

IVUG 26 M4.1 Vs2-wave Freq = 564KHz
Velocity in z direction (m/s)
x 10^-3
Simulation Time (sec)
x 10^-4

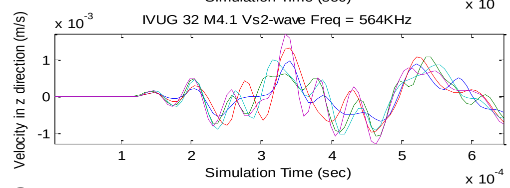

IVUG 27 M4.1 Vs2-wave Freq = 564KHz
Velocity in z direction (m/s)
x 10^-3
Simulation Time (sec)
x 10^-4

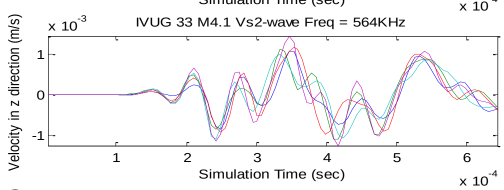

IVUG 28 M4.1 Vs2-wave Freq = 564KHz
Velocity in z direction (m/s)
x 10^-4
Simulation Time (sec)
x 10^-4

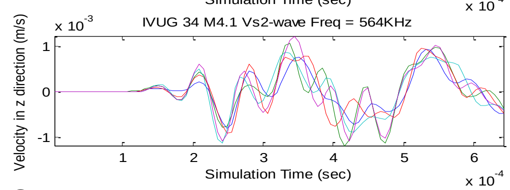

IVUG 29 M4.1 Vs2-wave Freq = 564KHz
Velocity in z direction (m/s)
x 10^-3
Simulation Time (sec)
x 10^-4

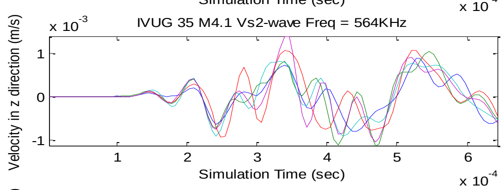

IVUG 30 M4.1 Vs2-wave Freq = 564KHz
Velocity in z direction (m/s)
x 10^-3
Simulation Time (sec)
x 10^-4

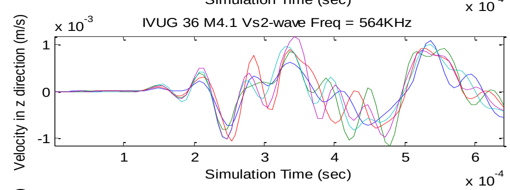

IVUG 31 M4.1 Vs2-wave Freq = 564KHz
Velocity in z direction (m/s)
x 10^-3
Simulation Time (sec)
x 10^-4

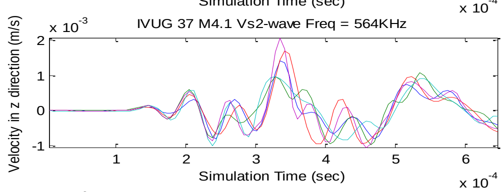

IVUG 32 M4.1 Vs2-wave Freq = 564KHz
Velocity in z direction (m/s)
x 10^-3
Simulation Time (sec)
x 10^-4

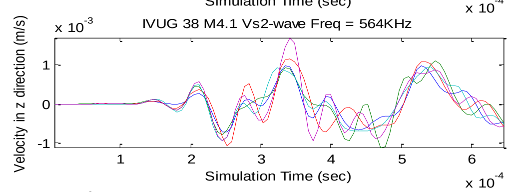

IVUG 33 M4.1 Vs2-wave Freq = 564KHz
Velocity in z direction (m/s)
x 10^-3
Simulation Time (sec)
x 10^-4

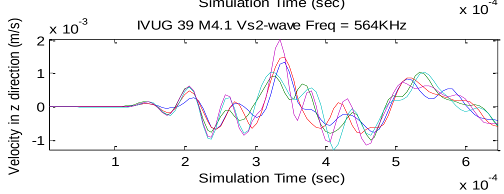

IVUG 34 M4.1 Vs2-wave Freq = 564KHz
Velocity in z direction (m/s)
x 10^-3
Simulation Time (sec)
x 10^-4

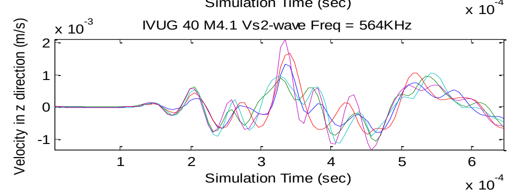

IVUG 35 M4.1 Vs2-wave Freq = 564KHz
Velocity in z direction (m/s)
x 10^-3
Simulation Time (sec)
x 10^-4

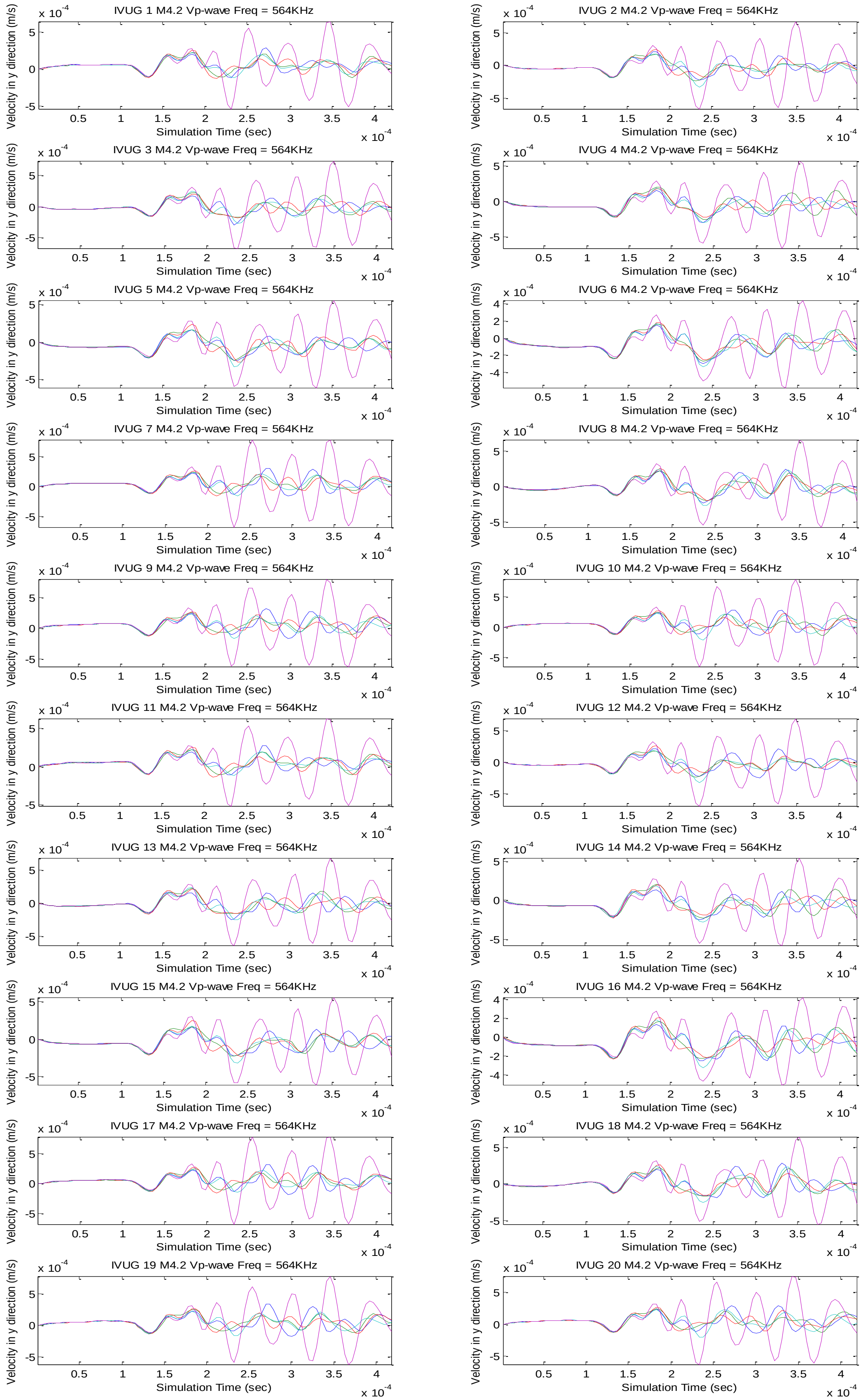

IVUG 36 M4.1 Vs2-wave Freq = 564KHz
Velocity in z direction (m/s)
x 10^-3
Simulation Time (sec)
x 10^-4

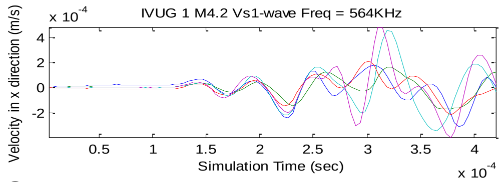
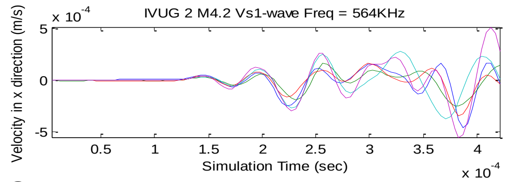
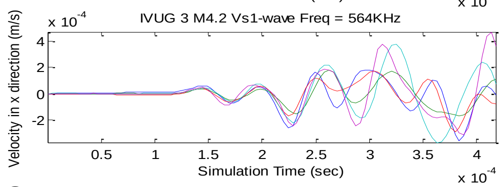
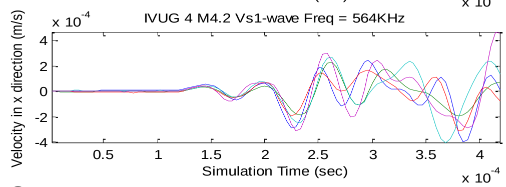
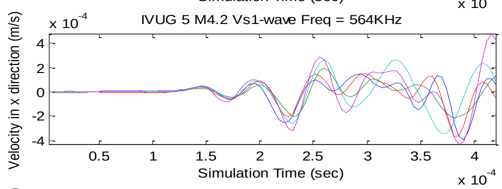
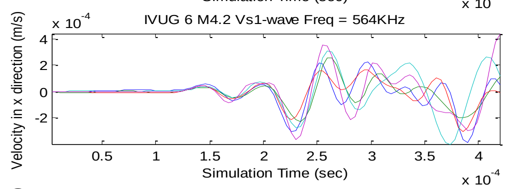
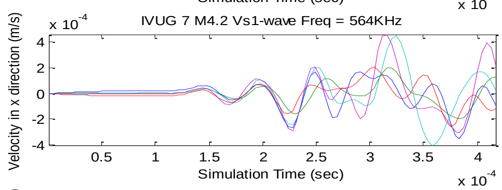
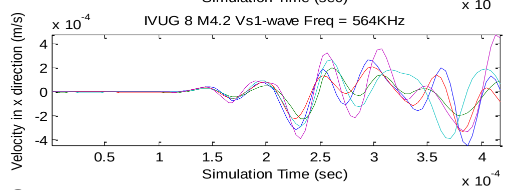
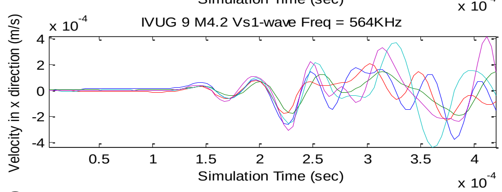
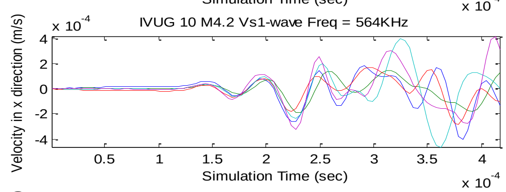
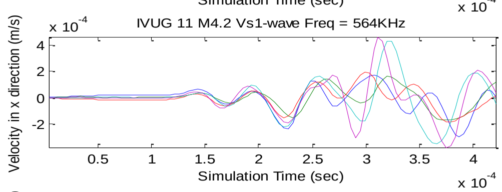
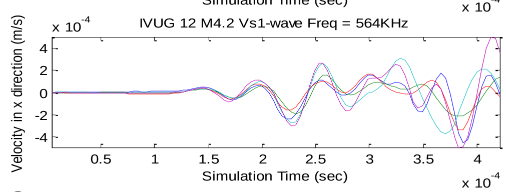
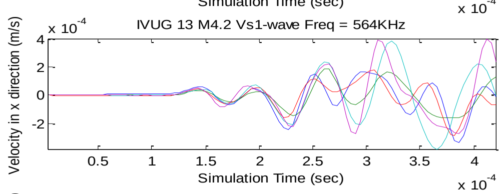
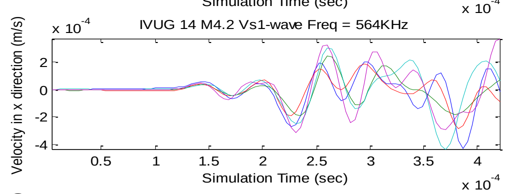
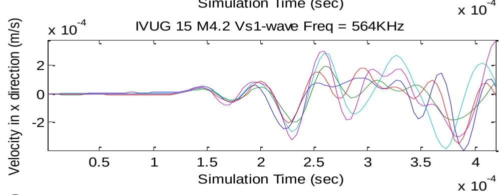
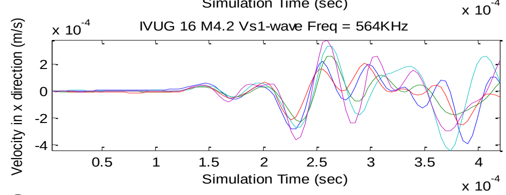
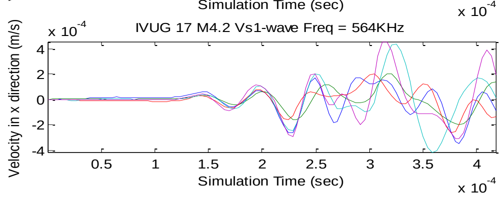
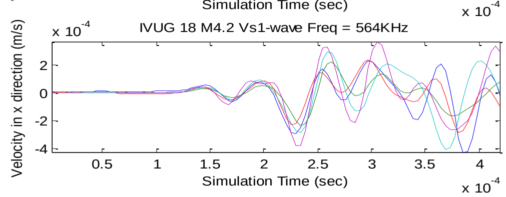
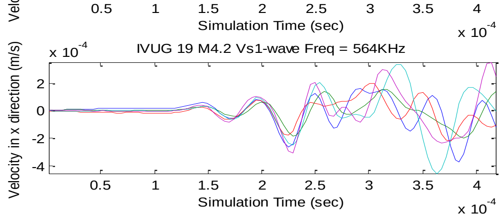
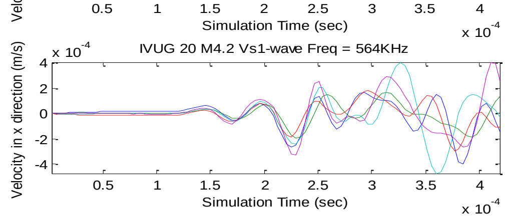

IVUG 37 M4.1 Vs2-wave Freq = 564KHz
Velocity in z direction (m/s)
x 10^-3
Simulation Time (sec)
x 10^-4

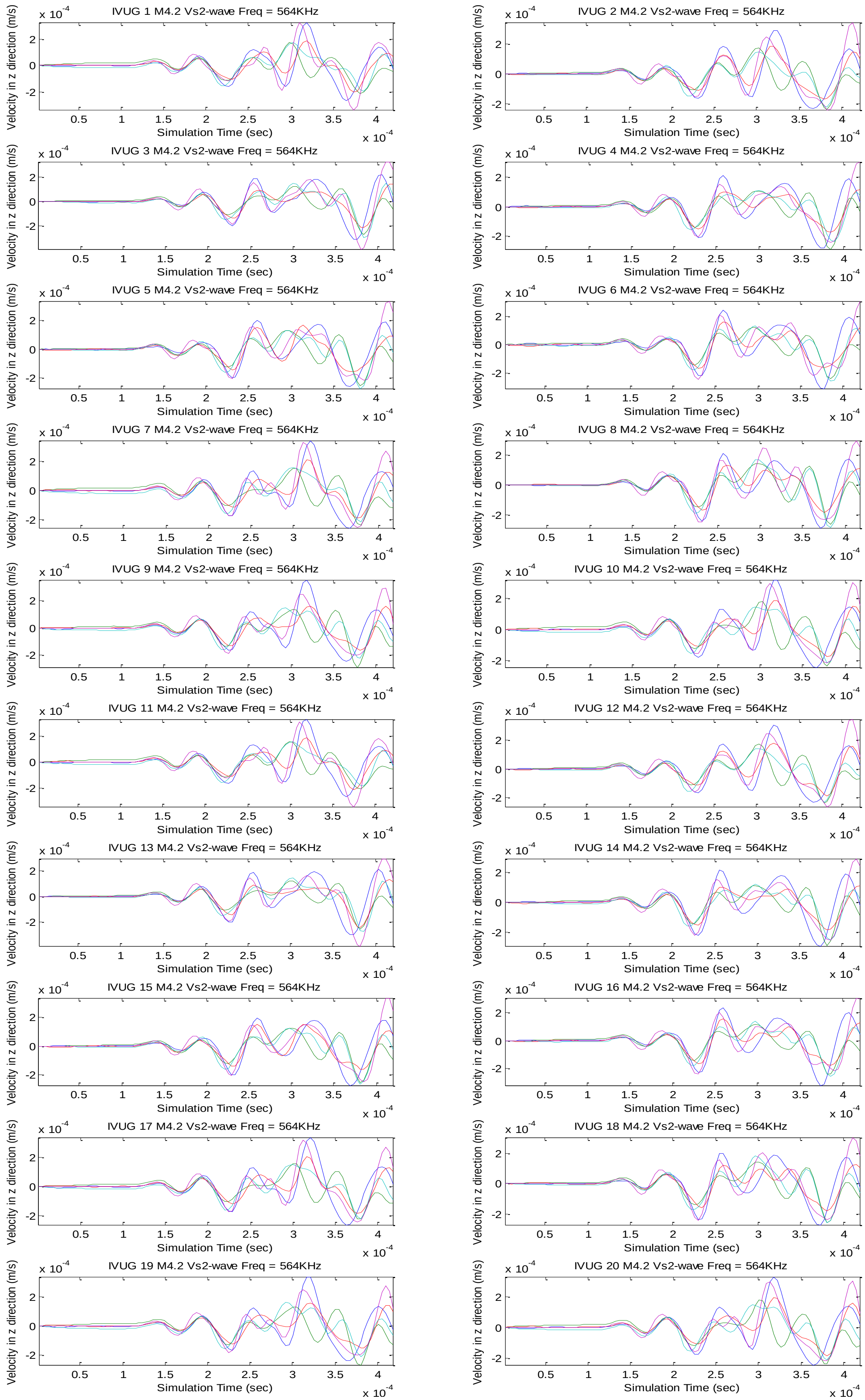

IVUG 38 M4.1 Vs2-wave Freq = 564KHz
Velocity in z direction (m/s)
x 10^-3
Simulation Time (sec)
x 10^-4

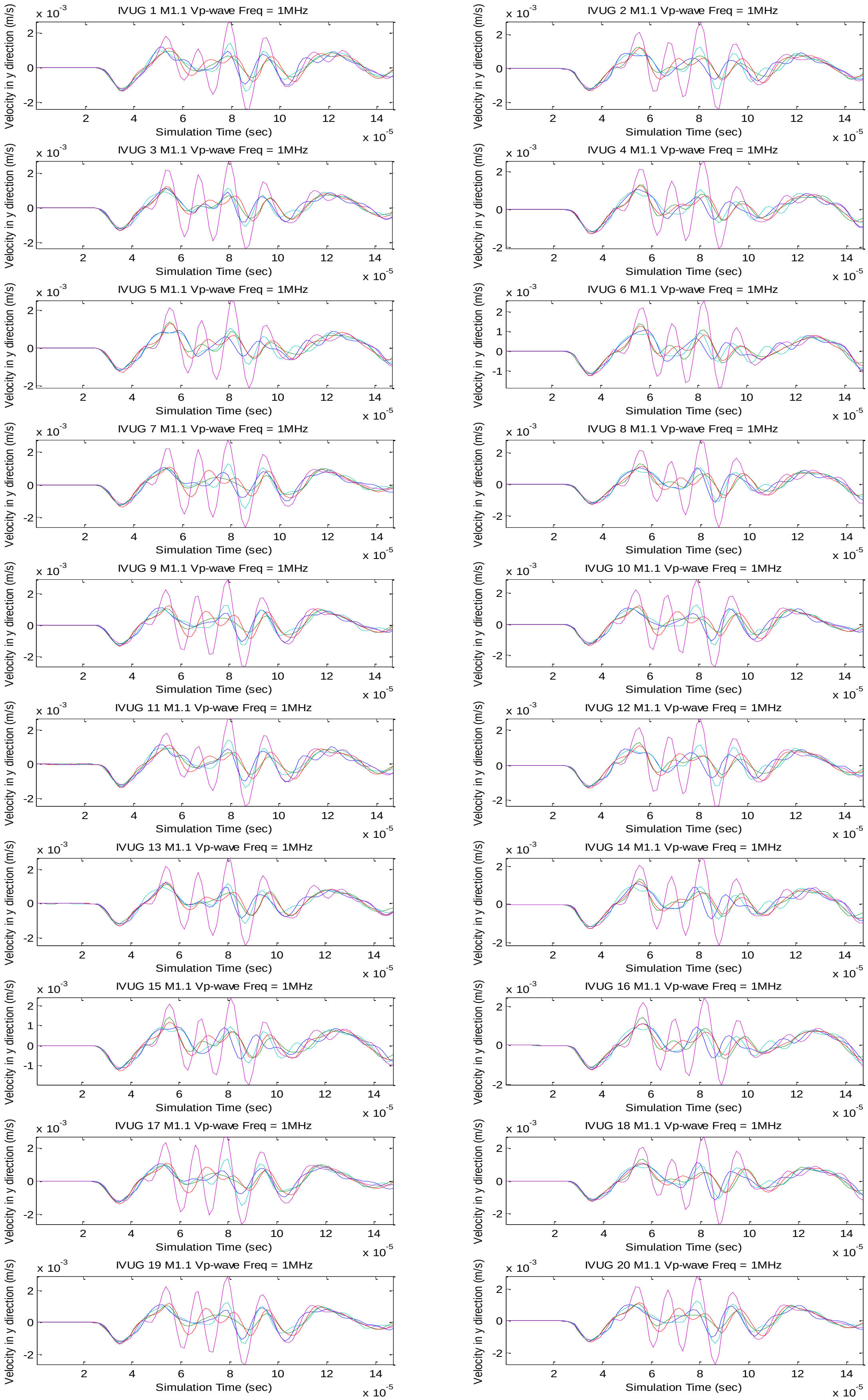

IVUG 39 M4.1 Vs2-wave Freq = 564KHz
Velocity in z direction (m/s)
x 10^-3
Simulation Time (sec)
x 10^-4

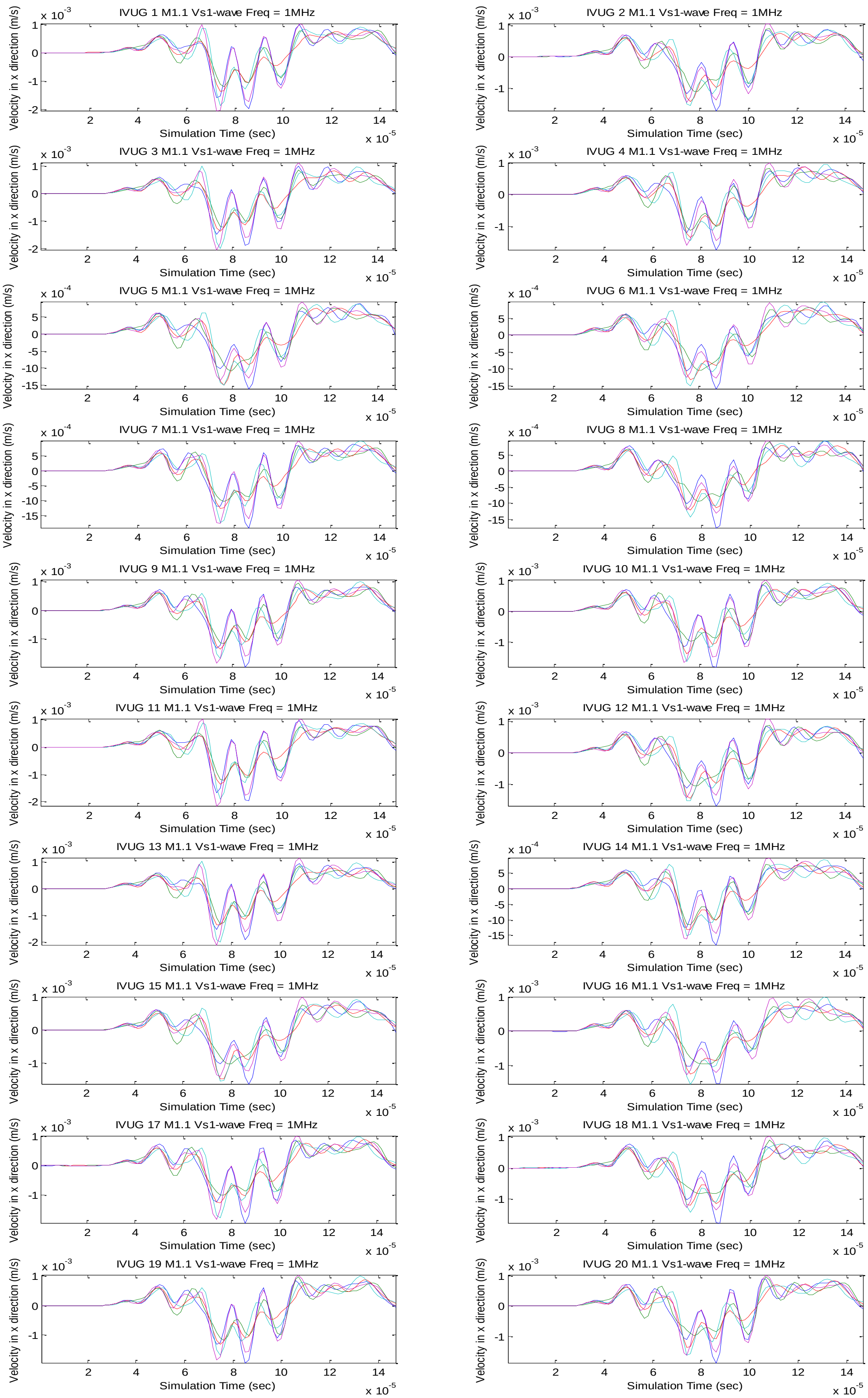

IVUG 40 M4.1 Vs2-wave Freq = 564KHz
Velocity in z direction (m/s)
x 10^-3
Simulation Time (sec)
x 10^-4

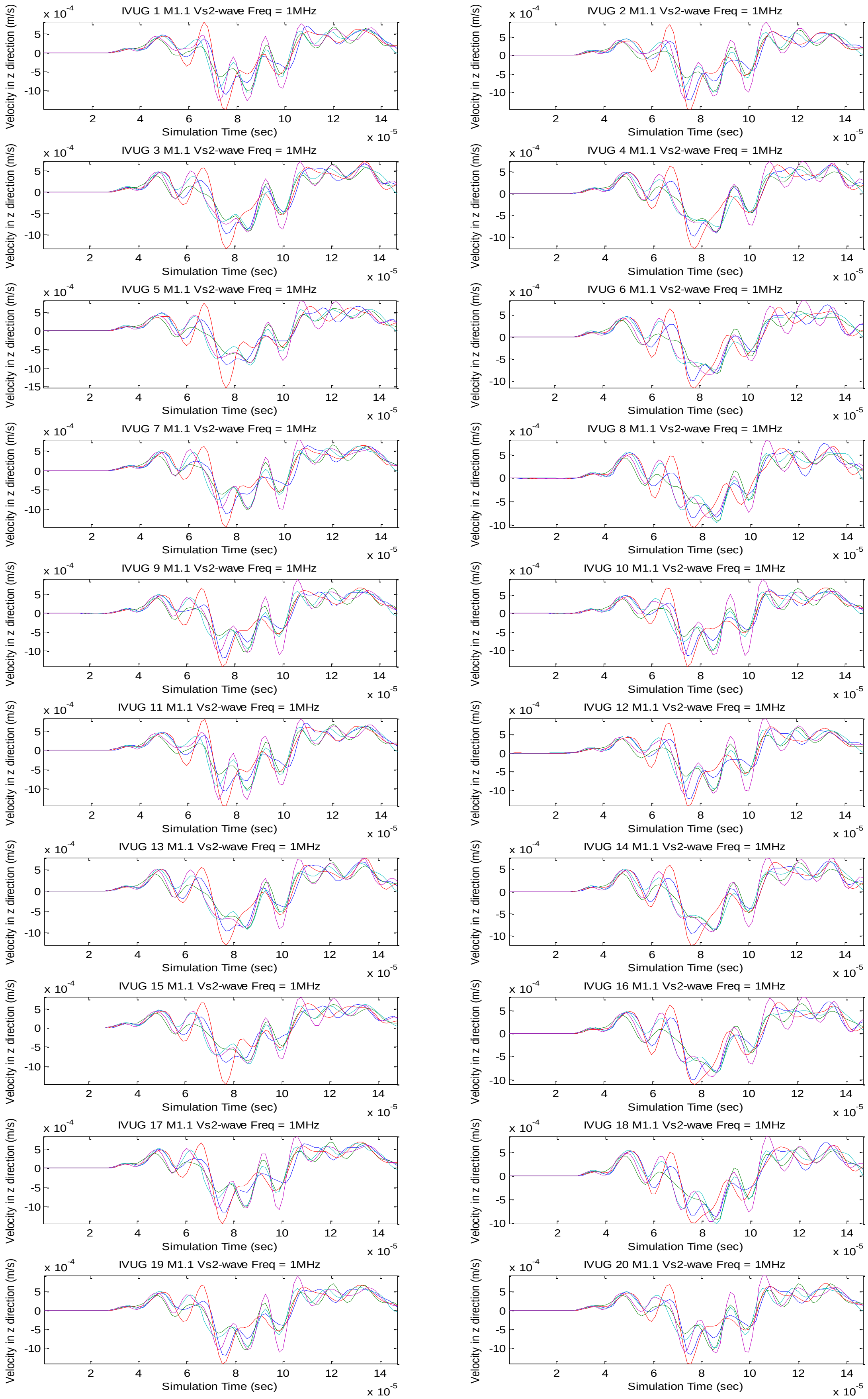

IVUG 1 M4.2 Vp-wave Freq = 564KHz
IVUG 2 M4.2 Vp-wave Freq = 564KHz
IVUG 3 M4.2 Vp-wave Freq = 564KHz
IVUG 4 M4.2 Vp-wave Freq = 564KHz
IVUG 5 M4.2 Vp-wave Freq = 564KHz
IVUG 6 M4.2 Vp-wave Freq = 564KHz
IVUG 7 M4.2 Vp-wave Freq = 564KHz
IVUG 8 M4.2 Vp-wave Freq = 564KHz
IVUG 9 M4.2 Vp-wave Freq = 564KHz
IVUG 10 M4.2 Vp-wave Freq = 564KHz
IVUG 11 M4.2 Vp-wave Freq = 564KHz
IVUG 12 M4.2 Vp-wave Freq = 564KHz
IVUG 13 M4.2 Vp-wave Freq = 564KHz
IVUG 14 M4.2 Vp-wave Freq = 564KHz
IVUG 15 M4.2 Vp-wave Freq = 564KHz
IVUG 16 M4.2 Vp-wave Freq = 564KHz
IVUG 17 M4.2 Vp-wave Freq = 564KHz
IVUG 18 M4.2 Vp-wave Freq = 564KHz
IVUG 19 M4.2 Vp-wave Freq = 564KHz
IVUG 20 M4.2 Vp-wave Freq = 564KHz
Velocity in y direction (m/s)
Simulation Time (sec)
x 10^-4

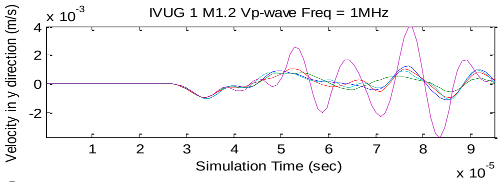

IVUG 1 M4.2 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
Simulation Time (sec)

IVUG 2 M4.2 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
Simulation Time (sec)

IVUG 3 M4.2 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
Simulation Time (sec)

IVUG 4 M4.2 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
Simulation Time (sec)

IVUG 5 M4.2 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
Simulation Time (sec)

IVUG 6 M4.2 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
Simulation Time (sec)

IVUG 7 M4.2 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
Simulation Time (sec)

IVUG 8 M4.2 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
Simulation Time (sec)

IVUG 9 M4.2 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
Simulation Time (sec)

IVUG 10 M4.2 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
Simulation Time (sec)

IVUG 11 M4.2 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
Simulation Time (sec)

IVUG 12 M4.2 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
Simulation Time (sec)

IVUG 13 M4.2 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
Simulation Time (sec)

IVUG 14 M4.2 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
Simulation Time (sec)

IVUG 15 M4.2 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
Simulation Time (sec)

IVUG 16 M4.2 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
Simulation Time (sec)

IVUG 17 M4.2 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
Simulation Time (sec)

IVUG 18 M4.2 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
Simulation Time (sec)

IVUG 19 M4.2 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
Simulation Time (sec)

IVUG 20 M4.2 Vs1-wave Freq = 564KHz
Velocity in x direction (m/s)
Simulation Time (sec)

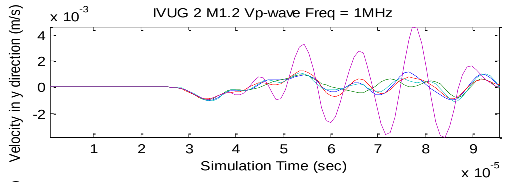

IVUG 1 M4.2 Vs2-wave Freq = 564KHz
IVUG 2 M4.2 Vs2-wave Freq = 564KHz
IVUG 3 M4.2 Vs2-wave Freq = 564KHz
IVUG 4 M4.2 Vs2-wave Freq = 564KHz
IVUG 5 M4.2 Vs2-wave Freq = 564KHz
IVUG 6 M4.2 Vs2-wave Freq = 564KHz
IVUG 7 M4.2 Vs2-wave Freq = 564KHz
IVUG 8 M4.2 Vs2-wave Freq = 564KHz
IVUG 9 M4.2 Vs2-wave Freq = 564KHz
IVUG 10 M4.2 Vs2-wave Freq = 564KHz
IVUG 11 M4.2 Vs2-wave Freq = 564KHz
IVUG 12 M4.2 Vs2-wave Freq = 564KHz
IVUG 13 M4.2 Vs2-wave Freq = 564KHz
IVUG 14 M4.2 Vs2-wave Freq = 564KHz
IVUG 15 M4.2 Vs2-wave Freq = 564KHz
IVUG 16 M4.2 Vs2-wave Freq = 564KHz
IVUG 17 M4.2 Vs2-wave Freq = 564KHz
IVUG 18 M4.2 Vs2-wave Freq = 564KHz
IVUG 19 M4.2 Vs2-wave Freq = 564KHz
IVUG 20 M4.2 Vs2-wave Freq = 564KHz
Velocity in z direction (m/s)
Simulation Time (sec)
x 10^-4

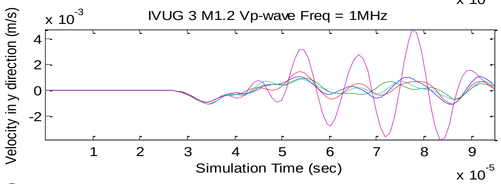

IVUG 1 M1.1 Vp-wave Freq = 1MHz
IVUG 2 M1.1 Vp-wave Freq = 1MHz
IVUG 3 M1.1 Vp-wave Freq = 1MHz
IVUG 4 M1.1 Vp-wave Freq = 1MHz
IVUG 5 M1.1 Vp-wave Freq = 1MHz
IVUG 6 M1.1 Vp-wave Freq = 1MHz
IVUG 7 M1.1 Vp-wave Freq = 1MHz
IVUG 8 M1.1 Vp-wave Freq = 1MHz
IVUG 9 M1.1 Vp-wave Freq = 1MHz
IVUG 10 M1.1 Vp-wave Freq = 1MHz
IVUG 11 M1.1 Vp-wave Freq = 1MHz
IVUG 12 M1.1 Vp-wave Freq = 1MHz
IVUG 13 M1.1 Vp-wave Freq = 1MHz
IVUG 14 M1.1 Vp-wave Freq = 1MHz
IVUG 15 M1.1 Vp-wave Freq = 1MHz
IVUG 16 M1.1 Vp-wave Freq = 1MHz
IVUG 17 M1.1 Vp-wave Freq = 1MHz
IVUG 18 M1.1 Vp-wave Freq = 1MHz
IVUG 19 M1.1 Vp-wave Freq = 1MHz
IVUG 20 M1.1 Vp-wave Freq = 1MHz
Velocity in y direction (m/s)
Simulation Time (sec)
x 10^-3
x 10^-5

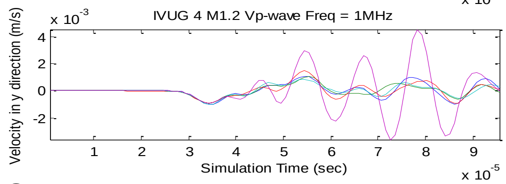

IVUG 1 M1.1 Vs1-wave Freq = 1MHz
IVUG 2 M1.1 Vs1-wave Freq = 1MHz
IVUG 3 M1.1 Vs1-wave Freq = 1MHz
IVUG 4 M1.1 Vs1-wave Freq = 1MHz
IVUG 5 M1.1 Vs1-wave Freq = 1MHz
IVUG 6 M1.1 Vs1-wave Freq = 1MHz
IVUG 7 M1.1 Vs1-wave Freq = 1MHz
IVUG 8 M1.1 Vs1-wave Freq = 1MHz
IVUG 9 M1.1 Vs1-wave Freq = 1MHz
IVUG 10 M1.1 Vs1-wave Freq = 1MHz
IVUG 11 M1.1 Vs1-wave Freq = 1MHz
IVUG 12 M1.1 Vs1-wave Freq = 1MHz
IVUG 13 M1.1 Vs1-wave Freq = 1MHz
IVUG 14 M1.1 Vs1-wave Freq = 1MHz
IVUG 15 M1.1 Vs1-wave Freq = 1MHz
IVUG 16 M1.1 Vs1-wave Freq = 1MHz
IVUG 17 M1.1 Vs1-wave Freq = 1MHz
IVUG 18 M1.1 Vs1-wave Freq = 1MHz
IVUG 19 M1.1 Vs1-wave Freq = 1MHz
IVUG 20 M1.1 Vs1-wave Freq = 1MHz
Velocity in x direction (m/s)
Simulation Time (sec)

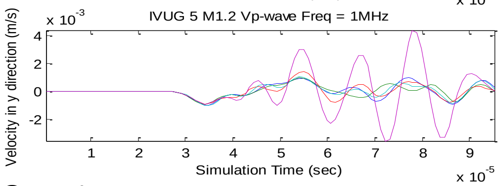

IVUG 1 M1.1 Vs2-wave Freq = 1MHz
IVUG 2 M1.1 Vs2-wave Freq = 1MHz
IVUG 3 M1.1 Vs2-wave Freq = 1MHz
IVUG 4 M1.1 Vs2-wave Freq = 1MHz
IVUG 5 M1.1 Vs2-wave Freq = 1MHz
IVUG 6 M1.1 Vs2-wave Freq = 1MHz
IVUG 7 M1.1 Vs2-wave Freq = 1MHz
IVUG 8 M1.1 Vs2-wave Freq = 1MHz
IVUG 9 M1.1 Vs2-wave Freq = 1MHz
IVUG 10 M1.1 Vs2-wave Freq = 1MHz
IVUG 11 M1.1 Vs2-wave Freq = 1MHz
IVUG 12 M1.1 Vs2-wave Freq = 1MHz
IVUG 13 M1.1 Vs2-wave Freq = 1MHz
IVUG 14 M1.1 Vs2-wave Freq = 1MHz
IVUG 15 M1.1 Vs2-wave Freq = 1MHz
IVUG 16 M1.1 Vs2-wave Freq = 1MHz
IVUG 17 M1.1 Vs2-wave Freq = 1MHz
IVUG 18 M1.1 Vs2-wave Freq = 1MHz
IVUG 19 M1.1 Vs2-wave Freq = 1MHz
IVUG 20 M1.1 Vs2-wave Freq = 1MHz
Velocity in z direction (m/s)
Simulation Time (sec)
x 10^-4
x 10^-5

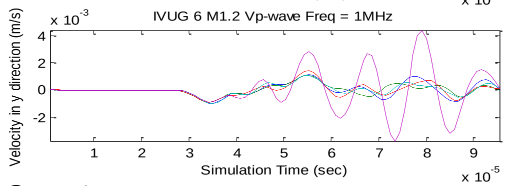

IVUG 1 M1.2 Vp-wave Freq = 1MHz
Velocity in y direction (m/s)
x 10-3
Simulation Time (sec)
x 10-5

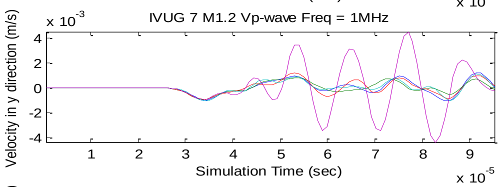

IVUG 2 M1.2 Vp-wave Freq = 1MHz
Velocity in y direction (m/s)
x 10-3
Simulation Time (sec)
x 10-5

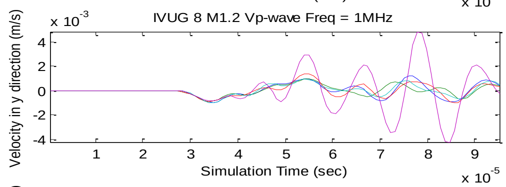

IVUG 3 M1.2 Vp-wave Freq = 1MHz
Velocity in y direction (m/s)
x 10-3
Simulation Time (sec)
x 10-5

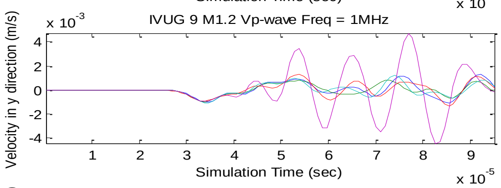

IVUG 4 M1.2 Vp-wave Freq = 1MHz
Velocity in y direction (m/s)
x 10-3
Simulation Time (sec)
x 10-5

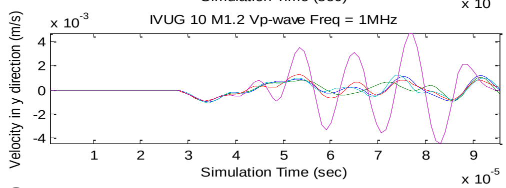

IVUG 5 M1.2 Vp-wave Freq = 1MHz
Velocity in y direction (m/s)
x 10-3
Simulation Time (sec)
x 10-5

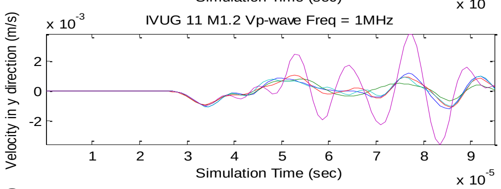

IVUG 6 M1.2 Vp-wave Freq = 1MHz
Velocity in y direction (m/s)
x 10-3
Simulation Time (sec)
x 10-5

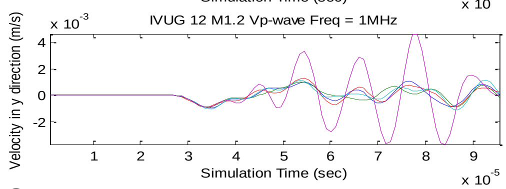

IVUG 7 M1.2 Vp-wave Freq = 1MHz
Velocity in y direction (m/s)
x 10-3
Simulation Time (sec)
x 10-5

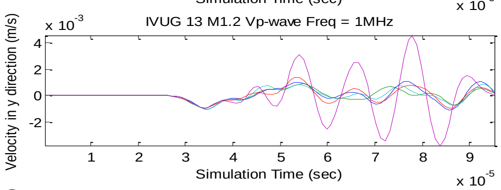

IVUG 8 M1.2 Vp-wave Freq = 1MHz
Velocity in y direction (m/s)
x 10-3
Simulation Time (sec)
x 10-5

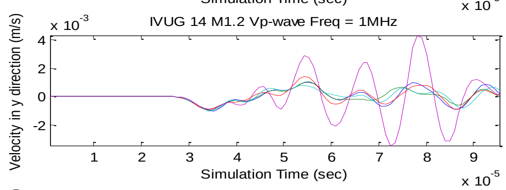

IVUG 9 M1.2 Vp-wave Freq = 1MHz
Velocity in y direction (m/s)
x 10-3
Simulation Time (sec)
x 10-5

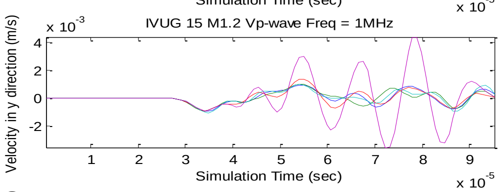

IVUG 10 M1.2 Vp-wave Freq = 1MHz
Velocity in y direction (m/s)
x 10-3
Simulation Time (sec)
x 10-5

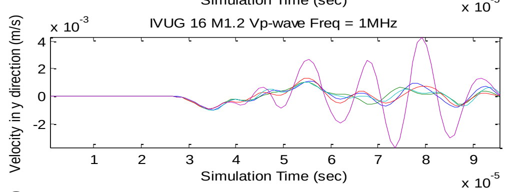

IVUG 11 M1.2 Vp-wave Freq = 1MHz
Velocity in y direction (m/s)
x 10-3
Simulation Time (sec)
x 10-5

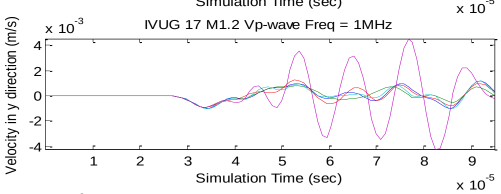

IVUG 12 M1.2 Vp-wave Freq = 1MHz
Velocity in y direction (m/s)
x 10-3
Simulation Time (sec)
x 10-5

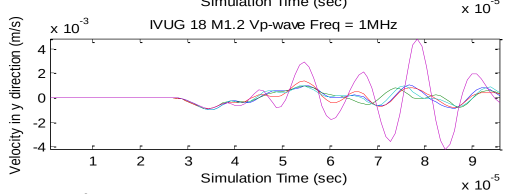

IVUG 13 M1.2 Vp-wave Freq = 1MHz
Velocity in y direction (m/s)
x 10-3
Simulation Time (sec)
x 10-5

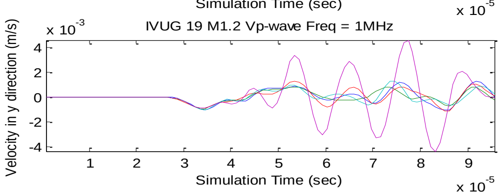

IVUG 14 M1.2 Vp-wave Freq = 1MHz
Velocity in y direction (m/s)
x 10-3
Simulation Time (sec)
x 10-5

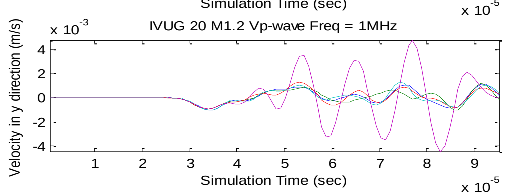

IVUG 15 M1.2 Vp-wave Freq = 1MHz
Velocity in y direction (m/s)
x 10-3
Simulation Time (sec)
x 10-5

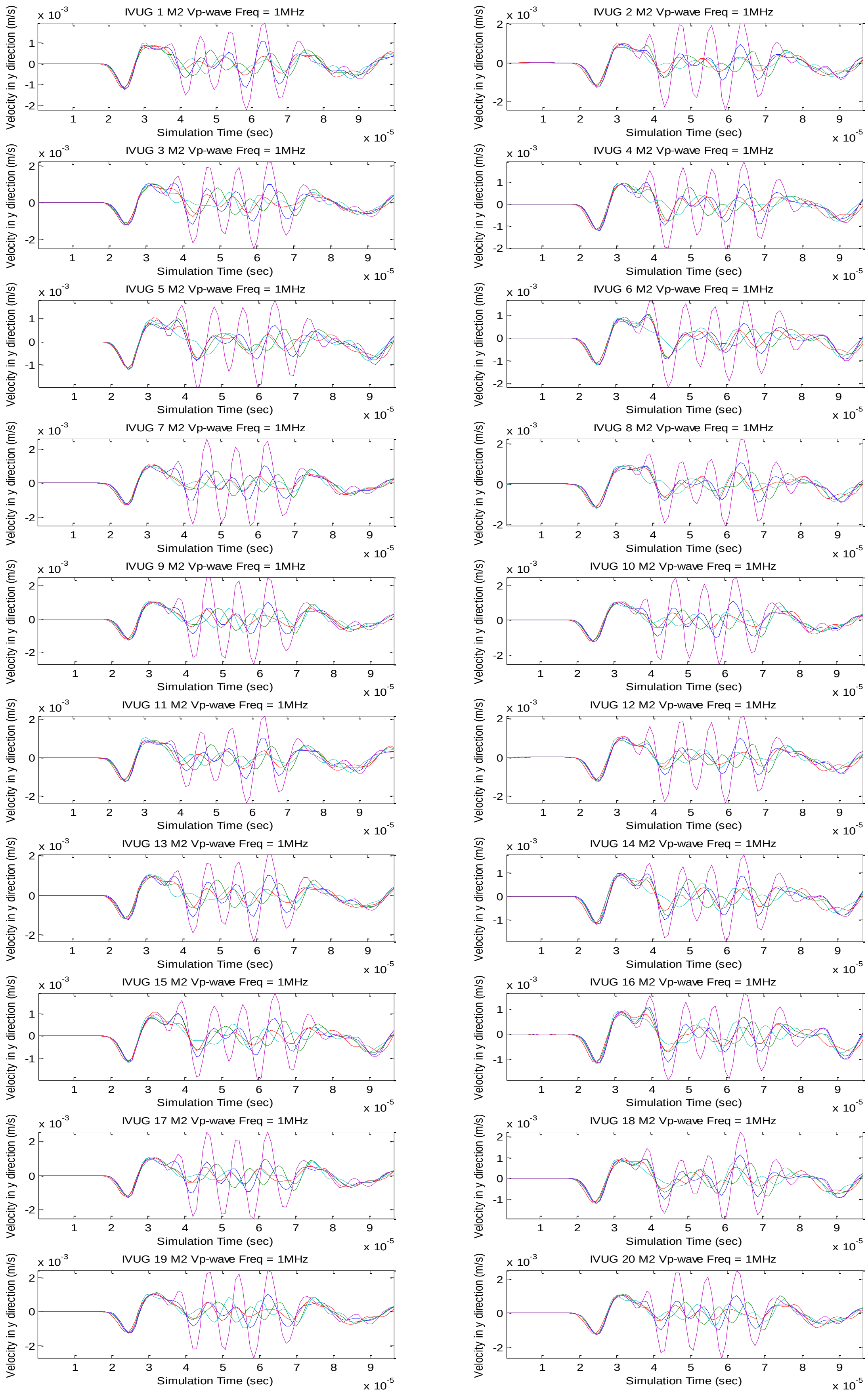

IVUG 16 M1.2 Vp-wave Freq = 1MHz
Velocity in y direction (m/s)
x 10-3
Simulation Time (sec)
x 10-5

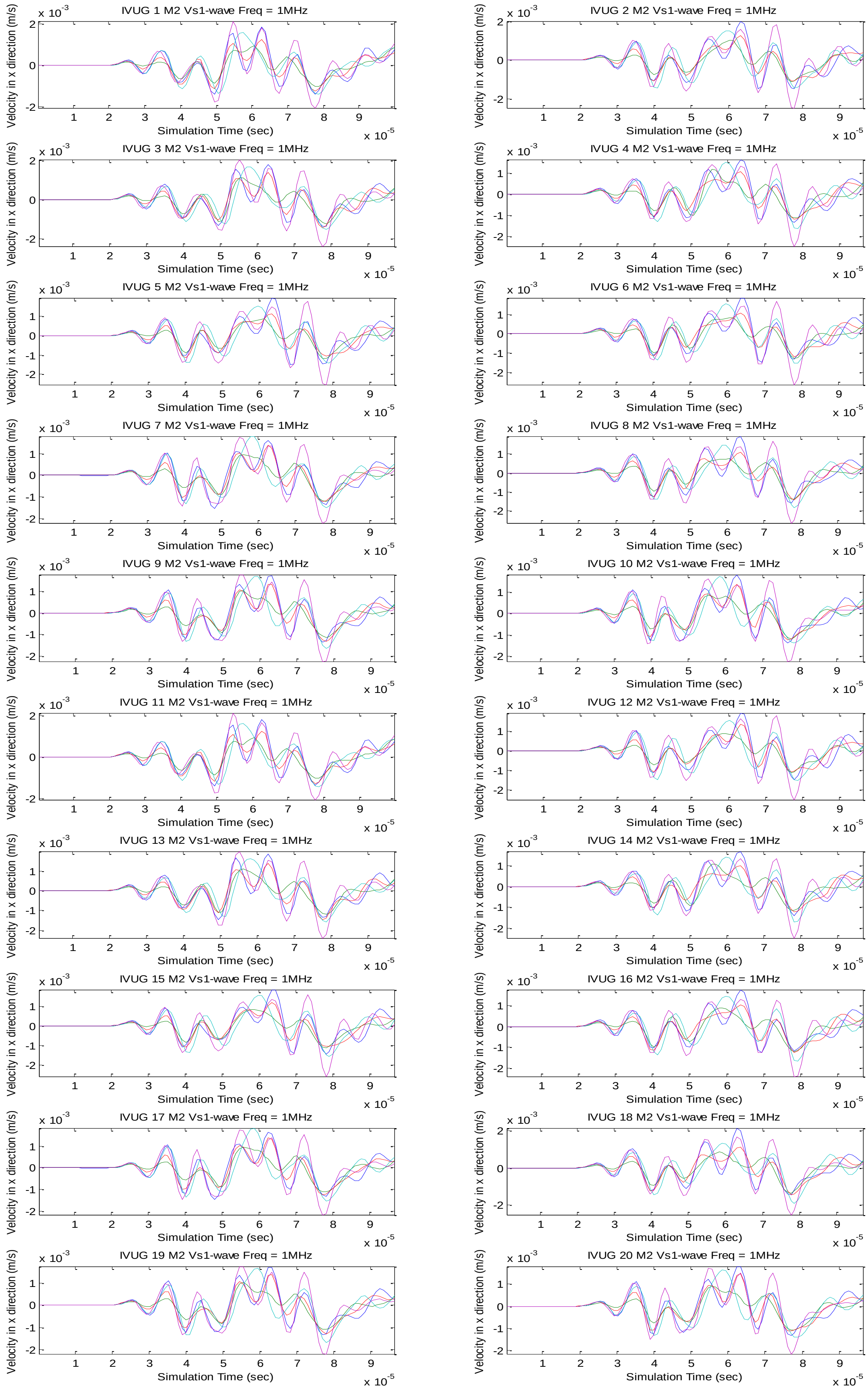

IVUG 17 M1.2 Vp-wave Freq = 1MHz
Velocity in y direction (m/s)
x 10-3
Simulation Time (sec)
x 10-5

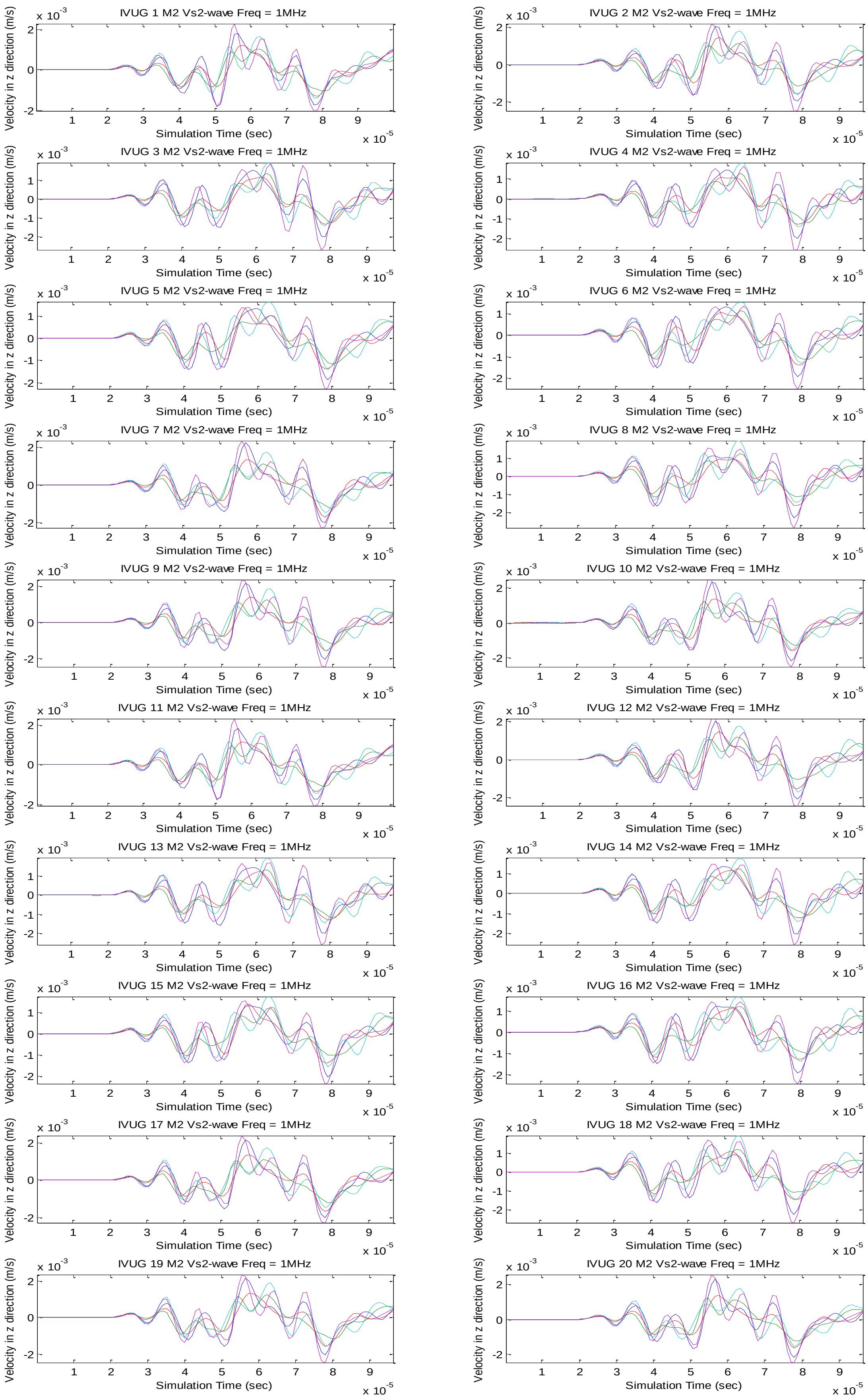

IVUG 18 M1.2 Vp-wave Freq = 1MHz
Velocity in y direction (m/s)
x 10-3
Simulation Time (sec)
x 10-5

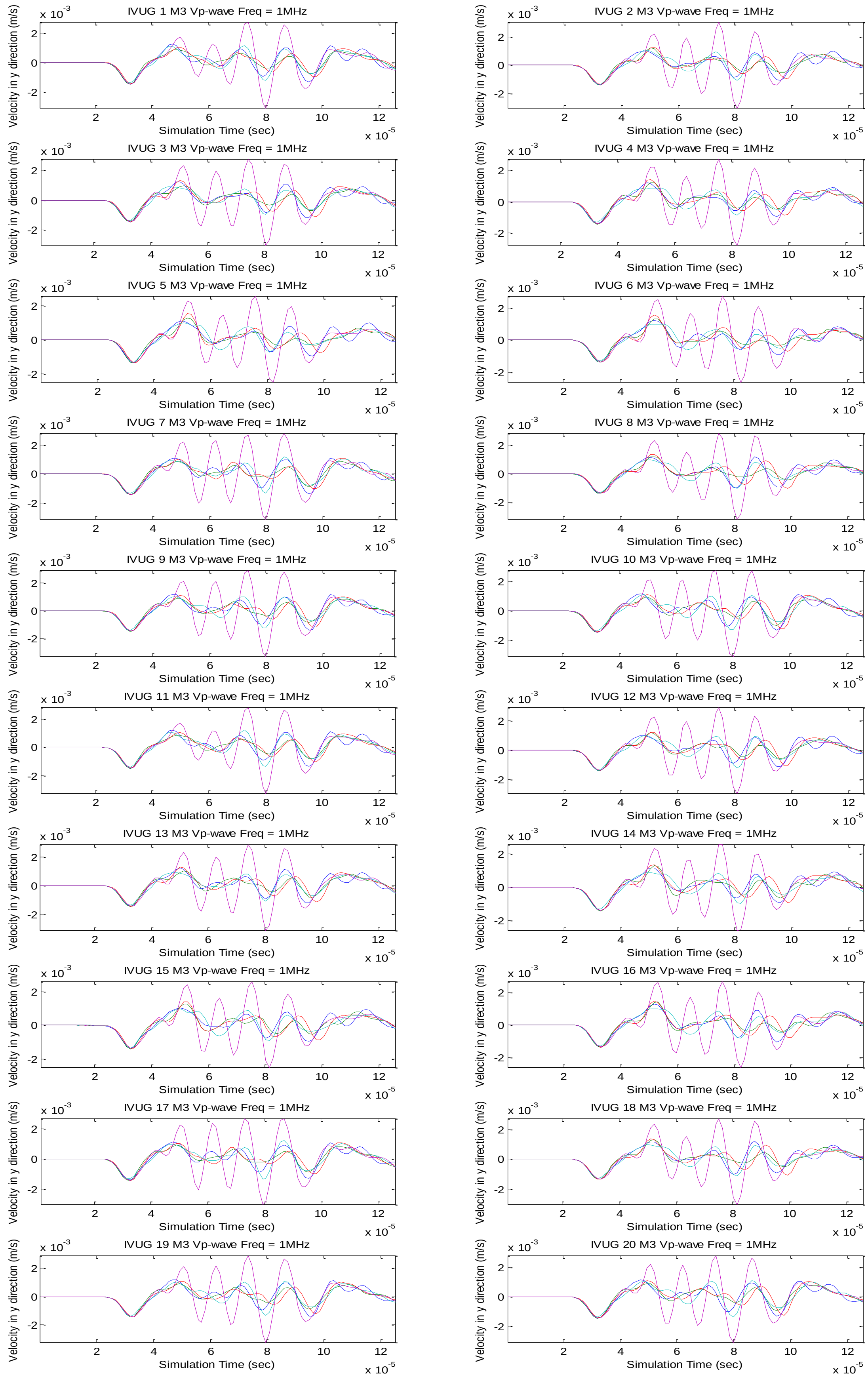

IVUG 19 M1.2 Vp-wave Freq = 1MHz
Velocity in y direction (m/s)
x 10-3
Simulation Time (sec)
x 10-5

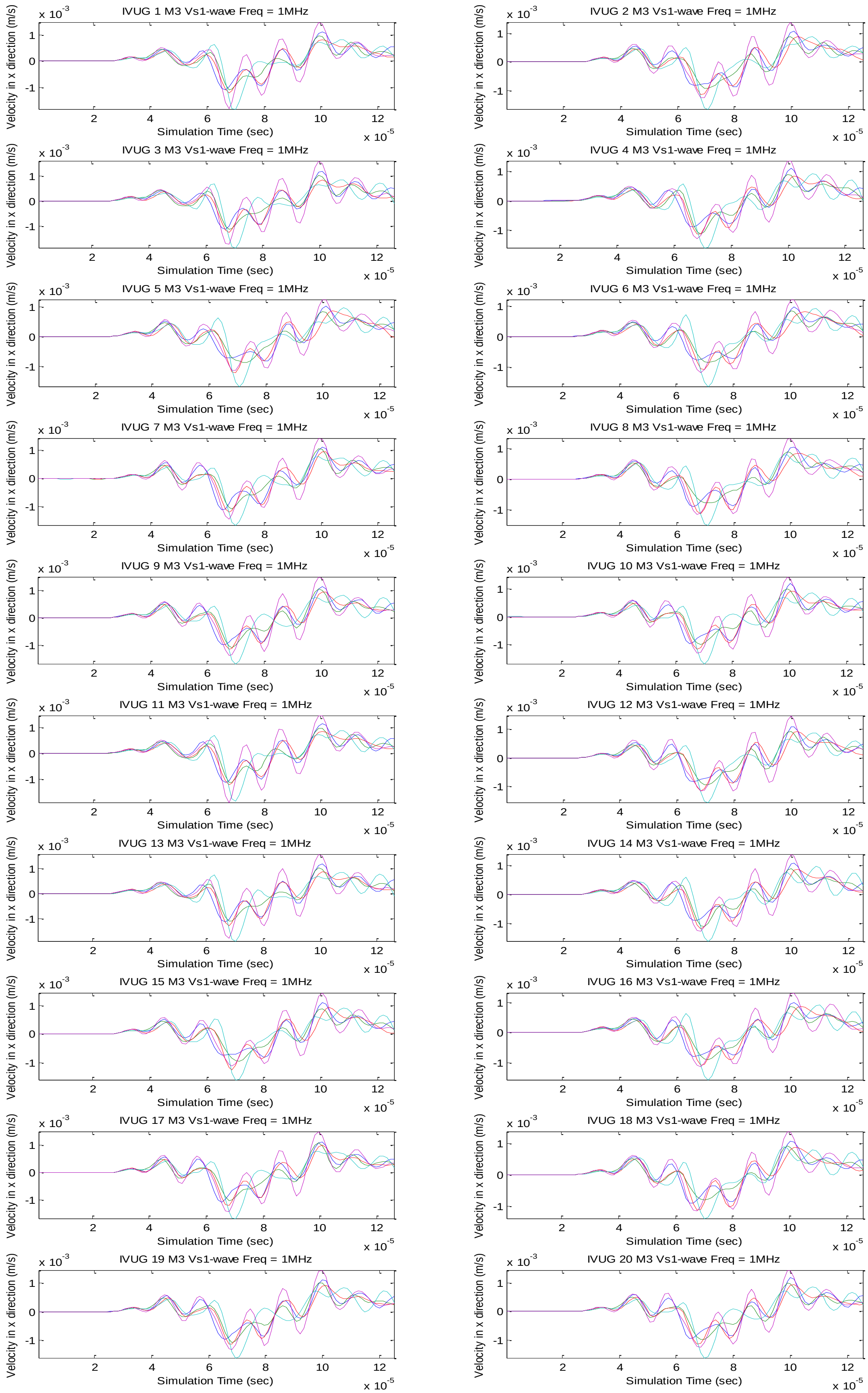

IVUG 20 M1.2 Vp-wave Freq = 1MHz
Velocity in y direction (m/s)
x 10-3
Simulation Time (sec)
x 10-5

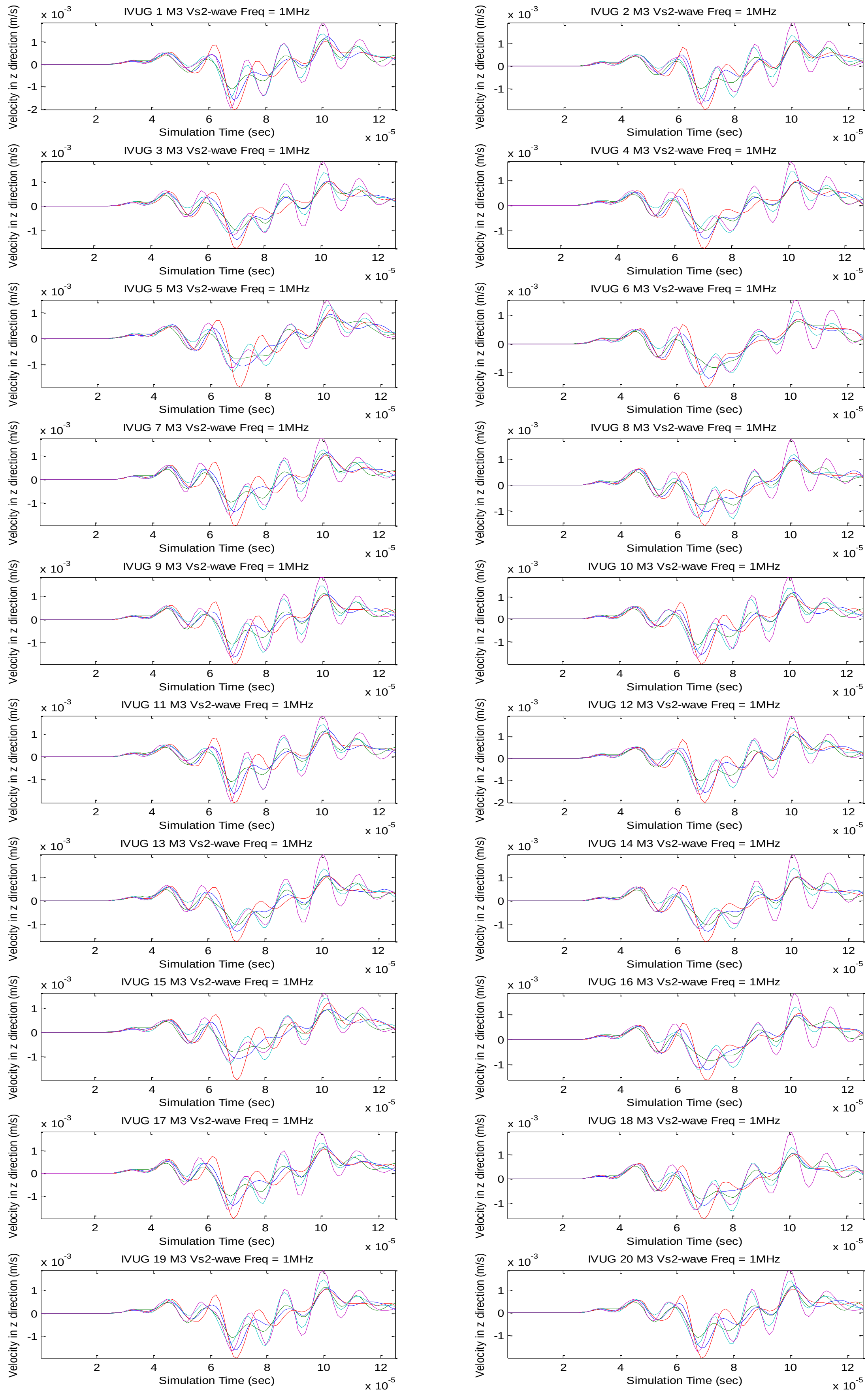

IVUG 1 M2 Vp-wave Freq = 1MHz
IVUG 2 M2 Vp-wave Freq = 1MHz
IVUG 3 M2 Vp-wave Freq = 1MHz
IVUG 4 M2 Vp-wave Freq = 1MHz
IVUG 5 M2 Vp-wave Freq = 1MHz
IVUG 6 M2 Vp-wave Freq = 1MHz
IVUG 7 M2 Vp-wave Freq = 1MHz
IVUG 8 M2 Vp-wave Freq = 1MHz
IVUG 9 M2 Vp-wave Freq = 1MHz
IVUG 10 M2 Vp-wave Freq = 1MHz
IVUG 11 M2 Vp-wave Freq = 1MHz
IVUG 12 M2 Vp-wave Freq = 1MHz
IVUG 13 M2 Vp-wave Freq = 1MHz
IVUG 14 M2 Vp-wave Freq = 1MHz
IVUG 15 M2 Vp-wave Freq = 1MHz
IVUG 16 M2 Vp-wave Freq = 1MHz
IVUG 17 M2 Vp-wave Freq = 1MHz
IVUG 18 M2 Vp-wave Freq = 1MHz
IVUG 19 M2 Vp-wave Freq = 1MHz
IVUG 20 M2 Vp-wave Freq = 1MHz
Velocity in y direction (m/s)
Simulation Time (sec)
x 10^-3
x 10^-5

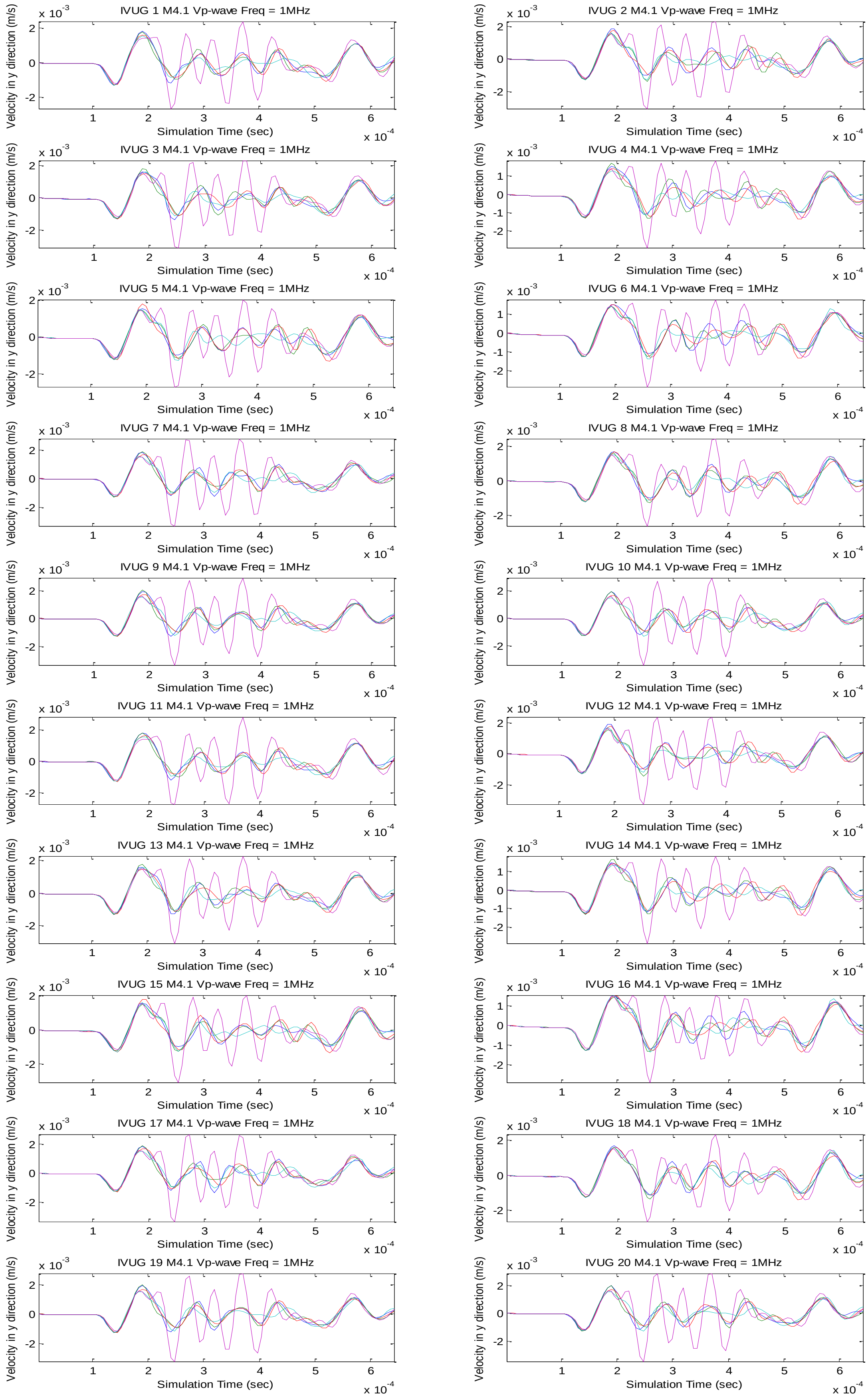

IVUG 1 M2 Vs1-wave Freq = 1MHz
IVUG 2 M2 Vs1-wave Freq = 1MHz
IVUG 3 M2 Vs1-wave Freq = 1MHz
IVUG 4 M2 Vs1-wave Freq = 1MHz
IVUG 5 M2 Vs1-wave Freq = 1MHz
IVUG 6 M2 Vs1-wave Freq = 1MHz
IVUG 7 M2 Vs1-wave Freq = 1MHz
IVUG 8 M2 Vs1-wave Freq = 1MHz
IVUG 9 M2 Vs1-wave Freq = 1MHz
IVUG 10 M2 Vs1-wave Freq = 1MHz
IVUG 11 M2 Vs1-wave Freq = 1MHz
IVUG 12 M2 Vs1-wave Freq = 1MHz
IVUG 13 M2 Vs1-wave Freq = 1MHz
IVUG 14 M2 Vs1-wave Freq = 1MHz
IVUG 15 M2 Vs1-wave Freq = 1MHz
IVUG 16 M2 Vs1-wave Freq = 1MHz
IVUG 17 M2 Vs1-wave Freq = 1MHz
IVUG 18 M2 Vs1-wave Freq = 1MHz
IVUG 19 M2 Vs1-wave Freq = 1MHz
IVUG 20 M2 Vs1-wave Freq = 1MHz
Velocity in x direction (m/s)
Simulation Time (sec)
x 10^-3
x 10^-5

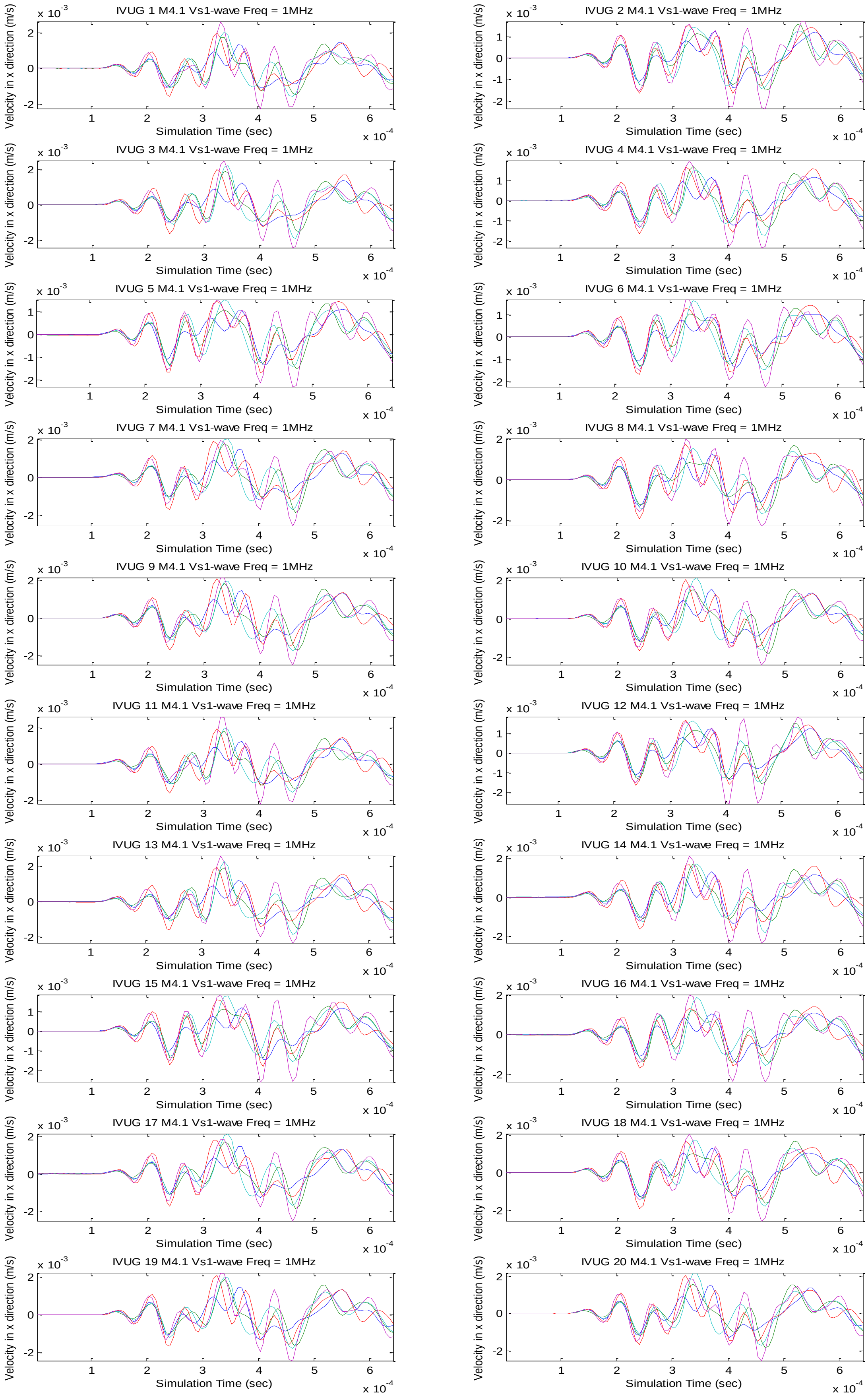

IVUG 1 M2 Vs2-wave Freq = 1MHz
IVUG 2 M2 Vs2-wave Freq = 1MHz
IVUG 3 M2 Vs2-wave Freq = 1MHz
IVUG 4 M2 Vs2-wave Freq = 1MHz
IVUG 5 M2 Vs2-wave Freq = 1MHz
IVUG 6 M2 Vs2-wave Freq = 1MHz
IVUG 7 M2 Vs2-wave Freq = 1MHz
IVUG 8 M2 Vs2-wave Freq = 1MHz
IVUG 9 M2 Vs2-wave Freq = 1MHz
IVUG 10 M2 Vs2-wave Freq = 1MHz
IVUG 11 M2 Vs2-wave Freq = 1MHz
IVUG 12 M2 Vs2-wave Freq = 1MHz
IVUG 13 M2 Vs2-wave Freq = 1MHz
IVUG 14 M2 Vs2-wave Freq = 1MHz
IVUG 15 M2 Vs2-wave Freq = 1MHz
IVUG 16 M2 Vs2-wave Freq = 1MHz
IVUG 17 M2 Vs2-wave Freq = 1MHz
IVUG 18 M2 Vs2-wave Freq = 1MHz
IVUG 19 M2 Vs2-wave Freq = 1MHz
IVUG 20 M2 Vs2-wave Freq = 1MHz
Velocity in z direction (m/s)
Simulation Time (sec)
x 10^-3
x 10^-5

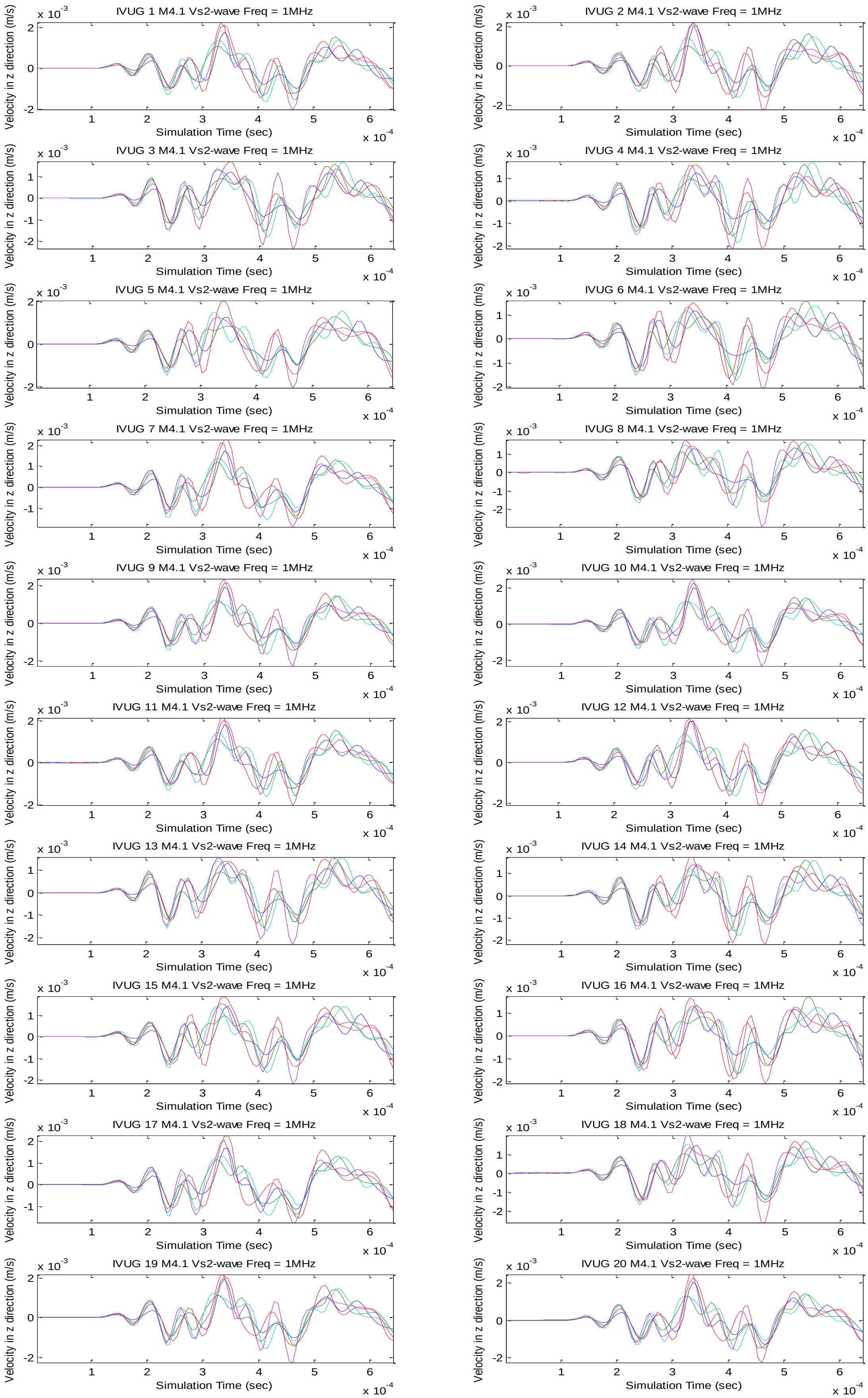

IVUG 1 M3 Vp-wave Freq = 1MHz
IVUG 2 M3 Vp-wave Freq = 1MHz
IVUG 3 M3 Vp-wave Freq = 1MHz
IVUG 4 M3 Vp-wave Freq = 1MHz
IVUG 5 M3 Vp-wave Freq = 1MHz
IVUG 6 M3 Vp-wave Freq = 1MHz
IVUG 7 M3 Vp-wave Freq = 1MHz
IVUG 8 M3 Vp-wave Freq = 1MHz
IVUG 9 M3 Vp-wave Freq = 1MHz
IVUG 10 M3 Vp-wave Freq = 1MHz
IVUG 11 M3 Vp-wave Freq = 1MHz
IVUG 12 M3 Vp-wave Freq = 1MHz
IVUG 13 M3 Vp-wave Freq = 1MHz
IVUG 14 M3 Vp-wave Freq = 1MHz
IVUG 15 M3 Vp-wave Freq = 1MHz
IVUG 16 M3 Vp-wave Freq = 1MHz
IVUG 17 M3 Vp-wave Freq = 1MHz
IVUG 18 M3 Vp-wave Freq = 1MHz
IVUG 19 M3 Vp-wave Freq = 1MHz
IVUG 20 M3 Vp-wave Freq = 1MHz
Velocity in y direction (m/s)
Simulation Time (sec)
x 10^-3
x 10^-5

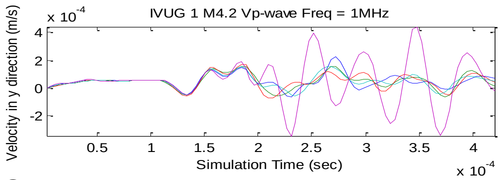
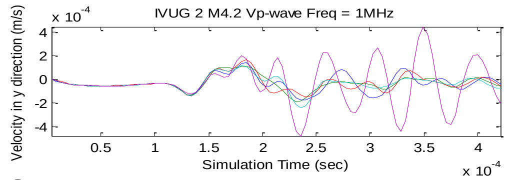
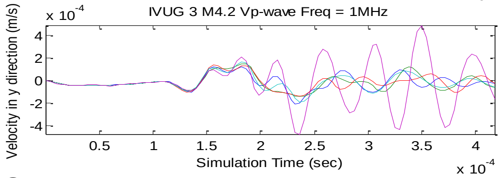
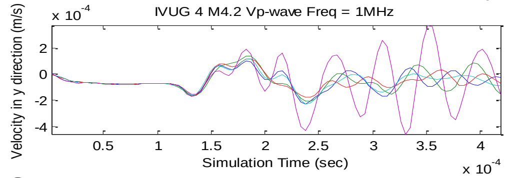
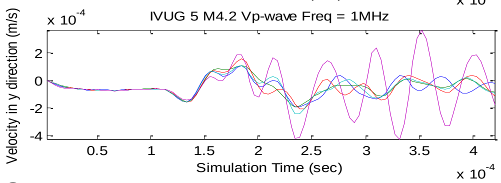
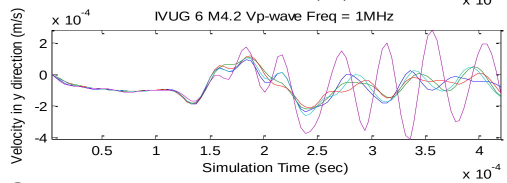
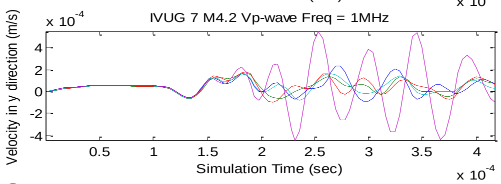
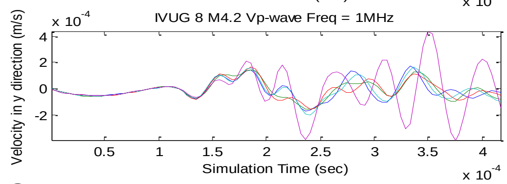
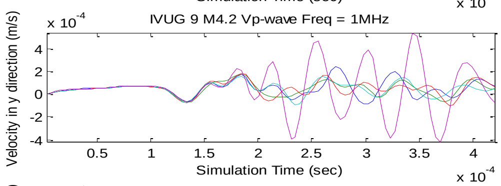
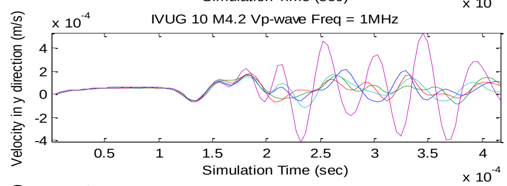
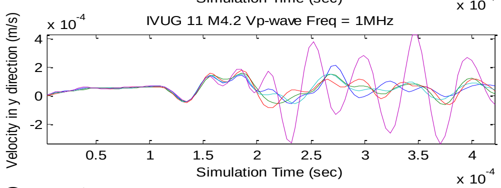
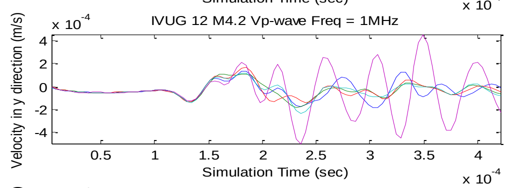
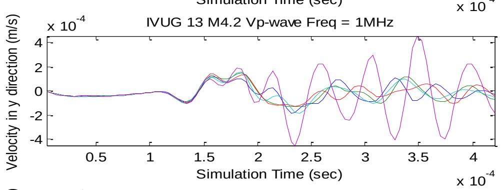
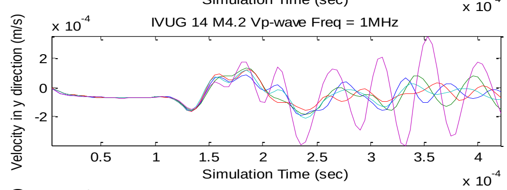
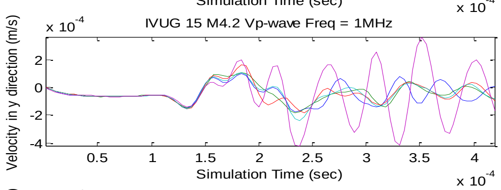
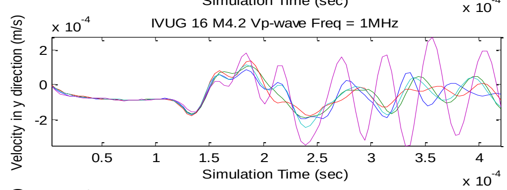
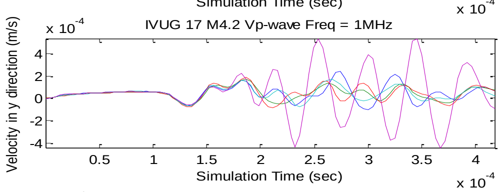
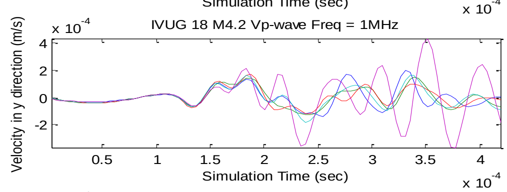
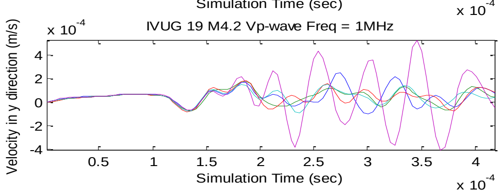
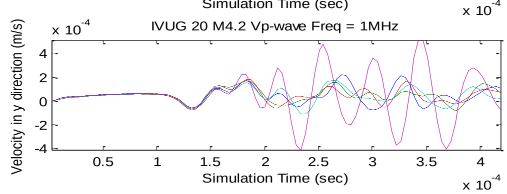

IVUG 1 M3 Vs1-wave Freq = 1MHz
IVUG 2 M3 Vs1-wave Freq = 1MHz
IVUG 3 M3 Vs1-wave Freq = 1MHz
IVUG 4 M3 Vs1-wave Freq = 1MHz
IVUG 5 M3 Vs1-wave Freq = 1MHz
IVUG 6 M3 Vs1-wave Freq = 1MHz
IVUG 7 M3 Vs1-wave Freq = 1MHz
IVUG 8 M3 Vs1-wave Freq = 1MHz
IVUG 9 M3 Vs1-wave Freq = 1MHz
IVUG 10 M3 Vs1-wave Freq = 1MHz
IVUG 11 M3 Vs1-wave Freq = 1MHz
IVUG 12 M3 Vs1-wave Freq = 1MHz
IVUG 13 M3 Vs1-wave Freq = 1MHz
IVUG 14 M3 Vs1-wave Freq = 1MHz
IVUG 15 M3 Vs1-wave Freq = 1MHz
IVUG 16 M3 Vs1-wave Freq = 1MHz
IVUG 17 M3 Vs1-wave Freq = 1MHz
IVUG 18 M3 Vs1-wave Freq = 1MHz
IVUG 19 M3 Vs1-wave Freq = 1MHz
IVUG 20 M3 Vs1-wave Freq = 1MHz
Velocity in x direction (m/s)
x 10-3
Simulation Time (sec)
x 10-5

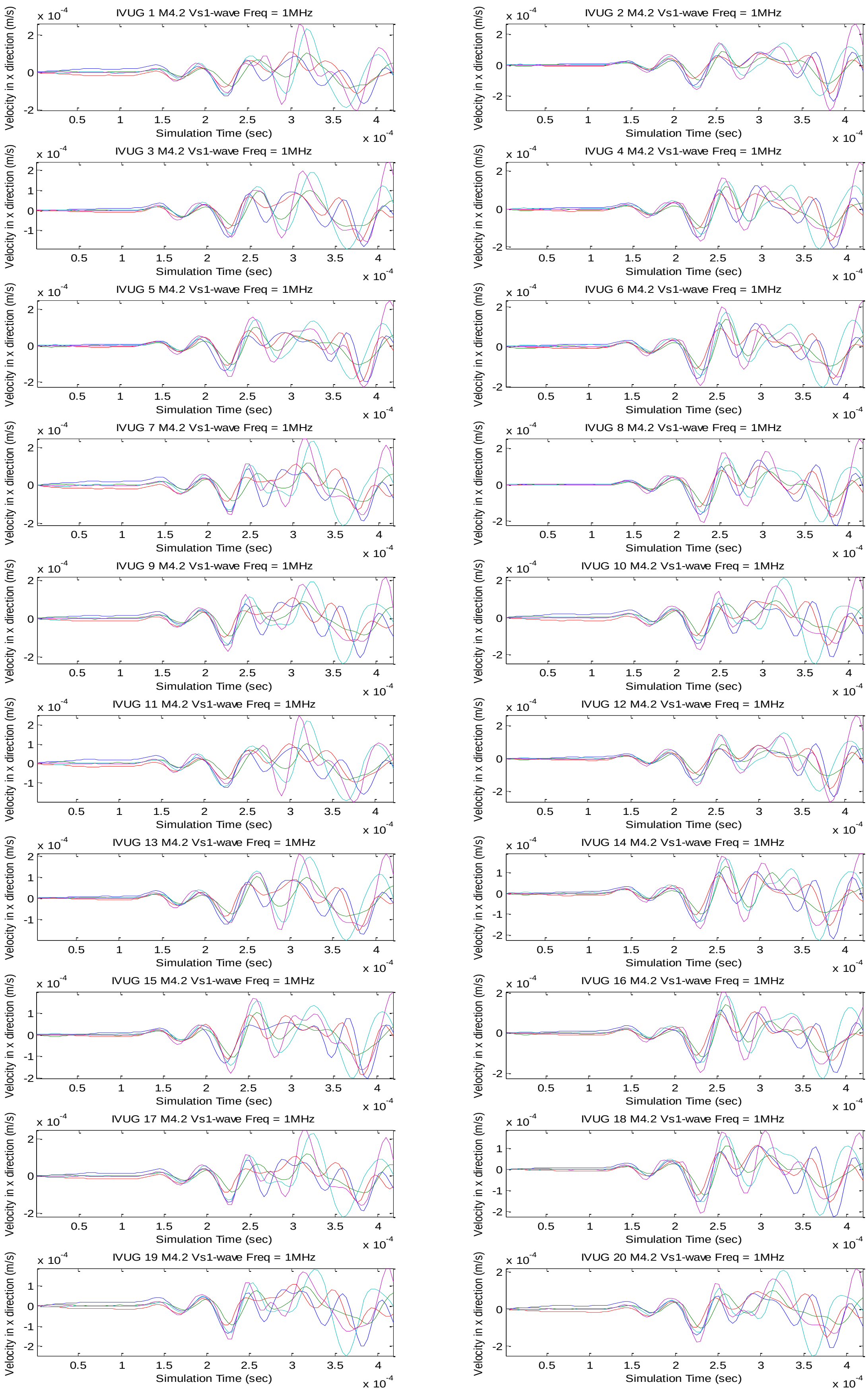

IVUG 1 M3 Vs2-wave Freq = 1MHz
IVUG 2 M3 Vs2-wave Freq = 1MHz
IVUG 3 M3 Vs2-wave Freq = 1MHz
IVUG 4 M3 Vs2-wave Freq = 1MHz
IVUG 5 M3 Vs2-wave Freq = 1MHz
IVUG 6 M3 Vs2-wave Freq = 1MHz
IVUG 7 M3 Vs2-wave Freq = 1MHz
IVUG 8 M3 Vs2-wave Freq = 1MHz
IVUG 9 M3 Vs2-wave Freq = 1MHz
IVUG 10 M3 Vs2-wave Freq = 1MHz
IVUG 11 M3 Vs2-wave Freq = 1MHz
IVUG 12 M3 Vs2-wave Freq = 1MHz
IVUG 13 M3 Vs2-wave Freq = 1MHz
IVUG 14 M3 Vs2-wave Freq = 1MHz
IVUG 15 M3 Vs2-wave Freq = 1MHz
IVUG 16 M3 Vs2-wave Freq = 1MHz
IVUG 17 M3 Vs2-wave Freq = 1MHz
IVUG 18 M3 Vs2-wave Freq = 1MHz
IVUG 19 M3 Vs2-wave Freq = 1MHz
IVUG 20 M3 Vs2-wave Freq = 1MHz
Velocity in z direction (m/s)
x 10^-3
Simulation Time (sec)
x 10^-5

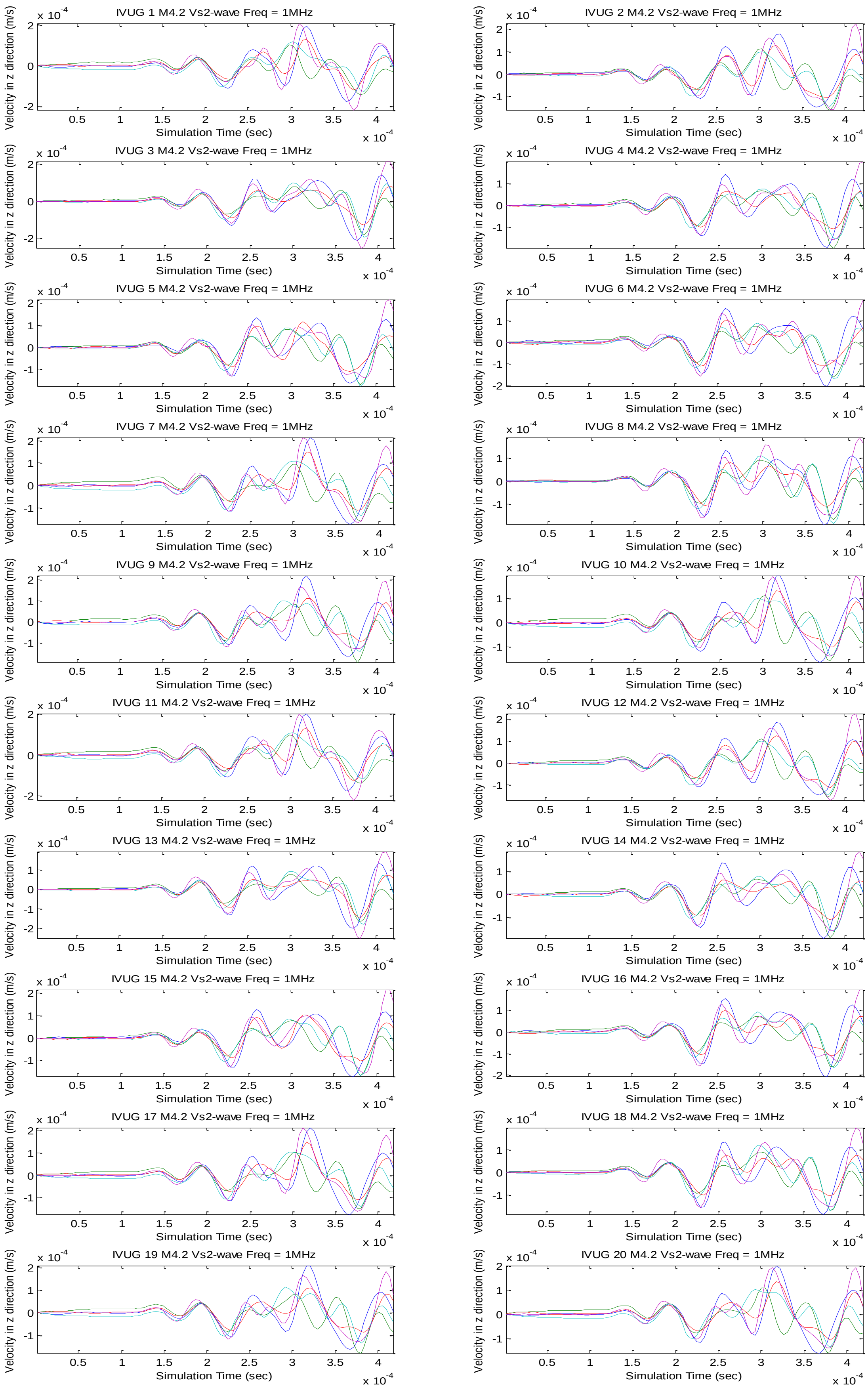

IVUG 1 M4.1 Vp-wave Freq = 1MHz
IVUG 2 M4.1 Vp-wave Freq = 1MHz
IVUG 3 M4.1 Vp-wave Freq = 1MHz
IVUG 4 M4.1 Vp-wave Freq = 1MHz
IVUG 5 M4.1 Vp-wave Freq = 1MHz
IVUG 6 M4.1 Vp-wave Freq = 1MHz
IVUG 7 M4.1 Vp-wave Freq = 1MHz
IVUG 8 M4.1 Vp-wave Freq = 1MHz
IVUG 9 M4.1 Vp-wave Freq = 1MHz
IVUG 10 M4.1 Vp-wave Freq = 1MHz
IVUG 11 M4.1 Vp-wave Freq = 1MHz
IVUG 12 M4.1 Vp-wave Freq = 1MHz
IVUG 13 M4.1 Vp-wave Freq = 1MHz
IVUG 14 M4.1 Vp-wave Freq = 1MHz
IVUG 15 M4.1 Vp-wave Freq = 1MHz
IVUG 16 M4.1 Vp-wave Freq = 1MHz
IVUG 17 M4.1 Vp-wave Freq = 1MHz
IVUG 18 M4.1 Vp-wave Freq = 1MHz
IVUG 19 M4.1 Vp-wave Freq = 1MHz
IVUG 20 M4.1 Vp-wave Freq = 1MHz
Velocity in y direction (m/s)
x 10^-3
Simulation Time (sec)
x 10^-4

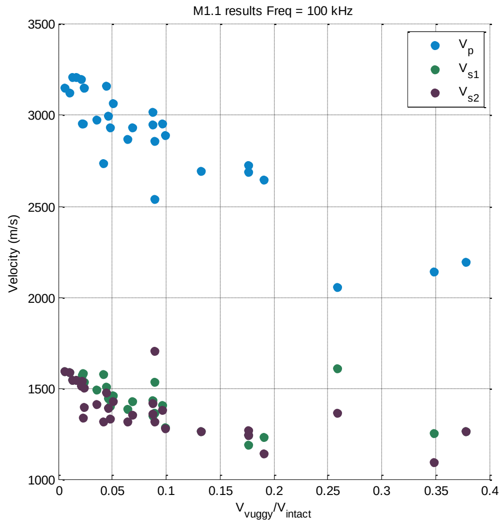

IVUG 1 M4.1 Vs1-wave Freq = 1MHz
IVUG 2 M4.1 Vs1-wave Freq = 1MHz
IVUG 3 M4.1 Vs1-wave Freq = 1MHz
IVUG 4 M4.1 Vs1-wave Freq = 1MHz
IVUG 5 M4.1 Vs1-wave Freq = 1MHz
IVUG 6 M4.1 Vs1-wave Freq = 1MHz
IVUG 7 M4.1 Vs1-wave Freq = 1MHz
IVUG 8 M4.1 Vs1-wave Freq = 1MHz
IVUG 9 M4.1 Vs1-wave Freq = 1MHz
IVUG 10 M4.1 Vs1-wave Freq = 1MHz
IVUG 11 M4.1 Vs1-wave Freq = 1MHz
IVUG 12 M4.1 Vs1-wave Freq = 1MHz
IVUG 13 M4.1 Vs1-wave Freq = 1MHz
IVUG 14 M4.1 Vs1-wave Freq = 1MHz
IVUG 15 M4.1 Vs1-wave Freq = 1MHz
IVUG 16 M4.1 Vs1-wave Freq = 1MHz
IVUG 17 M4.1 Vs1-wave Freq = 1MHz
IVUG 18 M4.1 Vs1-wave Freq = 1MHz
IVUG 19 M4.1 Vs1-wave Freq = 1MHz
IVUG 20 M4.1 Vs1-wave Freq = 1MHz
Velocity in x direction (m/s)
x 10^-3
Simulation Time (sec)
x 10^-4

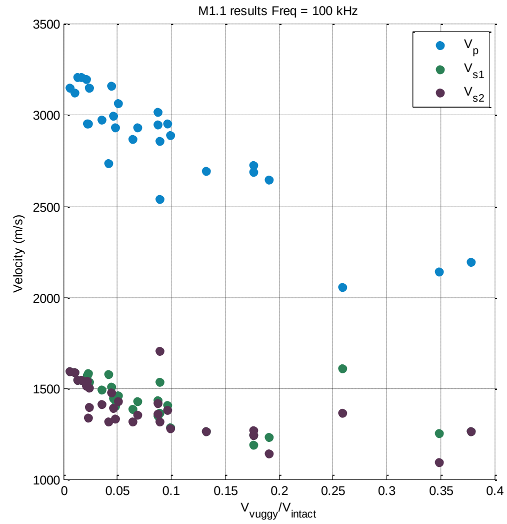

IVUG 1 M4.1 Vs2-wave Freq = 1MHz
IVUG 2 M4.1 Vs2-wave Freq = 1MHz
IVUG 3 M4.1 Vs2-wave Freq = 1MHz
IVUG 4 M4.1 Vs2-wave Freq = 1MHz
IVUG 5 M4.1 Vs2-wave Freq = 1MHz
IVUG 6 M4.1 Vs2-wave Freq = 1MHz
IVUG 7 M4.1 Vs2-wave Freq = 1MHz
IVUG 8 M4.1 Vs2-wave Freq = 1MHz
IVUG 9 M4.1 Vs2-wave Freq = 1MHz
IVUG 10 M4.1 Vs2-wave Freq = 1MHz
IVUG 11 M4.1 Vs2-wave Freq = 1MHz
IVUG 12 M4.1 Vs2-wave Freq = 1MHz
IVUG 13 M4.1 Vs2-wave Freq = 1MHz
IVUG 14 M4.1 Vs2-wave Freq = 1MHz
IVUG 15 M4.1 Vs2-wave Freq = 1MHz
IVUG 16 M4.1 Vs2-wave Freq = 1MHz
IVUG 17 M4.1 Vs2-wave Freq = 1MHz
IVUG 18 M4.1 Vs2-wave Freq = 1MHz
IVUG 19 M4.1 Vs2-wave Freq = 1MHz
IVUG 20 M4.1 Vs2-wave Freq = 1MHz
Velocity in z direction (m/s)
Simulation Time (sec)
x 10^-3
x 10^-4

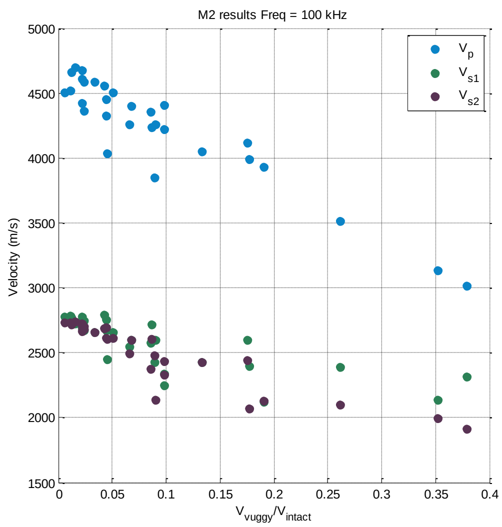

IVUG 1 M4.2 Vp-wave Freq = 1MHz
Velocity in y direction (m/s)
Simulation Time (sec)

IVUG 2 M4.2 Vp-wave Freq = 1MHz
Velocity in y direction (m/s)
Simulation Time (sec)

IVUG 3 M4.2 Vp-wave Freq = 1MHz
Velocity in y direction (m/s)
Simulation Time (sec)

IVUG 4 M4.2 Vp-wave Freq = 1MHz
Velocity in y direction (m/s)
Simulation Time (sec)

IVUG 5 M4.2 Vp-wave Freq = 1MHz
Velocity in y direction (m/s)
Simulation Time (sec)

IVUG 6 M4.2 Vp-wave Freq = 1MHz
Velocity in y direction (m/s)
Simulation Time (sec)

IVUG 7 M4.2 Vp-wave Freq = 1MHz
Velocity in y direction (m/s)
Simulation Time (sec)

IVUG 8 M4.2 Vp-wave Freq = 1MHz
Velocity in y direction (m/s)
Simulation Time (sec)

IVUG 9 M4.2 Vp-wave Freq = 1MHz
Velocity in y direction (m/s)
Simulation Time (sec)

IVUG 10 M4.2 Vp-wave Freq = 1MHz
Velocity in y direction (m/s)
Simulation Time (sec)

IVUG 11 M4.2 Vp-wave Freq = 1MHz
Velocity in y direction (m/s)
Simulation Time (sec)

IVUG 12 M4.2 Vp-wave Freq = 1MHz
Velocity in y direction (m/s)
Simulation Time (sec)

IVUG 13 M4.2 Vp-wave Freq = 1MHz
Velocity in y direction (m/s)
Simulation Time (sec)

IVUG 14 M4.2 Vp-wave Freq = 1MHz
Velocity in y direction (m/s)
Simulation Time (sec)

IVUG 15 M4.2 Vp-wave Freq = 1MHz
Velocity in y direction (m/s)
Simulation Time (sec)

IVUG 16 M4.2 Vp-wave Freq = 1MHz
Velocity in y direction (m/s)
Simulation Time (sec)

IVUG 17 M4.2 Vp-wave Freq = 1MHz
Velocity in y direction (m/s)
Simulation Time (sec)

IVUG 18 M4.2 Vp-wave Freq = 1MHz
Velocity in y direction (m/s)
Simulation Time (sec)

IVUG 19 M4.2 Vp-wave Freq = 1MHz
Velocity in y direction (m/s)
Simulation Time (sec)

IVUG 20 M4.2 Vp-wave Freq = 1MHz
Velocity in y direction (m/s)
Simulation Time (sec)

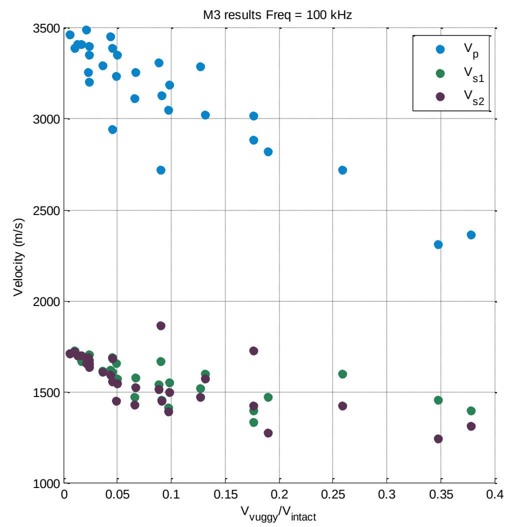

IVUG 1 M4.2 Vs1-wave Freq = 1MHz
IVUG 2 M4.2 Vs1-wave Freq = 1MHz
IVUG 3 M4.2 Vs1-wave Freq = 1MHz
IVUG 4 M4.2 Vs1-wave Freq = 1MHz
IVUG 5 M4.2 Vs1-wave Freq = 1MHz
IVUG 6 M4.2 Vs1-wave Freq = 1MHz
IVUG 7 M4.2 Vs1-wave Freq = 1MHz
IVUG 8 M4.2 Vs1-wave Freq = 1MHz
IVUG 9 M4.2 Vs1-wave Freq = 1MHz
IVUG 10 M4.2 Vs1-wave Freq = 1MHz
IVUG 11 M4.2 Vs1-wave Freq = 1MHz
IVUG 12 M4.2 Vs1-wave Freq = 1MHz
IVUG 13 M4.2 Vs1-wave Freq = 1MHz
IVUG 14 M4.2 Vs1-wave Freq = 1MHz
IVUG 15 M4.2 Vs1-wave Freq = 1MHz
IVUG 16 M4.2 Vs1-wave Freq = 1MHz
IVUG 17 M4.2 Vs1-wave Freq = 1MHz
IVUG 18 M4.2 Vs1-wave Freq = 1MHz
IVUG 19 M4.2 Vs1-wave Freq = 1MHz
IVUG 20 M4.2 Vs1-wave Freq = 1MHz
Velocity in x direction (m/s)
x 10^-4
Simulation Time (sec)
x 10^-4

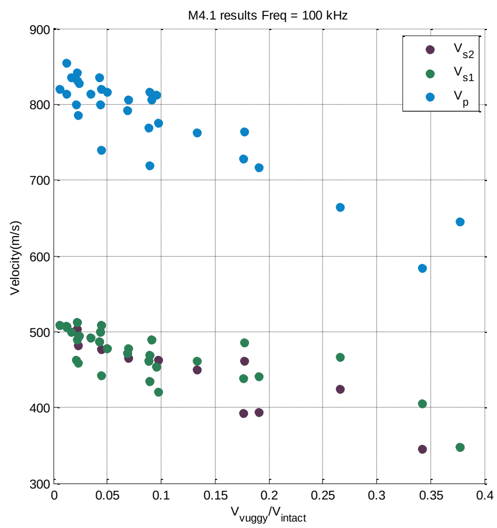

IVUG 1 M4.2 Vs2-wave Freq = 1MHz
IVUG 2 M4.2 Vs2-wave Freq = 1MHz
IVUG 3 M4.2 Vs2-wave Freq = 1MHz
IVUG 4 M4.2 Vs2-wave Freq = 1MHz
IVUG 5 M4.2 Vs2-wave Freq = 1MHz
IVUG 6 M4.2 Vs2-wave Freq = 1MHz
IVUG 7 M4.2 Vs2-wave Freq = 1MHz
IVUG 8 M4.2 Vs2-wave Freq = 1MHz
IVUG 9 M4.2 Vs2-wave Freq = 1MHz
IVUG 10 M4.2 Vs2-wave Freq = 1MHz
IVUG 11 M4.2 Vs2-wave Freq = 1MHz
IVUG 12 M4.2 Vs2-wave Freq = 1MHz
IVUG 13 M4.2 Vs2-wave Freq = 1MHz
IVUG 14 M4.2 Vs2-wave Freq = 1MHz
IVUG 15 M4.2 Vs2-wave Freq = 1MHz
IVUG 16 M4.2 Vs2-wave Freq = 1MHz
IVUG 17 M4.2 Vs2-wave Freq = 1MHz
IVUG 18 M4.2 Vs2-wave Freq = 1MHz
IVUG 19 M4.2 Vs2-wave Freq = 1MHz
IVUG 20 M4.2 Vs2-wave Freq = 1MHz
Velocity in z direction (m/s)
Simulation Time (sec)
x 10^-4

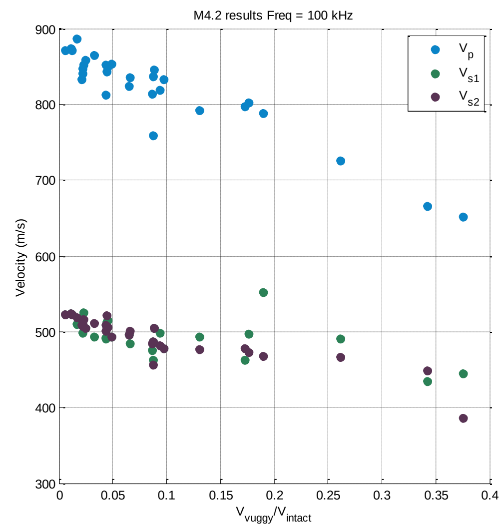

M1.1 results Freq = 100 kHz
Velocity (m/s)
V_{vuggy}/V_{intact}
V_p
V_{s1}
V_{s2}

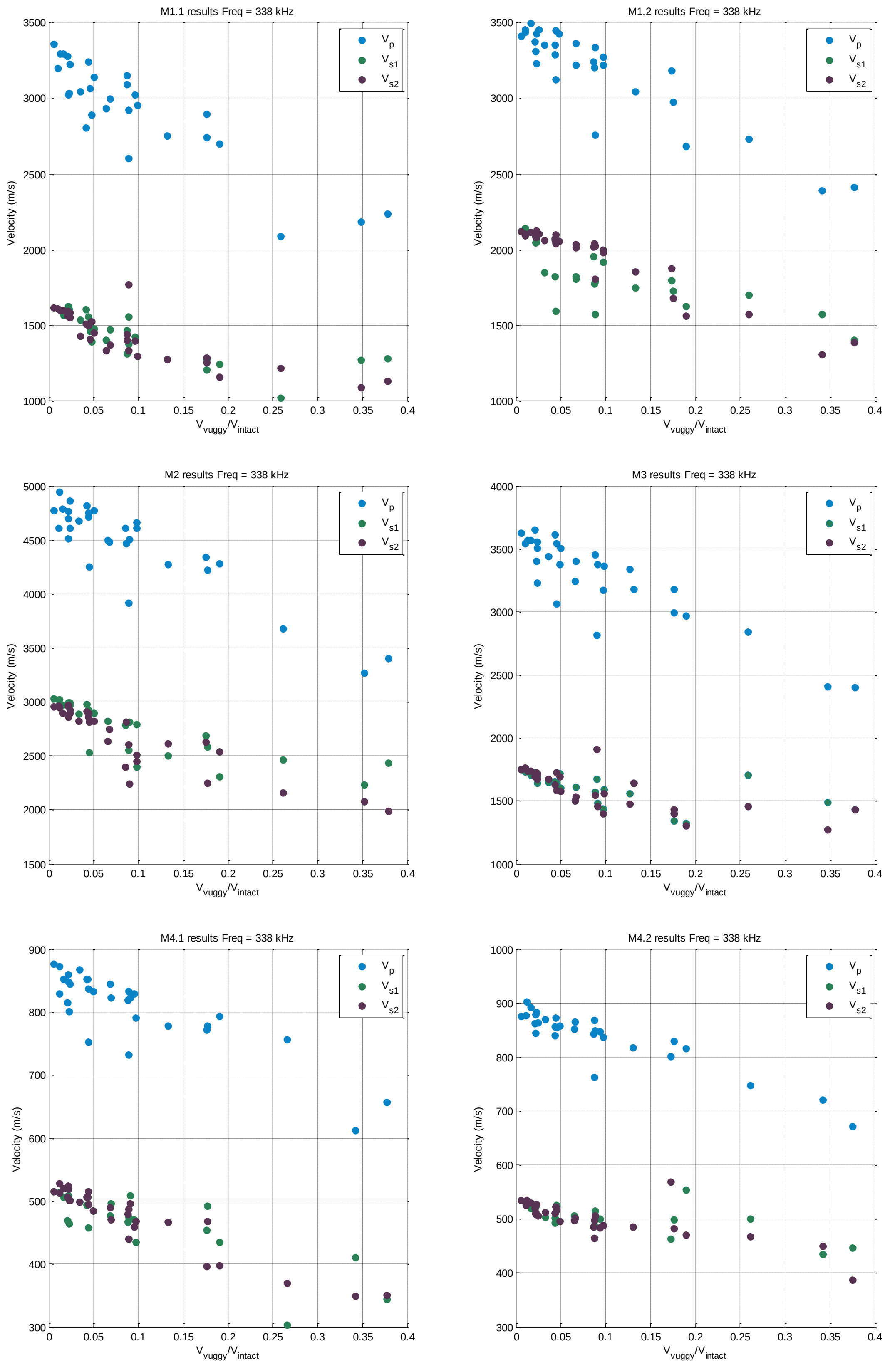

M1.1 results Freq = 100 kHz
Velocity (m/s)
V_{vuggy}/V_{intact}
V_p
V_{s1}
V_{s2}


M2 results Freq = 100 kHz
Velocity (m/s)
V_{vuggy}/V_{intact}
V_p
V_{s1}
V_{s2}

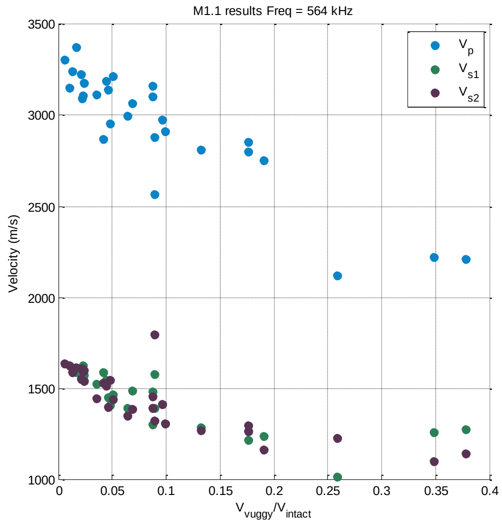

M3 results Freq = 100 kHz
Velocity (m/s)
V_{vuggy}/V_{intact}
V_p
V_{s1}
V_{s2}

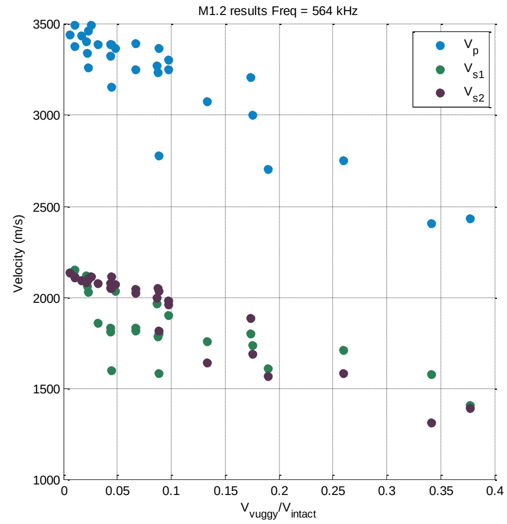

M4.1 results Freq = 100 kHz
Velocity(m/s)
V_{vuggy}/V_{intact}
V_{s2}
V_{s1}
V_p

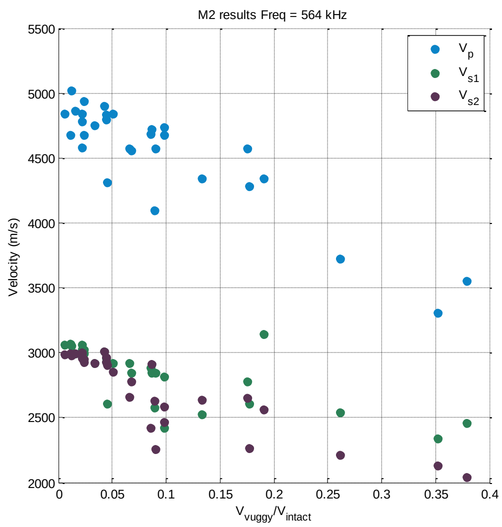

M4.2 results Freq = 100 kHz
Velocity (m/s)
V_{vuggy}/V_{intact}
V_p
V_{s1}
V_{s2}

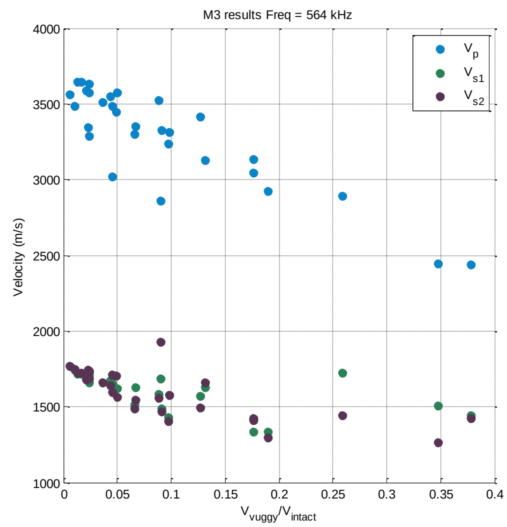


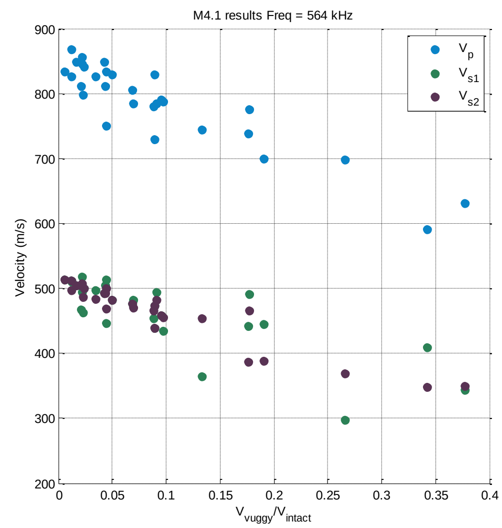
M1.1 results Freq = 338 kHz
M1.2 results Freq = 338 kHz
M2 results Freq = 338 kHz
M3 results Freq = 338 kHz
M4.1 results Freq = 338 kHz
M4.2 results Freq = 338 kHz
Velocity (m/s)
V_{vuggy}/V_{intact}
V_p
V_{s1}
V_{s2}
0 0.05 0.1 0.15 0.2 0.25 0.3 0.35 0.4
1000 1500 2000 2500 3000 3500
1500 2000 2500 3000 3500 4000 4500 5000
1000 1500 2000 2500 3000 3500 4000
300 400 500 600 700 800 900
300 400 500 600 700 800 900 1000

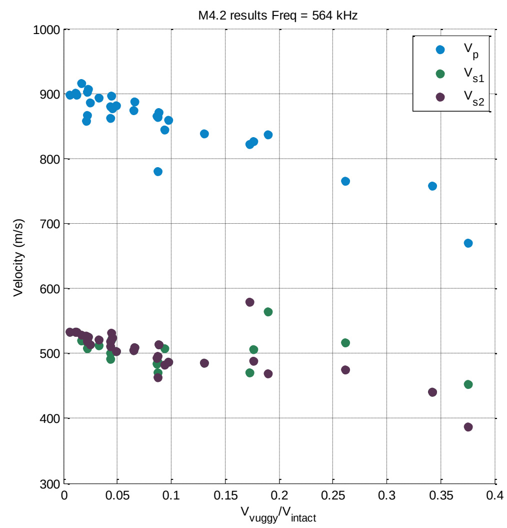

M1.1 results Freq = 564 kHz
Velocity (m/s)
V_{vuggy}/V_{intact}
V_p
V_{s1}
V_{s2}

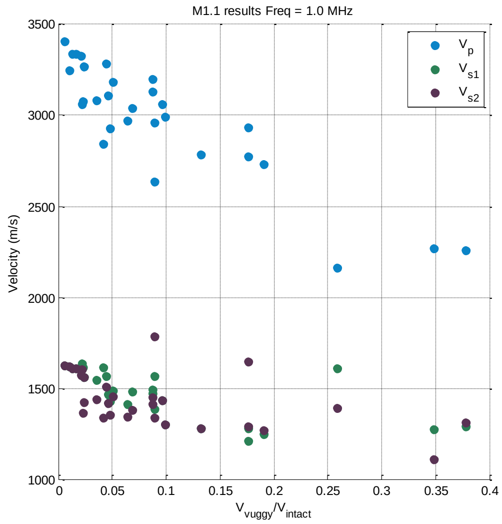

M1.2 results Freq = 564 kHz
Velocity (m/s)
V_{vuggy}/V_{intact}
V_p
V_{s1}
V_{s2}

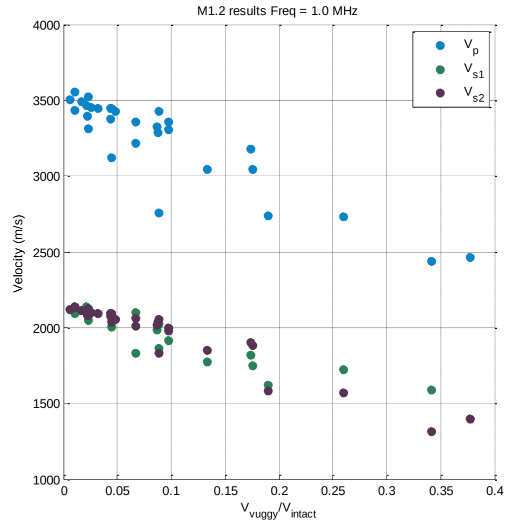

M2 results Freq = 564 kHz
Velocity (m/s)
V_{vuggy}/V_{intact}
V_p
V_{s1}
V_{s2}

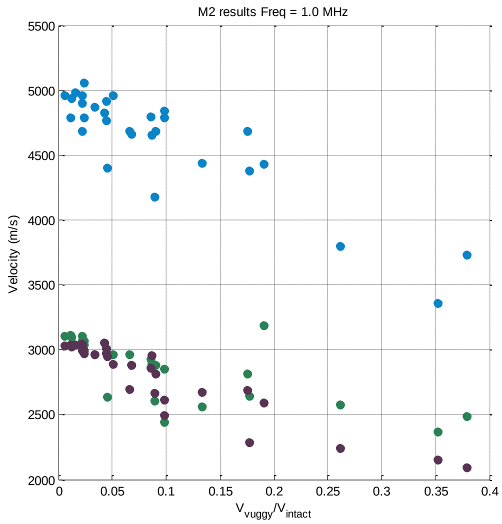

M3 results Freq = 564 kHz
Velocity (m/s)
V_{vuggy}/V_{intact}
V_p
V_{s1}
V_{s2}

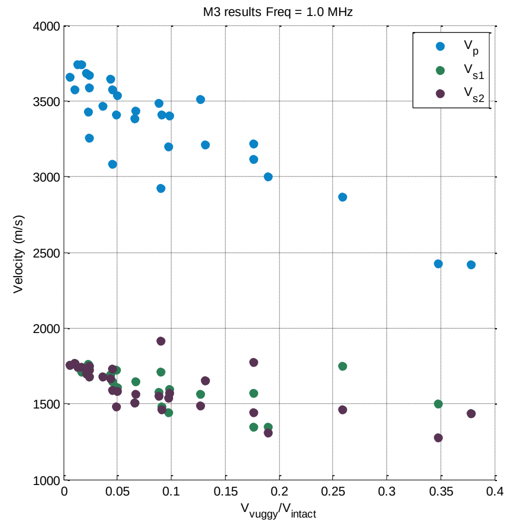

M4.1 results Freq = 564 kHz
Velocity (m/s)
V_{vuggy}/V_{intact}
V_p
V_{s1}
V_{s2}

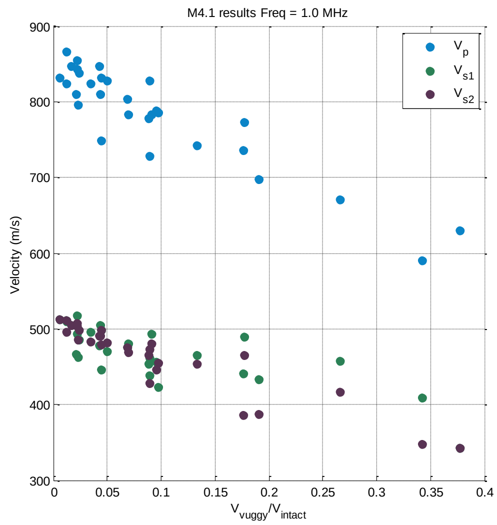

M4.2 results Freq = 564 kHz
Velocity (m/s)
V_{vuggy}/V_{intact}
V_p
V_{s1}
V_{s2}

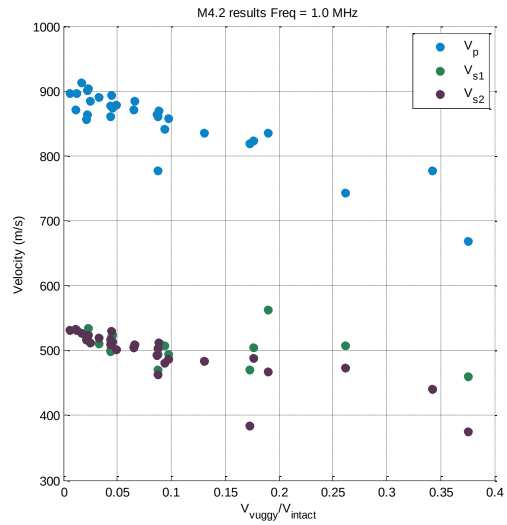

M1.1 results Freq = 1.0 MHz
Velocity (m/s)
V_{vuggy}/V_{intact}
V_p
V_{s1}
V_{s2}


M1.2 results Freq = 1.0 MHz
Velocity (m/s)
V_{vuggy}/V_{intact}
V_p
V_{s1}
V_{s2}


M2 results Freq = 1.0 MHz
Velocity (m/s)
V_{vuggy}/V_{intact}


M3 results Freq = 1.0 MHz
Velocity (m/s)
V_{vuggy}/V_{intact}
V_p
V_{s1}
V_{s2}


M4.1 results Freq = 1.0 MHz
Velocity (m/s)
V_{vuggy}/V_{intact}
V_p
V_{s1}
V_{s2}


M4.2 results Freq = 1.0 MHz
Velocity (m/s)
V_{vuggy}/V_{intact}
V_p
V_{s1}
V_{s2}

APPENDIX B: Program Codes and Other Calculations

```
;=======================================FISH CODE==================================
; fname: PSwavePulse.fis %fist%\3d, PFC3d
; Created by Alma Ornes, University of ALberta
; PURPOSE: PFC3D test environment for wave propagation
;
; Transducers and receivers will be placed inside the sample
; A pulse (displacement) will be applied to the transducer
; the time to get to the receivers will be measured
;
;============================================================================
;=============================================================================
def et3_v_probes
; create transducer
    trans_y = (0.95*mv_Hc)/2.0
    trans_x = 0.0
    trans_z = 0.0
    trans_bid = b_id(ball_near3(trans_x,trans_y,trans_z))
    command
        group velo_trans range id = @trans_bid
    end_command
; create receiver probes
    _x1 = 0.0
    _x2 = 0.5*mv_Rc
    _x3 = 0.0
    _x4 = -0.5*mv_Rc
    _x5 = 0.0
;
    _y_b = -0.95*mv_Hc/2
;
    _z1 = -0.5*mv_Rc
    _z2 = 0.0
    _z3 = 0.5*mv_Rc
    _z4 = 0.0
    _z5 = 0.0
;
    vel_bid1 = b_id(ball_near3(_x1,_y_b,_z1))
    vel_bid2 = b_id(ball_near3(_x2,_y_b,_z2))
    vel_bid3 = b_id(ball_near3(_x3,_y_b,_z3))
    vel_bid4 = b_id(ball_near3(_x4,_y_b,_z4))
    vel_bid5 = b_id(ball_near3(_x5,_y_b,_z5))
;
    command
        group vel_probe range id = @vel_bid1
        group vel_probe range id = @vel_bid2
        group vel_probe range id = @vel_bid3
        group vel_probe range id = @vel_bid4
        group vel_probe range id = @vel_bid5
    end_command
;
end
; =============================================================================
def rec1p_vel
;  define the histories to track the velocity at the probes
  rec1p_vel = b_yvel(find_ball(vel_bid1))
  rec2p_vel = b_yvel(find_ball(vel_bid2))
  rec3p_vel = b_yvel(find_ball(vel_bid3))
  rec4p_vel = b_yvel(find_ball(vel_bid4))
  rec5p_vel = b_yvel(find_ball(vel_bid5))
; velocity in the x and z directions
  rec1s1_vel = b_xvel(find_ball(vel_bid1))
  rec2s1_vel = b_xvel(find_ball(vel_bid2))
  rec3s1_vel = b_xvel(find_ball(vel_bid3))
  rec4s1_vel = b_xvel(find_ball(vel_bid4))
  rec5s1_vel = b_xvel(find_ball(vel_bid5))
  rec1s2_vel = b_zvel(find_ball(vel_bid1))
  rec2s2_vel = b_zvel(find_ball(vel_bid2))
  rec3s2_vel = b_zvel(find_ball(vel_bid3))
  rec4s2_vel = b_zvel(find_ball(vel_bid4))
  rec5s2_vel = b_zvel(find_ball(vel_bid5))
;
```

```
    ProblemTime = time
end
;==================================================================================
def rec1s1_vel
;  define the histories to track the velocity at the probes
    rec1s1_vel = b_xvel(find_ball(vel_bid1))
    rec2s1_vel = b_xvel(find_ball(vel_bid2))
    rec3s1_vel = b_xvel(find_ball(vel_bid3))
    rec4s1_vel = b_xvel(find_ball(vel_bid4))
    rec5s1_vel = b_xvel(find_ball(vel_bid5))
; velocity in the y direction
    rec1p_vel = b_yvel(find_ball(vel_bid1))
    rec2p_vel = b_yvel(find_ball(vel_bid2))
    rec3p_vel = b_yvel(find_ball(vel_bid3))
    rec4p_vel = b_yvel(find_ball(vel_bid4))
    rec5p_vel = b_yvel(find_ball(vel_bid5))
; velocity in the zdirection
    rec1s2_vel = b_zvel(find_ball(vel_bid1))
    rec2s2_vel = b_zvel(find_ball(vel_bid2))
    rec3s2_vel = b_zvel(find_ball(vel_bid3))
    rec4s2_vel = b_zvel(find_ball(vel_bid4))
    rec5s2_vel = b_zvel(find_ball(vel_bid5))
;
    ProblemTime = time
end
;==================================================================================
def rec1s2_vel
;  define the histories to track the velocity at the probes
    rec1s2_vel = b_zvel(find_ball(vel_bid1))
    rec2s2_vel = b_zvel(find_ball(vel_bid2))
    rec3s2_vel = b_zvel(find_ball(vel_bid3))
    rec4s2_vel = b_zvel(find_ball(vel_bid4))
    rec5s2_vel = b_zvel(find_ball(vel_bid5))
;velocity in the y direction
    rec1p_vel = b_yvel(find_ball(vel_bid1))
    rec2p_vel = b_yvel(find_ball(vel_bid2))
    rec3p_vel = b_yvel(find_ball(vel_bid3))
    rec4p_vel = b_yvel(find_ball(vel_bid4))
    rec5p_vel = b_yvel(find_ball(vel_bid5))
; velocity in the x direction
    rec1s1_vel = b_xvel(find_ball(vel_bid1))
    rec2s1_vel = b_xvel(find_ball(vel_bid2))
    rec3s1_vel = b_xvel(find_ball(vel_bid3))
    rec4s1_vel = b_xvel(find_ball(vel_bid4))
    rec5s1_vel = b_xvel(find_ball(vel_bid5))
;
    ProblemTime = time
end
; ================================================================================
def run_wave_test
; runs either a P or an S wave
;
; set up the probes and history plots
   et3_v_probes
   command
            WALL property xvel 0.00
            WALL property yvel 0.00
            WALL property zvel 0.00
            md_zerovel
            SET et3_servo_pon=0
            SET et3_servo_son=0
            md_eq
            history reset
            history nstep = 10
   end_command
   CASE_OF wave_type
; Set up p-wave (y-direction)
   CASE 0
     command
            history id=1 rec1p_vel
```

```
        history id=2 rec2p_vel
        history id=3 rec3p_vel
        history id=4 rec4p_vel
        history id=5 rec5p_vel
;
        history id=6 rec1s1_vel
        history id=7 rec2s1_vel
        history id=8 rec3s1_vel
        history id=9 rec4s1_vel
        history id=10 rec5s1_vel
;
        history id=11 rec1s2_vel
        history id=12 rec2s2_vel
        history id=13 rec3s2_vel
        history id=14 rec4s2_vel
        history id=15 rec5s2_vel
   end_command
; Set up s1-wave (x-direction)
  CASE 1
    command
        history id=1 rec1s1_vel
        history id=2 rec2s1_vel
        history id=3 rec3s1_vel
        history id=4 rec4s1_vel
        history id=5 rec5s1_vel
;
        history id=6 rec1p_vel
        history id=7 rec2p_vel
        history id=8 rec3p_vel
        history id=9 rec4p_vel
        history id=10 rec5p_vel
;
        history id=11 rec1s2_vel
        history id=12 rec2s2_vel
        history id=13 rec3s2_vel
        history id=14 rec4s2_vel
        history id=15 rec5s2_vel
   end_command
; Set up s2-wave (z-direction)
  CASE 2
    command
        history id=1 rec1s2_vel
        history id=2 rec2s2_vel
        history id=3 rec3s2_vel
        history id=4 rec4s2_vel
        history id=5 rec5s2_vel
;
        history id=6 rec1p_vel
        history id=7 rec2p_vel
        history id=8 rec3p_vel
        history id=9 rec4p_vel
        history id=10 rec5p_vel
;
        history id=11 rec1s1_vel
        history id=12 rec2s1_vel
        history id=13 rec3s1_vel
        history id=14 rec4s1_vel
        history id=15 rec5s1_vel
;
   end_command
  END_CASE
  command
        SET time 0
        history id=16 ProblemTime
        ;PROPERTY damp=0.1
        DAMP local 0.0
        DAMP viscous normal 0.1
        DAMP viscous shear 0.1
        DAMP viscous notens off
        prop        xdisp=0.0 ydisp=0.0 zdisp=0.0
```

```
            prop            xvel = 0.0 yvel = 0.0 zvel=0.0
            clump prop xdisp=0.0 ydisp=0.0 zdisp=0.0
            clump prop xvel = 0.0 yvel = 0.0 zvel=0.0
            viewveltriax
            SET FISHCALL FC_CYC_MOT _pulse_wave
            CYCLE 1000
            SET mt_eq_lim=0.002
            PROPERTY damp=0.7
  end_command
    md_eq_rat = mt_eq_lim
    md_eq_steps = mt_eq_Ns
    md_eq
    if md_eq_stat # 0 then
              error = '_run_wave_test damping not sufficient check test.'
    end_if
end
;================================================================================
def _pulse_wave
; apply displacement only if time is lesser than a period
 Amplit = mt_saDel
 IF time > 1.0/Freq
     _vib_vel = 0.0
  ELSE
     _vib_vel = Amplit*(1.0-cos(2.0*pi*Freq*time))/2.0
; CHeck which type of wave to apply
    CASE_OF wave_type
     CASE 0
            command
                 Property yvel =@_vib_vel range group velo_trans
            end_command
     CASE 1
            command
                 Prop xvel =@_vib_vel range group velo_trans
            end_command
     CASE 2
            command
                 Prop zvel =@_vib_vel range group velo_trans
            end_command
    END_CASE
  ENDIF
; Run one pulse
end
;================================================================================
def viewveltriax
;create useful plots to view the p-wave or s-wave tests
;
  command
            plot create hist_vel
            plot add hist 1 2 3 4 5 10 vs 16
            plot show
            plot create P-S-waves
            plot set rot 100 0 0 mag 1.5
            plot set dist 0.5
            plot set plane n 0 0 -1 o 0 0 0
            plot set background white
            plot add axes brown
            plot add ball yellow plane behind
            plot add ball range group velo_trans green
            plot add ball range group vel_probe lblue
            plot add fish crk_item red blue red blue
            plot add disp scale 0.05
            plot show
  end_command
end
;===============================================================================
def wave_output
  _wavetest = md_run_name + '.txt'
  command
            history write 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16  file @_wavetest
  end_command
```

```
end

;==========================================================================
;EOF: PSWavePulse.fis
;=
;FILE FOR LOADING TEST
;---------------------------------------------------------------------------
;================================================================================
===========
;fname: vug.fis  %fist%\2d_3d, PFC2D/3D
;
; PURPOSE: vug support functions.
; Originally Developed by Nathan Deisman (2008)
; Modified by Alma Ornes (2011)
; University of Alberta
; ========================================================================
def vug_defs
    Array _buf4(4)  ; used for FISH I/O already defined in sj.fis
    ; Joint property array, slot 1: 1 is initial, 2 is new
    Array _vug_geom(10000000, 11)
    Array _vug_prop(50, 25)
end
vug_defs
; =======================================
def vug_Read
;
; ----- Read in vug data and store in llist and property set arrays.
;      The vug data consists of geometry and property information.
;      Geometry is a rectangualar or ellipse-shaped disk defined by its center, dip,
;      dip direction and radius.
;       Properties are yes to be determined, currently the shapes are simply deleted.
;
;
;      These are initial vug properties, new properties
;      are input via vug_ReadNewProp and assigned by vug_Add.
;      Each vug and property set has a unique positive number {1,2,...}
;      that corresponds with its order in the corresponding file.
;
; INPUT: vug_gfn - name of ASCII file containing vug geometry data
;       vug_pfn - name of ASCII file containing vug property data
;
;       File formats:
;         geometry-data file:
;           line 1: comment, read and discarded.
;           line 2: total number of vugs (integer)
;           For each vug, two lines containing:
;             <pn> <cen_x> <cen_y>                                            (2D)
;                 <dip> <rad1> <rad2> <sha>                                   (2D)
;             <pn>  <sha> <cen_x> <cen_y> <cen_z>                             (3D)
;                 <dip> <dd> <rad1> <rad2> <rad3>                             (3D)
;              pn    - property set number (integer, >= 1)
;              sha   - shape (0 - rect, 1 - ellipse)
;              cen_x - center, x-component (float)
;              cen_y - center, y-component (float)
;              cen_z - center, z-component (3D only, float)
;              dip   - dip (float, degrees)
;              dd    - dip direction (3D only, float, degrees)
;              rotd  - roation along dip axis (3D only float, degrees)
;              rad1  - radius1 (float)
;              rad2   - radius2 (float)
;              rad3   - radius3 (float)
;         property-data file:
;           line 1: comment, read and discarded.
;           line 2: total number of property sets (integer), must equal
;                   value specified for vug_np in driver file
;           For each property set, four lines containing:
;             <model_id> <den> <b_kmod> <b_rat> <rad_mult> <pb_mod>
;             <pb_krat> <pb_nor> <pb_nor_std> <pb_she> <pb_she_std>
;            <tp1> <tp2> <tp3> <tp4> <tp5>
;            <tp6> <tp7> <tp8> <tp9> <tp10>
```

```
;
;        model_id - 1 = change props, 0 = null
;        den - ball density
;        b_kmod - ball modulus
;        b_rat - ball stiffness ratio
;        rad_mult - bond raduis multiplier
;        pb_mod - bond modulus
;        pb_krat - bond stiffness ratio kn/ks
;        pb_nor - bond normal strength
;        pb_nor_std - bond  normal strength std dev
;        pb_she - bond shear strength
;        pb_she_std  - bond shear strength std dev
;        tp1 - thermal prop 1
;        tp2 - thermal prop 1
;        tp3 - thermal prop 1
;        tp4 - thermal prop 1
;        tp5 - thermal prop 1
;        tp6 - thermal prop 1
;        tp7 - thermal prop 1
;        tp8 - thermal prop 1
;        tp9 - thermal prop 1
;        tp10 - thermal prop 1
;
;        If any value is negative, then it will not be used when
;        assigning vug properties, and thus, the corresponding
;        inherited property will not be overridden.
;
; OUTPUT: vug_nj - total number of vugs
;      llist of vug data blocks
;
 _vug_ReadGeom
 _vug_ReadProp
end
; ==================================
def _vug_ReadGeom
 ii = open( vug_gfn, 0, 1 )  ; Read access, ASCII mode
 IF ii # 0 then
     error = '[_vug_ReadGeom]: Failed to open file.'
 end_if
 ii = read( _buf4, 2 )
 IF ii # 0 then
     error = '[_vug_ReadGeom]: Failed to read first 2 lines of file.'
 end_if
 _vug_tot = parse( _buf4(2), 1 )
 if _vug_tot < 1 then
     error = '[_vug_ReadGeom]: Total number of vugs not positive.'
 end_if

 loop _vug_num (1, _vug_tot)
 ii = read( _buf4, 2)
 _vug_set = parse( _buf4(1), 1 )
  IF _vug_set > vug_np then
      error = '[_vug_ReadGeom]: Property set number greater than vug_set.'
  end_if
  IF dim = 3 then
   _vug_geom(_vug_num, 1)  = parse( _buf4(1), 1 ) ; 1 = vugmodel
   _vug_geom(_vug_num, 2)  = parse( _buf4(1), 2 ) ; 2 = vugshape
   _vug_geom(_vug_num, 3)  = parse( _buf4(1), 3 ) ; 3 = xcen
   _vug_geom(_vug_num, 4)  = parse( _buf4(1), 4 ) ; 4 = ycen
   _vug_geom(_vug_num, 5)  = parse( _buf4(1), 5 ) ; 5 = zcen
   _vug_geom(_vug_num, 6)  = parse( _buf4(2), 1 ) ; 6 = dip
   _vug_geom(_vug_num, 7)  = parse( _buf4(2), 2 ) ; 7 = dip direction
   _vug_geom(_vug_num, 8)  = parse( _buf4(2), 3 ) ; 8 = rotation along dip axis
   _vug_geom(_vug_num, 9)  = parse( _buf4(2), 4 ) ; 9 = radius1
   _vug_geom(_vug_num, 10) = parse( _buf4(2), 5 ) ; 10 = radius2
   _vug_geom(_vug_num, 11) = parse( _buf4(2), 6 ) ; 11 = radius3
  ELSE
   _vug_geom(_vug_num, 1) = parse( _buf4(1), 1 ) ; 1 = vugmodel
   _vug_geom(_vug_num, 2) = parse( _buf4(1), 2 ) ; 2 = vugshape
   _vug_geom(_vug_num, 3) = parse( _buf4(1), 3 ) ; 3 = xcen
```

```
    _vug_geom(_vug_num, 4) = parse( _buf4(1), 4 ) ; 4 = ycen
    _vug_geom(_vug_num, 5) = parse( _buf4(2), 1 ) ; 5 = dip
    _vug_geom(_vug_num, 6) = parse( _buf4(2), 2 ) ; 6 = radius1
    _vug_geom(_vug_num, 7) = parse( _buf4(2), 3 ) ; 7 =  radius2
   end_if
  end_loop
  ii = close
end
; ===================================
def _vug_ReadProp  ;{i: p_idx}
  ;
  ii = open( vug_pfn, 0, 1 )  ; Read access, ASCII mode
  if ii # 0 then
   error = '[_vug_ReadProp]: Failed to open file.'
  end_if
  ;
  ii = read( _buf4, 2 )
  if ii # 0 then
   error = '[_vug_ReadProp]: Failed to read first 2 lines of file.'
  end_if
  ;
  _np = parse( _buf4(2), 1 )
  if _np # vug_np then
   error = '[_vug_ReadProp]: Number of property sets not equal to vug_np.'
  end_if
  ;
  loop _pnum (1, vug_np)
   ii = read( _buf4, 4 )
        _vug_prop(_pnum, 1) = parse( _buf4(1), 1 ) ; model
        _vug_prop(_pnum, 2) = parse( _buf4(1), 2 ) ; ball density
        _vug_prop(_pnum, 3) = parse( _buf4(1), 3 ) ; ball modulus
        _vug_prop(_pnum, 4) = parse( _buf4(1), 4 ) ; ball stiffness ratio
        _vug_prop(_pnum, 5) = parse( _buf4(1), 5 ) ; bond raduis multiplier
        _vug_prop(_pnum, 6) = parse( _buf4(1), 6 ) ; bond modulus
        _vug_prop(_pnum, 7) = parse( _buf4(2), 1 ) ; bond stiffness ratio kn/ks
        _vug_prop(_pnum, 8) = parse( _buf4(2), 2 ) ; bond normal strength
        _vug_prop(_pnum, 9) = parse( _buf4(2), 3 ) ; bond  normal strength std dev
        _vug_prop(_pnum, 10) = parse( _buf4(2), 4 ) ; bond shear strength
        _vug_prop(_pnum, 11) = parse( _buf4(2), 5 ) ; bond shear strength std dev
  ;
        _vug_prop(_pnum, 12) = parse( _buf4(3), 1 ) ; thermal prop 1
        _vug_prop(_pnum, 13) = parse( _buf4(3), 2 ) ; thermal prop 2
        _vug_prop(_pnum, 14) = parse( _buf4(3), 3 ) ; thermal prop 3
        _vug_prop(_pnum, 15) = parse( _buf4(3), 4 ) ; thermal prop 4
        _vug_prop(_pnum, 16) = parse( _buf4(3), 5 ) ; thermal prop 5
        _vug_prop(_pnum, 17) = parse( _buf4(4), 1 ) ; thermal prop 6
        _vug_prop(_pnum, 18) = parse( _buf4(4), 2 ) ; thermal prop 7
        _vug_prop(_pnum, 19) = parse( _buf4(4), 3 ) ; thermal prop 8
        _vug_prop(_pnum, 20) = parse( _buf4(4), 4 ) ; thermal prop 9
        _vug_prop(_pnum, 21) = parse( _buf4(4), 5 ) ; thermal prop 10
 ;
  end_command

  end_loop
  ii = close
end
; ===================================
def vug_Make
command
 SET extra ball 2 ; create one extra variable associated with each ball
 SET extra contact 2 ; create two extra variable associated with each contact
 SET echo on
end_command

 vug_Make_geom
 vug_Make_props
 vug_poros
;
command
  print _pvt, _pnv, _pmv
```

```
end_command
;
;command
;   SET extra ball 0 ; delete extra variable associated with each ball
;   SET extra contact 0 ; delete two extra variable associated with each contact
;end_command
end
; ====================================
def vug_Make_props
 ; _vug_prop(_vugmodel, 2)  ; ball density
 ; _vug_prop(_vugmodel, 3)  ; ball modulus
 ; _vug_prop(_vugmodel, 4)  ; ball stiffness ratio
 ; _vug_prop(_vugmodel, 5)  ; bond raduis multiplier
 ; _vug_prop(_vugmodel, 6)  ; bond modulus
 ; _vug_prop(_vugmodel, 7)  ; bond stiffness ratio kn/ks
 ; _vug_prop(_vugmodel, 8)  ; bond normal strength
 ; _vug_prop(_vugmodel, 9)  ; bond  normal strength std dev
 ; _vug_prop(_vugmodel, 10) ; bond shear strength
 ; _vug_prop(_vugmodel, 11)  ; bond shear strength std dev
 ; _vug_prop(_vugmodel, 12) ; thermal prop 1
 ; _vug_prop(_vugmodel, 13) ; thermal prop 2
 ; _vug_prop(_vugmodel, 14) ; thermal prop 3
 ; _vug_prop(_vugmodel, 15) ; thermal prop 4
 ; _vug_prop(_vugmodel, 16) ; thermal prop 5
 ; _vug_prop(_vugmodel, 17) ; thermal prop 6
 ; _vug_prop(_vugmodel, 18) ; thermal prop 7
 ; _vug_prop(_vugmodel, 19) ; thermal prop 8
 ; _vug_prop(_vugmodel, 20) ; thermal prop 9
 ; _vug_prop(_vugmodel, 21) ; thermal prop 10
 ; loop all balls
 ; if ball is in a vug, determine if the ball is
 ; a) contacted by a vug
 ; b) not contacted by a non-vug
 ; if a) assign the contact a marker of 1
 ; if b) assign the contact a marker of 2
 ; for all balls inside of vug, change the ball properties to those listed above
        _vug_Make_ball_props
        _vug_Make_contact_props
end
; ====================================
def _vug_Make_ball_props
 bp = ball_head
;
 loop while bp # null
   bnext = b_next(bp) ; required to keep from exiting loop once a ball is delelted, see pg 2-42 2D users guide
   _vugmodel = b_extra(bp,1)
;
   IF  _vugmodel > 0 then
;     b_color(bp) = 1
      b_dens(bp) = _vug_prop(_vugmodel, 2)
      ; assign mechanical props
        b_kn(bp) =  _vug_prop(_vugmodel, 3)
        b_ks(bp) = _vug_prop(_vugmodel, 3)/_vug_prop(_vugmodel, 4)
      ; assign thermal props
        b_thexp(bp) = _vug_prop(_vugmodel, 12)
        b_thsheat(bp) = _vug_prop(_vugmodel, 13)
      ; create contanct marker for vug-vug and vug-non-vug contact
    cp = b_clist(bp)
    loop while cp # null
      IF c_ball1(cp) = bp then ; find other ball
             bp_other = c_ball2(cp)
             cpnext = c_b1clist(cp)
      ELSE
             bp_other = c_ball1(cp)
             cpnext = c_b2clist(cp)
      end_if
; determine if bp_other is a vug or non-vug (or wall) pointer_type: ball=100, wall=101
      IF pointer_type(bp_other)=100 then ; this is a ball-ball contact
            _vugmodel_other = b_extra(bp_other,1)
         IF _vugmodel_other > 0 then ; determine if  vug - vug or vug - non-vug and mark
```

```
;             a vug-vug ball contact
              c_extra(cp,1) = 1 ; indicates a vug-vug contact
              c_extra(cp,2) = _vugmodel ; indicates model type
         ELSE
;             a vug - non-vug contact
              c_extra(cp,1) = 2 ; indicates a vug - non-vug contact
              c_extra(cp,2) = _vugmodel ; indicates model type
         end_if
      end_if
      cp = cpnext
    end_loop
  end_if
  bp = bnext
 end_loop
;
end
;
; ====================================
def _vug_Make_contact_props
; loop all contacts, search for extra 1 and 2 to update props
 cp = contact_head
  loop while cp # null
      IF c_extra(cp,1) > 0 then ; an vug - vug or a vug - non-vug is present
          pbp = c_pb(cp)
          IF pbp # null then
              _dum = pointer_type(cp)
              _dum1 = cp
              _dum2 = pbp
              _vugmodel = c_extra(cp,2)
;      change contact properties
              pb_rad(pbp) = _vug_prop(_vugmodel,5)
              pb_kn(pbp) = _vug_prop(_vugmodel,6)
              pb_ks(pbp) = _vug_prop(_vugmodel,6)/_vug_prop(_vugmodel,7)
              _pb_nstr = (_vug_prop(_vugmodel, 8)-_vug_prop(_vugmodel, 9))
              pb_nstrength(pbp) = _pb_nstr + urand * _vug_prop(_vugmodel, 9)
              _pb_sstr = (_vug_prop(_vugmodel, 10)-_vug_prop(_vugmodel, 11))
              pb_sstrength(pbp) =  _pb_sstr + urand * _vug_prop(_vugmodel, 9)
              c_thres(cp) =  _vug_prop(_vugmodel, 14)
          end_if
     end_if
     cp = c_next(cp)
 end_loop
end
;
; ====================================
def vug_Make_geom
;
; ----- Create vugs based on vug data.  The vugs are
;       added one at a time, starting with the lowest numbered vug.  For
;       each vug, the particles that are grouped into a common group and assigned
;       (a) a group name title vugnum +_vug
;       (b) the properties corresponding to the vugs
; INPUT: llist of vug data blocks
;       to rotate, first translate to local geometery, then rotate, then translate back to global geometry, then check
;       clear extra array variables and find total volume of particles in intact material
bp = ball_head
_sumVol = 0.0
loop while bp # null
      b_extra(bp,1) = 0
      b_extra(bp,2) = 0
      IF dim = 2 then
         _vol = pi*b_rad(bp)^2
      ELSE
         _vol = (4.0/3.0)*pi*b_rad(bp)^3
      end_if
      _sumVol = _sumVol + _vol
      bp = b_next(bp)
end_loop
;
loop _vug_num (1, _vug_tot)
```

```
;
 IF dim = 3 then
       _vugmodel = _vug_geom(_vug_num, 1)
       _vugshape = _vug_geom(_vug_num, 2)
       _xcen     = _vug_geom(_vug_num, 3)
       _ycen     = _vug_geom(_vug_num, 4)
       _zcen     = _vug_geom(_vug_num, 5)
       _dip      = _vug_geom(_vug_num, 6)
       _dipdir   = _vug_geom(_vug_num, 7)
       _rotd     = _vug_geom(_vug_num, 8)
       _radius1  = _vug_geom(_vug_num, 9)
       _radius2  = _vug_geom(_vug_num, 10)
       _radius3  = _vug_geom(_vug_num, 11)
 ELSE
       _vugmodel = _vug_geom(_vug_num, 1)
       _vugshape = _vug_geom(_vug_num, 2)
       _xcen = _vug_geom(_vug_num, 3)
       _ycen = _vug_geom(_vug_num, 4)
       _dip = _vug_geom(_vug_num, 5)
       _radius1 = _vug_geom(_vug_num, 6)
       _radius2 = _vug_geom(_vug_num, 7)
 end_if
 _model =_vug_prop(_vugmodel, 1)
 _a = _radius1
 _b = _radius2
 _c = _radius3
 _h = _xcen
 _k = _ycen
 _n = _zcen
 _dip = degrad*_dip
 _dd1 = 270.0 -_dipdir  ;rotation
 _dd1 = degrad*_dd1
 _rot = degrad*_rotd
 _x_max =  _a
 _x_min = -1*_a
 _y_max = _b
 _y_min = -1*_b
 _z_max = _c
 _z_min = -1*_c
 bp = ball_head
 loop while bp # null
    bnext = b_next(bp) ; required to keep from exiting loop once a ball is delelted, see pg 2-42 2D users guide
    IF dim = 2 thend
          _x_ball = b_x(bp)
          _y_ball = b_y(bp)
          _x_r = _x_ball - _h
          _y_r = _y_ball - _k
          ; rotate x,y here
          _X_p = cos(_dip)*_x_r - sin(_dip)*_y_r
          _Y_p = sin(_dip)*_x_r + cos(_dip)*_y_r
    ELSE
          _x_ball = b_x(bp)
          _y_ball = b_y(bp)
          _z_ball = b_z(bp)
          _x_r = _x_ball - _h
          _y_r = _y_ball - _k
          _z_r = _z_ball - _n
; rotate x,y,z here
; y-axis rotation
           _X_P1 = sin(_dd1)*_z_r + cos(_dd1)*_x_r
           _Y_P1 = _y_r
           _Z_P1 = cos(_dd1)*_z_r - sin(_dd1)*_x_r
; z-axis rotation
           _X_P2 = cos(_dip)*_X_P1 - sin(_dip)*_Y_P1
           _Y_P2 = sin(_dip)*_X_P1 + cos(_dip)*_Y_P1
           _Z_P2 = _Z_P1
; x-axis rotation, rotation about the dip axis for relative orientation of the minor axis
           _X_P = _X_P2
           _Y_P =  cos(_rot)*_Y_P2 + sin(_rot)*_Z_P2
           _Z_P = -sin(_rot)*_Y_P2 + cos(_rot)*_Z_P2
```

```
        end_if
        IF _vugshape = 0 then ; rectangle
             if _X_P <= _x_max then
             if _X_P >= _x_min then
             if _Y_P <= _y_max then
             if _Y_P >= _y_min then
             if dim = 2 then
                    dum3=1.0
              else
              if _Z_P <= _z_max then
              if _Z_P >= _z_min then
                   dum3=1.0
        end_if
        end_if
        end_if
        end_if
       end_if
       end_if
        end_if
        end_if
  IF _vugshape = 1 then ; ellipse
    IF dim = 2 then
     dum3 = ((_X_P)^2)/(_a^2) + ((_Y_P)^2)/(_b^2)
    ELSE
     dum3 = ((_X_P)^2)/(_a^2) + ((_Y_P)^2)/(_b^2) + ((_Z_P)^2)/(_c^2)
    end_if
  end_if
  IF _model = 0 then ; null the particle
    IF dum3 <= 1.0 then ; particle lies inside of shape
          cp = b_clist(bp)
          loop while cp # null
            bclp = b_clump(bp)
            IF bclp # null then ; determine if this ball is assigned to a clump
                       ii = cl_rel(bclp,bp)
            end_if
             IF c_ball1(cp) = bp then ; find other ball
               bp_other = c_ball2(cp)
               cpnext = c_b1clist(cp)
               if pointer_type(bp_other)=100 then
                      b_extra(bp_other,2) = 1 ; mark boundary of null ball
               end_if
             ELSE
                 bp_other = c_ball1(cp)
                 cpnext = c_b2clist(cp)
                 IF pointer_type(bp_other)=100 then
                      b_extra(bp_other,2) = 1; mark boundary of null ball
                 end_if
             end_if
             cp = cpnext
          end_loop
          ii=b_delete(bp)
   end_if
  end_if
  IF _model = 1 then ; assign model id to ball extra variable
       IF dum3<=1.0 then ; particle lies inside of shape
    ; change ball properties
         bclp = b_clump(bp)
         IF bclp # null then ; determine if this ball is assigned to a clump
            ii = cl_rel(bclp,bp)
          end_if
          b_extra(bp,1) =_vugmodel
        end_if
  end_if
  bp = bnext ; next item in list of balls
  dum3=2.0 ; reset dum3 to be greater than 1 for rectangular calc
 end_loop
end_loop
;
end
; ===================================
```

```
def vug_poros
;   determine amount of volume taken up by the vugs compared to the intact sample
;   Input = _sumVol
;   Determines fraction of total vugs
;   fraction of empty (null) vugs
;   fraction of filled (material) vugs
  bp = ball_head
  _sumVolmv = 0.0
  _sumVoli = 0.0
  _sumVolnv = 0.0
  _sumVoltv = 0.0
  loop while bp # null
       IF dim = 2 then
              _vol = pi*b_rad(bp)^2
       ELSE
              _vol = (4.0/3.0)*pi*b_rad(bp)^3
       end_if
       IF b_extra(bp,1) > 0 then    ; the particle is inside a material or filled vug
              _sumVolmv = _sumVolmv + _vol
       ELSE             ; the particle is intact material
              _sumVoli = _sumVoli + _vol
       end_if
       bp = b_next(bp)
  end_loop
  _sumVolnv = _sumVol - _sumVolmv - _sumVoli              ; get volume of null vugs
  _sumVoltv = _sumVol - _sumVoli                          ; get amount of total vugs
  _pvt = _sumVoltv / _sumVol                              ; fraction of total vugs
  _pnv = _sumVolnv / _sumVol                              ; fraction of null vugs
  _pmv = _sumVolmv / _sumVol                              ; fraction of material vugs
end
; ====================================
def plot_vug
 command
          plot create vug_view
          plot set title text 'Vuggy specimen'
          plot set rot 120 30 0
          plot set mag 3
          plot add fish vug_item
          plot add ball y r blu gre c mag o lgray &
                     lblue lmag lo w
          plot add axes br
  end_command
  IF mg_shape # 0 then
          command
                 plot add clump cyan
           end_command
   end_if
   command
; ------- Create plot view with only vugs
          group in_vug range fish vug_element
          vug_edge
          plot create vug_only
          plot set title text 'Location of Vugs in specimen'
          plot set rot 120 30 0
          plot set mag 3
          plot add group range fish vug_element
          plot add axes br
          plot add ball range fish vug_edge
   end_command
end
;====================================
def vug_element
  range_element
;
; defines vugs in a range for plotting
; INPUT: fc_arg(0)  - object being queried for range inclusion
; b_extra(bp,1)  - flag indicating ball is inside the vug
;
     bp = fc_arg(0)
     _vugmodel = b_extra(bp,1) ; indicate wether ball is in vug
```

```
    IF  _vugmodel > 0 then    ; ball is a vug
         vug_element = 1
    ELSE
         vug_element = 0
    end_if
end
;=====================================
def vug_edge
  range_element
;
; defines vugs in a range for plotting
; INPUT: fc_arg(0)  - object being queried for range inclusion
; b_extra(bp,1)  - flag indicating ball is inside the vug
;
    bp = fc_arg(0)
    _vugmodel = b_extra(bp,2) ; indicate wether ball is in vug
    IF  _vugmodel > 0 then    ; ball is a vug
         vug_edge = 1
    ELSE
         vug_edge = 0
    end_if
end
;=====================================
def intact_element
  range_element
;
; defines area of intact in a range for clump formation
; INPUT: fc_arg(0)  - object being queried for range inclusion
; b_extra(bp,1)  - flag indicating ball is inside the
;
    bp = fc_arg(0)
    _vugmodel = b_extra(bp,1) ; indicate wether ball is in vug
    IF  _vugmodel > 0 then    ; ball is a vug
         intact_element = 0
    ELSE
         intact_element = 1
    end_if
end
;=====================================
def vug_item
  plot_item
;-------Generates a plotitem for visualizing the set of vugs
;still in progress need to fix different colours for different models (future work)
;
; The plot item is activated by typing: PLOT ADD FISH VUG_ITEM
; then adding balls : PLOT ADD BALL colo1 colo2 color3 etc
;  Note: must especify as many colour as models plus 1
;  Two types of VUGs are supported
;  Null vugs:  (model ID =0)
;  the boundary balls are marked and the inside of the vug is empty
;  Material vugs: (model ID = 1)
; the boundary balls are marked to the same colour as the null balls
; inside balls are coloured according to their model number
;
  cp = contact_head
  loop while cp # null
        IF c_extra(cp,1) = 2 then ; it is a vug-nonvug contact
              bp = c_ball1(cp)
               b_color(bp) = 1
               bp = c_ball2(cp)
               b_color(bp) = 1
       end_if
      cp = c_next(cp)
  end_loop
  bp = ball_head
  loop while bp # null
       _vugmodel = b_extra(bp,1) ; indicate wether ball is in vug
       if  _vugmodel > 0 then
            b_color(bp) = _vugmodel + 1
        end_if
```

```
      _vugbnd = b_extra(bp,2)
      if _vugbnd = 1 then
          b_color(bp) = 1
      end_if
      bp = b_next(bp)
  end_loop
end
;
; ============================================
return
;EOF: vug.fis
```

================================MATLAB SCRIPT===================================

```
% mm for voxel pixel conversion
lp = 300;
% Image processing CarbV6 is matrix with of stacked CT images
lh = length(CarbV6(1,1,:));
min = 16*eps;
for i=1:lh
        Car6red(:,:,i) = imresize(CarbV6(:,:,i),(2.78/3));
        imshow(Car6red(:,:,i));
        title(sprintf('Slice # %d',i));
        pause(0.05)
end
for i=1:lh
        BWimseq(:,:,i) = im2bw(Car6red(:,:,i),0.07);
        imshow(BWimseq(:,:,i));pause(0.05)
        title(sprintf('Slice # %d',i));
end
% pose is matrix with coordinates for the ROI
% can use single set of coordinates
% or a varying progression if sample is tilted
for i=1:lh
        h = imellipse(gca,pose(i,:));
        BW(:,:,i) = createMask(h);
end
BWpores = imcomplement(BWimseq);
for i=1:lh
        BWvugs(:,:,i) = BWpores(:,:,i).*BW(:,:,i);
        imshow(BWvugs(:,:,i));pause(0.05);
        title(sprintf('Slice # %d',i));
end
C1vugs = bwconncomp(BWvugs);
[Lvugs,NUM] = bwlabeln(BWvugs);
Vuggy = regionprops(C1vugs,'PixelList','Area','Centroid');
RVuggy = regionprops(C1vugs,'PixelList');
c = struct2cell(RVuggy)';
d=zeros(1,3);
lv = length(RVuggy);
for i=1:lv
        lcheck=length(c{i,1});
        if lcheck > 2
                d=cat(1,d,c{i});
        end
        clear lcheck
end
pixelNUM = length(d)-1;
pix = d(2:length(d),:);
Csolid = bwconncomp(BWimseq);
Rsolid = regionprops(Csolid,'PixelList');
cs = struct2cell(Rsolid)';
ds=zeros(1,3);
lvs = length(Rsolid);
for i=1:lvs
        lcheck=length(cs{i,1});
        if lcheck > 1
                ds=cat(1,ds,cs{i});
        end
        clear lcheck
end
hrc = length(BWimseq(1,1,:));
cylr=bwconncomp(BW(:,:,round(hrc/2)));
Cylr=regionprops(cylr,'Area','Centroid','EquivDiameter');
rcr = Cylr.EquivDiameter(1)/2;
PFC_rot = [0 1 0; 0 0 1; 1 0 0];
sample_centr = [Cylr.Centroid(1,1) Cylr.Centroid(1,2) hrc/2];
lconv = lp/1000000;
vconv = lconv^3;
samp_true_cent = PFC_rot*sample_centr';
for i=1:NUM
        ar(i) = Vuggy(i).Area;
end
```

```
alist = ar';
volum = alist .* vconv;
SumVOL=sum(volum);
ellMat = zeros(NUM,9);
% Rotation matrix to transform from image pixel coordinates to PFC
% Image: dim1 y(+ve down), dim2 x (+ve right), dim3 z (+ve upwards)
% PFC right hand rule with x and z on the horizontal plane,
%     y is the elevation
% first rotate 90 ccw about z axis, second rotate 90 ccw about x axis
for j=1:NUM
   points = [Vuggy(j).PixelList(:,1) Vuggy(j).PixelList(:,2) Vuggy(j).PixelList(:,3)];
   n = size(points,1);
   if n >= 2
      center = [Vuggy(j).Centroid(1,1) Vuggy(j).Centroid(1,2) Vuggy(j).Centroid(1,3)];
      covPts = cov(points)/n;
      [U S] = svd(covPts);
      radii = 2*sqrt(diag(S)*n)';
      [radii ind] = sort(radii,'descend');
      U = U(ind,:);
      Urot = PFC_rot*U*PFC_rot';
      if Urot(1,1)<0
         Urot=-Urot;
         Urot(:,3)=-Urot(:,3);
      end
      bar1 = realsqrt((Urot(1,1))^2+(Urot(1,2))^2+(Urot(1,3))^2);
      bar2 = realsqrt((Urot(2,1))^2+(Urot(2,2))^2+(Urot(2,3))^2);
      bar3 = realsqrt((Urot(3,1))^2+(Urot(3,2))^2+(Urot(3,3))^2);
      Urotp(1,:) = Urot(1,:)./bar1;
      Urotp(2,:) = Urot(2,:)./bar2;
      Urotp(3,:) = Urot(3,:)./bar3;
      dip = asind(-Urotp(1,2));
      dipdir = acosd(abs(realsqrt((Urotp(1,1))^2+(Urotp(1,3))^2)));
      if abs(Urotp(2,2)) > 16*eps
        rotdir = atand(-Urotp(3,2)/Urotp(2,2));
      else
         if abs(cosd(dip)) < 16*eps
            rotdir = 90;
         else
            rotdir = 0;
         end
      end
      true_center = PFC_rot*center' - samp_true_cent;
      ell =[true_center' radii dip dipdir rotdir];
      ellMat(j,:) = ell;
   end
   clear ell S U points radii center true_center dip dipdir rotdir Urot n
end
t =1;
% Vug list equivalent ellipsoid method
ExcelVuggy = zeros(NUM,12);
for k=1:NUM
   if ellMat(k,4)> min
   ExcelVuggy(t,1) = 1;
   ExcelVuggy(t,2) = t;
   ExcelVuggy(t,3) = 1;
   ExcelVuggy(t,4) = round(ellMat(k,1)*lp)/1000000;
   ExcelVuggy(t,5) = round(ellMat(k,2)*lp)/1000000;
   ExcelVuggy(t,6) = round(ellMat(k,3)*lp)/1000000;
   ExcelVuggy(t,7) = round(ellMat(k,7)*10)/10;
   ExcelVuggy(t,8) = round(ellMat(k,8)*10)/10;
   ExcelVuggy(t,9) = round(ellMat(k,9)*10)/10;
   ExcelVuggy(t,10) = round(ellMat(k,4)*lp)/1000000;
   ExcelVuggy(t,11) = round(ellMat(k,5)*lp)/1000000;
   ExcelVuggy(t,12) = round(ellMat(k,6)*lp)/1000000;
   t= t+1;
   end
end
% vug list voxel method
totPixel = length(pix);
pixelVUGlist = zeros(totPixel, 3);
```

```
for n=1:totPixel
   pixelVUGlist(n,:) = PFC_rot*pix(n,:)'-samp_true_cent;
end
ExcelPixVug =zeros(totPixel,12);
for k=1:totPixel
   ExcelPixVug(k,1) = 1;
   ExcelPixVug(k,2) = k;
   ExcelPixVug(k,3) = 0;
   ExcelPixVug(k,4) = round(pixelVUGlist(k,1)*lp)/1000000;
   ExcelPixVug(k,5) = round(pixelVUGlist(k,2)*lp)/1000000;
   ExcelPixVug(k,6) = round(pixelVUGlist(k,3)*lp)/1000000;
   ExcelPixVug(k,7) = 0;
   ExcelPixVug(k,8) = 0;
   ExcelPixVug(k,9) = 0;
   ExcelPixVug(k,10) = (lp/2)/1000000;
   ExcelPixVug(k,11) = (lp/2)/1000000;
   ExcelPixVug(k,12) = (lp/2)/1000000;
end
% vug list equivalent sphere method
sphMat = zeros(NUM,6);
for k=1:NUM
        points = [Vuggy(k).PixelList(:,1) Vuggy(k).PixelList(:,2) Vuggy(k).PixelList(:,3)];
   n = size(points,1);
   if n >= 2
              center = [Vuggy(k).Centroid(1,1) Vuggy(k).Centroid(1,2) Vuggy(k).Centroid(1,3)];
      vol = (Vuggy(k).Area);
      rad = (3*vol/(4*pi))^(1/3);
      true_center = PFC_rot*center' - samp_true_cent;
      sph = [true_center' rad rad rad];
      sphMat(k,:) = sph;
   end
         clear sph vol center true_center rad points n
end
clear t
t =1;
ExcelSphVuggy = zeros(NUM,12);
for k=1:NUM
   if sphMat(k,4)> min
   ExcelSphVuggy(t,1) = 1;
   ExcelSphVuggy(t,2) = t;
   ExcelSphVuggy(t,3) = 1;
   ExcelSphVuggy(t,4) = round(sphMat(k,1)*lp)/1000000;
   ExcelSphVuggy(t,5) = round(sphMat(k,2)*lp)/1000000;
   ExcelSphVuggy(t,6) = round(sphMat(k,3)*lp)/1000000;
   ExcelSphVuggy(t,7) = 0;
   ExcelSphVuggy(t,8) = 0;
   ExcelSphVuggy(t,9) = 0;
   ExcelSphVuggy(t,10) = round(sphMat(k,4)*lp)/1000000;
   ExcelSphVuggy(t,11) = round(sphMat(k,5)*lp)/1000000;
   ExcelSphVuggy(t,12) = round(sphMat(k,6)*lp)/1000000;
   t=t+1;
   end
end
```

Ellipsoid Curvature

The following section was based mostly on the summary by Weisstein (2012). A general ellipsoid is given by the equation below

$$\frac{x^2}{a^2}+\frac{y^2}{b^2}+\frac{z^2}{c^2}=1$$

Where a, b, and c are the semi-axis length. One parameterization of the ellipsoid can be written as

$$x = a\cos u \sin v$$

$$y = b\cos v \sin u$$

$$z = c\cos v$$

Where u and v are angles within the range of $u \in [0, 2\pi)$ and v$\in [0, \pi]$

The mean curvature of an ellipsoid is given by the following equation

$$H=\frac{1}{\rho}=\frac{abc[3(a^2+b^2)+2c^2+(a^2+b^2-2c^2)\cos 2v-2(a^2-b^2)\cos 2u\sin^2 v]}{8[a^2b^2\cos^2 v+c^2(b^2\cos^2 u+a^2\sin^2 u)\sin^2 v]^{3/2}}$$

In the special case of a sphere where a=b=c=R the above equation reduces to:

$$H=\frac{1}{\rho}=\frac{1}{R}$$

where the radius of curvature is equal to radius of the sphere, as expected.

The point of maximum curvature is important to characterize the effect of shape in strength. For the shapes used in these simulations (a≥b≥ c), the maximum curvature occurs at the tip as we approach

$$x = \pm a, y = 0, and\ c = 0$$

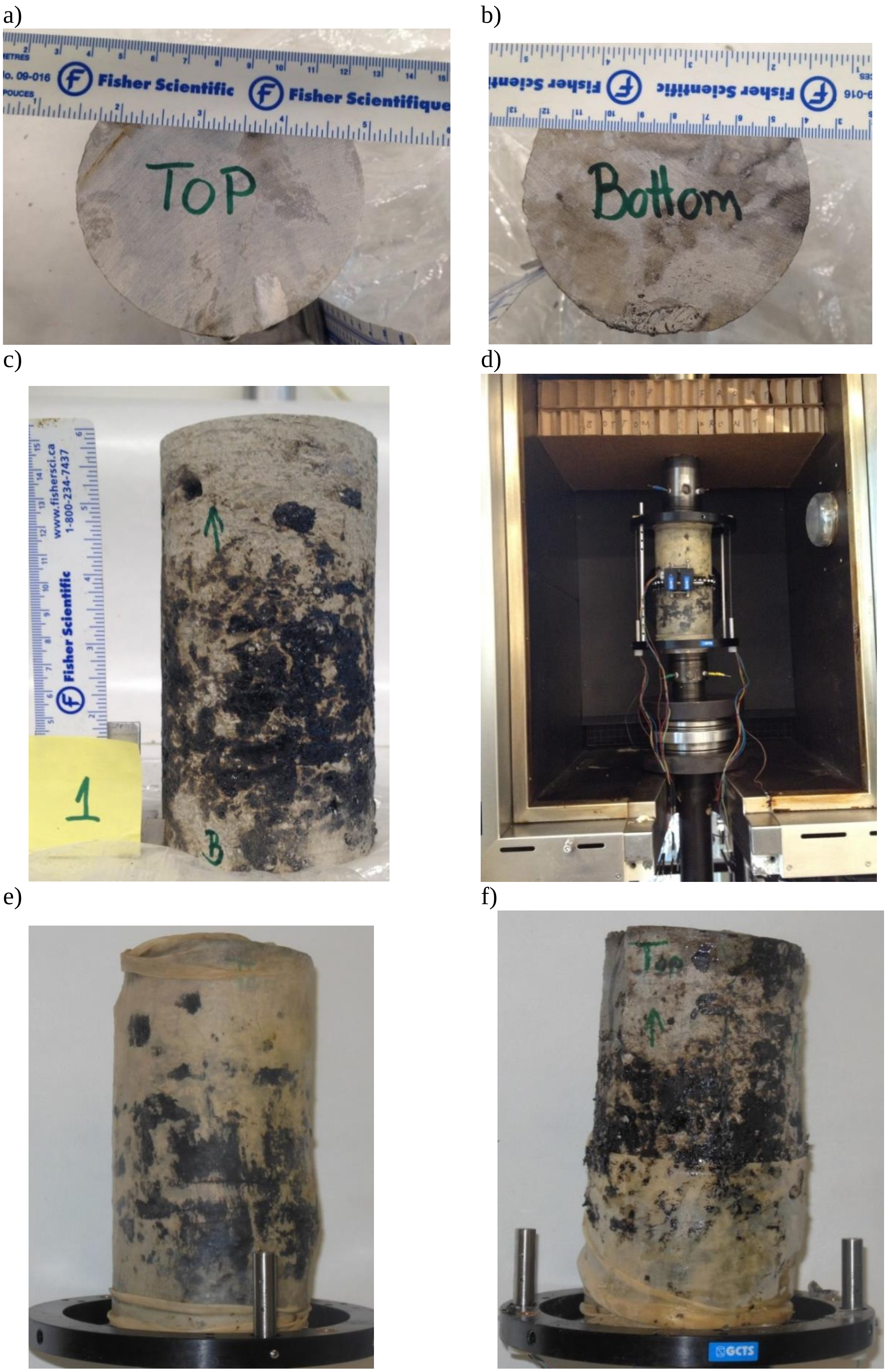
$$\therefore u = 0, \pi\ and\ v = \frac{\pi}{2}$$

Therefore the radius of curvature at the tip can be defined by the equation

$$\frac{1}{\rho}=\frac{a(b^2+c^2)}{2b^2c^2}$$

Griffith (1921) derived that the maximum stress experience by a material with a single elliptical crack was proportional to $\sqrt{c/\rho}$; where c is the half-length of the crack and ρ is the radius of curvature at the crack tip. Expanding this concept to the three dimensional inclusions used in these simulations, the parameter $\sqrt{a/\rho}$ was used to evaluate the effect of ellipsoidal shapes on the unconfined strength. In this case "a" is the longest semi-axis of the ellipsoid and ρ is the radius of curvature at tip as defined above.

APPENDIX C: Laboratory Test Data



Photographs of Sample S1: a) top face, b) bottom face, c) pre-test, d) test set-up, e) post-test, f) post-test

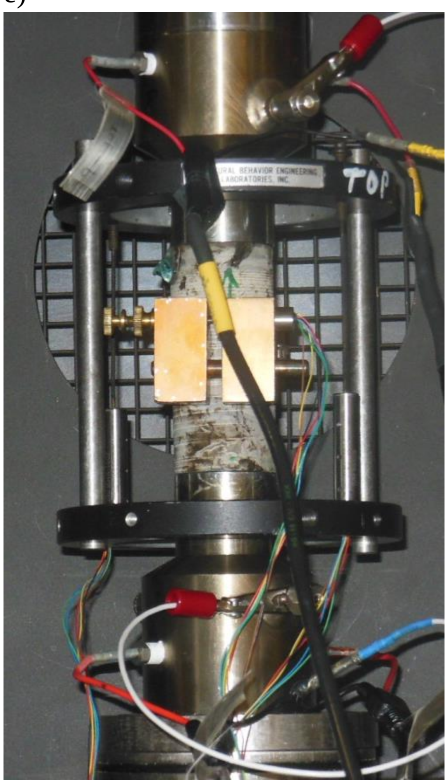
a)

b)

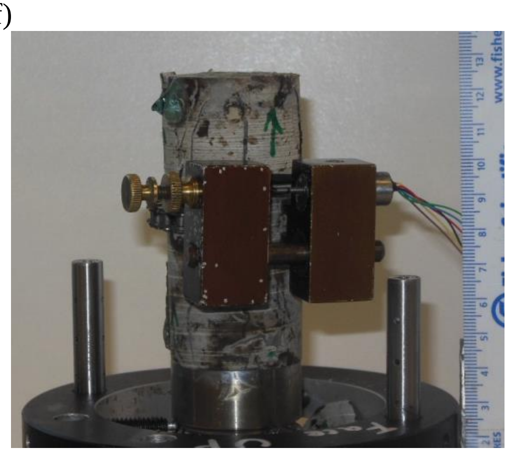
c)

d)

e)



f)

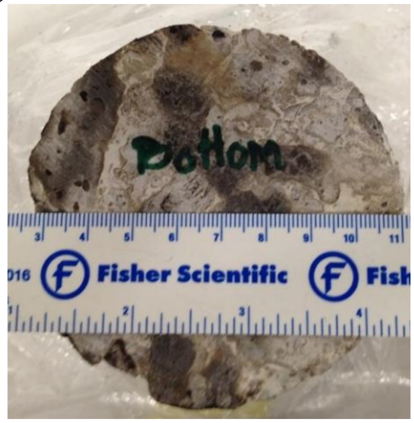


g)

Photographs for Sample S2: a)pre-coring, b)after coring and trimming, c)top face, d) bottom face, e) test set-up, f) post-test, g)post-test

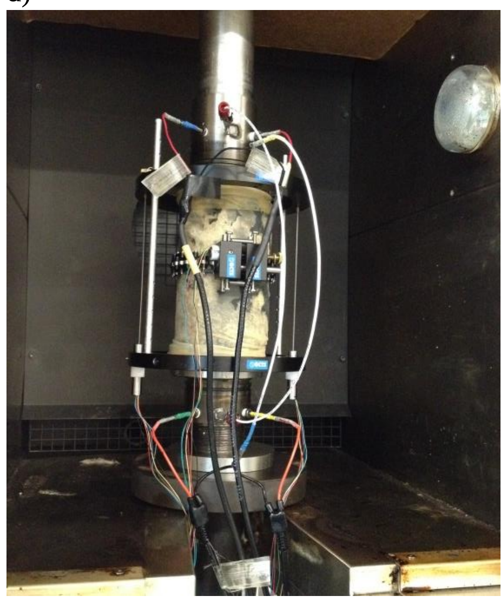
a)

b)



c)

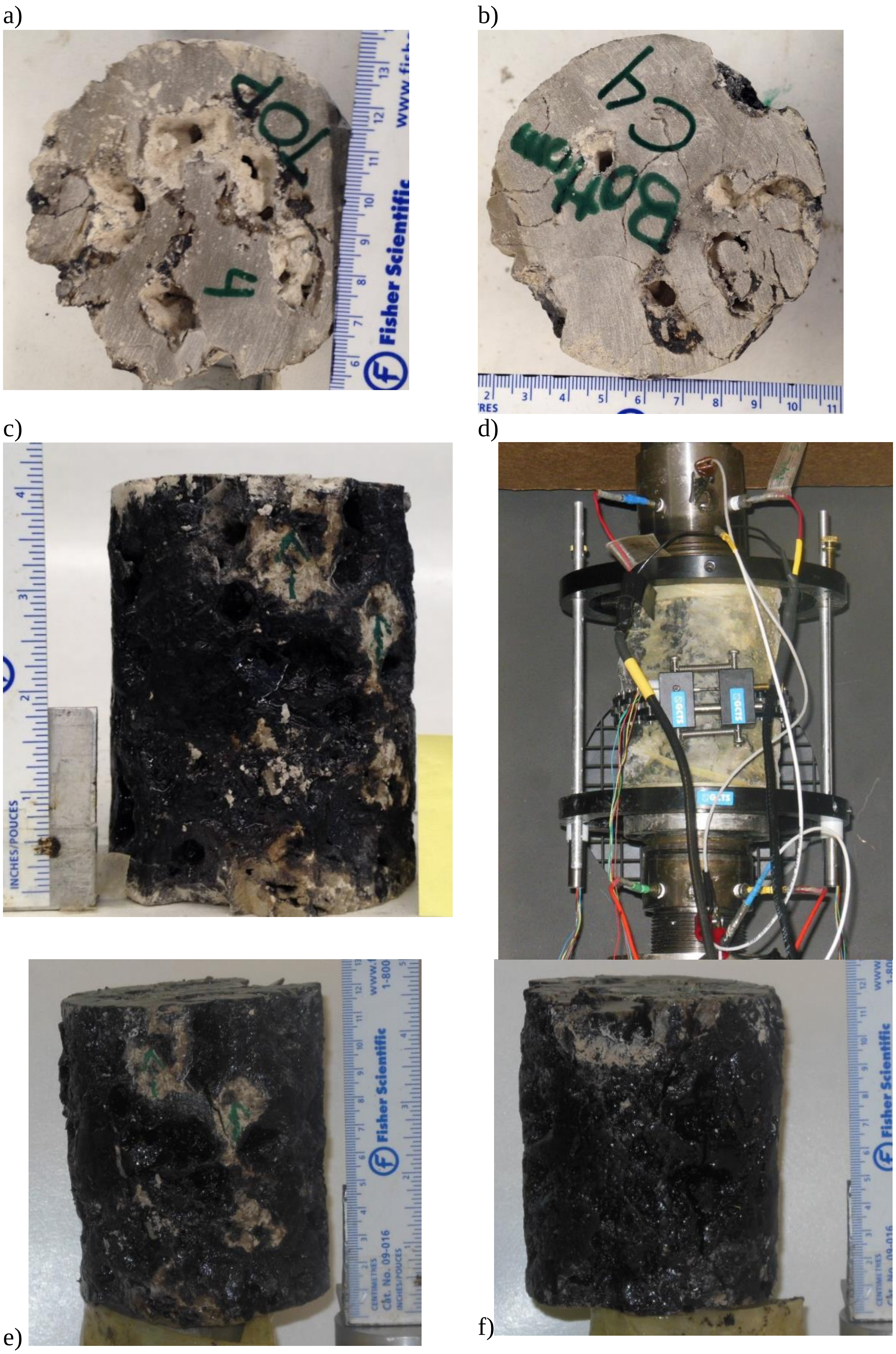
d)



e)

f)

Photographs of Sample S3: a) top face, b) bottom face, c) pre-test, d) test set-up, e) post-test, f) post-test



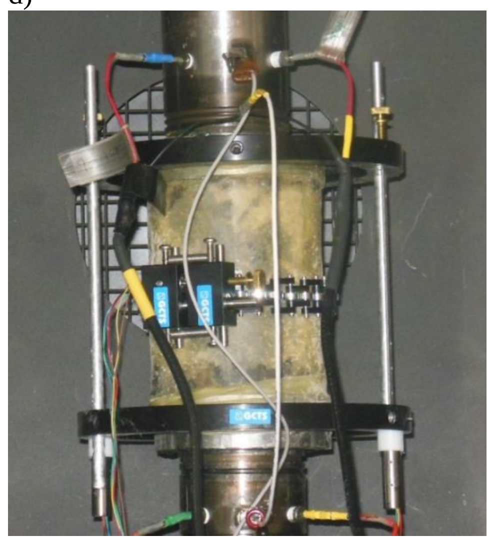
Photographs of Sample S4: a) top face, b) bottom face, c) pre-test, d) test set-up, e) post-test, f) post-test

a)

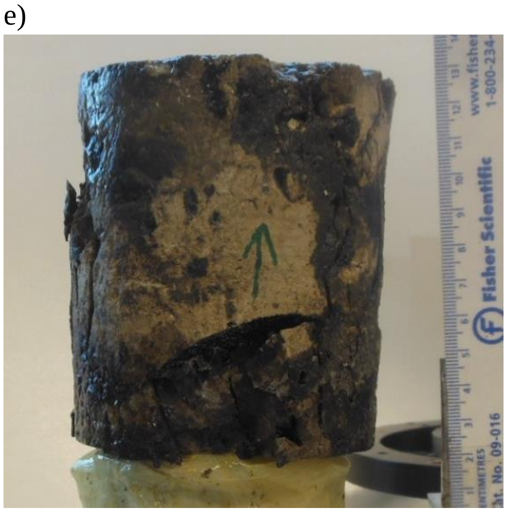
b)

c)

d)



e)

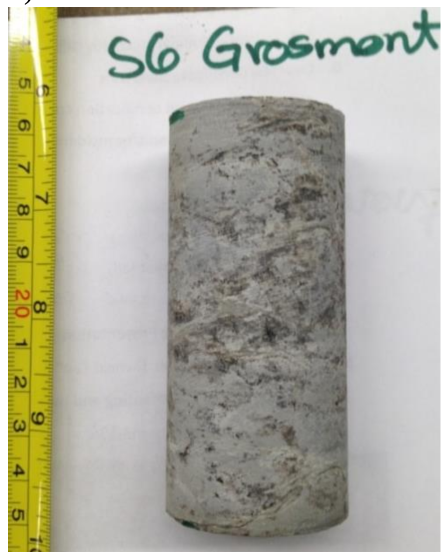


f)

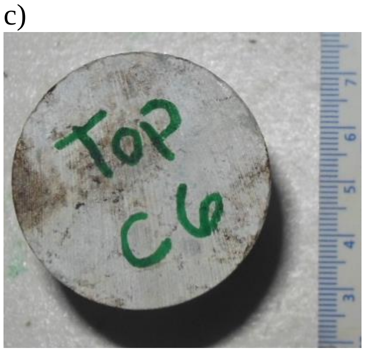
Photographs of Sample S5: a) top face, b) bottom face, c) pre-test, d) test set-up, e) post-test, f) post-test

a)

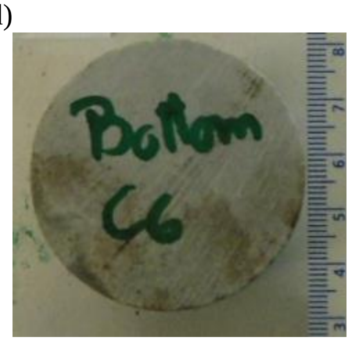
b)



c)



d)

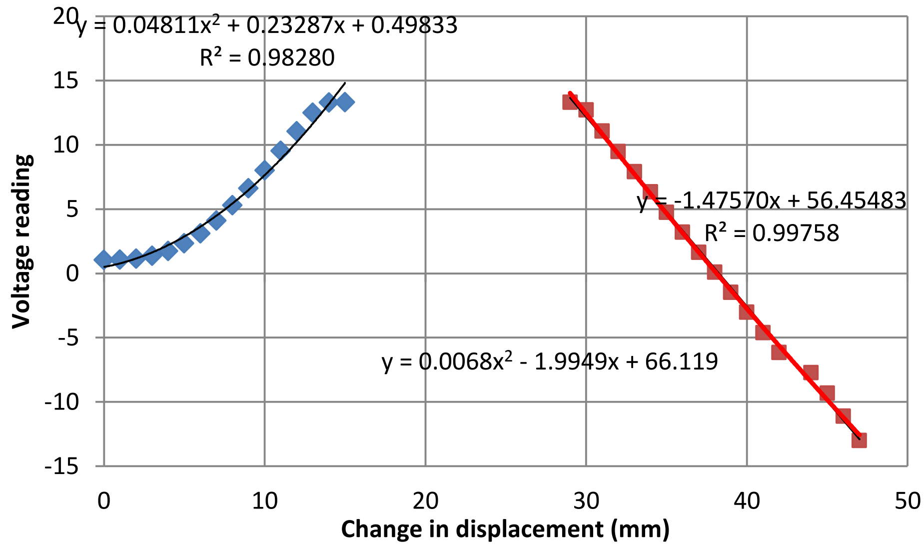


e)

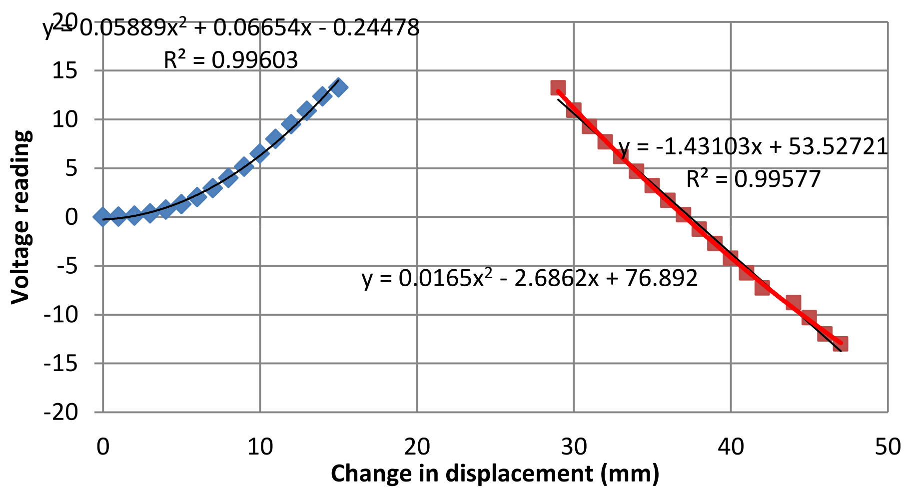
f)

Photographs for Sample S6: a) pre-coring, b) after coring and trimming, c) top face, d) bottom face, e) post-test, f) post-test

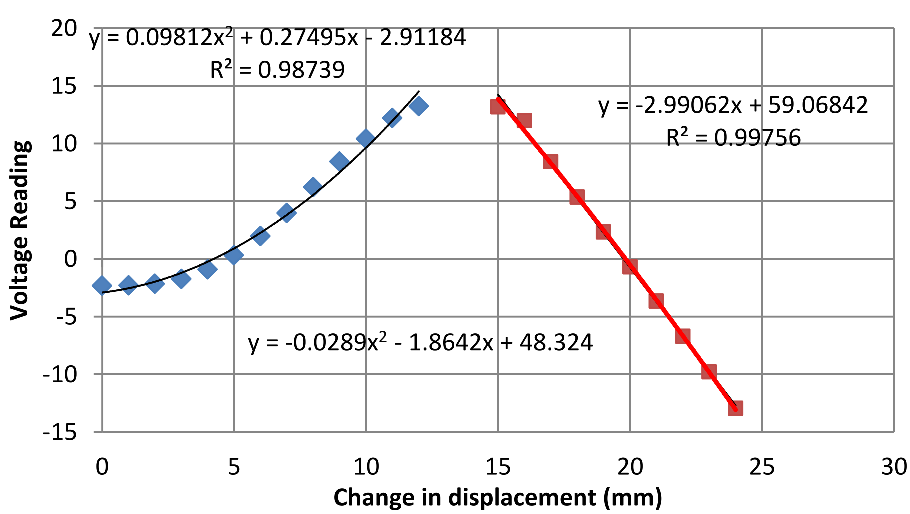
Gauge Calibration



Axial LP 1

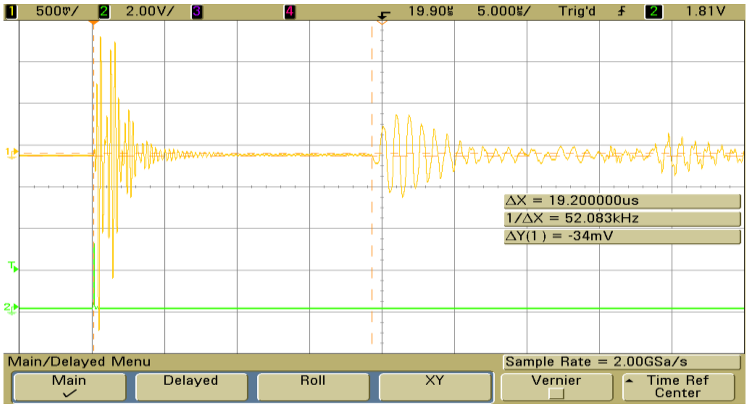


Axial LP2

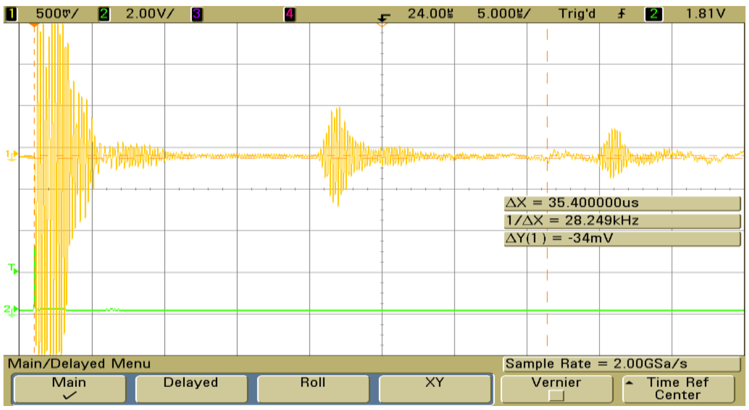


Radial LP

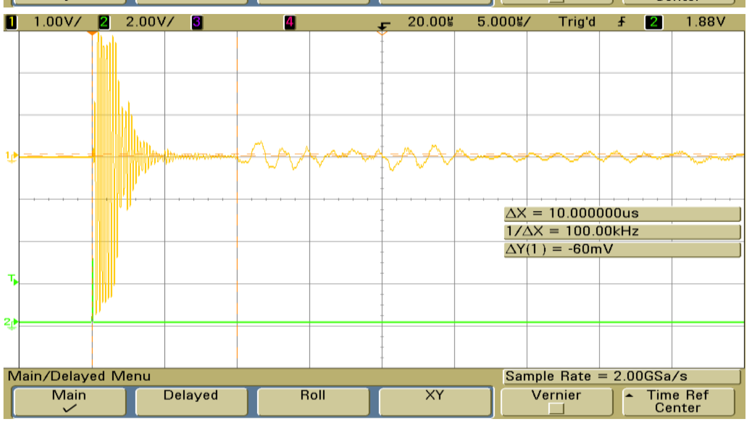
Ultrasonic Face to Face readings



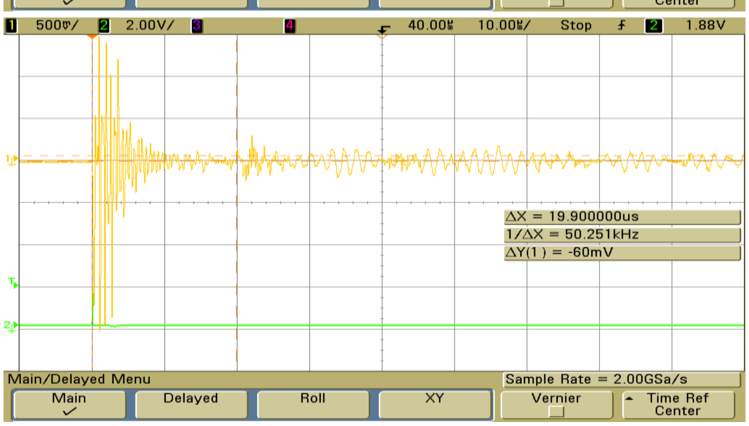
P wave 3.5” diameter



S wave 3.5” diameter

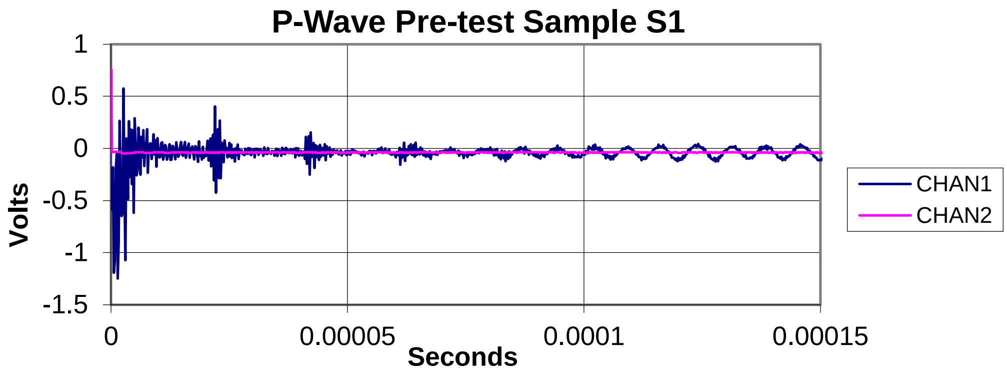


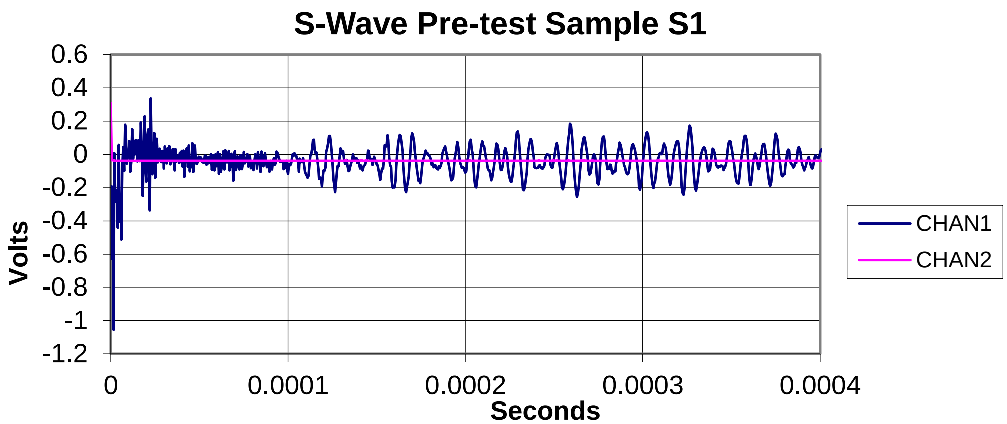
P wave 1.5” diameter

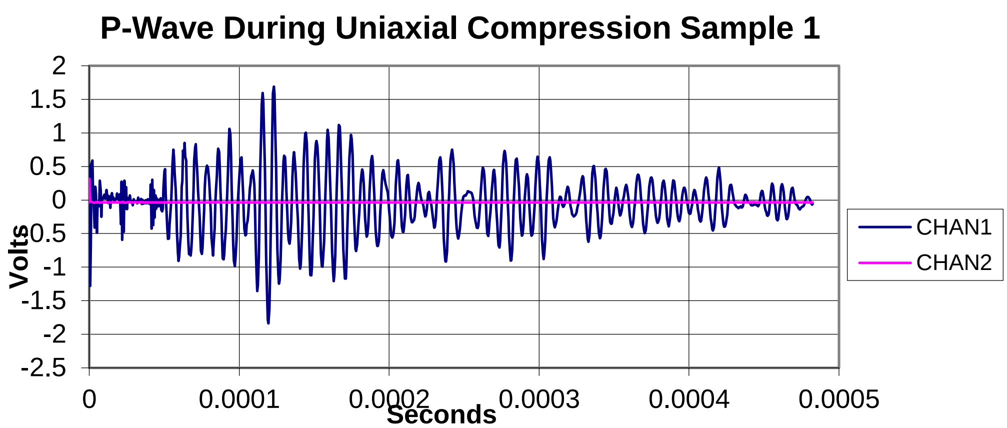


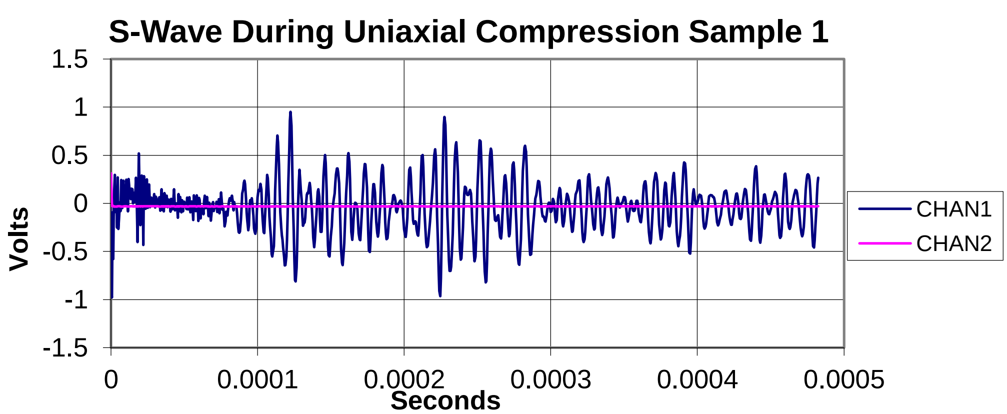
S wave 1.5” diameter

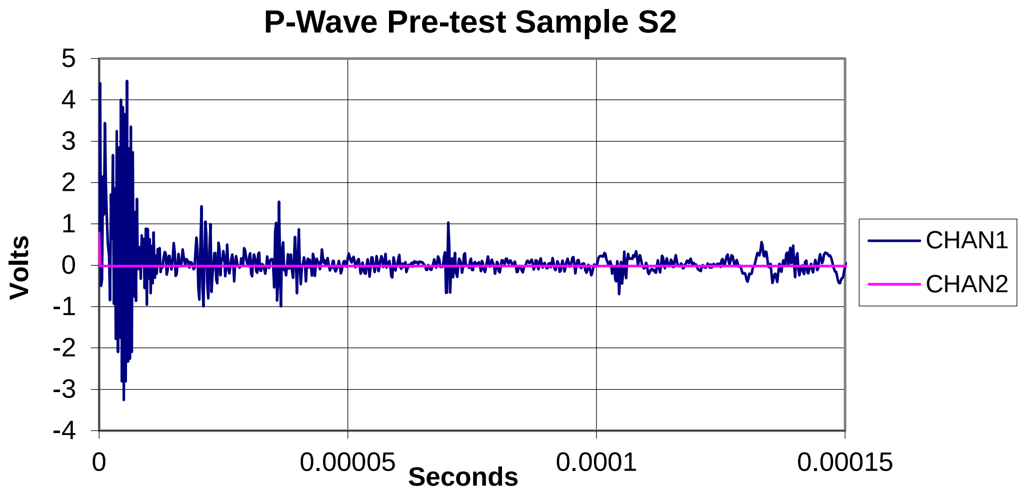
Ultrasonic measurements

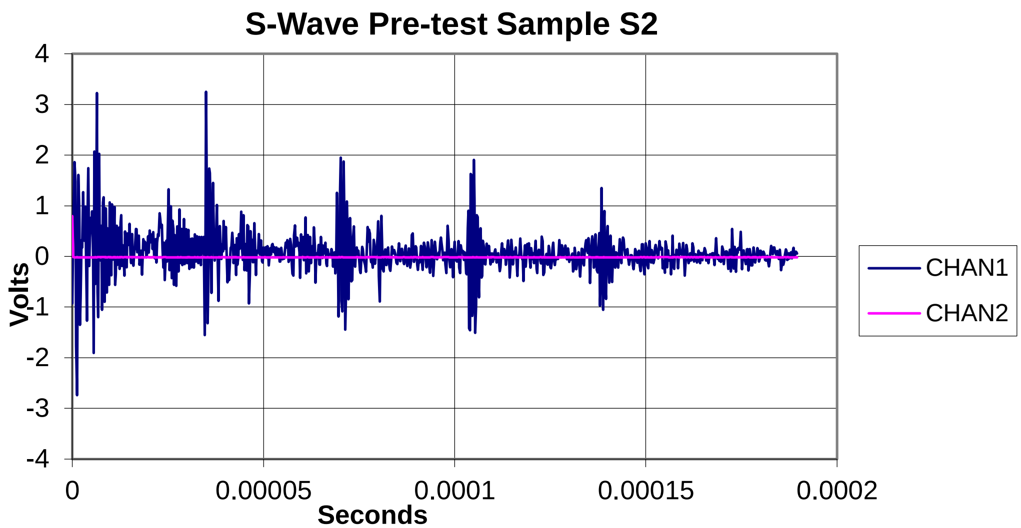


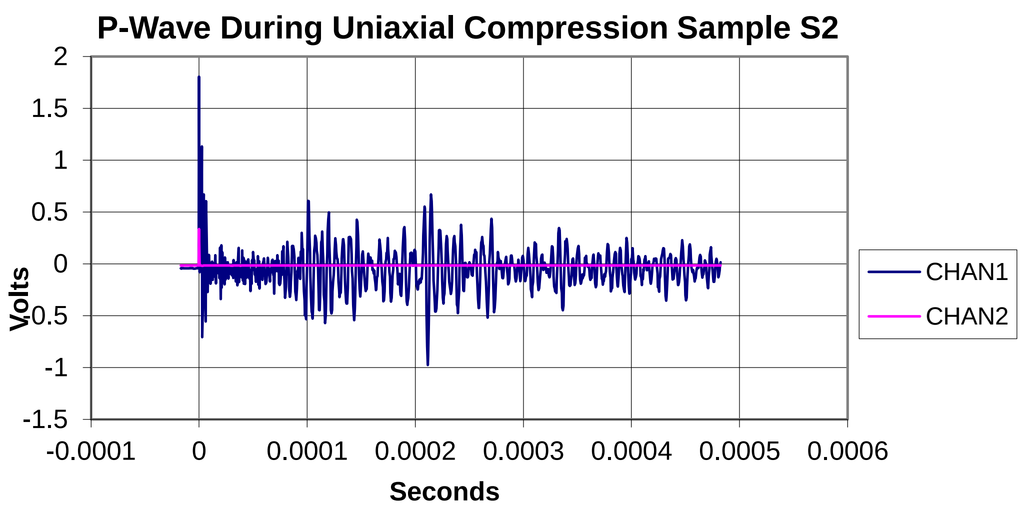


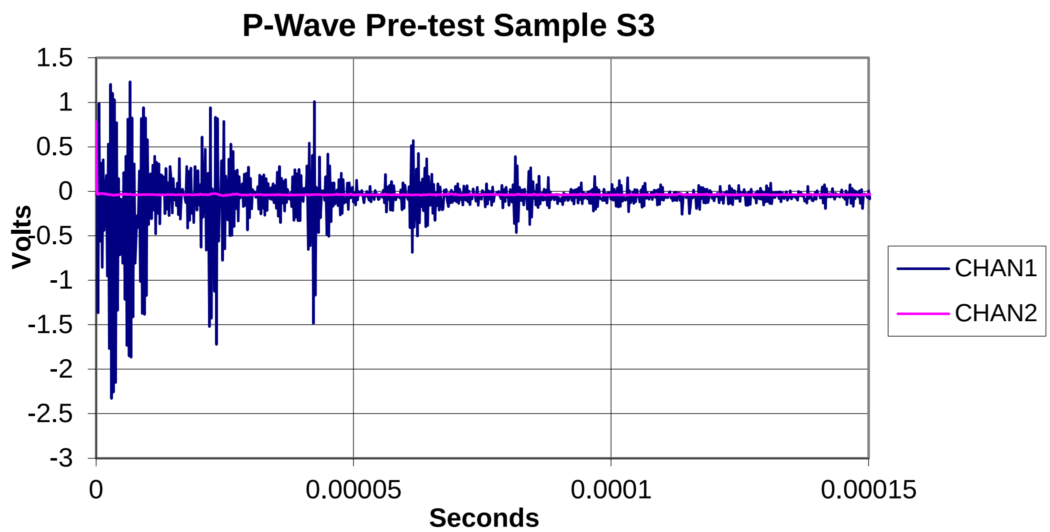


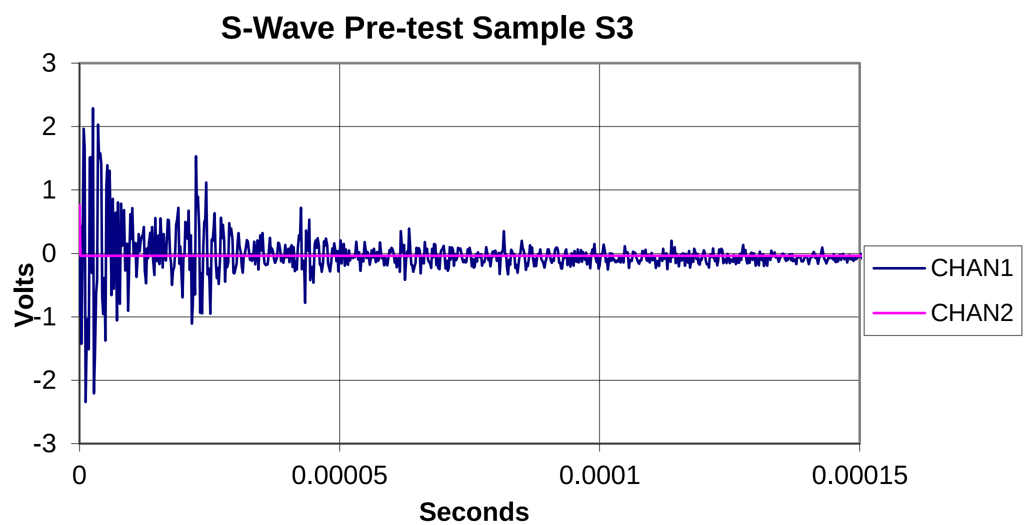


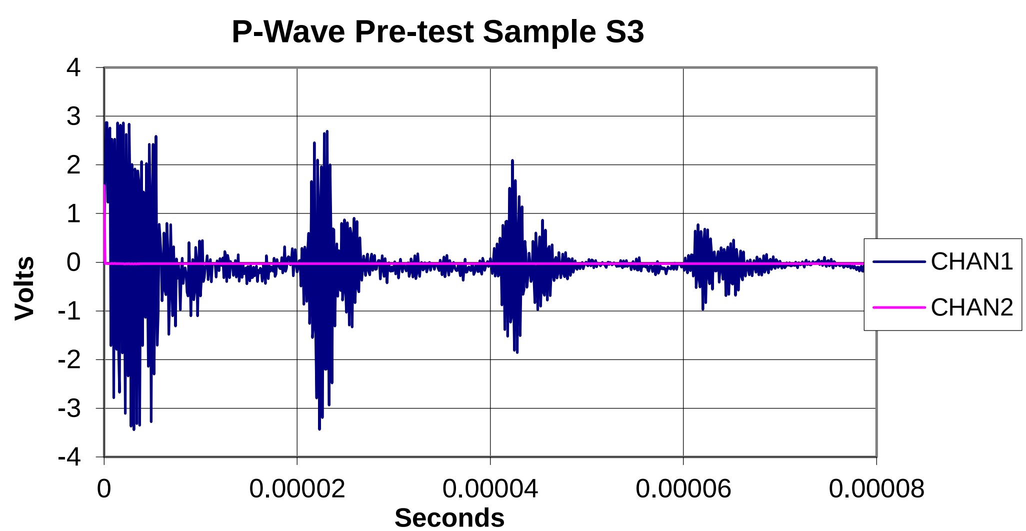

P-Wave Pre-test Sample S2
Volts
Seconds
CHAN1
CHAN2
5
4
3
2
1
0
-1
-2
-3
-4
0
0.00005
0.0001
0.00015

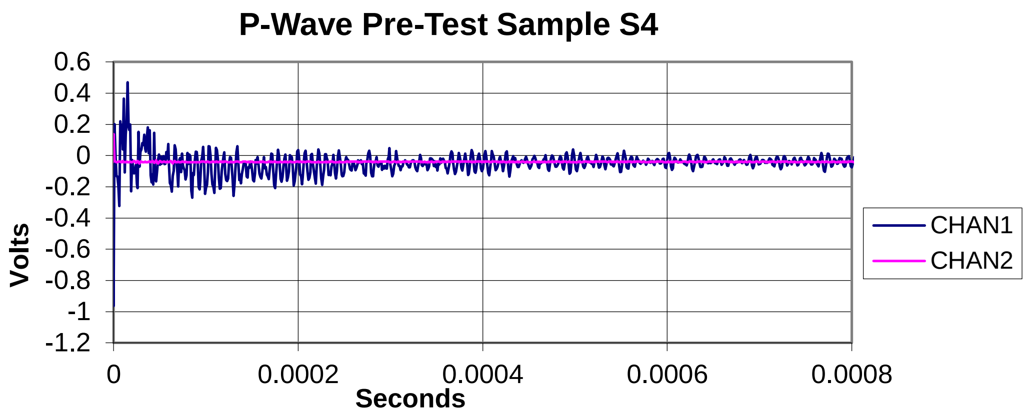

S-Wave Pre-test Sample S2
Volts
Seconds
CHAN1
CHAN2
4
3
2
1
0
-1
-2
-3
-4
0
0.00005
0.0001
0.00015
0.0002

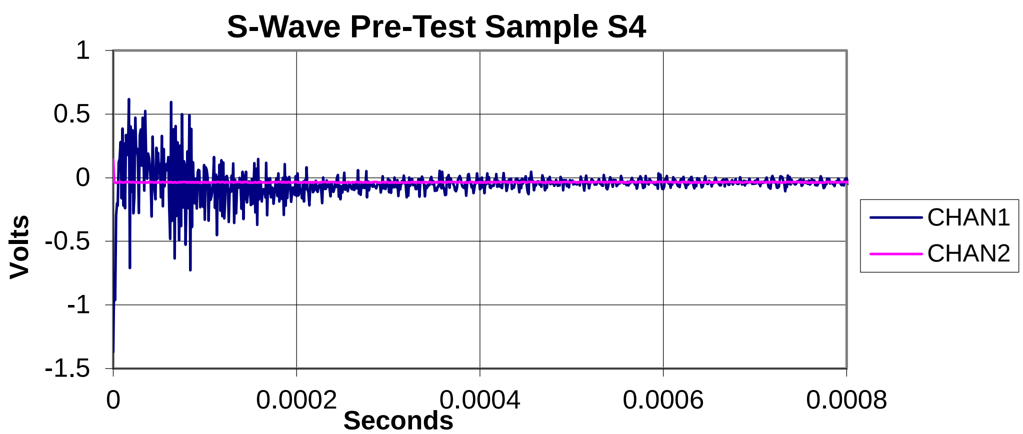

P-Wave During Uniaxial Compression Sample S2
Volts
Seconds
CHAN1
CHAN2
2
1.5
1
0.5
0
-0.5
-1
-1.5
-0.0001
0
0.0001
0.0002
0.0003
0.0004
0.0005
0.0006

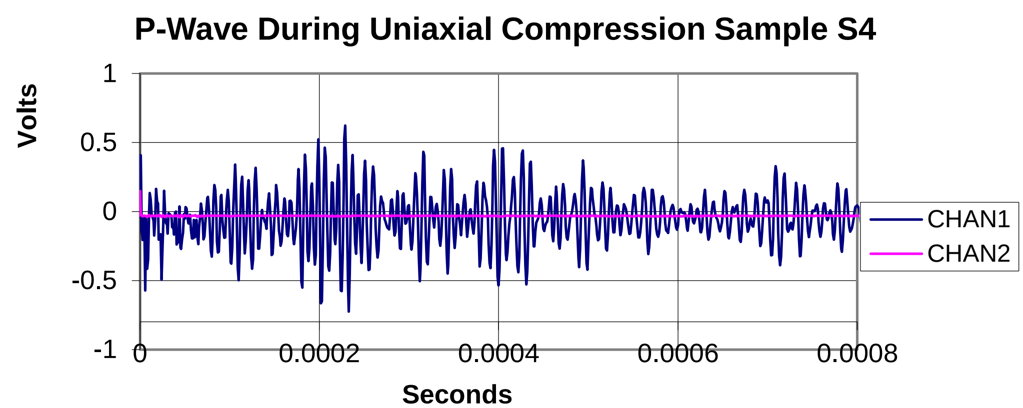

P-Wave Pre-test Sample S3
1.5
1
0.5
0
-0.5
-1
-1.5
-2
-2.5
-3
Volts
0
0.00005
0.0001
0.00015
Seconds
CHAN1
CHAN2

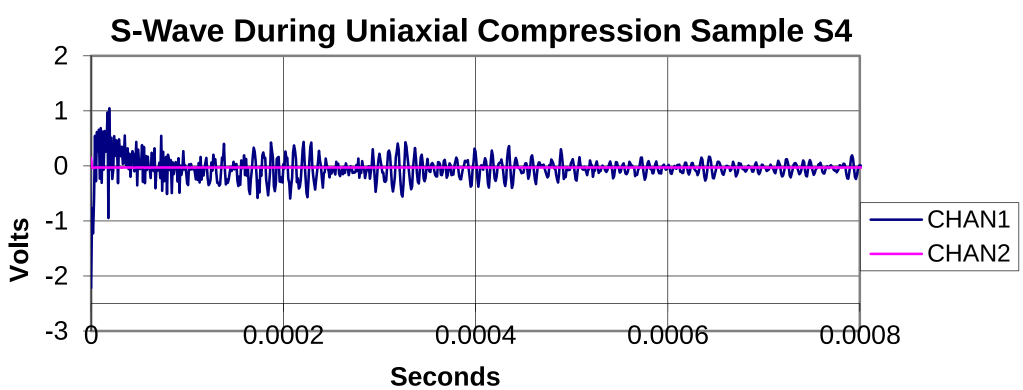

S-Wave Pre-test Sample S3
3
2
1
0
-1
-2
-3
Volts
0
0.00005
0.0001
0.00015
Seconds
CHAN1
CHAN2

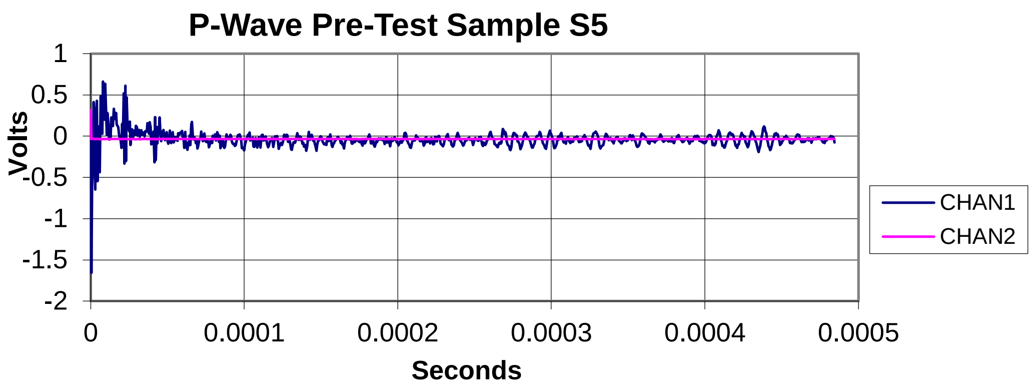

P-Wave Pre-test Sample S3
4
3
2
1
0
-1
-2
-3
-4
Volts
0
0.00002
0.00004
0.00006
0.00008
Seconds
CHAN1
CHAN2

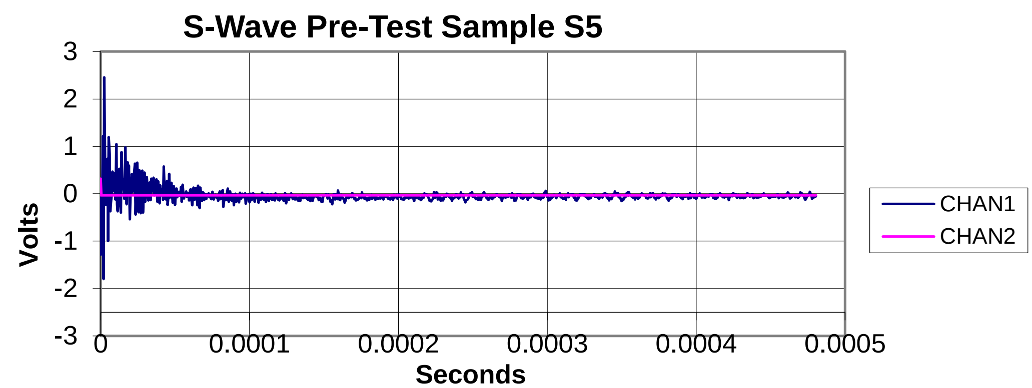

P-Wave Pre-Test Sample S4
Volts
Seconds
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
-1.2
0
0.0002
0.0004
0.0006
0.0008
CHAN1
CHAN2

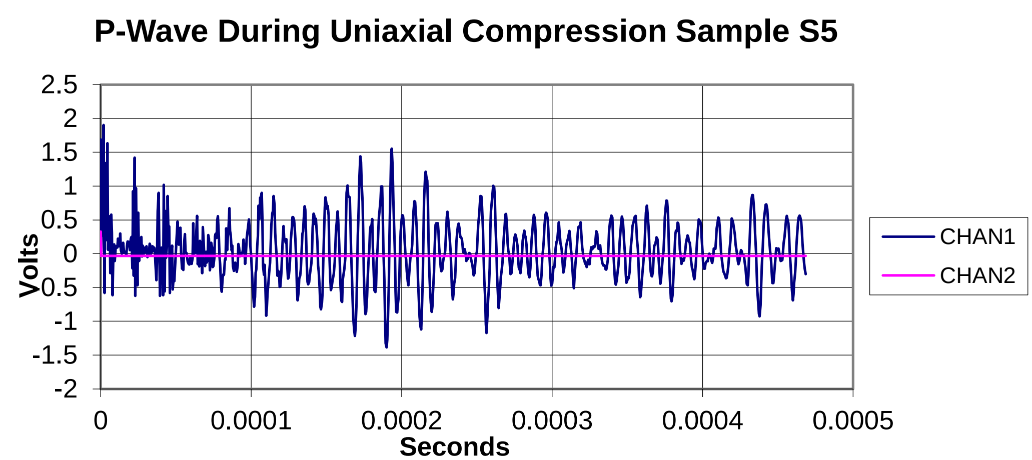

S-Wave Pre-Test Sample S4
Volts
Seconds
1
0.5
0
-0.5
-1
-1.5
0
0.0002
0.0004
0.0006
0.0008
CHAN1
CHAN2

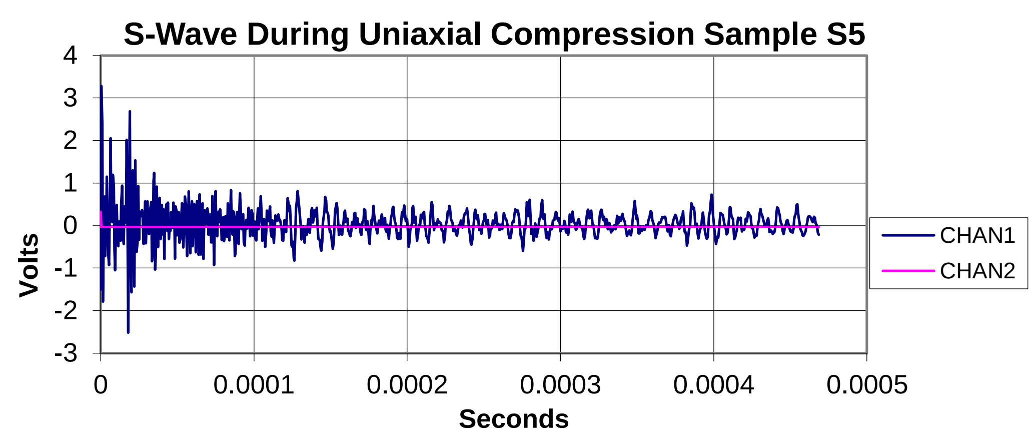

P-Wave During Uniaxial Compression Sample S4
Volts
Seconds
1
0.5
0
-0.5
-1
0
0.0002
0.0004
0.0006
0.0008
CHAN1
CHAN2

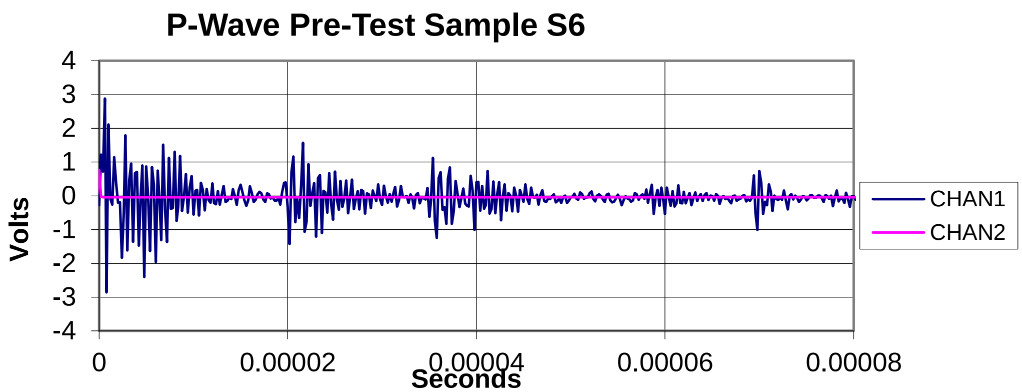

S-Wave During Uniaxial Compression Sample S4
Volts
Seconds
2
1
0
-1
-2
-3
0
0.0002
0.0004
0.0006
0.0008
CHAN1
CHAN2

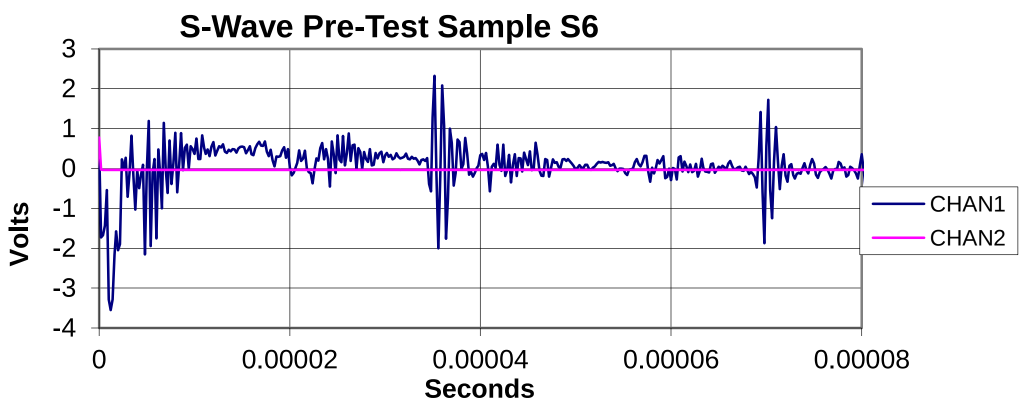

P-Wave Pre-Test Sample S5
Volts
1
0.5
0
-0.5
-1
-1.5
-2
0
0.0001
0.0002
0.0003
0.0004
0.0005
Seconds
CHAN1
CHAN2

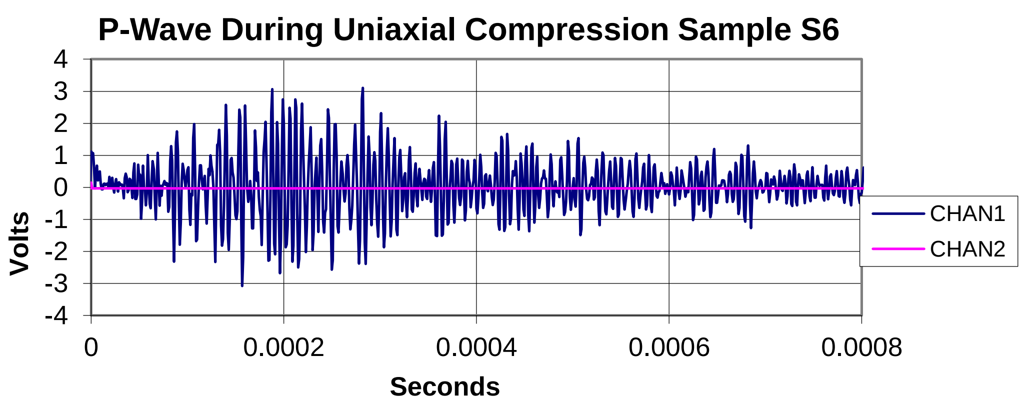

S-Wave Pre-Test Sample S5
Volts
3
2
1
0
-1
-2
-3
0
0.0001
0.0002
0.0003
0.0004
0.0005
Seconds
CHAN1
CHAN2

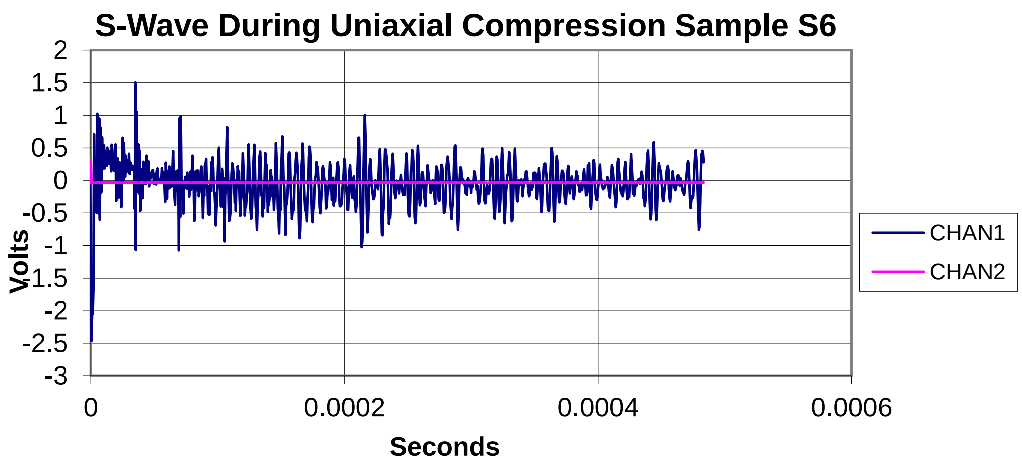

P-Wave During Uniaxial Compression Sample S5
Volts
2.5
2
1.5
1
0.5
0
-0.5
-1
-1.5
-2
0
0.0001
0.0002
0.0003
0.0004
0.0005
Seconds
CHAN1
CHAN2

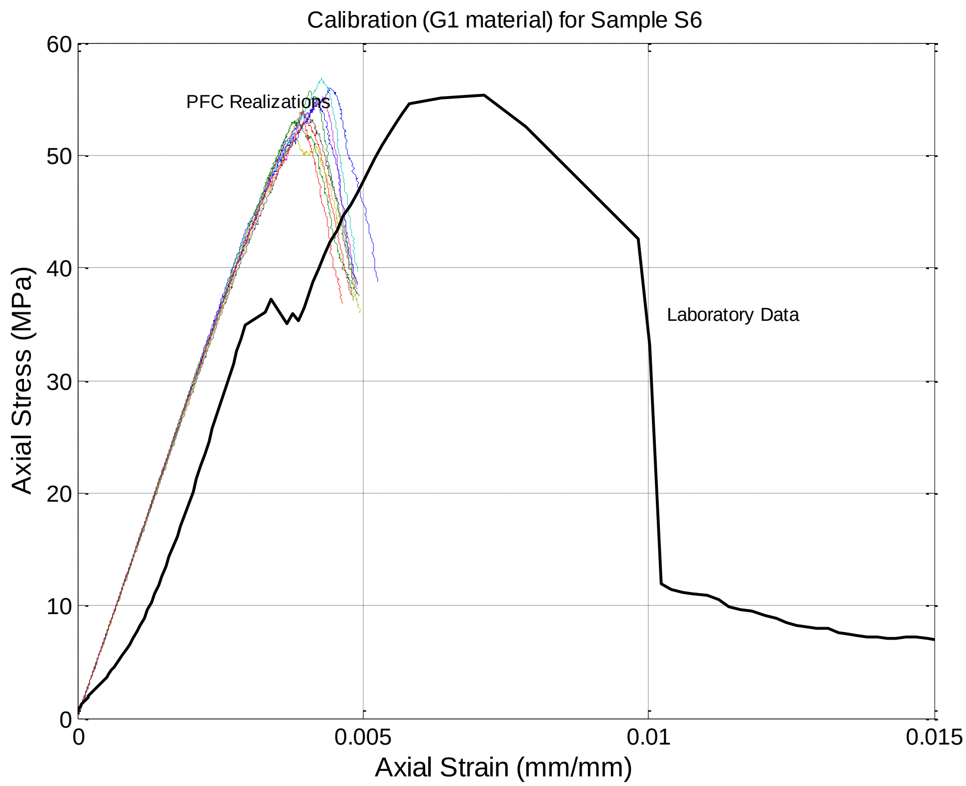

S-Wave During Uniaxial Compression Sample S5
Volts
4
3
2
1
0
-1
-2
-3
0
0.0001
0.0002
0.0003
0.0004
0.0005
Seconds
CHAN1
CHAN2


P-Wave Pre-Test Sample S6
Volts
Seconds
CHAN1
CHAN2
4
3
2
1
0
-1
-2
-3
-4
0
0.00002
0.00004
0.00006
0.00008

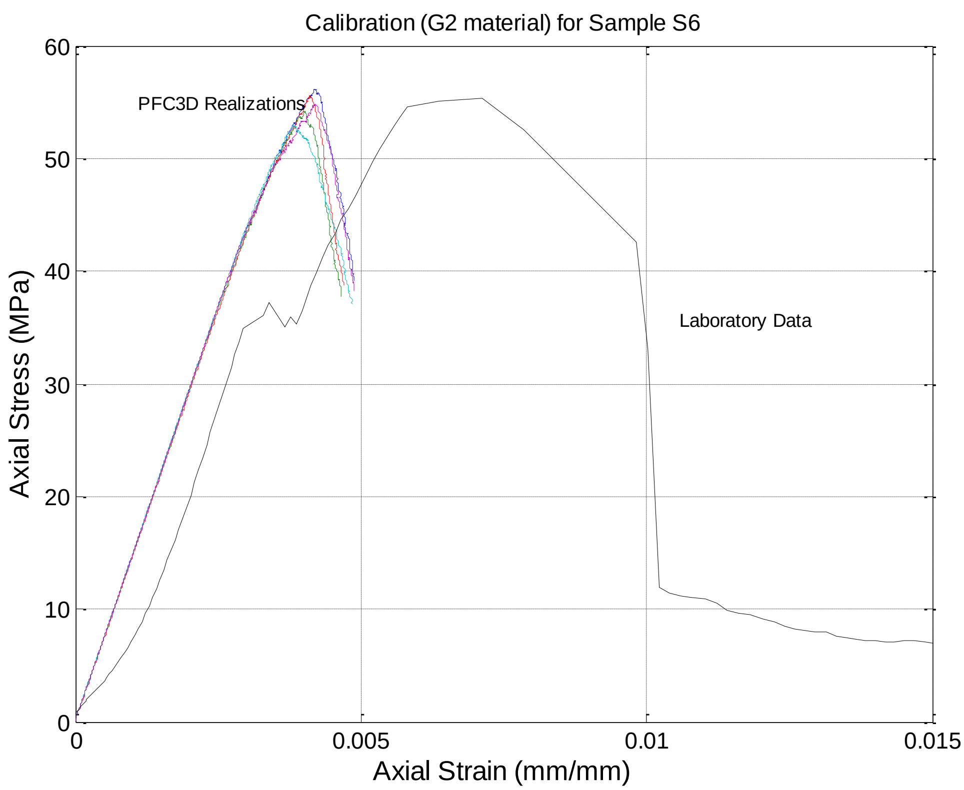

S-Wave Pre-Test Sample S6
Volts
Seconds
CHAN1
CHAN2
3
2
1
0
-1
-2
-3
-4
0
0.00002
0.00004
0.00006
0.00008


P-Wave During Uniaxial Compression Sample S6
Volts
Seconds
CHAN1
CHAN2
4
3
2
1
0
-1
-2
-3
-4
0
0.0002
0.0004
0.0006
0.0008


S-Wave During Uniaxial Compression Sample S6
Volts
Seconds
CHAN1
CHAN2
2
1.5
1
0.5
0
-0.5
-1
-1.5
-2
-2.5
-3
0
0.0002
0.0004
0.0006



Calibration for Material G1 with 10 realizations



Calibration for Material G2 with 5 realizations