University of Stavanger

Faculty of Science and Technology

MASTER'S THESIS

Study program/ Specialization: Mechanical and Structural Engineering and Materials science/Energy	Spring semester, 2012 Open
Writer: Gunn Elin Tøge	…… (Writer's signature)
Faculty supervisor: Bjørn Helge Hjertager External supervisor: Jørgen Osenbroch	
Title of thesis: The significance of Froude number in vertical pipes – A CFD study	
Credits (ECTS): 30	
Key words: VOF interFoam CFD OpenFOAM Froude number Vertical pipeline Transient	Pages: 88 + enclosure: 43 + CD Stavanger, 06.06.2012

ABSTRACT

The main objective is to develop a simulation model in OpenFOAM which can be used to study the transition from pressure to gravity driven flow in vertical pipelines. A pump is connected upstream, ensuring pressure driven flow while running. As the pump shuts down and water is drained a transition is expected to occur from pressure driven to gravity driven flow. A release valve is attached at the top of the vertical pipeline. It ensures air is induced as the pressure inside the pipeline decreases below the atmospheric pressure. The standard NORSOK P-001 (2006) recommends that vertical gravity lines shall be sized in accordance to a Froude number less than 0.3. An increased Froude number is expected to cause air entrainment and pulsations. In accordance to Osenbroch (2011) the origin of the design criteria is unknown, and to some extend it appears that a higher Froude number of 0.6-0.7 is used. A literature study shall therefore be performed with respect to relevant theory.
In OpenFOAM a two phase model named `interFoam` is adopted to study five vertical pipes of 20m height. Four cases are vented while the last is unvented. Three of the cases are associated with a 4" vent, while the main pipe diameter is set to respectively 12", 18" and 24". These changes are expected to indicate how the main pipe diameter affects the flow pattern. The fourth case is an 18" pipeline subjected to a 2" vent. An identical main pipe diameter is simulated without a vent. Three different vent designs of respective 4", 2" and unvented is therefore examined for an identical main pipe diameter of 18". Changes in vent design are expected to affect the flow pattern. The literature review examines the definition of the Froude number and its relationship to physical problems. A relevant topic regarding transient theory is incorporated. Generally it describes how flow is affected by a valve closure. The origin of the Froude number design criterion is also presented according to previous research. However, it is based on a scarce open literature.
Cases subjected to a 4" vent is fully drained during equal time intervals of 2 minutes. However, the total water volume differs due to the main pipe diameter of respectively 12", 18" and 24". Water must therefore drain at an increased flow rate as the main pipe diameter increases. All cases are characterized by a linear pressure decrease subsequent to the pump shut down. The presented transient theory describes this phenomenon. As the pump shuts down, pressure drops close to the pump due to flow retardation. This causes a negative pressure wave to propagate downstream. If pressure is sufficiently decreased vapor bubbles may form, which is also known as flashing. Propagation of a pressure wave is known as water hammer.
The Froude number criterion mentioned in the standard NORSOK P-001 (2006) appears to apply for draining from a process vessel. A Froude number less than 0.3 is recommended to ensure the vertical pipe is running full. If the Froude number is increased beyond 0.3, a weir is formed at the top surface and air is sucked into the liquid flow. However, this theory assumes a considerably wide process vessel. Transient theory seems to apply greater knowledge to future design, than does the Froude number. Water hammer may be the phenomenon which causes vibrations during shut down, rather than oscillations due to air entrainment. Water hammer is caused by abruptly changes in fluid flow, and it is the change in velocity which determines the magnitude of the liner pressure decrease. Rapid changes in flow velocity should therefore be avoided. It must be stated that solutions are found on very coarse meshes, which affects the accuracy of the results. Results are however assumed to indicate how the flow regime transforms and how it is affected by boundary conditions. The pump shut down in the simulation is also too rapid. In reality the flow must decrease during a finite time period.

ACKNOWLEDGEMENTS

This thesis is carried out in cooperation with Aker Solutions MMO. I would like to thank Aker Solutions MMO for giving me the opportunity to accomplish their proposed thesis. In addition, I would like to thank for their hospitality, assistance, and interest in students.
The thesis has been incredible challenging and exciting, and sometimes it has been too exciting. I would therefore like to thank my internal supervisor Bjørn Helge Hjertager for valuable suggestions and discussions, especially all the hours of work at his office regarding the development of the valve as a boundary condition. In addition, I thank my external supervisor Jørgen Osenbroch for his catching positive attitude about exploring the unknown. The task has been followed by ups and downs and I would therefore like to thank my friends and fellow students for their understanding and compassion. Special thanks are given to my cohabitant Håkon Albrektsen and my sister Anne Grete Tøge for their love and support.

TABLE OF CONTENTS

NOMENCLATURE

Latin symbols

a_N	Matrix coefficient of neighboring cell N	
a_P	Matric coefficient of cell P	
$a_{P,f}$	Matrix coefficient of cell P, face interpolated	
A	Cross sectional area	[m^2]
$[A]$	Square coefficient matrix	
$[A]_x$	Square coefficient matrix in x-direction	
$[A]_y$	Square coefficient matrix in y-direction	
$[A]_z$	Square coefficient matrix in z-direction	
$\mathcal{A}$	System of linear equations	
$\boldsymbol{b}$	Solution vector	
$\boldsymbol{b}_P$	Solution vector at cell P	
$\boldsymbol{b}_x$	Solution column vector of x-direction	
$\boldsymbol{b}_y$	Solution column vector of y-direction	
$\boldsymbol{b}_z$	Solution column vector of z-direction	
B	width normal to the paper plane	[m]
c	Sonic velocity	[m/s]
C_∞	Empirical coefficient	
C_0	Empirical coefficient	
Co	Courant number	
d	Internal pipe diameter	[inch]
D	Internal pipe diameter	[m]
E_v	Volume modulus of elasticity of the fluid	[kN/m^2]
F	Force	[N]
Fr	Froude number	
Fr_G	Densimetric gas Froude number	
Fr_L	Densimetric liquid Froude number	
g	Gravitational acceleration	[m/s^2]
$\boldsymbol{g}$	Vector of gravitational acceleration	[m/s^2]
g_G	Densimetric gravitational acceleration of gas	[m/s^2]
g_L	Densimetric gravitational acceleration of liquid	[m/s^2]
H	Height of liquid in a vessel	[m]
$\boldsymbol{i}$	Unit vector in x-direction	
l	Distance	[m]
L	Linear dimension significant for the flow pattern	[m]
L_p	Pipeline length between valve and reservoir	[m]
m	Mass	[kg]
n	Normal to the interface	
p	Pressure	[Pa]
p_d	Dynamic pressure	[Pa]
p_f	Pressure, face interpolated	[Pa]
p_s	Static pressure	[Pa]
P	Total pressure $P = p_d + \rho \boldsymbol{g} \cdot \boldsymbol{X}$	[Pa]
Q	Liquid flow rate	[USgpm]
S	Minimum depth of liquid	[inch]
$\boldsymbol{S_f}$	Surface vector, face interpolated	[m^2]
$\boldsymbol{S}_f^*$	Normalized surface vector of magnitude 1, face interpolated	[m^2]
t	Time	[s]

t_c	Valve closure time less than the pipe period	[s]
T_r	Pipe period	[s]
$\boldsymbol{U}$	Velocity vector	[m/s]
$\boldsymbol{U}^*$	Normalized velocity vector of magnitude 1	[m/s]
$\boldsymbol{U_f}$	Velocity vector, face interpolated	[m/s]
$\boldsymbol{U}_f^*$	Normalized velocity vector of magnitude 1, face interpolated	[m/s]
$\boldsymbol{U}_N$	Velocity vector of neighboring cell N	[m/s]
$\boldsymbol{U}_P$	Velocity vector of cell P	[m/s]
$\boldsymbol{U}_P^*$	Normalized velocity vector of magnitude 1 at cell P	[m/s]
$\boldsymbol{U_x}$	Velocity column vector in x-direction	[m/s]
$\boldsymbol{U_y}$	Velocity column vector in y-direction	[m/s]
$\boldsymbol{U_z}$	Velocity column vector in z-direction	[m/s]
V	Fluid velocity	[m/s]
V_B	Bubble rise velocity in a stagnant liquid	[m/s]
$V_{B,Abs}$	Absolute bubble rise velocity in a flowing liquid	[m/s]
V_G	Superficial gas velocity	[m/s]
V_L	Superficial liquid velocity	[m/s]
V_M	Average slug velocity	[m/s]
V_s	Lazy slug rise velocity	[m/s]
V_1	Velocity into a control volume	[m/s]
V_2	Velocity out of a control volume	[m/s]
$\dot{V}$	Volume flow rate	[m^3/s]
$\boldsymbol{W}$	Artificial velocity vector	[m/s]
$\boldsymbol{X}$	Position vector	[m]
$\boldsymbol{X_f}$	Position vector, face interpolated	[m]
$\boldsymbol{X_p}$	Position vector at cell P	[m]
x_0	Distance at which maximum pressure occurs	[m]
Z	Height of fluid flow	[m]

Greek symbols

α	Volume fraction	
δ	An infinitesimal change	
Δ	A macroscopic change	
μ	Dynamic viscosity	[Pa·s]
μ_{air}	Dynamic viscosity of air	[Pa·s]
μ_{eff}	Effective dynamic viscosity	[Pa·s]
μ_t	Turbulent dynamic viscosity	[Pa·s]
μ_{water}	Dynamic viscosity of water	[Pa·s]
ρ	Density	[kg/m^3]
ρ_{air}	Density of air	[kg/m^3]
ρ_G	Density of gas	[kg/m^3]
ρ_L	Density of liquid	[kg/m^3]
ρ_{water}	Density of water	[kg/m^3]
φ	Face flux	[m^3/s]
φ^*	Normalized face flux of magnitude 1	[m^3/s]
φ'	Corrected face flux	[m^3/s]

Definitions & abbreviations

AFS	Annular Flow Stabilizer
DIC	Diagonal incomplete-Cholesky
DILU	Diagonal incomplete-LU
LESModel	Large eddy simulation modeling
MULES	Multidimensional universal limiter for explicit solution
NVD	Normalized variable diagram
PBiCG	Preconditioned bi-conjugate gradient
PCG	Preconditioned conjugate gradient
RASModel	Reynolds-averaged stress modeling
TVD	Total variation diminishing
VOF	Volume of fluid

1 INTRODUCTION

In this chapter, the reader is introduced to the background of the thesis. Work done by previous scientists and the thesis objective is presented.

1.1 Background

At offshore installations draining commonly occurs, such as for produced water or cooling water. In accordance to Osenbroch (2011) draining typically occurs through a vertical pipeline associated with a pump and a release valve. The release valve shall ensure air is induced as the pressure inside the pipeline decreases below the atmospheric pressure. When the pump runs the flow is pressure driven. If the pump shuts down, a transition is assumed to occur from pressure driven to gravity driven flow. A special design criteria is required for gravity flow in vertical pipelines according to the standard NORSOK P-001 Process Design (2006). It requires that "*vertical gravity lines with or without submerged outlets shall be designed such that the Froude number is less than 0,3. This is to avoid air entrainment and ensure undisturbed flow without pulsations*". In addition the standard requires a vented line to be connected at the top of the vertical gravity line. This is to avoid vacuum, flashing or pulsations. It is required that the air volume flow rate from the vent line is equal to the water volume flow rate in the vertical pipe. The vent line shall be designed for a maximum pressure loss of 0,005bar. According to Osenbroch (2011) the origin of the design criteria is unknown, and to some extend it appears that a higher Froude number of 0.6-0.7 is used.

1.2 Previous work

This section is partly retrieved from Moon et al. (1987). In the years before 1985 most research in the field is concentrated on countercurrent and co-current upward two-phase flow in vertical pipes. In this period there has been little interest in downward flow. A.E. Dukler is known for his work regarding annular film thickness in 1960. In 1968 L.L. Simpson presented his work on the flooding transition for downward air-water flow with and without submerged outlets in terms of the densimetric Froude number. He observed that sufficient air entrainment produces slugs which rise up the vertical pipe countercurrent to the water flow.

At the time of the written article (Moon et al., 1987) no work has been performed to ensure that the Froude number sufficiently describes the flooding transition at altered pipe diameters. T. Oshinowo and M.E. Charles also concluded in 1974 that the Froude number would be a useful dimensionless parameter. This because the Froude number represents the ratio of inertial to gravitational forces which both are important in most flow patterns of two-phase flows. This view is widely supported, and the dimensionless parameters of densimetric gas- and liquid Froude number are well established in the characterization of countercurrent flow. B.K. Kozlov was the first to introduce these dimensionless parameters in 1954. A.G. Kelly is known for his studies in 1965 of minimum siphoning velocities for liquid-vapor systems. Siphoning denotes to draw off through a siphon. A siphon is a tube of an inverted U shape such that atmospheric pressure is sufficient to transfer liquid from one level to the next through a barrier higher than the first level. This definition is retrieved from Farlex (2004). Kelly was the only one to suggest the use of a constant Froude number in correlation to experimental results for co-current down flow. K.W. McQuillan in 1985 and G.B. Wallis in 1969 both studied flooding in counter current and upward flow in pipes. They both conclude that the mechanism of flooding involves bridging of the flow tube by large disturbance waves on the surface of the liquid film. D. Barnea supports this finding in 1981, and presents that such waves are formed for Fr in the range of 0.5-2.2. In 1976 Thwaites investigated waves in co-current down flow and established two types of waves, ripples and roller-waves. The roller-waves can only occur at high fluid flows. Co-current down flow was also investigated by

T.M. Verghese in 1985, and he suggested that the wave mechanism may be responsible for the flooding transition.
Previously performed work is further described in accordance to Thorpe at al. (1989). In 1983 A. E. Dukler detected waves much less than the pipe diameter, and he therefore disagreed that the wave mechanism caused flooding transition. N. G. McDuffie was one of many engineers who in 1979 examined the weir type entry in detail. The results are similar to what other scientists have observed. R. J. Keller is also known for the investigation of the weir type entry for top initiated flooding in 1977. In 1969 G. B. Wallis investigated the interfacial friction between liquid and gas, which is known to be small. He claims that the interfacial friction is a result of the form drag on the surface waves. The interfacial friction was found to be dependent on the ratio of film thickness to internal diameter. This was supported by D. Bharathan in 1978. In 1983 A. E. Dukler found a dependence on the gas flow rate, which at low gas Reynolds number caused a more dramatic variation for co-current up flow.

1.3 Thesis objective

The thesis objective is formed in accordance to Osenbroch (2011). The objective is to develop a simulation model in OpenFOAM, which can be used to study the transition from pressure to gravity driven flow in vertical pipelines as the pump is shut down. Different cases shall be examined where the boundary conditions remains almost unchanged and the geometry changes. It is mainly the diameter of the vent line and the diameter of the main pipeline which shall be varied, in addition to removal of the vent. Generally it can be said to be ten or more cases that could be relevant to study, but the task will be reduced to an extent that is possible to achieve. The thesis is seen as a research since the model must be developed, and it cannot be decided in advance how many cases it will result in. It is desirable to study how the changes in dimensions affect the transition. The Froude number shall be a key parameter in the analysis. In addition it has proven to be difficult to find the underlying theory for the design criteria. A literature study shall therefore be performed with respect to relevant theory and previous work.

2 LITTERATURE REVIEW

This chapter presents the Froude number and its application in different fluid dynamic problems. The relevance of transient theory due to a sudden pump shut down and opening and closing of a valve is outlined. At last an attempt is made to describe the Froude design criteria based on a scarce open literature.

2.1 Froude number

Subsequent section is taken from Finnemore et al. (2002). The Froude number is named after the naval architect William Froude who studied the resistance of ships subjected to waves. The Froude number represents the ratio of inertia to gravitational forces, which generally is presented as

(2.1)

$$\frac{\rho L^2 V^2}{\rho g L^3} = \frac{V^2}{gL}$$

Inertia and gravitational forces occurs among others in the related topic open channel flow. The liner dimension L is significant for the flow pattern. Commonly the Froude number is expressed by taking the square root, to achieve velocity of first power.

(2.2)

$$Fr = \frac{V}{\sqrt{gL}}$$

The Froude number is also known to be the ratio of flow speed to wave speed in accordance to Douglas et al. (2005). Pressure is constant from one point to the next at the free surface of an open channel flow. Thus if flow is disturbed, it will cause a surface wave rather than a pressure wave, see Figure 2.1.

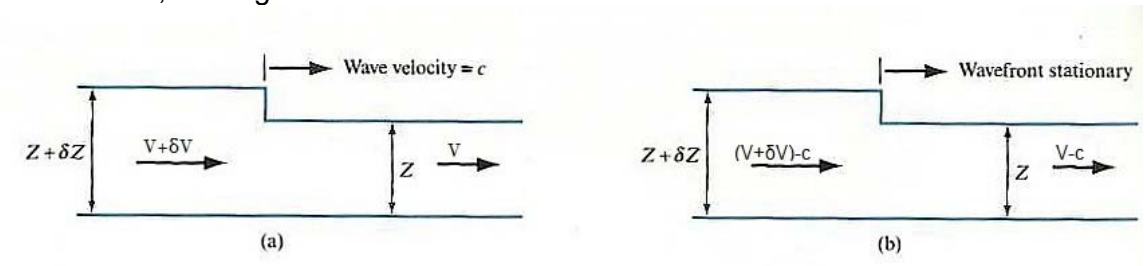

Figure 2.1 Surface wave. a) Unsteady flow observed from a stationary point of view, b) steady flow observed by a moving observer (Douglas et al., 2005).

The wave velocity is defined as c while the velocity of the remaining fluid flow is defined as V. If an observer moves with velocity equal to the wave velocity, the condition will appear as steady. The wave velocity can be derived by use of conservation of mass and Newton's second law. The mass flow rate on the left side of the wave front must equal that on the right side. Conservation of mass gives

(2.3)

$$\rho B(Z + \delta Z)(V + \delta V - c) = \rho BZ(V - c)$$

Here B is the width normal to the paper plane. The density is constant on both sides of the wave front, and therefore cancels each other. Equation (2.3) is simplified to

(2.4)

$$Z\delta V + V\delta Z + \delta Z\delta V - c\delta Z = 0$$

Rearranging

(2.5)

$$(c - V)\delta Z = (Z + \delta Z)\delta V$$

As the wave increases the liquid level produces a hydrostatic force which again causes a higher wave velocity. The hydrostatic force due to the increased level δZ must equal the mass flow rate times the change in velocity. Newtons second law is applied.

(2.6)

$$\rho g \delta Z B Z = \rho B Z (V - c)(-\delta V)$$

Rearranging

(2.7)

$$\delta V = \frac{g\delta Z}{c - V}$$

The fluid velocity is substituted from equation (2.5)

(2.8)

$$\frac{(c - V)\delta Z}{Z + \delta Z} = \frac{g\delta Z}{c - V}$$

Rearranging

(2.9)

$$(c - V)^2 = (Z + \delta Z)g$$

Assuming the wave height is small

(2.10)

$$(c - V)^2 = gZ$$

The velocity of the surface wave relative to the resulting fluid is

(2.11)

$$c - V = \sqrt{gZ}$$

The velocity downstream is set as positive direction. From a stationary point of view the wave velocity downstream equals $\sqrt{gZ} + V$. If the wave is travelling upstream the wave velocity equals $\sqrt{gZ} - V$. If the fluid velocity V is greater than the wave velocity $\sqrt{gZ}$, the wave cannot flow in the upstream direction, when viewed from the stationary location. Contradictory, if the fluid velocity V is less than the wave velocity $\sqrt{gZ}$, the surface wave propagates in both

directions. The Froude number can be defined as the ratio of fluid velocity to the surface wave velocity $c - V$.

(2.12)

$$Fr = \frac{V}{c - V} = \frac{V}{\sqrt{gZ}}$$

A Froude number equal to 1 indicates a stationary surface wave.

2.2 Design criteria

Design of pipelines subjected to gravity flow can be described in accordance to myChemE (2011). Gravity flow is subjected to a limited pressured drop in contrast to pipelines with an associated pump in service. For water flowing only due to gravity, it is the difference in height which limits the pressure drop. In pump systems one can simply change the pump specification to allow a wider range of pressure drops. High discharge pressure or excessive pipe lengths are inconvenient because the gravity flow cannot adapt these obstacles. In addition there is the possibility to achieve air entrainment. If this occurs, even higher pressure drops are obtained which in turn reduces the water flow rate. These disadvantages are the reason why gravity flows are designed with a larger pipe size. It is generally recommended to avoid air entrainment. This can be obtained in two different ways.

2.2.1 Connect to a storage tank

If the gravity line is equipped with a storage tank, a sufficient water depth can avoid air entrainment at the discharge (myChemE, 2011). A control valve can be utilized such as to maintain the water level. An equation expressed in US units provides the minimum liquid level required and is developed by the Hydraulic Institute. The equation is valid for velocities between 2 ft/s and 8 ft/s, or for volume flow rates between 25-300000 USgpm.

(2.13)

$$S = d + \frac{0.574 * Q}{d^{1.5}}$$

2.2.2 Froude number design criteria

If it is not possible to connect a storage tank, another opportunity is to design the vertical pipeline for self-venting (myChemE, 2011). This imply water velocities low enough such that any slugs of air at discharge can rise up the pipeline counter current to the water flow. Self-venting is found to occur for a Froude number less than 0,3. Thus the limiting Froude number can be efficiently achieved by increasing the pipe diameter and decrease the water velocity.

(2.14)

$$\frac{V}{\sqrt{gD}} < 0.3$$

2.3 Open channel flow – relation to the Froude number

Subsequent sections is described by Finnemore et al. (2002). Open channel flow is characterized by a free surface to the atmosphere. Thus the flowing water is not completely enclosed by solid boundaries. This causes water to flow due to gravity only, in contrast to an external pressure. Other designations of open channel flow are free-surface flow and gravity flow. Systems such as sewers, tunnels, channels build for drainage or water supply and pipelines not completely filled with water are examples of open channel flow problems. Velocities are typically highest close to the free surface, and decreases to zero at walls due to the no-slip condition. Open channel flow is basically based on experiments performed on water at natural temperatures. Few experiments are performed on other liquids. A brief description is presented with its associated relation to the Froude number.

2.3.1 Steady flow in open channels

Generally steady flow in open channels can be described in accordance to Cengel and Cimbala (2006). Steady flow in open channels remains unchanged at a specific location and does not vary with time. It is the flow depth or the average velocity that may differ with time and along the channel. Therefore steady flow indicates a constant depth which does not vary with time. The depth may however differ from one location to the next. Distinction is made between uniform and non-uniform steady flow. The difference is the variation of flow depth along the channel. In uniform flow the depth remains constant, which may occur in long sections with additional constant cross section and slope. The non-uniform steady flow is characterized by a varied flow depth in the flow direction. Change in cross section and slope, and the presence of an obstruction are examples which cause non-uniform flow. Much of the applied theories in this topic relates to horizontal channels which either are closed conduits not running full or gutters which are free to atmosphere along the pipe length. Such theory will not be presented as it is seen as having little relevance to the actual case.

2.3.2 Tranquil and rapid flow

In accordance to Cengel and Cimbala (2006) and Douglas et al. (2005) the Froude number is used to classify open channel flow as either tranquil, critical or rapid. Tranquil and rapid flows are also known as respectively subcritical and supercritical. If the Froude number is less than 1, flow is said to be tranquil and a value higher than 1 indicates rapid flow. Critical flow occurs when the Froude number equals 1. As explained earlier, the Froude number is described as the ratio of inertia to gravitational forces in accordance to equation (2.1). The denominator can be expressed as $2(\frac{1}{2}\rho V^2 L^2)$ where L^2 represents the cross sectional area. This is similar to multiplying the dynamic pressure $\frac{1}{2}\rho V^2$ with the cross sectional area. Rapid flow is therefore said to be dominated by inertia forces, while the tranquil flow is dominated by gravity forces. Tranquil flow is characterized by a deep slow flow. In this state, the fluid velocity is less than the distributed disturbance velocity defined equal to the relative surface wave velocity in equation (2.11). Disturbances are therefore enabled to propagate both upstream and downstream, causing the downstream conditions to control the flow behavior. Rapid flow is referred to as shooting. The fluid flow now exceeds the distributed disturbance velocity defined in equation (2.11). Upstream disturbances can therefore not occur, and the flow is no longer controlled by downstream conditions. The flow is controlled by upstream conditions, because fluid velocity V is larger than the disturbance wave velocity c. The velocity of the down flow equals V-c as seen from a stationary location. Thus rapid flow is controlled by upstream conditions. At Froude number equal to 1 the surface wave is stationary.

2.4 Transient flow problems

In accordance to Douglas et al. (2005) and Finnemore et al. (2002) velocity and depth changes with time relative to a fixed point when considering transient non-uniform flow. Generally one distinguishes between gradually and rapid varied transient flow. Examples of gradually varied flows are flood waves, tidal flow, gradually change of water level in a reservoir and gradually varied geometries such as gates. Rapidly varied flow includes pulsating flow, also known as roll waves. It may be caused by very steep slopes, fast closing and opening of a valve, surges or surge waves. Surge waves are also designated moving hydraulic jumps. Sudden change in operational controls or a sudden failure may both cause surges and surge waves. It is the rate of change of flow conditions that determines the magnitude of propagated surge.

2.4.1 Instantaneous valve closure

The mechanism of pressure propagation can be described by the sudden closure of a valve (Douglas et al., 2005). A valve is placed on a horizontal pipe between two reservoirs, Figure 2.2.

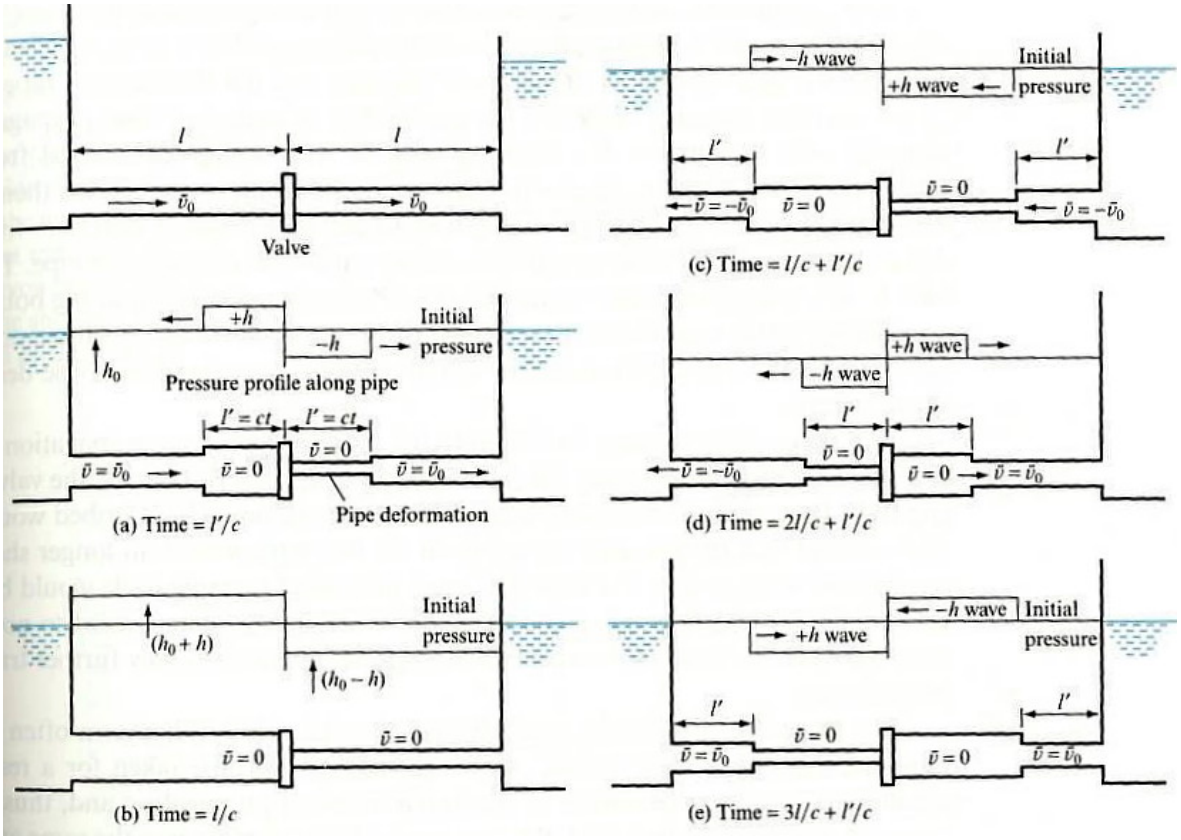

Figure 2.2 Pressure propagation and the pipe wall deformation upstream and downstream a valve closure (Douglas et al., 2005).

Friction in the pipeline is neglected, and the pipeline on each side of the valve is considered equal in size. When the valve closes, the upstream fluid is compressed and it causes the pipe walls to expand. A pressure wave is produced due to the increased pressure and it propagates in the upstream direction with a velocity equal to the sonic. The pressure wave results in a delay of fluid flow upstream. Downstream, pressure is decreased close to the valve due to flow retardation. This causes a negative pressure wave to propagate downstream, resulting in a delay of fluid flow. The negative pressure wave causes the pipe walls to contract. It is assumed that pressure does not decrease below fluid vapor pressure and that vapor bubbles do not form.

The wave front and the deformation are moved a distance l'=ct at time t, see Figure 2.2 a). Here c is the sonic velocity. At the time t=l/c both pressure waves reaches the reservoirs, see Figure 2.2 b). At the discharge to the reservoir pressure must equal that due to the depth in the reservoir. An unbalance occurs, causing a pressure wave sufficient to retain the pressure in front of the valve to be submitted back into the pipelines. The pressure wave introduced in the pipeline upstream reduces the pressure such that the pipe wall contracts again. Pressure in front of the reduced pressure wave is subject to an increased pressure causing fluid to flow back to the reservoir. Since friction is neglected, the reversed flow to the reservoir must be equal in magnitude to the initial velocity, Figure 2.2 c). In the pipeline downstream, the opposite occurs. A positive pressure wave propagates into the pipeline and causes fluid from the reservoir to flow into the pipeline, Figure 2.2 c). The restoring wave reaches the valve at t=2l/c. In front of the wave close to the valve upstream, there is no fluid to support the reversed flow from the reservoir, and a negative pressure occurs. The negative pressure wave propagates through the upstream pipeline causing contraction of walls and delay of fluid flow, Figure 2.2 d). Since it is assumed that pressure will not drop to vapor pressure and no friction occurs, pressure of the propagating wave must equal the difference obtained at valve closure. The pressure increment is named h, and all pressure waves are assumed to equal a pressure of ±h, Figure 2.2. Similar occurs in the downstream pipeline. The restoring pressure wave causes an increased pressure close to the valve at t=2l/c. A positive pressure of +h is propagated downstream and approaches the reservoir at t=3l/c. An unbalance occurs again at the discharge to the reservoir, and a restoring pressure wave propagates into the downstream pipeline. Similar, but opposite occurs at the upstream pipeline, Figure 2.2 e). Both pressure waves arrives again at the valve at t=4l/c and the conditions are now equal those at t=0. The valve is still closed, and the cycle will repeat. After a time of 4l/c a whole pressure cycle is covered. It is also known as a pipe period in transient analysis.
Pressure waves can be reduced in magnitude by increasing the time at which the valve closes. If the valve closure time increases beyond the pipe period it is called slow, and at times less than the pipe period it is called rapid. Theoretically the cycles can repeat itself infinitely many times, but friction will be present causing damped pressure oscillations. Friction must be considered if frictional losses are significant. These losses may cause increased pressure waves. If pressure is decreased to vapor pressure, vapor cavity is formed. When pressure increases again the cavity collapses. Generally this phenomenon is known as column separation.

2.4.2 Pressure wave velocity

In accordance to Finnemore et al. (2002) the sonic velocity of a pressure wave is given by

$$c = \sqrt{\frac{E_v}{\rho}} \tag{2.15}$$

Volume modulus of elasticity E_v for water is about 2,07*10^6 kN/m^2. The pressure wave velocity in water is typically 1440 m/s, but it varies with temperatures. If the pipe is made of an elastic material such as rubber the pressure wave velocity is reduced. A small quantity of free gas in the fluid also tends to decrease the wave velocity.

2.4.3 Water hammer

The topic water hammer is described by Finnemore et al. (2002). An abruptly change in fluid flow, such as when a valve is opened or closed rapidly, can cause water hammer. The valve may also be partially open. Water hammer can apply great damage to hydraulic systems. Generally Cengel and Cimbala (2006) imply that water hammer causes the pipe to vibrate and is characterized by the sound of a pipe being hammered.

Further theory is described in accordance to Finnemore et al. (2002). Assume a fluid flow upstream an instantaneous valve closure, Figure 2.3. At valve closure, a pressure wave with increased pressure propagates upstream.

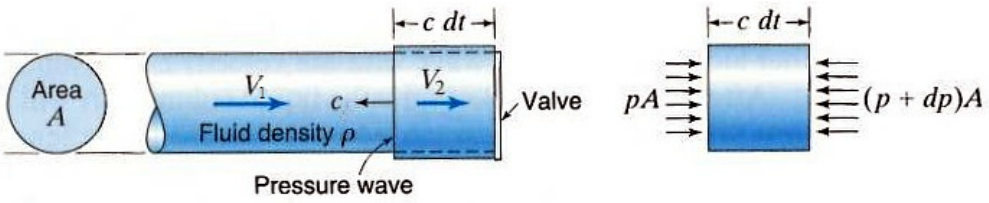

Figure 2.3 Fluid upstream an instantaneous closed valve (Finnemore et al., 2002).

As explained earlier for a sudden valve closure a pressure wave is moved the following distance after a short time Δt

(2.16)

$$l' = c\Delta t$$

Newtons second law implies

(2.17)

$$F\Delta t = m\Delta V = m(V_2 - V_1)$$

Friction is neglected. Mass during the same time can be expressed as

(2.18)

$$m = \rho A c \Delta t$$

When considering a fluid element close to the valve as seen in Figure 2.3, equilibrium gives

(2.19)

$$(pA - (p + \Delta p)A)\Delta t = (\rho A c \Delta t)(V_2 - V_1)$$

The change in pressure due to the valve closure is

(2.20)

$$\Delta p = \rho c(V_1 - V_2)$$

It is therefore the change in velocity which determines the increase in pressure. In the case of a totally closed valve V_2 is zero. Similarly, if flow downstream of the closed valve is studied V_1 is zero.
Rapid closure of the valve can never occur in reality. Consider the setup in Figure 2.4. The pipe period required for a pressure wave to travel from the valve to the reservoir and back again can be defined as

(2.21)

$$T_r = \frac{2L_p}{c}$$

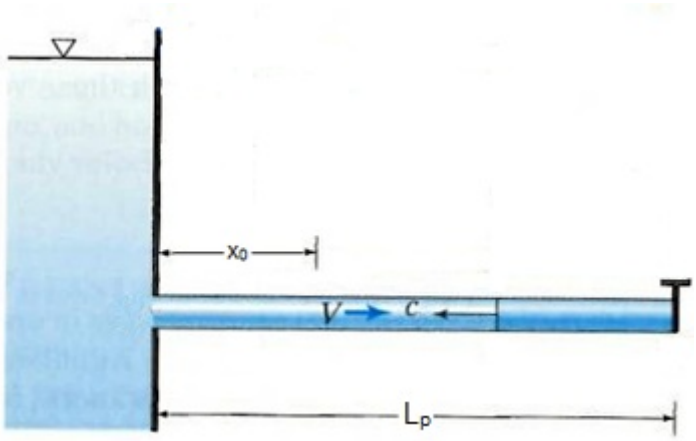

Figure 2.4 Valve closure at the pipe end (Finnemore et al., 2002).

A real rapid valve is examined. The closure time is larger than zero, but limited to t_c which is less than the pipe period, Figure 2.5.

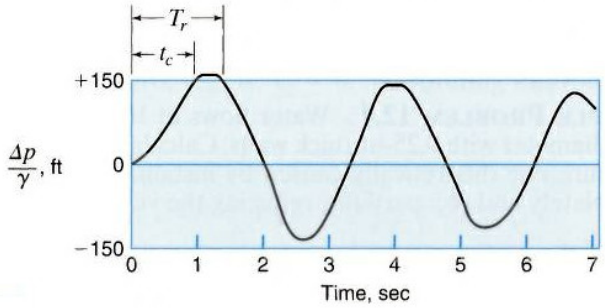

Figure 2.5 Rapid closure of a valve (Finnemore et al., 2002).

The slope of the curve is determined by the closure time t_c and its effect on the fluid velocity. If the closure time is zero, which is not physical, the sloped lines would be vertical. In both situations the maximum pressure is the same. The only difference is the duration of the maximum pressure, which is T_r -t_c in Figure 2.5, in contrast to the whole pipe period when the valve closure time is zero. If the valve closure time exactly equals T_r, the maximum pressure persists only at an instant. This would produce v-shaped peaks. At points close to the reservoir, the time cycle for the pressure wave is shorter than at locations close to the valve. This is because the wave turns at the discharge, heading back into the pipeline. Thus points close to the discharge has recently experienced the pressure wave on its way towards the reservoir. Thus when the pressure wave velocity and the closure time is known one can calculate the distance at which the maximum pressure occurs, see also Figure 2.5.

(2.22)

$$x_0 = \frac{ct_c}{2}$$

At points closer to the discharge than the distance x_0, the time cycle for the pressure wave is shorter than t_c. A time equal to t_c is still required to experience the maximum pressure. Therefore points closer to the discharge than the distance x_0, will not experience the maximum pressure. The maximum water hammer pressure will therefore be less at these locations. In this region pressure will vary from maximum at x_0 to zero at the discharge. A linear pressure variation is often assumed, but it is generally determined by how the valve is

opened. At all locations further away from the discharge, $x>x_0$, the maximum pressure will be experienced.
Slow closure indicates a closure time greater than the pipe period given by equation (2.21). The maximum pressure at slow closure is less than that obtained at rapid closure. In this case the restoring pressure wave reaches the valve before it closes, and prevents the pressure to increase further, Figure 2.6.

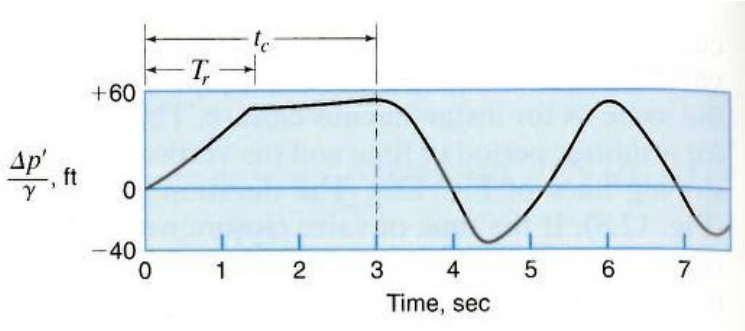

Figure 2.6 Slow closure of a valve (Finnemore et al., 2002).

The pressure increases linearly until the pressure wave reaches the valve at T_r. Maximum pressure given by equation (2.20) is never achieved. The maximum pressure experienced at slow valve closure is related to rapid closure in the following manner.

(2.23)

$$\frac{\Delta p'}{\Delta p} \approx \frac{L_p}{x_0}$$

By inserting equation (2.20) and (2.22), a slow valve closure can be expressed as

(2.24)

$$\Delta p' = \frac{L_p}{x_0}\Delta p = \frac{2L_p}{ct_c} * \rho c(V_1 - V_2) = \frac{2L_p\rho(V_1 - V_2)}{t_c}$$

Elastic pressure waves are produced after the restoring pressure reaches the valve. These are only referred to as complex, and are not described further.
A variety of devices are used as protection against water hammer. Generally the attempt is to keep the pressure within desired limits. The flow can be controlled by a slow valve closure or a by-pass valve can be used to prevent sudden changes in flow. It is possible to connect an air valve when the pressure is reduced below a certain limit. The air valve admits air into the pipeline causing the pressure to increase again. If the pressure is greater than a certain limit, due to fluid compression, a relief valve can be used to allow water to escape. Often these are initiated by pressure conditions, and the flow control will only work if the reflection time T_r is sufficiently long.

2.4.4 Surge control

Surge control is described by Douglas et al. (2005). The pressure drop or pressure increase caused by a sudden change in operational conditions may be referred to as surge. The main purpose of surge control devices is to reduce the change in transient conditions during a pump shut down and opening or closing of a valve. Surge is best controlled by placing the surge control device close to the transient source. Surge cannot be avoided in all system applications due to cost, design information or operational requirements. Controls that suppress surge are briefly described. As earlier mentioned, slow valve closure can reduce the surge pressure. Column separation may be a result of pump failure or shut down downstream of the pump. If flow rate is gradually reduced, by keeping the pump in a run-

down period, the possibility of column separations would be reduced. It can be achieved by installing a flywheel into the pump, but as a drawback it must run during start up. Therefore it is not a good solution. An inflow release valve may be used to restore an atmospheric pressure in the pipeline downstream of a valve closure or a pump shut down. The valve may be placed close to the location where column separation is expected to occur, and thus recover the atmospheric pressure. In some systems the introduction of air to the system may be troublesome during restart. Alternatively fluid can be introduced to the system. Bypass systems allows water from a sump to bypass to the downstream section of the pump when pressure is reduced below sump pressure.

2.5 Two phase flow

2.5.1 Slug flow

In accordance to Nolte (1978) bubbles can develop countercurrent to the downward liquid flow. A bubble may form such that it partially blocks the channel and disturbs the flow. L.L. Simpson studied this phenomenon, while Kelly provided valuable data. In accordance to Simpson (1968) the bubbles developed can form a flow regime known as slug flow. Slug flow is characterized by pressure pulsations which in turn can produce pipe vibrations. Gravitational forces controls slug flow, while surface forces are negligible. If the viscosity of the liquid is less than 10^{-3} Ns/m^2 and the pipe diameter is greater than 1", slug flow can be described by the densimetric Froude numbers.

(2.25)

$$Fr_L = \frac{V_L}{\sqrt{gD}}\sqrt{\frac{\rho_L}{\rho_L - \rho_G}} = \frac{V_L}{\sqrt{g_L D}}$$

(2.26)

$$Fr_G = \frac{V_G}{\sqrt{gD}}\sqrt{\frac{\rho_G}{\rho_L - \rho_G}} = \frac{V_G}{\sqrt{g_G D}}$$

Here V_G and V_L are the superficial velocities for respectively gas and liquid. Superficial velocity is defined as the volume flow rate of the phase entering the pipe cross section divided by the cross sectional area of the pipe (Govier and Aziz, 2008).
Due to equation (2.25) and (2.26), g_L and g_G are given by

(2.27)

$$g_L = \sqrt{\frac{g(\rho_L - \rho_G)}{\rho_L}}$$

(2.28)

$$g_G = \sqrt{\frac{g(\rho_L - \rho_G)}{(\rho_G)}}$$

Measurements are performed by Simpson on an unvented pipe discharging to atmosphere and an unvented pipe with submerged outlet, Figure 2.7. The pipe used is of diameter 13/16" and length 1m. Initially both pipes are empty. Water is applied to the pipe discharging to atmosphere and the flow rate is gradually increased. As the flow rate increases a water film is first formed at the pipe wall. Further increase in flow rate results in flooding of the pipe at Fr_L=2, and the total cross section is water filled. In the next setup where the discharge is submerged, the same procedure is performed. Results showed that at Fr_L<0.31 long bubbles tend to be trapped in the pipe, while at Fr_L>0.31 bubbles are swept down and out with the water flow.

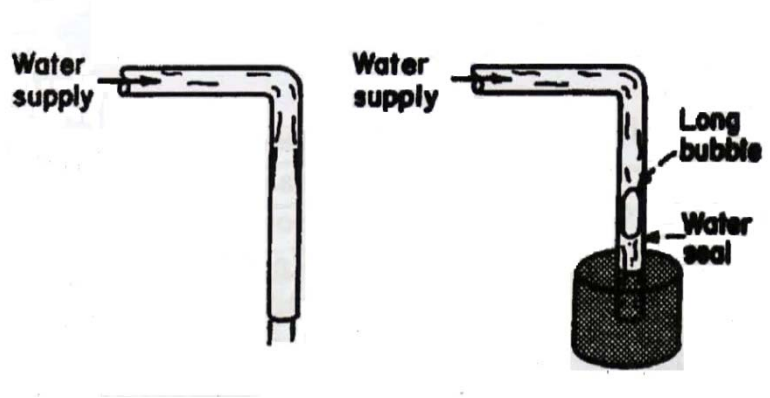

Figure 2.7 Downflow in vertical pipe (Simpson, 1968).

A.G. Kelly studied the minimum siphon velocities to ensure gas is removed from the pipe in down flow. The pipe diameters used are ¾", 2" and 10", and a pipe length of about 17m is applied. Results in Table 2.1 shows that Froude numbers remain considerably constant, even if the flow rate is increased for higher pipe diameters.

Table 2.1 Results from Kelly's study (Simpson, 1968).

Pipe size [in]	**Minimum siphon velocity [ft/s]**	**Minimum densimetric Froude number Fr_L**
¾	0.8	0.56
2	1	0.43
10	3	0.58

2.5.2 Taylor bubble in stagnant liquid

Subsequent section is taken from Simpson (1968). Bubble rise velocity in a tube containing stagnant liquid is studied by several scientists. Both experimental and theoretical studies are performed. The theoretical study performed by G. Taylor and R. M. Davies results in the following equation for the bubble rise velocity.

(2.29)

$$V_B = 0.328\sqrt{gD}$$

The best experimental results are obtained by T. Z. Harmathy.

(2.30)

$$V_B = 0.35\sqrt{g_L D}$$

Long bubbles are generally said to be trapped in down flow when the pipe diameter is larger than 1" and the liquid viscosity is less than 10^{-3} Ns/m^2.

(2.31)

$$V_L = 0.31\sqrt{g_L D}$$

Equation (2.31) calculates the superficial liquid velocity in the pipe which is equal but in opposite direction from the rising bubble velocity. If the superficial liquid velocity is less than that predicted in equation (2.31), bubbles will rise. A higher superficial liquid velocity will cause bubbles to be swept down and out of the pipe. Pressure pulsations and vibrations is expected in the presented range if a continuous gas source is available.

(2.32)

$$0.31 \leq \frac{V_L}{\sqrt{g_L D}} \leq 1$$

The equation above equals the liquid densimetric Froude number Fr_L. Frictional forces will outweigh gravitational forces if the densimetric Froude number increases above 1. This will result in the lack of a pressure gradient in the down flow.

2.5.3 Taylor bubble in a flowing liquid

In accordance to Govier and Aziz (2008), the velocity of a Taylor bubble in a moving liquid is expected to be the sum of the stagnant Taylor bubble velocity and the liquid velocity. Nicklin and Neal studied the bubble rise velocity when water flows upward with Reynolds numbers in the range between 0 and 60000. The absolute rise velocity of a Taylor bubble in a flowing liquid is presented as

(2.33)

$$V_{B,Abs} = C_\infty\sqrt{g_L D} + C_0 V_M$$

The first term is the bubble rise velocity in a stagnant liquid, where C_∞ is an empirical coefficient. The second term indicates the addition of liquid velocity, where V_M is the average velocity of the slug, i.e the liquid above and below the long bubble. C_0 is an empirical coefficient. At Reynolds numbers higher than 8000 Nicklin obtained C_0=1.2 and C_∞=0.35, while at lower Reynolds numbers C_0=2 and C_∞=0.35.
Upward and downward slug flow are described by Fabre and Line (2010). The elongated bubble is bullet shaped and axially centered when the liquid flow is upward in a vertical pipe. If the liquid flows downwards the elongated bubble migrates with an asymmetrical shape. In addition the elongated bubble is distorted first at one side of the vertical pipe, and then at the next as the liquid reaches a critical velocity. Figure 2.8 shows the experimental results in upward and downward flow. The bubble rise velocity is seen to be linear in the presented velocity interval.

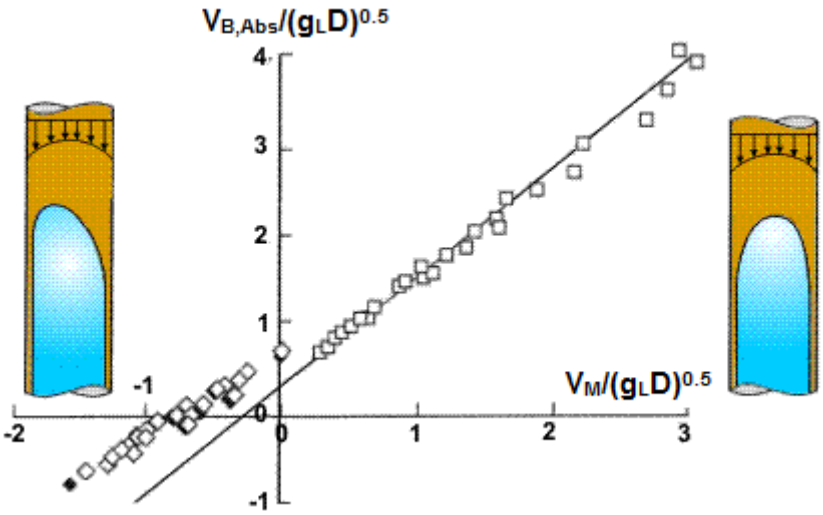

Figure 2.8 Bubble rise velocity as a function of average liquid velocity. D=50mm □, Fréchou, 1986; 140 mm ♦, 100 mm ◊, 26 mm ▲. Results are compared with C. S. Martin's equation with coefficients C_0=1.2 and C_∞=0.35 (Fabre and Line, 2010).

C. S. Martin performed experiments on slug flow when the liquid flows downwards and thus countercurrent to the elongated bubble. He discovered that the empirical coefficients must change to C_0=1 and C_∞=0.7 to fit the experimental data. When the liquid velocity is close to zero, Martin observed a transition in bubble nose shape. The bubble nose is centered in upward flow, while in downward flow it changes to an unstable and asymmetrical bubble nose. Likely this occurs when the inertia forces balance the surface tension force.

2.5.4 Down flow from a process vessel

Down flow from a process vessel can be described in accordance to Simpson (1968), McDuffy (1977) and Rochelle and Briscoe (2010). The rising bubble phenomenon inside a pipe is often mixed with that seen for discharge from a process vessel. In a process vessel irrotational flow can occur, and the flow pattern is more complex due to the liquid depth of the vessel and the entrance geometry. The different flow regimes which occur depends on the height to diameter ratio as seen in Figure 2.9. It must be stated that the process vessel is assumed considerably wide. At an H/D ratio less than 0.25, self-venting weir flow occurs. In the downstream pipe, liquid flows as a film on the pipe wall. Considerably small amounts of air are sucked into the flowing liquid from the weir formation. Souders presented in 1938 an equation for the H/D ratio less than 0.25 in terms of the liquid densimetric Froude number

(2.34)

$$Fr_L = 2.36\left(\frac{H}{D}\right)^{1.5}$$

Souders equation indicates self-venting above the plotted line in Figure 2.9, and the drain pipe is running full at points below the plotted line. The densimetric Froude number is the ratio of downwards drag force to upwards buoyancy force for an entrained bubble. If the drag force is greater than the buoyancy force, the bubble will be entrained in the drained downflow. In 1971 Anderson presented an approximately equal correlation for circular weir flow at H/D less than 0.25

(2.35)

$$Fr_L = 2.31\left(\frac{H}{D}\right)^{1.5}$$

In accordance to the above equations a liquid densimetric Froude number less than 0.3 or an H/D ratio less than 0.25 will cause the down pipe to run full. The transition is set to Fr_L=0.55 by D. S. Ullock.
If the densimetric Froude number is increased beyond 0.3, air is sucked into the flowing liquid unless a sufficient height of liquid in the vessel is maintained. In this second flow regime a cone is formed at which air is sucked into the liquid flow. The rate at which air is sucked into the liquid flow increases for a given diameter as the height H increases. This continuous until a sufficient liquid height is reached in which the induced air flow rate decreased to zero. The critical height is determined experimentally by A. A. Kalinske

(2.36)

$$Fr_L = 4.4\left(\frac{H}{D}\right)^2$$

In 1959 D. R. F. Harleman studied the critical height theoretically and obtained

(2.37)

$$Fr_L = 3.2\left(\frac{H}{D}\right)^{2.5}$$

Harleman's equation is suggested as conservative and shall ensure that air is not sucked into the down pipe.

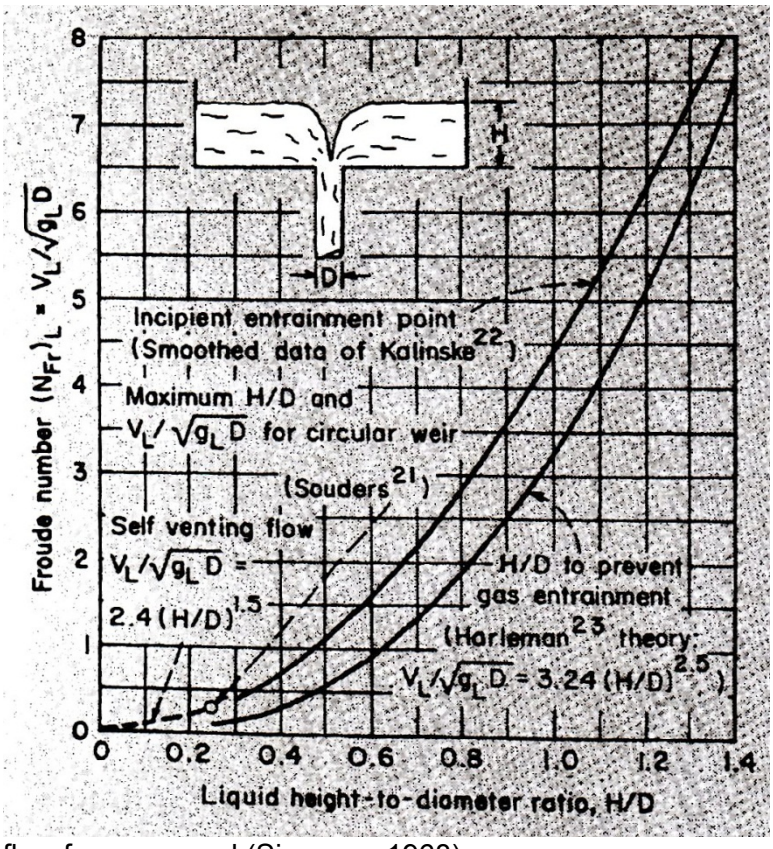

Figure 2.9 Downflow from a vessel (Simpson, 1968).

2.5.5 Flooding

Flooding can be described according to Cetinbudaklar and Jameson (1969). Assume that liquid flows down a vertical pipe as a film along the wall. Gas is flowing upwards in the core. At sufficiently low gas flow rates the liquid simply flows down the pipe. If the gas flow rate is sufficiently increased, liquid starts to move upwards. The upward moving liquid is characterized by surface waves, which sometimes may have large amplitudes. Further increase of the gas flow rate can cause the surface waves to bridge the tube, causing a rapid change in flow direction of the liquid. The net liquid flow is upwards and the event is known as flooding.

2.5.6 Experiment of Two Phase Down Flow - 1987

This section refers to the experiment performed by Moon et al. (1987) regarding the flooding transition of water flowing downwards in vented and unvented vertical pipes . The experiment is performed with special regard on caisson design, and the setup is shown in Figure 2.10. A caisson is a downward sloped pipeline used to discharge fluids such as sewerage, produced water and seawater to sea. It is important to maintain annular flow in such pipelines to prevent chaotic flow which tends to cause undesirable vibrations in the pipeline and to prevent flooding. The lack of data regarding discharge pipelines, such as caissons, has resulted in overdesign which may increase the cost. The experiment is performed on pipes of diameter 1", 2", 4" and 6". The pipe length is 1m, except for the 4" and 6" which is respectively 1,5m and 2m. Different entry conditions are applied during the experiment such as the tee-piece entry, weir entry and an annular flow stabilizer.

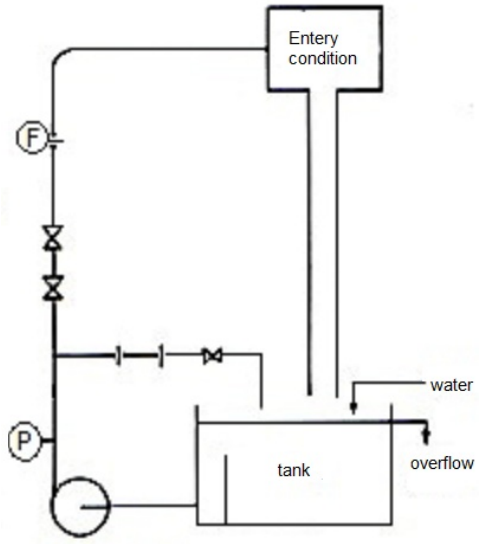

Figure 2.10 Setup of downward flow arranged with different entry conditions (Moon et al., 1987).

Annular flow is first introduced in the unvented pipe arranged with a tee piece entry condition, Figure 2.11. The flow rate is increased causing the development of waves which produce droplets within the gas-core. Further increase of the flow rate produces a plug of water at the discharge, and the trapped elongated bubble rises up the pipe when the flow rate is enhanced. Flooding is said to occur. The observations are similar for all pipe diameters. Froude numbers for flooding and deflooding can be viewed in Table 2.2.

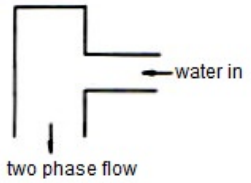

Figure 2.11 Unvented tee piece entry condition (Moon et al., 1987).

Table 2.2 Froude number of the flooding and deflooding transitions in the unvented pipe with tee-piece entry (Moon et al., 1987).

Pipe diameter [inch]	1	2	4	6
Fr flooding	1,55	0,89	0,90	0,92
Position of flood initiation	discharge	discharge	discharge	discharge
Fr deflooding	0,23	0,39	0,48	0,49

A similar procedure is carried out with a weir entry condition, Figure 2.12. By use of this entry condition flooding depends on the pipe diameter. The 4" and 6" pipe flood as described for the tee-piece entry condition. In the 1" pipe flooding is initiated by an air neck forming at the top of the vertical pipe. The air neck instantaneously forms an air slug at the top of the vertical pipe as the flow rate is increased, and the air slug is forced down through the discharge. For the 2" pipe, poor aperture design results in a vortex motion within the annular flow regime. Eventually flooding occurs as for the 1" pipe, but the vortex motion imposes a marked effect on the flow transition. In Table 2.3 flooding and deflooding Froude numbers are presented for the unvented weir entry condition. At the deflooding point the flow rate is decreased sufficiently such that an air slug enters the discharge and flows up the vertical pipe such that annular flow is maintained. The elongated bubble rises centrally for the 1" and 2" pipe, whereas it is displaced from the center when it rises in the 4" and 6" pipe. Elongated bubbles which instead of rising in the center of the tube attach themselves to the pipe wall are known as lazy slug. It takes the shape as if it was half of an elongated bubble, where the other half can be seen as a mirror image in the pipe wall. The bubble velocity depends on its nose shape, and therefore the lazy slug is assumed to travel with about $\sqrt{2}$ times the velocity of a central elongated bubble.

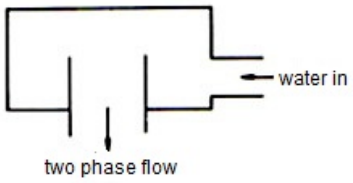

Figure 2.12 Unvented weir entry condition (Moon et al., 1987).

Table 2.3 Froude number of flooding and deflooding transitions for the unvented pipe with weir type entry (Moon et al., 1987).

Pipe [inch]	1	2	4	6
Fr flooding	0,66	1,35[1]	0,92[2]-1,02	0,92[2]-1,04
Position of flood initiation	Entry	entry	discharge	discharge
Fr deflooding	0,23	0,39	0,49	0,48

Annular flow is introduced in the vented pipe arranged with a tee piece entry condition, Figure 2.13. Increased flow rate results in the formation of a water plug at the entry which prevents air from flowing down the pipe. Air is periodically surged through the water plug and the flow pattern is defined as pulsating.

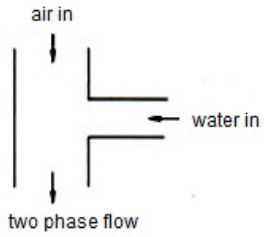

Figure 2.13 Vented tee piece entry condition (Moon et al., 1987).

The vented weir entry condition initially produces annular flow, Figure 2.14. As the flow rate is increased the air core is closing. Pulsating flow condition occurs as air periodically pierces the water which encloses the air core. The Froude number is observed to decrease as the diameter increases, Table 2.4.

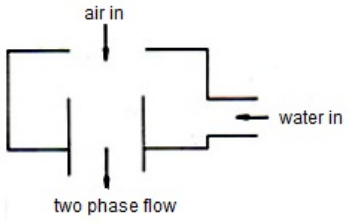

Figure 2.14 Vented weir entry condition (Moon et al., 1987).

Table 2.4 Froude number of annular to pulsating flow transition for the vented pipe with weir entry (Moon et al., 1987).

Pipe [inch]	1	2	4	6
Fr Transition	0,77	0,39	0,42	0,35

Experiments continued on an unvented 2" pipe with submerged outlet, Figure 2.15. The flow pattern observed here differed considerably. In the upper section of the vertical pipe annular flow is maintained. Further down the pipe the annular flow pattern strikes a water reservoir and forms bubbles. These bubbles are swept towards and expelled at the discharge of the vertical submerged pipe. This continues until the height of the annular to bubbly regime interface stabilizes in relation to the flow rate. The height of the interface increases with increased flow rate until it reaches the pipe inlet. Either pressure or the height may be used as a measure of flooding, Table 2.5.

[1] Considerably vortices observed

[2] Lower value corresponds to unbaffled flow at weir entrance

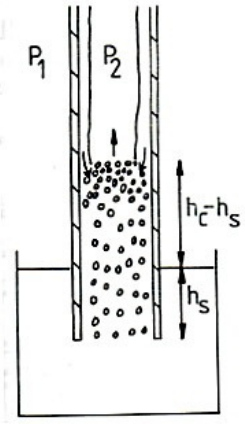

Figure 2.15 Pipe with submerged outlet (Moon et al., 1987).

Table 2.5 Froude number, height and air-core pressure for the unvented 2" pipe with submerged outlet and tee piece entry (Moon et al., 1987).

Fr	0,16	0,18	0,21	0,22	0,25
Height [mm]	0,048	0,100	0,695	0,745	0,845
Air core pressure [kPa]	0	-0,33	-6,59	-6,82	-7,89

The last experiment introduces an annular flow stabilizer as the entry condition, Figure 2.16. The AFS is expected to stabilize the liquid annulus by establishing an annular film at the entry to the vertical tube. It is tested on both a vented and unvented 1" pipe without submerged outlet. The unvented pipe obtained similar flow regimes as the unvented pipes explained previously, although the flooding transition occurred at an increased Fr of 1,84. In the vented pipe annular flow remained stable up to the maximum flow rate capacity. The Froude number at maximum flow equals 4.5, without any sign of bridging of the gas core in the annular flow.

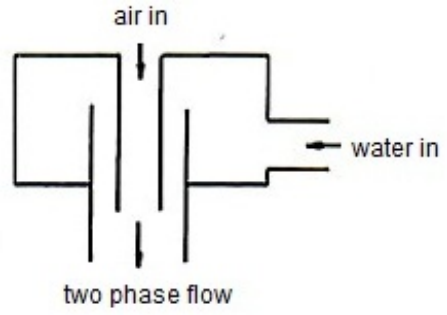

Figure 2.16 Vented annular flow stabilizer entry condition (Moon et al., 1987).

Subsequent sub section summarizes the conclusions of the experiment. According to (Moon et al., 1987) the bubble rise velocity of a lazy slug is assumed equal to

(2.38)

$$V_s = \sqrt{2} * 0{,}35 * \sqrt{gD} = 0{,}49 * \sqrt{gD}$$

This implies a Froude number of 0,49 for the lazy slug. The Froude number of 0,66 which is observed for the 1" unvented pipe with weir entry may be due to the distortion of the slug nose caused by the agitated flow pattern in the top of the flow tube. Flooding is initiated at the discharge for all unvented pipes with tee-piece entry condition and for 4" and 6" unvented pipes with weir entry condition. Results indicates that this type of flooding occurs at Fr =0,95

when the pipe diameter is greater than 2". The Froude number for deflooding was found to be independent of the entry conditions. Due to the lazy slug mechanism the Froude number for deflooding is expected to be 0,49. This correlates well with the results obtained for the 4" and 6" unvented pipes. In vented pipes with weir entry a transition from annular to pulsating flow occurs. The Froude number decreases with increasing pipe diameter, and a steady value of Fr =0,35 is reached at the greatest pipe diameter. At small pipe diameters surface tension is assumed to destroy lateral momentum, and a higher Froude number is therefore required to cause pulsating conditions. The unvented pipe with submerged outlet obtains highest Froude number at the largest pipe diameter. It equals Fr=0.25 for the 6" pipe. L.L. Simpson obtained flooding at Fr=0.31 but for a smaller pipe diameter. In the experiment of Simpson a higher length to diameter ratio is used, which may cause the difference. The AFS ensures annular flow up to Fr~4,5 for a 1" vented pipe without submerged outlet, and annular flow up to Fr~1,84 for the unvented 1" pipe without submerged outlet.

2.5.7 Experiment of Two-Phase Downflow – 1989

In the experiment of Thorpe et al. (1989) the design criterion of Fr=0,3 for caisson design is further investigated. The experiments performed consider vented and unvented vertical pipes with and without a submerged outlet. Larger pipe diameters are generally considered, because surface tension significantly affects flooding transition in pipes of small diameters. Experiments are performed with water and air, and Table 2.6 report the applied pipe sizes. The diameters correspond to 1", 2", 4" and 6". Figure 2.17 shows the test setup.

Table 2.6 The pipe dimensions used in the experiments (Thorpe et al., 1989).

Test section number	1	2	3	4	5	6
Internal diameter [mm]	25,8	57,4	57,4	57,4	89	137
Pipe length [m]	1,0	0,5	1,0	2,0	1,5	2,0

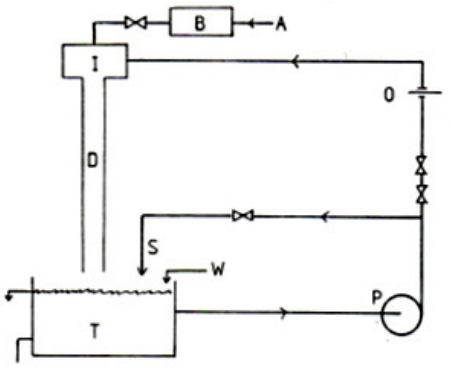

Figure 2.17 Test setup. A: air inlet, B: bubble soap meter, D: downcomer, I: entry condition, O: flow measuring orifice plate, P: pump, S: spill back line, T: tank, W: water (Thorpe et al., 1989).

Different entry conditions are tested in the experiment, tee piece, sharp weir and one of the mentioned entry conditions supplied with an AFS. Tee piece entry condition is easy to manufacture, and is therefore often used at offshore installations. The AFS forces the liquid entering the vertical pipe to flow through a channel which is slightly thicker than the thickness of the liquid film. This results in a less wavy film. At the top of the entry condition there is a tube which allows air to flow into the vertical pipe. This air is dragged in as water flows down the caisson. In addition three different exit conditions is used, straight edged, flanged and a skirt. The latter can be seen in Figure 2.18.

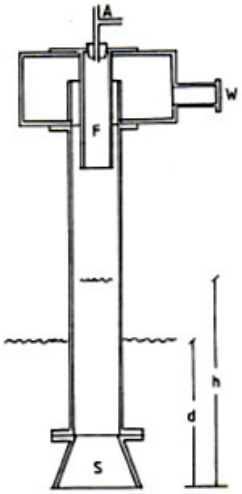

Figure 2.18 Cross section through a downcomer. A: air inlet, F: AFS, S: skirt, W: water inlet (Thorpe et al., 1989).

At low flow rate in an unvented vertical pipe expelling to atmosphere, a thin film flows down along the pipe wall. The air core is at a slightly reduced pressure and causes the liquid to be pulled against the center when it reaches the outlet, Figure 2.19.

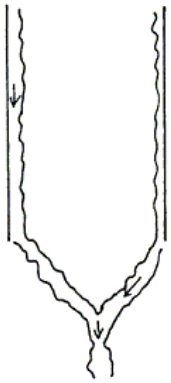

Figure 2.19 Unvented vertical pipe expelling to atmosphere at low flow rate (Thorpe et al., 1989).

Pressure difference between the air core and the atmosphere increases as the flow rate is increased. A bridged section forms at the outlet and effectively seals between the air core and the atmosphere, Figure 2.20.

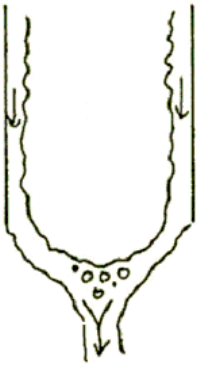

Figure 2.20 Unvented vertical pipe expelling to atmosphere at increased flow rate (Thorpe et al., 1989).

As the wavy liquid film strikes the bridged section air bubbles are drawn in, and subsequently flushed out, Figure 2.21. The bridged section rises up the pipe as air is removed from the column. Flooding occurs when the air core is significantly raised above the discharge. When the flow rate is sufficiently decreased, an air slug enters and rises up the vertical pipe. In this manner annular flow containing a gas core is achieved again. The mechanism is named deflooding. Flooding is initiated at discharge for all pipes unless the 1" pipe with weir entry, which is top initiated. Results agrees well with those of Moon et al. (1987), see Table 2.7.

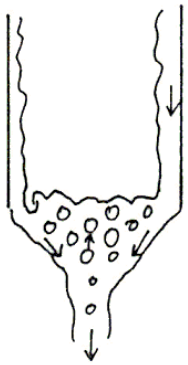

Figure 2.21 Unvented vertical pipe expelling to atmosphere. Bubbles are drawn in and flushed out (Thorpe et al., 1989).

Table 2.7 Comparison of Froude numbers for unvented vertical pipes expelling to atmosphere (Thorpe et al., 1989).

Test section Number	1	3	4	5	6
Fr flooding	0,71	1,09	1,14	1,02[3]	1,09
Fr flooding (Moon et al., 1987)	0,66	1,35[3]	-	1,02	1,04
Fr deflooding	0,30	0,35	0,27	0,67[3]	0,46
Fr deflooding (Moon et al., 1987)	0,23	0,39	-	0,49	0,48
Error from measurements	0,02	0,1	0,1	0,05	0,03

Flooding occurs at higher Froude numbers with an associated AFS. By use of an AFS, the Froude number is found to depend on the diameter to pipe length ratio. As this ratio decreases in value, the Froude number decreases to a value of about 2. This may be due to waves which grow in size down the vertical pipe.
In an unvented vertical pipe with submerged outlet, the liquid film strikes the water at the discharge and plunges into the pool, Figure 2.22.

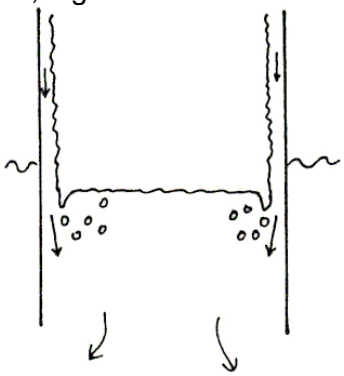

Figure 2.22 Unvented vertical pipe with submerged outlet. Liquid film strikes the water (Thorpe et al., 1989).

[3] Significant velocity was present which stabilized the annular flow

Surface waves in addition to the striking film causes an agitated flow regime at the discharge. Air from the air core is swept down by the agitated flow. These air bubbles are drained out from the pipe if the liquid velocity exceeds the bubble rise velocity, Figure 2.23.

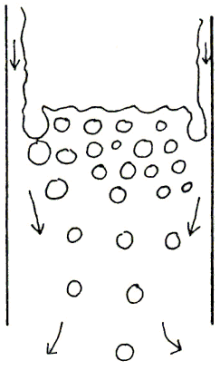

Figure 2.23 Unvented vertical pipe with submerged outlet. Bubbles are drained out of the vertical pipe (Thorpe et al., 1989).

The air core rises up the vertical pipe as air is drained out. An increased flow rate causes more bubbles to be drained out and the air core raises again, Figure 2.24. Flooding is obtained when the air core is removed.

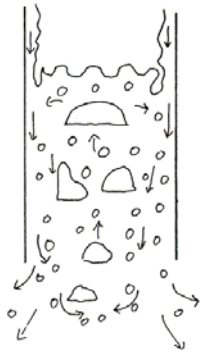

Figure 2.24 Unvented vertical pipe with submerged outlet. Air core rises up the vertical pipe (Thorpe et al., 1989).

Flooding data can be seen in Table 2.8 for different test setups of unvented vertical pipes with a submerged outlet. A skirt is applied at the discharge. Since air bubbles are observed to rise just outside the downcomer, a skirt seemed to have an improvement. Flooding occurred at Froude numbers of Fr=0.35 and Fr=0.41 for respectively small and large skirts.

Table 2.8 Unvented vertical pipes with submerged outlet (Thorpe et al., 1989).

Entry condition	Exit condition	Salt	Fr flooding	Fr deflooding
Weir, no AFS	Straight edge	No	0,245	-
Weir with AFS	Straight edge	No	0,260	0,222
Tee piece, no AFS	Skirt no.1	No	0,346	0,193
Tee piece, no AFS	Skirt no.2	No	0,414	-
Tee piece, no AFS	Straight edge	Yes	0,140	-
Tee piece, no AFS	Skirt no.1	Yes	0,200	-

As the flow rate of water and induced air is low the vented pipe expelling to atmosphere experience a decreased void fraction. As the liquid flow rate is further reduced, the top surface is moved down the vertical pipe. A lazy slug is entrained at the discharge and rises up the pipe before the top surface is moved the whole way down the vertical pipe. In the

experiments of the vented vertical 2" pipe discharging to atmosphere, the induced air restriction is increased, see Table 2.9.

Table 2.9 Vented vertical pipes expelling to atmosphere with restricted air flow rate (Thorpe et al., 1989).

	→ Increasing air restriction →							Unvented
Fr	1,99	1,94	2,01	1,89	1,73	1,70	1,7	1,29
$Fr_G/10^{-3}$ (flooded)	7,6	7,4	7,3	6,4	5,93	5,1	3,3	0
$Fr_G/10^{-3}$ (before flooding)	1,3	1,2	-	-	0,31	0,72	-	0

Generally Thorpe et al. (1989) emphasize the following. For unvented pipes with a submerged outlet a Froude number of 0.2 is more realistic than the design criteria of 0.3. The addition of a skirt improves bubble retention and therefore flooding occurs at a higher Froude number. For pipes expelling to atmosphere flooding is initiated at the discharge when the diameter is in excess of 0.03m. Unvented pipes expelling to atmosphere with diameter in excess of 0.057m is subject to flooding at about Fr=1.1. Vented caissons expelling to atmosphere obtains flooding at higher Froude numbers. An air restriction does not change the Froude number at which flooding occurs, but decreases the Froude number during deflooding due to the decreased fraction of air in the liquid flow.

3 NUMERICAL MODEL

3.1 Introduction to OpenFOAM

According to Hjertager (2009) OpenFOAM is an open source CFD software package written in C++. The development started at the Imperial College in London in the period 1990-1999. Key providers to the development are Henry Weller and Hrvoje Jasak. The function of OpenFOAM generally occurs via text files and unix style commands. A Linux operating system is required to run OpenFOAM, but in Windows one can use a virtual player such as VMware or VM Virtual Box to open and run OpenFOAM. The case structure in OpenFOAM can be viewed in Figure 3.1. Each case is saved as a directory, and it must contain at least the three directories 0, constant and system. In the constant directory one finds material properties, turbulence properties and mesh information. Solution controls, discretization schemes and time step controls are located in the system directory. In the 0 directory all the initial flow fields that are relevant for the current solver is found. Boundary conditions are set for each field. As the case is solved new time dumps are written which consists of new field data.

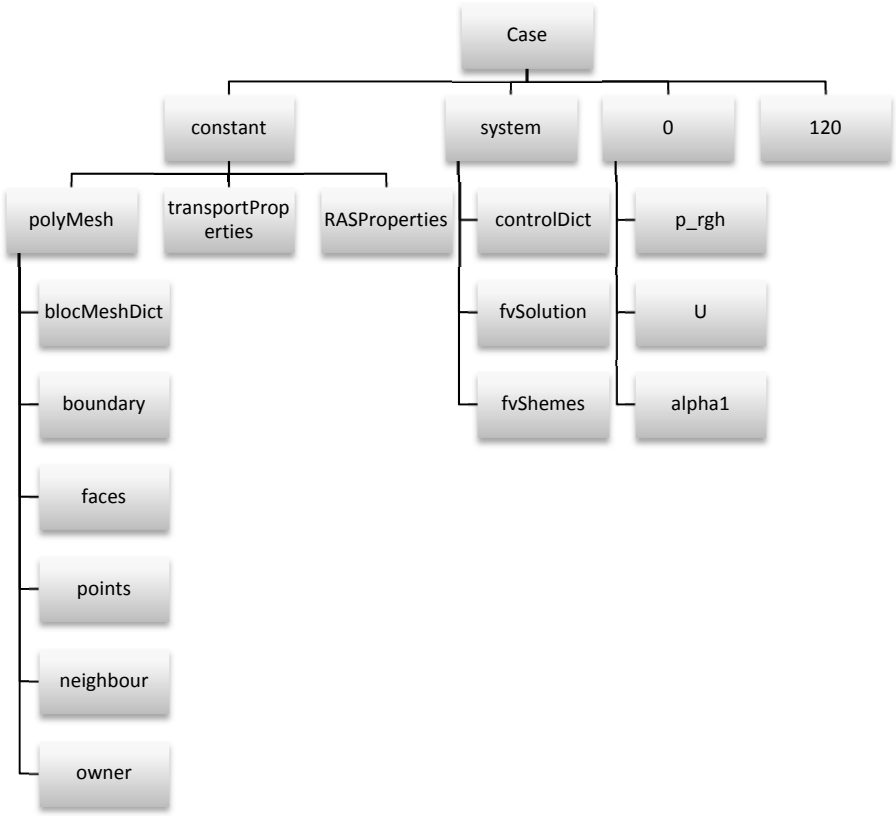

Figure 3.1 OpenFOAM case structure (Hjertager, 2009).

3.2 interFoam

Large bubbles are expected in the simulation of pressure driven to gravity driven flow. A model named interFoam is therefore adapted. In accordance to Vallier (2011), interFoam simulates bubbles larger than the grid size. According to the OpenFOAM Foundation (2011e) the solver interFoam is a two-phase algorithm based on the VOF approach. VOF is short for volume of fluid. In each cell the relative volume fraction of the two phases are determined by a transport equation. The interface between liquid and gas is never sharply defined. Rather, the interface occupies a volume around the region where this sharp interface should exist. The value of the phase fraction can be anywhere between 0 and 1. Other physical properties

are calculated as a weighted average based on the phase fraction value. The interFoam solver applies to two incompressible, isothermal immiscible fluids. Table 3.1 presents the variables used in the interFoam model.

Table 3.1 Symbol description (Maki, 2011).

Symbols openFoam	**Symbols equations**	**Description**
p_rgh	p_d	dynamic pressure
p	P	total pressure ($P = p_d + \rho\, \mathbf{g}\cdot\mathbf{X}$)
alpha1	α	volume fraction
U	**U**	velocity vector
phi	$\mathbf{S_f} \cdot \mathbf{U_f}$	volume flux
rhoPhi	$\mathbf{S_f} \cdot \rho_f \mathbf{U_f}$	mass flux
gh	$\mathbf{g} \cdot \mathbf{X_P}$	hydrostatic pressure over density at cell center
ghf	$\mathbf{g} \cdot \mathbf{X_f}$	hydrostatic pressure over density at face center

The VOF model was first introduced by B.D. Nichols and C.W. Hirt in 1975, and by W.F Noh and P.R. Woodward in 1976. The method was further expanded by B.D. Nichols and C.W. Hirt in 1981. Later on, the model was improved by M. Rudman in 1997 and by W.J. Reider and D.B. Kothe in 1998 (Chung, 2002). The VOF model used in interFoam is described in accordance to Maki (2011), OpenFOAM Foundation (2011f) and Rusche (2002), otherwise it is cited. Generally the VOF model used in interFoam is based on the interface-capturing methodology. The model includes the surface tension between the two fluids but neglects heat and mass transfer. Firstly, the volume fraction α is conserved during space and time.

(3.1)

$$\alpha = \alpha(\boldsymbol{X}, t)$$

Initially the transport equation and the continuity equation for the volume fraction are calculated.

(3.2)

$$\frac{\partial \alpha}{\partial t} + (\boldsymbol{U} \cdot \nabla)\alpha = 0$$

(3.3)

$$\frac{D\alpha}{Dt} = 0$$

It is difficult to calculate the gradient of alpha as it varies from 0 to 1 over an infinitesimal thickness. The result is smearing of the interface between the two phases. The volume fraction α is a scalar value always between 0 and 1. A value close to 0 indicates a pure gas phase, and a value close to 1 indicates a pure liquid phase. Material properties such as density and dynamic viscosity are weighted by the volume fraction α.

(3.4)

$$\rho(\boldsymbol{X}, t) = \rho_{water}\alpha + \rho_{air}(1 - \alpha)$$

(3.5)

$$\mu(\boldsymbol{X}, t) = \mu_{water}\alpha + \mu_{air}(1 - \alpha)$$

It is possible to introduce a modified transport equation in interFoam to compress the surface (Galdamez et al., 2011). The compression is obtained by adding a compression term in the transport equation. This should maintain a clearer interface. In equation (3.6) **U** is the physical velocity and **W** is an artificial velocity which is directed normal and towards the interface.

(3.6)

$$\frac{\partial \alpha}{\partial t} + \nabla \cdot \boldsymbol{U}\alpha + \nabla \cdot \boldsymbol{W}\alpha = 0$$

The magnitude of the artificial velocity **W** can be managed by use of the compression factor `cAlpha` in `fvSolution`. This factor indicates the degree of compression. A value of 0 indicates no compression and a value of 1 indicates conservative compression. Enhanced compression is obtained with a value higher than 1. A value of 1 is recommended (OpenFOAM Foundation, 2011c). The artificial velocity is multiplied by the term $\alpha(1-\alpha)$ which ensures that the compression computation only is activated in the thin interface region. Therefore the surrounding volumes significantly far away from the interface will not be affected by this term.

(3.7)

$$\frac{\partial \alpha}{\partial t} + \nabla \cdot \boldsymbol{U}\alpha + \nabla \cdot \boldsymbol{W}(\alpha(1-\alpha)) = 0$$

In interFoam the fluid motion is assumed to behave in accordance to the Navier-Stokes equation. Further one has to account for the surface tension force at the interface between the two fluids and the differences in material properties. For the velocity **U** and the total pressure P of a fluid with density ρ and dynamic viscosity μ the Reynolds-average momentum equations are expressed as

(3.8)

$$\frac{\partial \rho \boldsymbol{U}}{\partial t} + \nabla \cdot \boldsymbol{U}\boldsymbol{U} = -\nabla P + \rho \boldsymbol{g} + \nabla \cdot [(\mu + \mu_t)(\nabla \boldsymbol{U} + \nabla \boldsymbol{U}^T)]$$

The term $\rho \boldsymbol{g}$ in equation (3.8) is the acceleration of the fluid due to gravity. The total pressure is defined as the sum of the hydrostatic pressure and dynamic pressure respectively. This formulation is advantageous for defining the pressure at the interface between liquid and gas.

(3.9)

$$P = \rho \boldsymbol{g} \cdot \boldsymbol{X} + p_d$$

Taking the gradient of equation (3.9)

(3.10)

$$\nabla P = \nabla(\rho \boldsymbol{g} \cdot \boldsymbol{X}) + \nabla p_d$$
$$\nabla P = \rho \boldsymbol{g} + \boldsymbol{g} \cdot \boldsymbol{X} \nabla \rho \ + \nabla p_d$$

The term $\boldsymbol{g} \cdot \boldsymbol{X} \nabla \rho$ enables an efficient evaluation of the density difference at the interface between the fluids. At the interface $\nabla \rho$ is large, while at sites sufficiently far away from the interface it is zero. Equation (3.10) is inserted in equation (3.8) and the result is

(3.11)

$$\frac{\partial \rho \boldsymbol{U}}{\partial t} + \nabla \cdot \boldsymbol{UU} = -\nabla p_d - \boldsymbol{g} \cdot \boldsymbol{X} \nabla \rho + \nabla \cdot [(\mu + \mu_t)(\nabla \boldsymbol{U} + \nabla \boldsymbol{U}^T)]$$

The last term in equation (3.8) is added to the Navier-Stokes equation to account for the deformations and the stresses within the fluids. It is assumed that the two fluids obey the Newtonian law of viscosity. The stress term is reformulated to ease the evaluation of it

(3.12)

$$\begin{aligned} \nabla \cdot \left[\mu_{eff}(\nabla \boldsymbol{U} + \nabla \boldsymbol{U}^T)\right] &= \nabla \cdot \left(\mu_{eff} \nabla \boldsymbol{U}\right) + \nabla \cdot \left(\mu_{eff} \nabla \boldsymbol{U}^T\right) \\ &= \nabla \cdot \left(\mu_{eff} \nabla \boldsymbol{U}\right) + \nabla \boldsymbol{U} \cdot \nabla \mu_{eff} + \mu_{eff} \nabla (\nabla \cdot \boldsymbol{U}) \\ &= \nabla \cdot \left(\mu_{eff} \nabla \boldsymbol{U}\right) + \nabla \boldsymbol{U} \cdot \nabla \mu_{eff} \end{aligned}$$

The final form of the momentum equation is

(3.13)

$$\frac{\partial \rho \boldsymbol{U}}{\partial t} + \nabla \cdot \boldsymbol{UU} = -\nabla p_d - \boldsymbol{g} \cdot \boldsymbol{X} \nabla \rho + \nabla \cdot \left(\mu_{eff} \nabla \boldsymbol{U}\right) + \nabla \boldsymbol{U} \cdot \nabla \mu_{eff}$$

In the interFoam algorithm only the viscous and convection terms of the momentum equation are used to generate linear systems of the momentum terms $\mathbf{U_x}$,$\mathbf{Uy}$ and $\mathbf{U_z}$. In accordance to Springer et al. (2010) the discretized equation system which are only velocity dependent can be written as in equation (3.14) and (3.15).

(3.14)

$$\mathcal{A} \coloneqq \{[A]\boldsymbol{U} = \boldsymbol{b}\}$$

(3.15)

$$\mathcal{A} \coloneqq \left\{\frac{\partial \rho \boldsymbol{U}}{\partial t} + \nabla \cdot \boldsymbol{UU} = \nabla \cdot \left(\mu_{eff} \nabla \boldsymbol{U}\right) + \nabla \boldsymbol{U} \cdot \nabla \mu_{eff}\right\}$$

For each velocity direction the equations are discretized as

(3.16)

$$[A]_x \boldsymbol{U}_x = \boldsymbol{b}_x$$

(3.17)

$$[A]_y \boldsymbol{U}_y = \boldsymbol{b_y}$$

(3.18)

$$[A]_z \boldsymbol{U}_z = \boldsymbol{b}_z$$

Further the momentum components are solved by use of old values of the pressure and density gradient. This is optional in the interFoam solver.

(3.19)

$$\boldsymbol{U_x} = [A]_x^{-1} \cdot [\boldsymbol{b}_x - \nabla p_d \cdot \boldsymbol{i} - \boldsymbol{g} \cdot \boldsymbol{X} \nabla \rho \cdot \boldsymbol{i}]$$

The development of the pressure correction loop starts with the semi-discrete momentum equation.

(3.20)

$$[A]_x \boldsymbol{U}_x = [\boldsymbol{b}_x - \nabla p_d \cdot \boldsymbol{i} - \boldsymbol{g} \cdot \boldsymbol{X} \nabla \rho \cdot \boldsymbol{i}]$$

Equation (3.20) is examined for a single cell.

(3.21)

$$a_P \boldsymbol{U}_P + \sum a_N \boldsymbol{U}_N = \boldsymbol{b}_P - \nabla p_d - \boldsymbol{g} \cdot \boldsymbol{X} \nabla \rho$$

The velocity is calculated without the pressure gradient and the density gradient.

(3.22)

$$\boldsymbol{U}_P^* = a_P^{-1} (\boldsymbol{b}_P - \sum a_N \boldsymbol{U}_N)$$

The face flux is calculated using starred velocity.

(3.23)

$$\varphi^* = \boldsymbol{U}_f^* \cdot \boldsymbol{S}_f^*$$

The face flux is calculated taking into account the density gradient.

(3.24)

$$\varphi' = \varphi^* - \boldsymbol{g} \cdot \boldsymbol{X_f} \frac{\partial \rho}{\partial n} a_{P,f}^{-1} |\boldsymbol{S}_f|$$

The pressure that makes the velocity discretely divergence free is calculated by use of the continuity equation. Note, φ' will not satisfy continuity. The reason is its numerical approximation and the fact that it does not contain the pressure gradient term.

(3.25)

$$\nabla \cdot \boldsymbol{U} = \sum \boldsymbol{U}_f \cdot \boldsymbol{S}_f = \sum \varphi = 0$$

Once again the momentum equation for a single cell is examined. Note the use of the standard velocity.

(3.26)

$$\boldsymbol{U}_P = \boldsymbol{U}^* - a_P^{-1} \nabla p_d - a_P^{-1} \boldsymbol{g} \cdot \boldsymbol{X} \nabla \rho$$

Equation (3.26) is inserted into the continuity equation, and the pressure can be solved.

(3.27)

$$\nabla \cdot a_P^{-1} \nabla p_d = \sum \varphi'$$

Thereafter the face flux and the velocity are updated.

$$\varphi = \varphi^* - \nabla \cdot a_{P,f}^{-1} \nabla p_f \tag{3.28}$$

$$\boldsymbol{U} = \boldsymbol{U}^* - \boldsymbol{g} \cdot \boldsymbol{X_f} - \nabla p_d \tag{3.29}$$

The interFoam solver consists of a pressure-velocity PIMPLE corrector loop. In this loop the pressure is calculated and the momentum is corrected due to the pressure change.

3.3 Case setup

Five different cases are designed and run. Generally the geometries consist of a vertical pipeline completed with a bend and a horizontal pipeline, Figure 3.2. The bend is designed past a circular arc where the diameter is five times the pipe diameter. A slope of 1:100 is used for the horizontal pipe, as the defined minimum in NORSOK P-001 (2006).

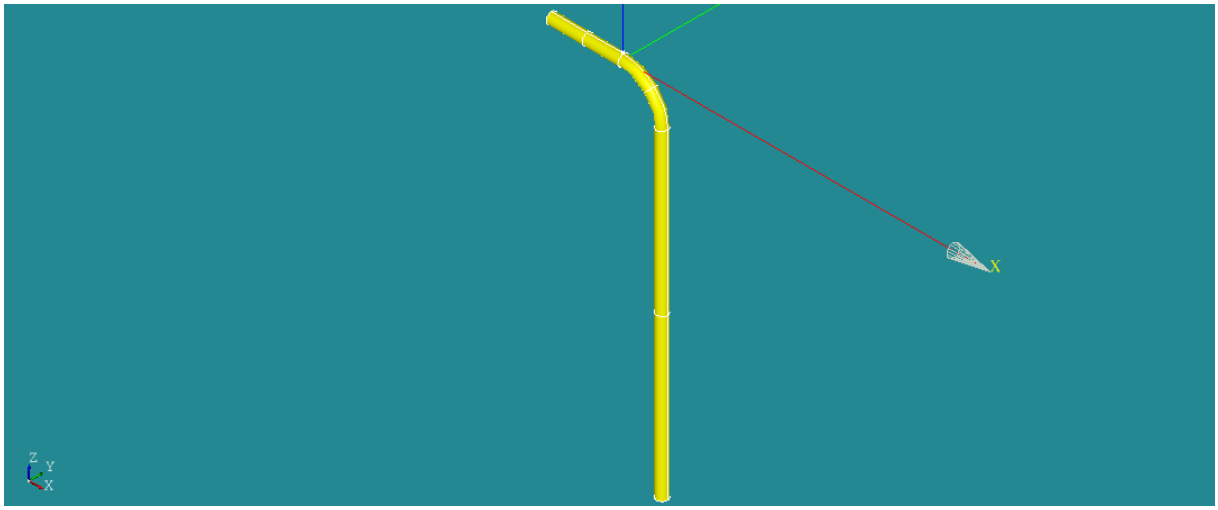

Figure 3.2 Geometry of d24v4

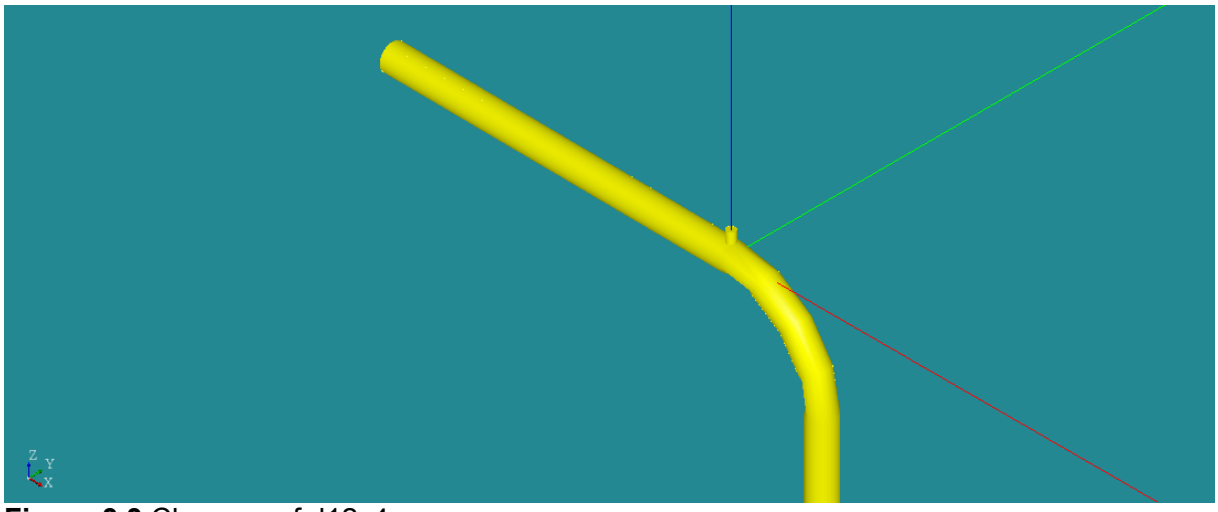

Figure 3.3 Close-up of d12v4

Firstly the diameter of the main pipeline is fixed to 18", while the diameter of the vent pipe is set to 4", 2", and at last the vent is removed. Then the diameter of the main pipeline changes respectively to 12" and 24", while the diameter of the valve is kept constant equal to 4". A vent with a diameter of 4" is chosen to reduce the computational time. After running the18" main pipeline with a 2" vent, it became clear that the velocity is significantly increased, while the cell size is decreased in the vent area. This led to small-time steps. The vent can be seen in Figure 3.3. The geometry is constructed in the xz plane where the origin is set at the intersection between the horizontal and vent pipe. Coordinate axis is shown in the lower left corner of Figure 3.2 and Figure 3.3. Samples performed in a cross sectional area in the vertical section refers therefore only to the depth down the negative z-direction. Each case is given a name to ease the discussion of the different cases, Table 3.2.

Table 3.2 Case names

Name	Diameter main pipe [inch]	Diameter vent [inch]	Height [m]	Horizontal length [m]	Description
d18v2	18	2	20	4	Main pipeline of diameter 18" subjected to a 2" vent
d18v4	18	4	20	4	Main pipeline of diameter 18" subjected to a 4" vent
d18u	18	-	20	4	Main pipeline of diameter 18" which is unvented
d12v4	12	4	20	4	Main pipeline of diameter 12" subjected to a 4" vent
d24v4	24	4	20	4	Main pipeline of diameter 24" subjected to a 4" vent

3.4 Mesh generation

The geometry and mesh is generated by use of Salome. Salome generally is a program with a graphical user interface which can be used for pre- and post-processing of numerical simulations. The pipeline geometry is simply created by use of points, vectors, curve, cylinders and a pipe extrusion along the curved path. The boolean operator fuse, is used to create one shape out of the pipe and cylinders. Faces are generated by exploding faces from the fused shapes. These can be renamed for the desired boundary name, and grouped together such as for walls.
Mesh parameters can be described in accordance to CAE/DEN et al. (2007).The mesh is generated by applying create mesh in the mesh menu. The algorithm Netgen-1D-2D-3D is selected which produces tetrahedron volumes. In the hypothesis section NETGEN 3D Parameters are set and the Hypothesis construction menu appears. In this menu arguments such as max size and min size can be set, which indicates the maximum and minimum linear dimensions of the mesh cells. It is possible to check in for second order. This indicates that a second order node is created such that the mesh gets quadratic. In the fineness section one can choose among very coarse, coarse, moderate, fine or very fine. The arguments growth rate, number of segments per edge and number of segments per radius is fixed in accordance to the choice of fineness. The growth rate is decreased and the number of segments per edge and number of segments per radius are increased as the mesh is refined. Yet there is a possibility to set the fineness to custom, which enables the user to define the growth rate, number of segments per edge and number of segments per radius. The growth rate indicates how much the linear dimension between two adjacent cells is allowed to differ. For example a value of 0.3 indicates 30%. Number of segments per edge and number of segments per radius sets the minimum amount of segments a radius and an edge is divided into. If the optimize box is checked in, the algorithm tries to ensure even

sides of elements. In addition one can define local sizes of elements on and near a specified edge, vertex or face. The object is picked from the geometry, and one selects face, edge or vertex in the menu window. A desired value for the element size can be set for the object. The mesh is computed and the number of nodes, edges, faces and volumes can be checked by selecting evaluate in the mesh menu. Table 3.3 shows mesh data obtained in Salome.

Table 3.3 Mesh data obtained in Salome

	d18v2	d18u	d18v4	d12v4	d24v4
Max size	195	2124.98[4]	2124.98[4]	169	300
Min size	10	122.949[4]	11.4852[4]	21.6485 [4]	19.7048[4]
Fineness	very coarse	very coarse	very coarse	very coarse	very coarse
Growth rate	0.7	0.7	0.7	0.7	0.7
Nb. segs per edge	0.3	0.3	0.3	0.3	0.3
Nb. segs per radius	1	1	1	1	1
Optimize	٧	٧	٧	٧	٧
Edges	182	130	151	186	124
Triangular faces	2220	1492	1592	2164	1272
Tetrahedron volumes	4630	2624	2843	4602	2596

According to Hjertager (2009) the mesh quality is crucial to the success and accuracy of results. The command `checkMesh` creates a list of specific mesh quality measures. Relevant measures are listed in Table 3.4. Complete `checkMesh` measures can be viewed in APPENDIX E. The value of boundary openness and max cell openness does not exceed a value of order 10^{-16}. Thus both meet the criterion of a nearly zero value. Max and min values of areas and volumes are checked to ensure positive values. Negative values are unphysical. Note that the d18v2 case achieves smaller minimum values of the area and volume. The summary of mesh qualities ends with the script `Mesh OK` at the end for all cases. However, non-orthogonality and skewness may affect the stability of the solution. Non-orthogonality is calculated on faces. It is defined as the angle between a center to center line between two neighboring faces and the line normal to the common border of the faces. Generally this value should be less than 30° to obtain an orthogonal mesh. Table 3.4 reports max non-orthogonality values which are higher, but the average non-orthogonality values stays below the criteria of 30°. Skewness is the ratio of the distance between the center of the common border and the point where the center to center line crosses, to the length of the center to center line. The reported skewness stays below 10, which is a criterion for obtaining convergence.

[4] Default value

Table 3.4 Mesh data received from the checkMesh command

	d18v2	d18u	d18v4	d12v4	d24v4
Points	1480	927	998	1484	854
Faces	10370	5994	6482	10286	5828
Internal faces	8150	4502	4890	8122	4556
Tetrahedra cells	4630	2624	2843	4602	2596
Boundary patches	6	4	6	6	6
Faces water_inlet	7	6	6	6	6
Faces water_wall	353	230	260	357	224
Faces air_inlet	6	-	6	6	6
Faces air_wall	76	-	38	37	35
Faces outlet	7	6	6	6	6
Faces walls	1771	1250	1276	1752	995
Max cell openness	2.18e-16	2.03e-16	2.13e-16	2.65e-16	2.09e-16
Max aspect ratio	6.07	6.11	16.72	4.16	4.48
Minimum face area	1.25e-04	8.93e-04	3.06e-04	5.12e-04	7.10e-04
Maximum face area	4.67e-02	5.10e-02	5.19e-02	2.37e-02	7.55e-02
Min volume	7.83e-07	4.85e-04	2.97e-06	5.25e-06	1.03e-05
Max volume	2.69e-03	3.02e-03	3.12e-03	9.73e-04	6.89e-03
Total volume	3.69	3.56	3.55	1.56	6.42
Max non-orthogonality	64.06	42.71	78.57	56.31	51.86
Average non-orthogonality	17.91	14.37	16.14	17.03	16.39
Nb. severely non-orthogonal faces	-	-	6	-	-
Max skewness	0.64	0.46	0.78	0.68	0.55

3.5 Boundary conditions

Boundaries of the pipeline are named `water_inlet:` water entering from pump, `air_inlet:` air entering from valve, `water_wall:` wall of horizontal pipe, `air_wall:` wall of the vent pipe, and `walls:` walls of bend and vertical pipe, and the outlet is `outlet`. A special boundary condition named `groovyBC` is used at the `water_inlet` to enable the pump to suddenly shut down. The velocity is set to 1.18 m/s when the pump is running, which is similar to a flow rate of 700 m^3/h through the 18" diameter pipe d18v4.

(3.30)

$$V = \frac{\dot{V}}{\frac{\pi}{4}D^2} = \frac{700\frac{m^3}{h} * \frac{1}{3600}\frac{h}{s}}{\frac{\pi}{4}(0.4572m)^2} = 1.18\frac{m}{s}$$

3.5.1 The various boundary conditions

After three seconds the pump is shut down and the water is gradually flowing due to gravity. The `groovyBC` uses the keyword `valueExpression` to define; if the time is less than three seconds the velocity is the vector with 1.18 m/s in the x-direction, otherwise it is the zero vector. The boundary conditions can be viewed in APPENDIX A. Since the velocity is fixed to 1.18 m/s or zero, the dynamic pressure at the `water_inlet` is set to `zeroGradient`. Thus the gradient of the dynamic pressure normal to the water inlet is zero, which enables the pressure to float as the velocity is fixed (Hjertager, 2009). When the velocity is zero, this boundary condition is similar as often defined for a wall. The volume fraction at the

water_inlet is set to inletOutlet, where the keywords inletValue and value is both set to uniform 1. Thus the fluid is water when there is an incoming flow to the domain; otherwise the volume fraction is set to zeroGradient. The gradient of the volume fraction is therefore zero normal to the water_inlet when the velocity is zero.
All the walls; water_wall, air_wall and walls are exposed to similar boundary conditions. The velocity is set to fixedValue in conjunction with the value uniform (0 0 0), while the dynamic pressure is set to zeroGradient. This is a normal approach to define wall boundary conditions. At the outlet the velocity field is set to pressureInletOutletVelocity. In accordance to the OpenFOAM Foundation (2011a) this is a mix of the inletOutlet boundary condition and pressureInletVelocity. Thus water flowing out of the outlet is exposed to zerogradient, while the air inflow is a fixed value. The fixed value of U is determined by the flux normal to the outlet when the pressure is known. Dynamic pressure p_rgh at the outlet is set to fixedValue of uniform 0, as is the defined atmospheric pressure. To ensure air is the inflow medium, inletOutlet is used in the alpha1 field with inletValue of uniform 0 and an initial value of uniform 1 since water is expected to instantaneously fill the pipeline.

3.5.2 Surface tension at walls

The choice of alpha1 at walls is more comprehensive. In accordance to the OpenFOAM Foundation (2011c) the boundary condition alphaContactAngle can be used to specify the surface tension at the interface between fluid and walls. If this boundary condition is used, one has to specify the keywords theta0 : the static contact angle, thetaA : the leading edge dynamic contact angle, and thetaR : the trailing edge dynamic contact angle. According to the OpenFOAM Foundation (2011b) the keyword limit must be defined, which determines how the gradient of alpha1 is calculated on the wall. There are four possible choices for this keyword, namely none, gradient, alpha and zeroGradient. If none is set, the gradient is calculated without limiter. The limit gradient calculates the gradient such that alpha1 is bounded on the wall. If alpha is set, the calculated alpha1 is bounded on the wall, while the zeroGradient sets the gradient of alpha1 to zero at the wall. If either none, gradient or alpha is used; the pressure boundary condition for p_rgh must be set to:

```
walls
    {
        type            fixedFluxPressure;
        adjoint         no;
    }
```

This is to ensure that the flux is corrected to zero at the wall. Otherwise there are no restrictions for the pressure boundary condition. It is also possible to use the dynamicAlphaContactAngle in which one also has to specify a velocity scaling function uTheta. In this project it is decided to ignore the surface tension effect between wall and the interface, because of the large pipe diameter. This decision is based on statements in the article of Barajas and Panton (1992), where the "*effect of contact angles on two phase flow in capillary tubes*" are studied. They report that pressure drop and mass transfer depends upon the flow regimes and that these are well defined in large diameter pipes. In small diameter tubes < 1", research has shown that surface tension becomes more pronounced, and the flow patterns are not well defined. In this project the surface tension is ignored by setting the wall boundary condition of alpha1 to zeroGradient, as described by OpenFOAM Foundation (2011c). Another possibility is to set the static contact angle theta0 to 90° and the velocity scaling function uTheta to 0 in the dynamicContactAngle boundary condition.

3.5.3 Development of the vent boundary condition

Considerably effort is spent on the air_inlet to obtain a check valve. This boundary condition is programmed, because openFoam does not provide such a boundary condition. It was examined what boundary conditions which existed in openFoam, and what was previously done by searching the openFoam forum at cfd-online. The groovyBC was adopted, which avoids programming the boundary condition in C++. In accordance to Bgschaid (2009) groovyBC is a mixed boundary condition used to set a non-uniform boundary condition. It is part of the swak4Foam directory and can be downloaded in accordance to the guide by Bgschaid (2010). According to Bgschaid (2009) the `fractionExpression` determines whether the patch is set to Dirichlet (1) or Neumann (0). If not specified it defaults to 1. In openFoam, boundary conditions are divided into either Dirichlet or Neumann types. Dirichlet indicates a fixed value, while Neumann sets a fixed gradient normal to the boundary of the dependent variable (Nabla Ltd, 2007).
The following description of groovyBC is taken from Bgschaid (2009), and gained experience. If the keyword `fractionExpression` is used, one has to define the keywords `valueExpression` and `gradientExpression`, which corresponds to Dirichlet and Neumann respectively. In all keywords ending with `Expression`, one can set up a conditional operator. The conditional operator is defined as `(test ? input1 : input2)`. Here, test is the expression which can be defined by arithmetic operators, fields, functions, logical and comparison operators, etc. See the source sited for more information regarding operators. The string implies; if `test` is true set boundary to `input1`, else (if false) set the boundary to `input2`. If a conditional operator is used in the `fractionExpression`, the inputs must be set to 0 and 1 in the desired order for true and false. This enables a non-uniform mixed boundary condition, where the test condition either sets a fixed value or a fixed gradient. A conditional operator can also be used in `valueExpression`, which enables a fixed value to change during run. The true and false inputs are set to the desired fixed values. Similar can be set for `gradientExpression`, which enables the fixed gradient to change during run. It is optionally to define the keyword `variables`. This allows the user to provide large functions with names. The keyword `value` must be set if `valueExpression` is not, because it is used as value for the first time step. It is never less wise to define `value` because when it is not defined 0 is set for the first time step, which may cause a floating point exception.
The valve is meant to open when the fluid pressure is less than the atmospheric, otherwise it shall be closed. Prusinski et al. (2010) discuss the possibility of creating a check valve, and inspired the first setup. Generally the suggestion inspired to use the `fractionExpression` with the conditional operator `(max(p) < 0 ? 1 : 0)` for p_rgh and alpha field, and `(max(p) < 0 ? 0 : 1)` for U. Thus the pressure was set to a fixed pressure of 0 and alpha1 set to a fixed value of 0 for air when the condition was true, otherwise, both were set to zerogradient. The velocity was set to zerogradient when true, and to a zero vector if false. The main problem was the direction of the fluid when the valve opened. As it opened, there was water flowing out of the valve. There were also problems regarding the opening and closing mechanism. Other pseudo-variables such as `average` and `min` where tried out in addition to exchanging p with p_rgh in the expression. In some simulations the valve did not open at all, or it did not close after it opened. Certain combinations of boundary conditions did not seem to be physical, and thus the simulation would not run at all.
An early idea of Hjertager (2012a) was to set the velocity direction by use of the conditional operator in the `valueExpression`. The velocity could then be set by defining two different vectors with a specified velocity in z direction for the opening velocity. Since the velocity of incoming air was not known, the idea was to calculate it from phi. The pressure was simply set to a fixed value of zero. At this time phi was not fully understood in the interFoam solver, and the velocity was calculated by taking phi divided with the product of rho and the area. Therefore the velocity at the inlet became quite small, and could not be compared to the high velocity obtained by setting the velocity to zerogradient as earlier. Later on, when the

interFoam solver was examined, phi turned out to be equal to the volume flow rate in m^3/s. The decision to divide by the area was therefore incorrect.
A promising boundary condition was selected and it can be cited in APPENDIX A. This setup is inspired by a similar boundary condition in ANSYS, the inlet vent (ANSYS, 2009). The pressure boundary is set to `totalPressure` which sets a fixed total pressure P. According to the OpenFOAM Foundation (2011a) the total pressure is calculated as

(3.31)

$$\mathrm{P} = p_s + \frac{1}{2}\rho|\boldsymbol{U}|^2$$

Thus P is adjusted in accordance to changes in **U**. The total pressure is set to 0, which is defined as the atmospheric pressure. According to Finnemore at al. (2002) the first term p_s in this expression is the static pressure, while the second term expresses the dynamic pressure. Therefore when **U** is negative on the patch, a suction pressure will arise, which is the desired condition to open the valve. As the pump is running, a positive pressure and thus a positive velocity at the patch is expected most of the time, such that the valve remains closed. Alpha1 is set to `inletOutlet`. The `inletValue` is set to `uniform 0` and `value` is set to `uniform 1`. Thus, at the first time step there is water at the air inlet, and as the valve open air enters. The boundary condition ensures a fixed value of pure air as air flows in, and is set to zerogradient if the flow is outwards (Hjertager, 2009). The groovy boundary condition is used at the velocity field, and the velocity is either set to a fixed vector of zero by Dirichlet or to a zerogradient condition by Neumann. Average pressure and minimum phi less than zero are set as the conditions in the `fractionExpression`. The choice of conditions may look strange, but it was not possible to run the simulation if max phi and max pressure or average pressure was set as conditions. In paraFoam the vent was studied, but outwards velocities was not noticed. Review of results showed that the valve opens and let air out at oscillating intervals after about 30 seconds of the simulation. Therefore the valve does not work as a one-way valve.
The boundary condition of the velocity field was reviewed again at the `air_inlet`. `totalPressure` and `inletOutlet` for respectively the dynamic pressure and the volume fraction remains unchanged. The idea of using `valueExpression` to set the velocity direction at the air_inlet was resumed. First phi was tried once again as a criteria for opening the valve in addition to max(p). This was to ensure that the velocity `Z` always would be inwards to the domain. After about 44.2 seconds the simulation crashed due to decreased size of time steps. The condition can be viewed below. Review of results regarding volume flow rate indicated opening and closing of the valve with no outflow of air or water.

```
variables “Z=average(phi)/sum(mag(Sf()));”;
valueExpression“(max(p)<0&&max(phi)<0)?vector(0,0,Z):vector(0,0,0)”;
```

The selected boundary condition can be viewed in APPENDIX B. Phi as a direction tool is rejected. Instead, the velocity `Z` is calculated by taking the magnitude of average phi, which is then divided by the patch area. Thus the velocity gets directionless. In the `valueExpression` a negative sign is applied in front of `Z` in the z-direction of the opening velocity. Thus air is either flowing into the domain, or the valve is closed. Opening and closing of the valve is determined by the maximum pressure at the patch. When max pressure at the patch is less than 0, the valve opens. A small change in the boundary condition for the valve is made for the two cases d12v4 and d24v4 because the valve opened and induced considerably air into the pipe prior to the pump shutdown. An additional condition is set such that the valve cannot open at all until after the pump shutdown. The groovyBC is applied at the valve in the alpha1 file. It allows alpha1 to change between zerogradient and a fixed value after the pump is shut down. Adjustments can be viewed in APPENDIX B.

3.6 Fluid properties

The material properties are defined in the `transportProperties` dictionary, and can be viewed in Appendix C. This dictionary can be described in accordance to the OpenFOAM Foundation (2011c). Water and air is defined in the sub dictionaries of respectively `phase1` and `phase2`. For each phase the keyword `transportModel` is set to `Newtonian`. In accordance to Finnemore et al. (2002) a Newtonian fluid is characterized by a constant kinematic viscosity which is kept unchanged with the rate of deformation. Keyword `nu` defines the kinematic viscosity and is set to 1e-06 and 1.48e-05 for respectively water and air. These values correspond to those previously used in the laminar dam break tutorial found in openFoam v2.0.1. Density is set to 1000kg/s and 1.2kg/s for water and air respectively, by means of the keyword `rho`. These values are similar to those used in the laminar dam break tutorial. Other models such as CrossPowerLaw and BirdCarreau are defined in sub dictionaries with their corresponding coefficients. Keyword `sigma` defines the surface tension between the two phase's air and water. The value of 0.07 is adopted from the dam break tutorial.

3.7 Gravitational acceleration

The value and direction of the gravitational acceleration can be set in the dictionary `g`. Gravity is set to 9.81 m/s^2 in negative z-direction. The dictionary can be reviewed in Appendix C. In accordance to the OpenFOAM Foundation (2011c) gravity is a uniform vector field across the computational domain.

3.8 Turbulence modeling

The dictionary `turbulenceProperties` sets the choice of turbulence modeling by use of the keyword `simulationType` (OpenFOAM Foundation, 2011c). In accordance to the OpenFOAM Foundation (2011g) it can be set to either `laminar`, `RASModel` or `LESModel`. If a turbulent model such as RAS or LES is selected one has to define additional model coefficients in the dictionary RASProperties and LESProperties respectively.
The turbulence property is set to laminar at the initial stage such as to limit the complexity of the case. Insufficient time is one of the reasons why the simulation does not proceed in a turbulent model. In addition a turbulent model would cost additional computational time. Turbulence is expected in the simulation, and therefore the laminar model obtained is assumed to be less accurate. Dictionary `turbulenceProperties` can be viewed in APPENDIX C.

3.9 Time step and data output control

According to the OpenFOAM Foundation (2011c) the surface tracking algorithm in interFoam is significantly more sensitive to the Courant number than other models which calculates simple fluid flows. In the region of the interface it is preferable with a Courant number not exceeding 0.5. The velocity propagation is not easily predicted, and a fixed time step is therefore impractical. Automatic time step adjustment is supplied to interFoam and is set in `controlDict`, which can be viewed in APPENDIX D. It is applied by setting the keyword `adjustTimeStep` to `yes`. `maxAlphaCo` and `maxCo` are both set to 0.5. `maxAlphaCo` applies to alpha1, and ensures a max Courant number not exceeding 0.5. The keyword `maxCo` applies to other fields such as p_rgh and U in this case. The keyword `maxDeltaT` determines an upper limit to the time step, which is not expected to be exceeded during run time. It is set to a value of 1, similar to the dam break tutorial. The automatic time step control causes calculations to be performed at arbitrary time steps. Results are however allowed to be written at fixed times. This is done by time step adjustment in the automatic time step procedure such that times coincides with the specified times set by the keyword

writeInterval. The keyword writeControl is set to adjustableRunTime to allow results to be written at fixed time intervals. writeInterval is set to 0.05.
In accordance to the OpenFOAM Foundation (2011e) inputs for time control and the keyword writeInterval are mandatory. Keywords regarding time control are startFrom, startTime, stopAt, endTime and deltaT. OpenFoam offers different entries for each keyword. The startFrom keyword is set to startTime, and startTime is set to 0. Thus the first field data inputs must be read form a directory named 0. The end time is set such as to empty the pipeline for water. The time required differs for different layouts of vented and unvented pipelines, and are determined due to trial and error. It is discovered that an endTime of 120 seconds is sufficient for the 18" pipeline with an associated 4" vent, d18v4. The former keyword stopAt is set to endTime. In openFoam the Courant number is defined as:

(3.32)

$$Co = \frac{\delta t |\boldsymbol{U}|}{\delta \boldsymbol{X}}$$

It is defined for a cell where $\delta \boldsymbol{X}$ is the distance through the cell in the velocity direction, the velocity is taken as the magnitude and δt is the time step. The adjust time step control calculates the time step in accordance to the maximum Courant number. Max Courant number is the result of a small cell size and a high velocity, which both are true statements close to the air_inlet. Salome seems to produce the same amount of cell faces for different pipe diameters. Thus the amount of faces is about equal for the air_inlet and the outlet. Therefore the cell size must be considerably smaller at the air_inlet than at the outlet. The high velocity occurs at the air_inlet as the valve opens. As the diameter of the vent decreases, the velocity increases. Time directories may be saved in a cyclic pattern by overwriting earlier time directories. This is set by purgeWrite with a corresponding value that limits the times a directory is overwritten. Overwriting time directories at a cyclic basis is not desirable, and thus the purgeWrite keyword is set to 0. The writeFormat is set to ascii which is the default setting in openFoam. It sets the format to ASCII, in contrast to binary which sets a binary format. writePrecision is set in conjunction with writeFormat and defaults to a value of 6 in openFoam. Compression of data files are defined by writeCompression, and is set to compressed. This is to reduce the space occupied by the case on the virtual disc in openFoam. At first run, a case with associated samples and a logfile occupied more than 10 Gb on the disc. It was later discovered that the logfile increased considerably in size. This is probably due to the samples that were logged along with the residuals. Thus samples where saved twice, once in its own file and second in the logfile. Therefore each case runs twice, one with samples, and the next with a logfile sampling residuals. The timeFormat is set to general which is the default entry in openFoam. It sets a scientific format which is specified by an integer in the timePrecision. In openFoam an integer of 6 is the default vale.

3.10 fvSolution

The pressure velocity coupling scheme in fvSolution is set to PIMPLE. In accordance to Hjertager (2009) there are two loops in PIMPLE, one inner and one outer. In the outer loop all equations are solved while in the inner loop only the continuity equation is solved. It is important to ensure that the continuity error of the forgoing loop stays small. The max and mean Courant number shall not grow large, but they may have a higher value than that satisfied in the PISO pressure velocity coupling. Initial residuals should be kept small for the last outer loop. These remarks are important to ensure a reasonable converged solution at all times.
The following explanations are taken from Maki (2011), otherwise it is sited. The keyword momentumPredictor accounts for variations by reconstructing the flux (Passalacqua,

2010). It has a relatively small additional computational expense and is therefore recommended. nCorrectors indicates how many times the pressure equation is solved in the outer loop, and for strict time accuracy it is recommended to use minimum 2. nNonOrthogonalCorrectors are used if the mesh is sufficiently non-orthogonal. When time steps are small and nCorrectors are in use this value may be set to 0 in most cases. Note that values of nNonOrthogonalCorrectors and nCorrectors should be kept low since they are calculated nCorrectors times per outer loop and nNonOrthogonalCorrectors times per inner loop (Hjertager, 2009). A loop over the volume fraction is indicated by the keyword nAlphaCorr. It is recommended to use a value of 1-2 for transient flows and 0 for steady state flows. In accordance to the OpenFOAM Foundation (2011c) nAlphaSubCycles represents number of sub-cycles the field alpha1 is calculated. In fvSolution it is set to 3 which is one more than in the dam break tutorial. It indicates that alpha is calculated three times during each time step. Increased number of sub-cycles retains stability without reducing the time step size and thus increasing the solution time. The keyword cAlpha sets the compression term in the modified transport equation, and thus the compression at the interface. A value of 1 indicates conservative compression, while compression is deactivated at a value of 0. According to the OpenFOAM Foundation (2011c) enhanced compression is obtained by any value larger than 1. Generally a value of 1 is recommended and adopted from the dam break tutorial.
The sub-dictionary solvers in fvSolution can be described in accordance to the OpenFOAM Foundation (2011e). In this sub-directory the method of number crunching for each discretized equation solved by interFoam is specified. The equations solved are the pressure correction loop pcorr, the first pressure loop p_rgh, the second and last pressure loop p_rghFinal and the velocity equation U. For all equations regarding the pressure the keyword solver is set to PCG. PCG is short for preconditioned conjugate gradient, and can be applied to symmetric matrices. The structure of the equation determines the symmetry of the matrix. If however a symmetric solver is applied to an asymmetric matrix, openFoam would produce an error message such as to change to an asymmetric solver. For the velocity equation solver is set to PBiCG. PBiCG is short for preconditioned bi-conjugate gradient, which yields asymmetric matrices. All keywords and values in the solvers sub-directory are adopted from the dam break tutorial. Solvers generate iterative solutions and therefore residuals are generated over successive solutions. A residual indicates the error in the solution, and the accuracy therefore increases as the residual decreases. Residuals are normalized such that they will not be affected by the scale of the problem. The residuals are calculated due to the current value before the next equation is solved. It is also calculated after each number of iterations for the specific equation. Disturbances that will stop the solver are residuals lower than the tolerance, a ratio of current to initial residual lower than relTol, or if number of iterations overruns the number defined by maxIter. tolerance is set to 1e-07 for the two pressure loops and to a smaller value of 1e-10 for the pressure corrector loop. The velocity equation is exposed to a tolerance of 1e-06. These values are assumed to produce a sufficiently accurate solution. The relative tolerance relTol is set to 0 for all equations except the first of the two pressure equations. Here the relTol is set to 0.05. In transient cases the relative tolerance is often set to zero such that the solution converges to the defined tolerance in each time step. It can be argued that a relative tolerance of zero for the first pressure loop is unnecessary since it regardless is calculated with the output of zero relTol in the second equation. In addition it is expected to save some computational time, as it is assumed the first equation will produce less number of iterations. The keyword maxIter is optional, and not defined. Different options can be selected in conjunction with the preconditioner keyword. For all equations regarding pressure it is set to DIC. DIC is short for diagonal incomplete-Cholesky, which is a symmetric preconditioner for matrices. The preconditioner for the velocity equation is asymmetric and set to DILU, diagonal incomplete-LU.

3.11 Discretisation schemes

This section is largely drawn from the OpenFOAM Foundation (2011c). All settings in the `fvScheme` dictionary are adopted from the dam break tutorial. MULES are used in the interFoam solver to keep the phase fraction bounded regardless of the mesh structure and numerical scheme. MULES is short for multidimensional universal limiter for explicit solution. This causes the possible selection of schemes other than those that are strongly stable or bounded such as upwind differencing. The convection schemes are defined under the keyword `divSchemes`. In the sub dictionary `divSchemes` the convection term of the momentum equation $\nabla \cdot \rho \boldsymbol{UU}$ is defined as `div(rho*phi,U)`. The scheme `Gauss limitedLinearV 1` is used to ensure good accuracy. The value 1 in the end expresses the coefficient ϕ, which is the flux `phi` required for the interpolation of the convection term. It is assumed that the best stability is obtained with a value of 1. The next numerical scheme `div(phi,alpha)`, also expressed as $\nabla \cdot \boldsymbol{U}\alpha$ is set to `Gauss vanLeer`. `div(phirb,alpha)` formulates the expression $\nabla \cdot \boldsymbol{U}_{rb}\boldsymbol{\alpha}$ and are set to `Gauss interfaceCompression`. The scheme `vanLeer` is also an option, but the `interface Compression` scheme generally produces a smoother interface.
Following descriptions are taken from the OpenFOAM Foundation (2011e), otherwise it is cited. The keyword `ddtSchemes` represents the choice of time scheme. `Euler` is adopted, and indicates a first order bounded implicit scheme. In accordance to Maki (2011) first order accuracy is sufficient due to the small time steps produced by the Courant number restriction. `gradSchemes` sets discretization schemes for gradient terms ∇. It is set to `default Gauss linear`. When `default` is specified, `Gauss linear` is applied to all the gradient terms in the application. Here, Gauss represents the discretization where values are interpolated from cell centers to face centers. It is followed by the selected interpolation scheme `linear`, which has proved to be effective in most cases. The Laplacian scheme is represented by the keyword `laplacianSchemes`, and is applied to terms of the Laplacian operator ∇^2. `default` is followed by `Gauss linear corrected`. The only scheme selectable is the Gauss scheme, but it must be followed by an interpolation scheme and a surface normal gradient scheme respectively. The interpolation scheme `linear` is typically selected. A `corrected` surface normal gradient scheme, indicates a numerical behavior which is unbounded, conservative and of second order. `interpolationShemes` are set to `default linear`. Interpolation schemes in openFoam are generally divided into 4 categories: Centered, Upwind convection, total variation diminishing (TVD) and normalized variable diagram (NVD). Centered schemes are general, while the other three categories often are used for specific divergence or convection terms. `linear` is a centered scheme which indicates linear interpolation. Surface normal gradient schemes, is set by the keyword `snGradScemes`. It evaluates the gradient normal to the face center which is shared by two cells. It is set to `default corrected`, which indicates it is provided with explicit non-orthogonal correction. The last keyword `fluxRequired` denotes the fields which are solved prior to the flux. `p_rgh`, `pcorr` and `alpha1` are the required fields and calculation loops for flux generation.
It has been particularly difficult to obtain a sufficient small non-orthogonal mesh in the d18v2 case. Therefore non-orthogonality is corrected by setting the `snGradScheme` to `limited .33` and the `nNonOrthogonalCorrector` is set to 1. In accordance to (Hjertager, 2009) a limited scheme is usually more stable than a fully corrected scheme when the non-orthogonality is increased beyond 60°. The required change in scheme was discovered late in the working process, and therefore it is only utilized on the d18v2 case. It is only tested on the d18v4 to ensure the scheme does not produce considerably different results.

3.12 Sampling

Data to be sampled is specified under the keyword `function` in the dictionary `controlDict`. Functions are listed in APPENDIX D. Pressure is simply sampled by specifying probes, which is points within the domain. Five points are measured across the diameter of the vertical pipe. These points are later referred to as zero diameter, ¼ diameter, ½ diameter, ¾ diameter and full diameter. Thus points are distributed with equal distance across the diameter. The pressure probes at zero and full diameter is not set directly on the pipe wall, but at a distance of 5cm from the wall. In addition these five distributed probe points are repeated at six elevations in the vertical pipe. Similarly measurements are done across the diameter of the vent pipe at three different elevations. Such samples are expected to indicate how pressure varies across the diameter of the pipe and the elevation. In the horizontal pipe pressure is sampled in the center of the pipe with 1m intervals in the longitudinal direction.

Apparently more effort is required to sample expressions on an area inside the domain. `swakExpression` is suitable for the purpose. In accordance to Gschaider (2011) `swakExpression` evaluates arbitrary expressions at a specific entity. The additional keyword `valueType` specifies the entity at which the calculation is performed. Entities may be `patch`, `internalField` or `surface`. The expression of choice is defined after the keyword `expression`. Computed values must be presented as a single value, which is defined next to the keyword `accumulations`. One or more of the mathematical methods is listed, and may consist of `sum`, `min`, `max` or `average`. The keyword `valueType` refers to the surface at which the calculations are performed. Surfaces inside the domain must therefore be defined and given a name such that it can be recalled in the `swakExpression`. `CreateSampledSurface` is used and inspired by Gschaider (2010). Information about this `swakFunctionObject` appears to be limited. Even in OpenFoam the description is limited. The surfaces are therefore defined due to trial and error. In OpenFOAM one can simply set `type` to `CreateSampledSurface` in the `controlDict`, and run the solver even if additional keywords are missing. OpenFOAM simply writes an error to the screen and defines the next keyword which is missing to complete the function. Surfaces are defined at -5m, -10m, -15m and -19m elevation as seen in APPENDIX D. These cross sectional areas are used when superficial velocities, the Froude number and the volume fraction is calculated. Mass flow rate and volume flow rate is sampled at the `water_inlet`, `air_inlet` and at the `outlet`. Mass flow rate is simply expressed by velocity normal to the patch multiplied by the total patch area and the density. Density is a function of the volume fraction. Similar the volume flow rate is velocity normal to the patch multiplied with the total patch area. Expressions can be viewed in APPENDIX D. The negative sign in front of the mass flow rate and the volume flow rate at the `air_inlet` is worth mentioning. Since the flow at the `air_inlet` is directed in negative z-direction, a negative mass and volume flow rate occurs. At the `water_inlet` and the `outlet`, they will be respectively positive and negative. The negative sign is inserted at the `air_inlet` such that both inlets obtains a positive value, while the fluid flowing out of the domain is negative.

The superficial water and air velocities are also sampled at surfaces. These parameters are common in two phase flow, and may contribute valuable information to observations, especially at the discharge where air entrainment is expected. Water and air is distinguished by the volume fraction which is a value between 0 and 1. It is possible to obtain a value in between, indicating a mixture of water and air. Air is therefore defined for alpha1<0.5 while water is defined as alpha1>0.5. This distinction between water and air is often used, but not quite realistic because air cannot be dissolved in water. If air is present due to the condition, superficial air velocity is calculated; else it is set to zero. Superficial air velocity is calculated as velocity normal to the surface multiplied by the area at which air is present, which is then divided by the total surface area. Similar expression is used for the superficial water velocity,

see APPENDIX D. An attempt is made to express the densimetric Froude number due to the condition of the volume fraction of water and air. As explained in the theory, these parameters are often used to describe the flow pattern in two phase flow. Although results were poor, and no conclusions could be made regarding the quite strange graphs obtained. The Froude number which is a key parameter is sampled at all the surfaces. It is simply expressed as the velocity divided by the square root of the diameter and the gravitational acceleration, seen in APPENDIX D. The square root of the diameter and gravitational acceleration is calculated prior to insertion in `controlDict`. In each case where the main diameter changes a new value is calculated. An area weighted average of the volume fraction is sampled at all surfaces. It is expected to contribute some to the explanation of the flow regime. Note that the accumulations average, max and min all are calculated in the simulation, although only average values will be discussed. The reader may investigate minimum and maximum values by running the cases in APPENDIX F. In addition a `simpeleFunction` in the swak4Foam dictionary allows the user to calculate the total water volume inside the domain. It is named `volumeIntegrate` and is presented by Bgschaid (2010). This is used to calculate the total water volume as function of time.

3.13 Running the code

The code is run by typing `interFoam`. Functions specified in `controlDict` are sampled, and written to different directories. Residuals are logged during a separate run of the case, where functions in `controlDict` are converted to text. The logfile is growing to a size of about 10 Gb when both functions and residuals are run together. This probably occurs because samples are written to the screen every time step, and therefore it is expected that these results are logged in addition to the residuals. Residuals are logged by typing `interFoam | tee logfile`.
Execution time of the cases has considerably limited the amount of cells in the cases. As the cell size decrease, so does the time step, due to the fixed Courant number of maximum 0.5. The time step in the presented cases is of the order 1e-06. Increased number of cells will reduce the time step further in addition to the extra computational time of computing governing equations for additional cells. The large velocity at the `air_inlet` is also assumed to cause small time steps.
Running a code in parallel is quite common, but this tool is not used in any of the cases. According to Hjertager (2012b) it is not common to run cases with such small number of cells in parallel. This is because dividing the mesh also requires the computer to communicate results across the divided parts. Generally this is more beneficial for large cases where number of cells are of the order of 100 000.

3.14 Post-processing

ParaView and Grace are used to post-process results. Residuals and samples are plotted by use of xmgrace while animations are produced in ParaView. Animations of the pressure, velocity and volume fraction for each case can be viewed in APPENDIX F. In accordance to Lucchini (2009) ParaView is an open-source program used for visualization. Any data saved as VTK format can be read in ParaView. ParaView is simply launched by typing `paraFoam` in the terminal window. In accordance to the Grace Team (2008) Grace is an open-source application used to plot two dimensional graphs. The graphical user interface is an advantage of Grace compared to the script based tool Gnuplot. Grace is launched by typing `xmgrace` in the terminal window.

4 RESULTS AND DISCUSSSIONS

4.1 General observations

As the pump shuts down the vent opens and induces air into the domain. The pipeline runs full of water as water is drained from the pipeline, Figure 4.1. Air entrainment and bubble formation is not observed until the water column is decreased to about 5m in height, Figure 4.2.

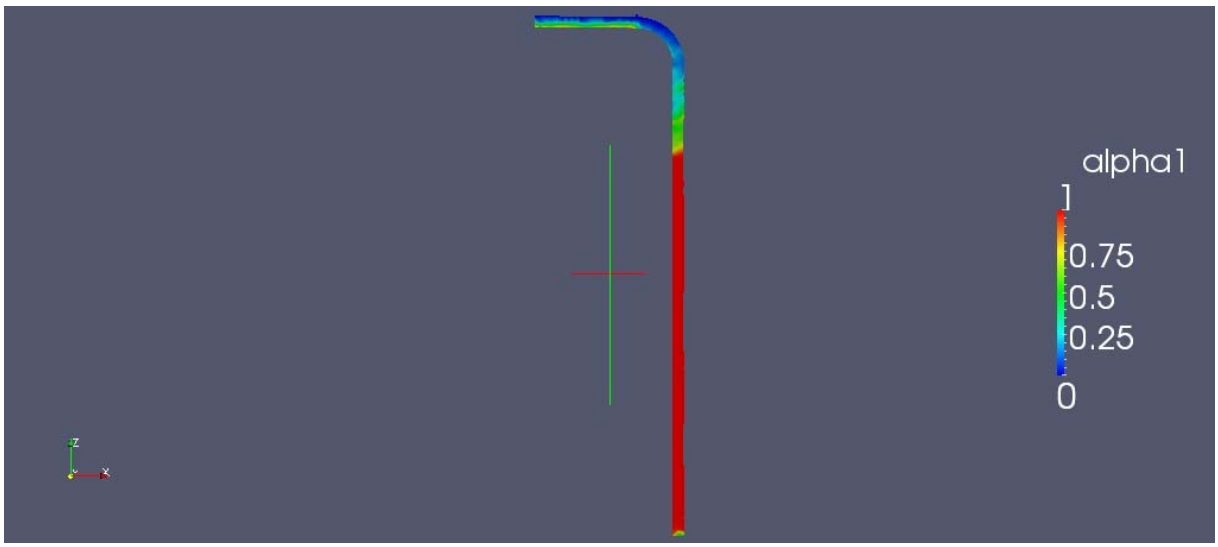

Figure 4.1 The pipeline runs full of water. Case d18v4.

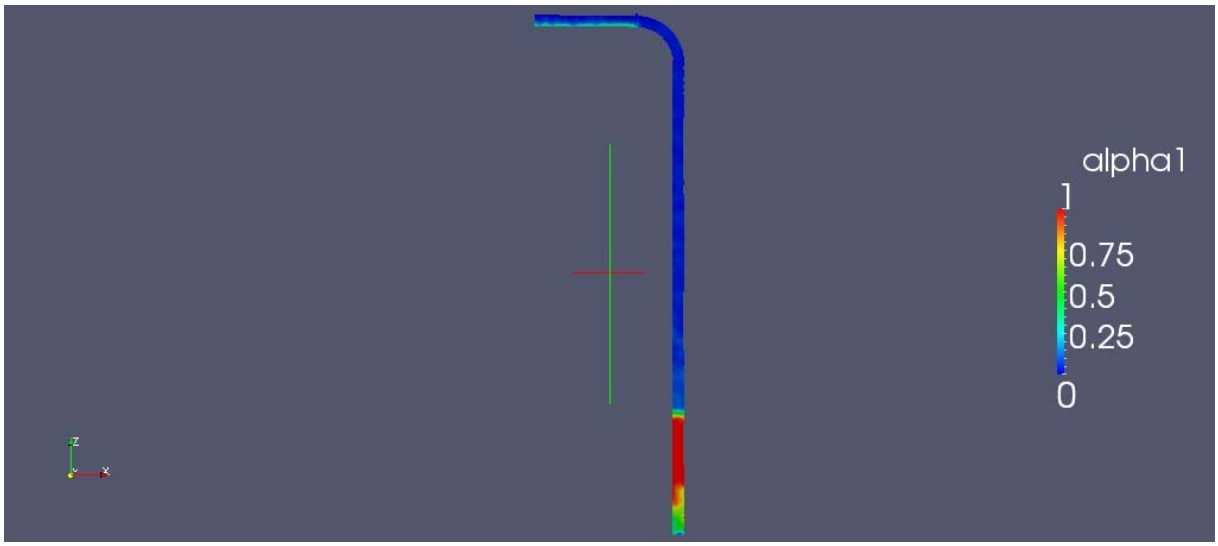

Figure 4.2 Air entrainment at the outlet. Case d18v4.

An unrealistic pressured drop occurs at t=3s as the pump is shut down, Figure 4.3. This pressure drop obtains its highest value close to the water_inlet, and decreases in value at locations further away. The pressure drop is in the range of about -140bar to -9bar. It lasts for a very short time, causing a sharp peak. The largest pressure drop of -140bar occurs in d12v4 while the smallest of -9bar occurs in d24v4. The enormous pressure drop may be due to the instantaneous shut down of the pump caused by the boundary conditions at the water_inlet. In reality it is not possible to shut down instantaneously. There must be a decreasing volume flow rate during a finite time period. Elsewhere in the time line the pressure samples stays within a realistic range. The instantaneous pressure drop is ignored

in further graphs of pressure samples.

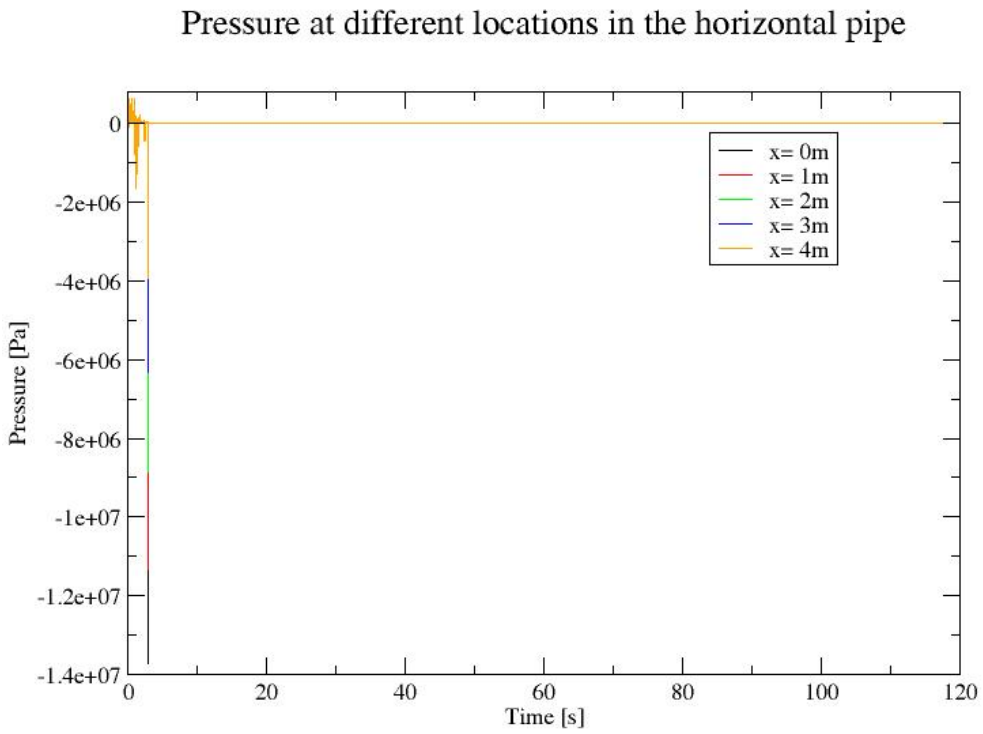

Figure 4.3 Pressure as function of time. Pressure close to the water_inlet occurs at x=0 in the d12v4 case

All vented cases are characterized by an almost linear pressure decrease after the pump shut down. The duration of the linear pressure decrease varies between about 22s and 58s for the different vented cases. In the d18v4 the maximum time period for the linear pressure decrease is about 22s while it is 58s in d18v2. After the linear decrease, the negative pressure increases again. At the outlet and at an elevation of -19m pressure increase is considerably fluctuating, Figure 4.4. Pressure is subject to a steep and even rise at higher elevations in the vertical pipe and resumes an almost atmospheric pressure during a shorter time interval Figure 4.5. Transients of a sudden pump shut down are previously explained in chapter 2.4. As the pump shuts down pressure is decreased close to the water_inlet due to flow retardation. A pressure wave is propagated downstream resulting in a delay of water flow. The induced air from the vent reduces the pressure decrease by recovering an atmospheric pressure above the water level. Animations of the volume fraction can be viewed in APPENDIX F. It is the change in velocity which determines the magnitude of the decreased pressure. As water is drained from the pipe, the water column decreases in height and therefore the maximum decrease in pressure never occur at high elevations. In accordance to transient theory, the pump shut down can be classified as rapid. This indicates that closure time is less than the pipe period, which is the time for a pressure wave to reach the outlet and return a restoring wave back into the pipeline. The pressure will therefore decrease until it is restored to an atmospheric pressure by the top induced air.

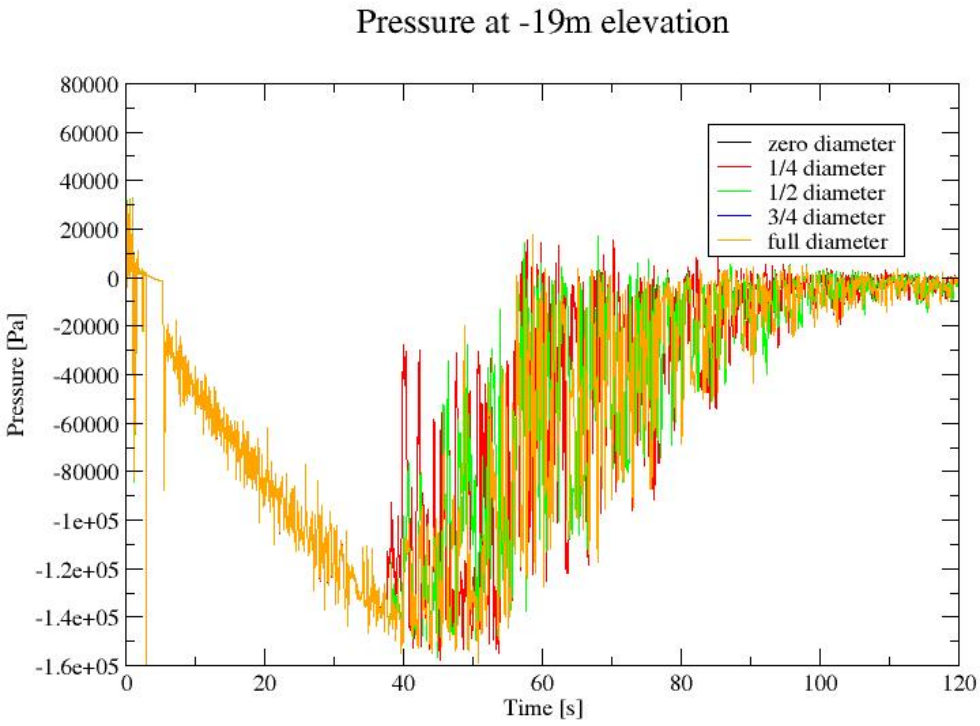

Figure 4.4 Pressure as function of time. Fluctuations occurs when pressure increases, case d12v4.

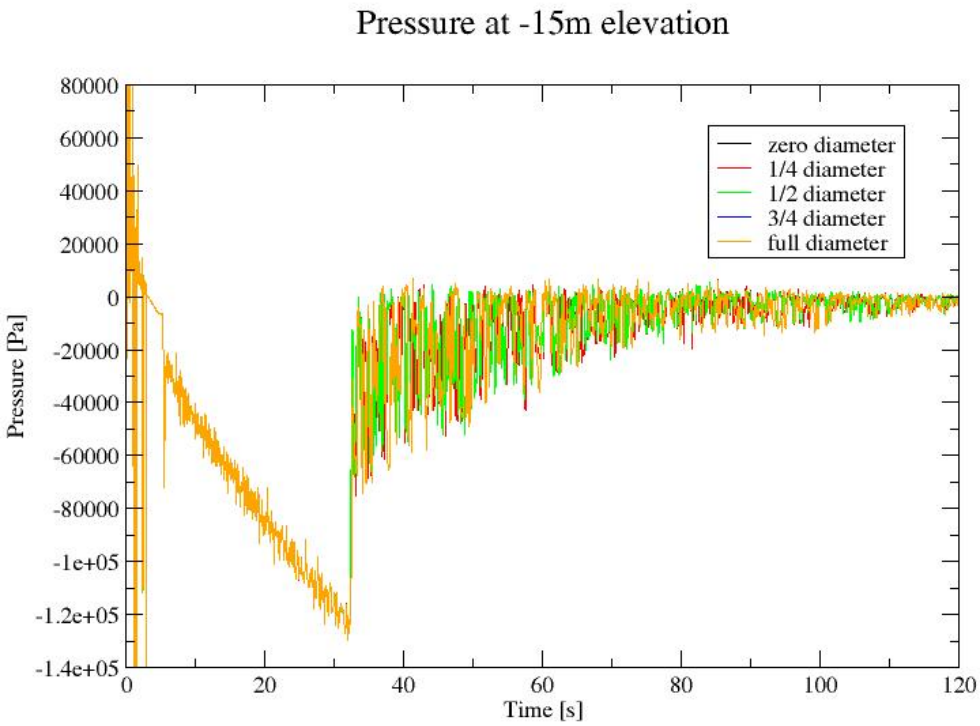

Figure 4.5 Pressure as function of time. A sharp and even pressure rise in case d12v4.

Pressure drops to -1.7bar right after the pump shut down in the unvented pipe d18u, Figure 4.6. Atmospheric pressure is gradually retained, but oscillations are significant.

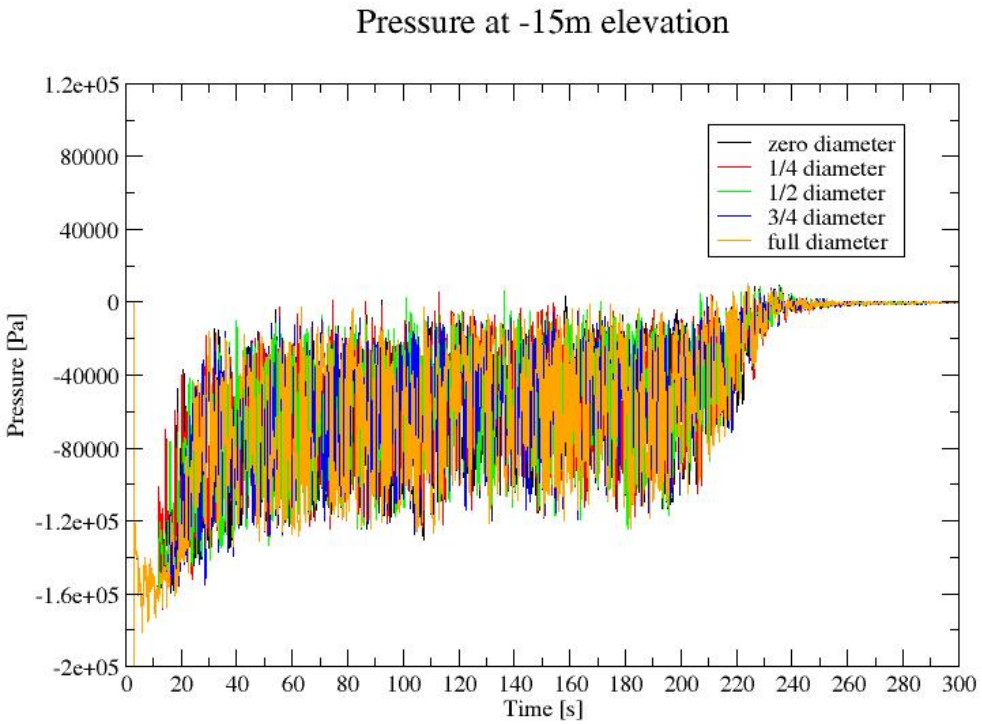

Figure 4.6 Pressure as function of time. Pressure drop in d18u.

Sampled Froude numbers are generally highest at the outlet where water is free to the atmosphere. At other locations, the Froude number is simply in the range of 0.35-0.4. No slug flow is observed during the simulation. Only at the end when the weight of the water column seems to balance surface tension forces, air is entrained at the discharge. As presented in the theory part, the Froude number is used when water has a free surface, either at a boundary or in a bubble. Samples of the Froude number inside the pipe flowing full, does not apply to any physical problems where the Froude number has its significance. Therefore it is mainly the Froude number at the outlet which will be discussed. The time averaged Froude numbers at the outlet is given in Figure 4.7.

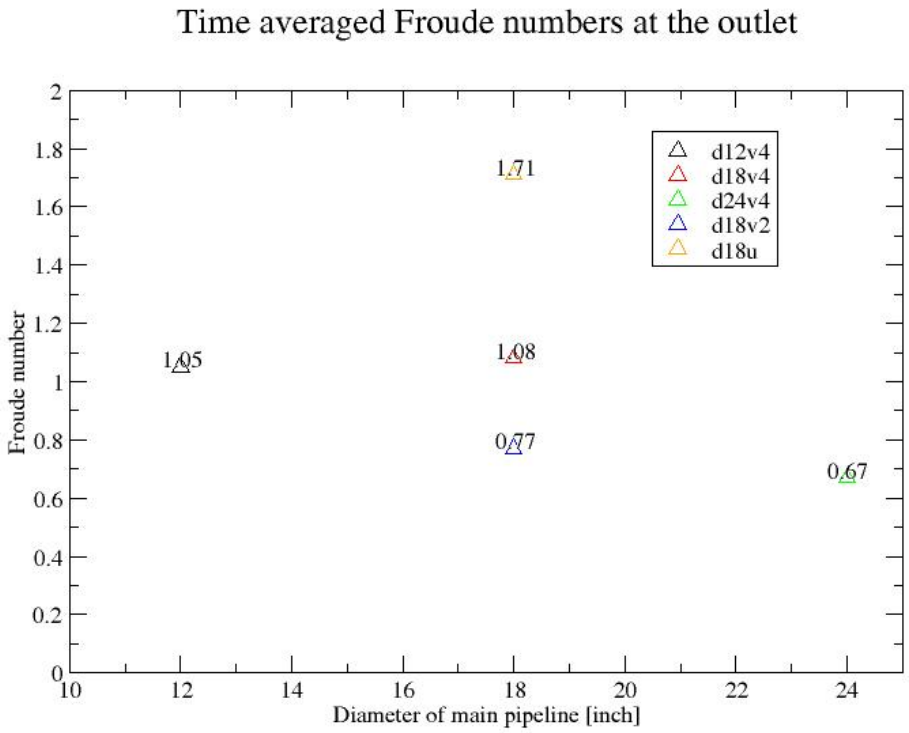

Figure 4.7 Froude number as function of time. Time averaged Froude numbers at the outlet for the different cases.

4.2 Case d18v4

The pressure decreases linearly until it reaches a minimum of about -1.6bar at -19m elevation, see Figure 4.8. At elevations of -15m and -10m pressure decreases to a minimum of -1.4bar and -0.9bar respectively, Figure 4.9 and Figure 4.10. This is because atmospheric pressure is retained earlier at these elevations as the water column decreases in height. Animation of the pressure can be viewed in APPENDIX F. At elevations of -5m and -2m, even smaller pressure drops of -0.4bar and -0.02bar occurs, as shown in Figure 4.11 and Figure 4.12. Note that the pressure does not start to decrease linearly until at t=13s. As seen in the animation of alpha1 in APPENDIX F, the water column starts to decrease down the pipeline at t=8s. The pressure is however maintained close to the atmospheric due to three rapid openings of the valve in the period between 8s and 13s.

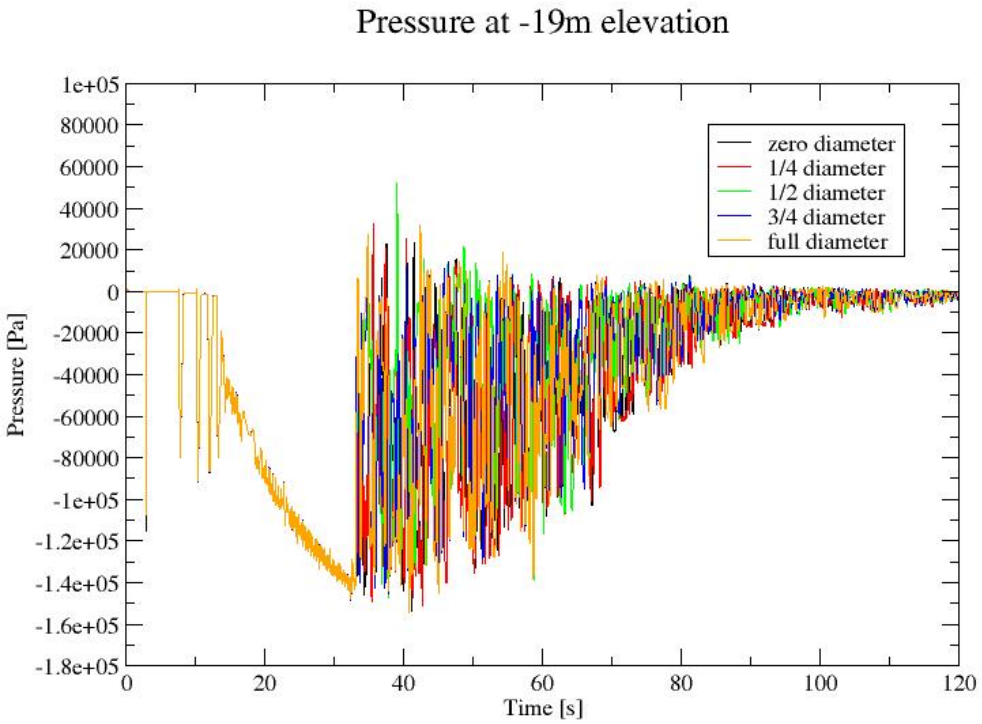

Figure 4.8 Pressure as function of time. Pressure drop at -19m elevation in d18v4.

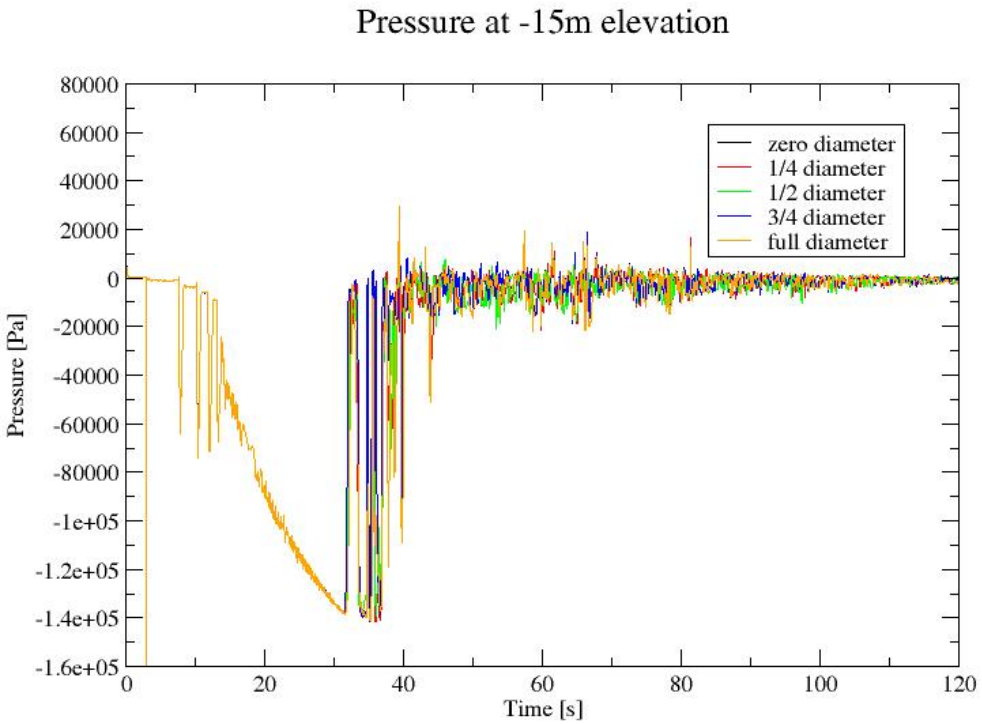

Figure 4.9 Pressure as function of time. Pressure drop at -15m elevation in d18v4.

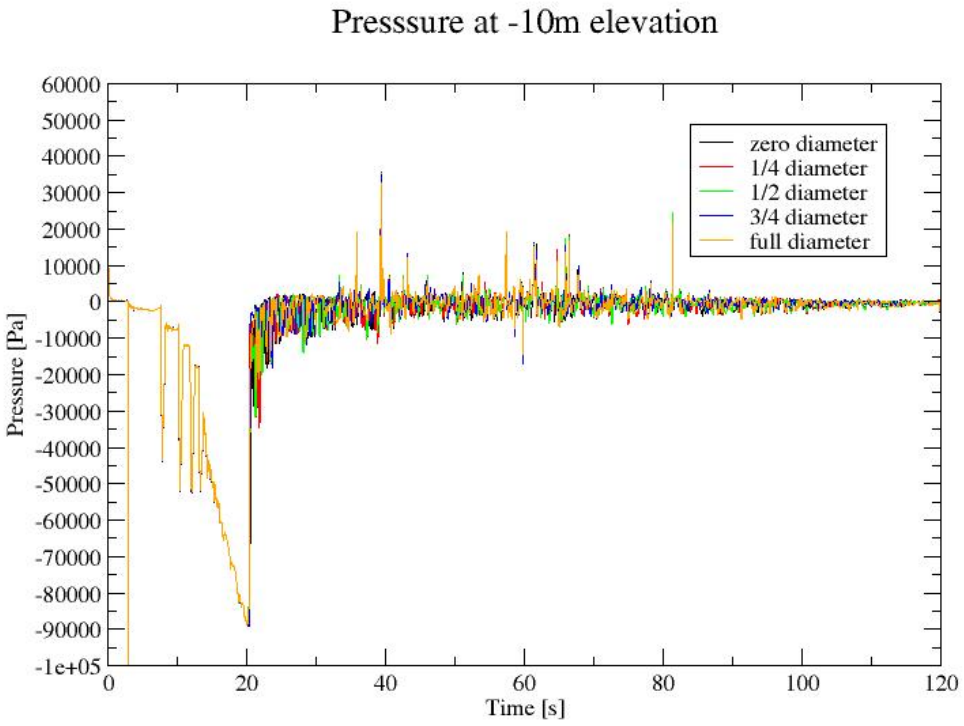

Figure 4.10 Pressure as function of time. Pressure drop at -10m elevation in d18v4.

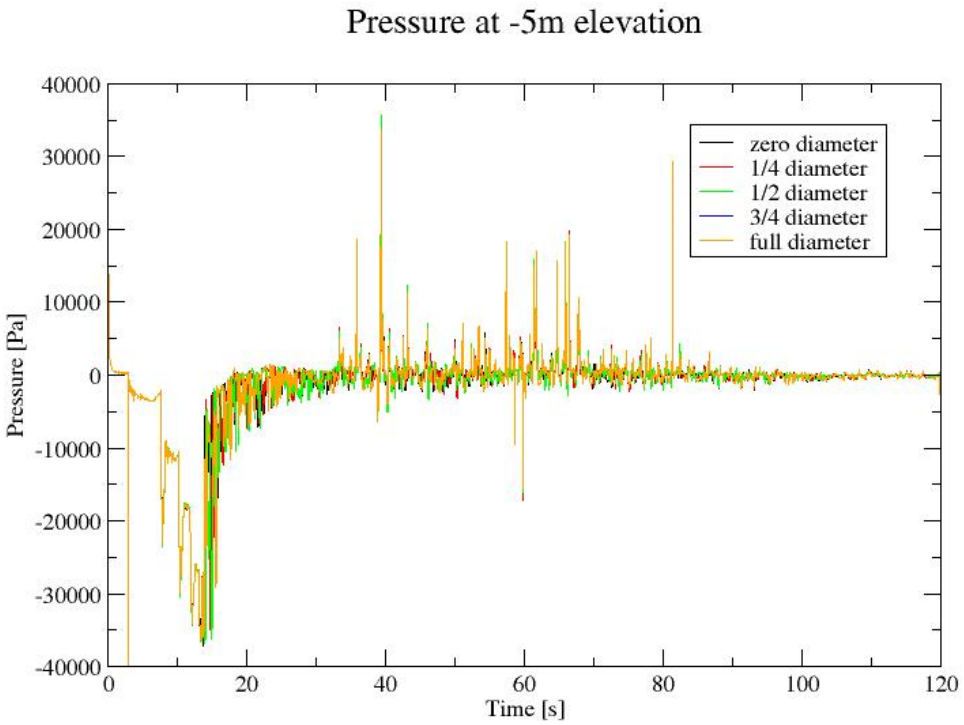

Figure 4.11 Pressure as function of time. Pressure drop at -5m elevation in d18v4.

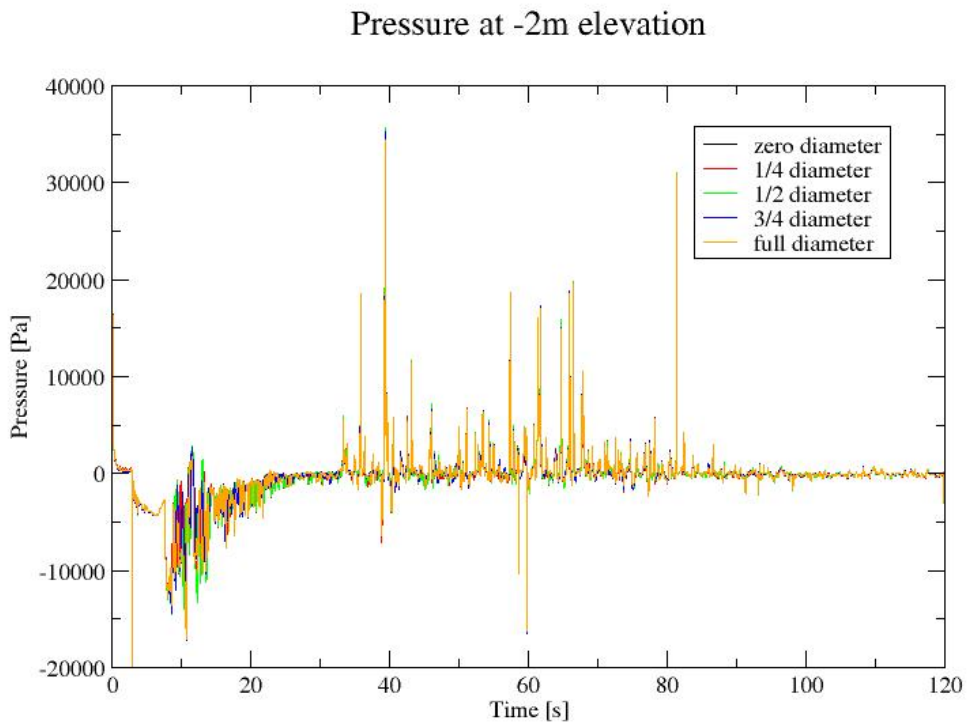

Figure 4.12 Pressure as function of time. Pressure drop at -2m elevation in d18v4.

The pressure is considerably constant along the pipe diameter as seen in the above figures. Only at the outlet pressure varies for different pipe diameters, Figure 4.13. Here zero diameter and ¼ diameter remains at about an atmospheric pressure, while pressure decreases at the other pipe diameters. This may be due to replacement of air at zero and ¼ diameter as water flows out at other diameter locations. Air enters only slightly within the outlet at early stages of the simulation. See animation of alpha1 in APPENDIX F.
The change in volume fraction at different elevations can be viewed in Figure 4.14. It generally supports how atmospheric pressure is achieved at higher altitudes as the water column is drained. Water is maintained at a value of 1 until air sufficiently fills the upper part of the column causing the value to drop close to 0. The change in volume fraction is steep everywhere unless at -19m elevation and at the outlet. At the outlet, air is always present. Air is entrained when the water column decreases sufficiently in height. This occurs when the water column is about 5m high, as observed from the animation of alpha1. Air entrainment also affects the volume fraction value at -19m elevation, and it is clear that entrainment occurs after about 35s of the simulation.
When the pump is running, the water volume flow rate is observed to be 0.16 m^3/s at the water_inlet. This equals 576 m^3/h and is less than 700 m^3/h from which the water inflow velocity is calculated. The error may be due to the coarse mesh consisting of only 6 faces at the water inlet. Thus flow rate is calculated as velocity times a smaller area than is provided in reality. The coarseness of the cross section can be seen in Figure 4.15.

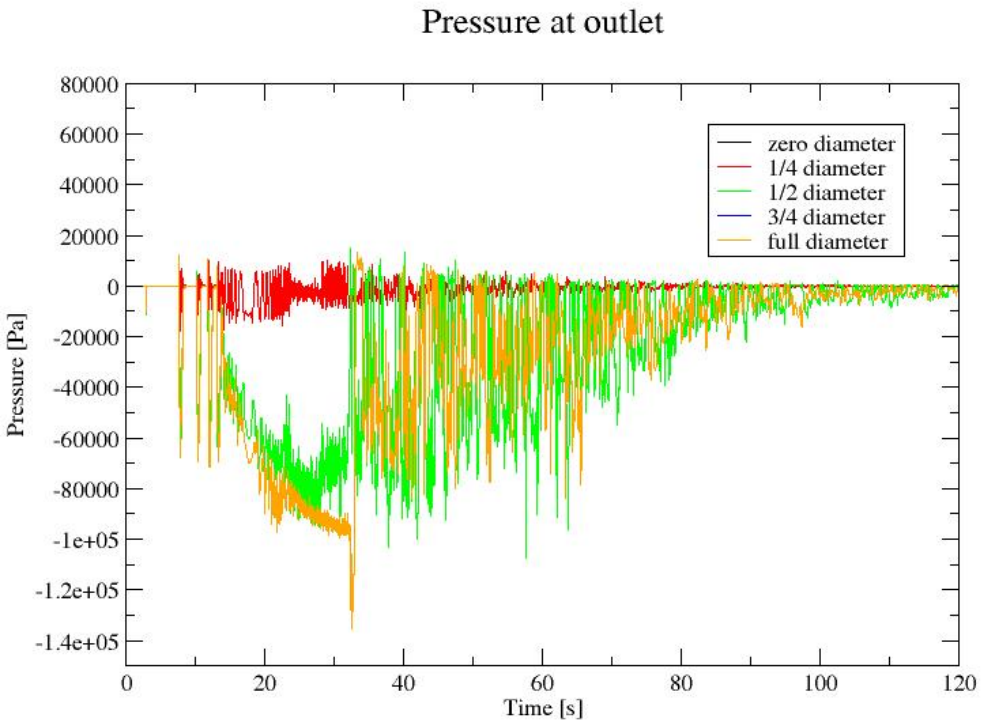

Figure 4.13 Pressure as function of time. Pressure drop at outlet in d18v4.

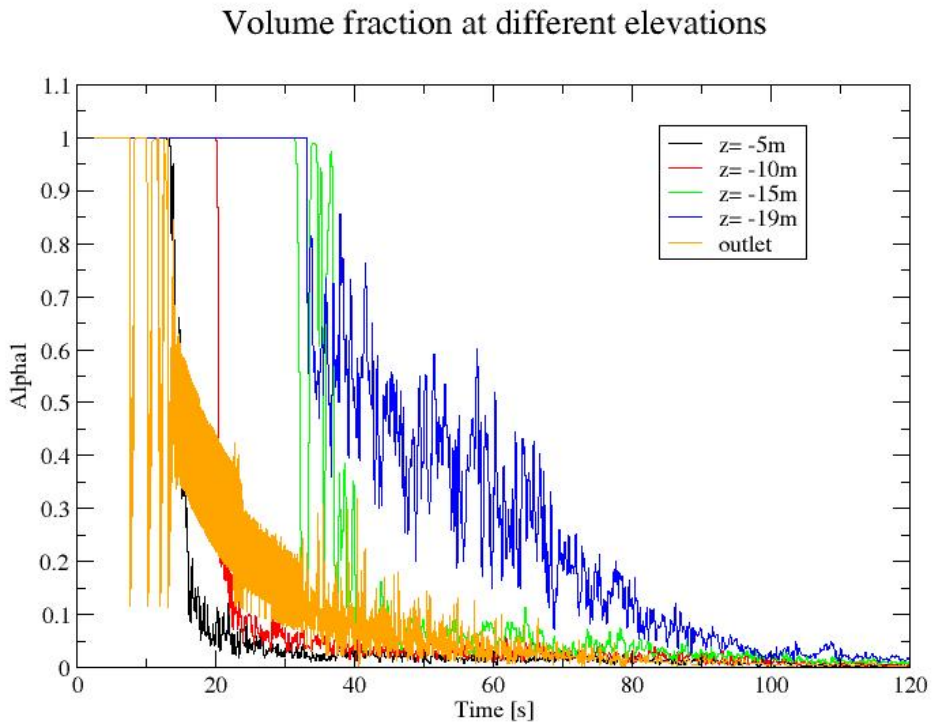

Figure 4.14 Volume fraction as function of time. Volume fraction at different elevations in d18v4.

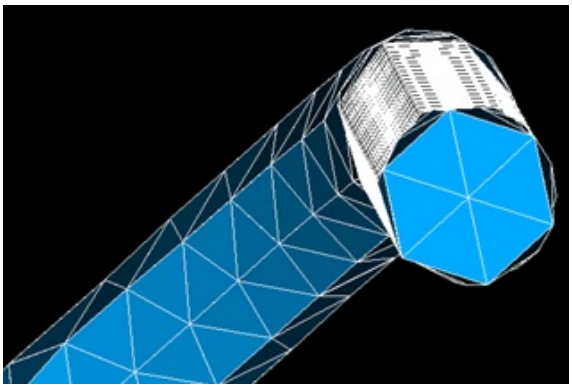

Figure 4.15 Coarseness of the water inlet in d18v4.

The volume flow rate and mass flow rate can be viewed in Figure 4.16 and Figure 4.17. During the period the pump is running the air flow rate at the vent is zero. As the pump shuts down, volume flow rate is decreased at the outlet, due to flow retardation. In the time interval between 3s and 13s, the valve is observed to close three times with decreasing time intervals. Closing of the vent clearly affects the pressure downstream. As seen in Figure 4.9 and Figure 4.10 peaks of pressure drops are observed in the same time interval. In the time interval between 13s and about 35s, air is induced at a smooth decreasing flow rate. At the outlet the flow rate is mirrored, but generally higher flow rates occur. The pressure drop of air is the only driving force for air to be induced at the air_inlet. At the outlet, however weight of the water column contributes to a higher flow rate. Air flow rate at the air_inlet generally controls the flow rate of water at the outlet.

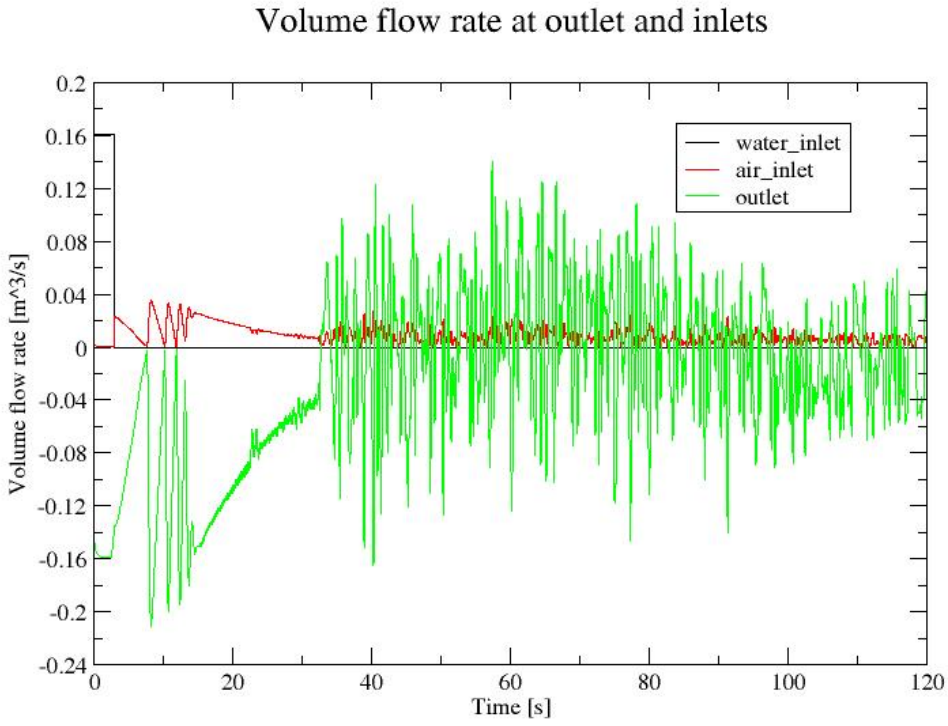

Figure 4.16 Volume flow rate as function of time. Behavior of valve during pump shut down in d18v4.

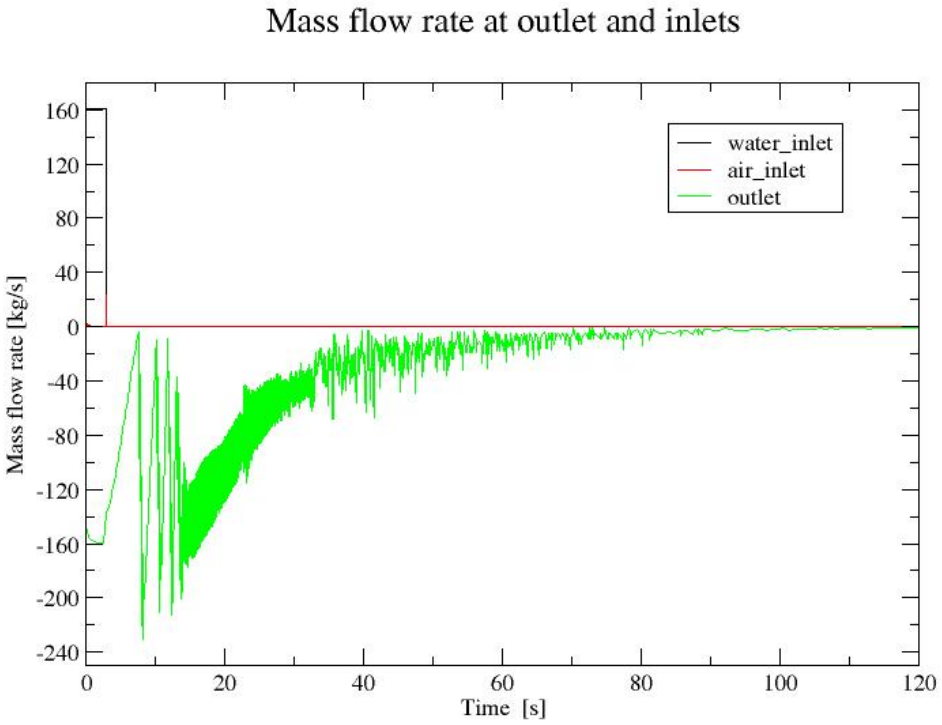

Figure 4.17 Mass flow rate as function of time. Almost zero mass flow rate is observed for the vent in d18v4.

After about 35s of simulation, pressure has retained an atmospheric pressure at the air_inlet as seen in Figure 4.18. The valve closes as atmospheric pressure is reached, causing retardation of the water flow and the pressure drops sufficiently to open the valve again. Air is induced and atmospheric pressure is retained. This cycle of events continues until the water is fully drained from the pipeline. As can be seen in Figure 4.16 a positive inflow occurs at the outlet when the valve is closed for a small time interval. Thus air is entrained at the outlet. It is at this stage distorted bubbles are entrained at the outlet. See animation of alpha1 at t=35s in APPENDIX F. Probably it is the air entrainment at the outlet which causes the pressure at the air_inlet to fluctuate between atmospheric and a slightly negative pressure, see Figure 4.18. Due to presented theory, bubbles will entrain when forces due to hydraulic head of the water column equals the surface tension for water and air. The mass flow rate shown in Figure 4.17 cannot describe the valve operation sufficiently. Water is thousand times denser than air, causing the valve to appear as closed at all times. There is a mass flow rate at the valve, but it is close to zero compared to mass flow rate of water at the outlet. The total water volume inside the pipeline as function of time is shown in Figure 4.19. It decreases exponentially. Initially, water volume decreases rapidly, but as time continues the rate diminishes. The decrease in volume flow rate at the outlet supports this finding.

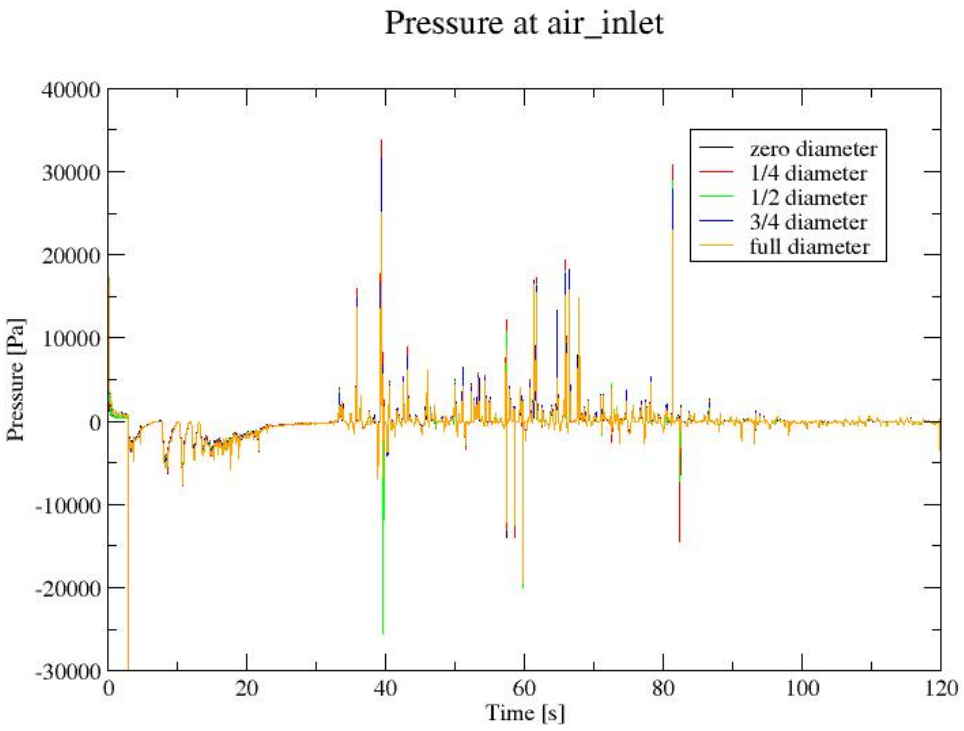

Figure 4.18 Pressure as function of time. Pressure variations at the air_inlet in d18v4.

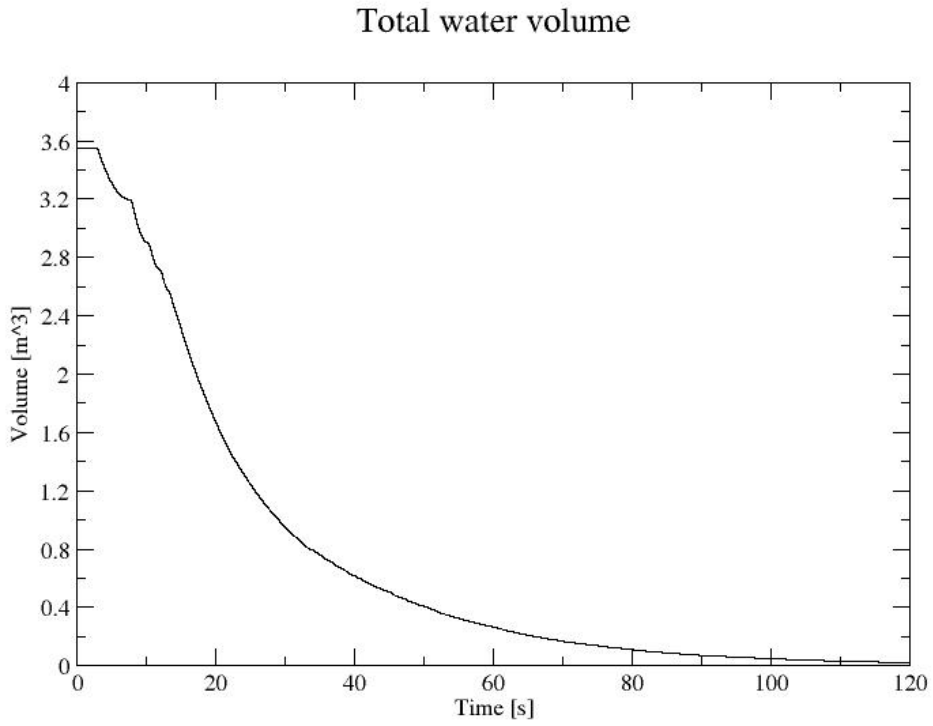

Figure 4.19 Water volume as function of time in d18v4.

The superficial water velocity at different elevations can be viewed in Figure 4.20. It is evident that the superficial water velocity decreases to zero at high elevations as the water column is reduced and air is replaced. The superficial water velocity behaves somewhat differently at the outlet and at -19m elevation. At -19m elevation the superficial water velocity decreases as the flow rate in the water column decreases until at t=35s. Considerably air entrainment occurs causing a higher and more oscillating superficial water velocity. Three peaks are also observed as the valve is closed three times initially in the simulation. Flow is retarded at valve closure, and increases again as the valve is opened. At the outlet superficial water velocity obtains high values in the time interval between 13s and 28s. In this time range the vent stays open and the volume flow rate at the outlet is even. The superficial water velocity reduces to zero afterwards. Since water is defined as alpha1>0.5, water still exists at the outlet, but the volume fraction is less than 0.5.

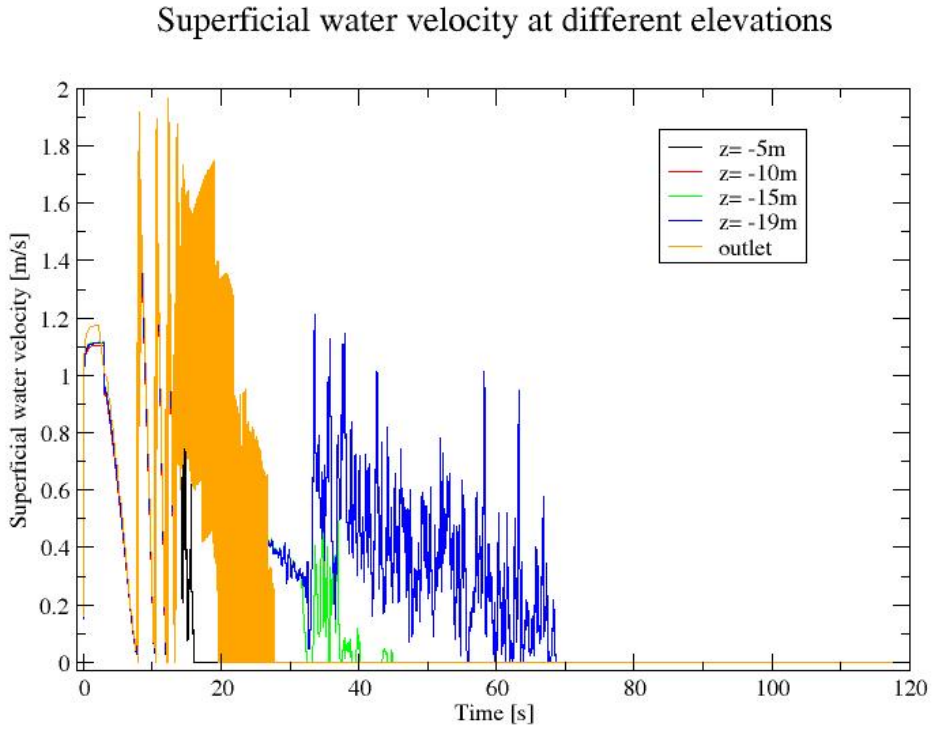

Figure 4.20 Superficial water velocity as function of time at different elevations in the vertical pipeline of d18v4.

Superficial air velocity is shown in Figure 4.21. Three peaks are observed initially in the simulation. Closure of the valve causes retardation of water flow in the pipe, and therefore an increase in the superficial air velocity at the outlet. As the valve opens it decreases again. The peaks obtain a value of about 2m/s, which is similar as for the superficial water velocity. Superficial air velocity is zero at all elevations where the pipe is running full of water. As the water column decreases, superficial air velocities of non-zero value occur. This is first obtained at -5m elevation and proceeds down the vertical pipe. Superficial air velocity which occurs due to air recovery is small and in the range of 0.5-1m/s. At the outlet however, a free surface exist where air is continuously supplied. In the range between 13s and 35s superficial air velocity is increased, but it is also subject to fluctuations. The increase occurs as the volume flow rate at the outlet decreases, Figure 4.16. As considerable amounts of air is entrained at t=35s, the superficial air velocity remains its magnitude but becomes even more fluctuating. Probably air is entrained at an increasing velocity as pressure is decreased to its minimum. In addition valve closure causes flow retardation, increasing the probability of air entrainment. In multiphase flow it is known that air flows at a considerably higher velocity

than water due to the large difference in viscosity. This supports the described behaviour.

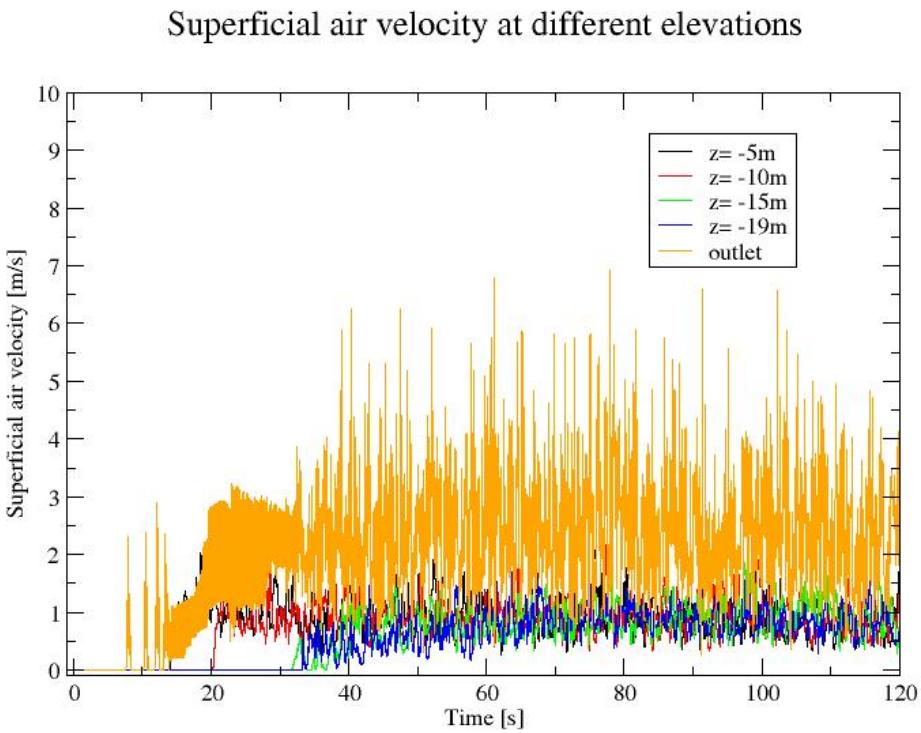

Figure 4.21 Superficial air velocity as function of time. Case d18v4.

The Froude number as function of time is seen in Figure 4.22. As previously explained only Froude numbers at the outlet will be evaluated. The Froude number obtains a time averaged value of 1.08, and the Froude number is considerably fluctuating over time. The behavior is similar to what occurs for the superficial air velocity. It is probably this velocity which affects the high value of the Froude number. The measured Froude number does not consist with theory. It is the velocity of water which shall be used in calculation of the Froude number. A first thought would be to use the superficial water velocity instead, but this is generally zero at the outlet in large parts of the simulation due to volume fractions lower than 0.5. It is difficult to draw a conclusion for the Froude number, but an attempt is made.
The observed flow pattern does not cause a weir where air is drawn into the flow from the top surface. This event is one in which the design criteria is used for, and it is probably more realistic to occur in a process vessel than in a pipe due to the width of the water surface. The other occasion for which the design criterion is used is in slug flow. Slug flow is said to occur if the Froude number decreases below 0.3 in a stagnant liquid. According to Fabre and Line (2010) C. S. Martin found the coefficient C_∞ to be 0.7 for the bubble rise velocity in counter current down flow. In addition the velocity of the down flowing liquid must be added in accordance to equation (2.33). Elongated bubbles are expected to cause oscillations. Some distorted bubbles can be viewed in the animation of alpha1 at t=35s. According to Simpson (1968) the design criterion of 0.3 is found for an unvented pipe with submerged outlet. He found that flooding occurred at a Froude number of 2 when the unvented pipe is free to the atmosphere.
It seems most realistic that the design criterion applies for the event of a weir in a process vessel. If entrainment occurs downstream, such as at the outlet due to bubble entrainment it does not make sense to design for bubble entrainment to occur. If the flow rate of water is sufficiently reduced due to the Froude number criterion elongated air bubbles are entrained due to presented theory. However, this is in a stagnant liquid. If the liquid is flowing

downwards countercurrent to the elongated bubble, the bubble rise velocity must be greater, and hence the Froude number criterion is greater. In accordance to the theory of Fabre and Line (2010) a more comprehensive equation of (2.33) is referred to when the influence of a flowing liquid is considered. Due to research, transient theory applies greater knowledge to future design, than does the Froude number. Flow control should be an objective, where one strives to reduce rapid changes in flow velocity. Rapid changes cause a pressure wave to propagate and water hammer to occur. Water hammer may be the phenomenon which causes vibrations during shut down, rather than oscillations due to air entrainment. It must be stated that meshes are very coarse, which affects the accuracy of the results. Results are however assumed to indicate the behavior of the flow regime.

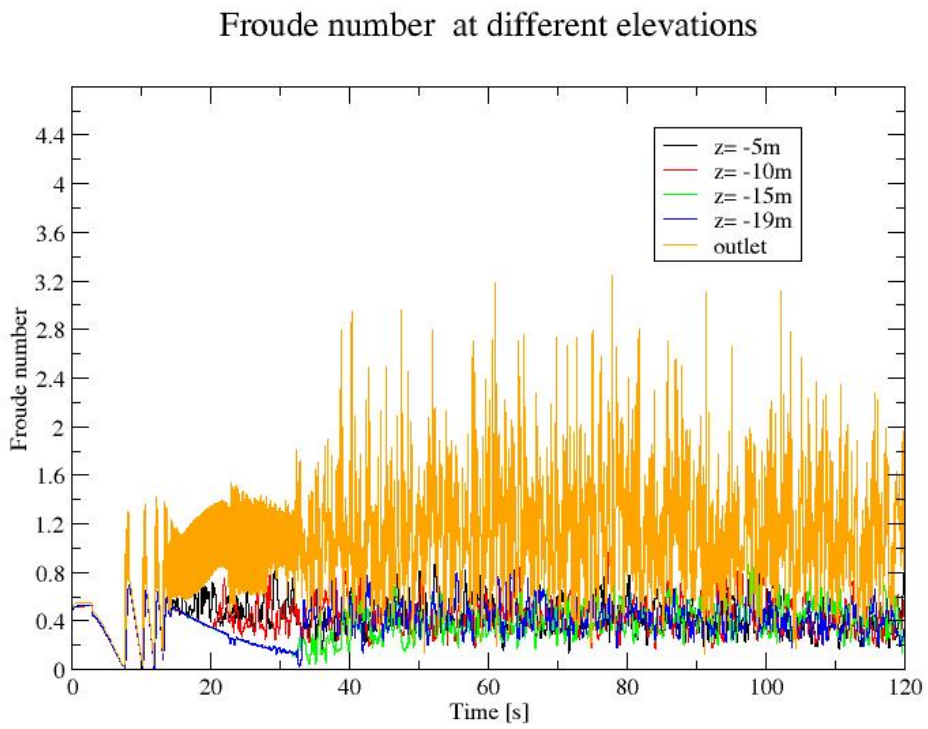

Figure 4.22 Froude number as function of time. Case d18v4.

4.3 Case d12v4

The minimum pressure is -1.6bar at -19m elevation as previously. Minimum pressure drop at -15m, -10m, -5m and -2m are respectively -1.3bar, -0.7bar, -0.4bar and -0.07bar. Pressure drops are therefore approximately equal at the different elevations. The linear pressure decrease starts at an earlier stage of t=5s compared to t=13s in d18v4. Due to the animation of alpha1 in APPENDIX F, the water level in the vertical pipe starts to decrease at t=3s which is earlier than for d18v4. The difference in bend geometry and volume of the horizontal part may affect the time at which pressure starts to decrease. The bend is designed with a radius equal to five times the diameter of the main pipeline. In the subsequent sections only results which considerably differ from d18v4 will be presented.
The volume flow rate differs at an early stage in the simulation. Figure 4.23 shows the volume flow rate. As the pump shuts down, the valve is closed only once before a smooth volume flow rate is maintained from t=8s to t=32s. Generally the maximum air flow rate at the second valve opening is reduced to 0.01m^3/s from previously 0.035m^3/s in d18v4. The volume flow rate at outlet obtains a smooth decreasing flow rate of -0.06m^3/s. In d18v4 the volume flow rate is -0.18m^3/s. A reduced diameter of the main pipeline causes a considerably

decrease in the volume flow rate at both the outlet and air_inlet. Flow retardation of a smaller water volume may cause a smaller pressure decrease close to the air valve. Pressure at air_inlet can be viewed in Figure 4.24.

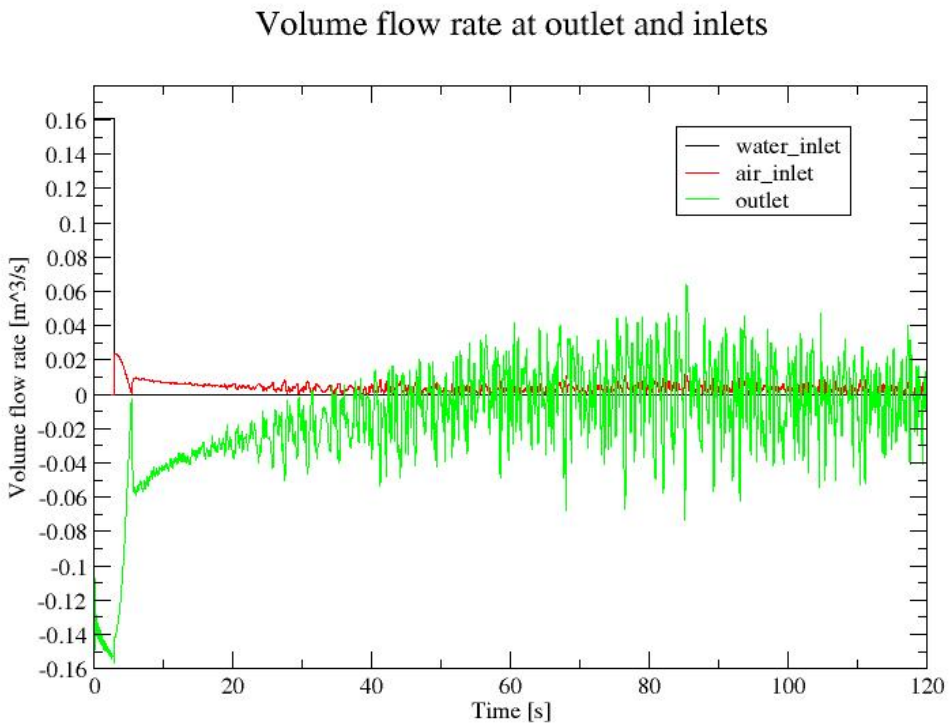

Figure 4.23 Volume flow rate as function of time. Case 212v4.

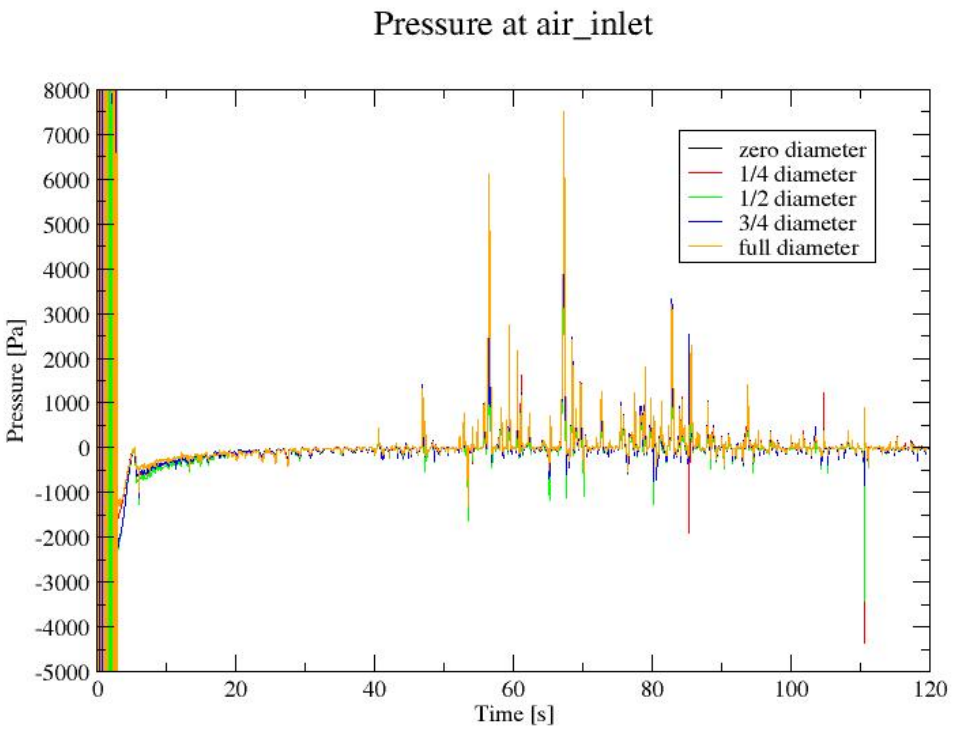

Figure 4.24 Pressure as function of time at the air_inlet in the d12v4 case.

Since the time used to empty the pipeline is equal for a smaller volume, a decreased flow rate seems reasonable. The total volume of the pipeline is 1.6m^3 against the previously 3.6m^3, see Figure 4.25. Water volume decreases exponentially as previously. Samples are elsewhere similar to that observed in the d18v4 case.

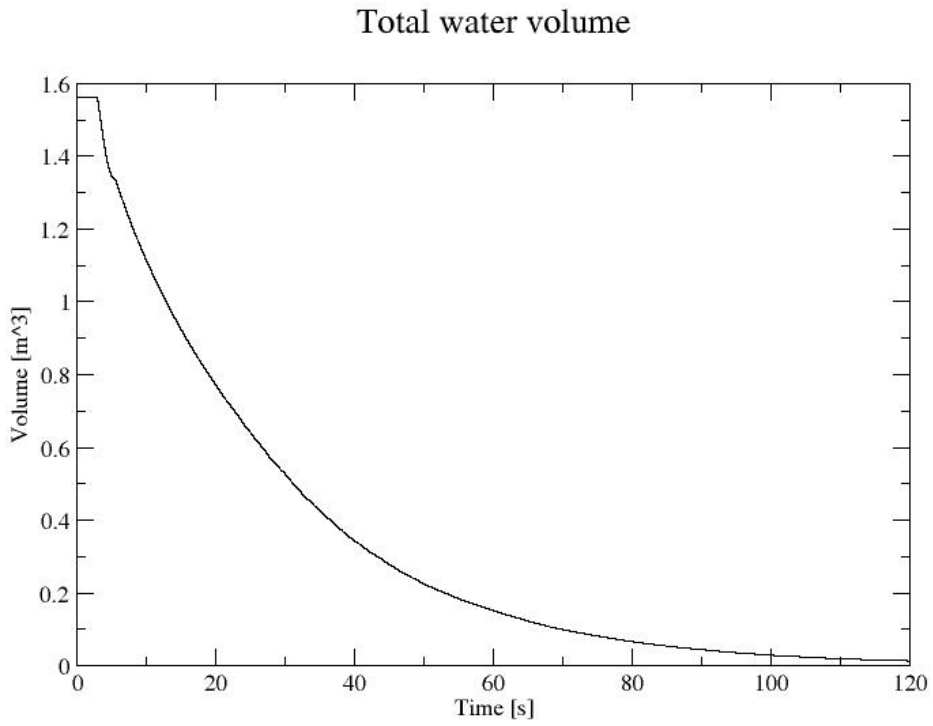

Figure 4.25 Total water volume as function of time in the d12v4 case.

4.4 Case d24v4

Pressure decreases linearly to a minimum of -1.6bar at -19m elevation as previously. At -15m, -10m, -5m and -2m pressure decreases to a minimum of respectively -1.4bar, -0.9bar, -0.4bar and -0.02bar. Thus the pressure drop is approximately unchanged at the different elevations. Note that the time at which the pressure starts its linear decrease is delayed to t=13 as in d18v4, Figure 4.26. In the time interval between 3s and 13s the horizontal pipeline is completely filled with air and at t=13s the water column starts to decrease in height. Animation of alpha1 can be viewed in APPENDIX F.
Volume flow rate differs for the new pipe dimension as seen in Figure 4.27. The air flow rate is increased to 0.05m^3/s and the flow rate at the outlet starts to decrease from -0.3m^3/s at t=13s. Retardation of an increased water volume as the pump shuts down may be the reason why a higher pressure drop occurs close to the valve Figure 4.28. Higher pressure drop at the air_inlet, causes a higher volume flow rate. The pressure drop at the air_inlet is of about -0.07bar compared to -0.05bar for the d18v4 cases.

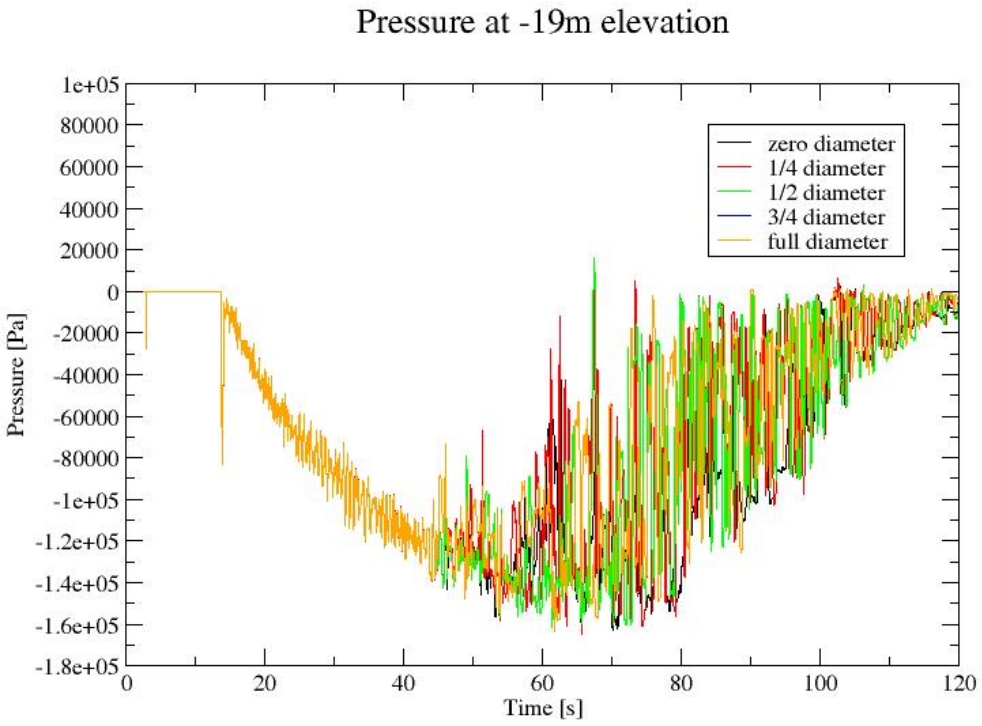

Figure 4.26 Pressure as function of time. The figure shows how pressure decrease is delayed in d24v4.

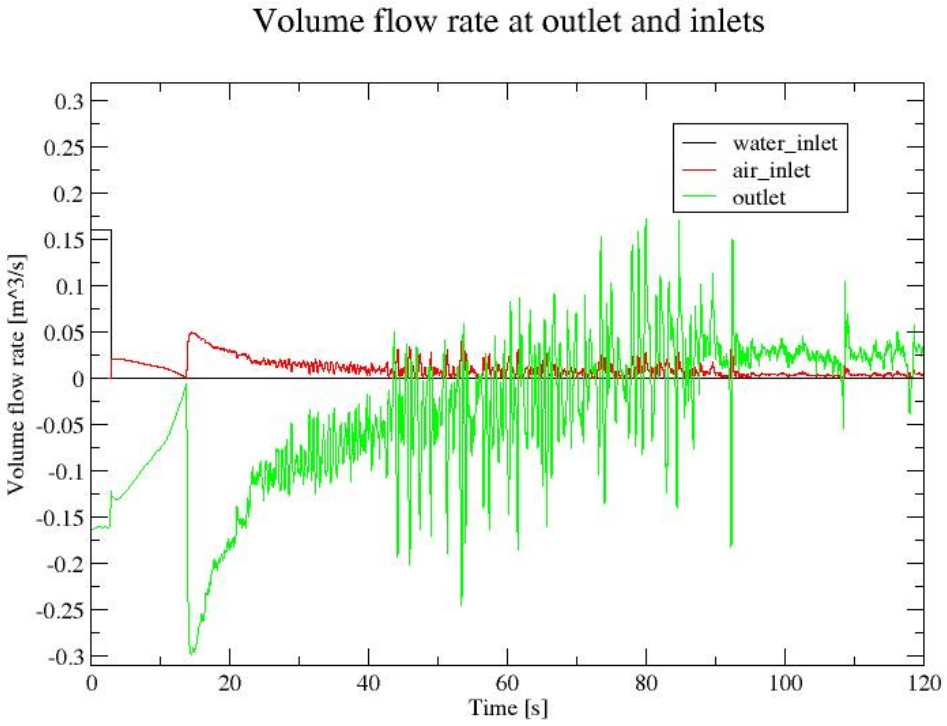

Figure 4.27 Volume flow rate as function of time. Case d24v4.

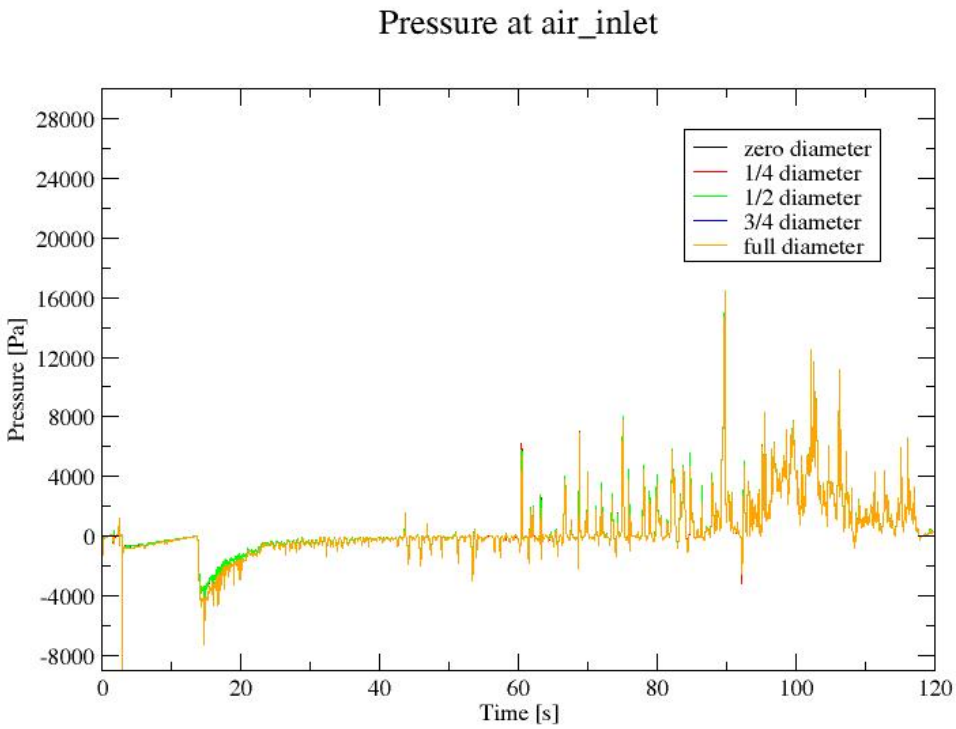

Figure 4.28 Pressure as function of time at the air_inlet in the d24v4 case.

The total water volume of the pipeline is increased to 6.4m^3 as viewed in Figure 4.29, but still equal amount of time is required to empty the pipeline. The flow rate must therefore increase. Volume decreases exponentially, as previously. The time averaged Froude number at the outlet differs some, and is equal to 0.67. In Figure 4.30 Froude number is seen as function of time for the d24v4 case. As explained earlier there is a greater volume fraction of air than water at the outlet. From the animation of alpha1, one observes that disturbed air bubbles does not entrain until at t=60s. A higher flow rate probably causes a delay in air entrainment. The decrease in superficial air velocity and therefore the Froude number may be due to the increased volume flow rate of water.

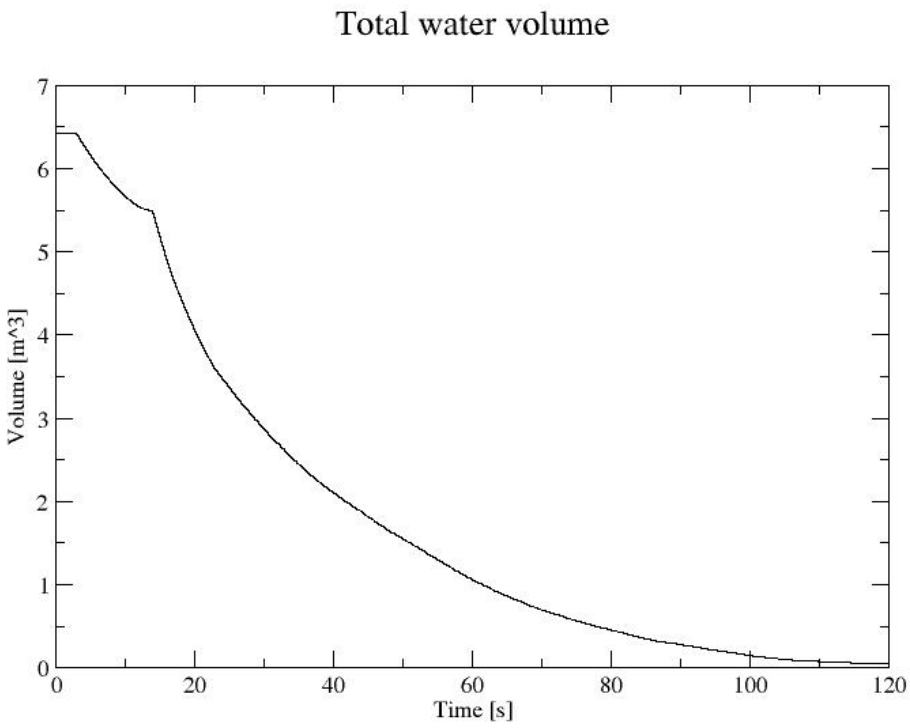

Figure 4.29 Total water volume as function of time. Case d24v4.

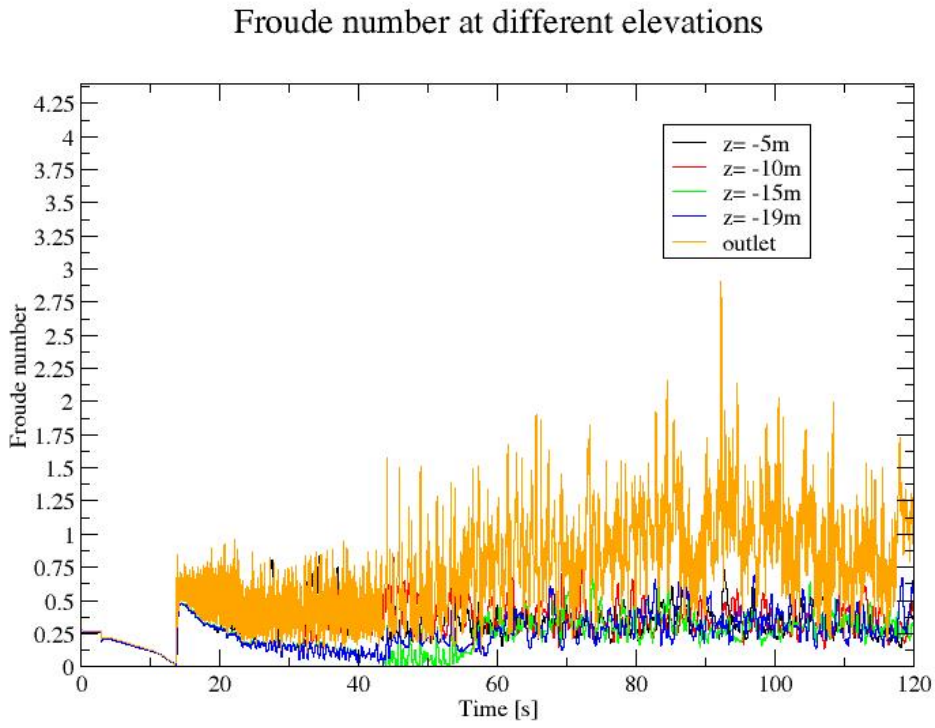

Figure 4.30 Froude number as function of time. Case d24v4.

4.5 Case d18v2

Minimum pressure drop of -1.6 bar occurs at -19m elevation. At -15m, -10m, -5m and -2m elevation pressure is decreased to -1.3bar, -0.9 bar, -0.4bar and -0.02bar. Propagation of the pressure drop is therefore approximately unchanged. Similar to the d24v4 case the linear pressure decrease is delayed and it does not start until at t=67s, see Figure 4.31.

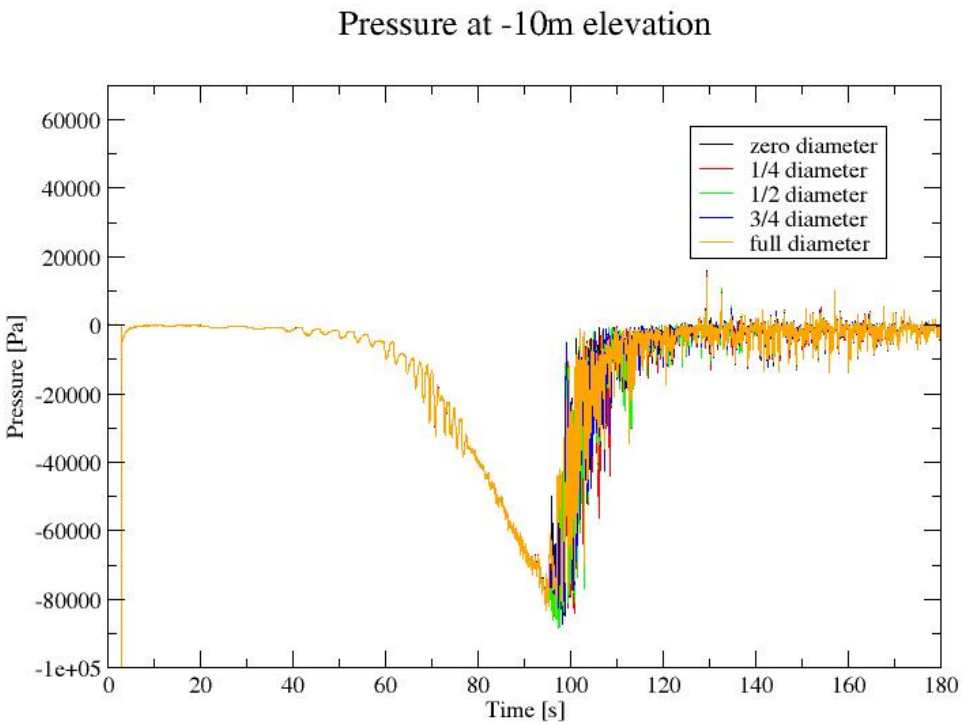

Figure 4.31 Pressure as function of time. Delayed linear pressure decrease of d18v2.

The interval from t=3s to t=22s is subject to a decreasing flow rate at outlet and at the air_inlet, see Figure 4.32. Volume flow rate is considerably more equal in magnitude at the outlet and the air_inlet compared to the d12v4 and d18v4 cases. This causes a decreased flow retardation and may be the reason for the delayed pressure drop. Air flow rate is reduced to 0.018 m^3/s compared to the previous 0.025m^3/s for vented pipes subjected to a 4" vent. The valve is subject to 12 closures prior to air entrainment in the interval t=22s to t=67s. The increased number may be caused by the change of `snGradScheme` to `limited .33`. This scheme is tested on d18v4 and it did not seem to considerably change the results when compared to the `corrected` scheme. The emptying time, minimum pressure and time at which considerably air entrainments occur remains the same. The only difference is some increased number of valve closures as seen in Figure 4.32. A pressure wave is not propagated until at t=67s. At t=60s water starts to decrease in elevation in the vertical pipe, see animation of alpha1 in APPENDIX F. Pressure remains about -0.01bar until the pressure drops to -0.08bar at t=67s, see Figure 4.33. In the d18v4 pressure drops to -0.05bar.
At the last and longest valve opening the flow rate differs from previous cases. It is considerably more fluctuating, and it can be seen to first increase and then decrease again. The largest volume flow rate of air is now 0.012m^3/s, while at the outlet it is decreased to about -0.075m^3/s. The valve controls the flow at the outlet, and thus the volume flow rate fluctuates here too. When the valve closes, flow retardation occurs causing the pressure to decrease in front of the air_inlet. A negative pressure causes the valve to open again. The air and water volume flow rates are lower than what is observed for the d18v4 case which is equal in water volume. The air volume flow rate is reduced from 0.035m^3/s in d18v4 to 0.012m^3/s in d18v2 for the last long valve opening. This implies that more time is used to

recover an atmospheric pressure in front of the valve. This may be the reason why the d18v2 uses an additional minute to drain the pipeline. Some higher velocities are observed at the air_inlet of d18v2 than in other cases, see the animation of Uz in APPENDIX F. However, it varies a short time interval, and is only slightly increased compared to the d18v4 case. The highest velocity observed in the d18v2 is about 70m/s while it is 60m/s in the d18v4 case. Considerably air entrainment occurs at t=118s, see Figure 4.32. The water column height is about 4.9m compared to 4.3m in the d18v4 case. The reading of the alpha1 animation and the graph of the total water volume may affect the correctness of the calculated water height. The water column heights are therefore assumed considerably equal.

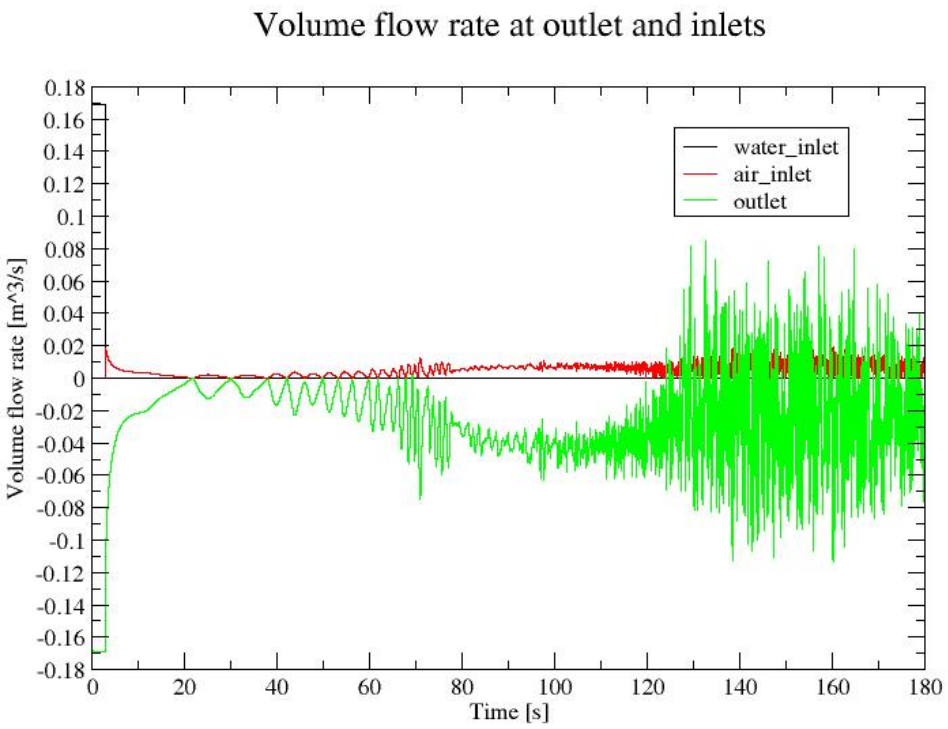

Figure 4.32 Volume flow rate as function of time. Case d18v2.

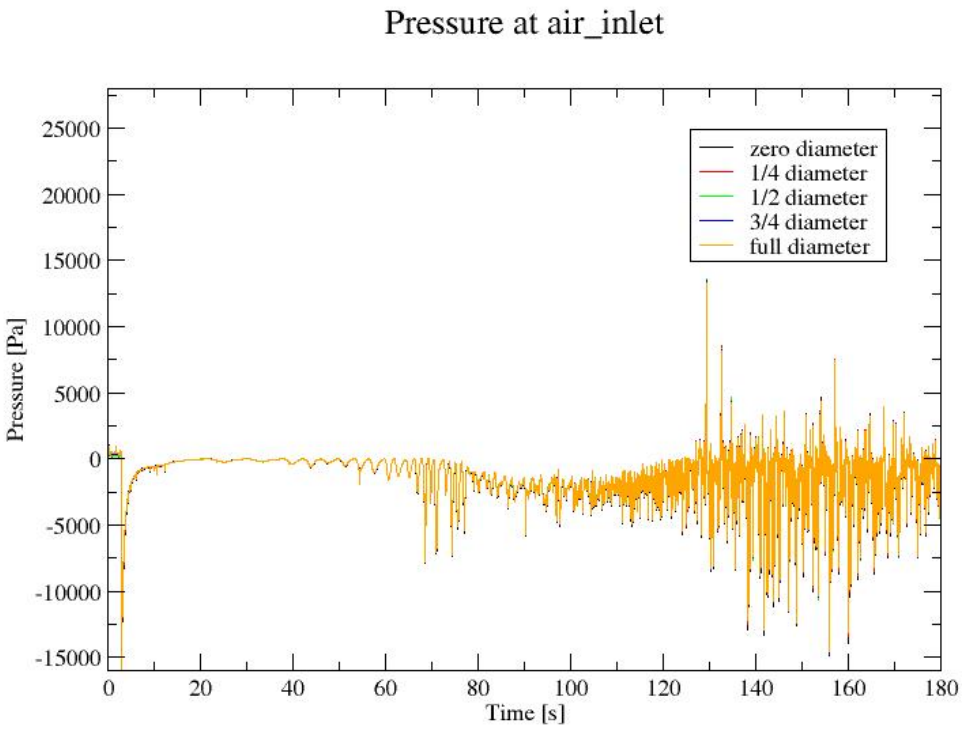

Figure 4.33 Pressure as function of time. Increased pressure drop at t=67s in d18v2.

The Froude number is subject to a lower value of 0.77 as in the d24v4 case. The Froude number and superficial air velocity varies with time as seen in Figure 4.34 and Figure 4.35. The air and water volume flow rates remains low until the valve opens for a longer period at t=67s. The superficial air velocity is close to zero in this time interval. As the valve opens between t=67s and t=118s, the superficial air velocity increases some, but highest values occurs after t=118s when considerably air entrainment occurs. The time averaged Froude number may obtain a smaller value since it is averaged over a larger time interval. In addition the decrease in water column height and therefore the propagation of a pressure wave is delayed. Thus considerably air entrainment does not occur until t=118s which is after 2/3 of the whole time interval.

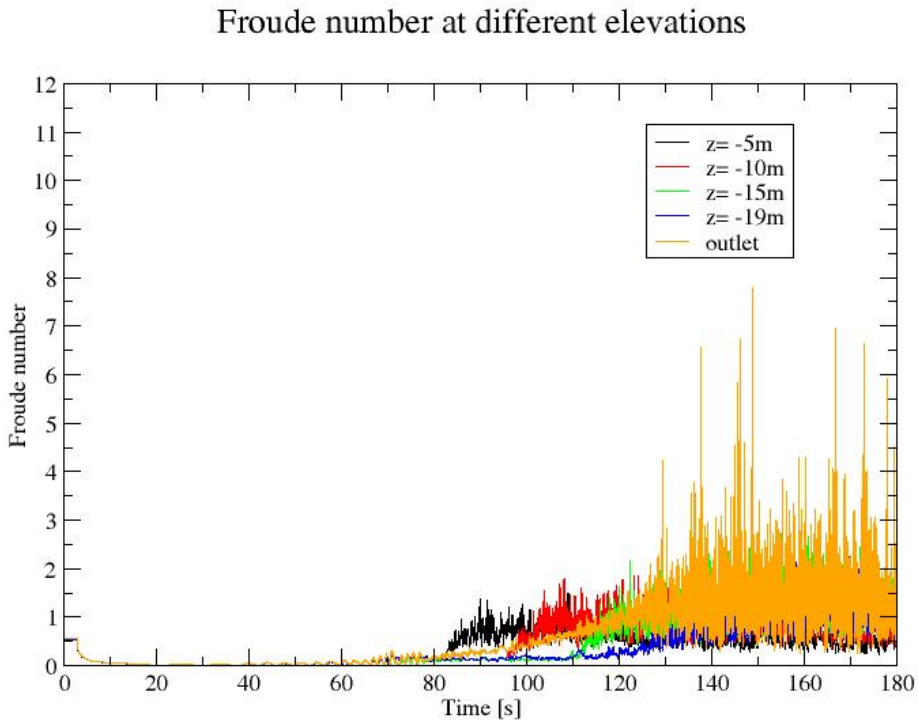

Figure 4.34 Froude number as function of time. Case d18v2.

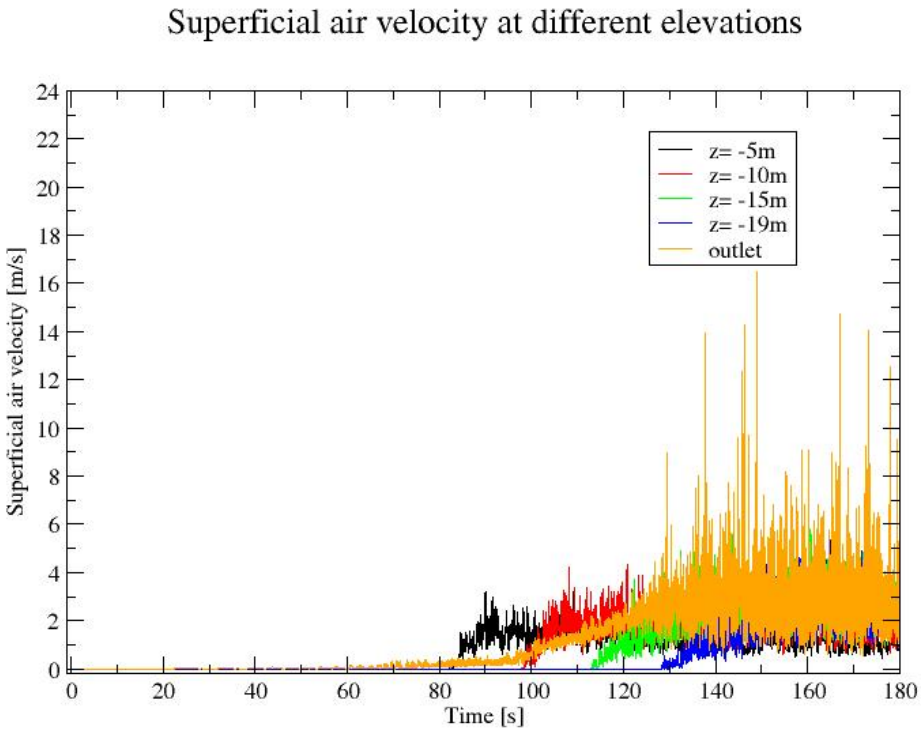

Figure 4.35 Superficial air velocity as function of time. Case d18v2.

The decrease in total water volume differs from the vented cases subjected to a 4" vent, see Figure 4.36. In the interval between t=3s and t=67s the valve is subject to 12 closures, and the water is only removed in the horizontal part of the pipe. This may explain the mild slope observed. After about 67s the valve stays open for a longer time interval, and the air volume flow rate is somewhat increased. The valve controls the volume flow rate at the outlet, and the water volume flow rate is therefore increased. This supports the steeper slope observed. When considerably air entrainment occurs, the slope decreases again. Air entrainment at the outlet and considerably opening and closing of the valve reduces the rate at which water is drained form the pipeline.

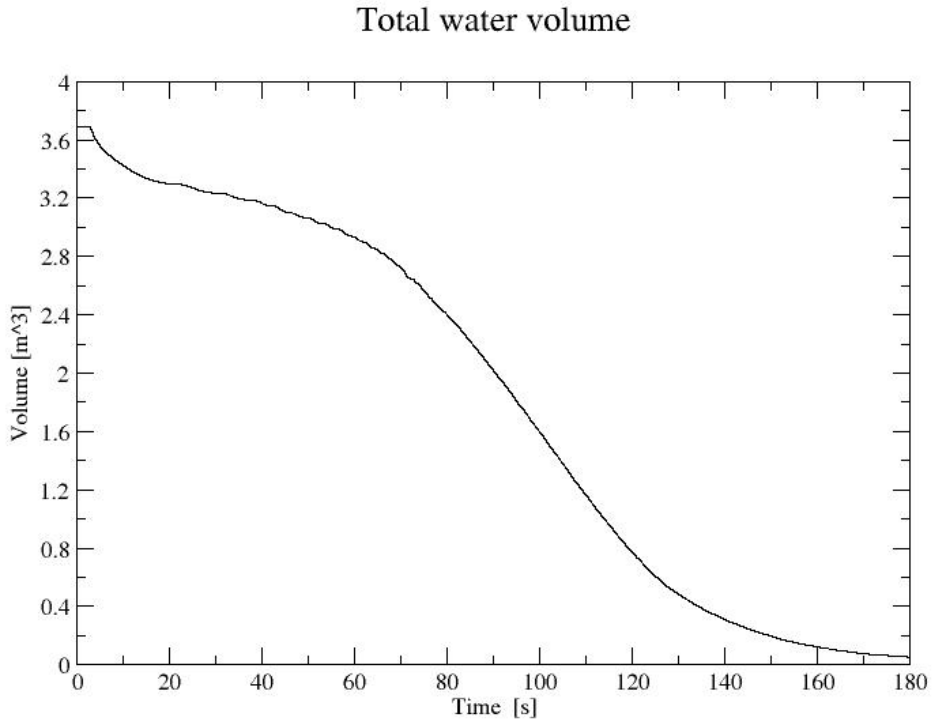

Figure 4.36 Water volume as function of time. Case d18v2.

4.6 Case d18u

The unvented case behaves quite differently. As the pump shuts down, pressure drops to its minimum of -1.7bar. This can be explained by the sudden flow retardation when the velocity turns to zero inside the pipeline. Due to transient theory, the magnitude of a pressure wave is proportional to the change in water velocity. Without a vent the change in velocity increases to its maximum. The pressure drops to a minimum of -1.7bar at all elevations as seen in Figure 4.37, Figure 4.38, Figure 4.39, Figure 4.40 and Figure 4.41.

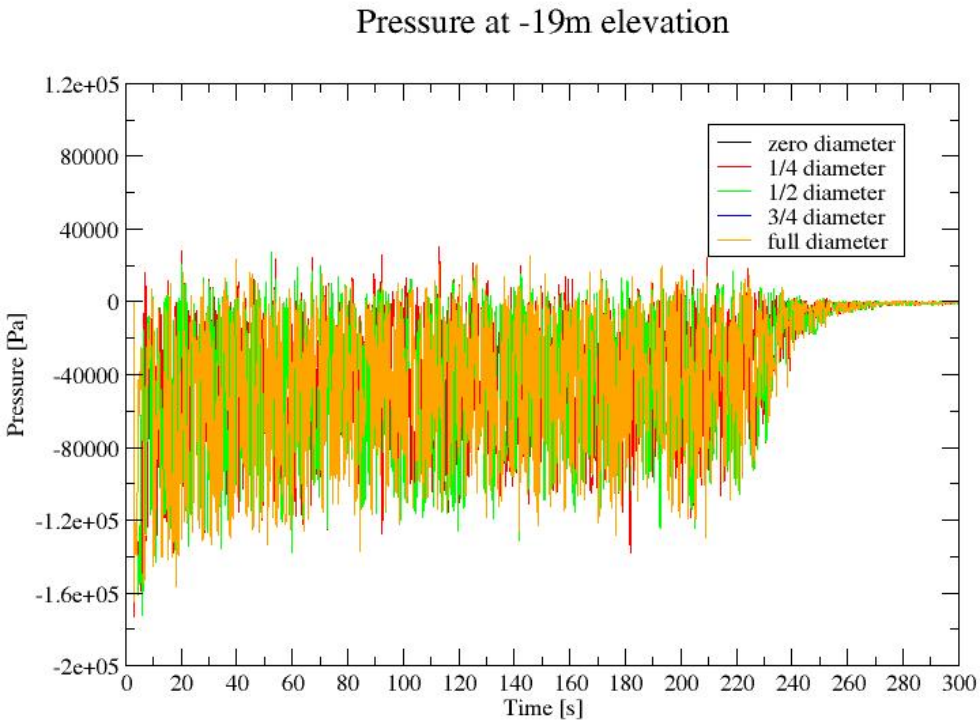

Figure 4.37 Pressure as function of time. Case d18u.

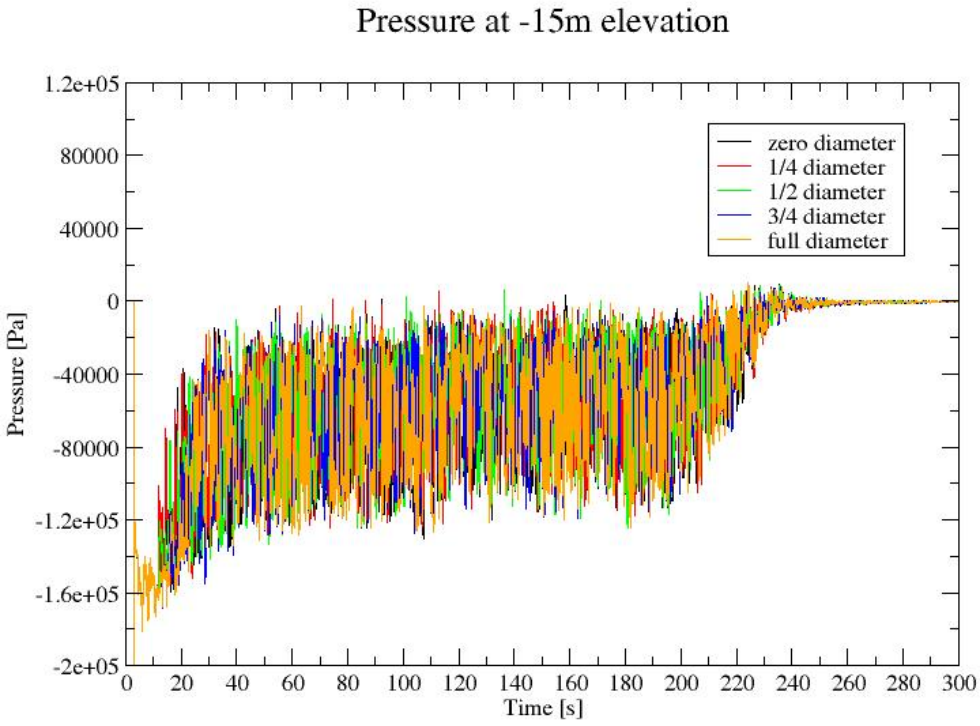

Figure 4.38 Pressure as function of time. Case d18u.

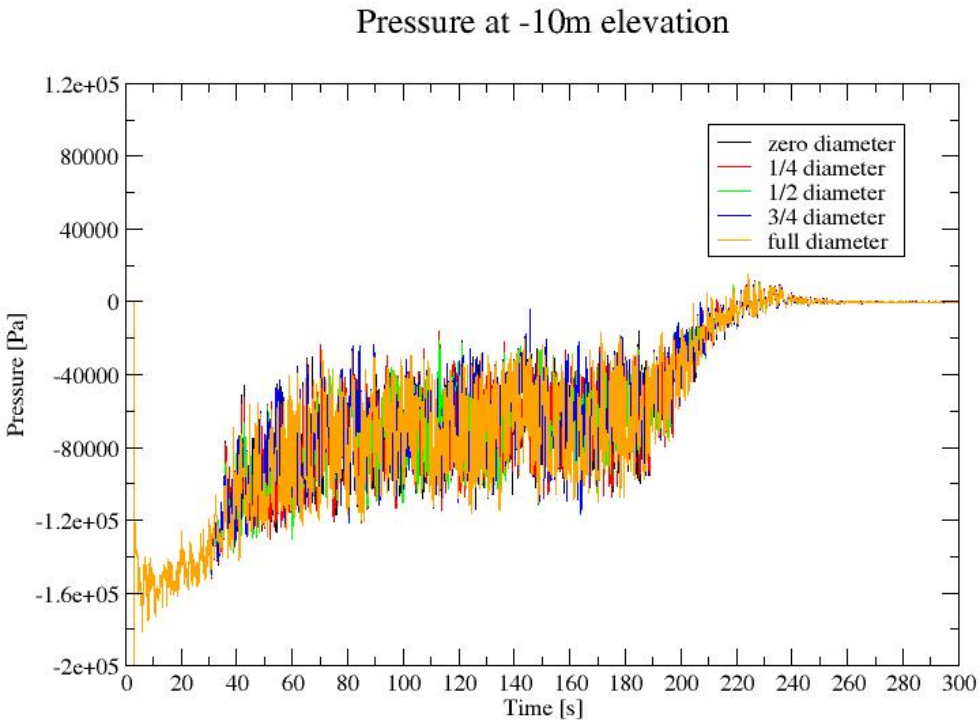

Figure 4.39 Pressure as function of time. Case d18u.

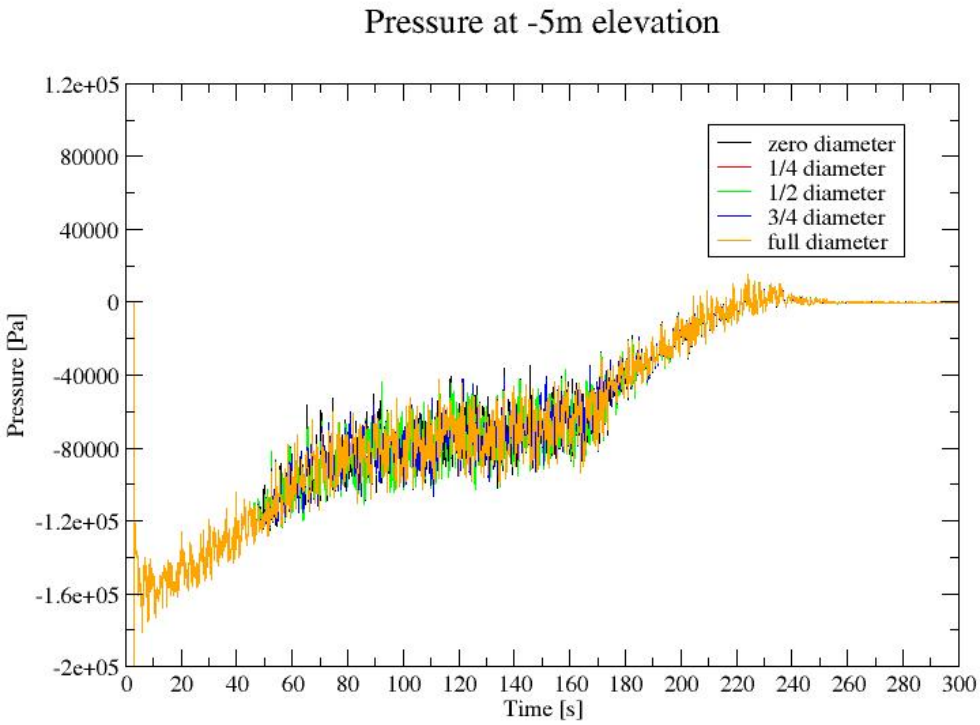

Figure 4.40 Pressure as function of time. Case d18u.

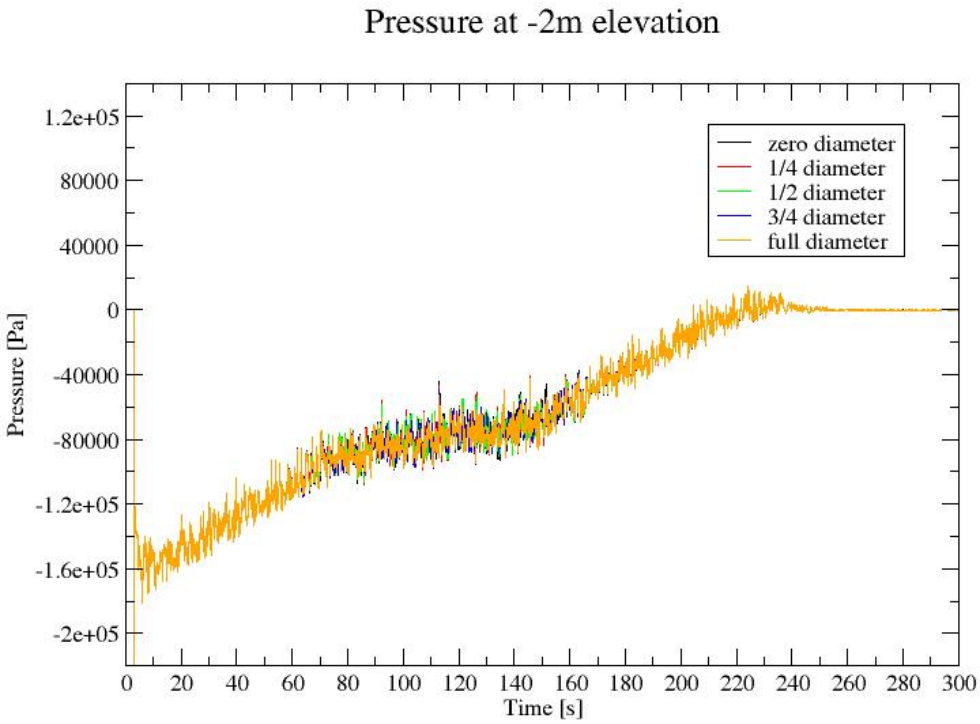

Figure 4.41 Pressure as function of time. Case d18u.

The pressure drop is vertical due to the sudden shut down. The volume flow rate of water does not experience a decrease over time such as by use of a vent. However, such a vertical pressure drop is not realistic, because the water volume flow rate of the pump must decrease over a finite time period as it shuts down. Only at the outlet pressure is reduced to a minimum of -1.5bar, Figure 4.42. This is probably due to air which is entrained at the moment the pump shuts down. View the animation of alpha1 in APPENDIX F. In the animation, air is entrained and rises up the pipe in order to empty the pipeline. Suction is produced in the whole pipeline which cannot be relieved by an air flow upstream in the pipe such as in the vented cases. Therefore air must be entrained at the outlet and rise up the pipeline. From Figure 4.37, Figure 4.38, Figure 4.39, Figure 4.40 and Figure 4.41 one observes that pressure is considerably more fluctuating at low elevations of -10m, -15m and -19m than at -5m and -2m. Pressure may be more disturbed at lower elevations due to the fact that air enters the same cross section as water leaves. The fluctuations occur first at elevations close to the outlet since it takes some time before air reaches the top of the pipeline. Animations of alpha1 and p_rgh can be viewed in APPENDIX F. An elevation of about -5m is not reached by air until at about 60s, and from Figure 4.40 this is the time at which fluctuations starts.
Volume flow rate and mass flow rate differs from the vented cases, as shown in Figure 4.43 and Figure 4.44. Water volume flow rate reduces to zero at the water_inlet as the pump is shut down at t=3s. Thereafter volume flow rate at the outlet is seen to fluctuate between about -0.015m^3/s and 0.03m^3/s. Thus the volume flow rate into the pipeline seems to be higher than that flowing out. This may be reasonable since pressure must be relived to an atmospheric pressure. Figure 4.44 shows a negative mass flow rate at the outlet. Even if air flows in at a higher volume flow rate, water is thousand times denser causing the mass flow rate out to be greatest.

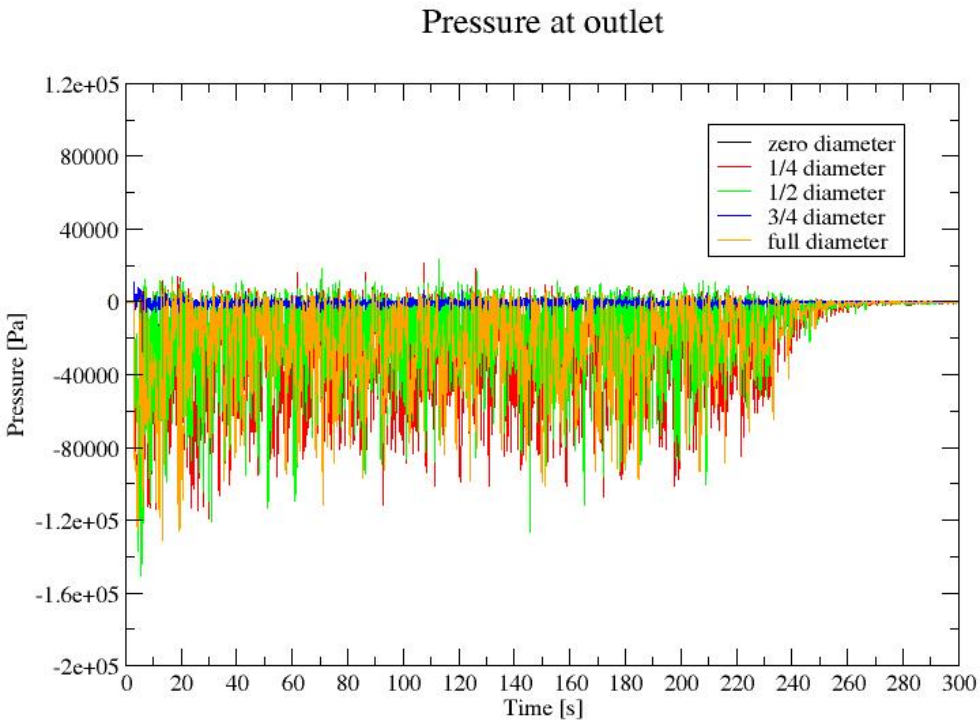

Figure 4.42 Pressure as function of time. Case d18u.

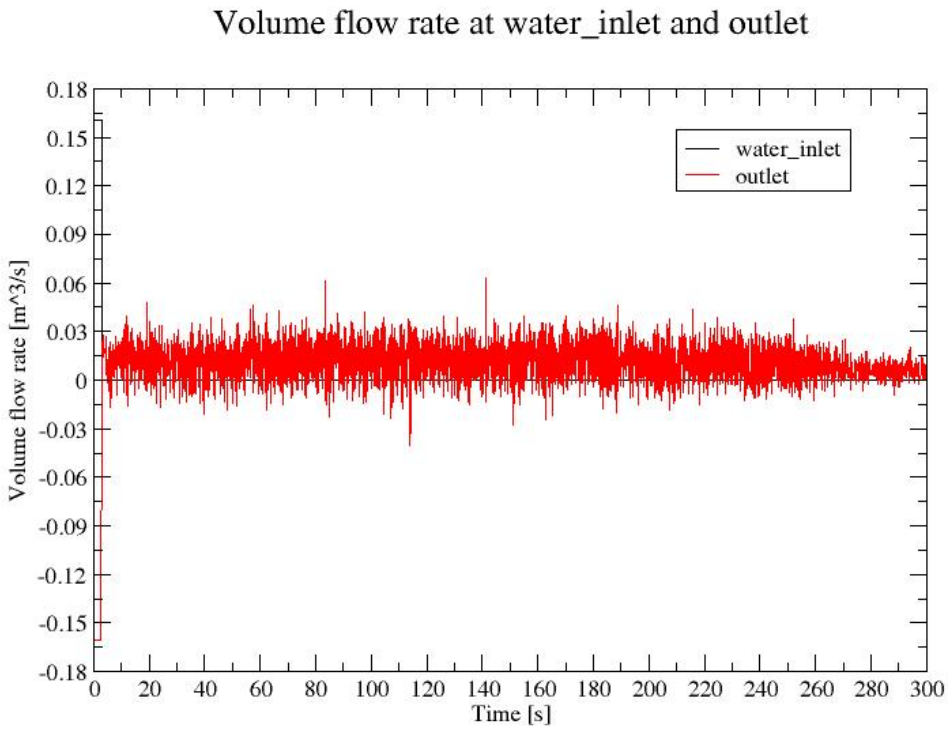

Figure 4.43 Volume flow rate as function of time. Case d18u.

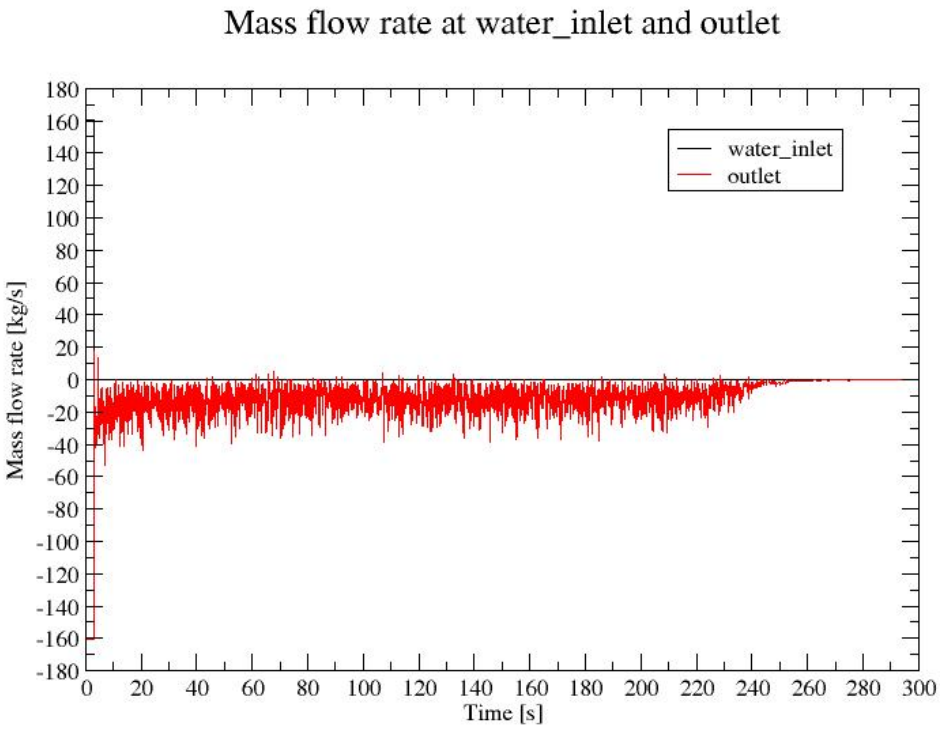

Figure 4.44 Mass flow rate as function of time. Case d18u.

Superficial water velocity decreases to zero at the outlet when the pump shuts down, Figure 4.45. Water must exist at the outlet at later times, but the volume fraction may be less than 0.5 causing the sampled superficial water velocity to equal zero. The superficial water velocity is fluctuating and decreases to zero at all other elevations. First it decreases to zero at -2m elevation, and proceeds further down the pipe. The animation of alpha1 shows how pure air is recovered in the top of the pipeline.
The unvented pipe is subject to the highest Froude number at the outlet, Figure 4.46. The time averaged Froude number is 1.71. High Froude number is caused by the superficial air velocity observed at the outlet, Figure 4.47. The lack of a vent causes suction of air into the pipeline. It is likely the cause for high superficial air velocities. As previously explained, the Froude number should be a measure of the liquid velocity, and not the air velocity. The sampled Froude number may therefore be misleading. A drawback with the interFoam model is the fact that only one velocity field is present. Thus it is not possible to sufficiently distinguish between velocity of water and air. However there exists a solver named `compressibleTwoPhaseEulerFoam` which provides two velocity fields, one for each phase. In accordance to the OpenFOAM Foundation (2011d) this model accounts for heat transfer and solves for bubbles dispersed in another phase. Another model may be the `twoPhaseEulerFoam`. According to the OpenFOAM Foundation (2011h) it is a solver for two incompressible fluids. One phase may be dispersed in the other such as for bubbles in a liquid or particles in a gas. Also here velocity fields exist for each phase.

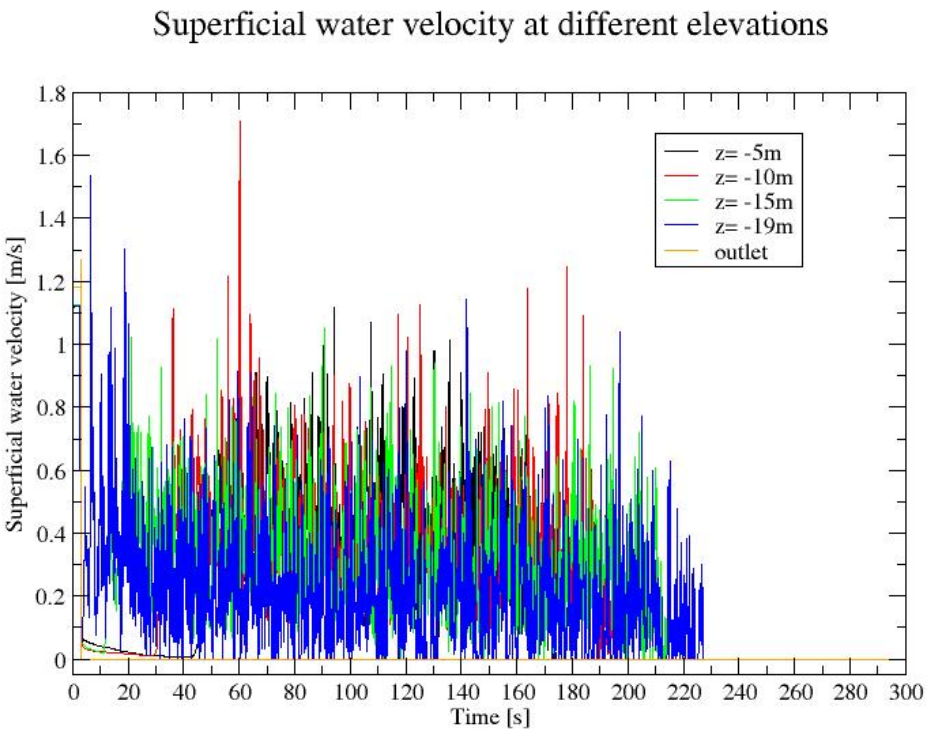

Figure 4.45 Superficial water velocity as function of time. Case d18u.

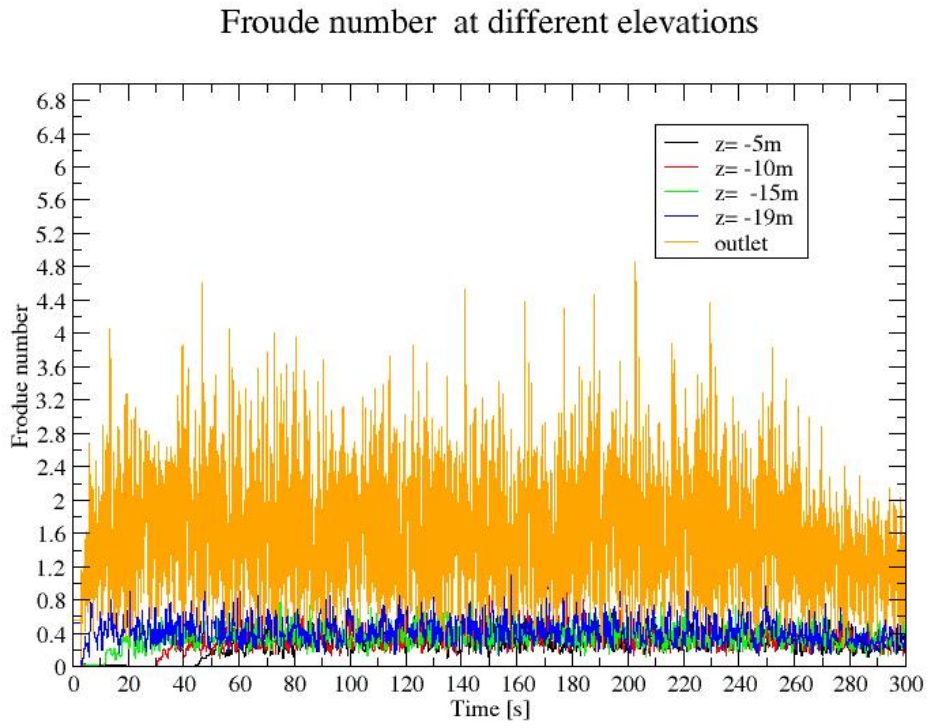

Figure 4.46 Froude number as function of time. Case d18u.

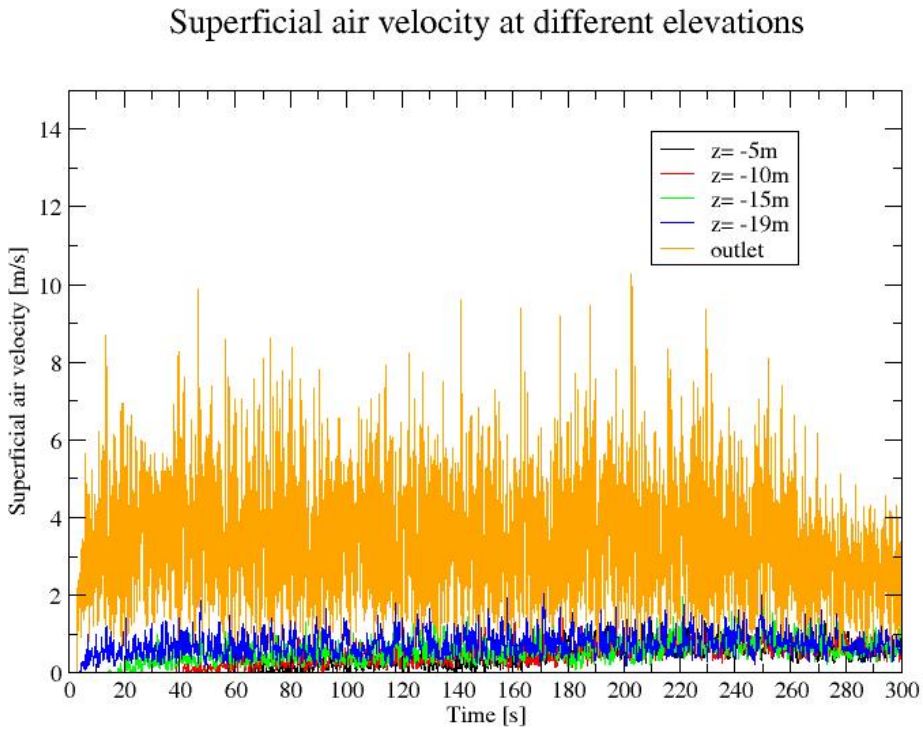

Figure 4.47 Superficial air velocity as function of time. Case d18u.

4.7 Summary of results

In Table 4.1 the main results for the different cases are summarized. Note that the time required to empty the pipeline is equal for all the vented cases subjected to a 4" vent. The d18v2 requires an additional minute to drain. In addition all vented cases decreases to an equal minimum pressure at -19m elevation. The flow rate in and out of the domain at the first slow valve opening is approximately equal for the 4" vented cases. Air flows in at about $0.025m^3/s$ while water flows out at $-0.12m^3/s$. Remember that outflow of the domain is defined as negative. In the d18v2 the induced volume flow rate is reduced to $0.018m^3/s$ during the same valve opening. At the last and longest slow valve opening, flow rate appears to adapt to the total water volume in the pipeline. In the d18v4 case air flows in at about $0.035m^3/s$ and water flows out at about $-0.18m^3/s$. A smaller diameter of 12" produces an air flow rate of $0.01m^3/s$ and water outflow of $-0.06m^3/s$. Volume flow rates are adjusted to higher values for the 24" main diameter. Air flows in at $0.05m^3/s$ and water flows out at $-0.3m^3/s$. This may be related to the pressure drop close to the air_inlet which is -0.01, -0.05 and -0.07bar for respectively d12v4, d18v4 and d24v4 at the last slow valve opening. One should remember that pressure drop is the driving force for air to be induced at the air_inlet. In the d18v2 case air volume flow rate at the last and longest valve opening is reduced to $0.012m^3/s$ compared to $0.035m^3/s$ in the d18v4 case. Water volume flow rate is therefore also reduced to $-0.075m^3/s$ compared to $0.18m^3/s$ in the d18v4 case. Another interesting observation is the difference in elevation of the water column at which distorted air bubbles are entrained at the outlet. The ease at which distorted bubbles are formed appears to decrease as the pipe diameter increases. Considerably air entrainment occurs at a water column height of 5.1, 4.3 and 3.8m for respectively d12v4, d18v4 and d24v4. In the d18v2 air entrainment occurs at a water column height of 4.9m, which is approximately equal to what is observed for the d18v4 case. Minimum pressure is reached at approximately the same time

as the distorted air bubbles are entrained. The ease of bubble entrainment may also be affected by the higher flow rate which occurs at increased pipe diameters.

Table 4.1 Summary of results

	d18v4	d12v4	d24v4	d18v2	d18u
Total pipeline volume [m^3]	3.55	1.56	6.42	3.69	3.56
Emptying time [s]	120	120	120	180	300
Minimum linear pressure decrease [bar]	-1.6	-1.6	-1.6	-1.6	-1.7
z- location of minimum pressure decrease [m]	-19	-19	-19	-19	All probe points
Time at which minimum pressure occurs [s]	33	39	54	125	3
Start time of linear pressure decrease [s]	13	5	13	67	3
Time when water level decrease occurs [s]	8	3	13	60	-
Time averaged Froude number at outlet	1.08	1.05	0.67	0.70	1.71
Time when considerably air entrainment occurs [s]	35	38	60	118	3
Water volume at air entrainment [m3]	0.7	0.37	1.1	0.8	-
Height of water column at air entrainment [m]	4.3	5.1	3.8	4.9	-
Number of slow valve opening prior to air entrainment	4	2	2	12	-
Time intervals of slow valve opening [s]	3-8 8-11 11-13 13-35	3-8 8-32	3-13 13-35	3-22 22-30 30-37 37-42 42-50 50-53 53-56 56-60 60-62 62-66 67-68 68-120	-
Max air volume flow rate at 1st opening interval [m^3/s]	0.025	0.025	0.025	0.018	-
Max water volume flow rate at 1st opening interval [m^3/s]	-0.12	-0.12	-0.12	-0.12	-
Max air volume flow rate at the last and longest opening interval [m^3/s]	0.035	0.01	0.05	0.012	-
Max water volume flow rate at the last and longest opening interval [m^3/s]	-0.18	-0.06	-0.3	-0.075	-
Minimum pressure at the air_inlet during slow valve opening intervals [bar]	-0.04 -0.06 -0.07 -0.05	-0.02 -0.01	-0.01 -0.07	-0.02 -0.01 -0.01 -0.01 -0.01 -0.01 -0.02 -0.01 -0.01 -0.01 -0.08 -0.08	-

The lowest time averaged Froude number occurs for d24v4 and d18v2. What characterizes these cases is that air entrainment and minimum pressure drops are considerably delayed. In the d24v4 it occurs after about a minute, while it occurs after two minutes in the d18v2 case. In addition the valve stays open for almost a minute in the d18v2 case. The time period for the last valve opening in the d24v4 does not differ from previous cases, but the avoidance of early air entrainment causes the valve to open and close over larger time intervals afterwards. Five minutes is required to empty the unvented pipe, and pressure is also extended to a minimum of -1.7bar. The unvented pipe differs because pressure is reduced to its minimum at all probe points in the domain. Pressure decreases to its minimum immediately when the pump shuts down. The change in velocity is at its maximum since there is no valve to maintain a flow rate. Thus flow retardation is at its maximum, and a pressure wave is propagated. No air enters upstream to relieve the propagated pressure drop.

4.8 Residuals

The following subsection is taken from Hjertager (2009). In any CFD solution convergence is important. Results of an unconverged solution may be misleading. However, if the solution is converged it may still give unphysical solutions which may be caused by an inadequate physical model, or considerably large discretization errors. Results are expected to be accurate for each time step in order for convergence to be achieved in a transient solution. Important outputs of a transient solution are the mean and max Courant number which indicates if good time discretization is achieved. The Courant number shall stay below a value of 1. Solutions are more stable as the Courant number decreases. In all cases `adjustableRunTime` is used with a limit of 0.5 on both `maxAlphaCo` and `maxCo`. Recall that the PIMPLE method solves all equations in the outer loop and only the continuity equation in the inner loop. Convergence of the PIMPLE solution is achieved if initial residuals and the continuity error of the last loop stay small in addition to the criterion of a small Courant number.
In the dictionary `fvSolution` the keywords `tolerance` and `relTol` are set for pcorr, p_rgh, p_rghFinal and U,see APPENDIX D. In accordance to Hjertager (2009) the keyword `tolerance` defines the accuracy of the solution. The specific equation is solved until the initial residual falls below the `tolerance`. The relative tolerance determines the accuracy of solving an equation within an iteration loop. The solution must converge to the tolerance if the relative tolerance is set to zero. In the review of residuals the typical range in which they varies are mentioned. Average values are not presented since results are expected to be accurate for each time step in a transient solution. Residuals may be examined in more detail in APPENDIX F.
The Courant number is of great importance. The maximum Courant number of the outer loop generally stays below the limited value of 0.5, Figure 4.48. Some higher values are observed in the inner loop, and a considerably peak is observed at t=3s as the pump is shut down. It is the outer loop which is of importance. All equations are solved in the outer loop, while only the continuity equation is solved in the inner loop. A significantly lower range applies to the mean Courant number. It generally varies between 0 and 0.08. The initial pressure residuals of the outer loop mainly consist of values in the of order of 0.001, Figure 4.49. The `relTol` is set to 0.05 for the p_rgh residual and the `tolerance` is set to 1e-07. Relative tolerance is defined as the ratio of current to initial residual in accordance to the OpenFOAM Foundation (2011e). The initial pressure residual is not expected to converge to the tolerance since the relative tolerance is non-zero. The tolerance of the final pressure residual seems to be reached at the outer loop. The final pressure residual of the outer loop remains below 1e-07, Figure 4.50. The tolerance and relative tolerance is set to respectively 1e-07 and 0. Thus solution is forced to converge to the tolerance in each time step. Number of iterations is

highest at the outer loop and range between 25 and 320. It must increase to reach the tolerance as the relative tolerance is set to zero. Global and local continuity error remains small during the simulation time. They range between 0 and 1e-05 for the outer loop. The cumulative continuity error is observed to vary between –0.0001 to 0.006. Solutions are considered sufficiently converged according to the briefly explained convergence criterions for the PIMPLE pressure coupling.

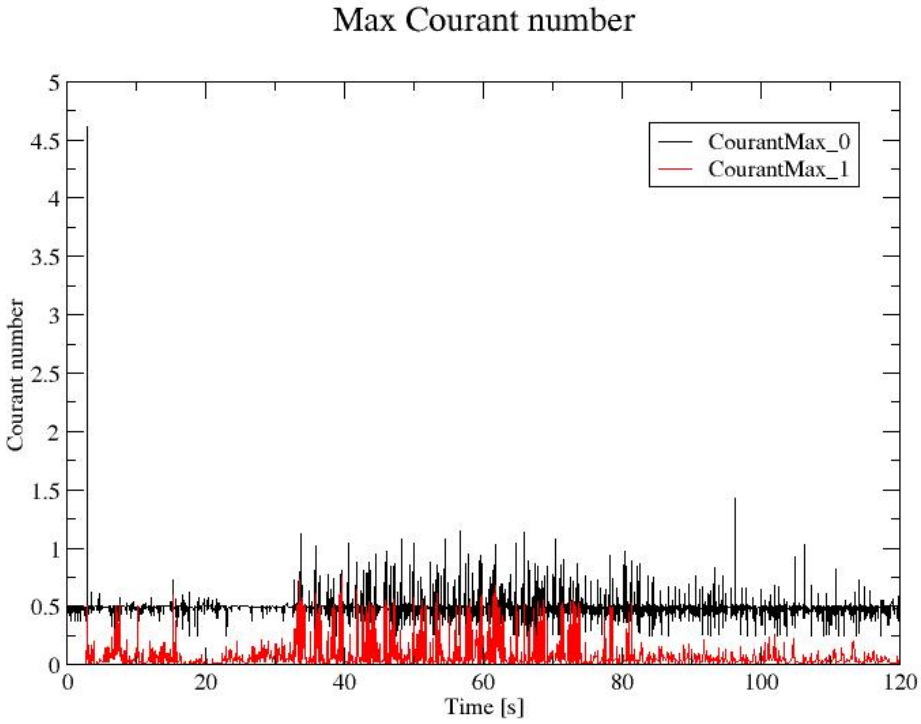

Figure 4.48 Courant number as function of time. Max Courant number of d18v4.

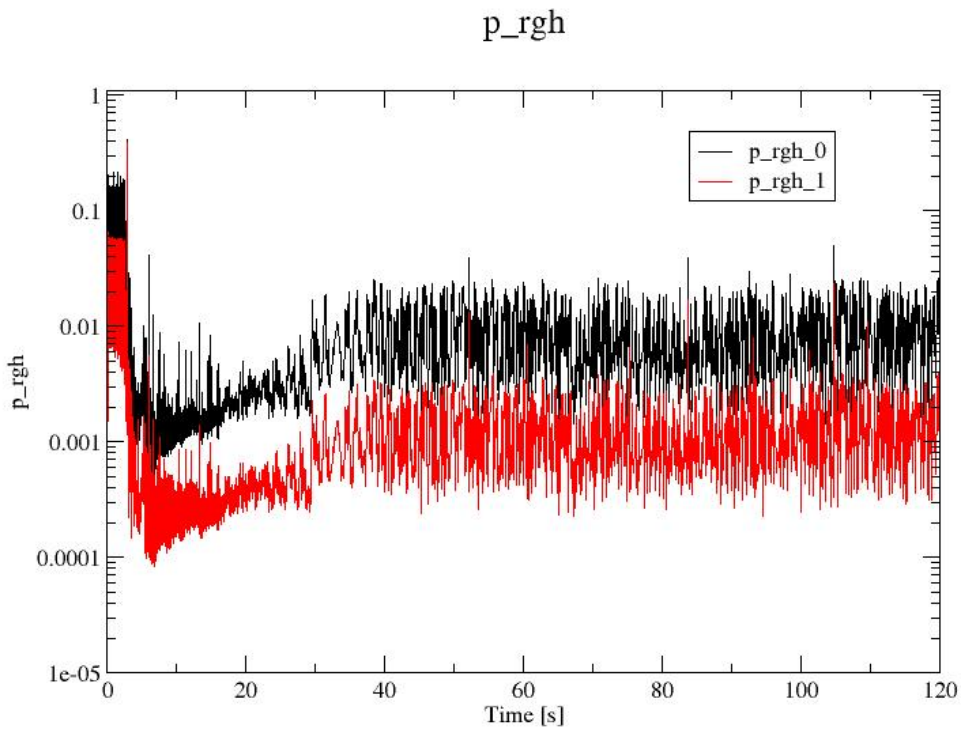

Figure 4.49 p_rgh residual as function of time. Typically p_rgh residuals, d12v4.

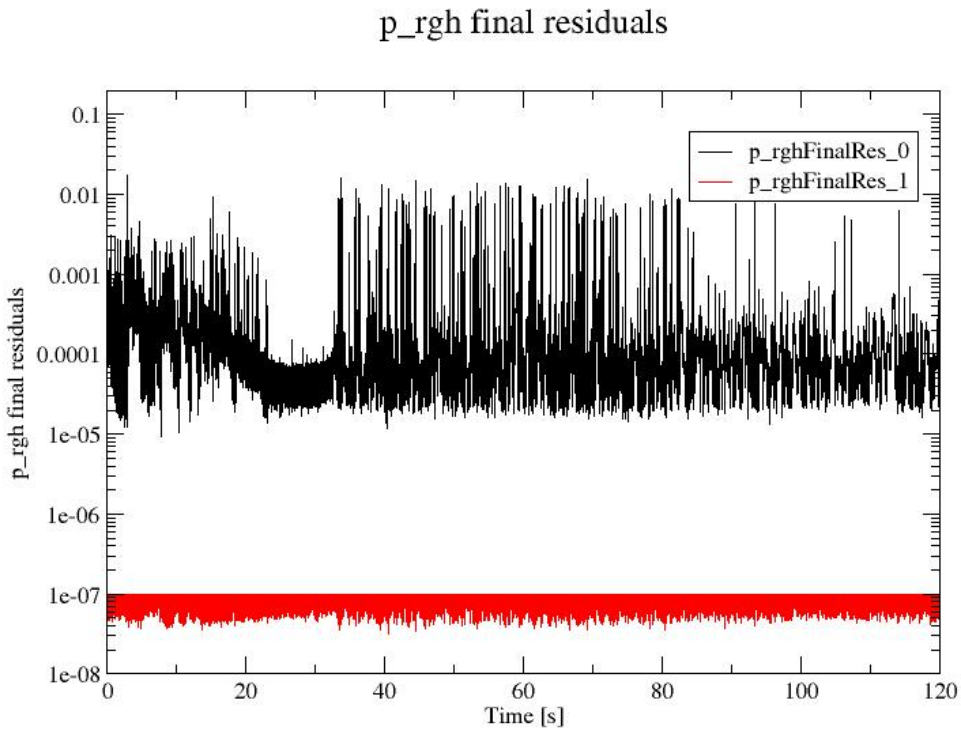

Figure 4.50 Final pressure residual as function of time. Results from d18v4.

5 CONCLUSIONS

Vented vertical pipes subjected to a 4" vent drains during equal time intervals, even if total water volume differs for the different diameters of 12", 18" and 24". The d18v2 requires an additional minute to drain. In the pipeline of largest diameter, a greater volume of water is delayed as the pump suddenly shuts down. The pressure drop in front of the valve is slightly increased, which again increases the volume flow rate of induced air and thus the volume flow rate at the outlet. Similarly the volume flow rate of the 12" pipeline decreases due to a smaller pressure drop as a decreased volume is delayed during pump shut down. In the unvented pipeline air enters the outlet and rises up the pipe to relieve pressure and drain the pipe. The lack of a vent considerably increases the time required to drain the pipeline.
All cases are characterized by a linear pressure decrease subsequent to the pump shut down. In accordance to transient theory pressure propagation can be described by the sudden pump shut down. Pressure is decreased close to the valve due to the flow retardation. This causes a negative pressure wave to propagate downstream, resulting in a delay of fluid flow. The negative pressure wave causes the pipe walls to contract. If pressure is sufficiently decreased vapor bubbles may form, which is also known as flashing. In all vented cases pressure drops to its minimum close to the outlet, and the pressure drop decreases in size at locations further up the pipeline. This is due to the induced air from the valve which retains an atmospheric pressure at higher elevations as the top of the pipeline is filled with air. The pressure propagation is known as water hammer which causes the pipe to vibrate and is characterized by the sound of a pipe being hammered. It is the change in velocity which determines the decrease in pressure. In the unvented pipe the water velocity suddenly turns to zero as there is no vent to maintain a flow rate. Thus pressure decreases instantaneously to its minimum at all locations in the pipeline.
The Froude number criterion mentioned in the standard NORSOK P-001 (2006) appears to apply for draining from a process vessel. The theory presented for slug flow due to a rising Taylor bubble does not match the criterion since the bubble will rise up the column as the Froude number is reduced below 0.3. In a process vessel a Froude number less than 0.3 is recommended to ensure the vertical drain pipe is running full without air being entrained through a weir formation at the top surface. However, this theory assumes a considerably wide process vessel. Transient theory seems to apply greater knowledge to future design, than does the Froude number. Flow control should be an objective, where the aim is to reduce rapid changes in flow velocity. This can be managed by slow closure of the pump in addition to slow opening of the valve. A slow valve opening may be fully opened or partially opened, but stays open for a time period longer than the pipe period. The pipe period is the time for a pressure wave to propagate down the pipe and send a restoring pressure wave into the pipe and back to the pump location. Water hammer may be the phenomenon which causes vibrations during shut down, rather than oscillations due to air entrainment. It must however be stated that solutions are obtained with very coarse meshes, which affects the accuracy of the results. Results are however assumed to indicate how the flow regime transforms and how it is affected by boundary conditions. The pump shut down in the simulation is also too rapid. In reality the flow must decrease during a finite time period.

6 FUTURE WORK

The interface between water and air is never sharply defined in interFoam. Rather, the interface occupies a volume around the region where this sharp interface should exist. In addition the approach simulates bubbles larger than the grid size. The interFoam model limits the ability to sample the velocity of water only. Since only one velocity field is present, a volume fraction larger than 0.5 is applied to distinguish water velocity from the velocity of air. In future studies one should consider adopting either the `twoPhaseEulerFoam` or the `compressibleTwoPhaseEulerFoam`. These models contain two velocity fields, one for each phase. It simplifies and ensures a more accurate sampling of water velocity, especially at the outlet where the volume fraction of air is significant. The mentioned models may ensure that air velocity does not affect the calculation of the Froude number. In addition these models enable modeling of smaller bubble formation.
It may be desirable to refine the mesh in future studies. The current mesh size range between 2000 and 5000 cells, and the execution time range between four hours and three days. Refinement requires smaller time steps and increases the computational effort due to the increased number of cells. Significantly increased processor capacity is needed to refine the mesh without a considerably increase in the calculation time. It may be advantageous to run a case in parallel if the mesh size is increased sufficiently.
Considerably effort and time are spent on the valve as a boundary condition. Insufficient time and selected priority results in a laminar model. However, it is assumed that turbulence will occur due to high velocities of air. It is certainly a potential for developing a turbulent model if cases are to be further investigated. Additional layouts of the draining pipe may also be an option. The d18u and d18v2 may be subject to different main pipe diameters. Other interesting setups may be a submerged outlet, the influence of waves on a submerged outlet and the behavior of a vertical vent pipe free to the atmosphere without a valve boundary condition.

7 REFERENCES

ANSYS, I. 2009. *7.3.6 Inlet vent boundary conditions* [Online]. Available: https://www.sharcnet.ca/Software/Fluent12/html/ug/node242.htm [Accessed 24.02 2012].

BARAJAS, A. M. & PANTON, R. L. 1992. The effects of contact angle on two phase flow in capillary tubes. *International Journal of Multiphase Flow,* 19**,** p. 337-346.

BGSCHAID. 2009. *Contrib groovyBC OpenFOAM wiki* [Online]. Available: http://openfoamwiki.net/index.php/Contrib_groovyBC [Accessed 08.02 2012].

BGSCHAID. 2010. *Contrib/swak4Foam* [Online]. Available: http://openfoamwiki.net/index.php/Contrib/swak4Foam [Accessed 10.02 2012].

CAE/DEN, EDF R&D & OPEN CASCADE. 2007. *Salome platform documentation* [Online]. Available: http://docs.salome-platform.org/salome_5_1_6/smesh/user/netgen_2d_3d_hypo_page.html [Accessed 17.03.2012 2012].

ÇENGEL, Y. A. & CIMBALA, J. M. 2006. *Fluid mechanics: fundamentals and applications,* Boston, McGraw-Hill, p.43,680-685.

CETINBUDAKLAR, A. G. & JAMESON, G. J. 1969. The mechanism of flooding in vertical countercurrent two-phase flow. *Chemical Engineering Science,* 24**,** p. 1669-1680.

CHUNG, T. J. 2002. *Computational fluid dynamics,* Cambridge, Cambridge University Press, p.902-913.

DOUGLAS, J. F., GASIOREK, J. M., SWAFFIELD, J. A. & JACK, L. B. 2005. *Fluid Mechanics,* Harlow, Person Prentience Hall, p.146-148,510,531-532,674-682,695-696.

FABRE, J. & LINE, A. 2010. *Slug flow* [Online]. Available: http://www.thermopedia.com/content/38/?tid=104&sn=1297 [Accessed 25.04 2012].

FARLEX, I. 2004. *The free dictionary* [Online]. Huntingdon Valley. Available: http://www.thefreedictionary.com/siphoning [Accessed 13.01 11:55 2012].

FINNEMORE, E. J., FRANZINI, J. B. & DAUGHERTY, R. L. 2002. *Fluid mechanics with engineering applications,* Boston, McGraw-Hill, p.32,139,237,407,481,558-569.

GALDAMEZ, R. G., WOOD, S. & GOKALTUN, S. 2011. *Numerical Simulations of Pulsed-Air Mixing Technology using Multiphase Computational Fluid Dynamics Methods* [Online]. Miami. Available: http://fellows.fiu.edu/doc/Fellows/WM_2011_11301_final.pdf [Accessed 08.03 2012].

GOVIER, G. W. & AZIZ, K. 2008. *The flow of complex mixtures in pipes,* Richardson, Tex., Society of Petroleum Engineers, p.306,400-401.

GRACE TEAM. 2008. *Grace's user guide* [Online]. Available: http://plasma-gate.weizmann.ac.il/Grace/doc/UsersGuide.html#ss1.1 [Accessed 13.05 2012].

GSCHAIDER, B. 2010. *HG changeset patch* [Online]. Available: http://openfoam-extend.hg.sourceforge.net/hgweb/openfoam-extend/swak4Foam/raw-rev/f10e84154489 [Accessed 24.03 2012].

GSCHAIDER, B. F. W. 2011. *Swak4Foam-History and new features: Beyond funkySetFields and groovyBC* [Online]. Available: http://openfoamwiki.net/images/d/da/Swak4FoamLeoben2011.pdf [Accessed 09.02 2012].

HJERTAGER, B. H. 2009. Lecture notes in OpenFOAM. Stavanger: University of Stavanger, p.3-12,37-46,57-62,70-85. .

HJERTAGER, B. H. 20.02. 2012a. *RE: Discussion of valve boundary condition.* Type to TØGE, G. E.

HJERTAGER, B. H. 28.02. 2012b. *RE: Running in parallel.* Type to TØGE, G. E.

LUCCHINI, T. 2009. *OpenFOAM postprocessing and advanced running options* [Online]. Available: http://www.openfoamworkshop.org/2009/4th_Workshop/basicTraining/ofPostProcessing.pdf [Accessed 13.05 2012].

MAKI, K. 2011. *Ship resistance simulations with OpenFOAM: 6th OpenFOAM Workshop* [Online]. Pensylvania Available: http://www.openfoamworkshop.org/6th_OpenFOAM_Workshop_2011/Program/Training/maki_slides.pdf [Accessed 07.03 2012].
MCDUFFY, N. G. 1977. Vortex free downflow in vertical drains. *AIChE Journal* 23**,** p. 37-40.
MOON, R. K., MOORE, R. M., THORPE, R. B. & VERGHESE, T. M. 1987. Two phase downfow. *Proceedings of the 3rd international conference on multiphase flow***,** p. 365-382.
MYCHEME, C. E. P. 2011. *Designing for Gravity Flow* [Online]. Available: http://www.mycheme.com/designguides/gravity-flow.html [Accessed 02.01 2012].
NABLA LTD. 2007. *FOAM Programmer's guide* [Online]. Available: http://www.foamcfd.org/Nabla/guides/ProgrammersGuidese11.html [Accessed 26.03 2012].
NOLTE, C. B. 1978. *Optimum pipe size selection,* Clausthal, Trans Tech Publications, p.55-66.
NORSOK P-001 2006. Process Design. *Section 6.2.5 Liquid Flowing By Gravity.* Lysaker, Norway: Standards Norway, p.10-11.
OPENFOAM FOUNDATION. 2011a. *5.2 Boundaries* [Online]. Available: http://www.openfoam.org/docs/user/boundaries.php [Accessed 26.03 2012].
OPENFOAM FOUNDATION. 2011b. *alphaContactAngleFvPatchScalarField.H* [Online]. Available: http://foam.sourceforge.net/docs/cpp/a08398_source.html [Accessed 25.03 2012].
OPENFOAM FOUNDATION. 2011c. *Breaking of a dam* [Online]. Available: http://www.openfoam.org/docs/user/damBreak.php [Accessed 08.03 2012].
OPENFOAM FOUNDATION. 2011d. *compressibleTwoPhaseEulerFoam.C* [Online]. Available: http://foam.sourceforge.net/docs/cpp/a02728_source.html [Accessed 07.05 2012].
OPENFOAM FOUNDATION 2011e. Open Foam: The Open Source CFD Toolbox User Guide. 2.1.0 ed. Boston, p.23-25,33,43, 57-68,104,111-129,174-178.
OPENFOAM FOUNDATION. 2011f. *OpenFOAM C++ Documentation: interfoam.C* [Online]. Available: http://foam.sourceforge.net/docs/cpp/a02958_source.html [Accessed 08.03 2012].
OPENFOAM FOUNDATION. 2011g. *Turbulence models* [Online]. Available: http://www.openfoam.org/docs/user/turbulence.php [Accessed 17.03 2012].
OPENFOAM FOUNDATION. 2011h. *twoPhaseEulerFoam.C* [Online]. Available: http://foam.sourceforge.net/docs/cpp/a03001_source.html [Accessed 07.05 2012].
OSENBROCH, J. 29.11 2011. *RE: Kick off meeting.* Type to TØGE, G. E. & HJERTAGER, B. H.
PASSALACQUA, A. 2010. *twoPhaseEulerPimpleFoam* [Online]. Available: http://albertopassalacqua.com/?p=772 [Accessed 12.03 2012].
PRUSINSKI, P., SMITH, K. & GSCHAIDER, B. 2010. *Bounaries and patches* [Online]. Available: http://www.cfd-online.com/Forums/openfoam/74043-boundaries-patches.html [Accessed 12.02 2012].
ROCHELLE, S. G. & BRISCOE, M. T. J. 2010. *Predict and prevent air entrainment in draining tanks. Chemical Engineering, p.37-43* [Online]. Available: http://accessintelligence.imirus.com/Mpowered/book/vche1/i9/p57 [Accessed 19.04 2012].
RUSCHE, H. 2002. *Computational fluid dynamics of dispersed two-phase flows at high phase fractions. Phd, p. 147-158* [Online]. London. Available: http://powerlab.fsb.hr/ped/kturbo/openfoam/docs/HenrikRuschePhD2002.pdf [Accessed 09.03 2012].
SIMPSON, L. L. 1968. Sizing piping for process plants. *Chemical engineering***,** p. 192-214.
SPRINGER, J., KISSLING, K., JASAK, H., URBAN, K., SCHUTZ, S. & PIESCHE. 2010. *A pressure based solution algorithm based on the volume-of-fluid approach for two or more immiscible fluids* [Online]. Ulm. Available:

http://web.student.chalmers.se/groups/ofw5/Presentations/JuliaSpringerSlidesOFW5.pdf [Accessed 08.03 2012].
THORPE, R. B., BERESFORD-PEIRSE, J. D., GIANCOLA, F. L., MCVAY, J. & TABBERER, R. J. 1989. Two-phase downflow. *Multi-phase flow: proceedings of the 4th international conference, Nice, France, 19-21 June 1989*, p. 397-418.
VALLIER, A. 2011. *Coupling of VOF with LPT in OpenFOAM* [Online]. Lund University Sweden. Available: http://www.tfd.chalmers.se/~hani/kurser/OS_CFD/OF_kurs_LPT_120911.pdf [Accessed 28.02 2012].

APPENDIX A

A.1 Boundary conditions

```
/*--------------------------------*- C++ -*----------------------*\
| =========                 |
|
| \\      /  F ield         | OpenFOAM: The Open Source CFD Toolbox
|
|  \\    /   O peration     | Version:  2.1.x
|
|   \\  /    A nd           | Web:      www.OpenFOAM.org
|
|    \\/     M anipulation  |
|
\*---------------------------------------------------------------------*/
FoamFile
{
    version     2.0;
    format      ascii;
    class       volVectorField;
    location    "0";
    object      U;
}
// * * * * * * * * * * * * * * * * * * * * * * ** * * * * * * * * * * * * //

dimensions      [0 1 -1 0 0 0 0];

dimensions      [0 1 -1 0 0 0 0];

internalField   uniform (0 0 0);


boundaryField
{
    water_inlet
    {
      type            groovyBC;
      valueExpression "(time()< 3?vector(1.184,0,0):vector(0,0,0)";
      value           uniform (1.184 0 0);
    }

    air_inlet
    {
      type groovyBC;
      fractionExpression  "(min(phi)< 0 && average(p) < 0) ? 0 : 1";
      valueExpression     "vector(0,0,0)"; // (1) Dirichlet
      gradientExpression  "vector(0,0,0)"; // (0) Neumann
      value               uniform (0 0 0);
    }
```

```
    walls
    {
        type            fixedValue;
        value           uniform (0 0 0);
    }

    water_wall
    {
        type            fixedValue;
        value           uniform (0 0 0);
    }

    air_wall
    {
        type            fixedValue;
        value           uniform (0 0 0);
    }

    outlet
    {
         type            pressureInletOutletVelocity;
         phi             phi;
         rho             rho;
         value           uniform (0 0 0);
    }
}


// ************************************************************************* //
```

```
/*--------------------------------*- C++ -*----------------------*\
| =========                 |
|
| \\      /  F ield         | OpenFOAM: The Open Source CFD Toolbox
|
|  \\    /   O peration     | Version:  2.1.x
|
|   \\  /    A nd           | Web:      www.OpenFOAM.org
|
|    \\/     M anipulation  |
|
\*---------------------------------------------------------------------*/
FoamFile
{
    version     2.0;
    format      ascii;
    class       volScalarField;
    object      p_rgh;
}
// * * * * * * * * * * * * * * * * * *  * * * * * * * * * * * * * * //

dimensions      [1 -1 -2 0 0 0 0];

internalField   uniform 0;

boundaryField
{
    water_inlet
    {
           type            zeroGradient;
    }

    air_inlet
    {
           type            totalPressure;
           p0              uniform 0;
           U               U;
           phi             phi;
           rho             rho;
           psi             none;
           gamma           1;
           value           uniform 0;
    }

    water_wall
    {
           type            zeroGradient;
    }

    air_wall
    {
           type            zeroGradient;
    }
```

```
    walls
    {
            type            zeroGradient;
    }

    outlet
    {
            type            fixedValue;
            value           uniform 0;
    }
  }

// ************************************************************************* //
```

```
/*--------------------------------*- C++ -*----------------------*\
| =========                 |
|
| \\      /  F ield         | OpenFOAM: The Open Source CFD Toolbox
|
|  \\    /   O peration     | Version:  2.1.x
|
|   \\  /    A nd           | Web:      www.OpenFOAM.org
|
|    \\/     M anipulation  |
|
\*---------------------------------------------------------------------*/
FoamFile
{
    version     2.0;
    format      ascii;
    class       volScalarField;
    location    "0";
    object      alpha1;
}
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //

dimensions      [0 0 0 0 0 0 0];

internalField   uniform 1;

boundaryField
{
    water_inlet
    {
           type            inletOutlet;
           inletValue      uniform 1;
           value           uniform 1;
    }

    air_inlet
    {
           type            inletOutlet;
           inletValue      uniform 0;
           value           uniform 1;
    }

    walls
    {
           type            zeroGradient;
    }

    water_wall
    {
           type            zeroGradient;
    }
```

```
    air_wall
    {
            type            zeroGradient;
    }

    outlet
    {
            type            inletOutlet;
            InletValue      uniform 0;
            value           uniform 1;
    }
}


// ************************************************************************* //
```

APPENDIX B

B.1 New velocity boundary condition at the air_inlet

```
/*--------------------------------*- C++ -*----------------------*\
| =========                 |
|
| \\      /  F ield         | OpenFOAM: The Open Source CFD Toolbox
|
|  \\    /   O peration     | Version:  2.1.x
|
|   \\  /    A nd           | Web:      www.OpenFOAM.org
|
|    \\/     M anipulation  |
|
\*---------------------------------------------------------------*/
FoamFile
{
    version     2.0;
    format      ascii;
    class       volVectorField;
    location    "0";
    object      U;
}
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //

dimensions      [0 1 -1 0 0 0 0];

dimensions      [0 1 -1 0 0 0 0];

internalField   uniform (0 0 0);


boundaryField
{

water_inlet
{
 type groovyBC;
 valueExpression "(time() <3) ? vector(1.184,0,0):vecto(0,0,0)";
 value uniform (1.184 0 0);
}

air_inlet
{
 type groovyBC;
 variables "Z=mag(average(phi))/sum(mag(Sf()));";
 valueExpression "(max(p) <0) ? vector(0,0,-Z):vector(0,0,0)";
 value uniform (0 0 0);
}
```

```
walls
{
 type            fixedValue;
 value           uniform (0 0 0);
}

water_wall
{
 type            fixedValue;
 value           uniform (0 0 0);
}

air_wall
{
 type            fixedValue;
 value           uniform (0 0 0);
}

outlet
{
 type            pressureInletOutletVelocity;
 phi             phi;
 rho             rho;
 value           uniform (0 0 0);
}

}


// ************************************************************************* //
```

B.2 Slightly changed BC at air_inlet for d12v4 and d24v4

```
/*--------------------------------*- C++ -*----------------------*\
| =========                 |
|
| \\      /  F ield         | OpenFOAM: The Open Source CFD Toolbox
|
|  \\    /   O peration     | Version:  2.1.x
|
|   \\  /    A nd           | Web:      www.OpenFOAM.org
|
|    \\/     M anipulation  |
|
\*---------------------------------------------------------------------*/
FoamFile
{
    version     2.0;
    format      ascii;
    class       volVectorField;
    location    "0";
    object      U;
}
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //

dimensions      [0 1 -1 0 0 0 0];

dimensions      [0 1 -1 0 0 0 0];

internalField   uniform (0 0 0);


boundaryField
{
    water_inlet
    {
     type            groovyBC;
     valueExpression "(time()<3)? vector(2.665,0,0):vector(0,0,0)";
     value           uniform (2.665 0 0);
    }

     air_inlet
    {
     type            groovyBC;
     variables       "Z=mag(average(phi))/sum(mag(Sf()));";
     valueExpression "(max(p)<0&&time()>3)?vector(0,0,-Z)
                     :vector(0,0,0)";
     value           uniform (0 0 0);
    }
```

```
    walls
    {
           type            fixedValue;
           value           uniform (0 0 0);
    }

     water_wall
    {
           type            fixedValue;
           value           uniform (0 0 0);
    }

    air_wall
    {
           type            fixedValue;
           value           uniform (0 0 0);
    }

    outlet
    {
           type            pressureInletOutletVelocity;
           phi             phi;
           rho             rho;
           value           uniform (0 0 0);
    }
}


// ************************************************************************* //
```

```
/*--------------------------------*- C++ -*----------------------*\
| =========                 |
|
| \\      /  F ield         | OpenFOAM: The Open Source CFD Toolbox
|
|  \\    /   O peration     | Version:  2.1.x
|
|   \\  /    A nd           | Web:      www.OpenFOAM.org
|
|    \\/     M anipulation  |
|
\*---------------------------------------------------------------------*/
FoamFile
{
    version     2.0;
    format      ascii;
    class       volScalarField;
    location    "0";
    object      alpha1;
}
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //

dimensions      [0 0 0 0 0 0 0];

internalField   uniform 1;

boundaryField
{
    water_inlet
    {
            type            inletOutlet;
            inletValue      uniform 1;
            value           uniform 1;
    }

    air_inlet
    {
            type groovyBC;
            fractionExpression    "(max(p)<0 && time()>3)?1:0";
            valueExpression "0";
            gradientExpression "0";
            value      uniform 1;
    }

walls
    {
            type            zeroGradient;
    }
```

```
    water_wall
    {
        type            zeroGradient;
    }

    air_wall
    {
        type            zeroGradient;
    }

    outlet
    {
        type            inletOutlet;
        inletValue      uniform 0;
        value           uniform 1;
    }
}


// ************************************************************************* //
```

APPENDIX C

C.1 transportProperties

```
/*--------------------------------*- C++ -*----------------------*\
| =========                 |
|
| \\      /  F ield         | OpenFOAM: The Open Source CFD Toolbox
|
|  \\    /   O peration     | Version:  2.1.x
|
|   \\  /    A nd           | Web:      www.OpenFOAM.org
|
|    \\/     M anipulation  |
|
\*---------------------------------------------------------------*/
FoamFile
{
    version     2.0;
    format      ascii;
    class       dictionary;
    location    "constant";
    object      transportProperties;
}
// * * * * * * * * * * * * * * * * * * * * * * * * ** * * * * * * * * * //

phase1
{
    transportModel         Newtonian;
    nu                     nu [ 0 2 -1 0 0 0 0 ] 1e-06;
    rho                    rho [ 1 -3 0 0 0 0 0 ] 1000;
    CrossPowerLawCoeffs
    {
        nu0                nu0 [ 0 2 -1 0 0 0 0 ] 1e-06;
        nuInf              nuInf [ 0 2 -1 0 0 0 0 ] 1e-06;
        m                  m [ 0 0 1 0 0 0 0 ] 1;
        n                  n [ 0 0 0 0 0 0 0 ] 0;
    }

    BirdCarreauCoeffs
    {
        nu0                nu0 [ 0 2 -1 0 0 0 0 ] 0.0142515;
        nuInf              nuInf [ 0 2 -1 0 0 0 0 ] 1e-06;
        k                  k [ 0 0 1 0 0 0 0 ] 99.6;
        n                  n [ 0 0 0 0 0 0 0 ] 0.1003;
    }
}
```

```
phase2
{
    transportModel          Newtonian;
    nu                      nu [ 0 2 -1 0 0 0 0 ] 1.48e-05;
    rho                     rho [ 1 -3 0 0 0 0 0 ] 1;
    CrossPowerLawCoeffs
    {
        nu0                 nu0 [ 0 2 -1 0 0 0 0 ] 1e-06;
        nuInf               nuInf [ 0 2 -1 0 0 0 0 ] 1e-06;
        m                   m [ 0 0 1 0 0 0 0 ] 1;
        n                   n [ 0 0 0 0 0 0 0 ] 0;
    }

    BirdCarreauCoeffs
    {
        nu0                 nu0 [ 0 2 -1 0 0 0 0 ] 0.0142515;
        nuInf               nuInf [ 0 2 -1 0 0 0 0 ] 1e-06;
        k                   k [ 0 0 1 0 0 0 0 ] 99.6;
        n                   n [ 0 0 0 0 0 0 0 ] 0.1003;
    }
}

sigma           sigma [ 1 0 -2 0 0 0 0 ] 0.07;


// ************************************************************************* //
```

C.2 g

```
/*--------------------------------*- C++ -*----------------------*\
| =========                 |
|
| \\      /  F ield         | OpenFOAM: The Open Source CFD Toolbox
|
|  \\    /   O peration     | Version:  2.1.x
|
|   \\  /    A nd           | Web:      www.OpenFOAM.org
|
|    \\/     M anipulation  |
|
\*---------------------------------------------------------------------*/
FoamFile
{
    version     2.0;
    format      ascii;
    class       uniformDimensionedVectorField;
    location    "constant";
    object      g;
}
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //

dimensions      [0 1 -2 0 0 0 0];
value           ( 0 0 -9.81 );


// ************************************************************************* //
```

C.3 turbulenceProperties

```
/*--------------------------------*- C++ -*----------------------*\
| =========                 |
|
| \\      /  F ield         | OpenFOAM: The Open Source CFD Toolbox
|
|  \\    /   O peration     | Version:  2.1.x
|
|   \\  /    A nd           | Web:      www.OpenFOAM.org
|
|    \\/     M anipulation  |
|
\*---------------------------------------------------------------------*/
FoamFile
{
    version     2.0;
    format      ascii;
    class       dictionary;
    location    "constant";
    object      turbulenceProperties;
}
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //

simulationType  laminar;


// ************************************************************************* //
```

APPENDIX D

D.1 controlDict

```
/*--------------------------------*- C++ -*----------------------*\
| =========                 |
|
| \\      /  F ield         | OpenFOAM: The Open Source CFD Toolbox
|
|  \\    /   O peration     | Version:  2.1.x
|
|   \\  /    A nd           | Web:      www.OpenFOAM.org
|
|    \\/     M anipulation  |
|
\*---------------------------------------------------------------*/
FoamFile
{
    version     2.0;
    format      ascii;
    class       dictionary;
    location    "system";
    object      controlDict;
}
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //

application          interFoam;

startFrom            startTime;

startTime            0;

stopAt               endTime;

endTime              120;

deltaT               0.001;

writeControl         adjustableRunTime;

writeInterval        0.05;

purgeWrite           0;

writeFormat          ascii;

writePrecision       6;

writeCompression     compressed;

timeFormat           general;

timePrecision        6;

runTimeModifiable    yes;
```

```
adjustTimeStep        yes;

maxCo                 0.5;
maxAlphaCo            0.5;

maxDeltaT             1;

  functions
{
P_rgh-probes
      {
      type  probes;
      functionobjectLibs ("libsampling.so");
      probeLocations
      (
            (3.7946 0 -2)
            (3.7053 0 -2)
            (3.616 0 -2)
            (3.5267 0 -2)
            (3.4374 0 -2)

            (3.7946 0 -5)
            (3.7053 0 -5)
            (3.616 0 -5)
            (3.5267 0 -5)
            (3.4374 0 -5)

            (3.7946 0 -10)
            (3.7053 0 -10)
            (3.616 0 -10)
            (3.5267 0 -10)
            (3.4374 0 -10)

            (3.7946 0 -15)
            (3.7053 0 -15)
            (3.616 0 -15)
            (3.5267 0 -15)
            (3.4374 0 -15)

            (3.7946 0 -19)
            (3.7053 0 -19)
            (3.616 0 -19)
            (3.5267 0 -19)
            (3.4374 0 -19)

            (3.7946 0 -19.99)
            (3.7053 0 -19.99)
            (3.616 0 -19.99)
            (3.5267 0 -19.99)
            (3.4374 0 -19.99)

            (1.9542 0 0.2286)
            (1.9771 0 0.2286)
            (2.000 0 0.2286)
            (2.0229 0 0.2286)
            (2.0458 0 0.2286)
```

```
      (1.9542 0 0.2336)
      (1.9771 0 0.2336)
      (2.00 0 0.2336)
      (2.0229 0 0.2336)
      (2.0458 0 0.2336)

      (1.9542 0 0.2386)
      (1.9771 0 0.2386)
      (2.000 0 0.2386)
      (2.0229 0 0.2386)
      (2.0458 0 0.2386)

      (-1.95 0 0)
      (-1 0 0)
      (0 0 0)
      (1 0 0)
      (2 0 0)
);
fields     (p_rgh
);

outputControl   outputTime;
}

surface5m
{
type                  createSampledSurface;
surfaceName           surface5;
interpolationScheme   cellPoint;
interpolate           true;
surfaceFormat         vtk;
surface
{
   type plane;
   basePoint (3.616 0 -5);
   normalVector (0 0 1);
   }
}

surface10m
{
type                  createSampledSurface;
surfaceName           surface10;
interpolationScheme   cellPoint;
interpolate           true;
surfaceFormat         vtk;
surface
{
   type plane;
   basePoint (3.616 0 -10);
   normalVector (0 0 1);
   }
}
```

```
surface15m
{
type                  createSampledSurface;
surfaceName           surface15;
interpolationScheme   cellPoint;
interpolate           true;
surfaceFormat         vtk;
surface
{
   type plane;
   basePoint (3.616 0 -15);
   normalVector (0 0 1);
   }
}

surface19m
{
type                  createSampledSurface;
surfaceName           surface19;
interpolationScheme   cellPoint;
interpolate           true;
surfaceFormat         vtk;
surface
{
   type plane;
   basePoint (3.616 0 -19);
   normalVector (0 0 1);
   }
}

massFlux_airInlet
{
type            swakExpression;
valueType       patch;
patchName       air_inlet;
accumulations   (average max min);
outputControl   outputTime;
expression      "-(U.z*sum(area())*rho)";
verbose         true;
}

massFlux_outlet
{
type            swakExpression;
valueType       patch;
patchName       outlet;
accumulations   (average max min);
outputControl   outputTime;
expression      "U.z*sum(area())*rho";
verbose         true;
}
```

```
massFlux_waterInlet
{
type            swakExpression;
valueType       patch;
patchName       water_inlet;
accumulations   (average max min);
outputControl   outputTime;
expression      "U.x*sum(area())*rho";
verbose         true;
}

velocityFlux_airInlet
{
type            swakExpression;
valueType       patch;
patchName       air_inlet;
accumulations   (average max min);
outputControl   outputTime;
expression      "-(U.z*sum(area()))";    //phi
verbose         true;
}

velocityFlux_outlet
{
type            swakExpression;
valueType       patch;
patchName       outlet;
accumulations   (average max min);
outputControl   outputTime;
expression      "U.z*sum(area())";
verbose         true;
}

velocityFlux_waterInlet
{
type            swakExpression;
valueType       patch;
patchName       water_inlet;
accumulations   (average max min);
outputControl   outputTime;
expression      "U.x*sum(area())";
verbose         true;
}
```

```
 UwSup_outlet
  {
   type            swakExpression;
   valueType       patch;
   patchName       outlet;
   accumulations   (average
                    max
                    min);
   outputControl   outputTime;
   variables       "Us=alpha1>0.5? sum(mag(U.z)*area()):0;";
   expression      "Us/sum(area())";
   verbose         true;
}

UwSup_surface5
{
   type            swakExpression;
   valueType       surface;
   surfaceName     surface5;
   accumulations   (average
                    max
                    min);
   outputControl   outputTime;
   variables       "Us=alpha1>0.5? sum(mag(U.z)*area()):0;";
   expression      "Us/sum(area())";
   verbose         true;
}

UwSup_surface10
{
   type            swakExpression;
   valueType       surface;
   surfaceName     surface10;
   accumulations   (average
                    max
                    min);
   outputControl   outputTime;
   variables       "Us=alpha1>0.5? sum(mag(U.z)*area()):0;";
   expression      "Us/sum(area())";
   verbose         true;
}

UwSup_surface15
{
   type            swakExpression;
   valueType       surface;
   surfaceName     surface15;
   accumulations   (average
                    max
                    min);
   outputControl   outputTime;
   variables       "Us=alpha1>0.5? sum(mag(U.z)*area()):0;";
   expression      "Us/sum(area())";
   verbose         true;
}
```

```
 UwSup_surface19
{
  type            swakExpression;
  valueType       surface;
  surfaceName     surface19;
  accumulations   (average
                   max
                   min);
  outputControl   outputTime;
  variables       "Us=alpha1>0.5? sum(mag(U.z)*area()):0;";
  expression      "Us/sum(area())";
  verbose         true;
}

 UaSup_outlet
{
  type            swakExpression;
  valueType       patch;
  patchName       outlet;
  accumulations   (average
                   max
                   min);
  outputControl   outputTime;
  variables       "Us=alpha1<0.5? sum(mag(U.z)*area()):0;";
  expression      "Us/sum(area())";
  verbose         true;
}

UaSup_surface5
{
  type            swakExpression;
  valueType       surface;
  surfaceName     surface5;
  accumulations   (average
                   max
                   min);
  outputControl   outputTime;
  variables       "Us=alpha1<0.5? sum(mag(U.z)*area()):0;";
  expression      "Us/sum(area())";
  verbose         true;
}

UaSup_surface10
{
  type            swakExpression;
  valueType       surface;
  surfaceName     surface10;
  accumulations   (average
                   max
                   min);
  outputControl   outputTime;
  variables       "Us=alpha1<0.5? sum(mag(U.z)*area()):0;";
  expression      "Us/sum(area())";
  verbose         true;
}
```

```
UaSup_surface15
{
  type            swakExpression;
  valueType       surface;
  surfaceName     surface15;
  accumulations   (average
                   max
                   min);
  outputControl   outputTime;
  variables       "Us=alpha1<0.5? sum(mag(U.z)*area()):0;";
  expression      "Us/sum(area())";
  verbose         true;
}

UaSup_surface19
{
  type            swakExpression;
  valueType       surface;
  surfaceName     surface19;
  accumulations   (average
                   max
                   min);
  outputControl   outputTime;
  variables       "Us=alpha1<0.5? sum(mag(U.z)*area()):0;";
  expression      "Us/sum(area())";
  verbose         true;
}

sqr_Fr_water_outlet
{
  type            swakExpression;
  valueType       patch;
  patchName       outlet;
  accumulations   (average
                   max
                   min);
  outputControl   outputTime;
  variables       "Us=alpha1>0.5? sum(mag(U.z)*area()):0;";
  expression      "sqr((Us/sum(area()))*sqr((1000/(1000-
                  1.2))/(9.81*0.4572)))";
  verbose         true;
}

sqr_Fr_water_surface15
{
  type            swakExpression;
  valueType       surface;
  surfaceName     surface15;
  accumulations   (average
                   max
                   min);
  outputControl   outputTime;
  variables       "Us=alpha1>0.5? sum(mag(U.z)*area()):0;";
  expression      "sqr((Us/sum(area()))*sqr((1000/(1000-
                  1.2))/(9.81*0.4572)))";
  verbose         true;
```

```
}

sqr_Fr_water_surface19
{
  type            swakExpression;
  valueType       surface;
  surfaceName     surface19;
  accumulations   (average
                   max
                   min);
  outputControl   outputTime;
  variables       "Us=alpha1>0.5? sum(mag(U.z)*area()):0;";
  expression      "sqr((Us/sum(area()))*sqr((1000/(1000-
                  1.2))/(9.81*0.4572)))";
  verbose         true;
}

 sqr_Fr_air_outlet
{
  type            swakExpression;
  valueType       patch;
  patchName       outlet;
  accumulations   (average
                   max
                   min);
  outputControl   outputTime;
  variables       "Us=alpha1<0.5 ? sum(mag(U.z)*area()):0;";
  expression      "sqr((Us/sum(area()))*sqr(((1.2/(1000-
                  1.2)))/(9.81*0.4572)))";
  verbose         true;
}

sqr_Fr_air_surface15
{
  type            swakExpression;
  valueType       surface;
  surfaceName     surface15;
  accumulations   (average
                   max
                   min);
  outputControl   outputTime;
  variables       "Us=alpha1<0.5 ? sum(mag(U.z)*area()):0;";
  expression      "sqr((Us/sum(area()))*sqr(((1.2/(1000-
                  1.2)))/(9.81*0.4572)))";
  verbose         true;
}
```

```
sqr_Fr_air_surface19
{
  type            swakExpression;
  valueType       surface;
  surfaceName     surface19;
  accumulations   (average
                   max
                   min);
  outputControl   outputTime;
  variables       "Us=alpha1<0.5 ? sum(mag(U.z)*area()):0;";
  expression      "sqr((Us/sum(area()))*sqr(((1.2/(1000-
                  1.2)))/(9.81*0.4572)))";
  verbose         true;
}


Fr_outlet
{
  type            swakExpression;
  valueType       patch;
  patchName       outlet;
  accumulations   (average
                   max
                   min);
  outputControl   outputTime;
  expression      "mag(U.z)/2.118"; //divide sqr(9.81*0.4572)
  verbose         true;
}

Fr_surface5
{
  type            swakExpression;
  valueType       surface;
  surfaceName     surface5;
  accumulations   (average
                   max
                   min);
  outputControl   outputTime;
  expression      "average(mag(U.z))/2.118";
  verbose         true;
}

Fr_surface10
{
  type            swakExpression;
  valueType       surface;
  surfaceName     surface10;
  accumulations   (average
                   max
                   min);
  outputControl   outputTime;
  expression      "average(mag(U.z))/2.118";
  verbose         true;
}
```

```
Fr_surface15
{
  type            swakExpression;
  valueType       surface;
  surfaceName     surface15;
  accumulations   (average
                   max
                   min);
  outputControl   outputTime;
  expression      "average(mag(U.z))/2.118”;
  verbose         true;
}

Fr_surface19
{
  type            swakExpression;
  valueType       surface;
  surfaceName     surface19;
  accumulations   (average
                   max
                   min);
  outputControl   outputTime;
  expression      "average(mag(U.z))/2.118";
  verbose         true;
}


volumefraction_outlet
{
  type            swakExpression;
  valueType       patch;
  patchName       outlet;
  accumulations   (average
                   max
                   min);
  outputControl   outputTime;
  expression      "sum((alpha1*area()))/sum(area())";
  verbose         true;
}

 volumefraction_surface5
{
  type            swakExpression;
  valueType       surface;
  surfaceName     surface5;
  accumulations   (average
                   max
                   min);
  outputControl   outputTime;
  expression      "sum((alpha1*area()))/sum(area())";
  verbose         true;
}
```

```
    volumefraction_surface10
    {
      type            swakExpression;
      valueType       surface;
      surfaceName     surface10;
      accumulations   (average
                       max
                       min);
      outputControl   outputTime;
      expression      "sum((alpha1*area()))/sum(area())";
      verbose         true;
    }

    volumefraction_surface15
    {
      type            swakExpression;
      valueType       surface;
      surfaceName     surface15;
      accumulations   (average
                       max
                       min);
      outputControl   outputTime;
      expression      "sum((alpha1*area()))/sum(area())";
      verbose         true;
    }

    volumefraction_surface19
    {
      type            swakExpression;
      valueType       surface;
      surfaceName     surface19;
      accumulations   (average
                       max
                       min);
      outputControl   outputTime;
      expression      "sum((alpha1*area()))/sum(area())";

      verbose         true;
    }
     totalLiquid
     {
     type             volumeIntegrate;  //simpleFunctionobject
     fields           (alpha1);
     verbose          true;
     }
 }

libs
(
"libOpenFOAM.so" "libgroovyBC.so"
 "libswakFunctionObjects.so"
 "libsimpleSwakFunctionObjects.so"
);

// ************************************************************************* //
```

D.2 fvSchemes

```
/*--------------------------------*- C++ -*----------------------*\
| =========                 |
|
| \\      /  F ield         | OpenFOAM: The Open Source CFD Toolbox
|
|  \\    /   O peration     | Version:  2.1.x
|
|   \\  /    A nd           | Web:      www.OpenFOAM.org
|
|    \\/     M anipulation  |
|
\*---------------------------------------------------------------*/
FoamFile
{
    version     2.0;
    format      ascii;
    class       dictionary;
    location    "system";
    object      fvSchemes;
}
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //

ddtSchemes
{
    default          Euler;
}

gradSchemes
{
    default          Gauss linear;
}

divSchemes
{
    div(rho*phi,U)    Gauss limitedLinearV 1;
    div(phi,alpha)    Gauss vanLeer;
    div(phirb,alpha) Gauss interfaceCompression;
}

laplacianSchemes
{
    default          Gauss linear corrected;
}

interpolationSchemes
{
    default          linear;
}
```

```
snGradSchemes
{
    default           corrected;
}

fluxRequired
{
    default           no;
    p_rgh;
    pcorr;
    alpha1;
}


// ************************************************************************* //
```

D.3 fvSolutions

```
/*--------------------------------*- C++ -*----------------------*\
| =========                 |
|
| \\      /  F ield         | OpenFOAM: The Open Source CFD Toolbox
|
|  \\    /   O peration     | Version:  2.1.x
|
|   \\  /    A nd           | Web:      www.OpenFOAM.org
|
|    \\/     M anipulation  |
|
\*---------------------------------------------------------------*/
FoamFile
{
    version     2.0;
    format      ascii;
    class       dictionary;
    location    "system";
    object      fvSolution;
}
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //

solvers
{
    pcorr
    {
        solver          PCG;
        preconditioner  DIC;
        tolerance       1e-10;
        relTol          0;
    }

    p_rgh
    {
        solver          PCG;
        preconditioner  DIC;
        tolerance       1e-07;
        relTol          0.05;
    }

    p_rghFinal
    {
        $p_rgh;
        tolerance       1e-07;
        relTol          0;
    }
```

```
    U
    {
        solver          PBiCG;
        preconditioner  DILU;
        tolerance       1e-06;
        relTol          0;
    }
}

PIMPLE
{
    momentumPredictor           no;
    nCorrectors                 2;
    nNonOrthogonalCorrectors    0;
    nAlphaCorr                  2;
    nAlphaSubCycles             3;
    cAlpha                      1;
}


// ************************************************************************* //
```

APPENDIX E

E.1 Check mesh of d18v4

```
/*---------------------------------------------------------------------------*\
| =========                 |
|
| \\      /  F ield         | OpenFOAM: The Open Source CFD Toolbox
|
|  \\    /   O peration     | Version:  2.1.x
|
|   \\  /    A nd           | Web:      www.OpenFOAM.org
|
|    \\/     M anipulation  |
|
\*---------------------------------------------------------------------------*/
Build  : 2.1.x-eb976ba31c36
Exec   : checkMesh
Date   : Apr 16 2012
Time   : 12:29:52
Host   : "localhost.localdomain"
PID    : 11716
Case   : /home/user/Master/2tvent-test7
nProcs : 1
sigFpe : Enabling floating point exception trapping (FOAM_SIGFPE).
fileModificationChecking : Monitoring run-time modified files using
timeStampMaster
allowSystemOperations : Disallowing user-supplied system call
operations

// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //
Create time

Create polyMesh for time = 0

Time = 0

Mesh stats
    points:                1480
    faces:                 10370
    internal faces:        8150
    cells:                 4630
    boundary patches:      6
    point zones:           0
    face zones:            0
    cell zones:            0

Overall number of cells of each type:
    hexahedra:             0
    prisms:                0
    wedges:                0
    pyramids:              0
    tet wedges:            0
    tetrahedra:            4630
    polyhedra:             0
```

```
Checking topology...
    Boundary definition OK.
    Cell to face addressing OK.
    Point usage OK.
    Upper triangular ordering OK.
    Face vertices OK.
    Number of regions: 1 (OK).

Checking patch topology for multiply connected surfaces ...
Patch       Faces    Points   Surface topology
water_inlet  7        8          ok (non-closed singly connected)
water_wall   353      187        ok (non-closed singly connected)
air_wall     76       45         ok (non-closed singly connected)
air_inlet    6        7          ok (non-closed singly connected)
outlet       7        8          ok (non-closed singly connected)
walls        1771     896        ok (non-closed singly connected)

Checking geometry...
Overall domain bounding box (-2 -0.2286 -19.9996) (3.8446 0.2286
0.35)
Mesh (non-empty, non-wedge) directions (1 1 1)
Mesh (non-empty) directions (1 1 1)
Boundary openness (3.56709e-16 -5.18629e-17 2.68061e-17) OK.
Max cell openness = 2.18152e-16 OK.
Max aspect ratio = 6.06972 OK.
Minumum face area = 0.000124931. Maximum face area = 0.0467405.
Face area magnitudes OK.
Min volume = 7.8331e-07. Max volume = 0.00269181.  Total volume =
3.69034.  Cell volumes OK.
Mesh non-orthogonality Max: 64.0623 average: 17.9052
Non-orthogonality check OK.
Face pyramids OK.
Max skewness = 0.643544 OK.
Coupled point location match (average 0) OK.

Mesh OK.
```

E.2 Check mesh of d12v4

```
/*---------------------------------------------------------------------------*\
| =========                 |                                                 |
|
| \\      /  F ield         | OpenFOAM: The Open Source CFD Toolbox
|
|  \\    /   O peration     | Version:  2.1.x
|
|   \\  /    A nd           | Web:      www.OpenFOAM.org
|
|    \\/     M anipulation  |
|
\*---------------------------------------------------------------------------*/
Build  : 2.1.x-eb976ba31c36
Exec   : checkMesh
Date   : Apr 16 2012
Time   : 11:37:17
Host   : "localhost.localdomain"
PID    : 11555
Case   : /home/user/Master/vent4t_d12t_14.04
nProcs : 1
sigFpe : Enabling floating point exception trapping (FOAM_SIGFPE).
fileModificationChecking : Monitoring run-time modified files using
timeStampMaster
allowSystemOperations : Disallowing user-supplied system call
operations

// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //
Create time

Create polyMesh for time = 0
Time = 0

Mesh stats
    points:                1484
    faces:                 10286
    internal faces:        8122
    cells:                 4602
    boundary patches:      6
    point zones:           0
    face zones:            0
    cell zones:            0

Overall number of cells of each type:
    hexahedra:             0
    prisms:                0
    wedges:                0
    pyramids:              0
    tet wedges:            0
    tetrahedra:            4602
    polyhedra:             0
```

```
Checking topology...
    Boundary definition OK.
    Cell to face addressing OK.
    Point usage OK.
    Upper triangular ordering OK.
    Face vertices OK.
    Number of regions: 1 (OK).

Checking patch topology for multiply connected surfaces ...
Patch        Faces   Points   Surface topology
water_inlet   6      7          ok (non-closed singly connected)
water_wall    357    186        ok (non-closed singly connected)
air_inlet     6      7          ok (non-closed singly connected)
air_wall      37     25         ok (non-closed singly connected)
outlet        6      7          ok (non-closed singly connected)
walls         1752   884        ok (non-closed singly connected)

Checking geometry...
Overall domain bounding box (-4.00132 -0.152209 -20.0015) (1.23029
0.1524 0.2738)
Mesh (non-empty, non-wedge) directions (1 1 1)
Mesh (non-empty) directions (1 1 1)
Boundary openness (4.57022e-16 2.2541e-17 3.27903e-17) OK.
Max cell openness = 2.6479e-16 OK.
Max aspect ratio = 4.16107 OK.
Minumum face area = 0.000512308. Maximum face area = 0.0236613.
Face area magnitudes OK.
Min volume = 5.25087e-06. Max volume = 0.000973083.  Total volume =
1.56248.  Cell volumes OK.
Mesh non-orthogonality Max: 56.3091 average: 17.0262
Non-orthogonality check OK.
Face pyramids OK.
Max skewness = 0.677926 OK.
Coupled point location match (average 0) OK.

Mesh OK.

End
```

E.3 Check mesh of d24v4

```
/*---------------------------------------------------------------------------*\
| =========                 |                                                 |
|
| \\      /  F ield         | OpenFOAM: The Open Source CFD Toolbox
|
|  \\    /   O peration     | Version:  2.1.x
|
|   \\  /    A nd           | Web:      www.OpenFOAM.org
|
|    \\/     M anipulation  |
|
\*---------------------------------------------------------------------------*/
Build  : 2.1.x-eb976ba31c36
Exec   : checkMesh
Date   : Apr 16 2012
Time   : 12:02:32
Host   : "localhost.localdomain"
PID    : 11617
Case   : /home/user/Master/vent4t_d24t_14.04
nProcs : 1
sigFpe : Enabling floating point exception trapping (FOAM_SIGFPE).
fileModificationChecking : Monitoring run-time modified files using
timeStampMaster
allowSystemOperations : Disallowing user-supplied system call
operations

// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //
Create time

Create polyMesh for time = 0

Time = 0

Mesh stats
    points:                854
    faces:                 5828
    internal faces:        4556
    cells:                 2596
    boundary patches:      6
    point zones:           0
    face zones:            0
    cell zones:            0

Overall number of cells of each type:
    hexahedra:             0
    prisms:                0
    wedges:                0
    pyramids:              0
    tet wedges:            0
    tetrahedra:            2596
    polyhedra:             0
```

```
Checking topology...
    Boundary definition OK.
    Cell to face addressing OK.
    Point usage OK.
    Upper triangular ordering OK.
    Face vertices OK.
    Number of regions: 1 (OK).

Checking patch topology for multiply connected surfaces ...
Patch         Faces   Points     Surface topology
water_inlet   6       7          ok (non-closed singly connected)
water_wall    224     121        ok (non-closed singly connected)
air_inlet     6       7          ok (non-closed singly connected)
air_wall      35      24         ok (non-closed singly connected)
outlet        6       7          ok (non-closed singly connected)
walls         995     507        ok (non-closed singly connected)

Checking geometry...
Overall domain bounding box (-4.00285 -0.30474 -20.003) (2.45978
0.304791 0.4262)
Mesh (non-empty, non-wedge) directions (1 1 1)
Mesh (non-empty) directions (1 1 1)
Boundary openness (3.19663e-16 6.66025e-17 1.79957e-17) OK.
Max cell openness = 2.0869e-16 OK.
Max aspect ratio = 4.48088 OK.
Minumum face area = 0.000709884. Maximum face area = 0.0755445.
Face area magnitudes OK.
Min volume = 1.03009e-05. Max volume = 0.00688881.  Total volume =
6.42033.  Cell volumes OK.
Mesh non-orthogonality Max: 51.8639 average: 16.3945
Non-orthogonality check OK.
Face pyramids OK.
Max skewness = 0.549088 OK.
Coupled point location match (average 0) OK.

Mesh OK.
```

E.4 Check mesh of d18u

```
/*---------------------------------------------------------------------------*\
| =========                 |
|
| \\      /  F ield         | OpenFOAM: The Open Source CFD Toolbox
|
|  \\    /   O peration     | Version:  2.1.x
|
|   \\  /    A nd           | Web:      www.OpenFOAM.org
|
|    \\/     M anipulation  |
|
\*---------------------------------------------------------------------------*/
Build  : 2.1.x-eb976ba31c36
Exec   : checkMesh
Date   : May 10 2012
Time   : 08:35:57
Host   : "localhost.localdomain"
PID    : 16028
Case   : /home/user/Master/unvented_19.04
nProcs : 1
sigFpe : Enabling floating point exception trapping (FOAM_SIGFPE).
fileModificationChecking : Monitoring run-time modified files using
timeStampMaster
allowSystemOperations : Disallowing user-supplied system call
operations

// * * * * * * * * * * * * * * * * * *  * * * * * * * * * * * * * * * * * * //
Create time

Create polyMesh for time = 0

Time = 0

Mesh stats
    points:                927
    faces:                 5994
    internal faces:        4502
    cells:                 2624
    boundary patches:      4
    point zones:           0
    face zones:            0
    cell zones:            0
```

```
Overall number of cells of each type:
    hexahedra:             0
    prisms:                0
    wedges:                0
    pyramids:              0
    tet wedges:            0
    tetrahedra:            2624
    polyhedra:             0


Checking topology...
    Boundary definition OK.
    Cell to face addressing OK.
    Point usage OK.
    Upper triangular ordering OK.
    Face vertices OK.
    Number of regions: 1 (OK).


Checking patch topology for multiply connected surfaces ...
Patch        Faces    Points   Surface topology
water_inlet  6        7          ok (non-closed singly connected)
water_wall   230      121        ok (non-closed singly connected)
outlet       6        7          ok (non-closed singly connected)
walls        1250     631        ok (non-closed singly connected)


Checking geometry...
Overall domain bounding box (-2 -0.22853 -19.9996) (3.8446 0.22827
0.2686)
Mesh (non-empty, non-wedge) directions (1 1 1)
Mesh (non-empty) directions (1 1 1)
Boundary openness (-1.03111e-16 4.79973e-17 2.05929e-17) OK.
Max cell openness = 2.02566e-16 OK.
Max aspect ratio = 6.11286 OK.
Minumum face area = 0.00892658. Maximum face area = 0.0510272.  Face
area magnitudes OK.
Min volume = 0.000485215. Max volume = 0.00301899.  Total volume =
3.55627.  Cell volumes OK.
Mesh non-orthogonality Max: 42.7066 average: 14.3695
Non-orthogonality check OK.
Face pyramids OK.
Max skewness = 0.457479 OK.
Coupled point location match (average 0) OK.

Mesh OK.
```

E.5 Check mesh of d18v2

```
/*---------------------------------------------------------------------------*\

| =========                 |
|
| \\      /  F ield         | OpenFOAM: The Open Source CFD Toolbox
|
|  \\    /   O peration     | Version:  2.1.x
|
|   \\  /    A nd           | Web:      www.OpenFOAM.org
|
|    \\/     M anipulation  |
|
\*---------------------------------------------------------------------------*/
Build  : 2.1.x-eb976ba31c36
Exec   : checkMesh
Date   : May 10 2012
Time   : 08:49:49
Host   : "localhost.localdomain"
PID    : 16200
Case   : /home/user/Master/d18v2_06.05
nProcs : 1
sigFpe : Enabling floating point exception trapping (FOAM_SIGFPE).
fileModificationChecking : Monitoring run-time modified files using
timeStampMaster
allowSystemOperations : Disallowing user-supplied system call
operations

// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //
Create time

Create polyMesh for time = 0

Time = 0


Mesh stats
    points:                1480
    faces:                 10370
    internal faces:        8150
    cells:                 4630
    boundary patches:      6
    point zones:           0
    face zones:            0
    cell zones:            0
```

```
Overall number of cells of each type:
    hexahedra:             0
    prisms:                0
    wedges:                0
    pyramids:              0
    tet wedges:            0
    tetrahedra:            4630
    polyhedra:             0


Checking topology...
    Boundary definition OK.
    Cell to face addressing OK.
    Point usage OK.
    Upper triangular ordering OK.
    Face vertices OK.
    Number of regions: 1 (OK).


Checking patch topology for multiply connected surfaces ...
Patch        Faces    Points   Surface topology
water_inlet  7        8          ok (non-closed singly connected)
water_wall   353      187        ok (non-closed singly connected)
air_wall     76       45         ok (non-closed singly connected)
air_inlet    6        7          ok (non-closed singly connected)
outlet       7        8          ok (non-closed singly connected)
walls        1771     896        ok (non-closed singly connected)


Checking geometry...
Overall domain bounding box (-2 -0.2286 -19.9996) (3.8446 0.2286
0.35)
Mesh (non-empty, non-wedge) directions (1 1 1)
Mesh (non-empty) directions (1 1 1)
Boundary openness (3.56709e-16 -5.18629e-17 2.68061e-17) OK.
Max cell openness = 2.18152e-16 OK.
Max aspect ratio = 6.06972 OK.
Minumum face area = 0.000124931. Maximum face area = 0.0467405.
Face area magnitudes OK.
Min volume = 7.8331e-07. Max volume = 0.00269181.  Total volume =
3.69034.  Cell volumes OK.
Mesh non-orthogonality Max: 64.0623 average: 17.9052
Non-orthogonality check OK.
Face pyramids OK.
Max skewness = 0.643544 OK.
Coupled point location match (average 0) OK.


Mesh OK.
```

APPENDIX F

F.1 Content of enclosed CD

The enclosed CD consists of five folders of the respective cases. Each folder contains graphs in jpeg format of all residuals and all sampled results. One can also find movies of pressure, velocity and volume fraction in each case folder. A special extract of the valve is recorded to indicate the particular higher velocities present. A zipped file of each case with the necessary sub directories 0, constant and system are also included. The geometry and mesh ar included in the zipped folder and can be opened in the pre-processing program Salome.