University of Alberta

Data analysis for the classification of gas-liquid and liquid-solid (slurry) flows using digital signal processing

by

Roberto Jose Fedon Sarmiento

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

Master of Science

in

Chemical Engineering

Chemical and Materials Engineering

©Roberto Jose Fedon Sarmiento
Spring 2013
Edmonton, Alberta

DEDICATION

To Mariangel for always believing in me

To my family for all their support

ABSTRACT

The Canadian oil sand industry operates three phase gas-liquid-solid hydrotransport pipelines to transport oil sand ore from mines to processing facilities. This hydrotransport process is critical to downstream separation activities, given the conditioning steps that occur within it: (i) lump ablation, (ii) bitumen liberation and (iii) bitumen-air attachment. Achieving the appropriate hydrodynamic conditions within the pipe to promote the conditioning steps has been a challenge. Although operational targets are known, e.g. suspended solids and dispersed bubble flow, determining and furthermore monitoring online behaviour has not been satisfactorily achieved. To solve this, a novel modelling technique using digital signal processing was developed, and tested on gas-liquid and liquid-solid horizontal flows. Results show the model accurately describes the existence and transition between dispersed bubble and intermittent flows in the gas-liquid system. For slurry flow, the model accurately distinguishes a moving bed from a suspended flow, but does not describe the transition.

ACKNOWLEDGEMENT

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS

a: approximation coefficients (from wavelet transform)

b: number of latent variables

BIC: Bayesian information criterion

c: regression coefficients

d: detail coefficients (from wavelet transform)

E: observation noise term

HPF: high pass filter

i: latent variables counter

j: j-eth time step

k: excess kurtosis

LPF: low pass filter

m: low and high pass filter component counter

n: number of samples

p: X latent variables vector

P: X latent variables matrix

q: Y latent variables vector

Q: Y latent variables matrix

R: autocorrelation coefficient

s: skewness

S: discrete-time signal

t: X scores vector

T: X-scores matrix

u: Y scores vector

U: Y-scores matrix

w: covariance matrix between X and u

X: observations matrix

Y: raw classifier

Z: z-score

Greek symbols:

β: regression matrix

σ: standard deviation

ρ: Density

τ: time shift for autocorrelation test

μ: mean

ν: kinematic viscosity

Subscripts and superscripts:

L: liquid

G: gas

S: solids

i: wavelet decomposition level

j: counter for PLS iteration process

t: time step counter

T: transpose of matrix or vector

^: estimated variable

CHAPTER 1: PROBLEM STATEMENT

The following chapter contains information on current issues that plague the field of multiphase flows. Also, a specific problem is outlined, a set of objectives to solve the problem are presented, and research justification and limitations are presented.

1.1 Problem description

Multiphase flows are commonly found in many industries (Ducrocq 1997, Ayala et al. 2007). The reason behind their application varies from necessity – such as transporting slurries (Brock 2006), to efficiency improvement – such as catalysts in fluidized bed reactors (Dudukovic et al. 2002). Even though many different systems have been widely studied (Dukler 1969, Yemada et al. 1976, Orell 2007), their behaviour is still not fully understood (Joekar-Niasar et al. 2011). One of the most difficult behaviours to describe for these types of flows is the wide array of flow regimes or flow patterns that arise from the multiphase mixtures (Yemada et al. 1976, Triplett et al 1999).

The classification of multiphase flows depends on the nature of the different components of the mixture, and requires that all components remain insoluble and immiscible in each other. Multiphase flows can be two phase flows, where there are only two components: gas-liquid, liquid-liquid, gas-solid and liquid-solid; three phase flows, where there are three components: gas-liquid-solid and gas-liquid-liquid; and even more complex if there are more than three distinct phases. Each of these mixtures is so unique that they require a dedicated field of study to fully understand them.

One of many industrial applications of a multiphase flow is the hydraulic transport of solids, or hydrotransport, which is commonly used in most mining industries (Brock 2006). It involves the addition of a carrier fluid - commonly water - to create liquid-solid slurries to transport ores through a pipeline along greater lengths than most practical conveyer systems are able to. This technique provides a reliable and cost-effective method for steady-state supply to downstream separation operations (Chadwick 2011). In the Canadian oil sand industry, this technique is used for transporting the oil sand ore from the mines to the separation plants, through the production of a highly-concentrated, water-based slurry (Shook et al. 2002).

Since the 1990's, the hydrotransport process has been used as a preceding step to downstream separation processes in the mining of oil sand and extraction of bitumen (Chastko 2004), during which several conditioning processes occur (Masliyah et al. 2004). The first step is the lump ablation, in which lumps of oil sand are broken down by heat transfer (heating of the lump) and shearing away of the outer layer. The lump ablation continues until individual grains of sand, covered by a thin coat of oil, are exposed. Once the sand is exposed, several factors (fluid chemistry, pH, temperature, shear, etc) cause the oil coating on the grain of sand to slowly recede from the grain surface, forming an oil droplet. The oil droplet may remain attached to the grain of sand, with a much lower adhesion strength compared to the oil coating, or may be stripped away by shear forces. This step is referred to as oil liberation. Once the oil droplet is formed, the final step, air attachment, occurs. During this step, air bubbles dispersed in the three phase mixture become attached to the oil droplets, separating them from the sand grain if they are still attached, due to shearing and buoyancy forces. The effectiveness of the aeration process is vital to downstream gravity separators, given that non aerated bitumen has almost the same density as water. Although several gases are naturally present in the hydrotransport mixture, it is not enough for the complete conditioning of the liberated bitumen (Flynn et al. 2004, Sanders et al. 2007).

The aforementioned conditioning steps in the hydrotransport pipeline are a combination of several required hydrodynamic conditions. The two steps most relevant to this study are: (i) contact between the chemically loaded water and the oil covered sand grains to promote oil liberation, primarily liquid-solid interactions (Bello et al. 2005); and (ii) the availability of air bubbles to promote oil-air collisions, mostly gas-liquid interactions (Orell 2007). To ensure these conditions occur, the sand grains must be in contact with the liquid, which requires the suspension of the sand grains. To promote bitumen-air attachment, there must be sufficient air in the mixture and it must be adequately dispersed (Flynn et al. 2004).

The injection of additional gas appears to be the most successful method to ensure there is sufficient volume of gas in the hydrotransport mixtures, which typically involves the injection of air into the pipeline to create a gas-liquid-solid three phase flow (Wallwork et al. 2004). This three-phase system increases agitation and availability of

2

air, improving the possibility of contact between the oil droplets and air bubbles. However, to achieve favourable results, appropriate amounts of air must be injected. Due to the complex nature of the flow inside the pipelines, determining the appropriate amount has proven to be a challenge.

Several factors have been determined to affect the attachment of air to the bitumen droplets (Flynn et al. 2004, Masliyah et al. 2004, Wallwork et al. 2004, Sanders et al. 2007). The most important factors are the air concentration in the mixture (Flynn et al. 2004, Sanders et al. 2007) and the air bubble size distribution (Ahmed et al. 1985, Dai et al. 2000). Sanders et al. (2007) demonstrated that air attachment with no additional air injected into the pipeline led to very poor recoveries when compared to injection of air volumetric concentrations of 5% and 10%, with little improvement from 5% to 10%. Ahmed et al. (1985) proved that the finer air bubbles (75 μm) had a higher attachment efficiency to oil than larger bubbles (up to 600 μm) under all experimental conditions that were studied. The injection of air at the appropriate amount is easily controlled in an industrial setting, but the bubble size distribution is not.

In a general sense, the injection of gas into slurry flows will create new flow patterns, which will affect the pressure gradient and mixing intensity within the pipe. These three-phase flow regimes have been studied (Orell 2007) and a limited degree of understanding has been achieved. The most important conclusion for this research is that the highest pressure drops and lowest dispersion/mixing rates in these gas-liquid-solid mixtures occur in intermittent flow regimes (plug and slug flow), similar to two phase gas-liquid flows, over stationary solids beds, similar to liquid-solid flows. In intermittent flow the gas phase is not dispersed in the liquid but instead forms large slugs which flow along with the slurry without mixing (Crowe 2005). In flows involving stationary beds, the heavier solids deposit on the bottom of the pipe and do not flow with the mixture. These flow regimes should be avoided during the operation of oil sand hydrotransport pipelines, as they not only increase energy consumption but also decrease oil-air interaction. To obtain small bubbles and appropriate sand grain-water interactions, the flow pattern must be dispersed bubble with suspended solids flow, which is the current target for operating the hydrotransport pipelines.

For three phase systems, Orell (2007) stated that gas-liquid-solid flows behave analogous to a gas-liquid mixture flowing over a bed of solids (either stationary or moving), with solids suspended in the liquid phase. For high velocity systems, the solids may become suspended, being mostly found in the liquid phase. A study by Bello et al. (2005) also stated: "local sand holdup is caused by the slip velocity between the sand and the liquid phase of the multiphase mixtures and the increase rate of particle-particle collision". The study also states that although gas-solid collisions do occur, they result in horizontal displacement of the solids, not vertical (i.e. suspension). This indicates that the presence of a gas phase does not directly affect the suspension of the solids (indirectly, a gas phase creates turbulence which increases solid suspensions). From these two studies (Bello et al. 2005, Orell 2007), a reasonable assumption can be made that the three phase system can be decomposed into a gas-liquid component and a liquid-solid component, to simplify the system. This is necessary for the proposed research, as is explained later, given the novel method attempted for its modelling.

One method for determining gas-liquid flow patterns inside pipelines is through flow maps developed using dimensional analysis (Yemada et al. 1976). This method uses equilibrium states of competing forces between the phases as transition criteria for dominance of flow patterns. For this research, the important criterion to consider corresponds with the onset of dispersed bubble flow. At very low gas and liquid velocities the flow pattern is stratified flow, in which the gas phase flows on top of the liquid phase due to buoyancy forces, with a relatively flat interface between the phases. As velocities in both phases are increased, waves in the liquid phase being to form, with wave size proportional to velocity. Once velocity is high enough, the waves will reach the top of the pipe, trapping gas bubbles, forming intermittent flow regimes (slug and plug flow). If the liquid velocity is further increased, while maintaining a constant gas velocity, the size and force of the waves overcome the buoyant forces of the gas phase, causing it to become dispersed within the liquid phase. This increase in turbulence to buoyancy forces is what causes dispersed bubble flow. The fact that the onset of the desired flow pattern (dispersed bubble) is a direct transition from the undesired flow pattern (intermittent flow) is cause from many problems in industry.

For liquid-solid (slurry) flows, characterization of flow behaviour is much simpler. In essence, a slurry flow can behave in one of three ways: (i) solids drop to the bottom of the pipe and do not move, forming a stationary bed; (ii) solids are slightly suspended, causing the bed to move; or (iii) the solids become suspended and are dispersed (not necessarily uniform) throughout the pipe. Suspended flow can be additionally divided into heterogeneous and homogeneous suspension, if the solids concentration is constant or not throughout the pipe (Shook et al 2002)

Stationary bed flow has no practical value for industry as it involves the accumulation of solids in the pipeline, which can cause higher energy consumption per mass of solids transported (Sanders et al. 2004) and increased pipeline wear (Wilson et al. 2006). The transition from stationary to moving bed on the other hand marks the onset of industrial operability, and is defined by the deposition velocity of the solids. The deposition velocity is the minimum flowing velocity of the mixture that achieves suspension of the solids. Although it is estimated empirically, its basis is a force balance over the solid particle, where competing gravity and viscous forces are considered. Gravity forces are dependent on particle diameter, density and concentration, while viscous forces arise from turbulent diffusivity of the liquid exerting a drag force that acts to lift the solid (Shook et al. 2002). To achieve suspended flow, the velocity is increased beyond the deposition velocity, causing the moving bed to decrease in height. If the velocity is further increased, the bed height eventually reaches zero and concentration along the pipe begins to even out. Although fully suspended homogeneous flow allows the best oil-air interaction in hydrotransport pipelines, it is energy intensive, making it impractical for most particle sizes and slurry concentrations of interest. Most pipelines are operated at speeds slightly above deposition velocities (Wilson et al. 2006).

Although it has been determined which flow pattern is desirable, this still does not provide enough information for the industry to use. It is known that plug and slug flow over a stationary bed should be avoided, and that dispersed bubbly flow along with suspension of solids should be attained; however, determining the actual flow regime inside an operational, closed, pipeline transporting an opaque and erosive mixture is extremely difficult, from both a measurements (sensors) and analytical (equations) perspective.

The inapplicability of current methods of multiphase analysis in industry calls for an alternative method for identifying flow regimes, one that is both practical and accurate. For this, several authors (Li et al. 2001, Li 2002, Briens et al. 2005, Jana et al. 2006, Chakrabarti et al. 2007, Ding et al. 2007, Suman et al. 2010) have attempted to use digital signal processing applied to high frequency pipeline pressure measurements in search of repetitive patterns that allow for distinction among different flow regimes.

Briens et al. (2005) studied three phase gas-liquid-solid fluidized bed with air injection and two different types of solids, polypropylene and glass beads. Using a bubble conductivity probe, the authors studied the power spectra obtained from the Fourier transform (FFT), wavelet transform, chaos analysis and fractal analysis of the measurements as a function of varying gas and liquid velocities. Their most significant result was identifying the transition from coalesced to dispersed bubble regimes using the coefficient of variation (CV) of the wavelet detail coefficients.

For a gas-liquid system, the most relevant study to date was on an air-water vertical flow to identify slug flow patterns (Suman et al. 2010). Using similar tools as Briens et al. (2005), Fourier and wavelet analysis, as well as the probability density function (PDF) and statistical analysis, they decomposed conductivity measurements from an air-water vertical pipe flow. Among their results, they constructed an objective, logic (*if* operator) based flow pattern classifier using results from the PDF, ratio of peaks (ROP), skewness, slug frequency and wavelet parameters. However, the classification scheme is not fully presented, especially for the case of competing possible patterns. Probability based classification should present better results.

As yet, there have not been any relevant studies using these signal processing tools on liquid-solid slurries. The only studies on dispersed solids flow have been for gas-solid mixtures. Li et al. (2001, 2002) studied a horizontal air-polyethylene mixture using statistics, FFT and wavelet applied to static wall pressure measurements. These authors identified differences in the power spectrum and PDF as present only for dune flow or suspended flow. The marked differences could be enough to attempt modeling of the phenomena; however, this was not attempted.

Signal processing has proven to be a potentially viable method for characterization of gas-liquid, liquid-solid and possibly gas-liquid-solid flows, but is still

lacking conclusive results, due to the nature of the studies that have been conducted (Li et al. 2001, 2002, Briens et al. 2005). Up to now, researchers have focused on highlighting slight differences in the high frequency measurements of one flow pattern with another, but have not attempted to build a model which describes this. Preliminary results have shown this distinction to be possible; therefore this research project is focused on (i) verifying differences between measurements at different flow characteristics **and also** (ii) constructing a model through pattern recognition software. The use of computerized pattern recognition should eliminate human bias in the modeling process and increase the probability of success.

As with any "black box" approach to modeling, the possible tools that can be used are plenty and only a few prove to be viable. The tools chosen here for the decomposition of measurements, based on results in previous research (Briens et al. 2005, Chakrabarti et al. 2006, 2007, Suman et al. 2010), are simple statistical analysis, the wavelet transform and the Fourier transform. These are the preprocessing techniques which help extract relevant pieces of information from the original measurements.

For the pattern recognition and modeling portion, partial least squares discriminant analysis is used, which is highly effective at building models in which the importance of input variables is unknown (Indahl et al. 2007, Mateos-Aparicio 2011). This is achieved by using an algorithm to extract important features out of the data set, and use these variables to build a model. This technique is commonly used in many empirical fields with very good results. The validation and classification of new data sets is done through a naive Bayes classifier, which uses probability to determine the best classification decision (Zhang 2004, Kuncheva 2007).

1.2 Problem definition

Given the complexity of traditional flow analysis towards determining the flow regime in gas-liquid and gas-liquid-solid flows, a new method is required. This new method must be both simple and accurate, in such a way that online monitoring is possible in an industrial environment. Given this is the first attempt at providing such a model, the much more complex three phase system is not considered. Instead, the

underlying basis for three phase flow (gas-liquid and liquid-solid) systems is evaluated as proof-of-concept that such a model might be possible.

1.2.1   Current state

Current methods for analyzing gas-liquid, liquid-solid and gas-liquid-solid pipe flows were not designed for industrial applications, and thus limited knowledge of real time behaviour inside closed, opaque pipelines is available. Given the flow is not properly characterized, those systems are being operated under suboptimal conditions, leading to a decrease in yield and energy efficiency.

1.2.2   Desired state

Having a fully operational modelling procedure, the flow regime inside the hydrotransport pipelines, or other gas-liquid and/or slurry flow systems, is determined through digital signal processing and monitored online (real time). The increase in detection accuracy and decrease in detection delay allow a more efficient use of energy and materials, improving process reliability and productivity.

1.3 Objectives

The main objective of this research is to build a modeling procedure based on digital signal processing for determining flow conditions inside a pipe for two-phase gas-liquid and liquid-solid (slurry) flows.

The specific objectives are to:

- Analyze high frequency pressure measurements from lab scale pipe flow experiments through the use of wavelet, spectral and statistical analysis.
- Construct a procedure for building models that classify flow patterns and/or mixture flow behaviour.
- Validate the applicability of the procedure on both gas-liquid and liquid-solid flow systems.

1.4 Justification

The oil sand industry requires a set of very specific operating conditions in the hydrotransport pipeline to promote oil sand conditioning (Masliyah et al. 2004). The oil liberation step depends on fluid chemistry and efficient water-sand interactions (fully

suspended slurry), and the air attachment process is dependent on flow pattern (dispersed bubbly flow), air availability (concentration) and bubble size distribution (Sanders et al. 2007). Although the suspension of solids in a slurry is well understood, determining all these factors simultaneously in an industrial setting has proven to be a challenge. If a model for online detection of at least one of these factors (flow characteristics/flow pattern/solids suspension) is built, the operation of these pipelines would be greatly improved. Although a fully operational industrial model would require the study of three phase gas-liquid-solid flows, the study of two phase flows is the first step. By studying the behaviour of both gas-liquid and liquid-solid pipe flows using the same modeling technique, a precedent is created for future work in three phase gas-liquid-solid.

Additionally, although this research is focused on the oil sands industry, it may be applied to any field of study that involves a multiphase flow pipeline, including common industries such as water treatment.

1.5 Limitation

The main limitation of this research is the availability of experimental data. This affects the research in two ways. The first is that, given this is the first attempt at modelling flow regimes with digital signal processing, the modelling procedure is not optimal. Arriving at the best possible modeling scheme requires extensive work on numerous systems in order to truly capture the essence of the data. This would require a number of experiments not worth the effort before knowing if the modelling is possible. The second way the research is affected is in the accuracy of the models presented here. Since similar flow regimes can be obtained at different flow conditions, such as different pipe diameters, velocities, solids concentrations (to name only a few), the accuracy of the model will be poor if built only on measurements of a specific pipe diameter or using the same fluids and solids, given it will be biased towards this specific configuration of variables. The obtained model should not be obsolete on different systems, yet it is expected that significant error will be present. To construct a truly universal model for determining flow regimes, it would be necessary to calibrate it with experimental data from as many combinations as possible of all variables known to affect flow regimes.

References

Ahmed, N., and Jameson, G. J. (1985). "The effect of bubble size on the rate of flotation of fine particles." Int.J.Miner.Process., 14(3), 195-215.

Ayala, O. F., Ayala, L. F., and Ayala, O. M. (2007). "Multi-phase Flow Analysis in Oil and Gas Engineering Systems and its Modelling." Hydrocarbon World, 57-60.

Bello, O. O., Reinicke, K. M., and Teodoriu, C. (2005). "Particle Holdup Profiles in Horizontal Gas-liquid-solid Multiphase Flow Pipeline." Chem. Eng. Technol., 28(12), 1546-1553.

Briens, L. A., and Ellis, N. (2005). "Hydrodynamics of three-phase fluidized bed systems examined by statistical, fractal, chaos and wavelet analysis methods." Chemical Engineering Science, 60(22), 6094-6106.

Brock, S. T. H. (2006). "The use of hydrotransportation systems in the mining environment." The Journal of the Southern African Institute of Mining and Metallurgy, 106 789-792.

Chakrabarti, D. P., Das, G., and Das, P. K. (2007). "Identification of stratified liquid-liquid flow through horizontal pipes by a non-intrusive optical probe." Chemical Engineering Science, 62(7), 1861-1876.

Chakrabarti, D. P., Das, G., and Das, P. K. (2006). "The transition from water continuous to oil continuous flow pattern." AICHE J., 52(11), 3668-3678.

Chastko, P. A. (2004). *Developing Alberta's Oil Sands: From Karl Clark to Kyoto*. University of Calgary Press.

Crowe, C. T. (2005). *Multiphase Flow Handbook*. Taylor & Francis.

Dai, Z., Fornasiero, D., and Ralston, J. (2000). "Particle-bubble collision models - a review." Adv.Colloid Interface Sci., 85(2), 231-256.

Ding, H., Huang, Z., Song, Z., and Yan, Y. (2007). "Hilbert-Huang transform based signal analysis for the characterization of gas-liquid two-phase flow." Flow Meas.Instrum., 18(1), 37-46.

Ducrocq, J. (1997). "Numerical simulations of industrial processes involving fluid dynamics, combustion and radiation." 4th International Colloquium on Process Simulation.

Dudukovic, M. P., Larachi, F., and Mills, P. L. (2002). "Multiphase catalytic reactors: a perspective on current knowledge and future trends." Catalysis Reviews: Science & Engineering, 44(1), 123.

Dukler, A. E. (1969). *Gas-liquid flow in pipelines*. American Gas Association.

Flynn, M., Bara, B., Czarnecki, J., and Masliyah, J. (2001). "An investigation of the effect of air addition during oil sand conditioning." Can.J.Chem.Eng., 79(3), 468-470.

Indahl, U. G., Martens, H., and Naes, T. (2007). "From dummy regression to prior probabilities in PLS-DA." J.Chemometrics, 21(12), 529-536.

Jana, A. K., Das, G., and Das, P. K. (2006). "Flow regime identification of two-phase liquid–liquid upflow through vertical pipe." Chemical Engineering Science, 61(5), 1500-1515.

Joekar-Niasar, V., and Hassanizadeh, S. M. (2011). "Specific interfacial area: The missing state variable in two-phase flow equations?" Water Resour.Res., 47(5), W05513.

Kuncheva, L. I. (2006). "On the optimality of Naive Bayes with dependent binary features." Pattern Recog.Lett., 27(7), 830-837.

Li, H. (2002). "Application of wavelet multi-resolution analysis to pressure fluctuations of gas-solid two-phase flow in a horizontal pipe." Powder Technol, 125(1), 61.

Li, H., and Tomita, Y. (2001). "Characterization of pressure fluctuation in swirling gas-solid two-phase flow in a horizontal pipe." Advanced Powder Technology, 12(2), 169-185.

Masliyah, J., Zhou, Z. J., Xu, Z., Czarnecki, J., and Hamza, H. (2004). "Understanding Water-Based Bitumen Extraction from Athabasca Oil Sands." Can.J.Chem.Eng., 82(4), 628-654.

Mateos-Aparicio, G. (2011). "Partial Least Squares (PLS) Methods: Origins, Evolution, and Application to Social Sciences." Communications in Statistics - Theory and Methods, 40(13), 2305-2317.

Orell, A. (2007). "The effect of gas injection on the hydraulic transport of slurries in horizontal pipes." Chemical Engineering Science, 62(23), 6659-6676.

Richardson, J. F., and Zaki, W. N. (1954). "The sedimentation of a suspension of uniform spheres under conditions of viscous flow." Chemical Engineering Science, 3(2), 65-73.

Sanders, R. S., Schaan, J., and McKibben, M. M. (2007). "Oil sand slurry conditioning tests in a 100 mm pipeline loop." Can.J.Chem.Eng., 85(5), 756.

Shook, C. A., Gillies, R. G., and Sanders, R. S. (2002). *Pipeline Hydrotransport with Applications in the Oil Sand Industry*. SRC Publications, Saskatoon, SK.

Suman, Y. K., Mandal, T. K., and Das, G. (2010). "Use of Digital Signal Analysis to Identify Slug Flow in a Narrow Vertical Pipe." Chem.Eng.Commun., 197(10), 1287-1302.

Taitel, Y., and Dukler, A. E. (1976). "A model for predicting flow regime transitions in horizontal and near horizontal gas-liquid flow." AICHE J., 22(1), 47-55.

Triplett, K. A., Ghiaasiaan, S. M., Abdel-Khalik, S., and Sadowski, D. L. (1999). "Gas-liquid two-phase flow in microchannels Part I: two-phase flow patterns." Int.J.Multiphase Flow, 25(3), 377-394.

Wallwork, V., Xu, Z., and Masliyah, J. (2004). "Processibility of Athabasca Oil Sand Using a Laboratory Hyd ro t ransport Extraction System (LHES)." Can.J.Chem.Eng., 82(4), 687-695.

Zhang, H. (2004). "The Optimality of Naive Bayes." FLAIRS Conference. http://courses.ischool.berkeley.edu/i290-dm/s11/SECURE/Optimality_of_Naive_Bayes.pdf

CHAPTER 2: BACKGROUND

This chapter presents the objectives and relevant conclusions of previous research conducted in the field of digital signal processing for multiphase flows. Also, the theoretical background of the different tools used in this investigation for the modelling process are presented and explained.

2.1 Previous research

Advances in digital signal processing have had a broad impact on many fields, among which are data compression, digital imaging, neurophysiology and others. This is mostly attributed to the wide array of available tools that can be manipulated to one's needs. Depending on the situation, completely different applications can be given to the same tool. In the field of fluid mechanics, this method of analysis is a relatively recent but widely used approach to attempt to classify flow regimes. Although it has had varied success, a few promising results seem to keep the field open to new perspectives and methods. All attempts in this field revolve around the use of the wavelet transform.

The acceptance of the idea of using wavelets in this field is based on its simplicity when comparing this type of analysis with traditional/theoretical analysis. In the latter, most systems consist of a large number of equations to be solved simultaneously (Orell 2007) or must be clustered into dimensionless groups (Taitel and Dukler 1976). Both models appear to provide accurate predictions of flow characteristics only under very specific operating conditions and only when specific inputs can be accurately specified. In many industrial applications, either or both challenges occur, which greatly reduces the utility of these models. In the case of wavelet transforms, although they are highly complex mathematical tools, their application is simple and does not require extensive research into, or knowledge of, its mathematical background.

There have been several multiphase flow systems analyzed through this method, some of which are presented below.

- Li, Hui and Yuji Tomita, 2001. "Characterization of Pressure Fluctuation in Swirling Gas-Solid Two-Phase Flow in a Horizontal Pipe." Advanced Powder Technology 12 (2): 169-185.

The authors analyzed a swirling gas-solid two-phase flow under various conditions. The system differed from regular gas-solid flows in that the gas was given a swirling or spinning motion upon entering the pipe. The measurements taken were static wall pressure with a frequency of 800 Hz at three different locations, each data set consisting of 10 seconds. To analyze the data, the root mean square (RMS), power spectra distribution (PSD) using the fast Fourier transform (FFT) and the Morlet wavelet were used. The analysis consisted in comparing the PSD and wavelet analysis at three pipe axial locations (2, 4 and 6 m) and at different swirl numbers and velocities to try to identify exclusive trends and patterns in the measurements at each experimental condition.

Among their results, they identified general behaviors in the estimated values when applying the different tools, such as an increase or decrease in RMS attributed to an increase or decrease of velocities and swirl number. Also shown was the clear difference in spectral behaviour of the pressure measurements along the length of the pipe, for same experimental conditions. However, direct comparison of different experimental conditions (swirl number, velocity, etc) did not prove that useful. Although there were small differences observed between experiments at different air velocities, there was no attempt to construct a model to describe the suspension of the solids.

- Li, Hui, 2002. "Application of Wavelet Multi-Resolution Analysis to Pressure Fluctuations of Gas-Solid Two-Phase Flow in a Horizontal Pipe." Powder Technology 125 (1): 61.

This paper presented the same experimental setup as the previous one. The signal processing tools used were different for this paper, where the authors used the RMS, skewness and probability density function (PDF) applied to a Daubechies 4 wavelet. Most of the analysis focused on a time-frequency plane, decomposing the signal into its components at each frequency. Comparison of pressure measurements along the length of the pipe was again done.

The measured data were analyzed at different frequency bands, and a clear oscillation at low frequencies (< 1 Hz) was found for dune flow, and at higher frequencies (50 – 200 Hz) for suspended flow. However, a full spectral analysis with a PSD might have appropriately complemented the wavelet multi-resolution analysis, given it analyzes the entirety of the spectra. As in the previous paper, several differences were observed that could be attributed to an increase or decrease in velocity. However, there was not a clear trend for each flow rate as would be desired. From the plots and results that were shown, the tool that portrayed the most notable difference from one case study to another seemed to be the PDF, which showed small yet identifiable differences. Also, the data set that produced the clearest results was the pressure measurement farthest away from the pipe entrance. This was probably the distance required to produce fully developed flow.

- **Briens, Lauren A. and Naoko Ellis, 2005. "Hydrodynamics of Three-Phase Fluidized Bed Systems Examined by Statistical, Fractal, Chaos and Wavelet Analysis Methods." Chemical Engineering Science 60 (22): 6094-6106.**

The authors analyzed a three phase system composed of air, water and polypropylene or glass beads flowing in a vertical pipe. The orientation affected the possible flow regimes, given that a pure plug or slug flow is not as stable horizontally as it is vertically, as stated by the authors. Differential pressure and conductivity measurements were collected, both at 500 Hz. The use of a conductivity probe is theoretically more effective than a pressure sensor, given its instantaneous response and accurate detection of air/water bubbles because of the great difference in their electrical conductivities. The tools used for the analysis were statistical, fractal, and chaos analysis as well as the Daubechies order 5 wavelet and the PSD. The research objective was to study the minimum liquid fluidization velocity and the transition velocity between the coalesced and dispersed bubble flow regimes.

The identification of gas bubbles passing in front of the conductivity probe was possible because a sharp decrease in the conductivity measurement occurred each time a bubble passed. The span and magnitude of these step changes in conductivity provided an estimate of the size and velocity of the bubble for the cases where the solid dispersed phase was polypropylene beads. However, for the case with glass beads (higher

density), the signal was much more erratic, making such a distinction harder to identify. From statistical analysis of the data, the only results shown were those of the coefficient of variation (CV) applied to the conductivity probe. The CV measures the standard deviation as a percentage of the mean and was the most accurate way of identifying the transition velocities between flow regimes (for this specific case). From the spectral analysis, there were small differences between fixed bed and bubble flow regimes. However, it did not prove to be very effective at differentiating between the coalesced and dispersed bubble regimes, for which the plots were almost identical. The fractal and chaos analysis did not provide conclusive results. From the wavelet analysis, there did not seem to be a large difference in the coefficient values. However, the standard deviation of the wavelet coefficients proved efficient at observing the transition velocities between the bubbly flow regimes.

- **Chakrabarti, D. P., G. Das, and P. K. Das, 2007. "Identification of Stratified Liquid-Liquid Flow through Horizontal Pipes by a Non-Intrusive Optical Probe." Chemical Engineering Science 62 (7): 1861-1876.**

This paper presents the study of a two-component water-oil liquid-liquid system. The main objective was to obtain a classification technique that allowed flow regime recognition from data, using different mathematical tools. The different tools used were the PDF, the first four statistical moments (mean, standard deviation, skewness and kurtosis) and the Daubechies 4 wavelet multiresolution analysis. Measurements were obtained with an optical probe at 23 Hz for 2 min, and digital photographs which were used for flow classification through visual observation. Theoretical flow maps were not used to develop the flow recognition technique, although they were used to verify the accuracy of their classification technique.

The authors were able to differentiate between measurements at different conditions for low water velocities by direct comparison, but could not do so for higher water velocities where highly dispersed or bubbly flow regimes were predominant. Through further statistical analysis of the data, differences in all statistical moments were found for all regimes. For higher water velocities, although it was not as clear, the differences were still noticeable. The differences in the wavelet analysis did not appear as identifiable as in the statistical analysis, although the authors do mention certain

distinctions, and proceeded to establish a flow classifier. The parameters that were chosen for their flow map were the mean value of the response, the variance, the skewness and the fluctuation at the different levels of decomposition. The constructed flow map was compared to two (2) theoretical maps from literature. The agreement was good, although all boundaries seemed to be slightly offset. The flow regime classifier was not presented in the paper, nor was its method specified (mathematical model, probabilistic model, etc).

2.1.1 Impact on current investigation

Given the promising results described above, the tools selected for the present research are: the first four statistical moments (mean, standard deviation, skewness and kurtosis), the histogram, the Daubechies order 4 wavelet, and the power spectra distribution obtained through the Fourier transform. Also, given its simplicity and direct application to industry, the present study uses wall pressure measurements, at a frequency of 1 kHz.

A common problem in previous research was the difficulty of finding differences in flow regimes, which could be due to the fact that trend searches were done visually, where small variations may not have been detected. With a computerized search, better results might be found (Wold 1980). The present investigation uses pattern recognition algorithms to increase the possibility of successfully finding differences between flow regimes.

Finally, even though previous researchers have attempted to produce flow maps using signal processing (Chakrabarti et al. 2007), the modeling procedure has not been fully presented. To have a valid modeling procedure for the current study, partial least squares discriminant analysis is used to obtain a mathematical model for flow regime classification. To classify new data sets, a probability method is used, in the form of the naïve Bayes classifier.

To get a better understanding of how the selected tools work, the following sections explain the most relevant concepts required for the modeling procedure. Some examples from previous research are also presented.

2.2 Data pre-processing: digital signal processing

Digital signal processing consists of the mathematics, algorithms and techniques used to manipulate discrete-time representations of analog signals (Smith 2002). This manipulation of signals allows the extraction of underlying information contained within them. The tools used for this purpose are varied depending on the application; therefore, only those selected for the current modeling process, based on previous research, are explained below.

2.2.1 Variable centering and scaling

A common first step in signal processing for almost any application is to subtract the mean from each data set, a process referred to as "mean centering". This shifts the focus from the absolute value of the data to its variations from the mean, such that the latter becomes the important factor. This can be done by subtracting the mean of the entire data set, called global mean centering, or the data subset specific mean, called class mean centering. Mean centering makes signals of significantly different averages to be relatable in a similar order of magnitude (Kreft et al. 1995, Echambadi et al. 2007). However, the remaining values still carry a dimensionality which can be undesirable in some situations.

To eliminate the remaining dimensionality, scaling or normalization is used. Scaling consists of dividing all values by an arbitrary constant. This arbitrary constant is most commonly a value within the data set, such as the maximum, the standard deviation (normalization) or the mean of the data set. Scaling allows one to present oscillations as a dimensionless fraction of the arbitrary constant, irrespective of their original dimensionality and magnitude (Geladi et al. 1985).

2.2.2 Statistical moments and histogram

Statistical analysis is used for summarizing or describing a collection of data through numerical descriptors. In the current research, the histogram is used along with the first four statistical moments, which are: mean, standard deviation, skewness and excess kurtosis. The histogram, mean and standard deviation are commonly used and understood, and will therefore be excluded from this explanation.

The skewness is a measure of the symmetry of the distribution. Figure 2.1 presents some examples of distributions and their skewness. For a negative value, the bulk of the probability distribution will be on the right side of the mean; for a positive value, the bulk of the probability distribution will be on the left side. For a normal distribution, the skewness is zero, given both sides of the distribution have the same shape. The equation describing skewness can be written as (von Hippel 2005)

$$s = \frac{\frac{1}{n}\Sigma(x_i - \mu)^3}{\sigma^{3/2}}$$
(2.1)
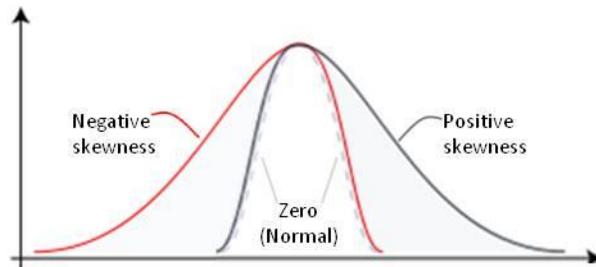
Figure 2.1. Skewness example for different curve shapes

The excess kurtosis measures the "peakedness" or flatness of the peak of the distribution, as well as the weight or persistence of the tails of a distribution. A positive valued excess kurtosis indicates a strong sharp peak and long tails; a negative value indicates a flatter peak with shorter tails. A normal distribution has an excess kurtosis of zero (Balanda et al. 1988). A graphic representation of several values of excess kurtosis can be observed in Figure 2.2. The equation for excess kurtosis is:

$$k = \frac{\frac{1}{n}\Sigma(x_i - \mu)^4}{\sigma^2} - 3$$
(2.2)
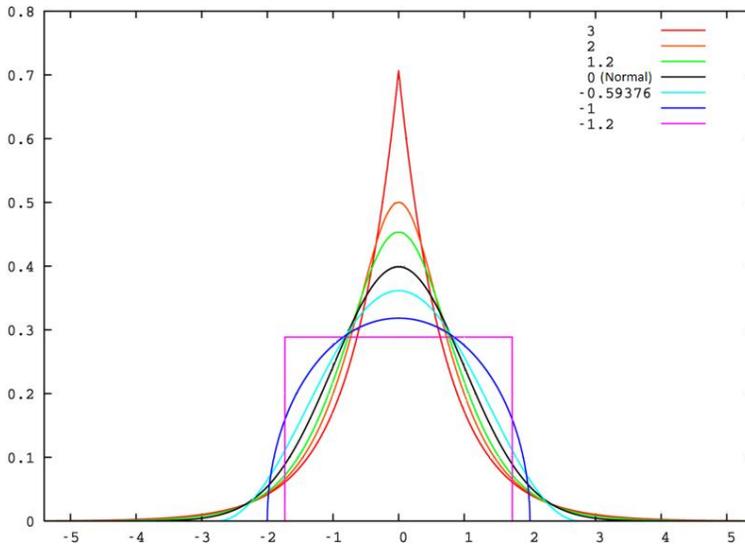
Figure 2.2. Kurtosis example for different curve shapes

Chakrabarti et al. (2006) used these tools to differentiate pressure measurements obtained for different liquid-liquid flow patterns. The histogram shape was significantly different for different flow regimes, as were the statistical moments, allowing for a clear distinction if classification is attempted.
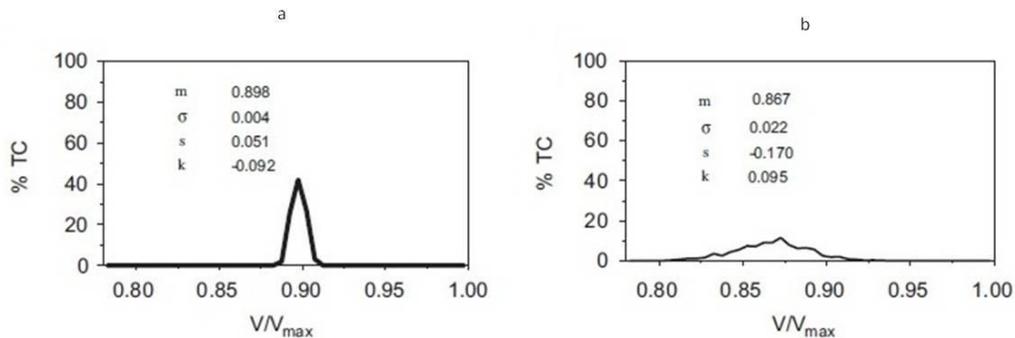
Figure 2.3 Histogram and statistical moments of an optical probe measurement for a liquid-liquid flow in: (a) stratified flow, (b) dispersed bubble flow (Chakrabarti et al. 2006)

2.2.3 Discrete Fourier Transform

Most signals can be approximated as a sum of sinusoidal functions. The Fourier transform does this, which allows one to look at the frequency information instead of at the time information contained within a signal. This is sometimes necessary as the spectral domain will portray the behaviour of periodic signals better than the temporal

domain. For discrete time signals, such as sampled and collected data, the discrete Fourier transform (DFT) is used. However, given the DFT requires N^2 calculations, the fast Fourier transform (FFT) is used instead, which only requires N·log(N) operations. The specific algorithm used for the current research is the MATLAB command "fft", which uses a modified Cooley-Tukey algorithm (Weeks 2010).

2.2.4 Power Spectrum

The power spectrum is a graph portraying the frequencies that comprise a signal, by plotting the amplitude or strength at each frequency. It is used to intuitively determine the important frequencies that comprise a signal. When determining the dominant frequencies, it is important to obey the Nyquist sampling theorem, which states that the sampling frequency should be at least double the bandwidth of the system ($f_s \geq 2B$). If the bandwidth is unknown prior to conducting the sampling and analysis of the signal, then only frequencies up to one half of the sampling frequency should be considered (Priestly 1982).

Briens et al. (2005) used the power spectrum to differentiate among different three phase gas-liquid-solid flow patterns, as shown in Figure 2.4. The lowest frequencies were observed for a flow with a fixed bed, and the highest frequencies were observed for the dispersed regime.
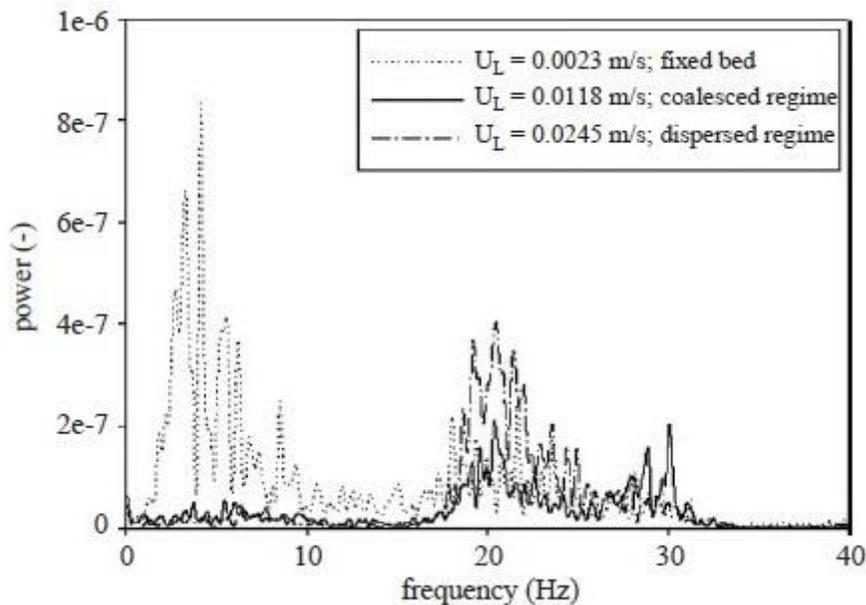
Figure 2.4 Power spectra profile from pressure measurements in an air-water-polypropylene fluidized bed (Briens et al. 2005).

2.2.5 Discrete Wavelet Transform

The wavelet transform is similar to the Fourier transform; however, the basis function is not a sinusoid, but rather a specially formulated function. These functions differ from the sine function in that they are designed to retain temporal as well as spectral information (Graps 1995). For this purpose, a Wavelet basis function has an initial amplitude of zero, reaches a maximum, and decays quickly to zero, contrary to sine functions which persist indefinitely. This creates an advantage when compared to the Fourier transform in that time localized peaks may be described by these short impulse functions, something impossible with a sine function (Weeks 2010).

The wavelet operation decomposes the signal in two, with each being half the original length. The two decompositions are the low frequency approximation coefficients of the original signal (original signal stripped of high frequency components), and the high frequency detail coefficients that were extracted from the approximation. This is achieved by filtering the data with a low pass filter and a high pass filter, and later down-sampling. The cut-off frequencies for the low pass and high pass filters are not fixed, but rather a fraction of the Nyquist frequency. If desired, a signal can be decomposed more than once by applying the wavelet transform to the approximation, dividing each time the effective bandwidth in half, given there are half as many samples. This subsequent decomposition is called the level. Figure 2.5 shows the wavelet decomposition process for three levels.
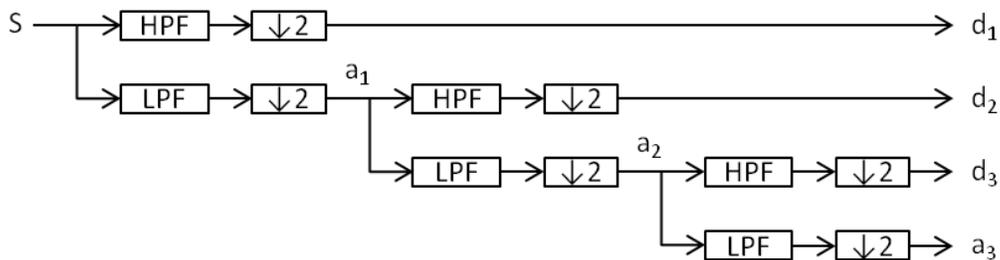
Figure 2.5. Schematic of the wavelet decomposition of a signal

There are many different Wavelet basis functions, called mother wavelets, as designed by various authors. The mother wavelet that has been chosen for the present study is

the Daubechies wavelet transform with 8 coefficients (Daubechies order 4 as stated by MATLAB). The high and low pass filters for this wavelet are:

$$LPF=[-0.0106\ 0.0329\ 0.0308\ -0.1870\ -0.0280\ 0.6309\ 0.7148\ 0.2304] \qquad (2.3)$$

$$HPF=[-0.2304\ 0.7148\ -0.6309\ -0.0280\ 0.1870\ 0.0308\ -0.0329\ -0.0106] \qquad (2.4)$$

The following two equations are used to calculate each detail coefficient and each approximation coefficient through the discrete wavelet transform, for the first level of decomposition. For subsequent decomposition (level 2 and above), the equations remain the same, changing the original signal for the previous level's approximation coefficients, as can be seen in Equations 2.5 and 2.6:

$$d_1(j) = \sum_{m=1}^{8} S(j + 1 - m) \cdot LPF(m) \qquad (2.5)$$

$$a_1(j) = \sum_{m=1}^{8} S(j + 1 - m) \cdot HPF(m) \qquad (2.6)$$

The wavelet transform is one of the most commonly used signal processing tools in flow regime recognition. Jana et al. (2005) decomposed a conductivity probe signal into 6 different frequency bandwidths, as shown in Figure 2.6. As the signal is subsequently decomposed (higher d_i), the remaining portions are the lower frequency components, as seen in the smoothness of the curves in d_4, d_5 and a_5 (approximation after 5 levels of decomposition)
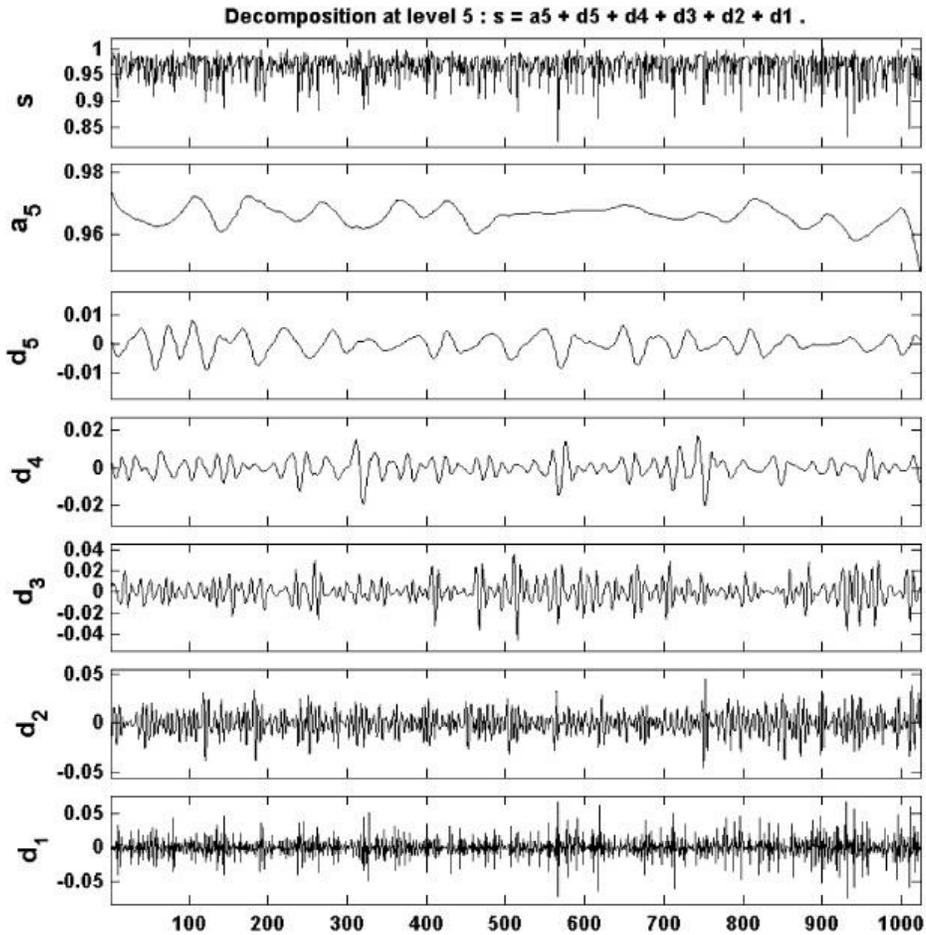
Figure 2.6 Wavelet analysis of a typical signal obtained from conductivity probe (Jana et al. 2005).

2.3 Modeling

The tools presented in the previous section allow for the extraction of the time, frequency and time-frequency information from signals, but do not classify flow rates. To obtain a classification modeling procedure, additional tools are necessary, which are presented below.

2.3.1 Autocorrelation test

The autocorrelation test detects similarities between a signal and itself when shifted in time. Mathematically it is the expected value of a signal multiplied by a time shifted version of itself. For stationary processes where the time does not affect the mean and variance of the system, the following equation may be used (Weeks 2010):

24

$$R(\tau) = \frac{E[(S_t - \mu)(S_{t+\tau} - \mu)]}{\sigma^2} \qquad (2.7)$$

If for any time shift $\tau \neq 0$ the value of R is different than zero, then there are repetitive patterns in the signal, such as those in periodic waves.

Before attempting to model the system, it is necessary to select a time window during which the system will be analyzed. That is, the time window that will be decomposed through the various tools and used as a single data point in the modeling process. This time window must contain enough information to represent the system, such that repetitiveness in all samples is ensured. The autocorrelation test helps in finding the best time window possible for each system. Knowing the time window to be used, the complete experimental data sets can be divided into smaller subsets and used as several points for the modeling, allowing for a better fit.

2.3.2   Pattern Recognition

The modeling is done through pattern recognition, in which the computer is trained to classify certain data sets into different groups of objects, as defined by the user. This type of modeling, called supervised learning, requires the use of two data sets: a training set and a validation set. The training set is used during the modeling step and its classification is input into the model. The validation set, also of known classification, is used on the finished model to verify that the classification is working properly (Theodoridis et al. 2010).

This type of pattern recognition has proven to be very effective when the relevant features of a certain group are used to classify it, and very inaccurate when the wrong features are used. As an example, if a blue square and a red circle were to be classified into groups named "Rectangle" and "Ovals", the feature to use would have to be their shape, not their color.

The specific variables taken into account are extremely important in the current study, given there are many pre-processing techniques being used. Of all the generated variables, most are probably not influential in separating flow patterns. Additionally, some of the variables may be correlated between them (called multicollinearity), negatively affecting the accuracy of the fit (Geladi et al. 1985).

2.3.3 Partial least squares discriminant analysis

To work around the multicollinearity and excess of variables, the modeling is done through partial least squares discriminant analysis (PLSDA). The PLS technique works by finding an alternate dimensional space that describes the highest variability for a set of observations and responses, similar to principal component analysis (Indahl et al. 2007). This reduces collinearity within variables by creating new latent variables that group collinear information (Mateos-Aparicio 2011). The main difference between PLS and PCA is that the latter only extracts the important features of an independent variable, whereas PLS does so for the dependent variable as well (Yeniay et al. 2002).

The end result of applying PLS is the regression model shown in Equation 2.8. This type of model is simple to derive and to apply, and its predictive power is comparable to more complex methods, if the described system is linear (Garthwaite 1994), i.e.

$$Y = X\beta + E \tag{2.8}$$

where both X and Y should be mean centered and scaled. The output of PLS regression, as is the case for other regression methods, is the regression parameters in β, which allow prediction of Y for new observations (Svante et al. 2001).

Both the dependent and independent variables (X and Y) are decomposed simultaneously to estimate the latent variables (Rosipal 2006). That is, latent variables are chosen such that underlying features from both X and Y are extracted, in a way that the inner relationship between them is considered. The following system of equations describes the previous statement (Jørgensen et al. 2007):

$$X = TP^T + E \tag{2.9}$$

$$Y = UQ^T + G \tag{2.10}$$

$$U = TC + H \tag{2.11}$$

$$\hat{\beta} = W(P^T)^{-1}\hat{C}Q^T \tag{2.12}$$

The algorithm chosen for calculating the regression parameters is the PLS2 (Jørgensen et al. 2007), a modified NIPALS algorithm (Noonan et al. 1977), which calculates the inner relationships between the latent variables in an iterative way. A typical procedure can be written as follows:

1. Initialize X_j and Y_j for the first latent variable as X and Y, respectively.

2. Initialize u_j as an arbitrary column of Y_j.

3. $w_j = \dfrac{X_j^T u_j}{\left\| X_j^T u_j \right\|}$ (2.13)

4. $t_j = X_j w_j$ (2.14)

5. $q_j = \dfrac{Y_j^T t_j}{\left\| Y_j^T t_j \right\|}$ (2.15)

6. $u_{j+1} = Y_j t_j$ (2.16)

7. If $\left\| u_{j+1} - u_j \right\|$ < established threshold (e.g. 1x10^{-8}), continue to step 8. Otherwise, return to Step 3.

8. $\hat{c}_j = \dfrac{t_j^T u_j}{t_j^T t_j}$ (2.17)

9. $p_j = \dfrac{X_j^T t_j}{t_j^T t_j}$ (2.18)

10. Calculate X and Y residual matrices $X_{j+1} = X_j - t_j p_j^T$ (2.19)

    and $Y_{j+1} = Y_j - \hat{c}_j t_j q_j^T$ (2.20)

11. Repeat steps 2-10 until reaching the desired number of latent variables.

Since there is no measured output in a classification problem, an output matrix must be generated in order to obtain the regression parameters. In PLSDA, this output matrix is a "dummy matrix" with m columns, each representing a class, and uses numerical values to represent the discrete event of belonging or not belonging to a class (Indahl et al. 2007). For a two class system, the dummy matrix has two columns. To symbolize a point as belonging to a class, the number one is used, and for not belonging, minus one is used. As an example consider the following dummy matrix:

$$Y_1 = \begin{bmatrix} 1 & -1 \\ -1 & 1 \\ -1 & 1 \end{bmatrix}$$

The first row, representing the first data point, belongs in group One, as denoted by the "1" in the first column, and does not belong to group Two, as denoted by the "-1". The second and third rows belong to group Two and not to group One (Rosipal et al. 2006). Although there are cases in which a data point can belong to multiple groups or to no groups at all, the present study considers the case where each data point must belong to a class and can only belong to a single class.

27

By using PLS, it is possible to extract the most influential parameters in large data sets, while getting rid of most of the irrelevant information, allowing for the development of a better model. This is done by selecting the optimum number of latent variables (Kurozumi et al. 2011). Here, the term "optimum" is used freely as it will depend on the criterion used for selecting the optimum, which is explained in detail ahead.

2.3.4 Occam's Razor

With any principal component style regression, an important issue to consider is how many principal components or latent variables to use. For PLS it is possible to use any number of latent variables up to the rank of the observation matrix. However, more than a third of the rank are considered too many due to over fitting problems (Geladi et al. 1985).

A simple criterion that is commonly used in selecting the number of latent variables is to ensure that a high percentage of variation (> 90 - 95%) is accounted for in both the dependent and independent variables. However, this is usually not very effective since an adequate description can be achieved with just a few latent variables, while the optimum for prediction purposes is usually achieved well past that threshold.

As multiple sources in literature have discussed (e.g. Geladi et al. 1985, Yeniay et al. 2002), there is a trade-off in choosing the number of latent variables: a higher number gives the model a better fit at the risk of overfitting the model to the training data. This trade-off must be analyzed carefully to select a model that will be equally accurate on most data sets. The selection criterion is commonly called the Parsimony Principle or Occam's razor, which involves choosing the best model that makes the fewest new assumptions (Zhang 2004).

2.3.5 Bayesian Information Criterion

There are many criteria that quantify the trade-off of accuracy vs. overfitting, giving the selection process a probabilistic significance. In this case, the Bayesian Information Criterion (BIC) is chosen since it penalizes quite heavily the growing number of latent variables used when compared to other methods (Kuncheva 2006). In PLS regression, it is desired that the chosen number of components be small, making the BIC an adequate method. The simplified BIC equation can be written as (Spiegalhalter et al. 2002)

$$BIC = n \cdot \ln(\hat{\sigma}^2) + k \cdot \ln(n) \tag{2.21}$$

As can be seen, the BIC value increases with both the error term and the number of latent variables, leading to a model selection in which trade-off arising from both terms is considered. The number of latent variables selected corresponds with the minimum BIC value.

For the error term ($\hat{\sigma}^2$), the predicted residual sum of squares (PRESS) is used, which is the square of the estimation error. The matrix form of the equation is as follows:

$$PRESS = (Y - X \cdot \hat{\beta})^T (Y - X \cdot \hat{\beta}) \tag{2.22}$$

2.3.6 Cross-validation

A model must first be built and then tested through a validation test to obtain the value for the PRESS (Geladi et al. 1985). For this, an 8-fold cross-validation is used (Yeniay et al. 2002). This type of cross-validation consists of separating the entire data set into eight smaller subsets, where each data point will belong to only 1 subset, and all data sets will have the same number of points (or at least approximately equal numbers). With the data separated into subsets, a model is constructed using seven of the subsets, and later validated on the other subset, calculating its PRESS. This procedure is repeated until all eight subsets have been used for validation exactly once. The total PRESS for the model will be the sum of each of the individual values of PRESS. This procedure is computationally time consuming given its repetitive nature, but allows for selection of the best model available.

2.4 Classification: Bayes decision theory

Classification of new data sets is achieved by multiplying new observations matrices by the regression parameters β, to obtain the classifier matrix Y. Values that are close to 1 in a given column of the estimated classifier indicate belonging to that class, where values close to -1 indicate not belonging. However, values that lie far away from the calibration values, especially those close to the middle point of zero, are especially hard to classify.

One way to avoid problems in classifying is by using a probabilistic method to decide which class to assign to new data sets (Schwarz 1978). For this, a naive Bayes classifier is

used, for a single-variable system, of equal prior probability (Zhang 2004). This is a simple classification process based only on the individual probability that the data belongs to a particular class. The highest probability of belonging among all classes is considered the correct classification decision (Indahl et al. 2007).

Since only one probability is required for classification, and given that the probability function is directly proportional to the Z-score, it is not necessary to calculate the actual probability, just the Z-score. The classification decision determines in which class the data point belongs by assigning it the class with the lowest z-score (highest probability), where

$$Z_{i,j} = \frac{\hat{Y}_i - \mu_j}{\sigma_j} \tag{2.23}$$

The mean and standard deviation of the classes are calculated using the regression model on the training data. Although the model is biased towards the training data due to calibration, it is expected that the fit will not be perfect, and a non-zero variance will occur.

References

Balanda, K. P., and Macgillivray, H. L. (1988). "Kurtosis: a critical review." *The American Statistician,* Vol. 42(No. 2), 111-119.

Briens, L. A., and Ellis, N. (2005). "Hydrodynamics of three-phase fluidized bed systems examined by statistical, fractal, chaos and wavelet analysis methods." Chemical Engineering Science, 60(22), 6094-6106.

Chakrabarti, D. P., Das, G., and Das, P. K. (2007). "Identification of stratified liquid-liquid flow through horizontal pipes by a non-intrusive optical probe." Chemical Engineering Science, 62(7), 1861-1876.

Chakrabarti, D. P., Das, G., and Das, P. K. (2006). "The transition from water continuous to oil continuous flow pattern." AICHE J., 52(11), 3668-3678.

Echambadi, R., and Hess, J. D. (2007). "Mean-Centering Does Not Alleviate Collinearity Problems in Moderated Multiple Regression Models." Marketing Science, 26(3), 438-445.

Garthwaite, P. H. (1994). "An Interpretation of Partial Least Squares." Journal of the American Statistical Association, 89(425), 122-127.

Geladi, P., and Kowalski, B. R. (1986). "Partial least-squares regression: a tutorial." Anal.Chim.Acta, 185(0), 1-17.

Gerlach, R. W., Kowalski, B. R., and Wold, H. O. A. (1979). "Partial least-squares path modelling with latent variables." Anal.Chim.Acta, 112(4), 417-421.

Graps, A. (1995). "An introduction to wavelets." Computational Science & Engineering, IEEE, 2(2), 50-61.

Haenlein, M., and Kaplan, A. M. (2004). "A Beginner's Guide to Partial Least Squares Analysis." Understanding Statistics, 3(4), 283-297.

Indahl, U. G., Martens, H., and Naes, T. (2007). "From dummy regression to prior probabilities in PLS-DA." J.Chemometrics, 21(12), 529-536.

Jana, A. K., Das, G., and Das, P. K. (2006). "Flow regime identification of two-phase liquid–liquid upflow through vertical pipe." Chemical Engineering Science, 61(5), 1500-1515.

B. Jørgensen, and Goegebeur, Y. (2007). "ST02: Multivariate Data Analysis and Chemometrics." http://statmaster.sdu.dk/courses/ST02/ (December 5, 2010).

Kreft, I. G. G., de Leeuw, J., and Aiken, L. S. (1995). "The Effect of Different Forms of Centering in Hierarchical Linear Models." Multivariate Behavioral Research, 30(1), 1-21.

Kuncheva, L. I. (2006). "On the optimality of Naive Bayes with dependent binary features." Pattern Recog.Lett., 27(7), 830-837.

Kurozumi, E., and Tuvaandorj, P. (2011). "Model selection criteria in multivariate models with multiple structural changes." J.Econ., 164(2), 218-238.

Li, H. (2002). "Application of wavelet multi-resolution analysis to pressure fluctuations of gas-solid two-phase flow in a horizontal pipe." Powder Technol, 125(1), 61.

Li, H., and Tomita, Y. (2001). "Characterization of pressure fluctuation in swirling gas-solid two-phase flow in a horizontal pipe." Advanced Powder Technology, 12(2), 169-185.

Mateos-Aparicio, G. (2011). "Partial Least Squares (PLS) Methods: Origins, Evolution, and Application to Social Sciences." Communications in Statistics - Theory and Methods, 40(13), 2305-2317.

Noonan, R., and Wold, H. (1977). "NIPALS Path Modelling with Latent Variables." Scandinavian Journal of Educational Research, 21(1), 33-61.

Orell, A. (2007). "The effect of gas injection on the hydraulic transport of slurries in horizontal pipes." Chemical Engineering Science, 62(23), 6659-6676.

Priestly, M. B. (1982). "Spectral Analysis and Time Series." J.Forecast., 1(4), 422-423.

Rosipal, R., and Kramer, N. (2006). "Overview and Recent Advances in Partial Least Squares; Subspace, Latent Structure and Feature Selection." C. Saunders, M. Grobelnik, S. Gunn, and J. Shawe-Taylor, eds., Springer Berlin / Heidelberg, 34-51.

Schwarz, G. (1978). "Estimating the Dimension of a Model." Annals of Statistics, 6(2), 461-464.

Smith, S. (2002). Digital Signal Processing: A Practical Guide for Engineers and Scientists. Butterworth-Heinemann.

Spiegelhalter, D. J., Best, N. G., Carlin, B. P., and Van, D. L. (2002). "Bayesian measures of model complexity and fit." Journal of the Royal Statistical Society: Series B (Statistical Methodology), 64(4), 583-639.

Taitel, Y., and Dukler, A. E. (1976). "A model for predicting flow regime transitions in horizontal and near horizontal gas-liquid flow." AICHE J., 22(1), 47-55.

Theodoridis, S., Pikrakis, A., Koutroumbas, K., and Cavouras, D. (2010). *Introduction to Pattern Recognition: A Matlab Approach*. Academic Press.

von Hippel, P. T. (2005). "Mean, median, and skew: Correcting a textbook rule." Journal of Statistics Education, 31(Number 2).

Weeks, M. (2010). *Digital Signal Processing Using MATLAB & Wavelets*. Jones & Bartlett Learning.

Wold, H. "Soft Modeling by Latent Variables; the Nonlinear Iterative Partial Least Squares Approach." Perspectives in Probability and Statistics.Papers in Honour of M.S.Bartlett.

Wold, H. (1980). "Model Construction and Evaluation When Theoretical Knowledge Is Scarce." Evaluation of Econometric Models, National Bureau of Economic Research, Inc, 47-74.

Wold, S., Sjöström, M., and Eriksson, L. (2001). "PLS-regression: a basic tool of chemometrics." Chemometrics Intellig.Lab.Syst., 58(2), 109-130.

Yeniay, O., and Goktas, A. (2002). "A comparison of partial least squares regression with other prediction methods." Hacettepe Journal of Mathematics and Statistics, 31 99-111.

Zhang, H. (2004). "The Optimality of Naive Bayes." FLAIRS Conference.
http://courses.ischool.berkeley.edu/i290-dm/s11/SECURE/Optimality_of_Naive_Bayes.pdf

CHAPTER 3: MODELLING PROCEDURE

This chapter presents the detailed procedure used at the different modelling steps: data pre-processing, modeling, model selection, and data classification.

3.1 Data pre-processing

The following procedure was used to prepare the pressure measurements before fitting them to a model.

3.1.1   Time windowing

As was mentioned in Section 2.3.1, a prerequisite to modeling was to select a time window during which the system was to be analyzed. There is a trade-off in selecting this interval: if the time window is too large, repetitiveness is ensured, but a time delay is inserted into the detection rate of the pipeline's current state. If the time window is too small, complete repetitiveness might not be ensured, but detection rate is fast (Mallat 2009). Examples of these two scenarios are as follows:

- If the time interval is large, a single gas bubble acting as a slug might go undetected;
- On the other hand, a small time window would detect the single gas bubble, but might cause slight changes in flow conditions to be detected as a different flow regime, given the model does not consider enough information to accurately describe the system.

An autocorrelation test was performed to obtain a reasonable estimate of an appropriate time window (Weeks 2010). If an autocorrelation test was inconclusive, a time window was chosen based on the autocorrelation test and the visual trend of the measurements, while keeping the aforementioned trade-off in mind.

3.1.2   Mean centering and scaling

From previous research (Li et al. 2001, Chakrabarti et al. 2006, 2007) it was clear that differences in the measurements were clearly dependant on the magnitude of both the average pressure and the oscillations. However, if a universal or multisystem model was to be built, it had to maintain the dimensionless nature of dimensional analysis.

Class mean centering was performed to remove the importance of the average pressure of the measurements (Kreft et al. 1995). To eliminate the remaining dimensionality of

the measurements, scaling was performed, also using the class mean. The final values were expected to be in the order of 10^{-1} to 10^{-2}, and were therefore multiplied by 100 to avoid rounding errors. This process transformed the original measurements into a percentage of variation with respect to the class mean.

3.1.3 Statistics

Once the pressure measurements were transformed to percentage variations (referred to, from this point forward, as "the data"), the four statistical moments were calculated to characterize the distribution through numerical values. To complement the four statistical moments, the histogram was also used. Although all measurements were mean centered and scaled, the span of the data (minimum to maximum value) was significantly different, making it difficult to compare data sets. The span of the data was therefore normalized to represent a total span of one, and 15 bins were constructed within this range. The values compared were those within the same numbered bin out of the 15. Since this information was later modeled using PLS, where the scale of the components affects their weight on the latent variables, values had to be scaled. Consequently, the histogram was calculated using the relative frequencies instead of the absolute, giving each bin a theoretical maximum value of one. Figure 3.1 shows a graphical representation of this process.
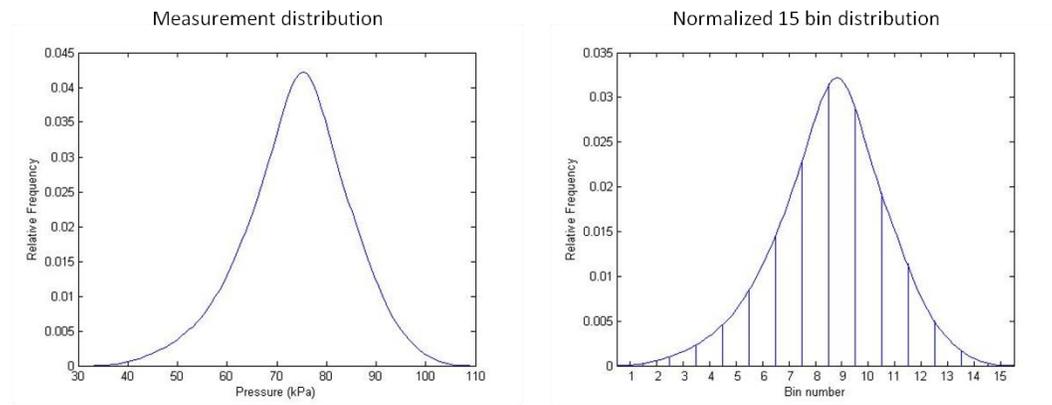
Figure 3.1. Comparison of the sample distribution of a measurement and it's normalized version using 15 bins.

3.1.4   Wavelet Transform

Following the statistical analysis, the data were decomposed using the Daubechies order 4 wavelet transform, for six levels of decomposition (Graps 1995). Previous researchers (Chakrabarti et al. 2006, 2007, Suman 2010) had used only five levels of decomposition. However, their sampling rate was lower than the 1 kHz used in this research, for which five levels of decomposition was considered insufficient to extract all the relevant information.

The decomposition yielded six detail coefficient vectors and one approximation coefficients vector. Previous research (Li 2002, Jana et al. 2006) had shown that the actual value of these was not the important feature, but rather their statistical moments, which were compared at each level. Also, the absolute value of each of the statistical moments was never discussed in the literature, but rather the comparison of each moment from one level to another, i.e. which level had the highest or lowest value for a given statistical moment. This indicated that the trend of the statistical moments was the important feature. To exploit this, all statistical moments of the six detail vectors and the approximation vector were normalized with respect to the highest of each statistical moment. This normalization left only the relative behaviour of the different levels. Also, the normalization positively affected the PLS regression since all values were on the same scale as the histogram (maximum of 1) (Wold et al. 2001, Yeniay et al. 2002).

3.1.5   Power Spectra Distribution

For the final pre-processing technique, each data set was analyzed with the Fourier transform to characterize its spectral components. The sampling rate of 1 kHz allowed a maximum detection frequency of 500 Hz, in accordance with the Nyquist sampling theorem (see Section 2.2.4) (Priestly 1982). However, each system was expected to have its own characteristic spectrum, which probably did not require the entire (0 – 500) Hz span. For each specific experiment, the spectrum was analyzed and the appropriate bandwidth was used. To simplify the system, the spectrum was smoothed out by only using the maximum amplitude at each 1 Hz increment (e.g. maximum amplitude within 0 to 1 Hz, 1 to 2 Hz, etc).

3.2 Modeling

The following section presents the procedures used for preparation of the dependent and independent variables for the PLS regression.

3.2.1   Independent variable (X): observations

Given the selected model (Section 2.3.3), the information portrayed in Section 3.1 had to be arranged in vector form so that it could be used in the regression model. The following describes how the different pre-processing techniques were input into the observations matrix:

- Components 1-3: The standard deviation, skewness and kurtosis of the data. The mean was omitted because the data were pre-processed using mean centering.
- Components 4-27: The four statistical moments of the detail coefficients for each level and of the level six approximation coefficients, normalized by the highest of each statistical moment.
- Components 28-42: The histogram calculated to 15 bins, using the relative frequencies.
- Components 43-N: The power spectra distribution plot divided into bins of 1 Hz each, for the required bandwidth, using the maximum amplitude within each bin.

The final number of variables (N) was 43 plus the effective bandwidth of the system expressed in Hertz. Each row of the observation matrix represented a single data point for the model, corresponding to a measurement interval lasting the selected time window.

As mentioned in Section 2.3.3, the matrices involved in Equation 4.8 had to be mean centered. This was achieved by subtracting the mean of each variable (column). In order to maintain consistency, the means were saved for use in subsequent steps, such as new data classification.

3.2.2   Dependent variable (Y): classifier

For the output matrix Y, a matrix was constructed with the same number of rows as the observations matrix, and as many columns as the number of classes (Indahl et al. 2007).

Each data point (row) was classified into only one of the possible classes, as was previously explained in Section 2.3.3.

3.3 Model selection

With the dependent and independent variables ready, a model could be built using any number of latent variables up to the rank of the observations matrix. However, since the optimum was desired, cross-validation was used, in combination with the BIC (Kuncheva 2006).

The following procedure was carried out to select the optimum model:

1. Divide the observations and classification matrices into 8 subsets, maintaining the appropriate X_i to Y_i pairing.
2. Select one subset to be used for validation.
3. Choose a number of latent variables (start at one latent variable and increase by one for subsequent modeling attempts).
4. Obtain a model through PLS regression.
5. Determine the PRESS on the validation subset.
6. Return to step 3 until the maximum number of latent variables is reached.
7. Return to step 2 until all data sets have been used for validation once and only once.
8. Sum the PRESS values for each number of latent variables.
9. Calculate the BIC for each number of latent variables.
10. Select the model with the lowest BIC value.

Having selected a model, the important parameters going forward were the regression parameters found in matrix β, which were used in subsequent estimation of the classifier.

3.3.1 Model parameters

As was explained in Section 2.4, the mean and standard deviation of the classifier for each class is necessary for estimation of the z-score for new data sets. With large amounts of data, it is recommended that the initial data be divided into three subsets: the training and validation data sets, as mentioned in Section 2.3.3, as well as a third set for population characterization. This characterization set is used to obtain numerous

estimations of the classifier for each class, and subsequently calculate its mean and standard deviation. This is the ideal case, given the model would not be biased towards this characterization data set, as it is towards the training set. However, this is only possible if large amounts of data are available, which was not the case for the present investigation. In the present case, the mean and standard deviation were calculated from the modeling data, which, as was explained in Section 2.4, should not be a perfect fit.

3.4 Classification

Once a model had been built, it could be applied to any other data set for classification purposes. New pressure measurements to be classified were treated the same preprocessing techniques to obtain an observations matrix, including mean centering with the same values that were used for the modeling step. Once the observation matrix was constructed, the classifier was calculated using Equation 4.8. This classifier value still required decision making to assign a class to it, so it was input into Equation 4.23 to calculate the z-score. Once the z-score had been calculated for all classes, the class with the lowest z-score was assigned as the correct classification of the data set.

References

Chakrabarti, D. P., Das, G., and Das, P. K. (2007). "Identification of stratified liquid-liquid flow through horizontal pipes by a non-intrusive optical probe." Chemical Engineering Science, 62(7), 1861-1876.

Chakrabarti, D. P., Das, G., and Das, P. K. (2006). "The transition from water continuous to oil continuous flow pattern." AICHE J., 52(11), 3668-3678.

Graps, A. (1995). "An introduction to wavelets." Computational Science & Engineering, IEEE, 2(2), 50-61.

Indahl, U. G., Martens, H., and Naes, T. (2007). "From dummy regression to prior probabilities in PLS-DA." J.Chemometrics, 21(12), 529-536.

Jana, A. K., Das, G., and Das, P. K. (2006). "Flow regime identification of two-phase liquid–liquid upflow through vertical pipe." Chemical Engineering Science, 61(5), 1500-1515.

Kreft, I. G. G., de Leeuw, J., and Aiken, L. S. (1995). "The Effect of Different Forms of Centering in Hierarchical Linear Models." Multivariate Behavioral Research, 30(1), 1-21.

Kuncheva, L. I. (2006). "On the optimality of Naive Bayes with dependent binary features." Pattern Recog.Lett., 27(7), 830-837.

Li, H. (2002). "Application of wavelet multi-resolution analysis to pressure fluctuations of gas-solid two-phase flow in a horizontal pipe." Powder Technol, 125(1), 61.

Li, H., and Tomita, Y. (2001). "Characterization of pressure fluctuation in swirling gas-solid two-phase flow in a horizontal pipe." Advanced Powder Technology, 12(2), 169-185.

Mallat, S. G. (2009). *A Wavelet Tour of Signal Processing: The Sparse Way*. Elsevier /Academic Press.

Suman, Y. K., Mandal, T. K., and Das, G. (2010). "Use of Digital Signal Analysis to Identify Slug Flow in a Narrow Vertical Pipe." Chem.Eng.Commun., 197(10), 1287-1302.

Weeks, M. (2010). *Digital Signal Processing Using MATLAB & Wavelets*. Jones & Bartlett Learning.

Wold, S., Sjöström, M., and Eriksson, L. (2001). "PLS-regression: a basic tool of chemometrics." Chemometrics Intellig.Lab.Syst., 58(2), 109-130.

Yeniay, O., and Goktas, A. (2002). "A comparison of partial least squares regression with other prediction methods." Hacettepe Journal of Mathematics and Statistics, 31 99-111.

CHAPTER 4: GAS-LIQUID SPRAY STABILITY

This chapter presents the overview of an experimental setup used to study the stability of a gas-liquid spray, as well as a novel method for detecting the stability using the modeling procedure described in Chapter 3.

4.1 Gas-liquid sprays

A gas-liquid spray is a colloid suspension, commonly referred to as aerosol, of liquid droplets in a gas. It is a type of dispersed two-phase flow, made up of a gaseous continuous phase and a discrete liquid phase (droplets).

4.1.1 Stability of a gas-liquid spray

A gas-liquid spray is considered stable if the flow of the mixture through the nozzle has constant composition in time, and unstable if it has a transient behaviour (Rahman et al. 2012). The most common instability in gas-liquid sprays is a pulsating and intermittent atomization. This type of instability is characterized by the ejection of liquid streams (non-atomized), followed by gas streams with very little liquid. When operating gas-liquid sprays, unstable flows are undesired and should be avoided.

Several authors (Hulet et al. 2003, Ariyapadi et al. 2005, Maldonado et al. 2008) have studied the stability of gas-liquid sprays, concluding that a dispersed bubbly flow upstream of the nozzle will create stable sprays, whereas intermittent flow will create unstable sprays, as shown in Figure 4.1. It has been stated that the gas-to-liquid ratio (GLR) does not dictate the stability of a spray, except when studying a constant pressure. For systems of constant pressure (or nearly so), a low GLR will guarantee a stable spray, and a higher GLR will result in unstable sprays (Maldonado et al. 2008). The transition from stable to unstable spray for the current study is set at $1 - 1.5\%$ GLR. That is, for a GLR lower than 1%, the spray is stable; for a GLR higher than 1.5%, the flow is unstable. This criterion is valid only for the range of pressures used in this research.
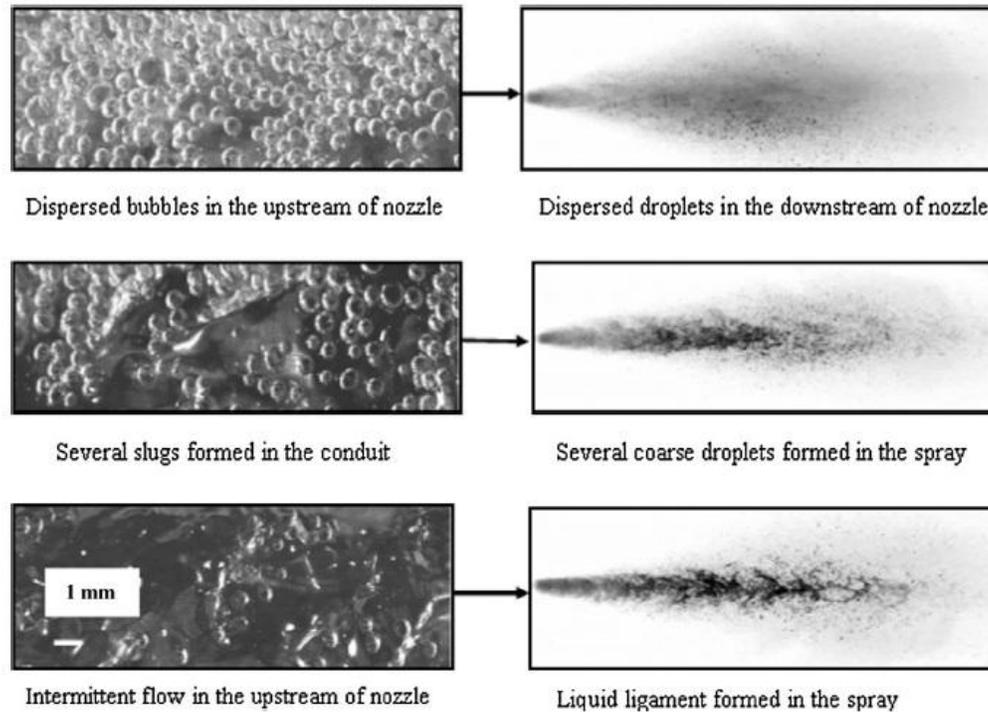
Figure 4.1. Illustration of the well-atomized and poorly-atomized spray as consequences of the upstream bubble size distribution and flow patterns observed in the quarter scale nozzle and obtained by the high speed video shadowgraphy (Rahman et al. 2012).

4.2 Experimental set-up

A schematic of the experimental set-up used by Rahman et al. (2012) to study the stability of a gas-liquid spray is shown here as Figure 4.2. It consisted of an air compressor, which drew in air at ambient temperature, and a water pump, which drew water from a storage tank. Both streams (air and water) were fed into a conduit of total length 36.8 cm and internal diameter (i.d.) of 6.35 mm, and exited through a 3.1 mm (i.d.) nozzle. Pressure in the system was controlled by the output of the compressor.
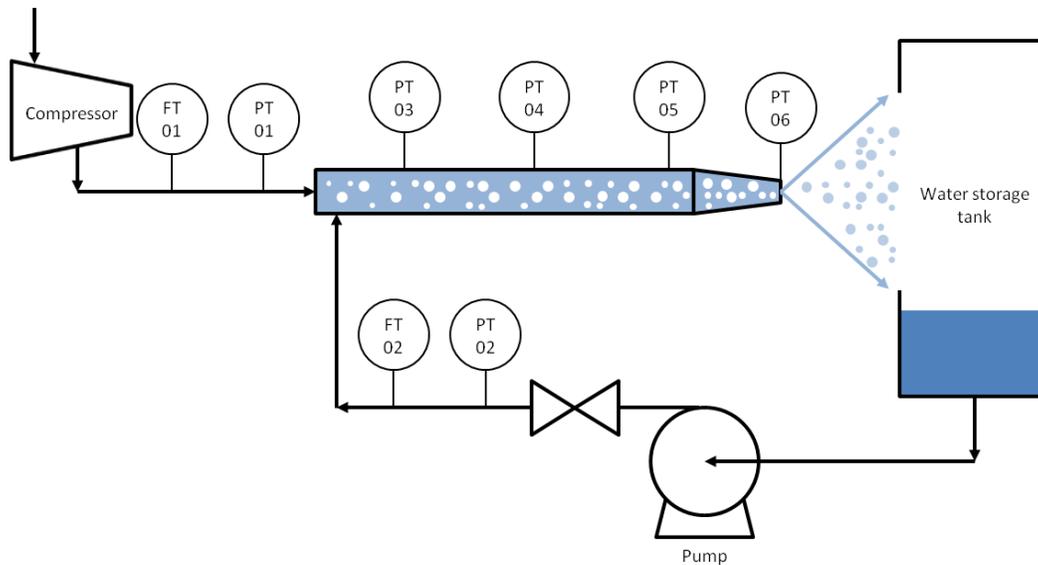
Figure 4.2. Experimental set-up for the gas-liquid spray system (not to scale)

There were two types of measurements of interest in the system: flow meter outputs from the compressor and pump (FT01 and FT02), and outputs from the pressure sensors in the feeding conduit and at the nozzle (PT03, PT04, PT05 and PT06). The flow measurements were used to estimate the gas to liquid ratio. The pressure measurements were taken at 1 kHz and used for the flow regime modeling using digital signal processing.

Three experimental sets of runs were conducted. The first experiments were done at a constant feeding pressure of 70 psi, for four different GLRs of 0.5, 2.1, 2.9 and 5.6%, and a total sampling time of 63.6 s. The second set of experiments were conducted at a constant feeding pressure of 90 psi, at three different GLRs of 0.6, 2.2 and 5.2% and a total sampling time of 48 s. For the final experiments the water flow rate was kept at a constant value of 1.5 gpm, and the GLR was slowly increased from 0.5% to 1.0, 1.3, 1.7, 2.0, 2.1 and 2.6%, for a total sampling time of 59.1 s. The pressure was not kept constant for these last experiments; hence, the increase in GLR caused an increase in feeding pressure from 65 psi at 0.5% GLR to 100 psi at 2.6% GLR. This change in pressure was considered small enough to not affect the stability transition criterion.

The stability of the sprays was determined visually using a high speed camera. Stable sprays were observed only at the lowest of GLRs of each run (0.5% for 70 psi, 0.6% for

90 psi and 0.5% for a constant liquid flow rate of 1.5 gpm). All other runs presented a pulsating behaviour, although at different intensities.

4.3 Stability detection using signal processing

To verify the applicability of the digital signal processing method for classification, the stability of the spray was modeled using the procedure outlined in Chapter 3, using the high frequency pressure measurements from the experiments.

4.3.1   Data selection

With a total of fourteen runs: three in stable flow, two possibly within the transition zone (1-1.5% GLR), and ten in unstable flow, it was necessary to choose two of these experiments to model the classification into stable and unstable flows. The transition zone was not modeled as a separate class. Sample sizes were held the same for both groups to maintain a consistent degree of accuracy for the models of the two classes. For the initial attempt, both data sets used for modeling belonged to the same experimental run (constant 70 psi, 90 psi or 1.5 gpm of liquid). This allowed for two different validation tests: validation within the same experimental conditions and validation on other experimental conditions. If the model was accurate only within its experimental runs, it confirms the applicability of this method, albeit in a rather limited fashion. In contrast, if the modeling for one run was accurate on other runs, the method had a broader range of applications.

In an attempt to ensure unbiased modeling, data selection was done randomly. Initially, the experimental run at a constant pressure of 70 psi was chosen as the modeling data for the first attempt. After analyzing all the results using the training data from runs at 70 psi, the other two runs were employed for modeling as well to ensure that data selection was not influential in the process. The results of all modelling attempts are presented and discussed.

Four data sets were taken from the 70 psi experiments. Only one run was conducted at a GLR that was low enough to ensure stable flow, i.e. GLR = 0.5%, and so this one was chosen for the stable flow model. While any of the remaining three data sets could have been used for the unstable flow model, the one known to be at conditions farthest from transition, i.e. GLR = 5.6%, was chosen.

The final issue regarding the data selection involved which measurement to use, that is, data from which specific pressure transducer in the experimental setup. This was not considered as a possible source for bias, but rather an important question to be answered: does the measurement point affect the results? As shown by previous researchers (Maldonado et al. 2008), pressure oscillations are not equal in all of the sensors involved due to flow development. As a starting point, the pressure signals from the transducer located on the feeding conduit just upstream of the nozzle (PT-05 in Fig. 4.2). This appeared to ensure that the pressure signals from this transducer would be associated with fully developed flow, based on the high-speed video recordings. It must be noted that for this type of analysis, the system studied must be at steady state, which is likelier for the selected pressure transducer than in the other three. As was the case with the previous data selection criteria, measurements from the other pressure transducers were also used for modeling and their results are briefly discussed later.

4.3.2   Data analysis and pre-processing

Pressure measurements

Pressure signals from PT-05, recorded over 4 s at a set pressure of 70 psi, are shown in Figure 4.3 (stable spray, GLR = 0.5%) and Figure 4.4 (unstable spray, GLR = 5.6%). Through the remainder of this section, all references to measurements from stable and unstable flows will refer to these specific experimental conditions.

The most noticeable difference between the measurements is that the overall pressure variation, of approximately 62-71 psi for the stable flow, is significantly smaller than that observed for the unstable flow (53-72 psi). A less noticeable difference is that the unstable flow measurements are mostly smooth curves, while in stable flow the measurements are rough, as if noise or several frequency components were present.
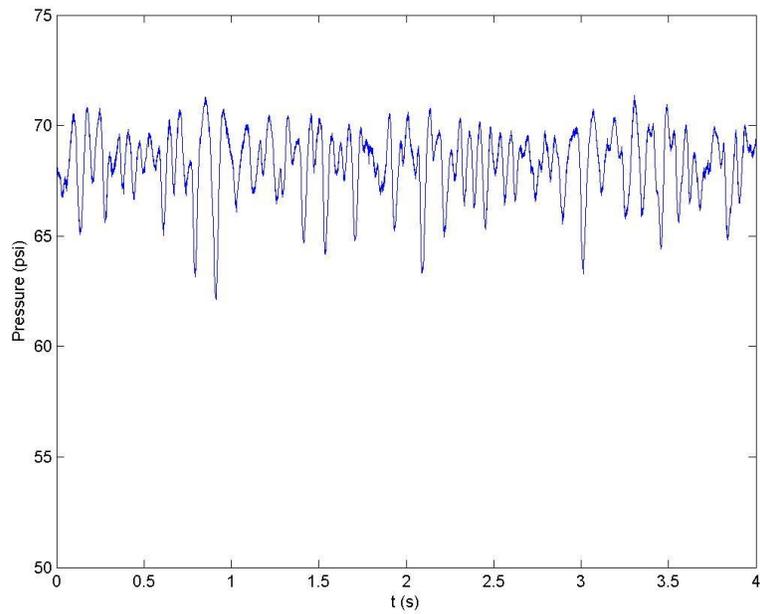
Figure 4.3. Pressure signal from the transducer upstream of the nozzle (PT-05) for a stable air-water spray at an operating pressure of 70 psi and a GLR of 0.5%.
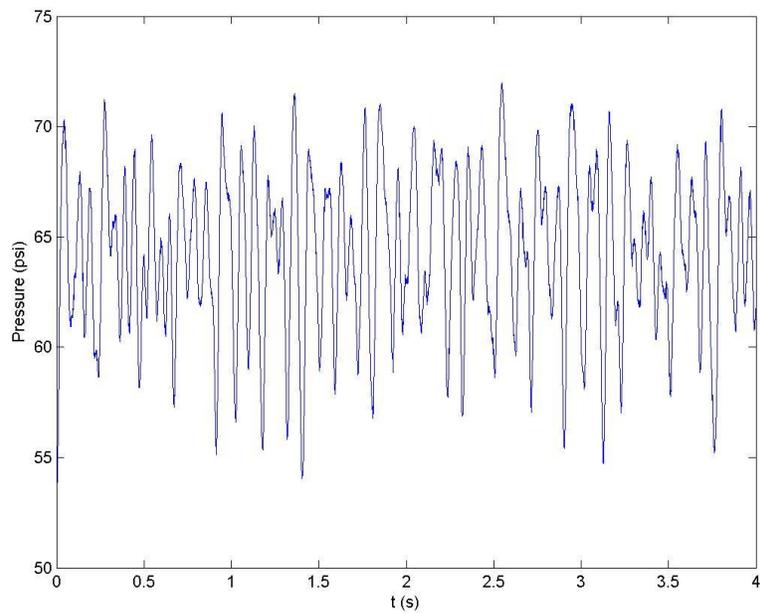
Figure 4.4. Pressure signal from the transducer upstream of the nozzle (PT-05) for an unstable air-water spray at an operating pressure of 70 psi and a GLR of 5.6%.

Time window

To determine a time window for the data partitioning, the autocorrelation test was performed on both pressure measurements, and can be seen in Figure 4.5 and Figure 4.6, for the stable and unstable flow, respectively.
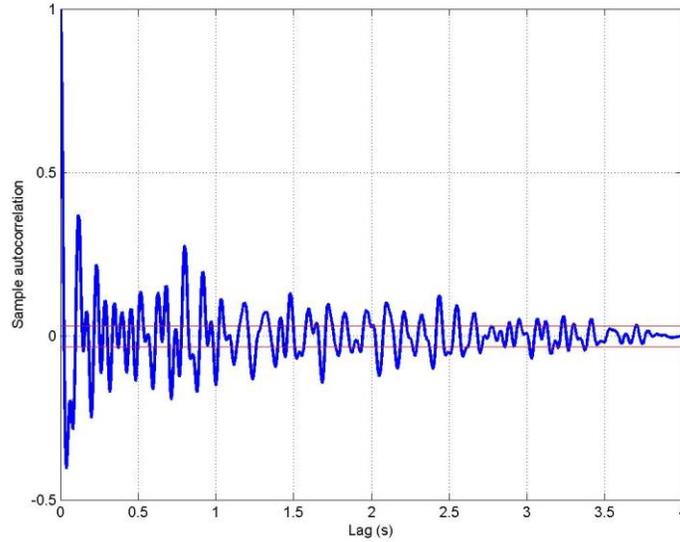
Figure 4.5. Autocorrelation test performed on pressure measurements for stable flow.
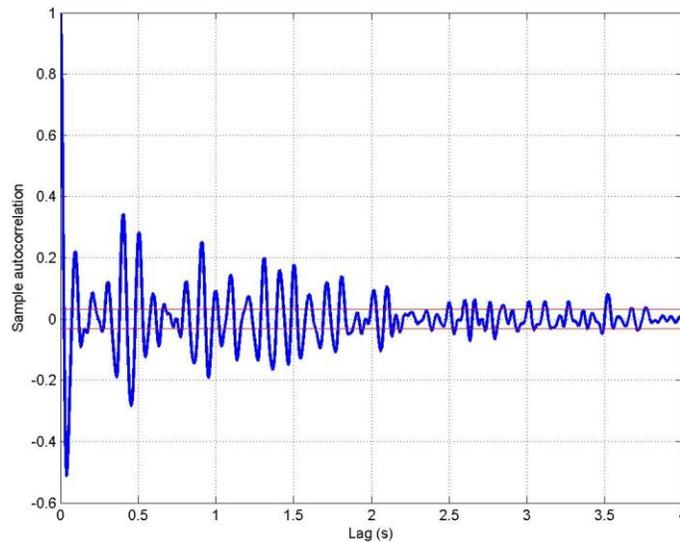
Figure 4.6. Autocorrelation test performed on pressure measurements for unstable flow.

There is no clear dominant peak in either system, with their highest peaks at less than half a second, and several peaks of slightly smaller magnitude scattered around. If the smallest time window is taken, such that only the highest peak is covered, some of the information could be lost. To ensure enough information is considered, the time window must cover all significant peaks so that every different periodic pattern completes at least a full cycle. However, if a time window that is too large is considered, the sensitivity of the model to transient behaviour is greatly diminished. Considering the above points, a time window of two seconds was used, which covers the dominant peaks in both measurements, without compromising sensitivity.

Mean centering and scaling

The results from the mean centering and scaling pre-processing procedures can be observed in Figure 4.7 (stable flow) and Figure 4.8 (unstable flow). The percentages of variation are significantly different between the stable and unstable flow, with the stable flow having much smaller deviations from the mean when compared to the unstable flow.
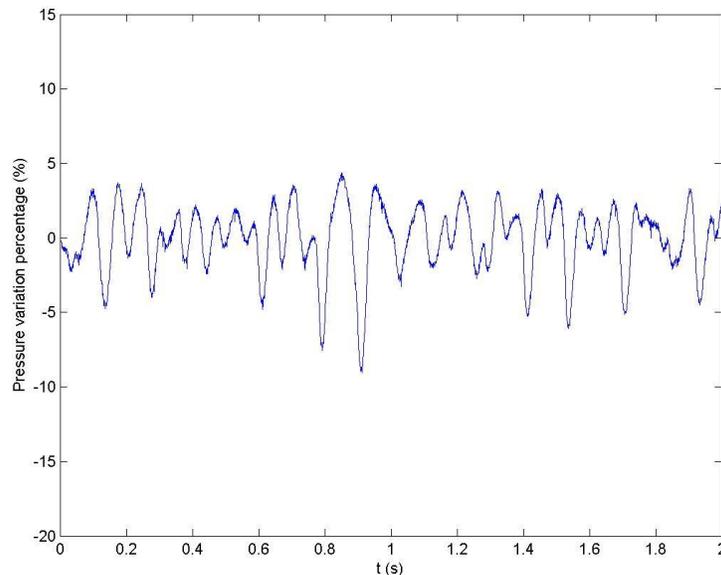
Figure 4.7. Mean centered and scaled pressure measurement (PT-05) as a percentage of variation for a stable air-water spray at an operating pressure of 70 psi and a GLR of 0.5%.
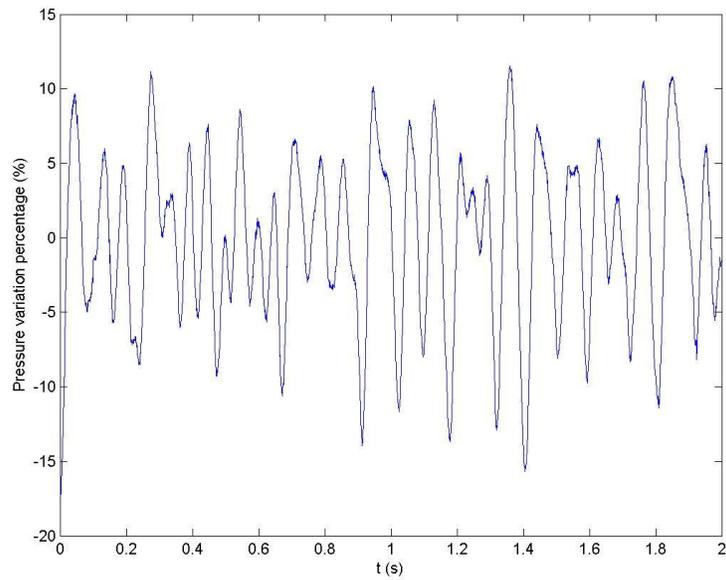
Figure 4.8. Mean centered and scaled pressure measurement (PT-05) as a percentage of variation for an unstable air-water spray at an operating pressure of 70 psi and a GLR of 5.6%.

Statistics

The histogram can be used to illustrate the differences in the distribution of the pressure measurements, as shown in Figure 4.9 (stable flow) and Figure 4.10 (unstable flow). Although both distributions are "right-leaning", the shapes are actually different. The wider pressure range of the unstable flow makes the distribution slightly more symmetrical and flat, compared to the asymmetrical and "peaked" stable flow. The actual pressure ranges, which are significantly different, have been re-scaled in order to allow for direct comparison between the distributions (recall Section 3.1.3).
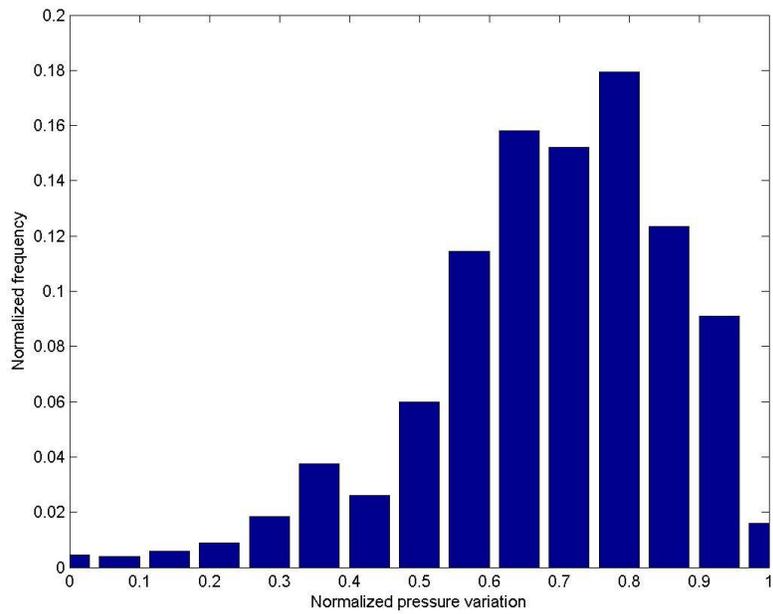
Figure 4.9. Histogram for the mean centered and scaled pressure measurement (PT-05) for a stable air-water spray at an operating pressure of 70 psi and a GLR of 0.5%.
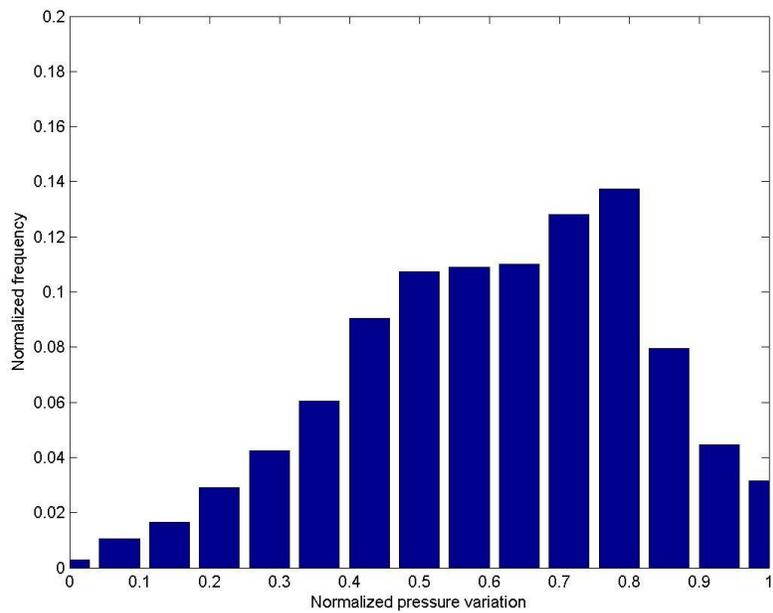
Figure 4.10. Histogram for the mean centered and scaled pressure measurement before the nozzle of an unstable air-water spray at an operating pressure of 70 psi and a GLR of 5.6%.

To further enhance the differences between the classes, the four statistical moments were calculated. These are presented in Table 4.1. Aside from the mean, which is close to zero from the mean centering, the values are different between the stable and unstable flows in the following ways:

- The standard deviation is significantly higher for the unstable flow, indicating a more erratic and higher variability in its pressure profile.
- Both have negative skewness, characterized by their right-leaning distributions, but the values are different.
- The excess kurtosis is positive for the stable flow, as seen in its strong peak, and negative for the unstable flow, portrayed by its flat top.

Table 4.1. Statistical moments for the mean centered and scaled pressure measurements before the nozzle of an air-water spray at an operating pressure of 70 psi and GLRs of 0.5% and 5.6%

Flow regime	Mean (μ)	Standard deviation (σ)	Skewness (s)	Excess kurtosis (k)
Stable	-1.51e-14	2.25	-0.94	1.20
Unstable	-2.68e-14	5.70	-0.36	-0.37

Wavelet Transform

Following the statistical analysis, the Daubechies order 4 wavelet was used. Figure 4.11 and Figure 4.12 show the wavelet decompositions of the pressure measurements for the stable and unstable flow, respectively. Both figures look reasonably similar, making a visual comparison too difficult. In order to obtain significant information from the wavelet decomposition, the four statistical moments were calculated at each level, as mentioned in Section 3.1.4. A simultaneous comparison of the stable and unstable flow statistical moments on the wavelet decomposition was also too complex given the high amount of variables (28 variables for each of the two flows), and was therefore omitted from this section. It should be mentioned that while slight differences were observed in several values, none indicated differences as clearly as the histograms. The inability to manually find differences is not considered to be a problem, given the PLS algorithm should detect relevant correlation on its own.
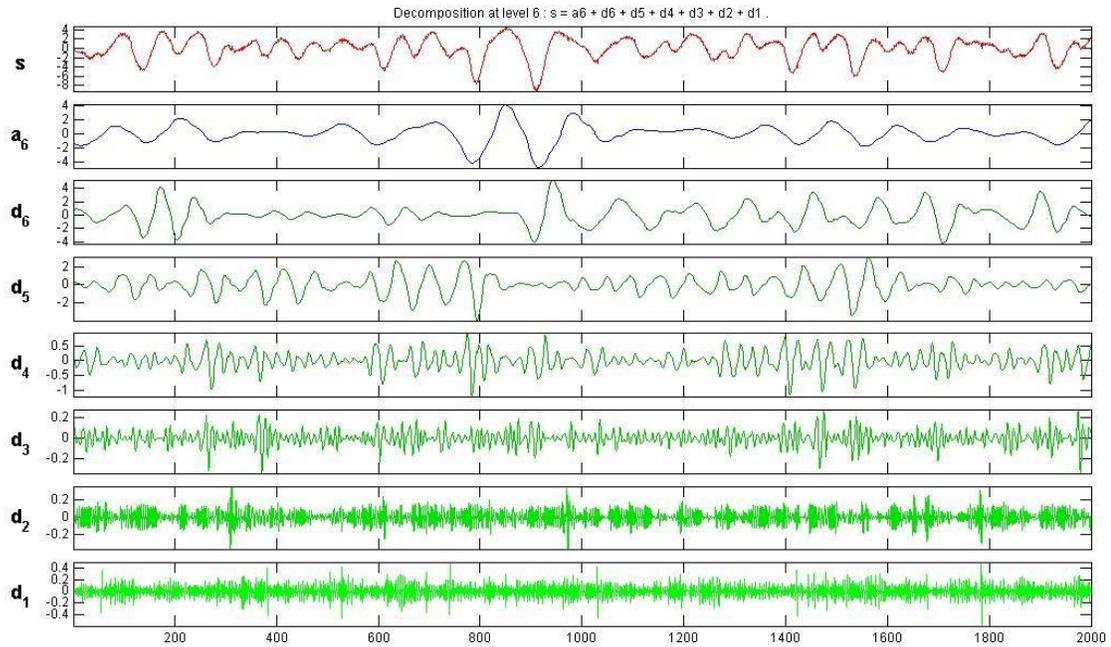
Figure 4.11. Wavelet decomposition using Daubechies order 4 and level 6 wavelet of the mean centered and scaled pressure measurements (PT-05) for a stable air-water spray at an operating pressure of 70 psi and a GLR of 0.5%.
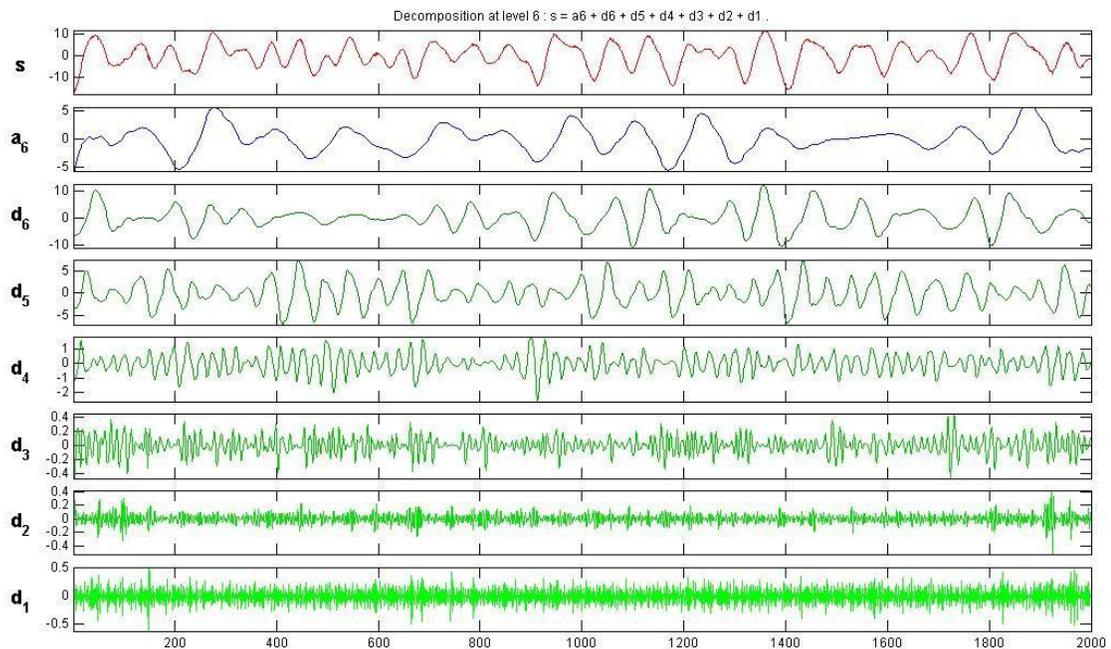
Figure 4.12. Wavelet decomposition using Daubechies order 4 and level 6 wavelet of the mean centered and scaled pressure measurements (PT-05) for an unstable air-water spray at an operating pressure of 70 psi and a GLR of 5.6%.

Power Spectra Distribution

The final preprocessing tool was the Fourier transform to perform the spectral analysis. Figure 4.13 and Figure 4.14 show the PSDs corresponding to the stable and unstable flows, respectively. The amplitudes were calculated up to the Nyquist frequency of one half the sampling frequency, or 500 Hz. However, both plots are shown up to just 50 Hz, given all values after this threshold were estimated as approximately zero.

As can be observed, the stable flow has several equally important frequencies from approximately 8 Hz to 17 Hz, while the unstable flow has only one dominant frequency with its peak around 12 Hz (the scatter around this central peak can be considered *leakage* as explained in Weeks 2010). The one dominant frequency in unstable flows is related to the intermittence of the system, having a relatively constant and large bubble size, characteristic of slug and plug flow. For the stable flow, it is known that the dominant flow regime is dispersed bubble, which has a much wider bubble size distribution, giving the system a characteristic bandwidth of several different frequencies. Also, a large difference in the amplitude is seen, which is proportional to the variation percentage of the data.
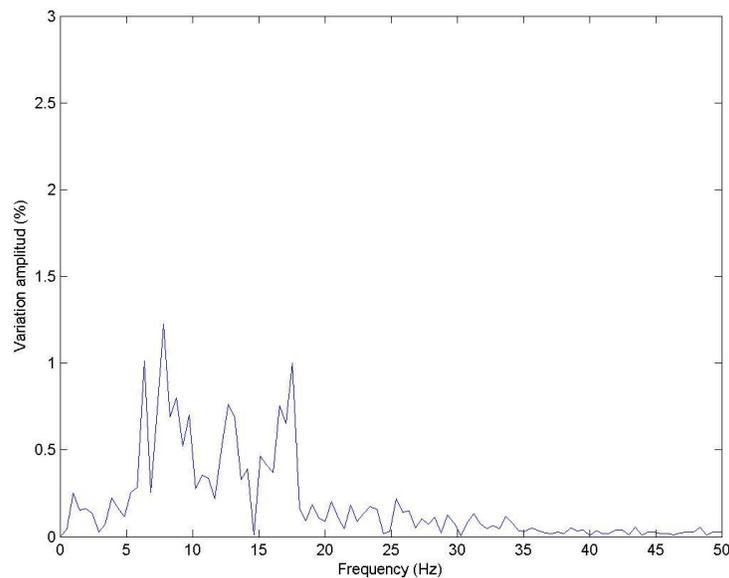
Figure 4.13. Power spectra distribution plot of the mean centered and scaled pressure measurements (PT-05) for a stable air-water spray at an operating pressure of 70 psi and a GLR of 0.5%.
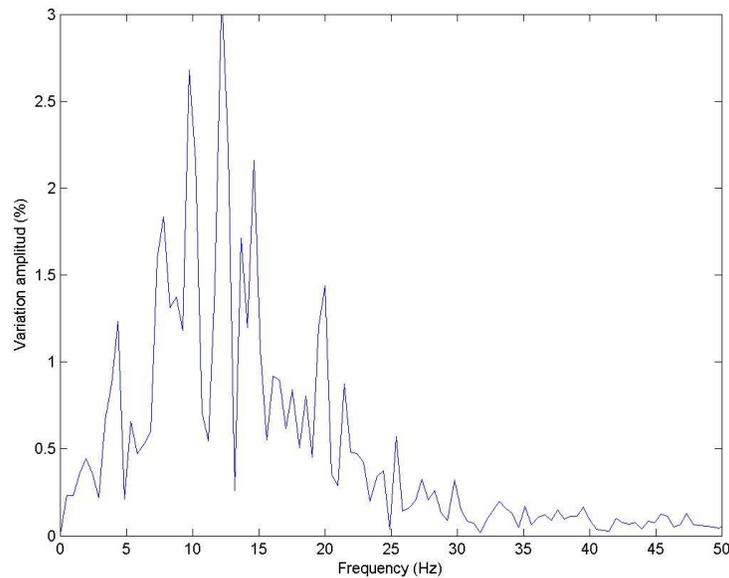
Figure 4.14. Power spectra distribution plot of the mean centered and scaled pressure measurements (PT-05) for an unstable air-water spray at an operating pressure of 70 psi and a GLR of 5.6%.

### 4.3.3   Model selection

As was mentioned in Section 3.2, the different decompositions were organized into a matrix, and several models were built. To select a model, several comparisons were made. The following section follows the model selection process.

Explained variance of X and Y

The variation percentage of X and Y described by the model for a growing number of latent variables (up to 10 out of a maximum of 30) are shown in Table 4.2. It can be seen that after just 3 latent variables, the variance accounted for in Y is already quite high (> 99%), indicating a low number of latent variables sufficient to adequately model the classifier. The variance accounted for in X, on the other hand is much lower, since it crosses the 80% mark after 9 latent variables (and the 90% threshold after 18, not shown). This is considered acceptable as most of the data in X is probably unnecessary to describe the stability of the flow.

Table 4.2. Cumulative percent variance described by the PLS model for different number of latent variables

Number of latent variables	X description (%)	Y description (%)
1	48.93	96.59
2	55.07	98.49
3	**60.41**	**99.19**
4	64.70	99.48
5	69.94	99.66
6	73.22	99.80
7	76.60	99.87
8	78.75	99.92
9	80.85	99.94
10	82.19	99.97

PRESS and BIC

Although the description values give a sense of how well the data are portrayed in the new dimensional space, the important criterion for selecting the number of latent variables is the BIC, which is shown in Figure 4.15, along with the PRESS value.
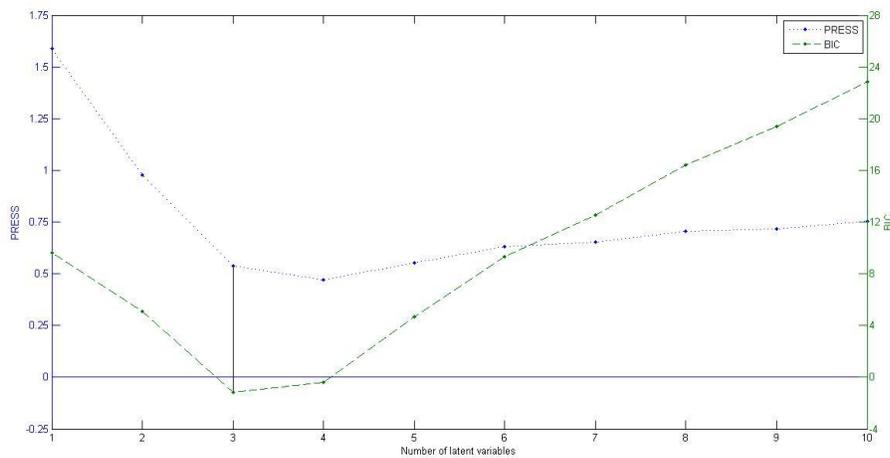
Figure 4.15. PRESS and BIC for the model as a function of latent variables

As can be seen, the lowest BIC value was obtained using three latent variables, despite the PRESS value continuing to decrease at 4 latent variables. This rise in the BIC indicates

that the small decrease in PRESS did not justify the increase from three to four latent variables. It must be noted that, although the PRESS can never be lower than zero, the BIC can, as in the present case, given the term $\ln(\hat{\sigma}^2)$ was negative.

Model statistics

The model corresponding to three latent variables was selected as the best from all models built. In order to apply the Bayes naive classifier, the mean and standard deviation of the classifiers were estimated for the modelling data, and are shown in Table 4.3.

Table 4.3. Mean and standard deviation of the stable and unstable classifiers

	True (belongs to)		False (does not belong to)	
Model	μ_j	σ_j	μ_j	σ_j
Stable	0.990	0.089	-0.990	0.115
Unstable	0.990	0.115	-0.990	0.089

As can be seen, the values are very close to the calibration values (1 for true, -1 for false), and the standard deviations are small, indicating the model has a good fit on the modeling set. Also, as can be noted, given this is a two-class model, one model is the exact opposite of the other. That is, the false statistics of the stable classifier correspond to the true statistics for the unstable classifier, and vice versa. If there were three or more classes, the false statistics of one class would not correspond to the true statistics of another class, as they do here. Given the case, it is sufficient to estimate the classifier from just one of the classes and make the classification decision based on it being true or false that it belongs to that class (in a two class problem, if it is false that it belongs to one class, it is assumed true to belong to the other). For this the stable flow classifier was used, where 1 represents belonging and -1 represents not belonging, i.e. unstable flow.

4.3.4   Model validation

Having selected a model, the two validation tests mentioned in Section 4.3.1 were performed.

Validation on identical experimental conditions

The first validation test was to verify the applicability of the classification within its own experimental conditions, which is more likely to succeed compared to a validation on different experimental conditions. For this, the model was validated on the other two experimental runs at 70 psi, as well as portions of the stable (0.5% GLR) and unstable (5.6% GLR) runs that were left out for validation (i.e. not used for modeling).

The results presented in Table 4.4 are the classifier (model output Y) and the Bayes classification decision of the selected model used on ten samples from the experimental run at 70 psi with a GLR of 2.1%, corresponding to an unstable flow. As can be seen, all ten data points have a highly negative value in the stable classifier (closer to -1), indicating not belonging to the stable class, leading to a classification as "unstable". Also, no values were close to zero (the approximate middle point between the two means of 0.990) where the standard deviation of each class come into play for calculating the probability.

Table 4.4. Classifier and classification decision of the model on an unstable flow with a GLR of 2.1% at 70 psi

Data point	Stable classifier	Classification Decision
1	-0.40	Unstable
2	-0.31	Unstable
3	-0.79	Unstable
4	-1.17	Unstable
5	-0.70	Unstable
6	-0.71	Unstable
7	-0.68	Unstable
8	-0.62	Unstable
9	-1.05	Unstable
10	-1.13	Unstable

The summarized results of the validation test on all the runs at 70 psi are presented in Table 4.5. The model accurately classified all data points belonging to these four data sets. It is evident that the model obtained was excellent, given there was not a single

misclassification in more than 90 attempts. It can be stated then that the modeling process in general is excellent, if used to determine the stability of a nozzle flow using pressure measurements, if the experimental conditions of the classified flow are identical to those on which the model was calibrated on.

Table 4.5. Flow stability classification for all runs at 70 psig

GLR (%)	Number of Classifications	
	Stable	Unstable
0.5	16	0
2.1	0	31
2.9	0	31
5.6	0	16

Validation on different experimental conditions

The second validation test was to apply the classifier on experimental data obtained on different experimental conditions. For this test, the model was validated on all other experimental runs. The summarized results are presented in Table 4.6.

Table 4.6. Flow stability classification for experimental runs at constant 90 psi pressure and at a constant liquid flow rate of 1.5 gpm

Experimental run	GLR (%)	Number of Classification	
		Stable	Unstable
Constant pressure at 90 psi	0.6	24	0
	2.2	0	24
	5.2	0	24
Constant liquid flow rate at 1.5 gpm	0.5	29	0
	1.0	8	21
	1.3	14	15
	1.7	3	26
	2.0	2	27
	2.1	0	29
	2.6	1	28

For the experimental runs at 90 psi, the results were also excellent. The model correctly classified all the measurements without a single misclassification. Similar accuracy was observed for the runs at 70 psi.

The results for the runs at a constant liquid flow rate of 1.5 gpm are almost as accurate as those previously mentioned. Once again, the stable flow is correctly classified with no errors. In the case of the unstable flows (GLR > 1.5%), three runs presented misclassifications, with the most being 3 out of 29 (10.3% misclassification rate for a GLR of 1.7%), which are all very good results.

As indicated in Section 4.1.1, the transition zone occurs approximately between 1 and 1.5% GLR. This transition zone is characterized by behaving as either stable or unstable flow at different points in time, in a somewhat stochastic manner. The model classified the two experimental runs found within the transition zone (GLRs of 1 and 1.3%), as sometimes stable and sometimes unstable.

The fact that the model recognizes a difficulty in classifying the data, and moving slightly closer to unstable as GLR increases proves that the modeling process is in fact modeling the flow regimes adequately. Of all the results, this one has the greatest value, proving that a classifier is not only accurate for the stable and unstable regimes, but also for the transition.

To better illustrate the difficulty presented by the system in classifying the transition zone, the average value of the classifier is presented in Figure 4.16, for all validation results. The zone between the blue lines is the transition zone.
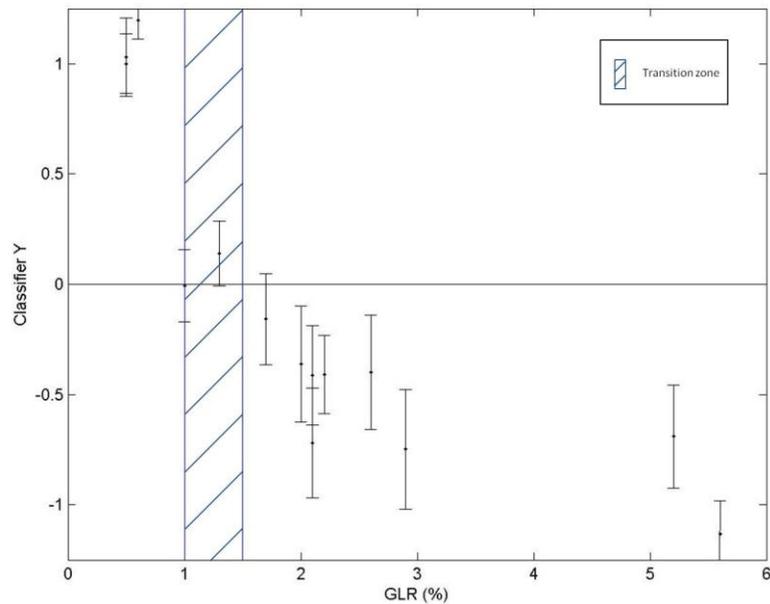
Figure 4.16. Average classifier output as a function of GLR using modeling data from experiments at 70 psi.

There was a clear trend in the classifier for a change in GLR: as GLR increased, the classifier decreased, moving farther away from 1 and closer to -1. Also, all values were clearly non-zero, except for the runs within the transition zone and the run at 1.7% GLR, for which the classifier was close to zero. This was expected as these values were within or near the transition zone, where a middle point value for the classifier is logical.

Given the good results, it is evident that the model exhibits a very high degree of correspondence with the physics of the system, since it describes the system even if modelled and validated on different experimental conditions.

Overall, the process yielded a very good model which accurately described the behaviour of the stable and unstable flow. The highest misclassification rate was a mere 10%. Additionally, the model was able to accurately describe the progressive transition from stable to unstable flow.

Influence of modeling data selection

A new model was obtained using the pressure measurements from the experimental runs at 90 psi to verify that the data selection did not affect the accuracy of the model.

The modeling sets were the runs with GLRs of 0.6% and 5.2%, for stable and unstable flow, respectively. The average classifier as a function of GLR is shown in Figure 4.17.
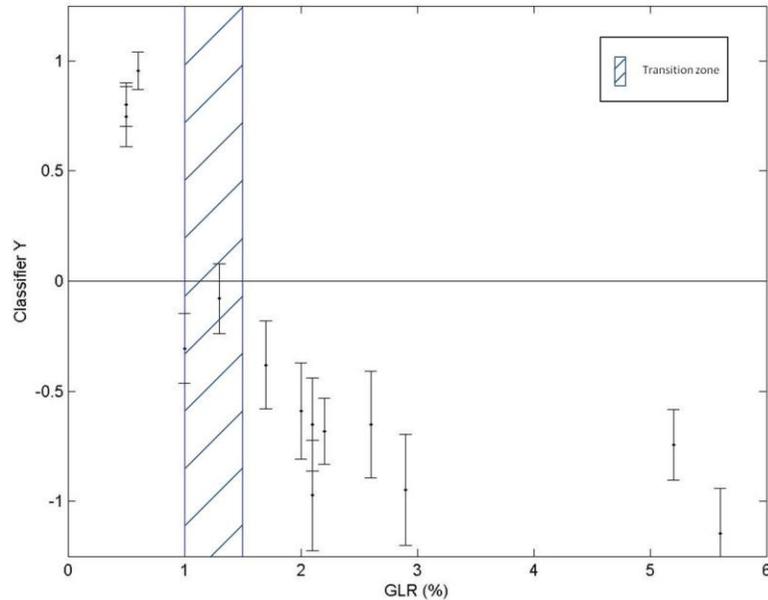
Figure 4.17. Average classifier output as a function of GLR using modeling data from experiments at 90 psi.

The same trend from previous results (Figure 4.16) is still present, with a slight shift of all points to a lower value. It seems the transition is slightly different at 90 psi, yet the data points are still adequately represented.

The accuracy of the model is slightly better than before for completely stable/unstable flows, having not a single misclassification in the twelve runs that lie outside the transition zone. As for the transition zone, given the model's statistics were extremely disparate (standard deviation was 0.111 for stable, 0.189 for unstable), all points were classified as unstable (higher probability to be unstable, given the z-score is inversely proportional to the standard deviation). The classification of all transition points as unstable is considered a very simple problem which can be easily solved by having more data points for the modelling, which would lead to a more compact distribution of the classifier, and hence a lower standard deviation.

All possible combinations of modelling with data from the same experimental runs were attempted. In general, all models classified the twelve stable and unstable runs with

minimal error, with slight differences in the transition zone, similar to the ones in the previously shown results. It was found that the models with the highest rate of misclassification (approximately 15%) were obtained using unstable data with low a GLR. These runs were: 2.1% for a pressure of 70 psi, 2.2% for a pressure of 90 psi, and all runs with a liquid flow of 1.5 gpm. Even for the least accurate model, the results were satisfactory.

Improved modelling

It has been demonstrated that a model can be built on a set of experimental conditions and applied to different experimental conditions with good results. To improve on this, the model can be built using data from all experimental conditions, to obtain a global model that describes the overall behavior much better. This is a fairly well known concept for regression analysis: given a better training set, a better model can be built.

In an ideal case, a model could be built using every different combination of GLR and system pressure, yielding a truly universal model. However, this would be impossible to do, and impractical, given there is no need for such accuracy. As was demonstrated, the behaviour of the stable and unstable flows can be described by just a few different experimental conditions.

In this sense, a "better" model was built using a portion of the data from one stable and one unstable run from each of the three experimental conditions. Only 20 seconds (10 data points) were used from each of the pressure measurements, and the rest were used for validation. The modeling data sets are shown in Table 4.7.

Table 4.7. Experimental data used for the improved modeling

Experimental conditions	Stable flow GLR	Unstable flow GLR
Pressure at 70 psi	0.5	5.6
Pressure at 90 psi	0.6	5.2
Liquid flow rate at 1.5 gpm	0.5	2.6

The model was validated on all 14 experimental runs, and the results are summarized in Table 4.8.

Table 4.8. Flow stability classification for experimental runs at constant 70 psi pressure, constant 90 psi pressure and constant liquid flow rate of 1.5 gpm

Experimental run	GLR (%)	Number of Classification	
		Stable	Unstable
	0.5	21	0
	2.1	0	31
Pressure at 70 psi	2.9	0	31
	5.6	0	21
	0.6	14	0
Pressure at 90 psi	2.2	0	24
	5.2	0	14
	0.5	19	0
	1.0	5	24
	1.3	4	25
Liquid flow rate at 1.5 gpm	1.7	0	29
	2.0	0	19
	2.1	0	29
	2.6	0	29

The validation results for this model are perfect in the classification of the stable and unstable flows, with no misclassifications in a total of 281 data points. This indicates an excellent description of the properties of the stable and unstable flows. In the case of the transition, the model still detects the shift from stable to unstable, however with a much higher preference for the unstable regime. To better illustrate this, the average classifier as a function of GLR are shown in Figure 4.18. As can be seen, the only experimental runs that are close to zero are those within the transition zone, as was expected.
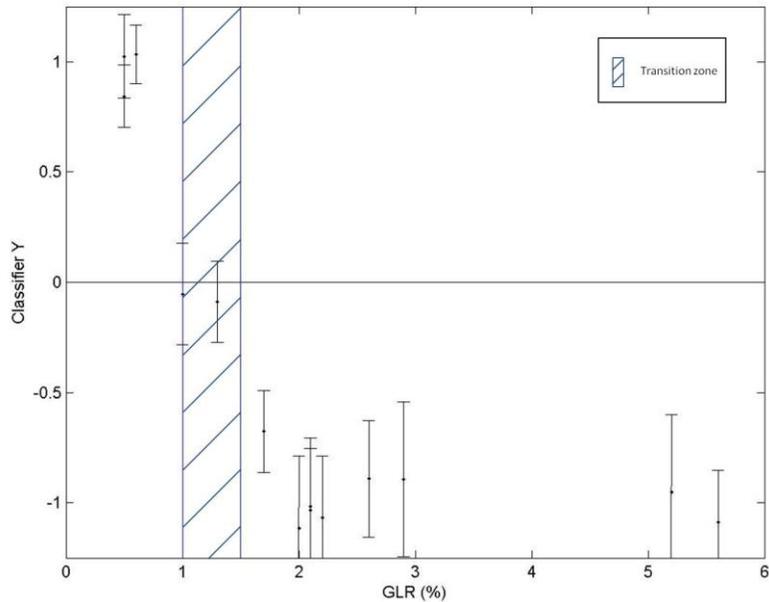
Figure 4.18. Average classifier output as a function of GLR using modeling data from all three experimental sets.

Measurement point

The models shown previously were built on the pressure measurements taken just before the nozzle, where the flow is fully developed. To further study the applicability of the modeling process, a model was built on data from each of the different measurement points.

The results obtained from this approach proved that the models built on any of the three feeding conduit pressure measurements were very accurate. The results were almost the same as the third model shown, again with slight differences in the transition zone. However, when using the nozzle measurements, the model was very inaccurate, even on the modeling data set, which it should be biased towards. Table 4.9 shows the mean and standard deviation of the classifier on the modeling data.

Table 4.9. Mean and standard deviation of the classifier using measurement at the nozzle

Model	True (belongs to)		False (does not belong to)	
	μ_j	σ_j	μ_j	σ_j
Stable	0.310	0.259	-0.471	0.446
Unstable	0.471	0.446	-0.310	0.259

Clearly, the model does not describe the system very accurately, given the average values are far from the calibration of +1 and -1. Additionally, the standard deviations are almost as high as the average values. A classification was attempted despite the deficiencies of the model, yielding extremely inaccurate results. It is evident that this type of modeling process does not work on pressure measurements at the nozzle.

The most noticeable difference in the nozzle measurements with respect to pipe measurements is an extreme dampening of the pressure oscillations. Figure 4.19 shows the pressure measurements taken at the nozzle for the experimental run at 70 psi with a GLR of 0.5%. As can be clearly seen, the amplitudes of the oscillations are much lower than those shown in Figure 4.3.
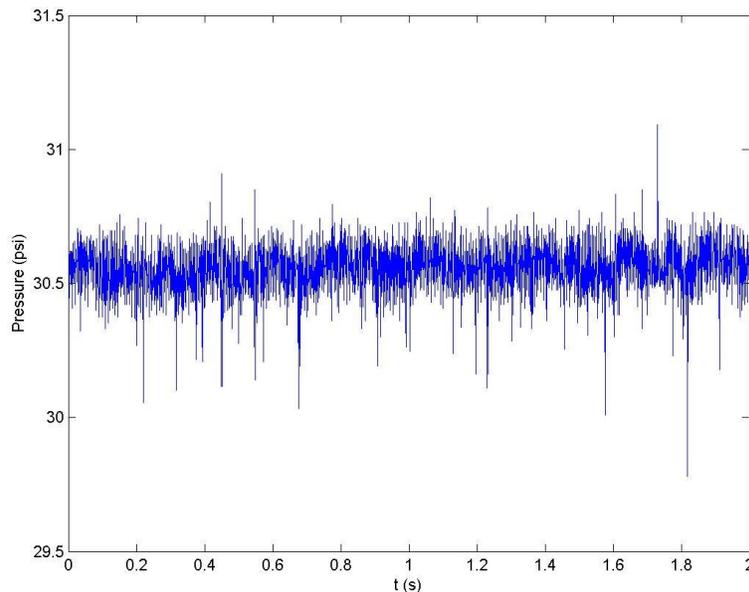
Figure 4.19. Pressure signal from the transducer at the nozzle (PT-06) for a stable air-water spray at an operating pressure of 70 psi and a GLR of 0.5%.

Mean centering and scaling of these measurements results in more than 9 out of 10 oscillations being less than 1% of the average. Comparison of these results those presented in Figure 4.7 shows a marked decrease in the magnitude of the oscillations. This dampening effect can be attributed to the high pressure drop the nozzle produces, which works to control the outlet pressure at a constant value. Due to this, modeling on this data was not possible.

4.4 Overview of gas-liquid modelling

Results show that modelling the distinction, and transition, from a stable spray flow (dispersed bubble) to an unstable spray flow (intermittent flow) is possible using the established modelling procedure. Also, by applying the modelling procedure to many different combinations of modelling/validation data, it was clearly shown that the information extraction process is robust enough that it captures the essential parameters in a wide variety of conditions.

Perhaps the most promising result is that the classifier follows a progressive change as the system transitions from stable to unstable flow as the GLR is increased. If more experiments could be performed at GLRs close to the transition (from 0.75-2%), the true accuracy of the model could be tested, and tuned for improvements.

Overall, the results are very encouraging, given they show that it is possible to model gas-liquid transition from dispersed bubble to intermittent flow, and that future work in gas-liquid multiphase flow modelling using signal processing should be able to achieve the required accuracy for industrial applications.

References

Ariyapadi, S., Berruti, F., Briens, C., Knapper, B., Skwarok, R., and Chan, E. (2005). "Stability of horizontal gas-liquid sprays in open-air and in a gas-solid fluidized bed" Powder Tech., 155(3), 161-174.

Hulet, C., Briens, C., Berruti, F., Chan, E. W., and Ariyapadi, S. (2003). "Entrainment and Stability of a Horizontal Gas-Liquid Jet in a Fluidized Bed" International Journal of Chemical Reactor Engineering, 1(1).

Maldonado, S., Fleck, B., Heidrick, T., Amirfazli, A., Chan, E. W., and Knapper, B. (2008). "Development of an experimental method to evaluate the stability of gas-liquid sprays." ATOMIZATION AND SPRAYS, 18(8), 699-722.

Rahman, M. A., Balzan, M., Heidrick, T., and Fleck, B. A. (2012). "Effects of the gas phase molecular weight and bubble size on effervescent atomization." Int.J.Multiphase Flow, 38(1), 35-52.

CHAPTER 5: SOLIDS SUSPENSION IN HORIZONTAL SLURRY FLOW

This chapter presents the overview and results of an experiment used to study the suspension of inert solids in a horizontal slurry flow, using the modeling procedure described in Chapter 3.

5.1 Slurry flow: Solids Suspension

As mentioned in Section 1.1, the suspension of solids occurs when the liquid flowing velocity is greater than the deposition velocity (Shook et al. 2002). Operating below the deposition velocity creates a stationary solids bed. Operation at or just above the deposition velocity creates a moving bed. For most slurries of industrial importance, operation at higher velocities reduces the bed height until the bed height reaches zero. Figure 5.1 presents a schematic illustrating these scenarios.
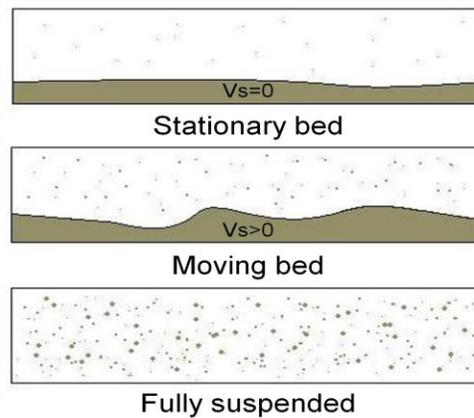
Figure 5.1. Schematic of different solid-liquid (slurry) flows.

5.2 Experimental set-up

Shown in Figure 5.2 is the schematic of the experimental set-up located at the Saskatchewan Research Council's Pipe Flow Technology Centre. It was used in this project to study the suspension of solids in horizontal slurry flow. Water and the selected solids (see below) were mixed in a slurry tank and pumped through a pipe loop. the temperature was maintained at 20 °C using two double tube heat exchangers employing glycol as the heating/cooling fluid. The internal diameter of the slurry pipe was 75.6 mm.
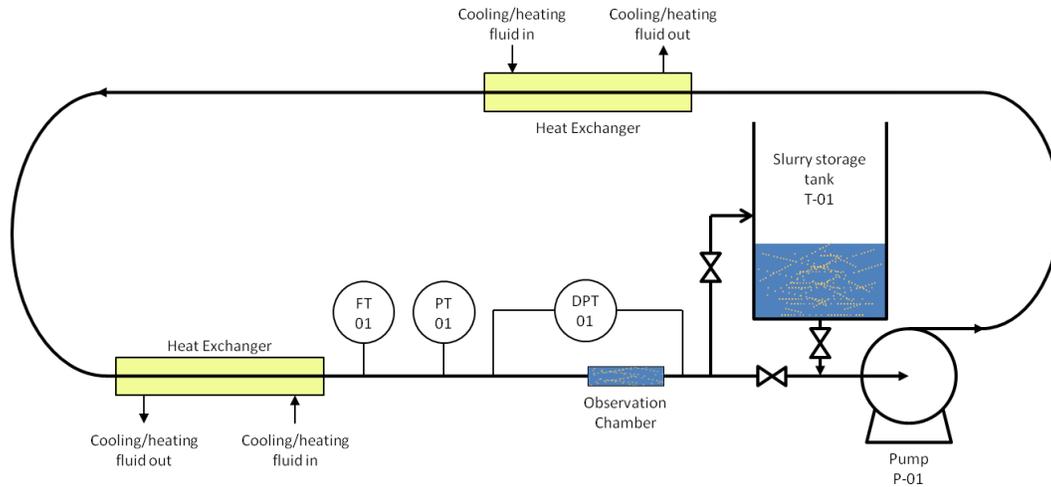
Figure 5.2. Experimental set-up for the slurry pipe loop (not to scale)

Pressure measurements were taken on the return (low pressure) section of the pipe loop (shown as PT01 in Figure 5.2) at a frequency of 1 kHz. Differential pressure was measured for purposes beyond this investigation. Flow rate was measured and manipulated by changing the pump speed to achieve different levels of suspension of the solid particles. The onset of solids suspension (moving bed regime) was determined visually in the observation chamber.

Experiments were conducted with silica sand (d50= 420 µm, ρ=2650 kg/m^3) and zirconium silicate (d50= 500 µm, ρ=4000 kg/m^3) at different concentrations and different velocities, as shown in Table 5.1 The velocity of 5.5 m/s was approximately the maximum velocity the pump system could handle at the given concentrations of solids.

Table 5.1. Experimental matrix for the slurry pipe loop

Solid particles	C_s (%)	Velocity (m/s)			
		V_d	3.5	4.5	5.5
Silica sand	30	✓	✓	✓	✓
	35	✓	✓	✓	✓
Zirconium silicate	5	✓	✓	✓	✓
	10	✓	✓	✓	✓
	15	✓	✓	✓	✓
	20	✓	✓	✓	✓

5.3 Solids suspension detection using signal processing

To study the moving bed and fully suspended flows using digital signal processing, the procedure presented in Chapter 3, excluding the data classification step (naive Bayes classifier), was used on the collected pressure measurements. The classification into a specific flow pattern was not performed given that the transition from moving bed to suspended flow is rather slow, since bed height gradually decreases to zero as velocity increases. Instead, the raw classifier (Y) is used to show relative progress from moving bed to suspended flow as velocity increases, in a case similar to the study of Y as a function of GLR for the gas-liquid system. Complete suspension of the solids (fully suspended flow) was not achieved for any experiments. However, in order to validate the modelling procedure, it was assumed that at maximum velocity (5.5 m/s), the flows were fully suspended.

To prepare the slurry mixtures, the mass of solids was calculated using Equation 5.1.

$$m_s = V_t \cdot C_s \cdot \rho_s \qquad (5.1)$$

The total volume of the loop (V_t) was estimated at 195 L.

5.4 Water-Sand flow

The following corresponds to the modelling and validation on the silica sand experiments.

5.4.1 Data selection

Data for the modelling was selected from every concentration (30% and 35%) at moving bed (approximated deposition velocity), and from every "fully suspended" flow (5.5 m/s). Although concentration affects the deposition velocity and required velocity for fully suspended flow, the small increment from 30% to 35% is considered small enough that both experiments can be modeled together.

5.4.2 Data analysis and pre-processing

Pressure measurements

Shown in Figure 5.3 and Figure 5.4 are two seconds of the pressure measurements taken at a concentration of 30% sand (by volume), for 2.5 m/s and 5.5 m/s, respectively.

Through the remainder of this section, all references to measurements from moving bed and suspended flows will refer to the mentioned experimental conditions.
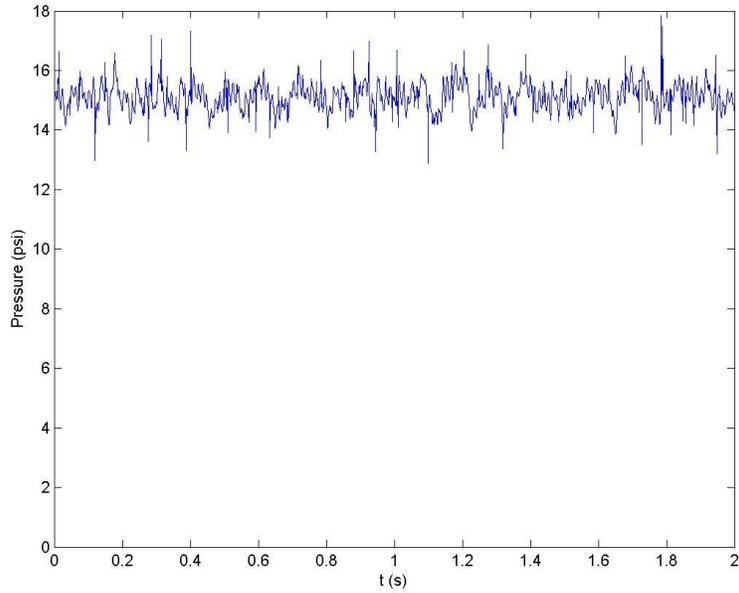
Figure 5.3. Pressure measurement for a water-silica sand flow at a velocity of 2.5 m/s and concentration of 30%, corresponding to moving bed flow.
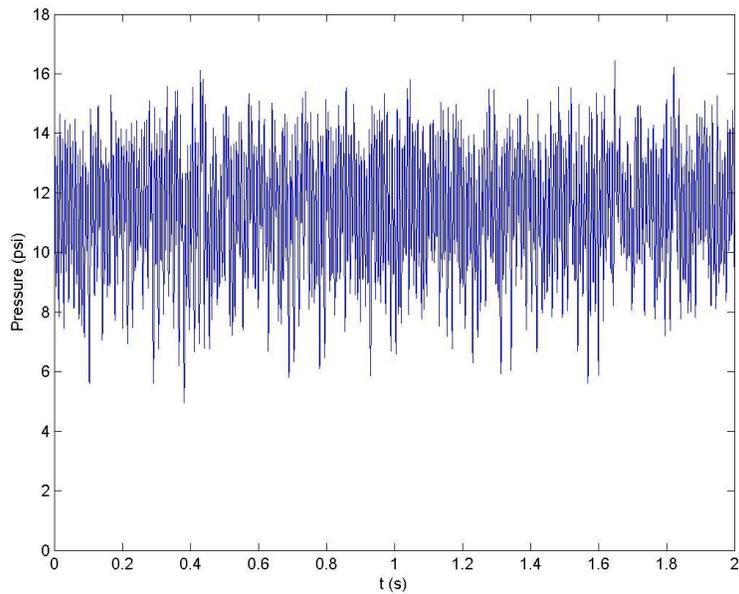
Figure 5.4. Pressure measurement for a water-silica sand flow at a velocity of 5.5 m/s and concentration of 30%, corresponding to suspended flow.

The pressure measurements from the different flows are clearly different. The moving bed flow has a much lower variation (13 – 18 psi), compared to the suspended flow (6 – 16 psi). Also, the suspended flows seem to have a much higher oscillation frequency, as there is no clear separation between one cycle (maximum to minimum). For the moving bed measurements, the oscillations are much more spread out in time.

Time window

To determine a time window for the data preprocessing, the autocorrelation test was performed on both pressure measurements, and can be seen in Figure 5.5 and Figure 5.6, for the moving bed and suspended flows, respectively.
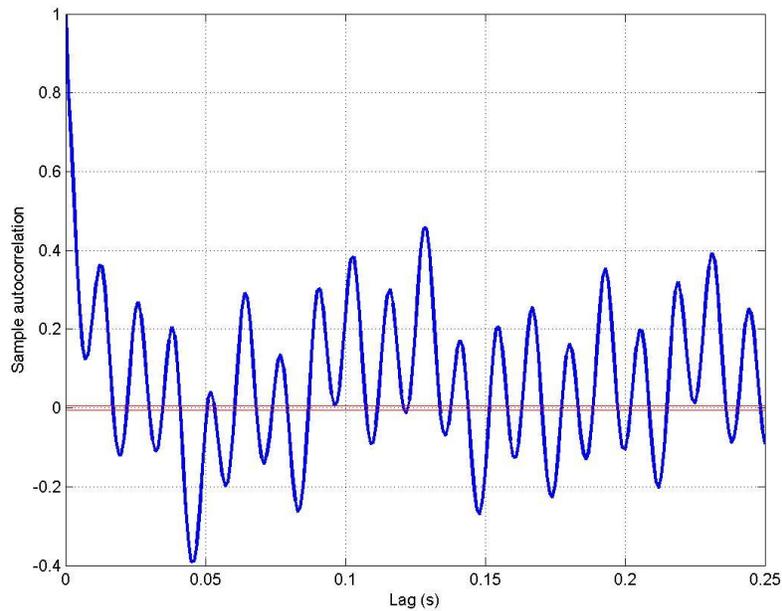
Figure 5.5. Autocorrelation test performed on pressure measurements for moving bed flow.
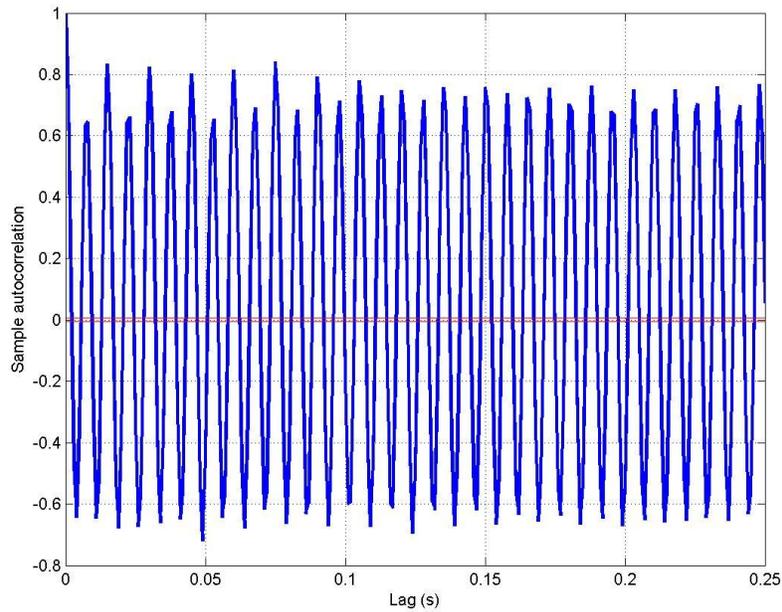
Figure 5.6. Autocorrelation test performed on pressure measurements for suspended flow.

Clearly, the autocorrelation test is inconclusive, as the correlation coefficients seem to have almost the same value at all time steps, although it does decay slowly. This indicates that either an extremely small time step is the maximum correlation and the rest are higher order harmonics, or that the data is non-stationary (Makridakis et al. 1997). Given that the data were checked for steady state behaviour, the only possibility is that a very small (< 0.01 s) small time step is the maximum correlation. Given the future of this research is to model solids suspension simultaneously with gas-liquid flow patterns, the larger of the two time requirements should be used. Therefore, the same time step as the gas-liquid system was used (2 s).

Mean centering and scaling

The results from the mean centering and scaling can be observed in Figure 5.7 and Figure 5.8, for moving bed and suspended flows, respectively. As can be seen, the percentages of variation are significantly different between the different flows, the moving bed flow having much smaller deviations from the mean, when compared to the suspended flow.
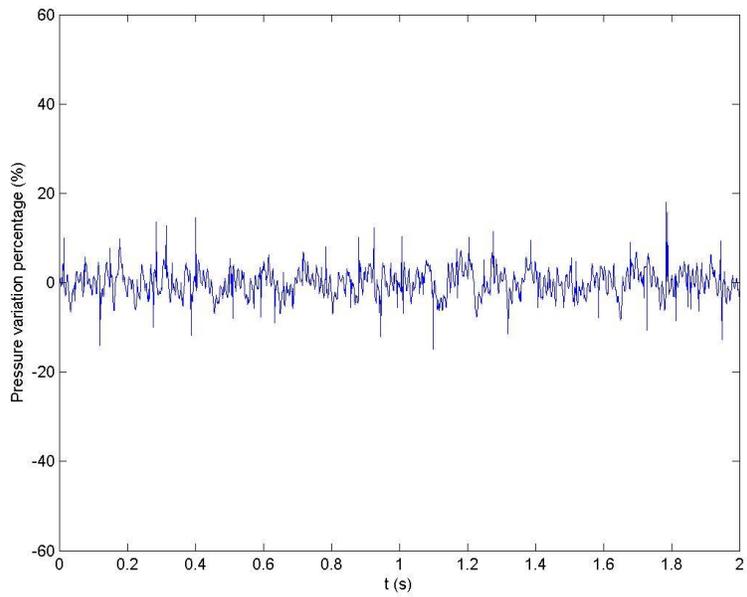
Figure 5.7. Mean centered and scaled pressure measurement for a water-silica sand flow at a velocity of 2.5 m/s and concentration of 30%, corresponding to moving bed flow.
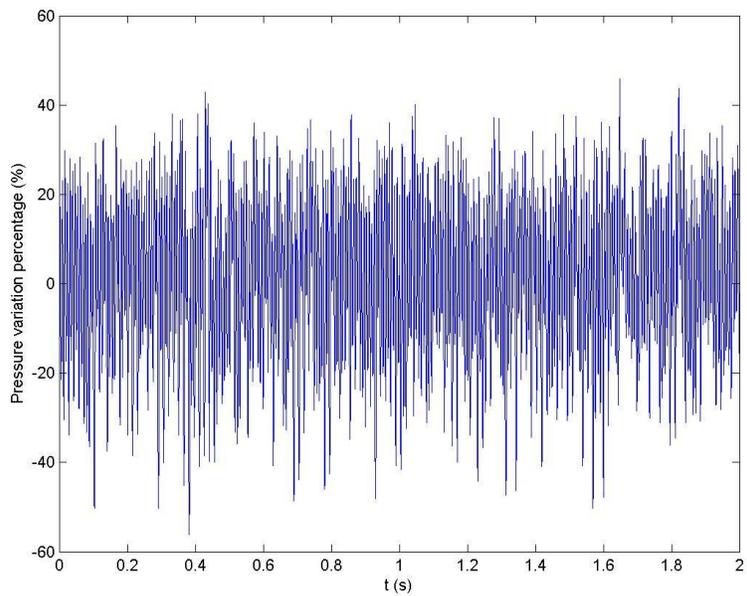
Figure 5.8. Mean centered and scaled pressure measurement for a water-silica sand flow at a velocity of 5.5 m/s and concentration of 30%, corresponding to suspended flow.

Statistics

The histograms are shown in Figure 5.9 and Figure 5.10 for the moving bed and suspended flows, respectively.
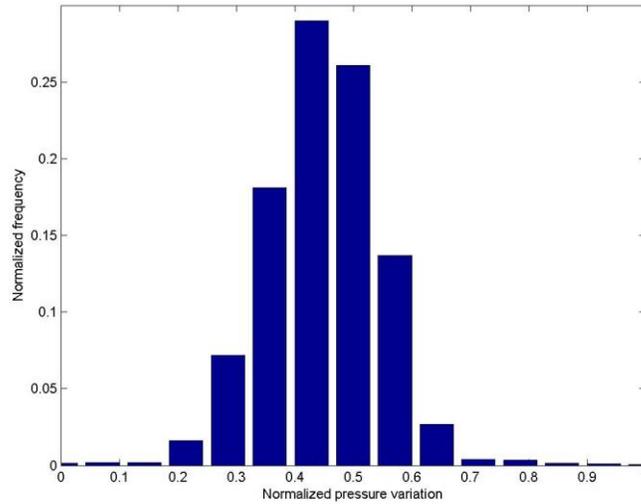

Figure 5.9. Histogram for the mean centered and scaled pressure measurement for a water-silica sand flow at a velocity of 2.5 m/s and concentration of 30%, corresponding to moving bed flow.
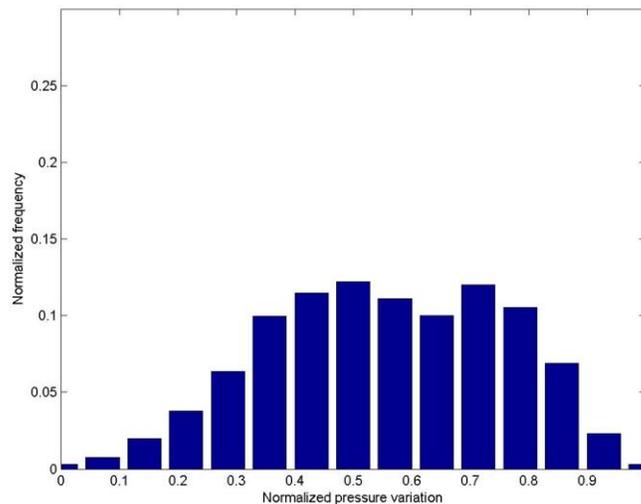
Figure 5.10. Histogram for the mean centered and scaled pressure measurement for a water-silica sand flow at a velocity of 5.5 m/s and concentration of 30%, corresponding to suspended flow.

The differences between the two histograms are quite clear. The moving bed flow histogram is left leaning with a sharp peak. The suspended flow histogram is right leaning, very flat and slightly bimodal. Based on this alone it is probably possible to classify the flows.

To further investigate the differences between the classes, the four statistical moments were calculated, and are presented in Table 5.2.

Table 5.2. Statistical moments for the mean centered and scaled pressure measurements from experiments with silica sand at 30% solids concentration

Flow regime	Mean (μ)	Standard deviation (σ)	Skewness (s)	Excess kurtosis (k)
Moving bed	1.00E-13	3.06	0.10	2.25
Suspended	-7.70E-15	19.15	-0.16	-0.71

Aside from the mean, which is close to zero from the mean centering, the values are different between the stable and unstable flows in the following ways:

- The standard deviation is significantly higher for the suspended flow.
- The moving bed flow has positive skewness (left leaning curve), whereas the suspended flow has negative skewness (right leaning curve).
- The excess kurtosis is positive for the moving bed flow, as seen in its strong peak, and negative for the suspended flow, portrayed by its flat top.

Wavelet Transform

The Daubechies 4 wavelet was performed on the data sets and the four statistical moments at each level were calculated. As they do not show any immediately observable differences, they are not presented.

Power Spectra Distribution

The Fourier transform was performed to obtain the PSDs, as shown in Figure 5.11 and Figure 5.12, corresponding to moving bed and suspended flows, respectively. Both plots are shown up to 300 Hz given all values after this threshold were estimated at zero. Note that, contrary to previous plots, the figures are not shown up to the same Y axis value. This is due to the maximum amplitudes being at different orders of magnitude.
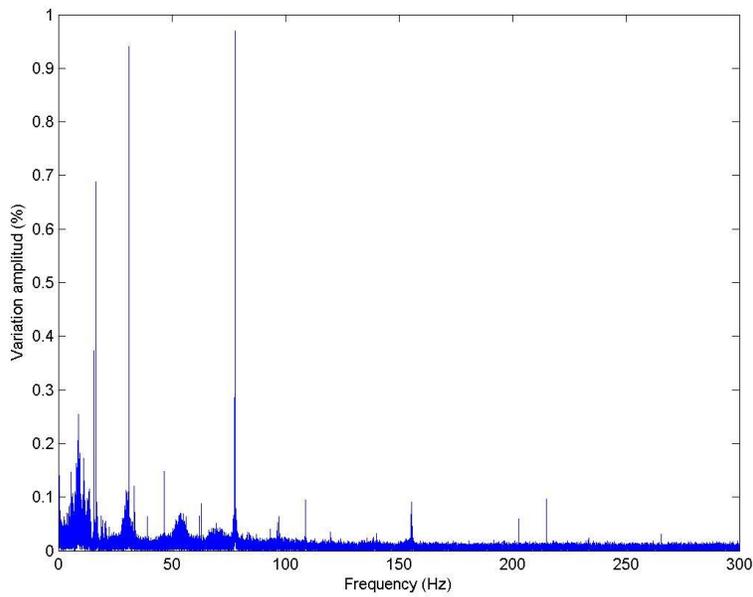
Figure 5.11. Power spectra distribution plot of the mean centered and scaled pressure measurement for a water-silica sand flow at a velocity of 2.5 m/s and concentration of 30%, corresponding to moving bed flow.
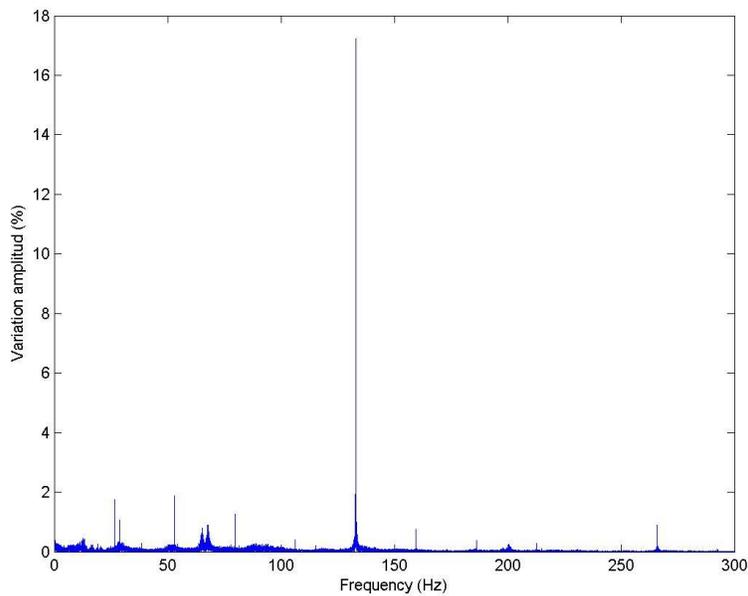
Figure 5.12. Power spectra distribution plot of the mean centered and scaled pressure measurement for a water-silica sand flow at a velocity of 5.5 m/s and concentration of 30%, corresponding to suspended flow.

The moving bed flow has several important peaks at 12, 25 and 75 Hz of amplitude approximately one. The suspended flow has its maximum at approximately 135 Hz, with an amplitude of over 17. Both cases have high frequency components, which confirms that the observed trends in the autocorrelation tests were in fact higher order harmonics caused by the underlying high frequencies in the measurements. The very high frequency components of the power spectra obtained for the suspended flow can probably be related to the stronger and more frequent particle-particle collisions occurring in the flow (Hashemi et al., 2012). For moving bed flow, there are fewer particles suspended, leading to less collisions, but still enough to have a high frequency response (80 Hz) in the pressure measurement. Also, there appears to be aliasing in the frequency response of the signals (example around 25 Hz for suspended flow), which might be caused by leakage.

An important contribution to the frequency behaviour can be attributed to the rotation speed of the pump. An example of the contribution of the pump to the spectrum of the data can be seen in Table 5.3, for a centrifugal pump with 5 impellers. As can be seen, the frequency of the pump matches peaks observed in the previously shown PSDs.

Table 5.3. Frequency contribution from the centrifugal pump

Solid particles	C_s (%)	Pump frequency (rpm [Hz]) at flowing velocity (m/s)			
		V_d	3.5	4.5	5.5
Silica sand	30	978 [82]	1133 [94]	1367 [114]	1611 [134]
	35	1009 [84]	1179 [98]	1411 [118]	1647 [137]
Zirconium silicate	5	714 [60]	934 [78]	1161 [97]	1379 [115]
	10	799 [67]	1007 [84]	1242 [104]	1461 [122]
	15	909 [76]	1093 [91]	1341 [112]	1547 [129]
	20	1010 [84]	1202 [100]	1435 [120]	1656 [138]

5.4.3   Model selection and Statistics

Using 15 seconds of pressure measurements from each experiment mentioned in Section 5.4.1, several models were built, and their BICs were compared. The model corresponding to eight latent variables was selected as the best from all models built, having a PRESS of only 0.026. Although classification of new data sets was not

performed, the mean and standard deviation of the model prediction on the modelling data (Table 4.3) were calculated, as they allow a first glimpse of the accuracy of the model. The standard deviation is very close to zero and the mean of the classifier differs from the unit in the fifth decimal place, indicating a nearly perfect fit.

Table 5.4. Mean and standard deviation of the classifier using measurements from experiments with silica sand at 30% and 35% solids concentration

	True (belongs to)		False (does not belong to)	
Model	μ_j	σ_j	μ_j	σ_j
Moving bed	1.000	0.008	-1.000	0.008
Suspended	1.000	0.006	-1.000	0.006

5.4.4   Model validation

Having selected a model, the transition from moving bed to suspended flow as velocity increases was studied, and is shown in Figure 5.13.
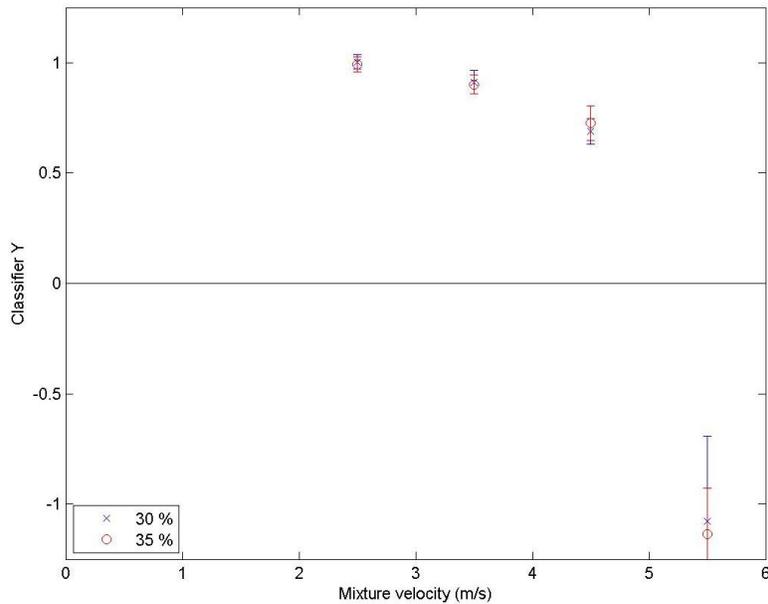
Figure 5.13. Average classifier output as a function of velocity using modeling data from experiments with silica sand.

Data sets from moving bed and from fully suspended flow are very close to 1 and -1, respectively, as they should be. This indicates the model is accurately describing each of

these behaviours. As for the transition, it appears that the progressive shift from moving bed (+1) to suspended (-1) does not occur in a linear manner. Both data sets for 3.5 m/s and 4.5 m/s are consistently placed around 0.9 and 0.73, respectively. Although a more pronounced gradient was expected, the results can be justified by arguing that a moving bed still exists before fully suspension occurs. Given the particle size and operating conditions tested here, a modelling result that predicts 'moving bed flow' is acceptable.

5.5 Water-Zirconium Silicate flow

The zirconium silicate beads are both heavier and denser, than the silica sand, which causes the deposition velocity to be greater. Because of the higher deposition velocity, experiments were only performed up to 20% solids concentration. This allowed almost complete suspension of the entire bed at maximum velocities.

5.5.1 Data selection

Contrary to the silica sand experiments, the concentration increment from 5% to 20% is significant enough to not allow them to be modeled together. Instead, experiments with a concentration of 5% and 10% were a first model, and experiments at 15% and 20% were a second model.

5.5.2 Model 1: 5% and 10%

Using 30 seconds of pressure measurements from each experiment at moving bed flow and suspended flow, the model corresponding to ten latent variables was selected as the best from all models built, having a PRESS of only 0.1. The mean and standard deviation of the model prediction on the modelling data are presented in Table 5.5. A very good fit was achieved.

Table 5.5. Mean and standard deviation of the classifier using measurement from experiments with zirconium silicate at 5% and 15% solids concentration

	True (belongs to)		False (does not belong to)	
Model	μ_j	σ_j	μ_j	σ_j
Moving bed	0.999	0.331	-0.999	0.323
Suspended	0.999	0.323	-0.999	0.331

Model validation

Having selected a model, the transition from moving bed to suspended flow as velocity increased was studied, and is shown in Figure 5.14.

Again, experiments from moving bed and suspended flow were modeled very close to +1 and -1, making the model very consistent with these two flows. For flows at velocities of 3.5 m/s and 4.5 m/s, the 5% experiments are being detected as slightly more suspended than the experiments at 10%, which is exactly what should occur. However, both 5% and 10% demonstrate again that the transition from moving bed to suspended flow is nonlinear within the model.
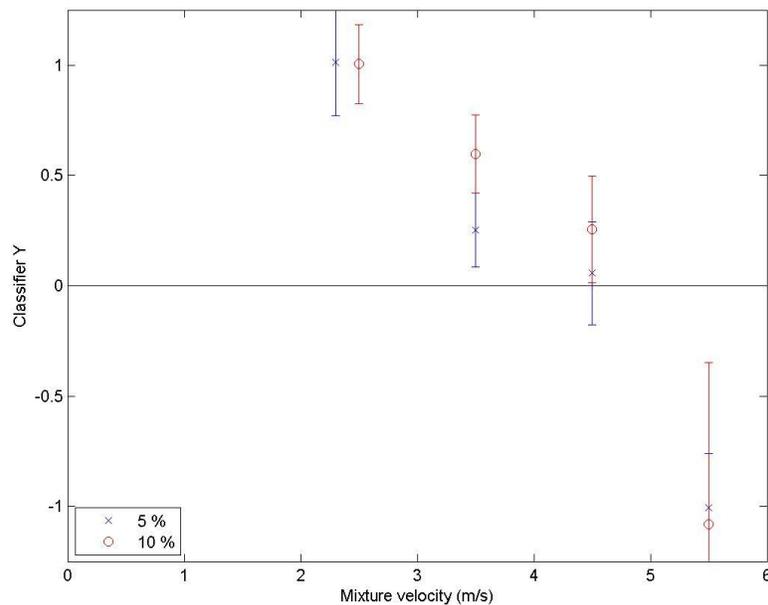
Figure 5.14. Average classifier output as a function of velocity using modeling data from experiments with zirconium silicate at 5% and 10% solids concentration.

5.5.3   Model 2: 15% and 20%

Using 15 seconds of pressure measurements from each experiment at moving bed flow and suspended flow conditions, the model corresponding to ten latent variables was selected as the best from all models built, having a PRESS of only 0.02. The mean and standard deviation of the model prediction on the modelling data are presented in Table 5.6. Again, a very good fit was obtained.

Table 5.6. Mean and standard deviation of the classifier using measurements from experiments with zirconium silicate at 15% and 20% solids concentration

Model	True (belongs to)		False (does not belong to)	
	μ_j	σ_j	μ_j	σ_j
Moving bed	1.000	0.015	-1.000	0.014
Suspended	1.000	0.014	-1.000	0.015

The standard deviation is very close to zero and the mean of the classifier differs from the unit in the fifth decimal place, indicating almost a perfect fit.

Model validation

Having selected a model, the transition from moving bed to suspended flow as velocity increases was studied, and is shown in Figure 5.15.
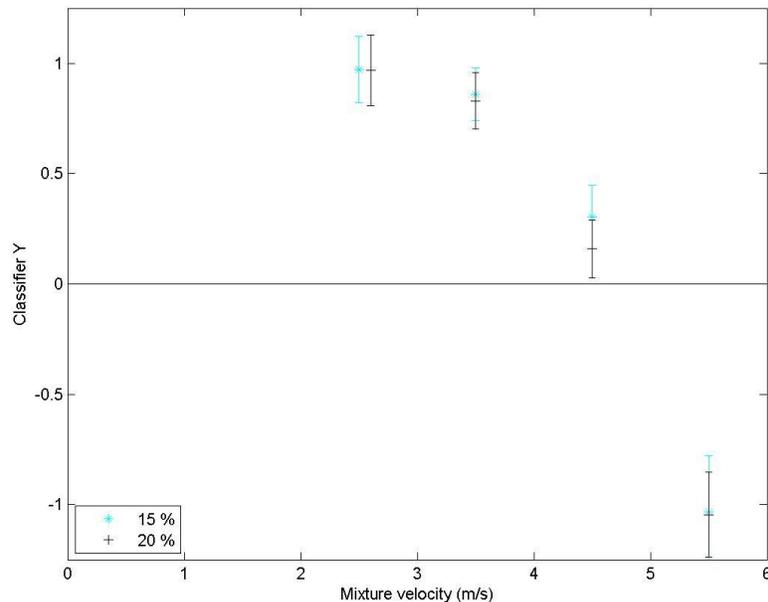

Figure 5.15. Average classifier output as a function of velocity using modeling data from experiments with zirconium silicate at 15% and 20% solids concentration.

Once again, the moving bed and suspended flows are very close to +1 and -1, as they should be. Given three different experiments consistently estimated the moving bed and suspended flows very well, it is clear that separation between the flows is possible and was achieved. For the experiments at velocities of 3.5 m/s and 4.5 m/s, both

concentrations presented very similar results, proving once again that the transition is not linear.

5.6 Overview of liquid-solid modelling

Given the excellent results in identifying the moving bed and suspended flows accurately, it can be stated that the modelling process distinguishes between the two flows very well. If only data from onset of moving beds and fully suspended flows were to be modelled, the modelling procedure presented in Chapter 3 is more than appropriate. However, results from velocities between these two flows showed a modelling preference to classify a tendency towards moving bed for all experiments. This could indicate a nonlinear relationship between the classifier and the flowing velocity. Due to this, special attention must be given towards adequately modelling the transition as linear, or using the nonlinear behaviour appropriately.

Overall, the results show a need for improvement, but are encouraging given the obtained differences between moving bed and suspended. With a wider arrange of experiments at different concentrations and with different solid particle sizes, adequate modelling of the transition from moving bed to suspended flow should be attainable.

References

Hashemi, S.A., Sadighian, A., Shah, S.I. and Sanders, R.S., "Solids concentration fluctuations in highly concentrated slurry pipe flows", submitted to J. Fluid Mechanics (2012).

Makridakis, S., and Hibon, M. (1997). "ARMA Models and the Box-Jenkins Methodology." J.Forecast., 16(3), 147-163.

Shook, C. A., Gillies, R. G., and Sanders, R. S. (2002). *Pipeline Hydrotransport with Applications in the Oil Sand Industry*. SRC.

CONCLUSIONS

1. Unstable air-water spray flows have a wider range of pressure oscillations caused by the disturbances generated by large bubbles passing through the pressure sensor, when compared to the smaller oscillations caused by the smaller bubbles in stable air-water spray flows.

2. Unstable air-water spray flow has a more erratic pressure profile as demonstrated by its higher standard deviation, when compared to the stable air-water spray flow.

3. The spectral behaviour of gas-liquid flows is pattern dependant, as shown in the differences between stable air-water spray flows which have a clear dominant frequency at around 12 Hz, compared to unstable spray flows which have a frequency bandwidth from 8-17 Hz.

4. Classification of stable and unstable air-water spray flows is possible with the proposed modelling process, and can achieve classification accuracies above 90%.

5. The established modeling procedure is able to accurately describe the transition from stable to unstable gas-liquid flows, as a relation between the model classifier and the GLR ratio.

6. Selection of the experimental conditions for the modelling data does not greatly affect the modeling procedure's accuracy.

7. Modeling with pressure measurements from within the pipe, at different lengths of the pipe does not significantly affect classification accuracy of a gas-liquid spray stability.

8. Flow through the nozzle dampens pressure oscillations, and does not allow the proposed modeling technique to describe the gas-liquid system accurately.

9. Variability in the pressure measurements, as indicated by the standard deviation, increased greatly as velocity increased, due to the higher degree of particle movement within the slurry flow

10. Liquid-solid suspended flow has a single dominant frequency at approximately 135 Hz, much higher than moving bed flow which has many lower frequency components of much lower amplitudes.

11. The pressure measurements from moving bed flow are very close to the average, having a very narrow histogram, compared to the wider and slightly bimodal distribution of the suspended flow.

12. The modelling procedure accurately identifies moving bed flow and suspended flow when operating a water-solid flow using silica sand and zirconium silicate as the solid particles, but does not accurately describe the transition.

13. The transition from moving bed to suspended flow appears to be non linear with respect to velocity within the presented modelling procedure.

RECOMMENDATIONS

1. Experiments for the gas liquid system only covered a small operating pressure range. In order to verify the applicability for other systems, more experiments should be conducted at higher and lower pressures in order to determine the range of applicability of a single model and of the modelling process in general.

2. The transition from stable spray to unstable spray flows was accurately modeled, but had very few points within the transition itself. Conducting more experiments within this transition could help fine tune the modelling process to achieve better results.

3. The suspension of solids from moving bed to fully suspended flow does not appear to occur in a linear relationship to velocity, when using the proposed modelling procedure. Nonlinear regression could be attempted as it might improve the modelling, at the cost of much higher computational complexity.

4. Physical occurrences in the experiments could not be matched to the pressure measurement's behaviour since high frequency videos could not be obtained. In order to understand why the modelling process works, time matching of data to video recordings should be performed.

5. Although the modelling process has been shown to work, the importance of each of the pre-processing techniques is unknown. If a more detailed study of the effect of each variable towards the accuracy is performed, a better model could be constructed.

6. Once a modelling process that is accurate enough in most gas-liquid and liquid-solid systems is ready, testing on three phase gas-liquid-solid flows should be performed.

APPENDIX A: MATLAB CODES

The following functions were developed and used in MATLAB for the modelling procedure, and are presented in alphabetical order.

- Classifier function

```
function [Y_val,X_val,Y_hat,Zt,Zf]=classifier(X,f,dt,beta,Xm,stats,o,l,f1,f2)
%CLASSIFIER Classifies data according to the statistics obtained from the
%modelling procedure
%   For any given data set X containing raw measurements or preprocessed
%   data, CLASSIFIER will preprocess it (if necessary) and classify using
%   probability calculations, according to the statistics in 'stats'.

% Transform data with spectral, wavelet and statistical analysis
if exist('f2','var')==1
    [~,X_val]=plsda(X,f,dt,2,[],o,l,f1,f2);
elseif exist('f1','var')==1
    [~,X_val]=plsda(X,f,dt,2,[],o,l,f1);
else
    if exist('l','var')==1
        [~,X_val]=plsda(X,f,dt,2,[],o,l);
    elseif exist('o','var')==1
        [~,X_val]=plsda(X,f,dt,2,[],o);
    else
        [~,X_val]=plsda(X,f,dt,2);
    end
end

% Eliminate mean
for i=1:size(X_val,1)
    X_val(i,:)=X_val(i,:)-Xm;
end

% Classify
if exist('flag','var')==0
    flag=0;
end
[Y_hat,Y_val,Zt,Zf]=modval(X_val,beta,stats);

end
```

- Cut function

```
function [ Y,k ] = cut( X,f,dt )
%cut Cuts a data sets into smaller data sets.
%   Cuts a vector data set into several shorter data sets the size of a
%   time window as specified by dt. The number of points in the sets is
%   equal to dt times the measurement frequency. If a matrix is the input,
%   only the first column will be cut.
%   X: data vector
%   f: frequency at which the data was measured
%   dt: time window for the new data set

l=floor(f*dt);          % Size of the subsets
k=floor(size(X,1)/l);   % Number of subsets
```

```matlab
% Preallocate size for speed
Y=zeros(l,k);

if l==1                    % Keep all data points
    Y=X;
else
    for i=1:k              % Cut into subsets
        for j=1:l
            Y(j,i)=X(l*(i-1)+j);
        end;
    end;
end;

end
```

- Cutarra function

```matlab
function [ Y,k,m,c ] = cutarra(  X,f,dt,o,l,f1,f2  )
%cutarra cuts a data set into smaller data sets and analyzes them as
%different sets, arranging them in a matrix.
%
%   [ Y,k,m ] = cutarra(  X,f,dt,o,l,f1,f2  )
%
%   Y: output data
%   k: number of data sets of each group
%   m: number of groups
%
%   The vector or matrix X will be cut down into smaller data sets using
%   the function 'cut'. It is assumed that if X is a matrix, each column
%   represents a different data set.
%
%   After cutting the data into smaller sets, it is analyzed using the
%   function 'statwave' through the intermediate function 'rearrange'.
%
%   The resulting matrix will have all the data from the analysis, where
%   rows will represent the different data sets and columns will be the
%   different statistical variables.
%

tag=2;
swpar

m=size(X,2);
for i=1:m
    if i==1
        [X_cut(:,:),k]=cut(X(:,i),f,dt);
        % Remove mean and divide by mean
        for j=1:k
            X_cut(:,j)=detrend(X_cut(:,j),'constant')*100/mean(X_cut(:,j));
        end
        Y(1+(i-1)*k:i*k,:)=rearrange(X_cut(:,:),f,o,l,f1,f2);
    else
        [X_cut(:,1+(i-1)*k:i*k),k]=cut(X(:,i),f,dt);
        % Remove mean and divide by mean
        for j=1:k
            X_cut(:,j+(i-1)*k)=detrend(X_cut(:,j+(i-1)*k),'constant')...
                *100/mean(X_cut(:,j+(i-1)*k));
        end
         Y(1+(i-1)*k:i*k,:)=rearrange(X_cut(:,1+(i-1)*k:i*k),f,o,l,f1,f2);
    end

end

% Original data histogram
histo=probdf(X_cut,15);    % Fixed number of bins
Y=[Y,histo];
```

```matlab
% Fourier Transform
FFT=fastft(X_cut,f);

% Use the fft results
% lY=size(Y,2);
Y=[Y,FFT];

end
```

- Daubfilt function

```matlab
function [ LowFilt,HighFilt ] = daubfilt( order )
% daubfilt returns the wavelet filter coefficients for a Daubechies wavelet
% of desired order
%   This will provide the coefficientes for a FIR structure of the
%   Daubechies low pass and high pass filter for orders 1 to 4

switch order        % The order determines the filter coefficients
    case 1
        LowFilt=[1/sqrt(2) 1/sqrt(2)];
        HighFilt=[1/sqrt(2) -1/sqrt(2)];
    case 2
        LowFilt=[-0.1294 0.2241 0.8365 0.4830];
        HighFilt=[-0.4830 0.8365 -0.2241 -0.1294];
    case 3
        LowFilt=[0.0352 -0.0854 -0.1350 0.4599 0.8069 0.3327];
        HighFilt=[-0.3327 0.8069 -0.4599 -0.1350 0.0854 0.0352];
    case 4
        LowFilt=[-0.0106 0.0329 0.0308 -0.1870 -0.0280 0.6309 ...
            0.7148 0.2304]/sqrt(2);
        HighFilt=[-0.2304 0.7148 -0.6309 -0.0280 0.1870 0.0308 ...
            -0.0329 -0.0106]/sqrt(2);
end

end
```

- fastft function

```matlab
function [ FFT ] = fastft( X,f )
%[ Fast Fourier Transform ] = fastft( Data vector , sampling frequency )
%   Calculates the Fast Fourier Transform using the MATLAB function for use
%   in the plsda function, removes the bottom 5 % of the values and
%   consider

L=size(X,1);
k=size(X,2);
NFFT=2^nextpow2(L);
freq=f/2*(0:2/NFFT:1);
FFT(:,1)=freq;
l=size(FFT,1);
X_d=detrend(X,'constant');
for i=1:k
    Y1(:,i)=fft(X_d(:,i),NFFT)/L;
    FFT(:,i+1)=2*abs(Y1(1:NFFT/2+1,i));
end

% Transpose
FFT=FFT';

% Cut up into bins
% lf=length(freq);
fband=300;                              % Only analyze f=(0-40)Hz
fsize=1;                                % Arbitrary window size of 1 Hz
binsize=floor(fsize*NFFT/f);
nbin=floor(fband/fsize);
```

```
FFTs=zeros(k,nbin);
for i=1:k
    for j=1:nbin
        FFTs(i,j)=max(FFT(i+1,1+binsize*(j-1):binsize*j));
    end
end

FFT=FFTs;

end
```

- Modval function

```
function [ Y_hat,Y_val,Zt,Zf ] = modval( x_v,beta,stats)
%MODVAL Returns the model output and the classification decision
%    For a data set x_v with preprocessed data, MODVAL uses the 'beta'
regressor
%    to estimate the model output 'Y_hat'. Then, using the mean and std
%    found in 'stats', assigns each data point in x_v a group in 'Y_val'

% Give X a column of ones
x_v=[ones(size(x_v,1),1) x_v];

% Calculate the model estimation Y
Y_hat=x_v*beta;              % Validation Y

% Determine the Z value for a normal distribution and classify
[n,k]=size(Y_hat);
Y_val=zeros(n,k);
Zt=Y_val;
Zf=Y_val;
Ratio=zeros(1,k);
for i=1:n
    flag=0;      % Indicates if a data sets is classified to a group
    for j=1:k    % Try to classify it into each group
        % Estimate Z values
        Zt(i,j)=abs((Y_hat(i,j)-stats(1,j))/stats(2,j));
        Zf(i,j)=abs((Y_hat(i,j)-stats(1,k+j))/stats(2,k+j));
        % Determine if it belongs in the group or not
        if Zt(i,j)<=Zf(i,j)  % It belongs
            if flag==0        % if it doesn't belong in any other group
                flag=j;
                Y_val(i,j)=1;
            elseif flag~=0  % Already in group j, use ratio to choose group
                if (Zt(i,j)/Zf(i,j))<(Zt(i,flag)/Zf(i,flag)) % New group
                    Y_val(i,flag)=-2;
                    Y_val(i,j)=1;
                    flag=j;
                else      % Group it already was in
                    Y_val(i,j)=-2;
                end
            end
        elseif Zf(i,j)<Zt(i,j) % It doesn't belong
            Y_val(i,j)=-1;
        end
    end
    if flag==0       % Data sets has not been classified in any group
        for j=1:k    % Determine ratios
            Ratio(j)=Zt(i,j)/Zf(i,j);
        end
        [~,indy]=find(Ratio==min(Ratio));  % Lowest ratio is best decision
        Y_val(i,indy)=2;
    end
end

end
```

- Normalize function

```
function [ x_n ] = normalize( x )
% normalize will normalize all values in a vector or in a matrix by column
%   For any given vector or matrix, the values of each entry will be
%   normalized to 1 by the highest absolute value of each comoponents.
%   For a vector, each comoponent is divided by the highest value
%   For a matrix, each column will be treated as a vector

%  Predefine size
max=zeros(size(x,2));
x_n=x;

%  Determine maximum in each vector
for j=1:size(x,2)
    for i=1:size(x,1)
        if x(i,j)<0           % Determine absolute value
            abs=(-1)*x(i,j);
        else
            abs=x(i,j);
        end
        if abs>max(j)         % Store maximum
            max(j)=abs;
        end
    end
end

% Normalize using the maximum
for j=1:size(x,2)
    if max(j)~=0
        x_n(:,j)=x(:,j)/max(j);
    end
end

end
```

- Optimize function

```
function [ beta,Sm,PRESS,g,cumvar,BIC]=optimize(x_m,Y,m)
%OPTIMIZE uses crossvalidation for choosing the number of principal
%componentes in the plsda model
%   For a modelling matrix, optimize will run the PLSDA function using an
%   8 fold crossvalidation technique, and will choose the number of
%   componentes trhough the Parsimony Principle.

% Separate data using 8 fold
[n,k]=size(x_m);
if n<k
    Ns=n/m;
else
    Ns=floor(k/m);
end
if Ns<8
    fold=Ns;
else
    fold=8;
end
fn=floor(Ns/fold);
frem=rem(Ns,fold);
if frem==0
    minsize=(Ns-fn)*m;
else
    minsize=(Ns-fn-1)*m;
end
PRESS=zeros(m,m,minsize);
for i=1:fold
```

```matlab
        % Divide into 8 subsets and put 7 in modeling and 1 in validation
        clear xm xv Yv Ym
        xm=x_m;
        Ym=Y;
        if i<=frem
            for j=1:fn+1
                xv(2*j-1,:)=x_m(1+7*(j-1)+i-1,:);
                xv(2*j,:)=x_m(Ns+1+6*(j-1)+i-1,:);
                Yv(2*j-1,:)=Y(1+7*(j-1)+i-1,:);
                Yv(2*j,:)=Y(Ns+1+6*(j-1)+i-1,:);
                xm(1+7*(j-1)+i-1,:)=[];
                xm(Ns+1+6*(j-1)+i-1-1,:)=[];
                Ym(1+7*(j-1)+i-1,:)=[];
                Ym(Ns+1+6*(j-1)+i-1-1,:)=[];
            end
        else
            for j=1:fn
                xv(2*j-1,:)=x_m(1+7*(j-1)+i-1,:);
                xv(2*j,:)=x_m(Ns+1+6*(j-1)+i-1,:);
                Yv(2*j-1,:)=Y(1+7*(j-1)+i-1,:);
                Yv(2*j,:)=Y(Ns+1+6*(j-1)+i-1,:);
                xm(1+7*(j-1)+i-1,:)=[];
                xm(Ns+1+6*(j-1)+i-1-1,:)=[];
                Ym(1+7*(j-1)+i-1,:)=[];
                Ym(Ns+1+6*(j-1)+i-1-1,:)=[];
            end
        end

        % Eliminate mean from xm and xv
        Xm=mean(xm);
        for yyy=1:size(xm,1)
            xm(yyy,:)=xm(yyy,:)-Xm;
        end
        for zzz=1:size(xv,1)
            xv(zzz,:)=xv(zzz,:)-Xm;
        end

        % Obtain all possible models
        betaaux=plsdafun(xm,Ym,1,minsize);
        if i==fold
            [betaaux,~,~,cumvar]=plsdafun(xm,Ym,1,minsize);
        end
        gs=size(betaaux,3);
        beta(:,:,1:gs,i)=betaaux(:,:,:);

        % Give xv a column of ones
        xv=[ones(size(xv,1),1) xv];

        % Calculate PRESS
        for counter=1:minsize
            PRESS(:,:,counter)=PRESS(:,:,counter)+(Yv-xv*beta(:,:,counter...
                ,i))'*(Yv-xv*beta(:,:,counter,i));
        end

    end

% Calculate BIC
BIC=zeros(m,minsize);
minCri=1000;
found=0;
for counter=1:minsize    % Check every model with different No. parameters
    Add=0;
    for i=1:m            % Add error for every model
        BIC(i,counter)=n/m*log(PRESS(i,i,counter))+counter*log(n/m);
        Add=Add+BIC(i,counter);
    end
```

```matlab
        if Add<minCri
            minCri=Add;
            g=counter;
        end
end

[beta,Sm,g,~]=plsdafun(x_m,Y,0,g);

end
```

- Plsda function

```matlab
function       [        x_m,x_v,beta,Xm,Sm,PRESS,g,cumvar,BIC        ]        =
plsda(X,f,dt,dataf,opf,o,l,f1,f2)
%plsda Performs the PLSDA algorithm on a matrix X.
%   For a given matrix X, plsda performs a PLS algorithm on it by
%   decomposing it into dt multiples of its sampling frequency f.
%   This data is first processed through wavelet decomposition and
%   statistical analysis before the plsregress is used on it. PLSDA can be
%   set to optimize by running all combinations of parameters.
%
%   plsda(X,f,dt,dataf,opf,o,l,f1,f2)
%
%   X: data matrix. Different column implies different class
%   f: sampling frequency
%   dt: how many seconds will a sample use
%   Data flag:
%    dataf = 0 separate data into modeling/validation sets (DEFAULT)
%    dataf = 1 all data is modelling
%    dataf = 2 all data is validation
%   Optimization flag:
%    opf = 0 do not optimize (DEFAULT)
%    opf = 1 optimize
%   o: order of the daubechis wavelet
%   l: level of the daubechies wavelet
%   f1: flag for correction for statistical bias (corrected is default f1=0)
%   f2: flag for normalization (normalized is default f2=0)

if nargin==3
    dataf=0;
    opf=0;
    tag=2;
elseif nargin==4
    opf=0;
    tag=3;
else
    tag=4;
end
swpar

% Get data matrix analyzed and arranged as needed
[x,k,m]=cutarra(X,f,dt,o,l,f1,f2);
z=size(x,2);

% Separation into modelling and/or validation
if dataf==0        % Data is for both modeling and validation
    n=floor(k*.75);
    Y=-ones(n*m,m);
    x_m=zeros(n*m,z);
    x_v=zeros((k-n)*m,z);
    for i=1:m
        rand=randperm(k);
        for j=1:k    % Separate data for modeling and validation randomly
            if j<=n
                x_m(j+(i-1)*n,:)=x(rand(j)+(i-1)*k,:);
            else
```

```
                    x_v(j-n+(i-1)*(k-n),:)=x(rand(j)+(i-1)*k,:);
                end
            end
            Y(1+(i-1)*n:i*n,i)=1;     % Dummy matrix
        end
    elseif dataf==1      % Data is only for modeling
        x_m=x;
        x_v=[];
        Y=-ones(k*m,m);
        for i=1:m
            Y(1+(i-1)*k:i*k,i)=1;     % Dummy matrix
        end
    else
        if dataf==2      % Data is only for validation
            x_m=[];
            x_v=x;
        else
            error('Data set flag error')
        end
    end

    if dataf~=2           % Data will be fitted to plsda

        % Eliminate mean
        Xm=mean(x_m);
        for i=1:size(x_m,1)
            x_m(i,:)=x_m(i,:)-Xm;
        end

        % Fit model using computational optimization or run it once
        if opf==1         % Optimize number of components
            [beta,Sm,PRESS,g,cumvar,BIC]=optimize(x_m,Y,m);
        elseif opf==0     % Do not optimize number of components
            % Determine fitted model coefficients using plsda function
            [beta,Sm,g,cumvar]=plsdafun(x_m,Y,opf);
        else
            error('Optimization flag error')
        end
    end

end
```

- Plsdafun function

```
function [ beta,Sm,g,cumvar ] = plsdafun( x_m,Y,opf,g )
%plsdafun Performs the pls2 algorithm and estimates prediction errors.
%   Given modelling data, plsdafun runs a PLS algorithm similar to NIPALs
%   and determines the confidence intervals of the model as well as its
%   accuracy on a validation data set.

if (exist('opf','var')==0) % Not optimizing is DEFAULT
    opf=0;
end

[n,k]=size(x_m);         % n: observations     k: variables
m=size(Y,2);             % Number of classes

% Give x column of 1
x_m=[ones(n,1) x_m];
k=k+1;

% PLS2 algorithm

% Determine number of latent variables if not specified (optimizing or not)
if  opf==0   % Not optimizing
    if exist('g','var')==0
```

```matlab
        if n>k          % Number of lv is one fifth the # observations or
variables
            g=floor(1/5*k);
        else
            g=floor(1/5*n);
        end
    end
elseif opf==1  % Optimizing through cross-validation
    if exist('g','var')==0
        if n>k
            g=k;
        else
            g=n;
        end
    end
else
    error('Optimization flag error')
end

% Preallocate size for speed
if opf==1
    beta=zeros(k,m,g);
else
    U=zeros(n,g);
    T=U;
    P=zeros(k,g);
    Q=zeros(m,g);
    W=P;
    C=zeros(g,g);
end
Ujp1=zeros(n,1);

% Set starting point for iterations
Xj=x_m;
Yj=Y;
for j=1:g
    Uj=Y(:,1);            % Set as an arbitrary column of Y
    flag=0;
    shortcircuit=0;
    while (norm(Uj-Ujp1)>0.00001) && shortcircuit<1000;
        if flag==0
            flag=1;
        elseif flag==1
            Uj=Ujp1;
        end
        W(:,j)=Xj'*Uj/norm(Xj'*Uj);
        T(:,j)=Xj*W(:,j);
        Q(:,j)=Yj'*T(:,j)/norm(Y'*T(:,j));
        Ujp1=Y*Q(:,j);
        shortcircuit=shortcircuit+1;
    end
    U(:,j)=Ujp1;
    C(j,j)=T(:,j)'*U(:,j)/(T(:,j)'*T(:,j));
    P(:,j)=Xj'*T(:,j)/(T(:,j)'*T(:,j));
    Xj=Xj-T(:,j)*P(:,j)';
    Yj=Yj-C(j,j)*T(:,j)*Q(:,j)';
    % Calculate beta for each iteration
    if opf==1
        beta(:,:,j)=W*(P'*W)^-1*C*Q';
    end
end

% Estimate % variability
xvar=zeros(1,g);
yvar=zeros(1,g);
cumvarx=zeros(1,g);
cumvary=zeros(1,g);
```

```matlab
flagthreshx=0;
flagthreshy=0;
for j=1:g
    xvar(j)=T(:,j)'*T(:,j)*P(:,j)'*P(:,j)/trace(x_m'*x_m);
    Yj(:,:,j)=[T(:,1:j) zeros(n,g-j)]*[C(:,1:j) zeros(g,g-j)]*...
        [Q(:,1:j) zeros(m,g-j)]';
    yvar(j)=1-trace((Y-Yj(:,:,j))'*(Y-Yj(:,:,j)))/trace(Y'*Y);
    if j==1
        cumvarx(j)=xvar(j);
    else
        cumvarx(j)=cumvarx(j-1)+xvar(j);
    end
    cumvary(j)=yvar(j);
    if cumvarx(j)>0.95 && flagthreshx==0
        threshx=j;
        flagthreshx=1;
    end
    if cumvary(j)>0.95 && flagthreshy==0
        threshy=j;
        flagthreshy=1;
    end
end
cumvar=[cumvarx;cumvary]';

if opf==0      % Not optimizing, calculate model statistics

    % Calculate the beta
    beta=W*(P'*W)^-1*C*Q';

    % Calculate the model estimation Y
    Y_m=x_m*beta;

    % Estimate model results statistics
    classes=size(Y,2);
    for i=1:classes
        truecount=0;
        falsecount=0;
        for j=1:size(Y,1)
            if Y(j,i)==1
                truecount=truecount+1;
                true(truecount,i)=Y_m(j,i);
            elseif Y(j,i)==-1
                falsecount=falsecount+1;
                false(falsecount,i)=Y_m(j,i);
            end
        end
    end
    Smt(1,:)=mean(true);
    Smt(2,:)=std(true);
    Smf(1,:)=mean(false);
    Smf(2,:)=std(false);
    Smt % Print statistics for 'True'
    Smf % Print statistics for 'False'
    Sm=[Smt,Smf];

else
    Sm=[];
end;

end
```

- Probdf function

```matlab
function [ Y ] = probdf( X,nbins )
%probdf Estimates the probability density function curve in discrete bins.
%   Creates a histogram for 'X' with number of bins 'nbins'.
```

```
% Initialization of variables
[m,n]=size(X);
Xmin=zeros(m,n);

% Determine number of bins
if nargin==1
    nbins=floor(sqrt(m));
end
Y=zeros(n,nbins);

% Normalize spread from 0 to 1 and determine step size
for j=1:n
    Xmin(:,j)=min(X(:,j));
    X(:,j)=X(:,j)-Xmin(:,j);
    h(j)=1/nbins;
    X(:,j)=X(:,j)/max(X(:,j)); % Normalize
    for i=1:m                    % Cycle for counting frequency of occurrence
        k=0;
        while k<nbins
            k=k+1;
            if X(i,j)<=(h(j)*k)
                Y(j,k)=Y(j,k)+1;
                k=nbins;
            end
        end
    end
    Y(j,:)=Y(j,:)/m;             % Normalize to total sum = 1
end

end
```

- Rearrange function

```
function [ Y ] = rearrange( X,f,o,l,f1,f2 )
% rearrange Returns values of the statistical and wavelet analysis in a
% vertical vector.
%   Organizes all variables returned by the statwave function in a single
%   horizontal vector for use in other programs.

tag=1;
swpar

% Determine number of classes
c=size(X,2);

% Set up the data matrix
for i=1:c
    [ca_s,cd_s]=statwave(X(:,i),o,l,f1,f2); % ca_s may contain original X
    if i==1
        % Preallocate size for speed
        [sa sm]=size(ca_s);       % cd_a and cd_s have same # columns
        Y=zeros(c,sm*l);

    end

    Y(i,1:3)=ca_s(1,2:sm); % signal, ca at level l and cd stats    2
    Y(i,4:7)=ca_s(sa,:); % signal, ca at level l and cd stats    2

    for k=1:l                  % Rearrange them in a single row of Y

        Y(i,3+(1+(k)*sm):3+(k+1)*sm)=cd_s(k,:); % signal, ca  at  level  l  and
cd stats    2

    end
```

```
end

end
```

- Statmom function

```matlab
function [Mom] = statmom(X,flag)
% statmom returns the mean, standard deviation, skewness and kurtosis.
%   Given a vecotr or matrix X, statmom will calculate the four statistical
%   moments, which are: mean, standard deviation, skewness and kurtosis. If
%   desired, values will be corrected for bias.
%
% All four values will be recorded on a single matrix with 4 columns, which
% take the form [Mean,StandarDeviation,Skewness,Kurtosis] = statmom(Data
% vector or matrix, Biased or Unbiased flag)
% Unbiased  = 0    uses n-1 (Default)
% Biased    = 1    uses n

if nargin==1                    % Set default if not specified
    flag=0;
end

% Initialize variables
cum(1)=0;                       % Will be used in StanDevi calculation
cum(2)=0;                       % Will be used in Skewness calculation
cum(3)=0;                       % Will be used in Kurtosis calculation

if size(X,1)==1                 % Data in a horizontal array or scalar
    n=size(X,2);
    Mom(1)=sum(X)/n;
    for i=1:size(X,2)
        cum(1)=cum(1)+(X(1,i)-Mom(1))^2;
        cum(2)=cum(2)+(X(1,i)-Mom(1))^3;
        cum(3)=cum(3)+(X(1,i)-Mom(1))^4;
    end
elseif size(X,2)==1             % Data in a vertical array
    n=size(X,1);
    Mom(1)=sum(X)/n;
    for i=1:size(X,1)
        cum(1)=cum(1)+(X(i,1)-Mom(1))^2;
        cum(2)=cum(2)+(X(i,1)-Mom(1))^3;
        cum(3)=cum(3)+(X(i,1)-Mom(1))^4;
    end
else                            % Data in a matrix
    Mcum=0;
    for i=1:size(X,2)
        Mcum=Mcum+sum(X(:,i));  % Mean for data in a matrix
    end
    n=size(X,1)*size(X,2);
    Mom(1)=Mcum/n;
    for i=1:size(X,2)
        for j=1:size(X,1)
            cum(1)=cum(1)+(X(j,i)-Mom(1))^2;
            cum(2)=cum(2)+(X(j,i)-Mom(1))^3;
            cum(3)=cum(3)+(X(j,i)-Mom(1))^4;
        end
    end

end

% Biased
Mom(2)=sqrt(cum(1)/n);
Mom(3)=cum(2)/(n*Mom(2)^3);
Mom(4)=cum(3)/(n*Mom(2)^4);

% Unbiased correction
```

```matlab
if flag==0
    if n>3
        Mom(2)=sqrt(n/(n-1))*Mom(2);
        Mom(3)=sqrt(n*(n-1))/(n-2)*Mom(3);
        Mom(4)=(n-1)/((n-2)*(n-3))*((n+1)*Mom(4)-3*(n-1));
    else
        error('You cannot correct statistics for unbiased when n<4')
    end
end

end
```

- Statwave function

```matlab
function [ s_ca_stat,s_cd_stat,s_ca,s_cd ] = statwave( s,o,l,f1,f2 )
%statwave determine the statistical moments and wavelet coefficients for
%the analyzed signal
%   s:  data vector to be analyzed
%   o:  daubechies wavelet order
%   l:  number of levels to use for the wavelet analysis
%   f1: correction for biased or not (corrected is default f1=0)
%   f2: normalized or not (normalized is default f2=0)

swpar

% Obtain wavelet transform
[s_ca,s_cd]=wavetran(s,o,l);
maxlength=size(s_cd,1);

% Obtain statistical moments for original data and wavelet transform
s_ca_stat=statmom(s,f1);        % Statistics for original data

for i=1:l                       % Statistics for wavelet transform
    stop=floor(maxlength/(2^(i-1)));
    s_ca_stat(1+i,:)=statmom(s_ca(1:stop,i),f1);
    s_cd_stat(i,:)=statmom(s_cd(1:stop,i),f1);
end

% Normalize
if f2==0

    auxv=normalize([s_ca_stat(l+1,:);s_cd_stat]); % ca at level l and cd
    s_ca_stat(l+1,:)=auxv(1,:);
    s_cd_stat=auxv(2:end,:);

end

end
```

- Swpar script

```matlab
%swpar Determines the parameters to be used for wavelet transforms and
%flags
%
%   This function is called from svminput, statwave, statmom, wavetran and
%   plsda. It returns the proper parameters to be used in the mentioned
%   functions. Any combination from none to all 4 parameters may be
%   specified and the remaining parameters will be set to their defaults.
%   Defaults are:
%   o=4;
%   l=5;
%   f1=0;
%   f2=0;
%
%   The variable tag refers to how many variables beyond X (data set) the
%   specific function uses.
```

98

```matlab
%
%   Depending on the number of inputs, several actions can be performed.
%   2 inputs:
%      A- o and l are specified
%      B- 1 specified by name (o, l, f1 or f2)
%   3 inputs:
%      A- o, l and f1 are specified
%      B- o is specified, plus 1 specified by name (l, f1 or f2)
%   4 inputs:
%      A- All parameters are specified
%      B- o and l are specified, plus 1 specified by name (f1 or f2)
%      C- Any 2 parameters specified by name

if exist('tag')==0
    tag=0;
end

nspecinp=nargin-tag-1;          % Number of specified inputs

switch nspecinp
    case 0                      % All must be default.
        o=4;
        l=5;
        f1=0;
        f2=0;
    case 1                      % Only o is specified, rest are default.
        l=5;
        f1=0;
        f2=0;
    case 2                      % Default flags or specific parameters.
        if isnumeric(l)==0      % Input o is specified by name
            switch l
                case 'o'        % Input o is o; l, f1 and f2 default.
                    l=5;
                    f1=0;
                    f2=0;
                case 'l'        % Input o is l; o, f1 and f2 default.
                    l=o;
                    o=4;
                    f1=0;
                    f2=0;
                case 'f1'       % Input o is f1; o, l and f2 default.
                    f1=o;
                    o=4;
                    l=5;
                    f2=0;
                case 'f2'       % Input o is f2; o, l and f1 default.
                    f2=o;
                    o=4;
                    l=5;
                    f1=0;
                otherwise       % The letter given for name is incorrect.
                    error('specified parameter name is incorrect')
            end
        else                    % o and l are specified, flags are default.
            f1=0;
            f2=0;
        end
    case 3                      % o specified and either l+f1 or other named.
        if isnumeric(l)==0
            error('input l must be an integer')
        end
        if isnumeric(f1)==0
            switch f1
                case 'o'        % o is first and second input, error.
                    error('wavelet order specified twice')
                case 'l'        % Input l is l; f1 and f2 default.
```

```matlab
                    f1=0;
                    f2=0;
                case 'f1'      % Input l is f1; l and f2 default.
                    f1=l;
                    l=5;
                    f2=0;
                case 'f2'      % Input l is f2; l and f1 default.
                    f2=l;
                    l=5;
                    f1=0;
                otherwise      % The letter given for name is incorrect
                    error('specified parameter name is incorrect')
            end
        else                   % o, l and f1 are specified, f2 is default.
            f2=0;
        end
    case 4         % Have to allow for any 2 to be specified
        if isnumeric([o l f1 f2])==0 % At least 1 input is a name
            if isnumeric(o)==0
                error('input o must be an integer')
            end
            if isnumeric(f1)==0
                error('input f1 must be an integer')
            end
            if isnumeric(l)==0    % Input o is specified by name
                switch l
                    case 'o'       % Input o is o, no action
                        firstinp='o';
                        clear l
                    case 'l'       % Input o is l
                        l=o;
                        firstinp='l';
                        clear o
                    case 'f1'      % Input o is f1
                        f1=o;
                        firstinp='f1';
                        clear o
                        clear l
                    case 'f2'      % Input o is f2
                        f2=o;
                        firstinp='f2';
                        clear o
                        clear l
                    otherwise      % The letter given for name is incorrect
                        error('specified parameter name is incorrect')
                end
                if isnumeric(f2)==1 % Input f1 is specified by name
                    error('input f2 must be a parameter name')
                else
                    if strcmpi(firstinp,f2)==1
                        error('variables l and f2 specify same parameter')
                    else
                        switch f2
                            case 'o'       % Input f1 is o
                                o=f1;
                                clear f1
                                clear f2
                            case 'l'       % Input f1 is l
                                l=f1;
                                clear f1
                                clear f2
                            case 'f1'      % Input f1 is f1, no action
                                clear f2
                            case 'f2'      % Input f1 is f2
                                f2=f1;
                                clear f1
```

```matlab
                                otherwise        % The letter given for name is
incorrect
                                    error('specified     parameter     name     is
incorrect')
                            end
                    end
                end
                % Must specify the 2 remaining parameters
                if exist('o')==0
                    o=4;
                end
                if exist('l')==0
                    l=5;
                end
                if exist('f1')==0
                    f1=0;
                end
                if exist('f2')==0
                    f2=0;
                end
            else                    % Input o and l specified, f1 is named
                if strcmpi(f2,'f2')==1
                    f2=f1;
                    f1=0;           % Default
                else
                    if strcmpi(f2,'f1')==1
                        f2=0;       % Default
                    else
                        error('input name f2 is incorrect or repetitive')
                    end
                end
            end
        end
    case 5

    case 6

end
```

- Wavetran function

```matlab
function [ ca,cd ] = wavetran( s,o,l )
% wavetran calculates the discrete wavelet transform for a Daubechies
% wavelet of order 1 to 4 of a signal for various levels
%   s: signal to be analyzed
%   o: daubechies order
%   l: number of levels to be calculated

switch nargin
    case 1                     % Default wavelet: order=4 level=5
        o=4;
        l=5;
    case 2                     % Order specified, default level=5
        l=5;
    case 3
        if lower(l)=='l'       % Level specified, default order=5
            l=o;
            o=4;
        end
end

% Retrieve filters. L is low pass filter, H is high pass filter
[L,H]=daubfilt(o);

% Preallocate size for speed
ca=zeros(floor(length(s)/2),l);
```

```matlab
cd=ca;

% Compute transform for multilevel cycle
for j=1:l
    % Assume periodicity for signal (copy beginning of signal to the end)
    s(length(s)+1:length(s)+2*o-1)=s(1:2*o-1);
    m=length(s);
    k=m-2*o+1;
    % Use low and high filters
    for i=1:k
        caj(i)=L*s(i:(i+2*o-1)); % L gives approximation coefficients
        cdj(i)=H*s(i:(i+2*o-1)); % H gives detail coefficients
    end
    % Downsample
    caj=caj(2:2:end);
    cdj=cdj(2:2:end);
    % Turn vertical and set up for next level
    ca(1:length(caj),j)=caj';
    cd(1:length(cdj),j)=cdj';
    s=caj';
end

end
```