An Intelligent Control Strategy for Container Filling Operations

Martyn Jeffries BEng (Hons.)

A thesis submitted in partial fulfilment of the requirements of the Nottingham Trent University for the degree of Doctor of Philosophy

This research programme was carried out in the
Department of Mechanical & Manufacturing Engineering,
Faculty of Engineering and Computing,
The Nottingham Trent University,
Burton Street, Nottingham NG1 4BU, UK

July, 2000

Abstract

As the need to reduce waste becomes an increasingly dominant force in the modern manufacturing world, the search for a reliable and accurate method to control container filling has been brought to the fore. Although modern bottling plants are sophisticated in mechanical design, the employed principles no longer provide sufficient scope for the optimisation necessary to meet waste reduction targets.

The use of ultrasound as the primary sensor for bottle filling plant has a great potential to provide the information on the process without significant redesign of mechanical systems or operational methods. However, the signals are strongly influenced by the properties of the fluids through which they pass – factors such as CO_2, thermals and turbulence – leading to a randomising of the measured variable and thus the measurement process is classed as stochastic.

The presence of measurement noise can undermine the efforts of control systems to provide robust control. However, estimation theory in the form of a Kalman filter has the potential to overcome this obstacle, whether the noise is distributed in a Gaussian or non-Gaussian manner. Furthermore, the Kalman filter is a means to permit the use of ultrasound measurement techniques within the disruptive environment of the bottle filling process and allows an intelligent control strategy to be implemented.

A simple test rig is outlined in its use as a means of obtaining information on the characteristics of level measurement. The noise is categorised and a gamma distribution is shown to provide a good approximation to the noise statistics. From these results a single vessel simulation is used to investigate the performance of the proposed Kalman Filter under a number of different operational scenarios. It is shown that under some conditions, like changes in valve closure characteristics, the filter does not provide an adequate solution alone and that in some cases the controller must also be modified.

Fuzzy logic has many benefits for the design of a bottle filling control strategy. It offers good levels of robustness and stability without necessarily having the full knowledge of the system dynamics by using heuristic conjecture. In order to provide an optimal solution, the mathematical transformations that represent the fuzzy process have been investigated. To this end several avenues of modifications have been offered to better represent the needs of the filling process. These have focused on the input-output relationship, allowing a 'Fuzzy Flow Estimator' system to be developed, capable of parallel sampling ultrasonic Doppler shifts, and thus intelligently inferring both individual and global flow rate changes. This information is then used to feedback into the Kalman filters of the individual valve controllers.

Using an expanded version of the single valve plant simulation, several examples are shown on the performance of the Fuzzy Flow Estimator within a multiple valve carousel. The simulation results indicate that the proposed approach offers useful characteristics that would be beneficial for the tracking of both local and global disturbances in flow rate. Furthermore, these flow rates can be exploited by a Kalman filter to improve height or volume estimation at the individual valves, which provides increased robustness and a solid foundation for liquid level control in a bottling environment.

Thanks to:

Dr Eugene Lai (supervisor),

The Department of Mechanical Engineering,

Detlef Plantenberg,

And the Staff and Students of The Nottingham Trent University

Contents

List of Figures

Chapter 1
Introduction

1.1 Background

The bottling industry has relied on mechanical means to facilitate control of liquid levels ever since its inception. However, as the need to reduce waste becomes an increasingly dominant force in the modern manufacturing world, the search for a reliable and accurate method to control container filling has been brought to the fore. Although modern bottling plants are sophisticated in mechanical design, the employed principles no longer provide sufficient scope for the optimisation necessary to meet waste reduction targets. The European bottling industry is regulated by legislation that requires bottles to be filled to within a specified tolerance of intentional quantity. Currently, the inherent variability present within the system often necessitates that this is achieved by overfilling the containers. Although this approach is effective in meeting the regulations, the total wastage from millions of bottles increases costs. It has been estimated that the UK milk bottling industry has spent over £12million in one year on overfilling bottles (Peers, 1992 and Hull et al, 1994).

A large proportion of the wastage from bottling plant can be ascribed to the initial 'start-up' period of the machine. This can be attributed to a number of factors, these predominantly being filling level and product composition, i.e. satisfactory carbonation, pasteurisation etc. Within current operating procedure, the main control effort is focused on ensuring satisfactory product, with only inaccurate bulk measurement techniques for fill level comparison. Therefore, additional control installed at the filling valve would improve delivery efficiency, allowing possibilities of reducing the time duration required to achieve satisfactory plant steady state (Ridgway et al, 1997). This would require dynamic measurement of the filling level, a challenging task in view of the fast process time (< 4 seconds per bottle/can) and the absolute requirement for non-contamination of the food products.

1.2 Plant Design

The overall design of beverage bottling plant is similar at all scales of operation, from small pilot plant, filling one or two vessels at a time, to the large carousel of filling heads at full production scale. The design can be categorised into a number of discrete elements; a pilot plant will be used as an example to describe the function of each element (courtesy of Britvic Ltd, Chelmsford).

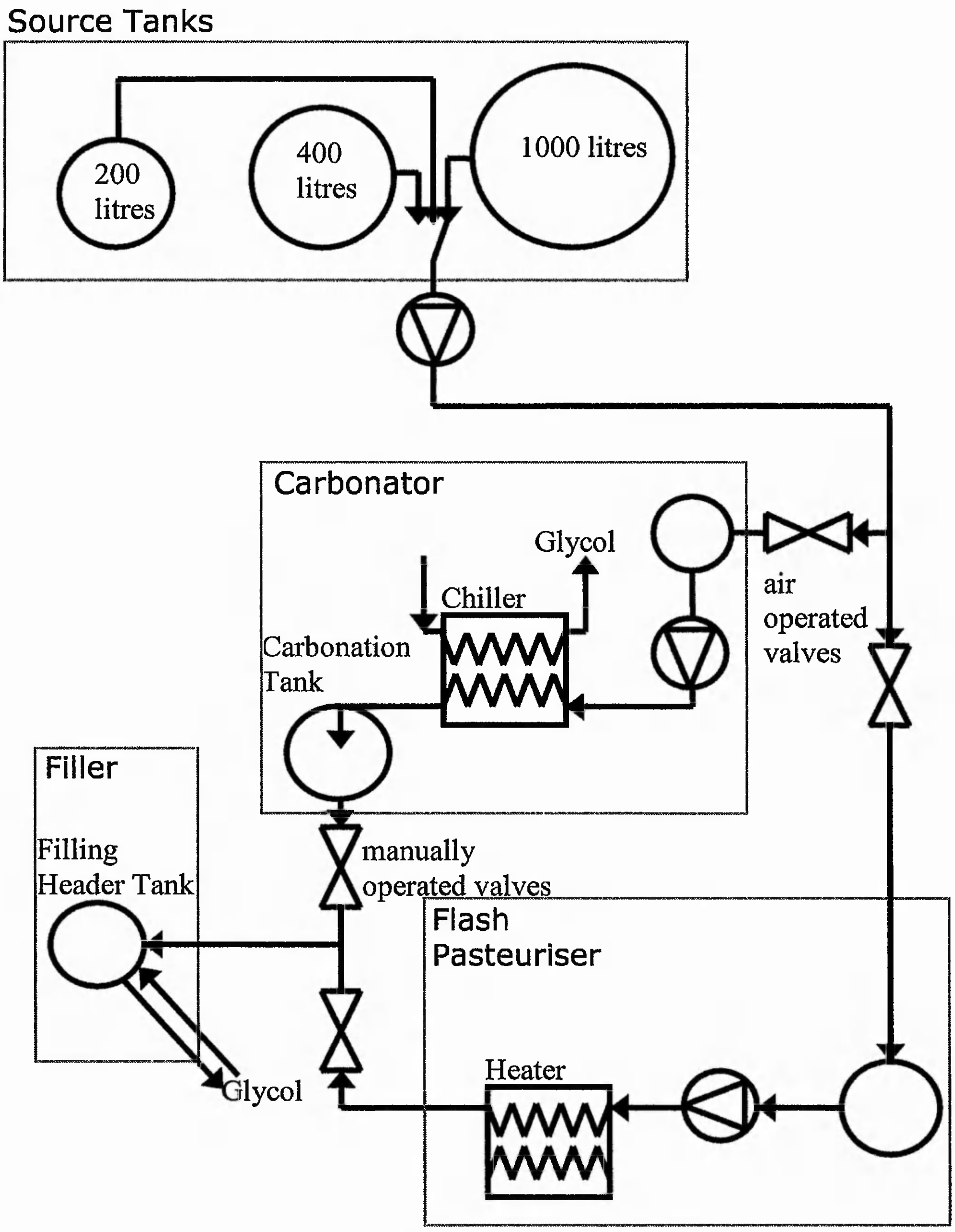


Figure 1.1 Pilot Plant Schematic

Firstly, there are the source tanks in which the raw product is stored, mixed and prepared. On the example plant there are three at 200, 400 and 1000 litre capacities, which allows for small runs of new, trial or special products, (Figure 1.2).

Figure 1.2 Source Tanks of the Pilot Plant

The second stage is one of two types, either carbonation (Figure 1.3) or flash pasteurisation (Figure 1.4). The role of carbonation is not only to carbonate the product but also to sterilise, allowing a long shelf life for the packaged drink. With non-carbonated (still) drinks, like fruit juices, pasteurising fulfils the sterilising role. By heating to around 90°C, bacteria are eliminated. The minimising of contamination of still drinks is a very critical part of filling operation, as the lack of carbonation prevents long term protection, such that any small amounts of contamination introduced at this stage can have significant effects on shelf-life and suitability of products for consumption. To this end it may be necessary to provide 'clean' production lines, with positive pressure rooms not unlike the type used for the production of microelectronics. The pilot plant does not provide the clean environment required by some products but can fill either pasteurised or carbonated drinks by manually changing valves to route product through the necessary stages.

Although the final filling stage is mechanically the same for either pasteurised or carbonated drinks, the product is filled hot in the pasteurised process and chilled in the carbonated approach. This will have an effect on the operation of sensor technologies employed to measure filling levels, and will be discussed in more detail later.

Figure 1.3 Carbonation Tank

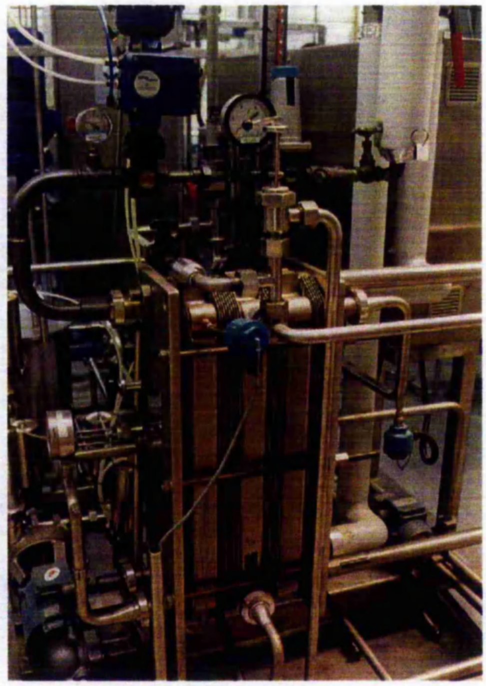


Figure 1.4 Flash Pasteuriser

In general, filling is designed around a carousel of valves, allowing many bottles to be filled in a continuous system (Figure 1.5). The valves are based on mechanical and pressure actuation; pressure to open, mechanical springs to close. The filling level is determined by an open-ended pressure balance tube that projects into the bottle from the valve. As liquid enters the bottle, the pressure is balanced until the end of the tube is covered, which releases the valve to close by spring return. For aluminium and steel cans, a floatation ball is often employed for compactness, as the final fill level is closer to the top of the can. In some plant pre-metered systems are used to ensure correct fill quantity but investment cost can be high, and impractical if existing plant is not scheduled for replacement.

From a control point of view, a major factor contributing to increased waste is plant transient time. It can take up to thirty minutes to achieve satisfactory plant steady-state when a new product run is initialised. The engineer has to balance pressures in the system to achieve desired fill but with no information or control feedback from the valves themselves the task is difficult. Therefore, the skill of the operator is brought very much to the fore.

Even when the mechanical systems employed are operating within design tolerance, an inherent lack of fine control over plant parameters is present. As the plant is subjected to wear and tear the performance efficiency is necessarily affected.

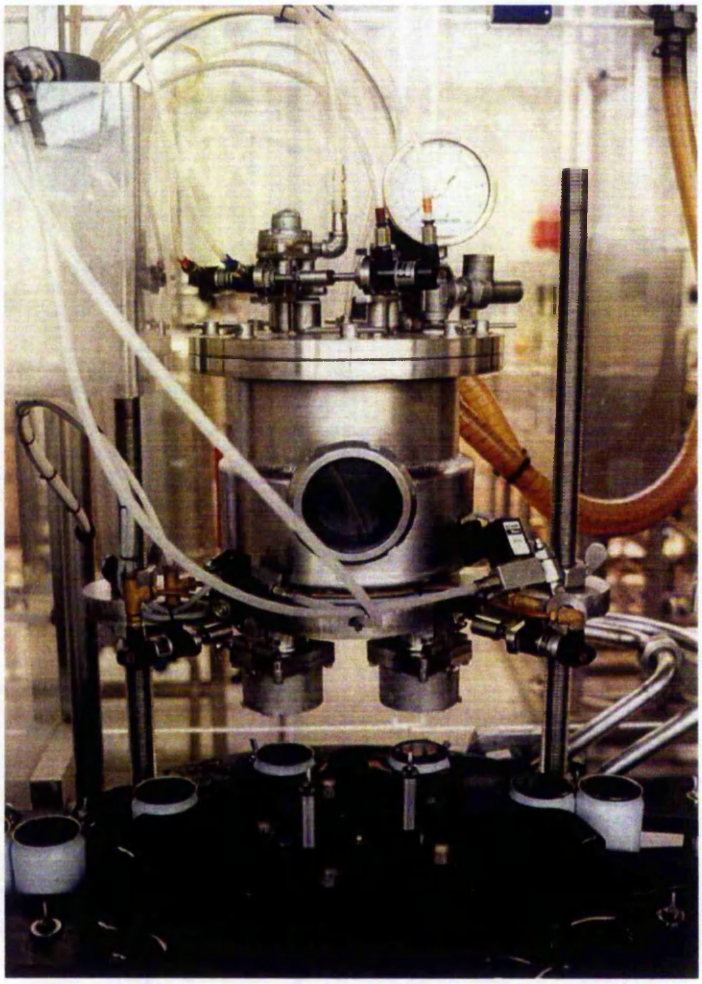


Figure 1.5 Header Tank and Small Filling Carousel

1.3 Project Overview

A project has been undertaken within the Department of Mechanical & Manufacturing Engineering at the Nottingham Trent University to develop a monitoring and control system for bottle filling operations. Its aim is to reduce the wastage from inefficiencies and through a combination of novel sensor technology and intelligent control strategy provide improvements for a wide range of filling operations. The project is split into two main themes, sensor technology and control strategy, tackled as two PhD research projects. Ultrasonic transducers are the main focus for sensor development, with investigation into the physics and mechanics of the measuring process. The work presented here represents the development of a control and monitoring strategy based on ultrasound as a primary measurement tool. The signals derived from the transducer will be complex and to maximise their potential a degree of intelligence will be required within the control strategy.

1.4 Aim

- To develop an intelligent monitoring and control system for bottle filling plant to complement the use of ultrasound as a primary sensor.

1.5 Objectives

- To maximise potential of ultrasonic measurement by using stochastic filtering techniques, i.e. Kalman filter
- To use experiments to quantify effects leading to reduction in sensor performance
- To develop a measurement and control strategy for the filling of a single vessel using a simulation of the filling process, with support from an experimental test rig.
- To expand the simulation to encompass a multiple valve filling carousel

- To develop an intelligent means of controlling the individual valves within carousel, whilst maintaining robust overall control.

1.6 Thesis Overview

As background to the developments undertaken, three sections on sensor technology, estimation theory and Fuzzy Logic are presented. Each of the three subjects is covered in sufficient depth for the assumptions used in the later work to be understood.

'*Sensor Technology*' focuses on the effects of the measuring environment in reducing the performance of ultrasonic transducers. The parameters governing this approach are given; however, development and improvement of the sensor technology for use in the filling process is not covered within this project.

'*Estimation Theory*' reviews the development of systems where observational data does not convey 100% accuracy of the process under scrutiny. The Kalman filter is introduced here, with a description of the basic operation.

The section on Fuzzy Logic covers the background theory of this technique and its use within modelling and control. Further to this, modifications to the basic theory are discussed in preparation for the developments undertaken for control of multiple valve carousels.

The bulk of the thesis is covered in four chapters, Modelling of a Single Filling Vessel, Measurement and Simulation of Liquid Level, Monitoring of Multiple Valve Carousels and Carousel Operational Scenarios.

'*Modelling of a Single Filling Vessel*' concentrates on the development of a simulation of the filling process and the practical application of a Kalman filter. Experiments described in '*Measurement and Simulation of Liquid Level*' on a purposely-built test rig demonstrate the uncertainty present when measuring liquid level with ultrasound in a real environment. These results are used to derive a model for the noise distribution and thus, improve the plant model within the developed simulation, which is shown under a number of different operational conditions.

The '*Monitoring of Multiple Valve Carousels*' builds on the work for the single vessel and develops a novel Fuzzy Logic-based system for improving the performance of individual valves, as well as providing a means of monitoring the plant as a whole. The possibility of using Doppler shift for flow measurement is discussed, along with a feasibility experiment demonstrating its potential. A Fuzzy Flow Estimator, based on the modifications to Fuzzy Logic covered in chapter 4, is discussed in detail. '*Valve Carousel Operational Scenarios*' demonstrates its performance within an expanded plant simulation, capable of modelling a carousel of up to 64 simultaneously filling heads.

The thesis is brought to a close with discussion on the developed technologies, its benefits to the ultrasonic approach and impact on the filling process. Conclusions are drawn on the work, highlighting the milestones and objectives reached.

Chapter 2

Sensor Technologies

2.1 Introduction

The success of any control strategy depends to a large extent on the accuracy and reliability of the sensors used to provide parameter information. In many situations measurements can be relied upon to give a result that can be directly and linearly related to a parameter or with understanding of non-linear factors, be processed to form a very close approximation (e.g. measurements of temperature via a thermocouple or pressure via a manometer). However, in some situations the process under investigation directly undermines the measurements by introducing noise or uncertainty to the sensors. This can lead to a randomising of the measured parameter and is classed as a stochastic process, which will be discussed in greater depth in Chapter 3. The exact manifestation varies from process to process; the stochastic factors can be introduced by the physics of the process, (e.g. vibration), the characteristics of a sensor in a given environment, (e.g. thermal characteristics), or a combination of the two. In this chapter, the effects governing ultrasonic measurements are discussed, with particular reference to the filling process.

2.2 Measurement Options

A number of sensor technologies are available to measure liquid level. They utilise physical parameter changes such as, force, pressure, electrical effects (capacitance & resistance) or indirect measurements using microwaves, infrared, nuclear, thermal or ultrasound (Cho, 1982). Each approach has its own benefits and drawbacks; however, a single constraint ultimately affects the selection – the current design of bottling plant. The space available for sensor mounting is minimal, which is the main reason current measurement and monitoring equipment for bottling plant is placed away from the valves at some point down the conveyor. The inherent limitation for feedback control is clear (i.e. transport lag); consequently, the current approach offers only a rejection system to prevent under-filled bottles being sent for distribution – quality control as opposed to process control.

The speed of the process (~5 seconds per vessel) requires any measurement approach to provide minimal interference to process mechanisms and the liquid, which is a food product, must be unchanged and uncontaminated by the process of measurement. Hull et al (1995) identified that timing of ultrasonic echoes reflected off of the liquid surface could provide a means of finding liquid height. The main benefits being that the ultrasound would not affect the liquid at low powers required for level determination and also that the small size of the sensors would allow placement close or possibly within the valve. Although, some limitations are present, which will be discussed, the ultrasonic approach potentially provides a solution requiring minimal redesign to the process or plant.

2.3 Ultrasound

A significant factor on the performance of an ultrasound sensor is the measurement environment. Unlike sensors that measure liquid level directly, ultrasound relies on the presence of an additional carrier substance between the sensor and the intended target. This can be solid, liquid, gas or a combination. Within the bottling process two approaches can be considered for obtaining fill level, measuring from below the surface, perhaps through the base of the bottle or from above from within the bottle neck as part of the valve assembly. The performance characteristics of each approach are different, dependent to a large extent on the properties of the transmission medium and the interface between the sensor and that medium.

The effectiveness with which ultrasound passes through and between substances is governed by the acoustic impedance, z, (defined as a the ratio of maximum acoustic pressure to maximum particle velocity). If the impedance is high (solids) then the substance will support high frequencies with low power, if the impedance is low (gas) then only lower frequencies are supportable. Examples of the impedance of materials likely to be encountered whilst measuring fill levels are given in table 2.1. (Blitz, 1971 & Asher, 1997)

Material	Acoustic Impedance, z (Kg/m^2s)
Air	430
Water	$1.5x10^6$
Glass (bottles)	$13x10^6$
Stainless Steel (valve/mechanisms)	$47x10^6$

Table 2.1 Acoustic impedance of materials typically encountered in Ultrasound Measurement

The impedance mismatch between materials governs the amount of power required to pass through a number of different materials and is the cause of echoes that will facilitate level measurement. The amount of ultrasound reflected by a material boundary can be determined easily from knowing the acoustic impedance of the two interfacing materials. The ratio between reflected and transmitted sound can be given in terms of pressure, R_P, where

$$R_p = \frac{z_2 - z_1}{z_2 + z_1} \tag{2.1}$$

Or in terms of intensity, R_I, where

$$R_I = \frac{(z_2 - z_1)^2}{(z_2 + z_1)^2} \tag{2.2}$$

For example, using the impedance for air(z_1) and water(z_2) it can be calculated that 99.9% of the pressure will be reflected from a air-water interface. This effect is necessary for level determination. However, the same physical law means that transmission of ultrasound into the air by the sensor will necessarily be inefficient from the impedance difference between the transmitter material and the air. Therefore, measurement of level from above the liquid will require higher power transmission than measurement from below, where the impedance mismatch between the transmitter, bottle and liquid will be much smaller, irrespective of the frequency used.

2.4 Transducer Types and Performance

The exact method of interfacing between sensor and the environment will depend on the type of transducer chosen. Blitz (1971) described five different types of transducer: piezoelectric, magnetostrictive, mechanical, electromagnetic and electrostatic. Piezoelectric materials in the form of crystals and ceramics provide great flexibility with an obtainable frequency range from 20kHz to 10GHz and are relatively inexpensive. The other types of transducer have typical frequency ranges below 200kHz, although electrostatic oscillators are able to detect frequencies as high as 200MHz. Electromagnetic are most suitable to low frequency / high intensity applications, where the ultrasound is required to interact with a substance, rather than just measurement. If gas transmission is required then frequencies in the range 20 to 200kHz are most desirable; transmission through solid and liquid uses higher frequencies, typically 1-10MHz. To be able to perform in either of these roles only piezoelectric is suitable. Mattiat (1971) details the construction of piezoelectric crystals and ceramics and their characteristics. These types of transducer are described as having a high Q value, which means they are sharply resonant in a single frequency with low damping. Thus, transducers would have to be selected to meet the requirements of placement, i.e. high frequency for transmission through solids or liquids or low frequency for transmission though gas. Although the efficiency is low when transmitting to the air, the interface (gas against transducer) is consistent. To transmit into the base of the bottle and further into the liquid then a solid to solid interface will not provide sufficient contact consistency to allow effective transmission. It is therefore necessary to use an intermediate agent such as gel, oil or other liquid between the transducer and bottle to provide a reliable interface. Within the constraints of the fast process time, it would be necessary to have a robust method for achieving this interface; this could perhaps be achieved with highly viscous fluids or non-newtonian substances, such as 'silly putty'. This would allow moulding of the interface at each interaction to compensate for inconsistent application. However, a further possibility would be to place the transducer below the surface, but from within the bottle. This would most likely entail a modification or replacement of the pressure balance tube to allow the mounting of a sensor. It is therefore simpler to use an air transmission approach initially, due to the reduced design constraints.

2.5 Sensor Placement

Placement of an air transmission sensor could be configured to meet one of two commonly employed designs: open surface measurement or the use of a still or waveguide (Asher, 1997).

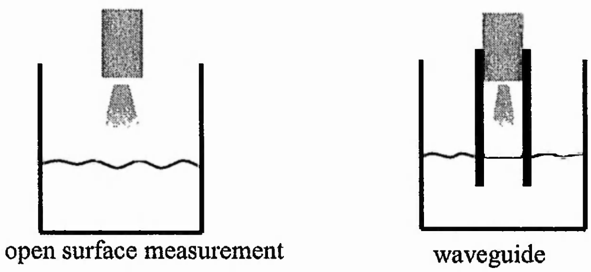


Figure 2.1 Ultrasonic sensor placement options for measuring liquid height

With open surface measurement, the sensor is placed at a convenient location pointing at the surface. The most critical factor in placement being to ensure the measuring range is outside the near field dead-band of the transducer. The closer the diameter of the transducer is to the wavelength the smaller the dead-zone (see Section 2.7), typically for this type of application the near field can be expected to be less than 20mm.

The signal will be vulnerable to two further problems. Firstly, spurious echoes from surfaces other than the liquid such as valve parts and internal bottle surfaces, and secondly, intermittent or lost echoes caused by the agitated liquid surface. The most problematic effect of spurious echoes is destructive interference with returning waveforms, but careful siting of the transducer can reduce this. The agitation of the surface by incoming liquid will be of concern for all methods, whether measuring from above or below the surface. For air transmission, the use of a waveguide goes some way to reduce the surface movement. Sometimes called stills, they keep surface waves from ingressing on the measurement area. A true waveguide has a constant cross-sectional dimension less than or equal to the wavelength of the ultrasound (Lynnworth, 1989 & Asher 1997). However, any tube will necessarily help to reduce the spurious echoes and surface agitation. In addition, within the design of a waveguide a benchmark surface can be incorporated (whose distance from the sensor is fixed and known) to allow compensation for speed of sound changes without additional sensors for temperature, pressure etc.

The waveguide approach is also suitable for measurement from below the surface if performed from within the bottle. Transmitting from the base of the bottle will be vulnerable to the same surface agitation as the air transmission open surface. Nevertheless, it is important to consider how much of the surface is actually interrogated by the ultrasound, and this depends on the size of the transducer, the frequency and the measuring distance.

2.6 Limitations and Problems

A significant problem facing the use of ultrasound in any configuration will be the attenuation and scattering of the signal between sensor and surface. In the case of air transmission, two factors reduce performance.

The first is the presence of carbon dioxide, which is used to pressurise the space above the filling liquid to prevent release of dissolved gas in carbonated bottling lines. Unfortunately, carbon dioxide severely attenuates ultrasound in the frequencies considered to be most effective through gas (5 – 200KHz). However, Kinsler (1982) (see Asher, 1997) showed that the presence of 1% water vapour within the carbon dioxide could shift the peak of attenuating frequencies into the megahertz range. Although, the presence of water vapour is highly likely within the bottle (from the incoming liquid), the exact quantities could be difficult to determine and thus the associated characteristics complex to define.

A second problem is the presence of thermals above the liquid, triggered by temperature differences between liquid and pressurising gas. With non-carbonated drinks, it is common practice to pasteurise the product at 90°C before filling to destroy bacteria; this is in contrast to carbonation, which naturally sterilises. The product is filled directly from the pasteurisation stage, allowing very little time for cooling. This leads to two adverse effects that undermine measurement accuracy. Firstly, the lack of a homogeneous temperature to establish the speed of sound. Secondly, thermal boundary layers and isotherms are generated, separating hotter and cooler sections of the gas. These layers, which are constantly changing, cause scattering of the signal and a reduction in signal strength. However, a waveguide would help to reduce this effect.

Measuring through the liquid to the underside of the surface would be subject to scattering and attenuation from the movement of the liquid as it flows around the bottle. This will inevitably lead to a reduced accuracy without measures to reduce the amount of turbulence through which the ultrasound must pass, i.e. using a waveguide. The greatest advantage of measuring from below the surface would be the absence of carbon dioxide gas, which would improve attenuation characteristics.

2.7 Signal Characteristics

Two areas define the shape of the field produced by an ultrasonic transducer: the near field and the far field.

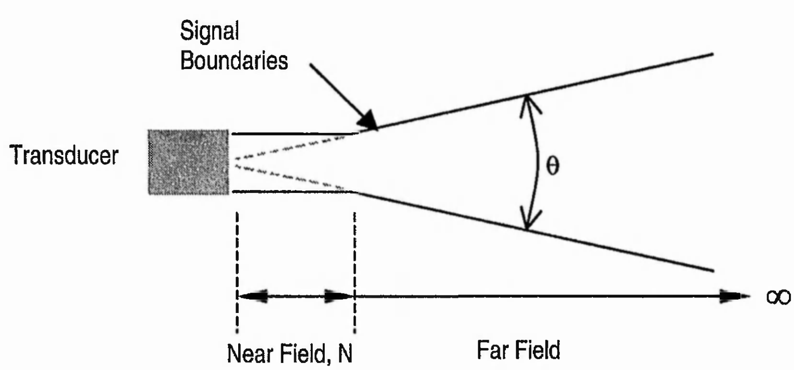


Figure 2.2 A Diagram of a Typical Ultrasonic Sound Field

The length of the near field from the transducer face can by found from,

$$N = \frac{D^2 - \lambda^2}{4\lambda} \tag{2.3}$$

Where, D is the diameter of the transducer and λ is the wavelength of the ultrasound.

For this application it is desirable to operate in the far field. The far field extends from N to infinity and is the area over which measurements can be made. The far field describes a cone, whose inclusive angle can be found from,

$$\theta = 2\tan^{-1}\left(\frac{D}{2N}\right) \tag{2.4}$$

Hence, for a 5MHz transducer of 5mm diameter transmitting through water the near field will extend approximately 20mm from the transducer face and the far field angle, θ, will be slightly less than 14°. With such an inclusive angle the beam will be around 49mm wide at a distance 200mm from the transducer. If the transducer diameter was increased then the interrogated area would be made smaller, although this would necessarily extend the near field.

If a single transducer is employed then echoes can be received from anywhere within the beam. For two transducers (one transmitter, one receiver), the far field cones must intersect for echoes to be detected by the receiver. This intersection is often called the information window, and is the critical parameter for the alignment of sensors to the measurement surface and each other.

2.8 Signal Types

The type of signal transmitted by an ultrasound transducer can be described as either pulsed or continuous. By means of a pulsed signal it is possible to measure the distance of an object or surface from the time it takes for an echo of the original pulse to return to the transducer (time-of-flight). The pulse can be a single frequency, a sweep across a frequency range (chirp) or a broadband signal generated by a voltage spike. The broadband signal contains the greatest energy and is suitable for air transmission. However, piezoelectric transducers, with characteristics such as low damping and resonance in a single frequency, makes them unsuitable for this type of signal. Moreover, only a very narrow frequency sweep can be accommodated. Therefore, this type of transducer is most suited for a tone burst at a single frequency, or for transmission of a continuous waveform.

Continuous signals are used to determine phase and Doppler shifts, from which it is possible to find the speed of moving objects or surfaces. Doppler shift is a characteristic change in the frequency of sound reflected or emitted from a moving object such that an approaching object increases the received frequency and a receding object reduces the frequency. Within the context of the bottling plant, Doppler shifts could be used to determine flow rates from the valve or within the bottle by analysing the speed of the filling liquid. This, depending on the achievable accuracy, could be used to support or substitute time-of-flight measurements (further details can be found in section 7.2). The sensors can be configured for both time-of-flight and Doppler measurements as they are constrained by the same factors (e.g. attenuation, placement), whilst ensuring the angle between the sensors should be small to maximise the relative speed of the surface to the sensors, thus creating the largest Doppler shift.

Transducers can also be placed normal to flow where the resulting frequency profile will be related to the flow profile. Asher (1997) described an ideal situation where the area under a frequency spectrum would be directly proportional to the average flow velocity, and the shape of the spectrum would be an indicator of laminar or turbulent flow (Figure 2.3). However, in practice multiple scattering and far wall reflections make this determination more difficult.

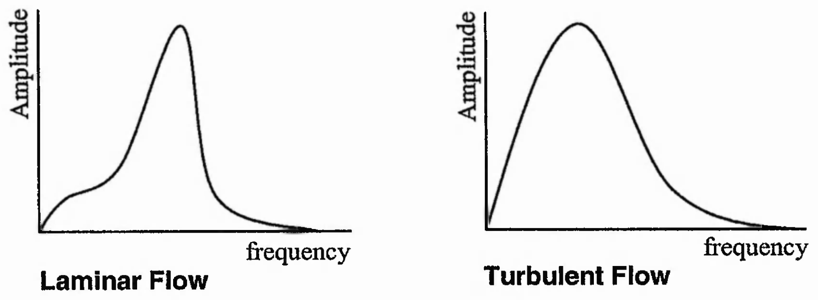


Figure 2.3 A diagrammatic example on the effect of fluid flow type on ultrasonic frequency profile

2.9 Electronics

Whichever ultrasound approach is used, the associated electronics play a significant role in the success of measurement. These can be divided into two sections, transmitter and receiver. The transmitter section must provide high power frequency bursts, whilst in contrast the receiving circuits have to be sensitive to very weak signals of the returning echoes. Therefore, careful design is critical to maximise ultrasonic performance.

The transmitter will be relatively uncomplicated, requiring only a waveform generator and an amplifier to reach the required power. Nevertheless, care must be taken to isolate this circuit from receiver circuits as the electrical noise generated through cross talk could be larger than the returning echo signal.

The receiver typically consists of a number of stages. Firstly, a pre-amp may be used to extract a signal from the sensor, followed by a filter or noise elimination stage to aid echo identification. The signal then may be further amplified to provide an adequate signal for further processing by a computer or dedicated hardware such as a Digital Signal Processor (DSP). Amplification gain may be in the order of 40 to 60dB (100-1000 times) to create a reasonable signal amplitude from the original transducer signal. The degree of processing required will depend on the Signal-to-Noise ratio (SNR) of the returning echo, which in turn will depend on the amount of attenuation introduced by the measuring environment as already discussed. Signal-to-Noise ratio is the ratio between the signal amplitude and the background noise and is normally expressed in decibels (dB). If the SNR is too small the signal can be masked by the noise making further processing difficult. The types and effects of noise are discussed in greater detail in section 3.2.

2.10 Signal Analysis

The simplest means of determining time-of-flight is to measure the number of samples before the echo in a data buffer that has been triggered by the initial sending pulse (Figure 2.4).

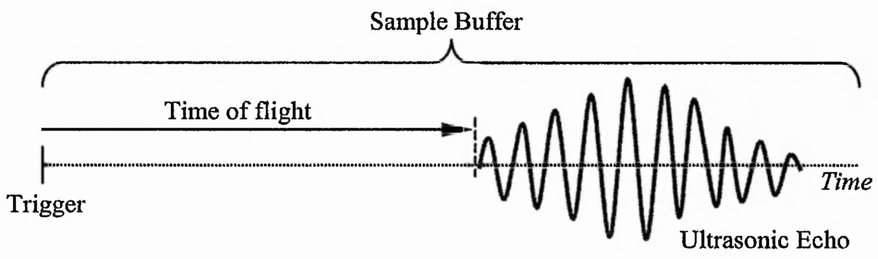


Figure 2.4 Simple method for determining time-of-flight from an ultrasonic echo

Using a threshold to find the echo, the number of samples will correspond directly to the time from a known sampling rate. However, amplitude variations in the echo caused by variable attenuation and scattering of the signal will necessarily affect accuracy. If the echo is difficult to locate due a poor Signal-to-Noise ratio, a mathematical technique called cross-correlation can be employed to locate the echo within a sampled buffer.

Cross-correlation compares the original tone burst with the sampled signal by incremental comparison of the two waveforms over the sample buffer, creating a peak in the cross-correlation function where a signal of the same frequency is present (Figure 2.5). The equation to generate a cross-correlation function can be written as follows;

$$c_{xy}(p) = \lim_{N\to\infty} \frac{1}{N} \sum_{n=0}^{N-1} x(n)y(n+p) \tag{2.5}$$

where, n is a specific sampled point, p is the number of sampling points by which $y(n)$ is offset in respect to $x(n)$, i.e. correlation lag, and $1/N$ is a normalisation scaling factor. (Terrell and Shark, 1996)

For an ultrasonic signal, $y(n)$ is the original pulse and $x(n)$ the received echo. The location of the spike should correspond exactly to the position of the echo, thus the time-of-flight can be found (Figure 2.5).

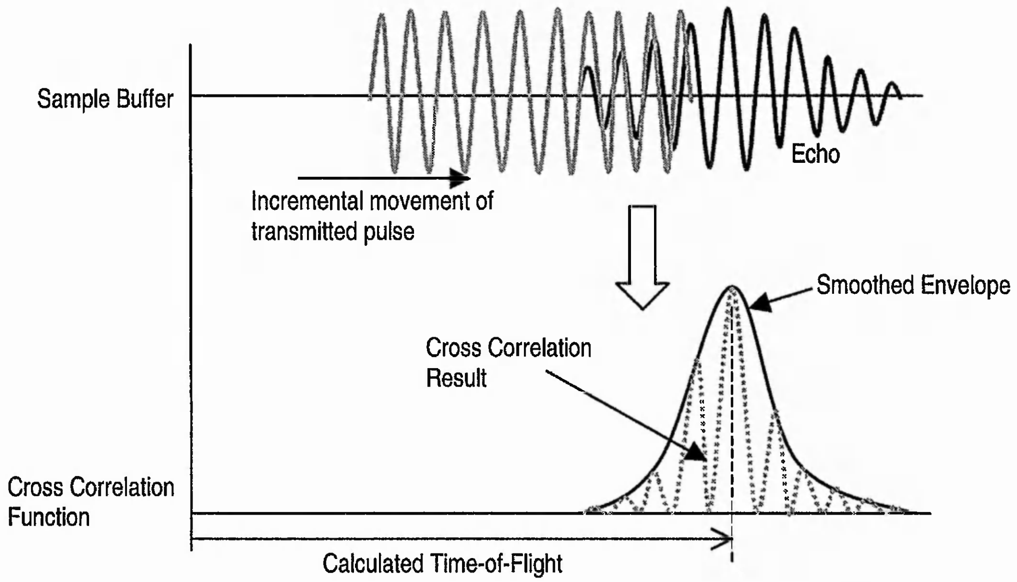


Figure 2.5 A simplified schematic example of the cross correlation method for determining the time-of-flight from an ultrasound pulse-echo signal

However, this approach is vulnerable to variations in echo length, which is often longer than the original tone burst due to ringing effects on the transmitter and receiver (see Figures 6.4 & 6.5 in Chapter 6 for example waveforms). This leads to flatter peak of the function envelope, because the original pulse matches over a wider range, therefore making an exact timing location by this method alone difficult.

The analysis of frequency and hence Doppler shifts can be carried out directly by hardware (i.e. frequency counters) or in software using tools such as discrete Fast-Fourier-Transforms (FFT). FFT is a feature of packages such as Matlab® or in freeware C source code such as FFTW1.3©. However, FFT resolution is limited by the size of the sample buffer and the sampling frequency. The minimal discrete frequency element that can be resolved by FFT is given by the following equation,

$$f = \frac{samplingfreq}{buffersize} \tag{2.6}$$

Where f is the base frequency from which the spectrum will be derived. For example, sampling at 1.5MHz for 8192 samples (8k) gives a base frequency of 183.1Hz. Increasing the buffer size will provide better accuracy but processing time is increased. This means that although software FFT is impractical within real-time applications, it nevertheless still provides a useful post-processing tool. The implications of FFT within this application will be discussed further in Chapter 7.

Figure 2.6 shows a typical error present in a liquid level measurement using ultrasound, in this case with the threshold method (see Section 6.2.2 for further experimental details). The level has been normalised such that a zero reading is equivalent to the best estimate of the true level. It will be shown in later chapters that the distribution of the error can be categorised and used within a simulation of the filling process.

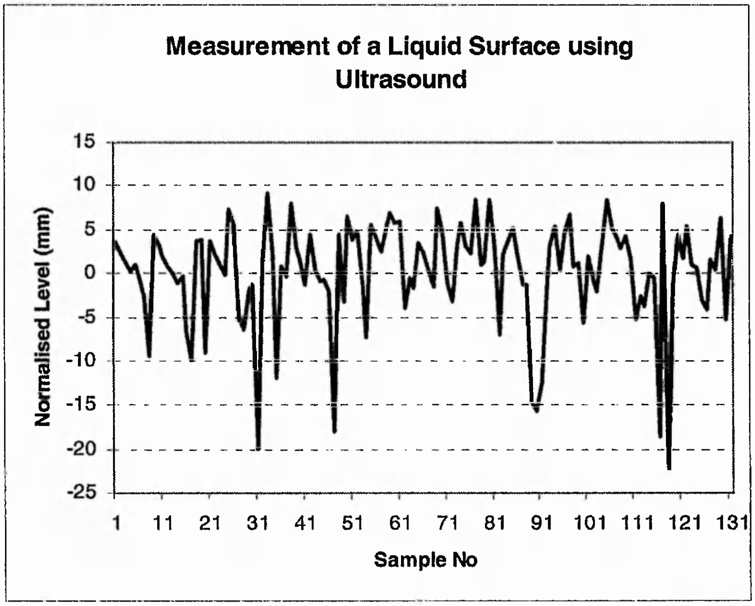


Figure 2.6 An example of ultrasound liquid level measurement using the threshold method for Time-of-Flight calculation

2.11 Summary

The use of ultrasound as the primary sensor for bottle filling plant has a great potential to provide the information on the process without significant redesign of mechanical systems or operational methods. With the ability to measure fill heights and flow rates, the approach has scope for condition monitoring as well as full feedback control. However, the signals are strongly influenced by the properties of the fluids through which they pass. Factors such as CO_2, thermals and turbulence mean theoretical accuracy is reduced. Careful placement and the use of waveguides can reduce some of these effects, but level or flow determination is necessarily impaired within the dynamic filling environment. With filtering and mathematical techniques, such as, cross-correlation, weak signals can be extracted from electrical noise allowing further processing to be performed. However, the environmental variations and processing limitations ensure an inherent uncertainty to any measurements made.

Chapter 3
Estimation Theory

3.1 Introduction

It has been outlined in the previous chapter that measurements made using ultrasound can be subject to interference and noise from the process, such that, without further processing, monitoring or control would be difficult. This chapter discusses how noise can be categorised and the methods commonly used to reduce its effects.

3.2 Types of Noise

Noise within measurements can be attributed to many things; electrical noise from power lines, cross talk from other signals, ground loops and from poor components. In addition, there is noise from process characteristics such as vibration, environmental properties (e.g. ultrasonic attenuation in gas) or uncertainty in the measured parameter, such as agitation on the liquid surface during filling.

Electrical noise can be minimised by careful design and the selection of good quality components. However, the noise from the process (as described in Sections 2.5-2.6) remains dominant and without further processing would undermine any control strategy.

The noise of the filling process exists within the time domain, caused by interference of the reflections of ultrasonic waves from the liquid surface. This will be characterised by an apparent shift in the liquid surface and a real shift of the echo within a sampled window (Figure 3.1).

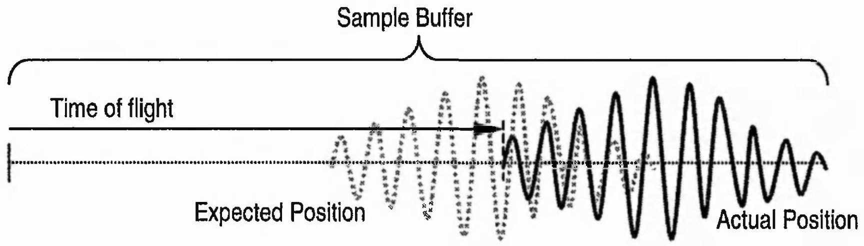


Figure 3.1 Time Shift of Ultrasonic Echo caused by disturbances in the signal path

Noise can be described as coloured or white, where coloured noise interferes in a non-random manner, compared to the purely random effects of white noise. To a certain degree, all noise encountered in real situations has an amount of coloration; as noise described as white assumes an infinite bandwidth, which is practically unachievable. However, in many cases, a white noise model often provides a close approximation and is mathematically simpler to describe within a simulation or model.

The probability distribution of white noise can be described by a large number of different ways, such as Gaussian, Gamma, uniform and many others. Gaussian (or Normal) distribution is the classic random variation within a population of events around a mean, where a standard deviation describes the spread of the distribution (Figure 3.2).

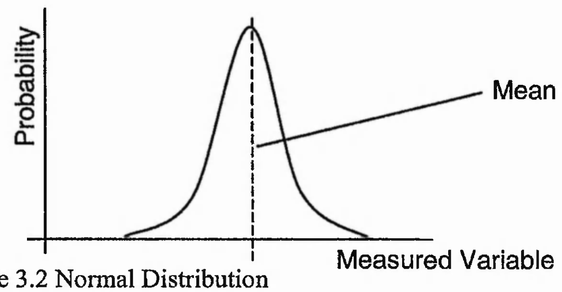


Figure 3.2 Normal Distribution

Uniform distribution is less common in practical situations, but is the usual method of random number generation within a computer. It is where a variable has equal random chance of being any value over a specified range (Figure 3.3).

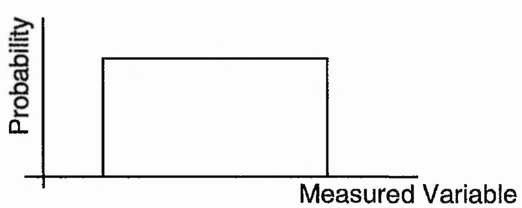


Figure 3.3 Uniform Distribution

A Gamma distribution can model exponential-like and skewed variations depending on two input parameters, α (shape) and β (scale) (Figure 3.4).

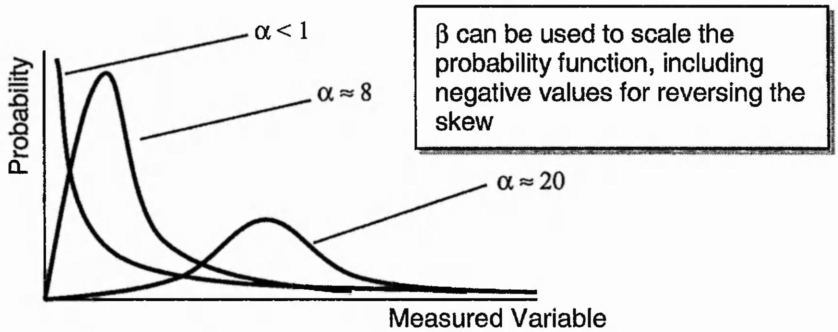


Figure 3.4 Gamma distributions achievable through modification to parameters α and β

As will be shown in Chapter 5, the container filling simulation will be capable of using a number of different noise distributions to model the variations present within the real process. Furthermore, experimental investigation of the noise associated with ultrasonic measurement is discussed in sections 6.2-6.3 and is shown to be most closely represented by the Gamma distribution.

3.3 Estimation Techniques

The identification of parameters and the modelling of processes can be categorised into three types: black box, white box and grey box (Graebe and Bohlin, 1993). The black box assumes no particular similarity between model and process other than a superficial match of data and is often based on linear assumptions; a white box has the opposite approach where the model exactly describes the process, using deterministic relationships. The grey box model falls between the two, allowing partial information to be used successfully by utilising statistical knowledge of process parameter relationships. Whalley and Mitchell (1997) presented a well-described mathematical procedure for the identification of system parameters within the white box approach. However, identification within environments where measurements are made with less than 100% certainty (for example, ultrasound) is called Estimation Theory and is necessarily a grey box approach. It is defined by Siouris (1996) as "*... the application of mathematical analysis to the problem of extracting information from observational data.*"

Although grey box models can be identified and validated without necessarily considering control aspects this is by no means a simple procedure (Holst et al, 1993). However, where the process under examination is relatively simple but is complicated mostly by the presence of noise, a Kalman filter can be employed to provide a stable means of real time parameter identification (Siouris, 1996). The filter is a recursive method for extracting measurements from observational data by comparing an internal model to the incoming data using knowledge of the noise variance (Kalman, 1960; see section 3.4 for further details of algorithms).

Although originally designed for linear systems, 'extending' the filter by linearising the internal model over the iteration time step it can also encompass some non-linear systems. This is used in a wide variety of applications including aircraft tracking, as demonstrated by Tonda and Huignard (1997). The Extended Kalman Filter (EKF) is a common approach for non-linear systems that can be described as close to linear across the filter iteration time step, and the smaller the step the better the representation. However, this approach is unsuitable for processes that cannot be linearly approximated. Daum (1988) discussed and reviewed recursive filters based on generalisations of the original Kalman filter, which aim to tackle non-linear or non-gaussian problems directly, thus removing the approximations and uncertainty of a standard EKF.

In later work, Julier and Uhlmann (1997) have proposed a 'new extension' to the Kalman filter. By using an 'unscented transform', non-linear transformations are more accurately modelled for systems that do not have local linearity – a significant factor in reducing stability in EKFs.

Classification of noise also plays an important role in the optimality and stability of the Kalman filter. Spurious signals can increase the weight of distribution tails, thus destabilising the filter with outlying data points. Pena and Guttman (1988) described an approach for increasing the robustness of the Kalman filter by reducing the influence of outlying data points with the use of a posterior expectation algorithm; this adds an extra step to the filter. This modification is made under the assumption that the noise can still be described as Gaussian, which the filter is primarily designed to accommodate. However, as will be shown with the bottling application in section 6.2, sometimes process noise can be non-Gaussian (e.g. Gamma distributions), which can lead to decreased stability of the filter.

Dwyer (1989) showed how non-Gaussian narrow-band noise can be removed from signals within the application of sonar measurements under arctic ice. The technique involves transforming between time and frequency domains using fast Fourier transforms (FFT). This allows the presence of signal spikes in the measurement data to be reduced, as the subsequent offset generated in the frequency domain can easily be minimised. However, the method is only applicable to post-processing data and lacks the real-time applicability of the Kalman filter. Tufts et al (1989) investigated the presence of combined high-amplitude non-Gaussian and low-amplitude Gaussian components within measurement signals, again with undersea acoustic measurements. The method attempts to simplify the problem by separating each factor and using a suitable technique for noise removal. It is able to classify and distinguish three noise sources: weak Gaussian background, a number of strong sinusoidal-like components occurring randomly and sporadic high intensity bursts (impulse noise). As with Dwyer (1989), the method is useful for identification during post-processing only, rather than real-time applications. However, the techniques offered, such as 'impulse blanking', are reported to be successful and can be valuable in helping in the identification of complex noise sources within the filling operation (i.e. measurements under extreme duress). This would allow for further modifications of the Kalman filter, but this is beyond the scope of the work presented here.

With reference to the Kalman filter, Li and Chu (1997) conducted a rigorous mathematical investigation of correlated noise, (i.e. dependent on the process state or coloured), in a stochastic time-varying system. However, few conclusions or analysis is given on the practical success of filters using the presented equations.

Maryak et al (1997) devised a method for using the Kalman filter in systems with unknown noise distributions. The method removes the need to have full knowledge of the non-gaussian distribution, which is identified as a limitation in other approaches. By attaching confidence regions to the state estimates, (similar to Daum's method (1988) for outlying data points), the approach provides a technique that helps characterise the estimation error in Kalman estimates and shows improvements over other techniques, such as the Chebyshev inequality. In addition, it is easy to implement and computationally undemanding, making it ideal for real-time applications.

The presence of a random bias in the state parameter can have a significant effect on the optimality. Hsieh and Chen (1995) presented a means of overcoming this problem by augmenting the state vector to include a bias term in a two-stage filter. Keller and Darouach (1997) extended this to prove optimality of such a proposed system with the assumption that plant and bias noise are uncorrelated.

From the above review of previous work, it is clear that the Kalman filter can be used in a variety of measurement scenarios with great success, and is especially useful for real-time applications. Moreover, techniques are available to help identify noise sources in non-Gaussian situations, where the standard filter is sub-optimal; thus, assisting the choice of filter modifications. Therefore, the filling controller will rely on a Kalman filter for robust measurement data. The measurement conditions encountered in this study are relatively uncomplicated, compared to what theoretically could be encountered, therefore by implementing a simple version of the filter initially, further enhancements will be possible in the future as the conditions for measurement are made more restrictive.

3.4 The Kalman Filter

Since the original definition of a recursive measurement filter by Kalman (1960) and Kalman-Bucy (1961), the Kalman filter has been an accessible solution to the problem of state estimation from noisy measurements. The filter requires two key ingredients: a model of the process for comparison against incoming measurements and statistical knowledge of parameter errors including those within the model. The errors are represented in covariance matrices and are used to verify if the original measurements or the model predictions are most representative of the process under scrutiny. A Kalman gain matrix balances the two sources (measurement and model) to provide a single robust output. Manipulation of the internal filter states is achieved through five equations which when implemented on a digital computer characterise a discrete Kalman Filter algorithm (Figure 3.5 and Eqs. 3.1-3.5).

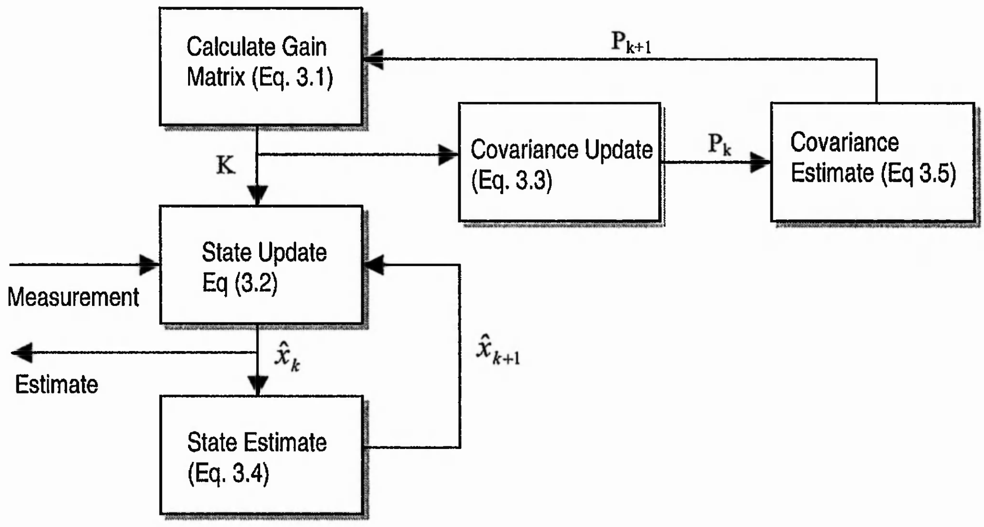


Figure 3.5 Kalman filter schematic showing the interaction of the five main components

Kalman Gain Update

$$K_k = \frac{P_k^- H_k^T}{(H_k P_k^- H_k^T + R_k)} \tag{3.1}$$

State Update

$$\hat{x}_k = \hat{x}_k^- + K_k (z_k - H_k \hat{x}_k^-) \tag{3.2}$$

Covariance Update

$$P_k = (I - K_k H_k) P_k^- \tag{3.3}$$

State Estimation

$$\hat{x}_{k+1} = \Phi \hat{x}_k \tag{3.4}$$

Covariance Estimation

$$P_{k+1}^- = \Phi P_k \Phi^T + S_k \tag{3.5}$$

where,

R	*measurement error matrix - expected noise from measurement represented as a variance about a zero mean.*
S	*plant/system error matrix - as R except noise from plant or system*
P	*error covariance matrix - difference between prediction and actual states*
$\hat{x}$	*state estimate matrix - state variables*
z	*measurements*
Φ	*state transition matrix - the process model*
K	*Kalman gain matrix - balances measurements to predictions based on P*
H	*observation matrix - transforms state variables to observation variables. If state variables are directly measured then H=I (identity matrix).*

The superscript $^{-}$ *represents the matrix in the previous iteration to that denoted in the subscript*

The Kalman parameters, *R, S, P,* etc., are commonly described as matrices. However, if the only input is the measured liquid height, the matrix calculations can be simplified to one-dimensional values and thus the explanation of the main filter characteristics made clearer. (A simple block diagram of the filling process can be found in Section 5.4). Filter calculations are performed dynamically from sampled data in an iterative manner, such that the value of *P* and therefore *K* vary from initial values until steady state values are reached. The numerical difference between *R (measurement error)* and *S (modelling error)* influences whether the Kalman filter 'trusts' the internal model or the external measurements. If *R* is higher than *S* then more weight is given to the internal model, which is characterised by a reduction in the steady state value of *K.* If *S* is larger than *R* then the measurements are given more consideration and steady state value of *K* is increased.

The initial dynamic characteristic of the filter can be altered by changing the value of *P*. A small or zero value of *P* will cause the internal model to be used almost exclusively, thus creating a degree of stability (Figure 3.6). If a large value of *P* is chosen, such that *K* approaches 1, then measurements will dominate initial estimates (Figure 3.7). This is useful if the internal model poorly represents the initialisation phase of a process.

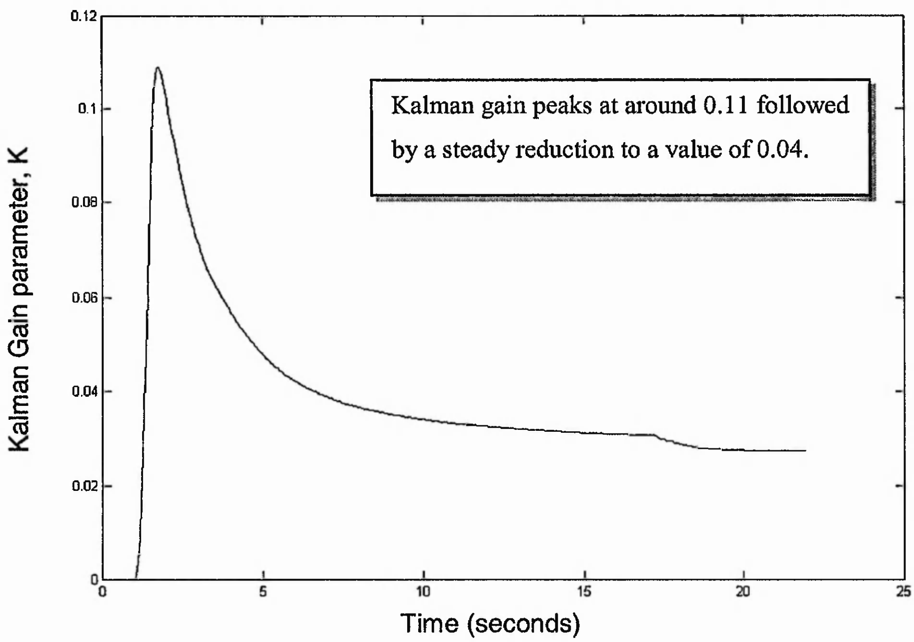


Figure 3.6 Error Covariance, P initialised to 0 and subsequent response of Gain, K

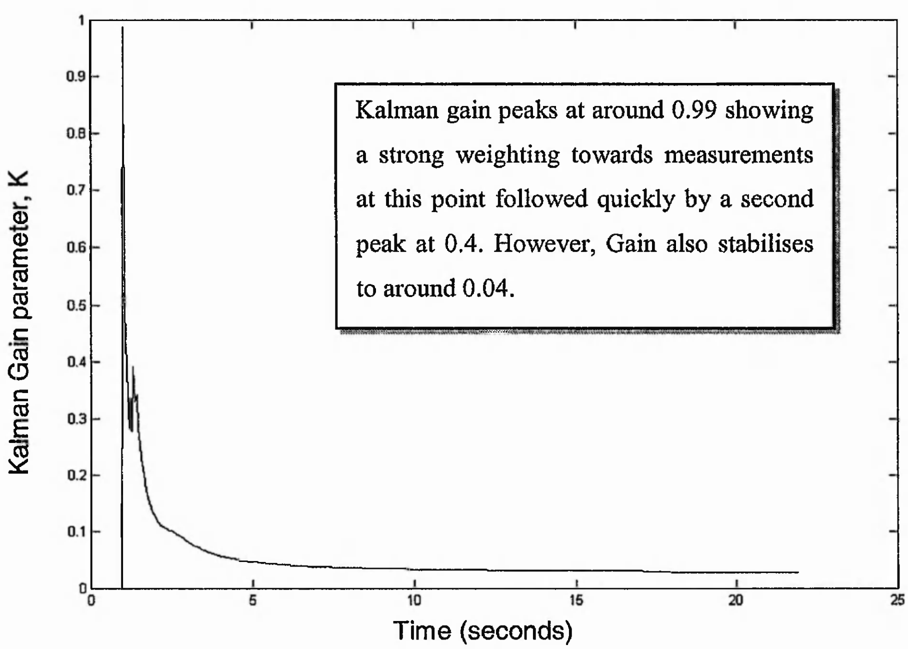


Figure 3.7 Error Covariance, P initialised to 99999 and subsequent response of Gain, K

The internal model, Φ, describes the transformation from the current state to the next state of the filter's input parameters. Critical to the success of the filter, is the accuracy of this model. Although some limitations can be overcome, as already described above, by careful choice of P and using S to describe the difference between the process and the model, overall performance will be maximised by the creation of an accurate model.

The model can be found by any reasonable means of identification, as it does not take into account the influence of noise. Such that, understanding of the underlying physics of the process can be enough to provide a suitable model. In the case of the bottle filling process, knowing the bottle shape (i.e. diameter related to height) and the flow rate is enough information to deduce the height of liquid.

3.5 Application to Liquid Level Measurement

Within applications where many samples can be taken per iteration of the controller, a simple averaging method can be employed to reduce the noise levels on unbiased signals. However, the filling process is fast compared to the sampling rate (~160Hz), such that any meaningful averaging of samples (>10 points) would make it necessary to interpolate between samples to accurately control the final fill height. Thus, a degree of modelling would be required to combat influence of non-linear factors. Therefore the use of a Kalman filter, with its inherent ability to model the process, independent to some extent of the measurements, would be an excellent foundation for building a robust control strategy.

3.6 Summary

The presence of measurement noise can undermine the efforts of control systems to provide robust control. Estimation theory in the form of a Kalman filter has the potential to overcome this potential obstacle, whether the noise is distributed in a Gaussian or non-Gaussian manner.

However, the filter requires an accurate model of the process to maximise performance and non-linear systems have to be linearised to provide any amount of stability. Nevertheless, a degree of flexibility can be afforded if covariance error matrices are initialised with different values (Figures 3.3 & 3.4).

Furthermore, the Kalman filter is a means that permits the possible use of ultrasound measurement techniques within the disruptive environment of the bottle filling process and will allow an intelligent control strategy to be implemented without significant plant redesign.

Chapter 4
Fuzzy Logic

4.1 Introduction

From the previous two chapters it can be seen that a level of uncertainty can be expected within a measuring process. To maintain a high level of robustness it will be necessary to ensure any supervisory control systems are stable in the face of uncertainty. To this end, fuzzy logic presents itself as a prime candidate. A basic theoretical background is covered here, and the developed system covered in depth in chapter 7.

4.2 The Basics

Fuzzy Logic has become a popular tool for creating systems with the ability to quantify variables that have non-discrete states. This allows for increased robustness across transitions between states and a continuous description over the whole of the variable range. The mathematical foundations on which fuzzy logic is based can be found in many texts, including Kosko (1992) and Cox (1994), and therefore will not be covered again here. However, a brief overview of the common methods and terminology will be presented to support the work undertaken.

The components of a rudimentary fuzzy logic system can be described in three sections: input fuzzification, inference and output defuzzification.

4.2.1 Input Parameter Fuzzification

An input parameter is fuzzified by defining a number of sets (or membership functions) over the expected value range to create a '*Universe of Discourse*'. The shape of the sets can be varied but is usually either Gaussian or triangular; the latter more commonly used within computer programs because of its ease of definition. Figure 4.1 shows an example of four sets describing a range of temperature. By finding the degree of membership a parameter value has to each of the defined sets, a set of rules can be applied to decide on an output action or value. The input sets relate variables to membership values by the shape of the defining lines such that a linear relationship is achieved from triangular sets, a non-linear relationship from Gaussian sets.

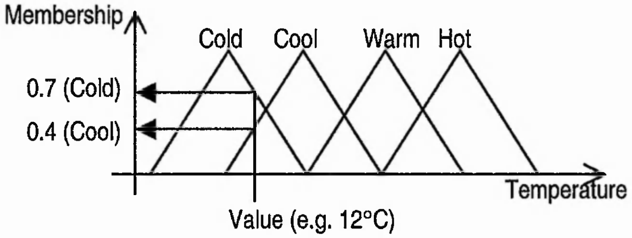


Figure 4.1 An Example of Fuzzy Input Sets

4.2.2 Inference

Inference is carried out by applying a set of rules to the membership values calculated from the current input parameters. The output of the rules is linked to another group of sets describing states of the output parameter. The size of the rule base is dependent on the number of input parameters and the number of sets used to describe each parameter.

If two or more input parameters are used in the same rule, then they can be combined either by using an OR operator or by an AND. Using the OR will result in the maximum membership value being used, AND will give the minimum.

e.g. Membership value

If A is 0.6 **OR** B is 0.2 then C is 0.6

If A is 0.6 **AND** B is 0.2 then C is 0.2

where A and B are input sets (e.g. describing properties such as fill height or volume) and C is the output set (e.g. valve actuation)

If two or more rules refer to the same fuzzy output set, the result can be reconciled by either adding the rule results (additive method) or taking the maximum value (maximum method). If the maximum method is used, then the OR conjecture is redundant within the rule base, as this also takes the maximum between two membership values.

For example, the following rule,

If A is 0.6 **OR** B is 0.2 then C is 0.6

can be written as,

If A is 0.6 then C is 0.6

If B is 0.2 then C is 0.2

because, by applying the maximum method, the rules would resolve to,

C is 0.6

which is the same result as using the OR statement.

This inference method is called min-max, indicating the use of only 'minimum' (AND) rules and 'maximum' rule output combination; this approach is widely used. Nevertheless, in choosing only maximum values, the method ignores some rule outcomes, which can be seen as a limitation. A modification is proposed in section 4.4.1 that maintains the results from all rules, and this is the method on which the fuzzy flow estimator described in section 6.5 is based.

The rule result is linked to the consequent output set by either correlation minimum, where the set is truncated at the given membership value; or by correlation product, where the output set is scaled to the value (Figure 4.2). Correlation minimum is the more commonly cited method.

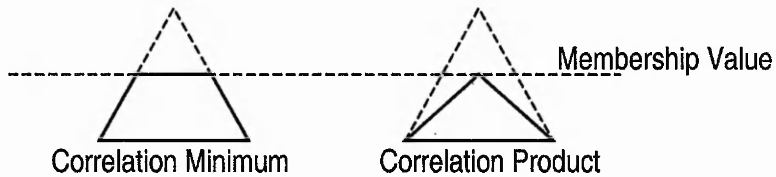


Figure 4.2 Graphical comparison of two commonly used correlation methods

This is applied to all the output sets, thus representing the current output from the rule base (Figure 4.3)

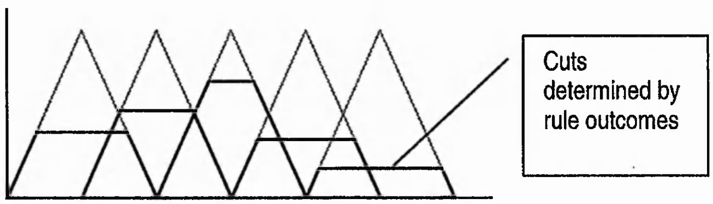


Figure 4.3 Fuzzy Rule Base Outcome Represented by Output Sets

4.2.3 Defuzzification

To create a useable output from the fuzzy logic, the truncated or scaled output sets are combined in a process called defuzzification. Two approaches are most commonly employed to achieve this: Centroid (composite moments) or maximum height (composite maximum).

The centroid method finds the 'centre of gravity' of all the consequent sets (Figure 4.4). This method is widely used, as the defuzzified value tends to move smoothly across the fuzzy output region making it very suitable for control applications.

$$C = \frac{\sum c_i A_i}{\sum A_i} \tag{4.1}$$

where

C *is total centroid (i.e. the defuzzified result)*

A_i *is the area of set i*

c_i *is the centroid for output set i.*

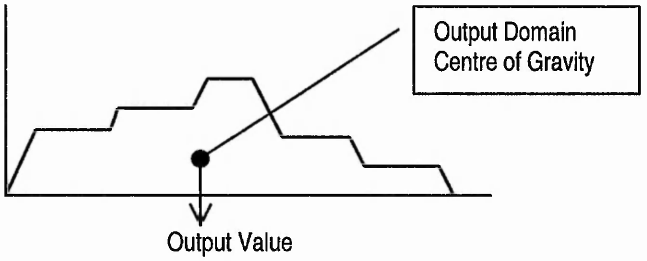


Figure 4.4 A diagram showing how Centroid Defuzzification can be used to find an output value from a number of fuzzy output sets

The maximum height method uses the position of the tallest of the consequent sets as the output value (Figure 4.5). Although this method is sensitive to dominant rules and the final value does not move as smoothly as the centroid method, it is well suited to risk assessment applications (Kosko, 1992). Two further variants exist called average maximum and centre of maximum, which seek to improve smoothness whilst maintaining this approach's main advantage – low computational overheads.

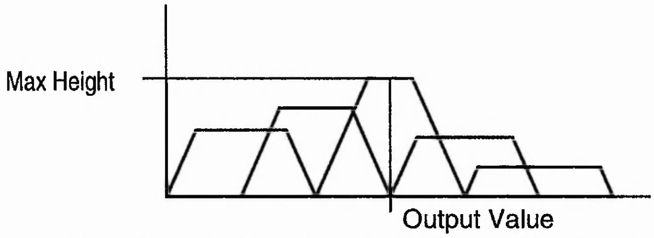


Figure 4.5 The Max Height Defuzzification Method for obtaining an output value

4.3 Fuzzy Modelling and Control

Verbruggen and Brujin (1997) attempted to outline the contribution of fuzzy control to classical or conventional control. A major point of disparity between the two approaches is the apparent lack of mathematical rigour with fuzzy systems. The main conclusion reached is that both methods have benefits and shortfalls, therefore a combination of the two will often yield the best result. Also, fuzzy logic is well suited to supervisory control situations, such as fault diagnosis, where it is able to mimic the decisions of a skilled operator through the use of inference rules. Rondeau et al (1997) issued a note of caution for fuzzy implementers, namely, the choice of defuzzification method should not be made as an arbitrary decision for least output error, but should be chosen so as to theoretically support the process under investigation. In other words, selection of inference or defuzzification method should be carefully considered if truly robust or optimal systems are to be developed. Thus, the modifications to fuzzy logic to be discussed in section 4.4 will support this view, with a more fundamental investigation and understanding of the mathematical transformations undertaken during fuzzy inference.

The fuzzy logic controller is a particularly common approach to applications that are difficult to quantify into discrete states. Takagi and Sugeno (1985) present an approach for linear modelling or control by means of a mathematical tool that seeks to minimise the complexity of the rule base. Kandiah (1996) used this approach within the development of a fuzzy model for predictive control of chemical processes. The limitations of linear modelling were overcome by combining a number of linear sub-models in to an overall non-linear process model. Investigations demonstrated that prediction horizon (the number of iterations or steps into the future) had a significant effect on transient and oscillatory behaviour. However, it also showed that only marginal additional benefit could be gained from attempting to predict over more than five steps in this example; the cost of greater computational processing not being rewarded by performance gains due to small errors in early iterations accumulating to decrease prediction confidence. However, other applications of the proposed method could have different a prediction horizon depending on, for example, accuracy of input variables or the iteration time step.

The generation of a complete control rule set was the major concern of Graham and Newell (1988) in their fuzzy control method for a liquid level rig. By attempting to maintain a fixed set point under the influence of load changes the results provided evidence that fuzzy set theory is comparable to other techniques (PI/feedforward), such that it may be suitable for wider range of applications than commonly assumed for this approach. Further investigations into a fuzzy adaptive control by Graham and Newell (1989) demonstrated the fuzzy control of a first-order process in the presence of noise. Although high levels of robustness were reported for the non-adaptive controller, the adaptive approach required modifications in the form of weighting factors to slow the speed of adaptation. The critical factor determining the inherent robustness of the fuzzy system to noise was the comparative width of the sets to the peak noise level. Noise levels smaller than the width of the set being more easily accommodated than noise levels spanning several sets. However, the type of noise used was non-gaussian and therefore not necessarily representative of some process noise situations.

Postlethwaite (1994) also examined the performance of a fuzzy-model-based controller compared to traditional control methods. Using a level control simulation, a more conventional model than that of Graham and Newell (1988,1989) was used from which the results indicated that the fuzzy approach can produce improved performance characteristics over the traditional approach, even in mainstream applications. However, Postlethwaite's fuzzy system did not have adaptive characteristics, which, as shown by Graham and Newell, could have a significant destructive effect on its long-term robustness and stability.

As shown by Feng et al (1997), fuzzy control systems can have guaranteed stability, even in non-linear situations. This is achieved by representing the fuzzy system as a family of local state space models, with a global controller considering each sub-system in conjunction with a compensating controller. However, it is identified that the stability conditions are quite restrictive, such that practical application would still require further study to maintain performance characteristics. Chao and Teng (1997) presented a PD-like self-tuning fuzzy controller. By tuning scaling factors both directly and indirectly (i.e. with back-propagation), examples are given for a time-invariant linear system and an unstable non-linear system. Results show zero steady state error, with good transient response. Adjustment or tuning of fuzzy sets has also been achieved using neural networks (So and Tse, 1997), thus eliminating the need for heuristic input in creating stable controllers for non-linear plant.

It can be seen that fuzzy logic has been applied in many contexts, sometimes in combination with classical control or other AI approaches, such as Neural Networks, sometimes on its own. However, it is clear that to maximise the available benefits, it is necessary to carefully consider the internal workings of fuzzy logic. Only by modifying standard theory to accommodate process characteristics can it be possible to approach any degree of optimality. In consideration of this, modifications have been proposed in the following section (4.4) to allow the implementation of fuzzy logic in a condition monitoring or supervisory role. Such a system for the filling plant is discussed in detail in chapters 7 and 8.

4.4 Fuzzy Logic Modifications

4.4.1 Matrix Method

The min-max inference method, as described in section 4.2.2, suffers from a disadvantage of not utilising all rule outcomes for determining output set truncation. As emphasis moves between rules, changes are only visible when the current maximum truncation value has been exceeded, even if the new rule outcome is almost equal in weighting. This leads to 'surprise' changes in the defuzzified output. In the additive inference approach, truncation values are added but a non-linear result is formed, as the relationship between a rule outcome and the area created by truncation of the output set is not equal for all cases. For example, if two rules modify the same output and both produce a result of 0.5, the consequent set would be of height 1.0. However, the second rule has only one-third the additive effect of a single rule, due to the triangular shape of the set defining the area. Where in fact the consequent set might be desired to have twice the area, which would double the effective weight of the set during the process of centroid defuzzification.

To overcome the described limitations of the standard methods, a 'matrix' approach has been developed (Jeffries et al, 1997). This allows the construction of many consequent sets, such that the outputs from all rules are given equal precedent in the calculation of the defuzzified result (Figure 4.6).

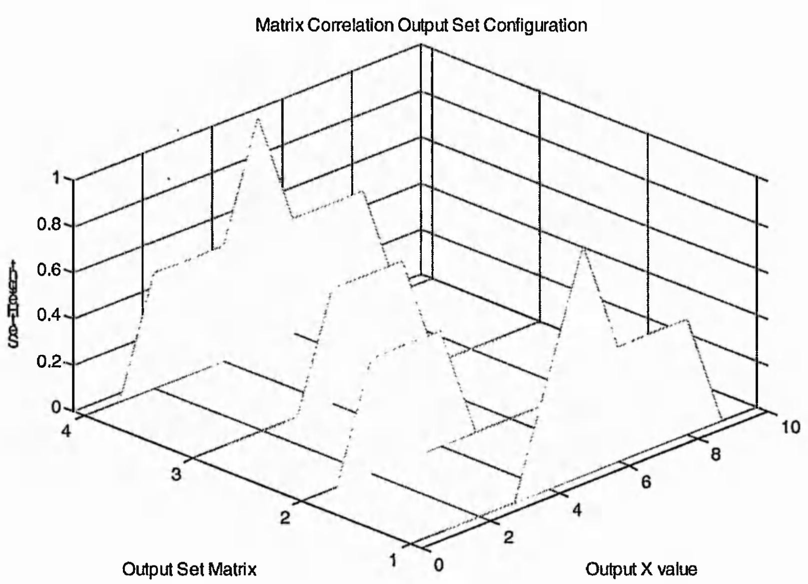


Figure 4.6 A Graphical Representation of the Matrix Inference Method

It has two main advantages: firstly, all rules are considered irrespective of the size of their contribution; secondly, rules producing equal results have equal influence on the final defuzzified value. The effect, therefore, is to subtly change the topology of the fuzzy decision surface linking input and output values (figure 4.7). The applicability of the approach has been demonstrated within the context of a condition monitoring system for packaging plant (Jeffries et al, 1997).

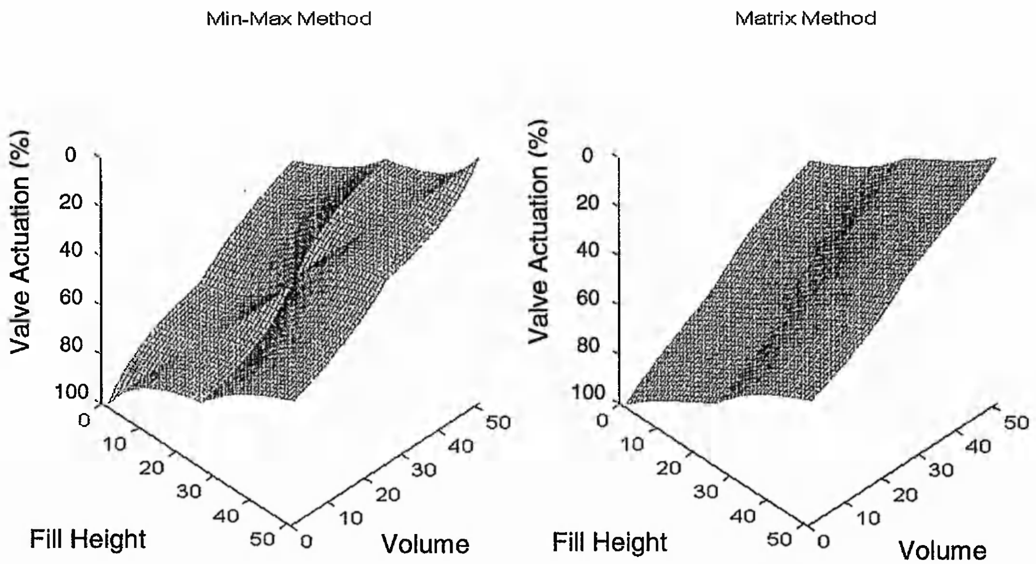


Figure 4.7 Fuzzy Decision Surface comparison shows Matrix method removes sharp changes in gradient, suggesting improved linearity and stability in an example for fill height control

4.4.2 Set Shape

It is common to describe both input and output variables in terms of sets (membership functions), often resembling each other in shape (e.g. triangular) and positioning (e.g. overlapping to form a continuous whole) - but they are not the same. The input sets relate variables to membership values by the shape of the defining lines. Output sets on the other hand, have various relationships depending on the de-fuzzification method chosen.

In the popular centroid method, the area of the set becomes the defining factor in influencing the output value; all set shapes (except rectangles) truncated by the 'correlation minimum' method create non-linear relationships. For example, if the output sets are triangular (often used in control systems) then reducing the height by half with a truncating membership of 0.5 will only reduce the significance of that set by 25% (the area removed). Using 'correlation product', linearity is restored, as area and height are directly proportional, which is the same as using rectangular sets with the minimum method. For many applications, especially those in control, a small degree of curvature can be desirable where the effect is to smooth the transitions between modes of output. Nevertheless, the degree of curvature is more a by-product of the methods chosen rather than a deliberate intention by design. If a linear surface is required then correlation product scaling will provide the necessary attributes.

Figure 4.8 shows an example fuzzy surface created by four different output set shape configurations and correlation minimum inference (Figure 4.9). To create the shown relationships, two output sets are used, one placed at the value 0, the other at 100. Two input sets are placed over the range 0 to 12. A simple rule base is assumed such that membership of the first input set relates to the truncation of the first output set; the second input set relates to the second output set. As the input value increases from 1 to 11, the truncation of the first output set declines from 1 to 0 and the second increases from 0 to 1.

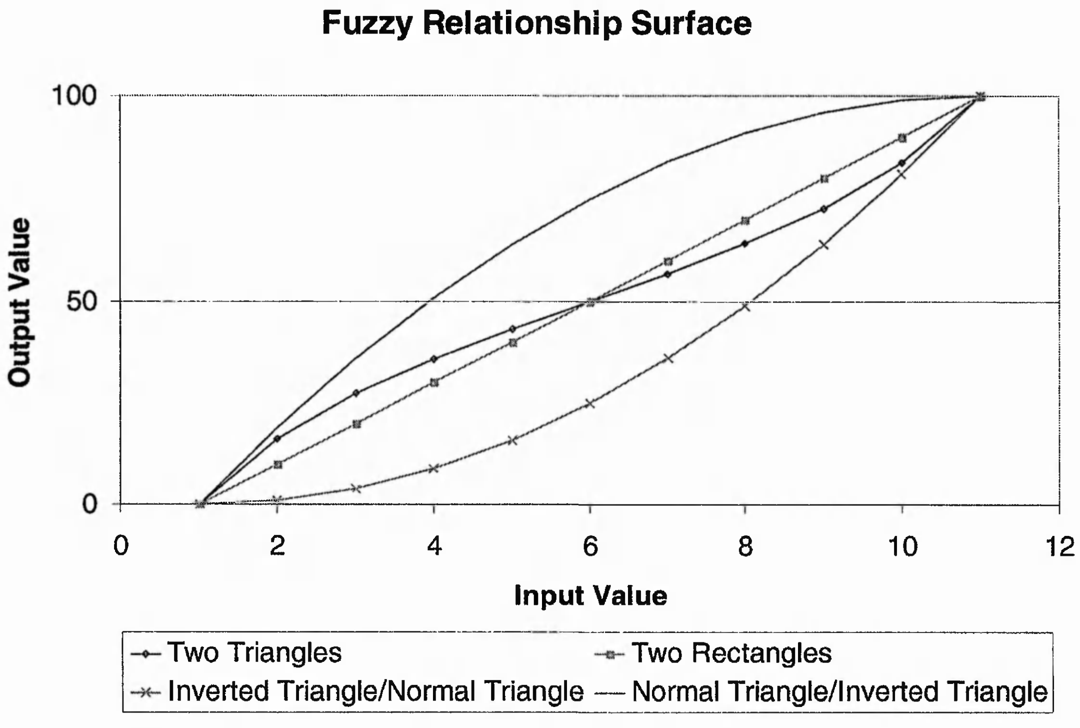


Figure 4.8 Fuzzy Input-Output Relationship Generated from Different Output Fuzzy Set Shapes

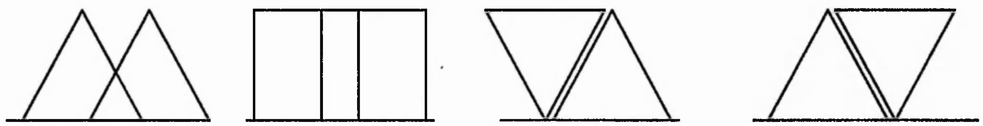


Figure 4.9 Fuzzy Output Set Configurations used to create different Input-Output Relationships

The first run uses two triangular output sets and a characteristic wave is created. The second employs two rectangular sets, which has produced a linear relationship. The third and fourth uses one inverted triangular set and one normal triangular set. As it can be seen in Figure 4.8, the relationship can be varied dramatically depending on the shape chosen.

It is evident from this example that the following factors are responsible for the shape of the fuzzy surface: the shape of the area of the output set and the method by which it is truncated or scaled. Therefore, why limit output set definition to the traditional shapes, except for the sake of easy visualisation? The shape of the fuzzy surface, hence the relationship between input-output, is controlled by the function of area and height of the shape, replacing this with equations chosen to form the desired linear or non-linear response, more direct control of the decision surface can be achieved.

A further method by which the fuzzy surface can be modified is by changing the centroid (position) of the output 'sets', which are usually pre-determined. If the output sets were allowed to 'move' during processing then an additional level of manipulation and control can be achieved. This approach applies to the previous truncation and shape modification procedures, but can be used most effectively in isolation, as the equations governing the movement of the centroids can represent the exact input-output relationship if the input values are used as the control variables (Figure 4.10).

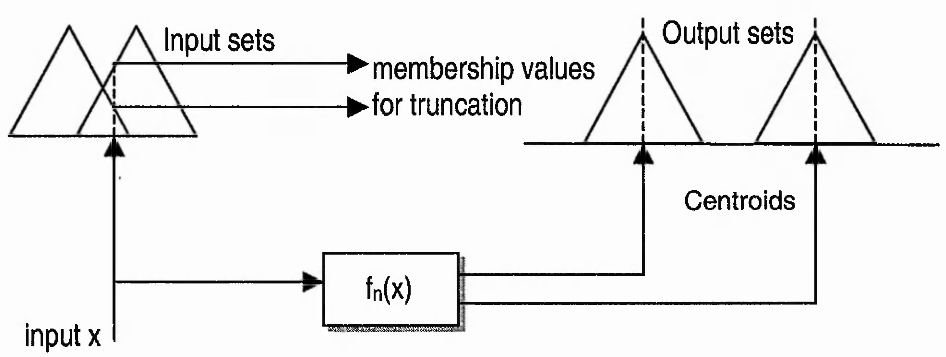


Figure 4.10 Input – Output Set Relationship where Output Set Position is a Function of the Input Value

In this approach the rule base is simplified, such that it is only used to link an input set to an output area. By understanding the desired type of input-output response, a function must be chosen to model the linear or non-linear shape. The function, which will be called the I/O equation, can be of any type from a simple linear gain, $y=mx$, or something more complicated such as Eq 4.4. This approach modifies the traditional heuristic nature of fuzzy logic, moving the main heuristic from the rule base to the choice of I/O equation. Unlike the normal approach of Fuzzy Logic where the input values are converted into membership values and used in that form throughout, the present approach retains both the membership value and the original input value. The membership value is not used to directly calculate the output, but to establish the relative weighting of each I/O equation via a modification of the centroid equation (Equation 4.2).

$$O = \frac{M \cdot Y}{\sum \mu(n)} \tag{4.2}$$

where,

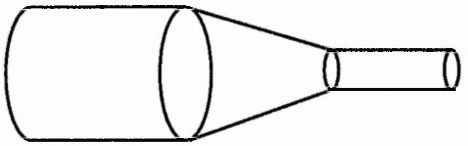
$M = [\mu(1) \quad \mu(2) \quad \ldots \quad \mu(n)]$ - a matrix of weights (memberships)

$Y = \begin{bmatrix} f_1(x) \\ f_2(x) \\ \ldots \\ f_n(x) \end{bmatrix}$ - the output of a function given the input x (IO equations)

O – the output value

In this new variant of the centroid equation, the individual set centroids are replaced by functions, whose result changes depending on the input value x. Instead of relying on the relationship between height and area of the output sets for weighting, the input membership values, μ are used directly. This is an important difference, as any further transformations would dilute the significance of I/O equations. In any case, non-linear properties can be modelled directly in the proposed I/O equations; therefore negating the need for anything but a linear output set representation.

The advantage of this method is the ability to model very complex non-linear relationships using a combination of simpler non-linear and linear equations. A simple example of this can be shown using the way in which volume changes with height in a bottle-like vessel.



Note: Vessel shown on its side

Figure 4.11 Simple Bottle Shape

Figure 4.11 shows a vessel with three different volumetric elements: a large diameter cylinder (linear volume/height), a conical section of reducing diameter (non-linear volume/height) and a small diameter cylinder (linear volume/height).

To model the fill height changes for a given volume, it is necessary to use two equations. The first (Eq. 4.3) describes the volume(V)/height(x) change for the cylindrical sections and is linear.

$$x = \frac{V}{\pi r^2} \qquad \textit{where } x \leq \textit{height of the section} \tag{4.3}$$

The second (Eq. 4.4) describes the volume/height change over the conical section and is non-linear.

$$x = \left[\frac{3Vh^2}{\pi r^2} - h^3\right]^{\frac{1}{3}} + h \tag{4.4}$$

where r is the radius of the largest section and h is the height of the cone.

To construct a single function to describe the change in height with volume for the whole vessel is difficult by conventional mathematical techniques, as the discontinuities between the volumetric elements, which are both linear and non-linear, cannot be described by a single equation. However, by using the modifications to fuzzy logic previously described, a model can be formed which is both continuous and robust. Each of the three sections is assigned an input fuzzy set in the volume domain, which overlap on the shape transitions. Linked to these are the characteristic equations for the individual shapes. This allows output from two equations to simultaneously contribute to the final output during transitions, providing the ability to smoothly combine the linear and non-linear characteristics (Figure 4.12).

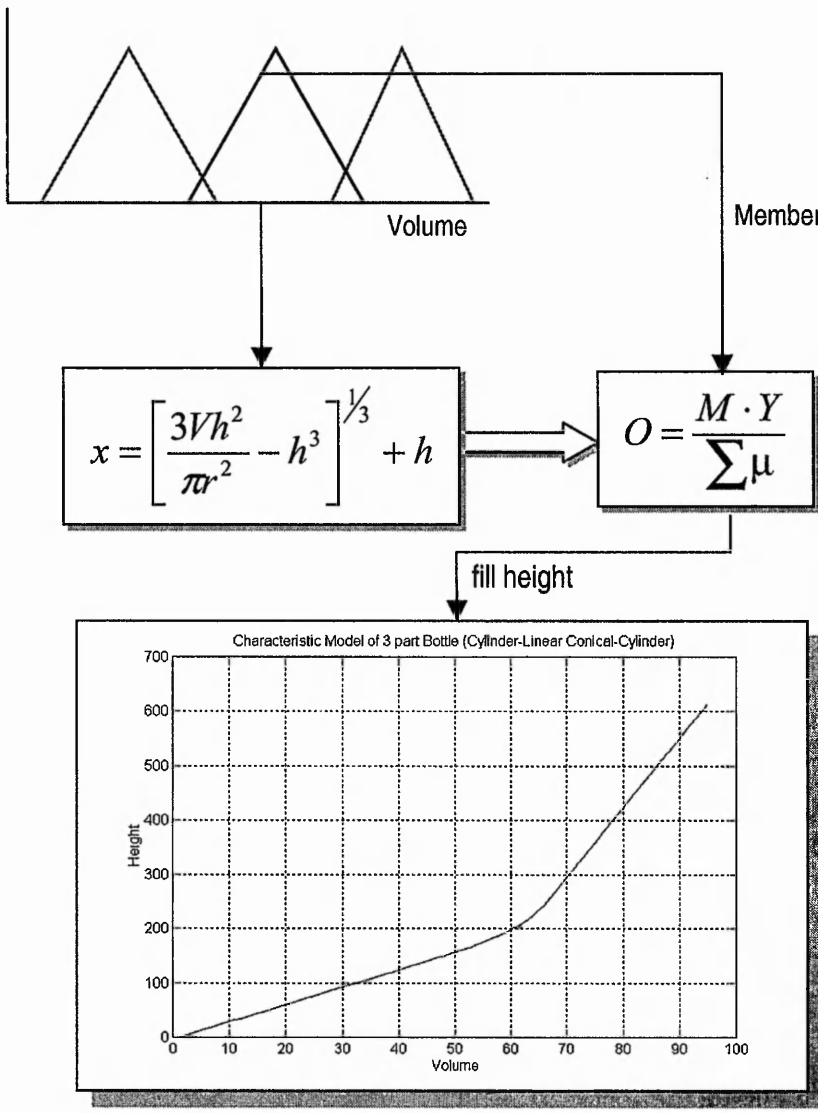


Figure 4.12 Continuous Model from Derived Fuzzy Logic Input Set Definitions

In this example all the equations were known, which could allow other computational algorithmic approaches to be used. However, in situations where only approximations to the non-linear characteristics can be found, robustness will not be compromised by modelling inaccuracies, although accuracy inevitably will.

4.5 Summary

Fuzzy logic has many benefits for the design of a bottle filling control strategy. It offers good levels of robustness and stability without necessarily having the full knowledge of the system dynamics by using heuristic conjecture. It has been demonstrated many times how it can benefit conventional and more commonly less-conventional control situations, where other methods have difficulties to provide the levels of robustness required within the industrial setting.

However, in order to provide a more optimal solution further effort is required to understand the mathematical transformations that represent the fuzzy process. To this end several avenues of modifications have been investigated to better represent the needs of the process under investigation. These have focused on the input-output relationship, allowing an approach to be tailored to the problem in hand whilst maintaining the natural benefits of the fuzzy method. These new modifications will be demonstrated later and be shown to be most useful in the control of the bottling plant, especially so regarding the multiple valve carousel situations.

Chapter 5

Modelling of a Single Filling Vessel

5.1 Introduction

In consideration for control of a bottle filling plant it is important to investigate two aspects. Firstly, the requirements of the individual valves and bottles as seen in the presence of environmental disturbances and secondly, the control and management of the plant such as to include the carousel of many valves and implications of larger and longer term disturbances. This chapter describes the methods employed to tackle the first aspect, specifically, to create a suitable simulation of the filling process. In addition, a simple test rig is outlined in its use as means of obtaining information on the characteristics of level measurement. The experimental and simulation results can be found in the following chapter. The management of a multiple valve carousel is covered in Chapters 7 and 8.

5.2 Valve Model and Characteristics

The operation of the valve used in bottling is intentionally simple, giving good repeatability and robustness. Actuation is achieved by a combination of pressure and mechanical means. As the bottle engages with the mechanism (Figure 5.1a), pressure is balanced between header tank and bottle, when this is achieved the valve opens and filling begins. During filling the valve is held open against the force of a return spring. On achieving the desired height, the valve is released to close by a pressure release mechanism triggered as the vent in the pressure balance tube is covered by liquid. For systems where the fill level is close to the top of the vessel, e.g. cans, a floatation ball is often employed (Figure 5.1b).

As can be expected, the closure of the valve will be swifter than that of opening. Unlike some control valves, proportional control is not normally available to bottling operations, the valve being either fully open or fully closed;

thus, the flow rate is constant. However, the ring orifice through which the liquid passes is subject to residue build-up, which can reduce the maximum achievable flow rate. Therefore, the following characteristics have to be considered for modelling the valve operation. Firstly, the valve has two positions open and closed, where closure can be expected to be marginally faster than opening. Secondly, the maximum flow rate can be reduced by the build-up of product residue.

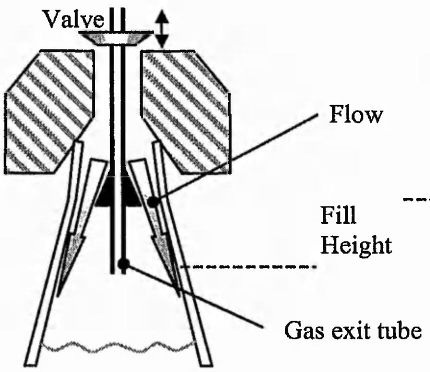


Figure 5.1a. Pressure Balance Valve

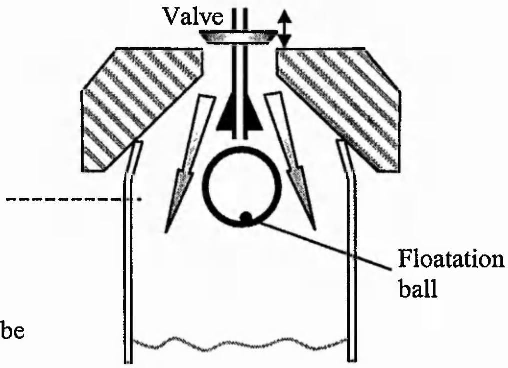


Figure 5.1b. Pressure Balance Valve with Floatation Ball

5.3 Filling Characteristics

For the pressure balance valve, the determination of filling level relies solely on the covering of the end of the pressure balance tube by the rising liquid. However, due to the high flow rates and vibration from the machine operation, the surface can be expected to be reasonably agitated, thus reducing the fill accuracy.

In the valve variant used for cans, where a floatation ball is used to block the pressure exit, the ball is subject to deterioration over time, reducing its effectiveness either in terms of buoyancy or ability to properly seal the pressure balance hole, this is in addition to mechanical vibration. Nevertheless, these do not affect flow rate, which can be assumed to be constant.

Within the single bottle filling operation, it can also be assumed that no significant pressure variation will occur within the header tank, as this will only feed a single valve. Therefore the single and most significant factor for determining fill height is the shape of the bottle.

Container and bottle shape is extremely varied, both across the industry and within a single manufacturer. Nevertheless, the overall shape can be categorised into a number of simple primitives (Figure 5.2) – ignoring, for the moment, the more exotic shapes achievable from plastic.

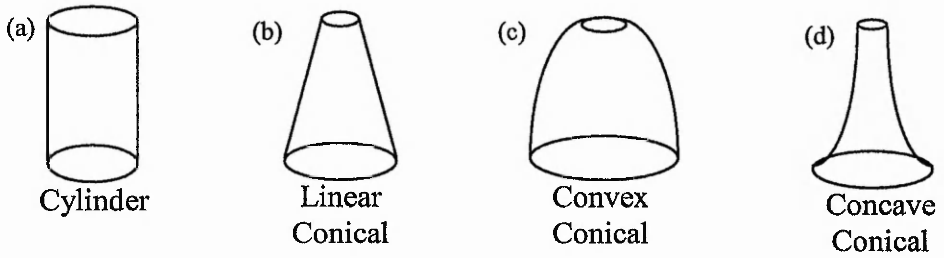


Figure 5.2.Four basic primitives of bottle shape

The first and simplest is the cylinder, creating a purely linear association between volume and fill level. The next primitive - the cone - has three variants, linear conical, convex conical and concave conical. These can be used to describe the neck region on a large number of bottles, however the relationship between volume and fill height is non-linear.

By developing an approximate mathematical description of the bottle curvature, a volumetric representation can be gained by performing a volume of revolution integration about the vertical axis; this creates a Characteristic Equation (CE) for each shape. This equation relates volume to fill height and is independent of bottle size. For example, a common shape used for bottle necks is the linear conical, which can be described by the following characteristic equation,

$$\frac{x^3}{3h^2} - \frac{x^2}{h} + x - \frac{V}{\pi r^2} = 0 \tag{5.1}$$

where; x = fill height, V = filled volume, r = radius of cone and h = height of cone

This can be rearranged to show fill height in terms of the filled volume. (Eq. 5.2)

$$x = \left[\frac{3Vh^2}{\pi r^2} - h^3 \right]^{1/3} + h \tag{5.2}$$

As can be seen the result is in a non-linear form even for a relatively simple shape (Figure 5.2(b)) and this represents only one section. Variations in shape are clearly the most significant factor in determining where fill height is expected from a given volume.

5.4 Controller

The filling process is an example of batch control, where a specific volume of liquid is required to be pumped sequentially into receiving bottles or containers. The filling of each vessel can be characterised by the simple block diagram shown in Figure 5.3.

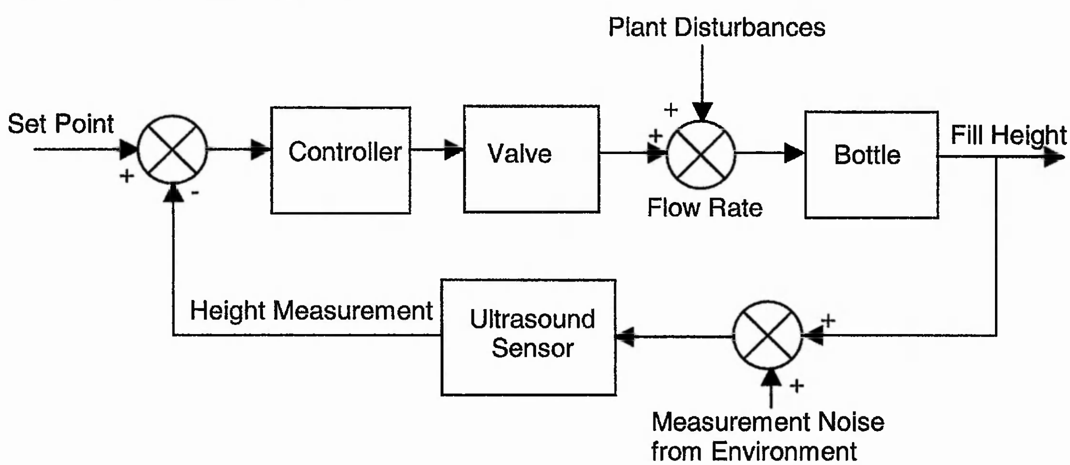


Figure 5.3 Simple Block Diagram of the Filling Control Process including Disturbances

As already discussed, the valve design means that the actual control required for the valve is minimal, requiring only on or off states. The greatest effort, therefore, will be in the monitoring of the fill level to determine when to switch the valve off. To improve this approach a predictive element has been added. Within the Kalman filter a prediction is made for the expected measurement in the next iteration. By passing this value to a controller a simple test can be performed, which can be interpreted as, if the difference between the current measurement and the set-point is smaller than the expected error in the next iteration then close the valve now (Figure 5.4).

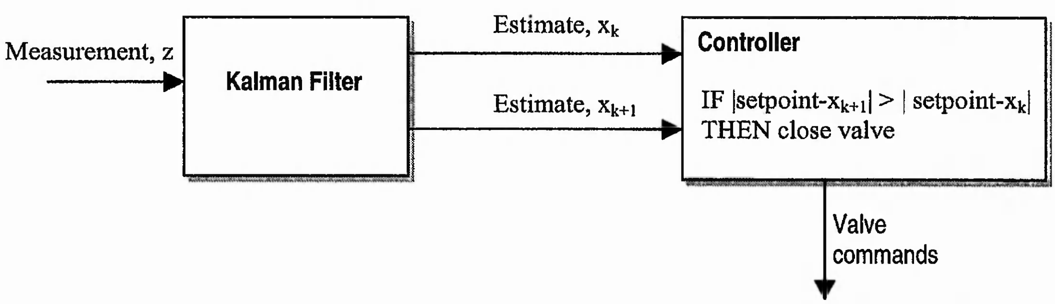


Figure 5.4 Schematic of a Simple Predictive Valve Controller using Kalman Filter Estimates

This method aims to reduce the natural overfill characteristic present if the decision is made to close the valve only when the setpoint is actually reached (or exceeded). See section 5.5.1.8 for implementation details.

5.5 Simulation

The ability to directly measure the performance of the plant is unachievable, as actual plant are in constant use, and the building of an experimental rig will only be able to model some aspects of the problem. Therefore the most practical solution is to simulate the plant, using the available data and known characteristics, in order to test algorithmic strategies in the control of the fill level.

It has been chosen to use the C programming language to implement the simulation. This is for two main reasons. Firstly, C allows significant programming flexibility, presenting the opportunity to eventually implement the created algorithms on different systems (e.g. Digital Signal Processors). Secondly, components covering the Kalman filters and the Fuzzy Logic have already been created by the author in C, thus minimising software development time.

However, one major drawback of C is its lack of a standard graphical output for the viewing of data. To overcome this Matlab® is employed as a front-end. This allows not only excellent graphical data representation but also provides the ability to execute pre-compiled programs from within its command line interface. This means once the simulation is written in C and then compiled into executable form, data analysis and running of the simulation can be performed solely within the Matlab environment. The compiler chosen for the simulation is Borland C/C++ 4.52.

5.5.1 Development of Software

A modular approach has been taken with the program design, such that by conducting experiments in parallel with the development of the simulation it will be possible to continually improve components such as the noise model without significant additional programming. This approach will also aid the conversion of the simulation from the single vessel case to the multiple valve carousel style.

The program structure is shown in Figure 5.5, detailing the relationship between various elements of the model. The functions of individual blocks are described in the following sections and the source code for the complete model can be found in Appendix A.

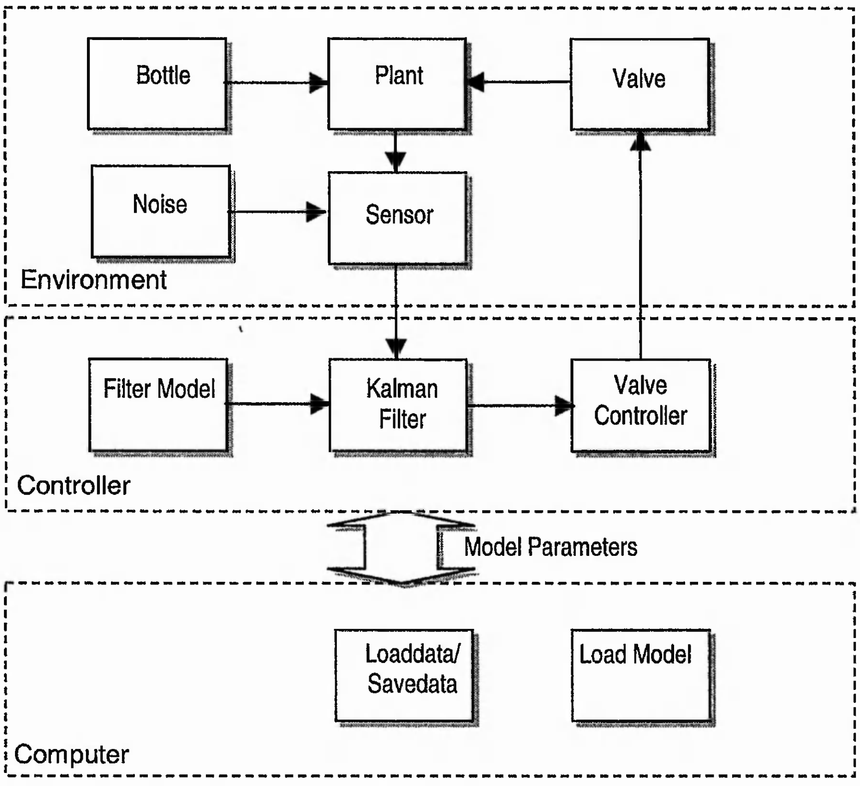


Figure 5.5 Plant Simulation Program Structure showing Software Modules

5.5.1.1 Bottle

The function *Bottle* contains a description of the bottle shape, relating height and volume. The current evolution of the software uses a data look-up table created from the characteristic equations processed through a program written with the Matlab script language. Values that are not present in the table are linearly interpolated.

5.5.1.2 Valve

This function aims to model the opening and closing characteristics of the valve to produce the flow rate. By setting a speed, measured in percentage open per second, the opening and closing of the valve can be modified separately. Currently, this is a linear model.

5.5.1.3 Plant

Converts the flow rate determined by *Valve* to a current total volume, and combining it with the shape relationship described by *Bottle*. This results in a value representing the actual height of the liquid, and the speed with which it is approaching the top of the bottle.

5.5.1.4 Noise

This function is capable of modelling range of noise distributions, such as Normal and Gamma. It is added to the actual height of the liquid to simulate the measurable value. The relevance of the different types to the process will be discussed later.

In addition, the speed of the liquid is converted into a representative Doppler frequency shift of the ultrasound signal. The significance of this frequency shift will be discussed further in Section 7.2.

5.5.1.5 Sensor

Uses *Plant* and *Noise* to obtain measured value for height. Also, allows resetting of these two functions at the end of the filling cycle.

5.5.1.6 Kalman Filter

This module is based on the standard 5-equation Kalman filter algorithm (see chapter 3), with additional processing elements to convert fill height to volumetric representation, thus allowing a linear filter model to be used (Figure 5.6).

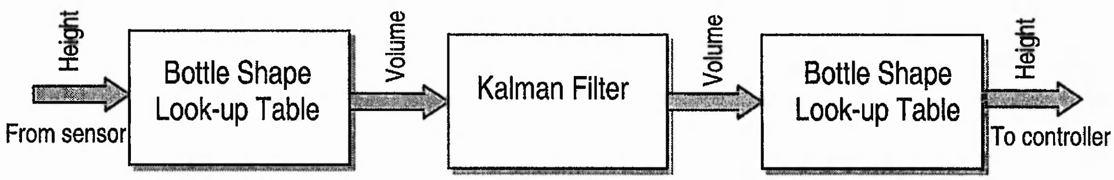


Figure 5.6 Pre- & Post-processing for the Kalman Filter to allow for a linear internal model

5.5.1.7 Filter Model

This function provides input to the Kalman filter algorithm allowing prediction of the next process state. The model is based on the filled volume, whose rate of change will be constant for a fixed flow rate.

5.5.1.8 Valve Control

The requirements for valve control are simple – either on or off. As shown in Section 5.4, the Kalman filter can provide values for a predictive controller, this is easy to implement within the program by passing internal values from the filter function to the control function.

5.5.1.9 Loaddata/Savedata

Major parameters are input and output to and from the simulation via text data files. Input variables include:

model run time, iteration time step, time for a bottle fill, filling set point, maximum flow rate, radius of the bottle, ultrasonic frequency, measurement noise, percentage data loss (a condition where complete loss of measurement data occurs randomly with a percentage probability specified in the input file), valve closure, valve opening, Kalman measurement variance, Kalman model variance, initial Kalman estimate, initial Kalman variance and assumed flow rate.

Any of the simulation parameters can be saved for later processing or display, commonly these are the Kalman performance parameters and actual and estimates of fill height and volume.

The output can be viewed either as raw text, Matlab graphs or Excel spreadsheets.

5.5.1.10 Loadmodel

The lookup table for bottle shape is loaded using this function. The data file used is created using a Matlab script file, which uses the characteristic equations of shape to create any number of different bottle shapes.

The units for distance in most of the model is centimetres, rather than metres or millimetres, for the convenience of scale.

5.6 The Test Rig

This project forms part of larger whole to develop a strategy for improved container filling operations and thus any test rig must be designed from the outset with flexibility very much to the fore. Apart from the obvious elements such as variable feed pressure and the opportunity to attach a variety of instrumentation, an important facility during further development could be the ability to vary the filling configuration – top filling, bottom filling, one bottle, two bottles etc. To this end, a rig has been developed that not only meets current needs, but should also provide useful function beyond the scope of the initial research.

5.6.1 Requirements

A critical aspect of the rig must be its flexibility. A variety of different filling situations will be required with a range of sensors including flow sensors, pressure sensors and ultrasound sensors. To maximise performance and usefulness, integration with a Personal Computer will be essential.

The main requirements are:

- Adjustable head of water from 1 to 2 metres, but constant during filling.
- Adjustable pipe work configuration for sensor mounting
- Linking to a PC

5.6.2 Design

A simple means to achieve a static head pressure is to use a tank placed at the desired height. To maintain that head in the presence of varying demand it is necessary to introduce a pump and overflow system (Figure 5.7).

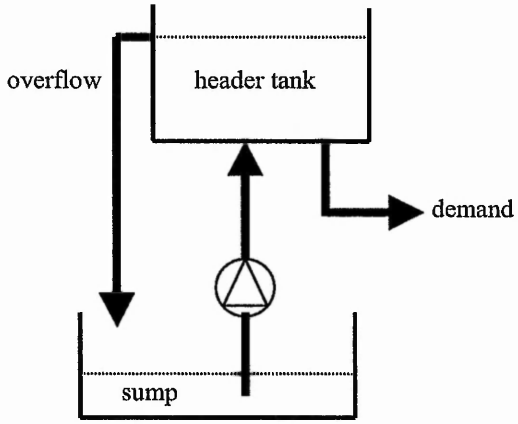


Figure 5.7 Test Rig Schematic Design

Thus, fluid level in the tank remains at the overflow level whilst the pump is running and the demand is lower than the maximum achievable performance of the pump. Although this arrangement can produce a near constant head, no means of adjusting that head is present.

Two methods present themselves as a means of overcoming this problem. The first is to alter the relative height of the tank; the second is to change the height of the overflow within the tank. A limitation of the second approach is that the depth of the header tank is the maximum achievable variation. However, to move the entire header assembly is unnecessarily cumbersome in mechanical design. Therefore, taking the second approach to its natural conclusion the header tank must be deep enough to accommodate the required 1m variation.

By using a two-metre long, large diameter tube (200mm) placed on its end and locating a telescopic overflow in the centre it is possible to adjust the available head to any value between 0.75 to 2m (Figure 5.7).

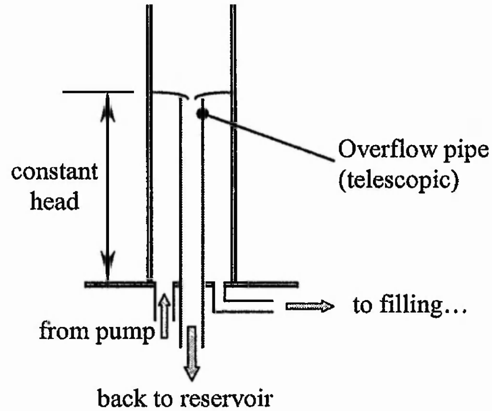


Figure 5.8 Overflow system to achieve constant head, as used on the test rig

A combination of polyethylene pipework and push-fit ('speedfit') connectors were chosen to allow for the required flexibility. Two diameters were used - 22mm (I/D ~17mm) for reservoir to pump to header tank, 15mm (I/D 11mm) for filling. From this knowledge and assuming Bernoulli's equation for flow from an open reservoir to atmosphere we can calculate the maximum achievable flow rate, i.e. no losses (Massey, 1989).

$$v = \sqrt{2gh} \tag{5.3}$$

v = *6.26 m/s* for a 2m head

Flow rate for 11mm diameter pipe

$$Q = \pi r^2 v \tag{5.4}$$

Q = *35.7 l/min*

As exit fittings are smaller than 11mm and losses will occur from bends, valves and friction, then the actual flow rate will be less.

A 2-litre vessel has been used to represent a bottle. It has straight sides and an exit in the base to allow water to be emptied without removing the vessel from the rig. This also allows the vessel to be filled either from the top or the bottom.

The exit nozzles, Figure 5.9, have been designed with two key features. The first is to place the exits at the sides so as to discharge the liquids horizontally into the sides of the vessel. This minimises turbulence that would be caused if incoming water were allowed to fall directly onto the liquid surface. Two variants have been designed; one with four 5.5mm holes (total area ~95mm^2), the other with twenty-four 2mm holes in six columns of four (total area ~75mm^2). The first variant allows ultrasound sensors to be placed above the surface without interference from input jets; the second provides less space but lower surface turbulence.

The second feature present in both variants is the ability to change the exit area by means of a screw adjustment placed in the base. This allows for fine adjustment of flow rate without using the relatively insensitive change in the overflow height.

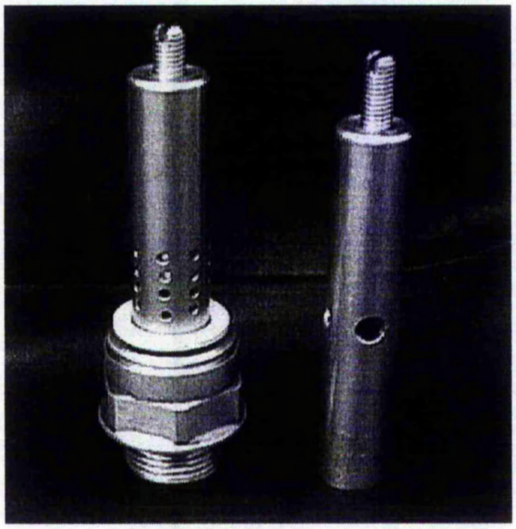


Figure 5.9 Brass Filling Nozzles with Adjustable Exit Opening

A pump was selected to maintain good flow rates of around 30l/min (equivalent to filling a 1-litre bottle in 2 seconds) whilst easily achieving 3m head, which is the approximate distance from the base of the reservoir to the top of the header tank. A pump (model: ES033/1) from Stuart Pumps, purchased through the Farnell Industrial Catalogue, met the required specification.

To utilise a valve as used in the actual filling plant would require the development of a system almost identical in design. However, the valve characteristics are relatively simple – either on or off – therefore a simple valve can be substituted. In any case a more critical factor at this stage is the measurement of dynamic fill level rather than the control of the valve, which, in comparison, is straightforward. To this end, valves were selected from the RS component catalogue to operate under zero-head opening conditions (i.e. do not require pressure to open), with 240v solenoids and ½ inch fittings.

With both pump and valves operating from 240v mains supply it was necessary to provide adequate safety procedure to prevent risk to operators of the rig. Although the actual risk is small, as all components are specifically designed and certified for use in water systems, a degree of inherent risk is always present when combining water and electrical systems. A safety assessment was undertaken and approved by the department's safety advisor; it has been summarised below.

- Operation of the valves and pump is by 5v relay.
- Unused wall sockets within an extended arm's reach of the rig are covered
- Normal operating distance is over 1m.

The relay box is a sealed unit with a splash-proof power switch and touch-proof fuse holder, which prevent contact with live connections even when changing fuses.

Switching of the pump and valves can be achieved either manually or automatically from a PC. The computer is fitted with an Analogue/Digital (A/D) interface card (PCI-ADC from Blue Chip Technology) that can output a digital signal in response to software commands. To achieve the current required by the 5v relays, an opto-isolator is employed to switch voltage from an additional power supply. A limitation of many standard opto-isolators is their relativity low current capacity (< 50mA), and initially this proved a problem as the relays have 36Ω impedance and will draw up to 140mA current; thus a Darlington-type opto-isolator was chosen with a maximum current of 160mA. A Darlington-type has an additional transistor cascaded in the output stage to improve the current capacity. If an even higher current is required then external transistors have to be used.

The PC interface card has a range of features suitable not only for outputting control signals but also taking measurements from sensors. Sixteen single ended analogue inputs are available for converting to a digital signal with 12-bit resolution. Adjustable gain on the input allows for signals from a range of ±5mV to ±5V. It is fitted into a PCI slot on a Pentium computer running Windows NT4. Software libraries are provided (at additional cost), which has allowed a simple control and measurement C program to be written. Additional features such as 4MHz counter/timers and analogue outputs are available but unused in the current configuration.

In order to provide a basic control method during development of the ultrasonic sensor technology, a pressure transducer sensitive to head up to 250mm has been placed within the filling vessel (395-241, RS). This allows the rig to fill and empty without user intervention and can be used as a benchmark for ultrasonic measurement. The signal from the transducer is read by one of the A/D input channels on the interface card. A piece of software has been written to perform a basic fill and empty cycle, whilst logging data from A/D converters.

5.7 Summary

A simulation of the filling process has been described showing the performance characteristics of a Kalman filter used to reduce the influence of noise attributed to the physical influences, such as surface agitation or environmental conditions. It has been designed to allow for a wide range of filling and measurement conditions to be tested and has scope for continual improvement as more information on the filling process is found. A test rig has also been designed to assist the simulation by providing valuable information on ultrasonic measurement characteristics.

Chapter 6

Measurement and Simulation of Liquid Level

6.1 Introduction

Two stages are important to the development of the measurement technique used to ascertain liquid level within a filling operation. Firstly, information on the characteristics of the sensor method from experimental data, and secondly, results from simulation allowing monitoring algorithms to be understood under a range of conditions. This chapter covers these two areas.

The test rig described in Chapter 5 is used to obtain liquid level measurements from ultrasound signals. In this chapter, the methods required to obtain level from the raw ultrasonic data are shown and comparisons made to a proprietary pressure transducer. The noise in the measurement is categorised and a proposal for a noise model made. From these results the developed single vessel simulation is used to investigate the performance of the proposed Kalman Filter under a number of different operational scenarios.

6.2 Test Rig Experiments

A number of experiments need to be conducted in order to provide the benchmarks on which the simulation can be based. As the ultrasonic sensor technology is under development concurrent with the control strategy it is necessary and useful to use the pressure sensor as a means of understanding the test rig performance, thus level measurements are initially obtained by this method. Therefore, when ultrasonic transducers are employed for measurement in the following set of experiments, efforts can be concentrated on understanding the characteristics of this particular measurement method, as opposed to the underlying test rig characteristic.

6.2.1 Level Measurement by Pressure Transducer

6.2.1.1 Apparatus

The pressure transducer is a differential type, where venting one port to atmosphere will give gauge measurements and connection to the filling vessel is achieved by a flexible tube (internal diameter 2.5mm). The transducer was placed on the outside of the vessel with the tube, which was pre-filled with water, looping over the top of the vessel. The transducer is accurate to ±0.25% of full-scale-output, thus theoretical detectable level change by pressure is ±0.625mm and linear to a good approximation. A 5v supply was required and was acquired from the 5v/12v power supply used to power the main relays (Figure 6.1).

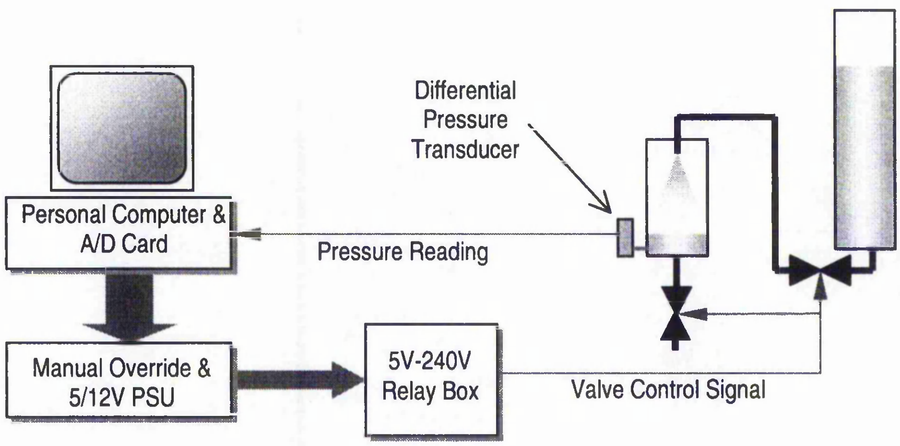


Figure 6.1 Computer Control and Monitoring of Test Rig

6.2.1.2 Method

A head of 1m above the nozzle exit was established and flow rate was adjusted at the nozzle to achieve a fill of the 2 litre vessel in approximately 15 seconds, which would provide a reasonable benchmark on which to prototype the pressure and ultrasonic measurements.

The auto control software was run with logging initiated. Thresholds were set for the minimum and maximum levels and the program cycled between the two heights by operating either the valve from the header tank (filling) or the valve to the reservoir (emptying).

The software has the ability to average several incoming data points to reduce noise. Runs were completed with varying degrees of averaging – from none to 1000 points.

The aim of the experiment was to qualify the use of a pressure transducer as primary control sensor for the rig and to identify the characteristics of the readings taken.

6.2.1.3 Results

The experimental results show a measurement with good linearity (Figure 6.2). This confirms the expectation from constant head test rig design and the quoted performance of the proprietary pressure transducer that this method provides a reliable benchmark for further investigations.

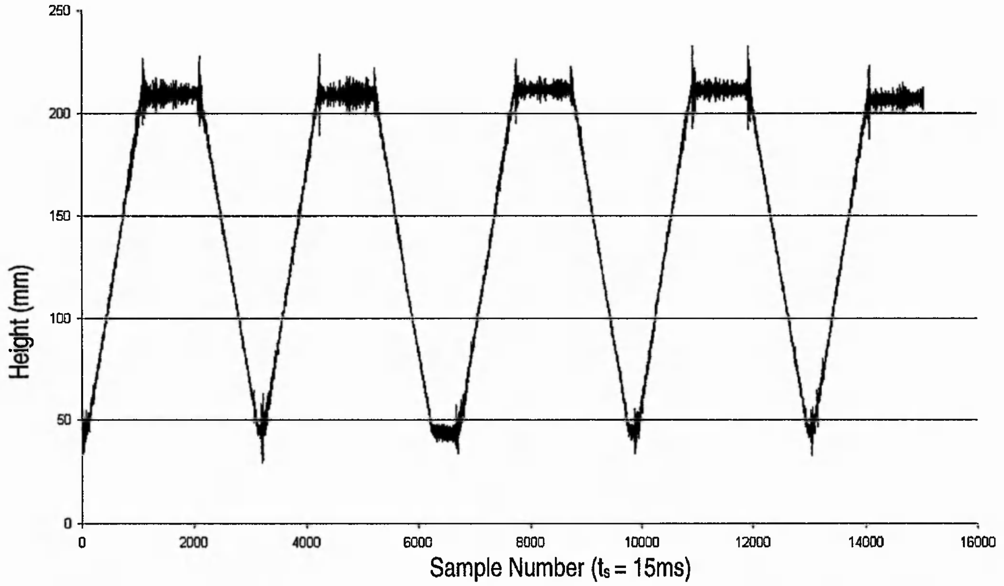


Figure 6.2 Measurements of liquid level from a pressure sensor during a 15second fill of a 2-litre vessel

However, it was found that electrical noise was interfering with the signal obtained from the transducer. After a short investigation two sources were identified. Firstly, low level interference from mains electricity, filtering through the power supply and indirectly through unshielded connectors. The second source was found to be from the operation of the solenoid valves. This was characterised by a spike in the received signal with low damping causing a degree of ringing afterwards. By running the transducer and the power relays from the same power supply the set-up is vulnerable to this kind of problem.

Although the effect of the noise was to reduce the accuracy of level measurements, it was decided to allow the noise at the current time, as the effort required to maximise sensor performance would not provide significant difference in the information gathered at this stage in the investigation. Moreover, a sensor with real noise could offer the possibility to test the control strategy developed in simulation without relying exclusively on ultrasonic transducer, thus broadening the scenarios tested.

Despite the reduction in performance, by analysing the gradient of the obtained data the overall flow rate could be determined. It was found to be 5.4 l/min, which is 7.2 l/min lower than the calculated value of 12.6 l/min based on available head and nozzle exit diameter; indicating losses in the system from fittings, valves and the nozzle.

The experiment also showed that control of the rig could be achieved through the use of a pressure sensor and a PC even in the presence of noise. However, it was also noted that, on occasion, the interface card would lose step with the sampling and return an error terminating the software, thus causing loss of control; therefore demonstrating that without supervision the risk of overfilling is present within the set-up. The problem was attributed to the lack of real-time performance criteria within the PC operating system and demonstrates the need for a dedicated system such as a Digital Controller or Digital Signal Processor (DSP). On this basis, it was decided that reliable real-time control of the rig could not be achieved through the present set-up. Moreover, the development time required to create the required reliability would severely interfere with the schedule of the research. As the system would effectively have to be of commercial standard and the electronic engineering tools not being readily available to the project, it was decided to concentrate on the classification and monitoring of the process rather than achieving an implemented control system within the research time-scale. However, consideration for the practical implementation of algorithms developed in the simulation will be given in the closing chapters.

6.2.2 Measuring Level with Ultrasonic Transducers

6.2.2.1 Apparatus

To use and process ultrasonic signals for level measurement requires several pieces of additional equipment; the sensors, a signal generator, a high quality pre-amp (to recover weak echoes) and a dedicated fast A/D card fitted to a standard PC.

The sensors selected were piezo-electric types with a narrow resonance peak around 160kHz. Two transducers were used, one to transmit (Tx) and one to receive (Rx). The signal generator was set-up to provide a pulse of 24 complete cycles at 30V p-p, with the trigger for the pulse also redirected to the A/D card to initiate sampling. The pre-amp, positioned between the Rx sensor and the A/D card, has selectable gain of either 40dB (100x) or 60dB (1000x) depending on the strength of the returning signal, which, with the available transducers, is typically around 1mV p-p. The A/D card has a selectable sampling rate from a maximum of 6.25MHz to minimum of 195kHz, halving each time. Determination of the time-of-flight can be achieved with the card by setting a hardware trigger from the control software, activated be a returning echo having amplitude greater than the specified value. A definable trigger delay can also be implemented such that the send pulse is not logged, as this is often several orders of magnitude larger than the receiving echo and will confuse the threshold trigger, and is in any case of little value. Both the time-of-flight and the full echo trace are stored for later analysis.

Control of the rig is achieved, as before, via the pressure transducer and control PC.

6.2.2.2 Method

The transducers were placed 260mm above the base of the vessel, leaving around 65mm gap between sensors and liquid when the vessel was full. They were placed at the side of the filling nozzle, in-between the spray jets of the 4-hole variant. The distance between the sensors was measured to be 45.5mm. This offset between transmitter and receiver requires a small correction to the time-of-flight calculation to achieve the correct associated liquid height.

As can be seen from Figure 6.3, the signal, after reflecting off of the surface, appears to originate from the virtual Tx with a distance, *T*, (calculated from time-of-flight multiplied by the speed of sound under the current transmission conditions; typically around 340m/s for air). The actual height of the transducer above the reflecting surface is *h* rather than *T*/2, which only be used when a single transducer is used – *x* is effectively zero.

The equation to find *h* is simply derived from the Pythagorean Rule:

$$T^2 = x^2 + (2h)^2 \tag{6.1}$$

rearranging to,

$$h = \sqrt{\frac{T^2 - x^2}{4}} \tag{6.2}$$

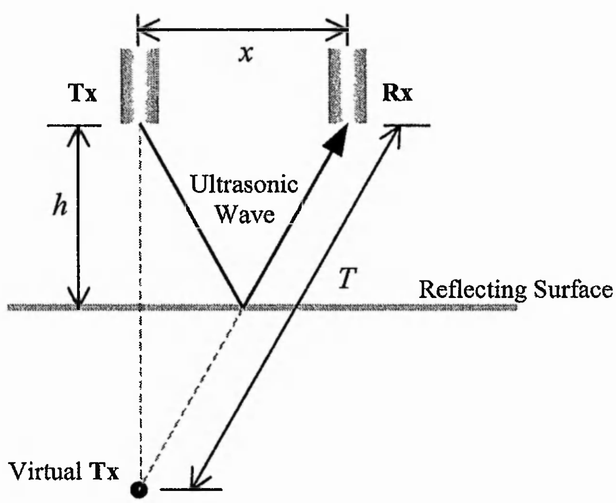


Figure 6.3 Ultrasonic Reflection Geometry

The sampling rate of the A/D card was set at the maximum hardware speed of 6.25MHz to give the clearest representation of the echo. In reference to the Nyquist criterion, a sampling rate of at least double the frequency of the measured signal must be used, although in practice a rate of ten-times often provides more realistic representation. In this case with a signal frequency of around 160kHz, the sampling rate will be almost forty-times the measured frequency. However, the time-of-flight can only be calculated for each complete echo received, and with limitations of PC architecture (i.e. lack of a true real-time operating system), the effective sampling time for the level measurement is somewhat less, at approximately 400Hz or under (16K buffer sampling at 6.25MHz). However, the Nyquist criterion has no bearing on this rate, as the signal is solely within the time-domain. Also, the signal generator governs the time between pulses, which is set such that echoes from previous pulse have reached the receiver or dissipated to the extent that they no longer interfere with threshold triggering. The delay between pulses can be determined by calculating the maximum expected time-of-flight, and setting the pulse delay slightly longer. In this instance, with a total travel distance of just over 500mm, the expected time-of-flight can be expected to be around 6ms, a pulse rate of 160Hz or less will prevent echo confusion.

The aim of the experiment was to quantify the noise present within the time domain, thus validating or improving the noise characteristics used within the plant simulation.

6.2.2.3 Results

The data files containing the raw signal data showed a characteristic envelope shape to the echo even with different peak-peak amplitudes (Figures 6.4 and 6.5). The maximum amplitude can be found at 24 cycles from the beginning of the echo, which corresponds with exactly with the pulse length from the signal generator. A ring-down period can be observed after the 24^{th} cycle, indicating actual total pulse length is longer. This ringing can be attributed to resonance in the transmitter and the receiver transducers, which are identical and tuned to the specific ultrasonic frequency used.

On measurements where the surface was close to the sensors, multiple echoes can be seen corresponding to the distance between transducer face and liquid surface. The secondary and tertiary echoes can be seen to have a similar envelope shape (Figure 6.6).

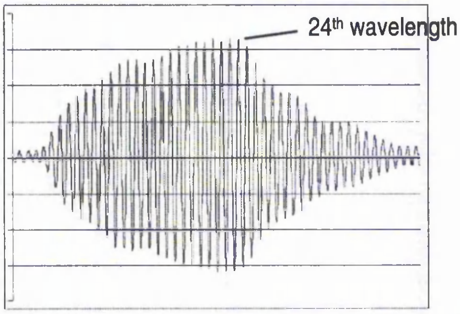


Figure 6.4 Ultrasonic Echo 65 points peak-peak

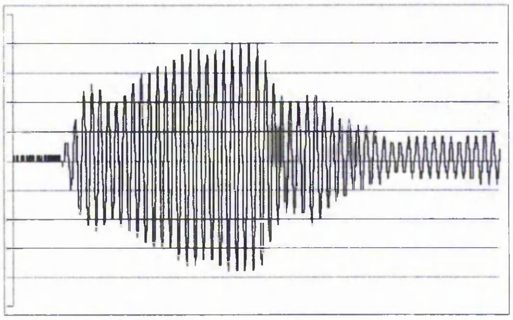


Figure 6.5 Ultrasonic Echo 40 points peak-peak

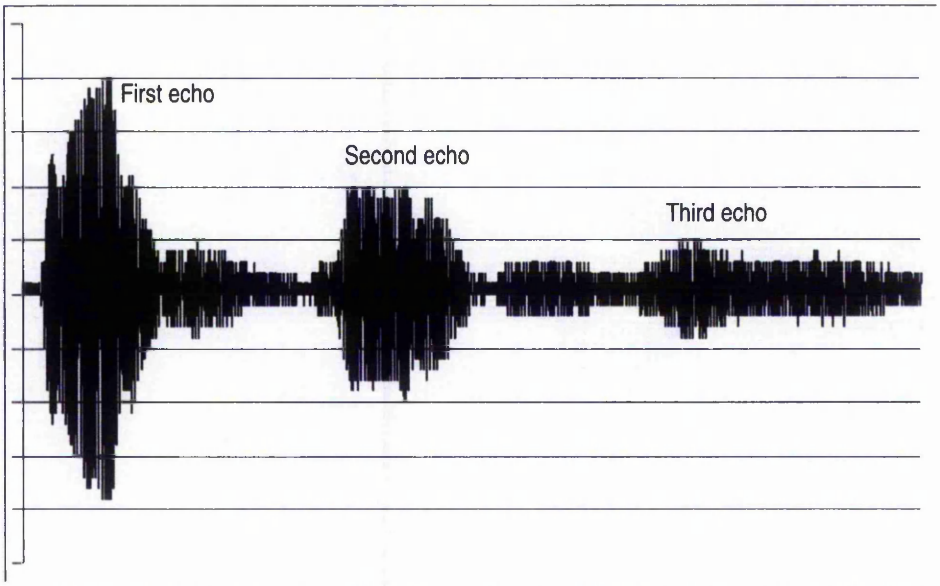


Figure 6.6 Ultrasound Echo Envelopes of a Signal Reflected from a Plane Surface

Figure 6.7 shows a graph showing measured liquid height, as found using hardware threshold monitoring, illustrates two process cycles (fill-empty-fill-empty). During the last emptying stage the valve was manually overridden three times to create areas of static level measurement and confirm that these could be accurately determined from the ultrasonic transducer method.

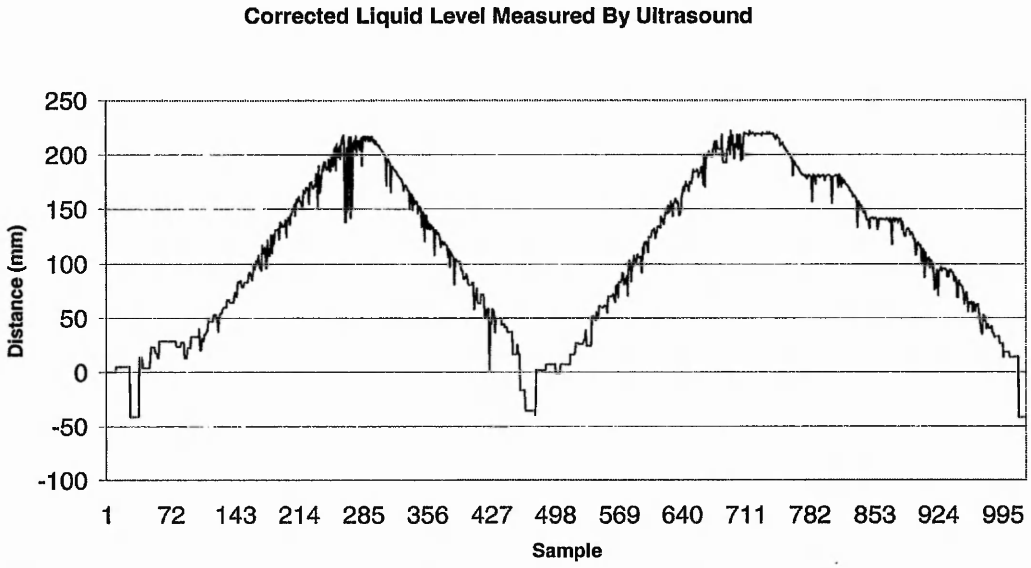


Figure 6.7 Liquid Level of a Vessel in Fill-Empty-Fill-Empty Cycle, as measured by Ultrasound

As clarification of the error observed if the distance found from the time-of-flight is simply halved to find the liquid height rather than using the method described in section 6.2.2.2, Figure 6.8 shows the increasing error from this over-simplification. It can be seen that error rises from 1mm to as large as 8mm at the maximum fill height through a non-linear curve, thus indicating the importance of using Equation 6.2 when transmitter and receiver are not concentric.

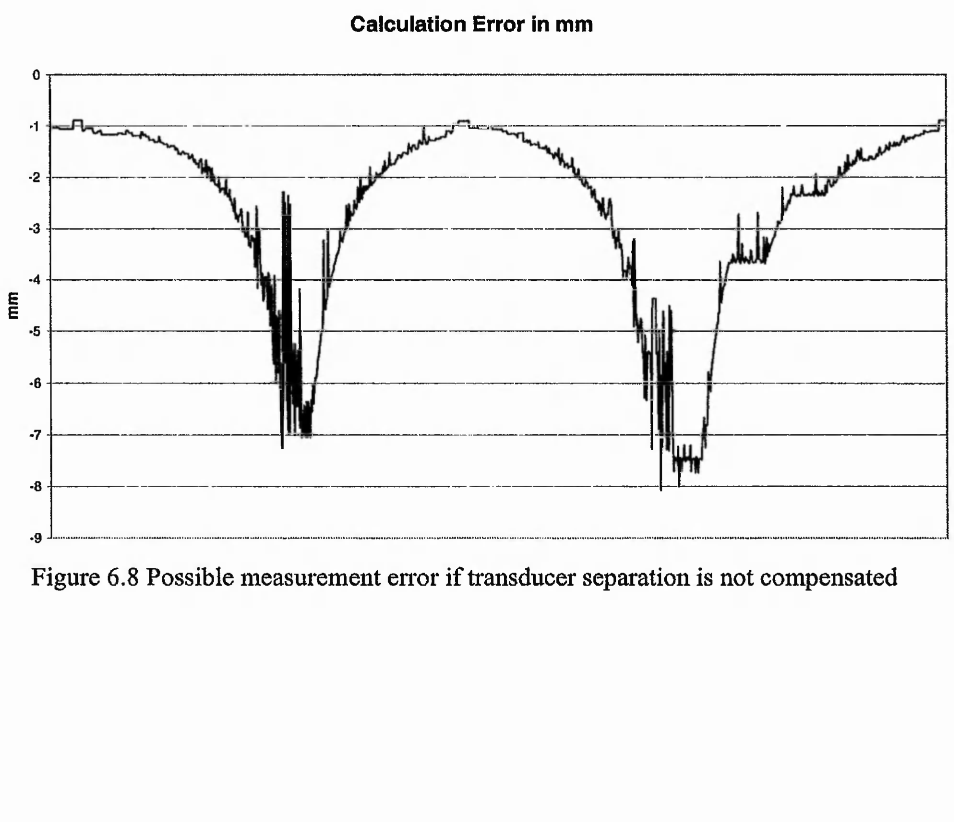


Figure 6.8 Possible measurement error if transducer separation is not compensated

In order to classify the noise present within the level measurement it is necessary to compare the data with an ideal data set. Within the experiment conducted here, the flow rate was maintained at a constant level, such that the fill gradient should compare favourably to a straight line. By finding the best-fit straight line then calculating the difference between the measured data and that line (Figure 6.9.), a table of errors can be formed, which can be further processed to form a histogram showing the distribution of noise during the fill stage (Figures 6.10 and 6.11). This approach can be employed for emptying; however, a slight curvature can be expected from the reducing head slowing the flow rate slightly. Nevertheless, a good indication of the noise distribution can still be found (Figure 6.12).

All three sets show a Gaussian-like distribution with a distinct skew to the right. This indicates a tendency for the ultrasonic sensors to over-estimate the distance of the surface, hence, under-estimate the overall height of the liquid. The spread of the data is 30 to 35mm. As a by-product of using a best-fit line, the zero error is shifted towards the negative by the spread of these under-estimated values. Thus the true liquid height can reasonably be expected to be 3 to 5mm in the positive direction from zero error, at approximately the centre-of-gravity of the main distribution density.

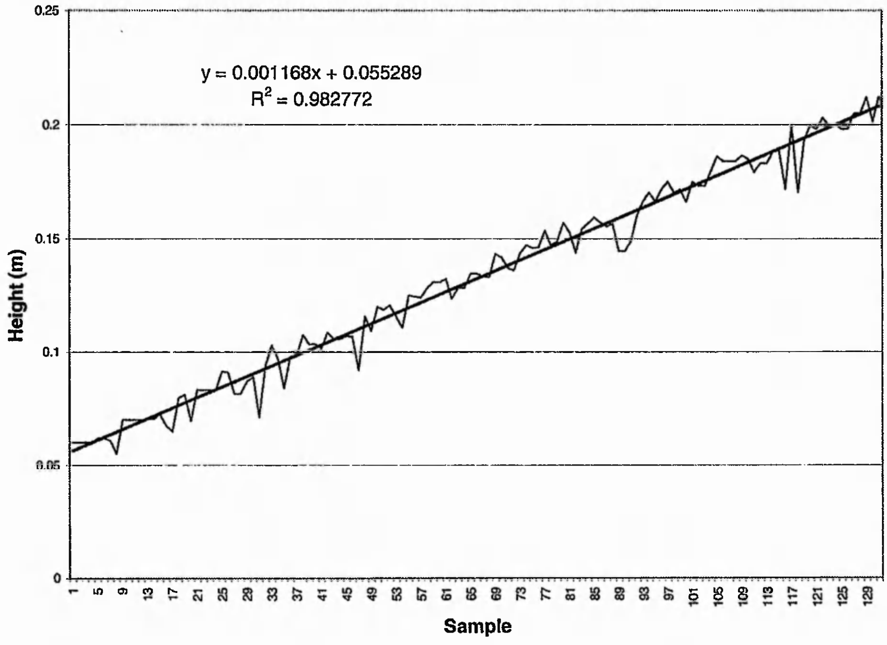


Figure 6.9 Best-fit line for error distribution calculation to determine noise characteristic

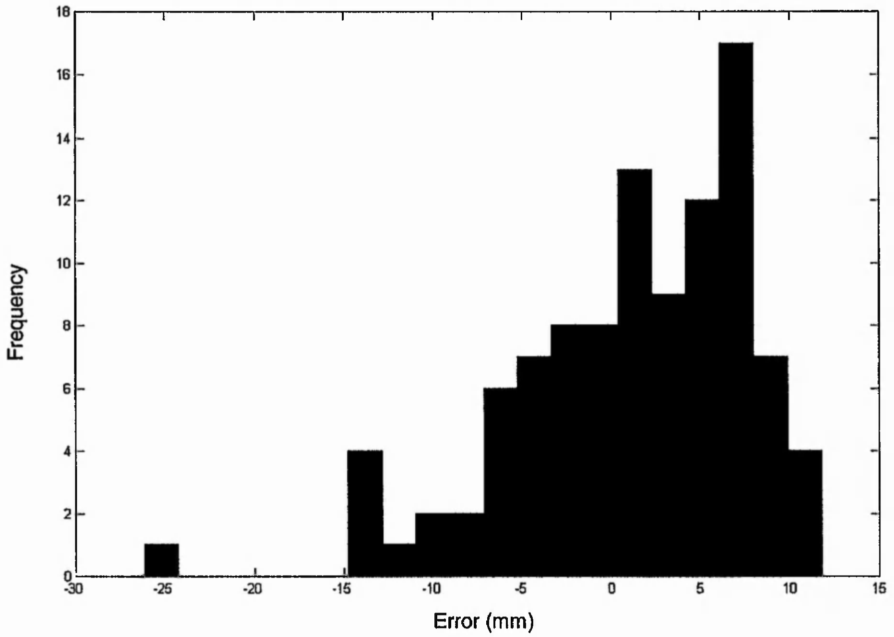


Figure 6.10 Measurement Error Distribution from 1st Fill

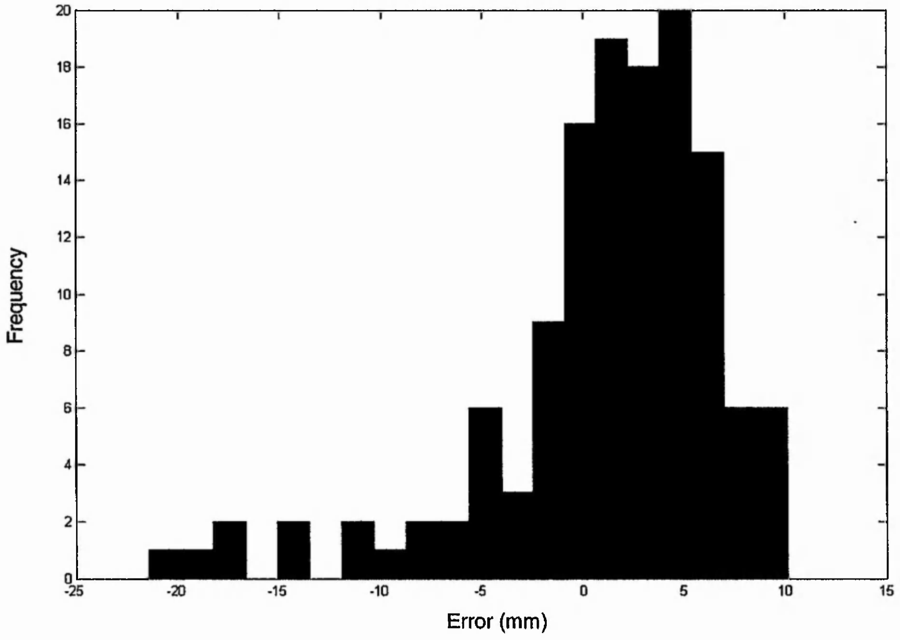


Figure 6.11 Measurement Error Distribution from 2nd Fill

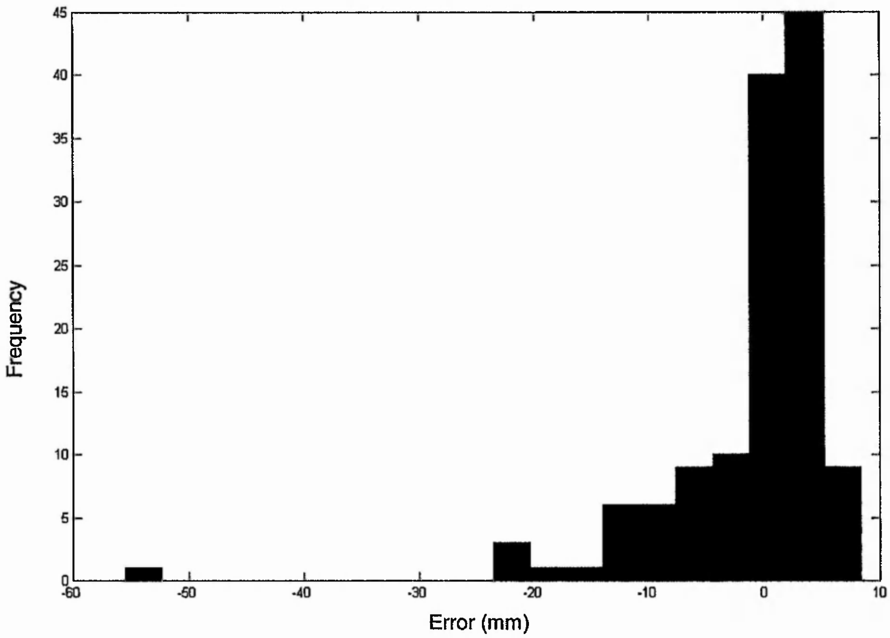


Figure 6.12 Measurement Error Distribution from emptying phase

In this experiment only surface agitation can be considered a major influence of ultrasonic performance, as the gas is air and temperature is stable, such that attenuation and scattering within the atmosphere above the liquid can be considered negligible. The surface agitation has two factors contributing to the distribution of noise. Firstly, the waves on the surface will cause the instantaneous level to vary about the mean level, which will be constantly increasing. This can be expected to produce a Gaussian-like distribution. The second factor of agitation is a degree of scattering, such that the first ultrasonic echo received may not have travelled the shortest distance from sensor to surface (Figure 6.13).

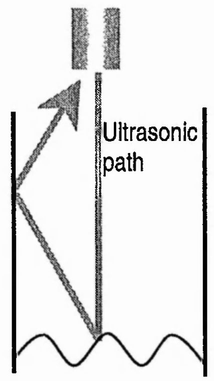


Figure 6.13 Possible Ultrasonic Reflective Path from a Turbulent Liquid Surface

Moreover, the threshold technique used to determine time-of-flight (TOF) is vulnerable to variation in echo envelopes from attenuation, leading to an apparent increase in time-of-flight (Figure 6.14). Both of these will lead to a skew of the data towards the right from an increased negative tail to the distribution. The obtained results therefore verify the expected performance of the ultrasound within the presented conditions.

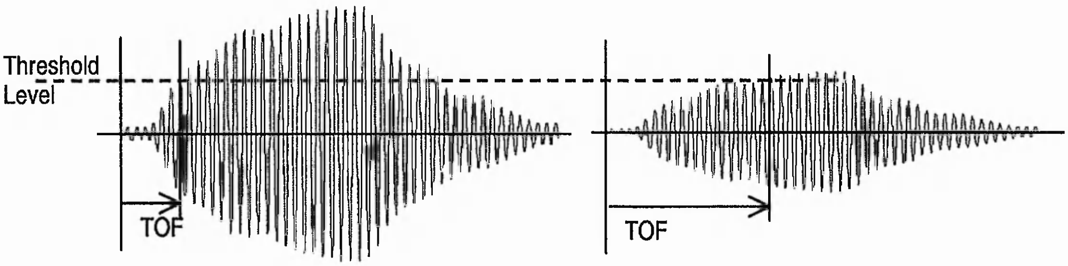


Figure 6.14 Amplitude Variation in Ultrasound Echo Envelopes

Other points of interest seen whilst conducting the experiments include large errors seen from drips hitting the surface from the valve during static measurements and the invulnerability to electrical noise from the solenoid valve actuation that plagued the pressure transducer measurements. This could have useful implications for implementation within an actual filling plant where electrical interference could be high.

6.3 Implications of Experiments to Plant Simulation

As a first approximation the noise attributed to the level measurement could be modelled as a standard normal curve, but although this would provide an adequate test of the Kalman filter, a better distribution is required to fully model the distinctive skew in the data.

A good candidate for a better noise model is the Gamma distribution, which through two parameters, α and β, can be made to represent a number of different types, including exponential and skewed data sets. The probability density of the Gamma distribution can be described by the following equation (Freund, 1992).

$$f(x) = \frac{1}{\beta^{\alpha}\Gamma(\alpha)} x^{\alpha-1} e^{-x/\beta} \qquad \text{for } x > 0 \tag{5.7}$$

$$f(x) = 0 \qquad \text{otherwise}$$

However, in order to use the distribution as a noise source within the simulation it is necessary to have an algorithm that can generate the density function from the uniformly distributed random number generator within computer systems. Devroye (1986), describes an algorithm by Best (1978), which is an efficient means of producing the Gamma distribution within a computer program for α values greater than 2. It is called Best's Rejection Algorithm XG and can be written as follows,

$b \leftarrow \alpha - 1, c \leftarrow 3\alpha - 3/4$

REPEAT

$u \leftarrow uniform[0,1], v \leftarrow uniform[0,1]$

$w \leftarrow u(1-u), y \leftarrow \sqrt{\frac{c}{w}}(u - \frac{1}{2}), x \leftarrow b + y$

IF (x >= 0)

$z = 64w^3v^2$

IF ($z <= 2y^2/x$) *accept*

IF NOT *accept*

IF ($\log(z) <= 2(b\log(x/b) - y)$) *accept*

UNTIL (*accept*)

$r = x\beta$

where r is a random variable with a gamma distribution

In this form the distribution can be applied within the plant model. Selecting values of α greater than 2 gives skewed distributions of a shape similar to that observed in the measurement data. By iterative analysis, an α value of around 8 gave the closest approximation to the distribution shape. The β value defines the spread of the density and, in this case, was found best to work with a value of -1 to give the closest approximation to the observed data spread. A negative value of β was chosen to map the tail of the distribution towards negative infinity.

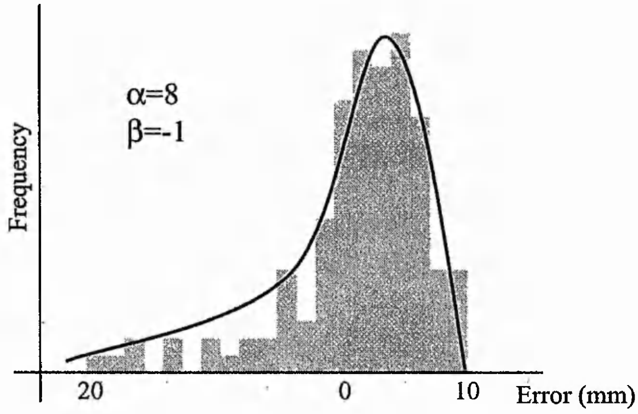


Figure 6.15 Comparison of Idealised Gamma Curve and Actual Measurement Data.

Figure 6.15 (also see figures 6.18 and 6.19 in section 6.4.2.2), compares the measured noise distribution, to that generated from the Gamma distribution with the chosen input parameters. It can be seen that for the experimental conditions the Gamma distribution is an excellent representation of the process.

6.4 Single Vessel Model Tests

With a good model for the type of noise expected from the level measurement it is possible to run the developed simulation (Section 5.5) to test the effectiveness of the Kalman filter. This section shows results from varying parameters, such as valve characteristics, initial Kalman parameters and noise levels. A model of the test rig is used for some initial validation, and then a model more closely matching actual filling plant is used. In addition, a comparison between using Gamma and Normal distribution, for which the Kalman filter is primarily designed, is made. Emphasis is strongly placed on the ability of the filter to track or monitor the fill height, rather than achievement of any set points by the controller, which is ultimately reliant on the tracking of the height parameter, rather than any control characteristics.

6.4.1 Kalman Filter Iteration Time

As mentioned previously in Section 3.3, the iteration time of the Kalman Filter can have an impact on the success of parameter estimation, either to affect the linearisation of non-linearities, or to adapt to noise more quickly. To have a very fast iteration time is computationally expensive and also the smallest iteration time is limited to the transit time of the ultrasound (~1.5ms @ 0.5m). To better understand the Filter iteration performance, a series of tests were carried out to show the improvements in noise reduction at a number of different iteration times and three different initial measurement Gaussian noise levels ($\sigma = 0.1, 0.5, 1$ cm). The results, as shown in Figure 6.16, show that the greatest improvements occur initially around the 1 to 0.1 second region, with flattening of the curve after this, therefore reducing iteration below 0.01 seconds (10ms) would not provide sufficiently better noise reduction performance to warrant faster iteration time. Hence, the ultrasound transit time cannot be considered a limiting factor on system performance. Further simulations of varying noise levels is discussed in Section 6.4.5.

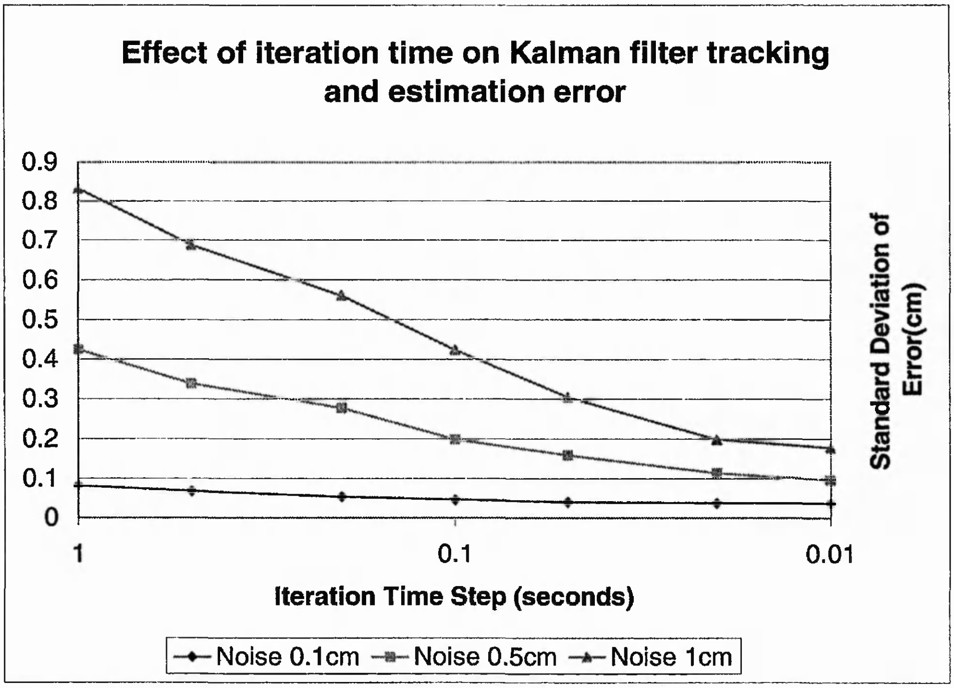


Figure 6.16 Estimation Error as related to Kalman Filter Iteration Time Step

6.4.2 Gamma versus Gaussian Distributions

The Kalman filter is generally applied to systems with zero mean bias and normal Gaussian distribution. However, in this case the noise expected from the level measurement has both a bias in the mean (i.e. the mean does not represent the actual value of the parameter) and a skewed distribution resulting in a general under-estimation of the true fill height. Several tests were run to determine the effect of different noise models.

6.4.2.1 Assumptions

- The flow rate will be constant
- The set point will be constant
- The measure of success will be the deviation of the estimate from the true value
- The final fill height will not be taken as a measure of filter effectiveness

The reason fill height will not (and cannot) be taken as a measure of the Kalman filter success is due to limitations of the simulation at this stage. The model is configured such that the iterative time of the plant model and controller is locked to the iteration time of the filter and noise; therefore, the controller can only operate at fixed timing intervals. Thus, unable to make control decisions between filter iterations the opening and closing of the valve is locked to discrete time steps and final fill height suffers as a result. For example, a flow rate of 90cm^3/s, which will fill a 0.5 litre vessel in around 5.5 seconds, will cause the surface speed of the liquid to be around 28cm/s in a narrow bottle neck of 2cm diameter. Even with an iteration time as fast as 0.005 seconds the controller will not be able to affect fill height in less steps of less than 1.4 mm. In practice, an actual controller can be designed to make timing decisions independent of the measurement interval.

However, the accuracy of the monitoring is unaffected by this timing property as the underlying mathematical models of the simulation are continuous in nature, thus are merely sampled at the measurement time step.

6.4.2.2 Results

The standard deviation of the Normal distribution was set so that the spread was similar to that of the pre-determined Gamma distribution. It can be seen from Figures 6.17 and 6.18 that the spread of the estimates in either case is similar, the most noticeable difference being a bias towards overestimation with Gamma input noise. This is to be expected given that the Kalman filter is designed for zero bias noise, whereas the Gamma distribution of noise found from the experimental ultrasound results has a distinct bias. A means of overcoming this problem depends on what the bias is attributed to. If the bias were a real phenomenon then an extra term would have to be added to the filter. If, however, it is the result of the location of best-fit line originally used to find the noise (section 6.2.2.3) then the distribution can be moved to offset the bias found in the Kalman estimates. This second scenario is more likely cause as the best fit line assumes a normal distribution, thus an extended tail in the gamma distribution will pull the line towards a mean value which is not representative of the zero error position. In consideration of this, the noise can be moved to place the centre of gravity closer to zero error, hence, correcting the artificially introduced offset.

With the Gamma distribution moved to reduce the estimation bias, it can be seen from Figure 6.19 that no significant detriment to Kalman filter performance is created from using the chosen Gamma distribution compared to the Normal distribution for which the filter is designed.

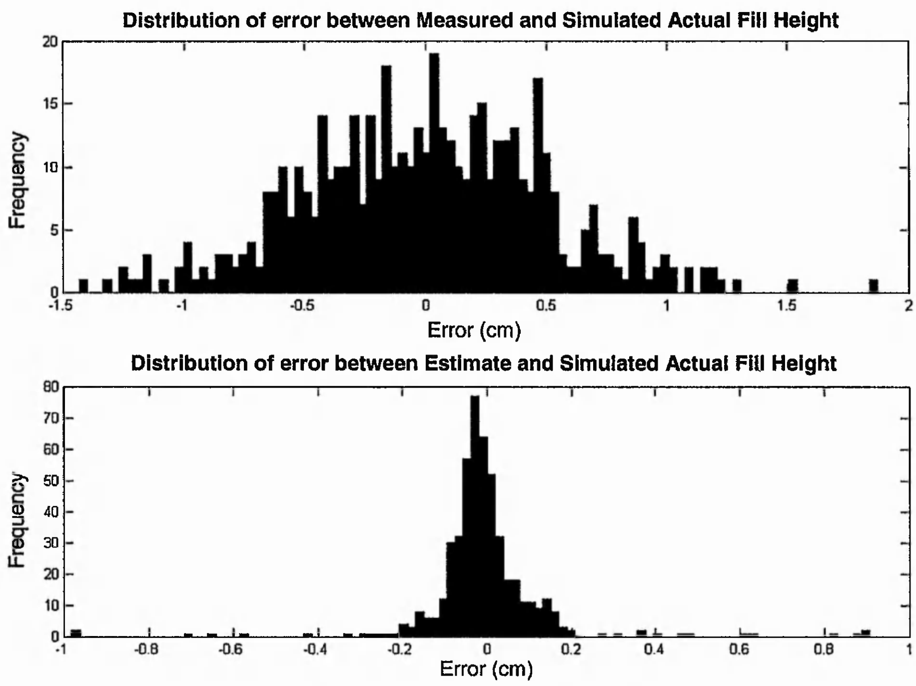


Figure 6.17 Measurement Noise Modelled with Normal Distribution

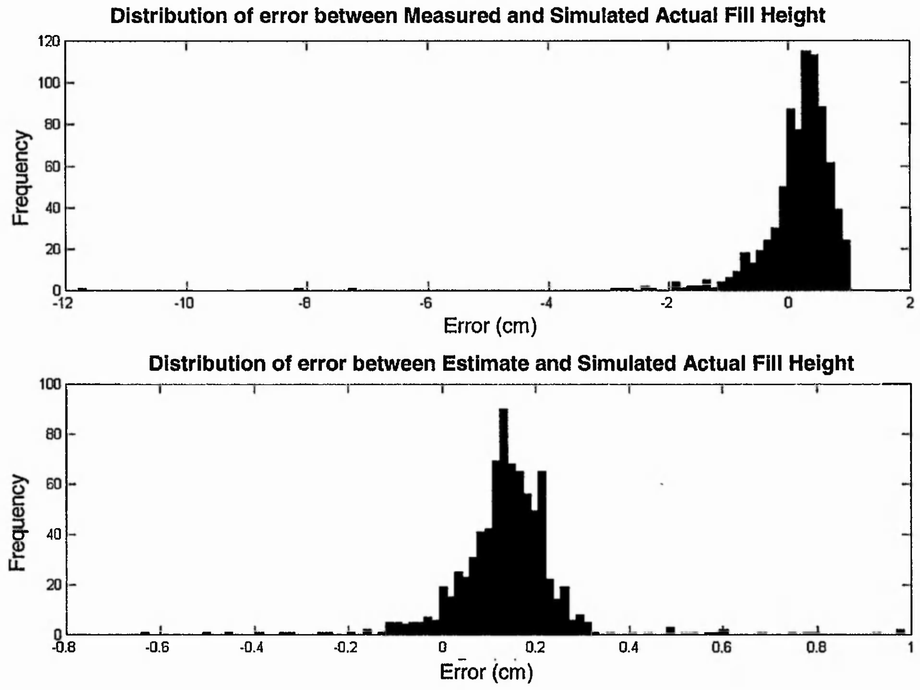


Figure 6.18 Measurement Noise Modelled with Biased Gamma Distribution

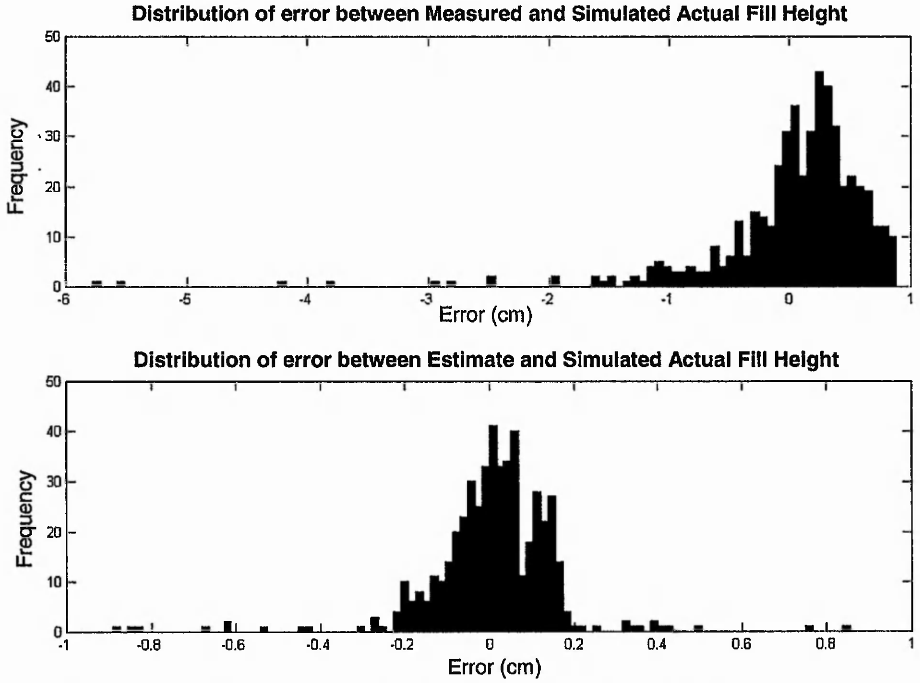


Figure 6.19 Measurement Noise Modelled with Corrected Gamma Distribution

6.4.2.3 Remarks

With the characteristics from the experimental data reported in section 6.2.2.3, it can be considered that a gamma distribution gives very similar results to the Gaussian distribution in Kalman estimates. This is helpful in determining the noise statistics for setting up the Kalman filter. Moreover, using a standard deviation value as a measure of the spread will be a fair assumption at this stage. It can also be noted that the estimate demonstrates a Gaussian-like distribution, despite the gamma distribution of the measurements.

6.4.3 Simulation of the Test Rig

By placing the shape of the test rig filling vessel and filling characteristics within the developed simulation it is possible to investigate some of the operation of the test rig. An element that is not modelled is the emptying stage of the cycle, which is necessary part of the rig but is not part of actual filling plant. The aim of this simulation is to show the validity of the models used in comparison to the experimental data obtained from the test rig. However, the bulk of the simulation results will be in reference to more complicated situations than that of the test rig, i.e. more elaborate filling shapes, and later a multiple valve carousel.

6.4.3.1 Test Parameters

The vessel has a diameter of 95mm and can be filled to a height 200mm. From the experimental data it can be found that the flow rate was approximately 90 cm^3/s and the filling time was 15 seconds.

Gamma noise with a distribution similar to that found by the ultrasound experiments was used throughout the model, ($\alpha = 8$, $\beta = -1$; see Section 6.3). It was scaled to match the numerical spread of real data. It is necessary to describe the variance of the noise for use by the Kalman filter. With Gaussian distributions the value is easy to obtain, but this is harder to define for Gamma distributions. To overcome this, an estimate of the variance was made by taking the standard deviation of the distribution, which is only slightly skewed and hence, not overly incorrect in describing the data.

As already mentioned in Section 5.5.1, the internal model of the filter uses volume rather than height as the control parameter, to keep the linearity of the model independent of the vessel shape. Therefore, the variance value was converted to represent changes in volume rather than height by simply multiplying by the cross-sectional area of the vessel, which in this case was constant over the filling range.

6.4.3.2 Results

The first thing to note is the similarity between the noise distribution found by experiment and the distribution used within the model (Figures 6.20 and 6.21). It can be seen that the spread from +10mm to –20mm is closely matched and the shape follows very similar contours. Considering that random variables are involved the likeness is very strong. This supports the choice of Gamma distribution for level measurement noise modelling.

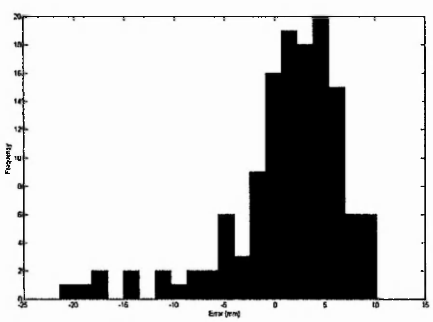


Figure 6.20 Experimental Measurement Noise Distribution

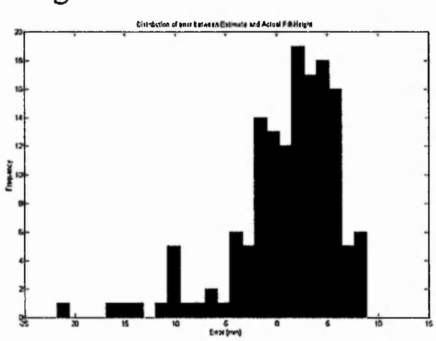


Figure 6.21 Model Measurement Noise Distribution

It was found that the spread of the distribution from the Kalman state estimate was reduced by approximately a factor of 4 (σ reduced from 0.61 to 0.16) to that obtained from measurement. Overall the filter provided good estimates, which was to be expected in this relatively simple case.

With no complicated characteristics the Kalman gain and error covariance showed good stability throughout the filling cycle. Figure 6.22 shows Kalman Gain.

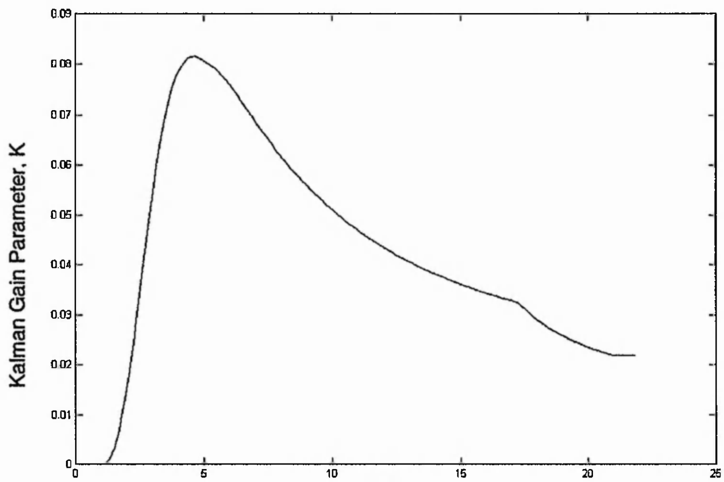


Figure 6.22 Kalman Gain Parameter under nominal operating conditions

As a point of interest, the model finishes calculations several orders of magnitude faster than the time elapsed within the model. This indicates real-time application of the Kalman filter would not impinge on the process speed.

From this simple simulation it can be seen that ultrasound liquid level measuring process, can be successfully modelled within the computer and the Gamma distribution produces excellent results. This validates the approach for further study into the filling process.

6.4.4 Initialisation Characteristics

In reference to the performance variations mentioned in section 3.4 regarding initialisation of the error covariance matrix, namely whether the filter uses the internal model or measurements for initial estimates, it can be demonstrated that possible benefits can be gained within the bottle filling application from this characteristic. Using two intialisation values of 0 and 99999 for error covariance P, and identical other parameters, it can be seen from figures 6.23 and 6.24, that destabilising the filter initially, (P = 99999) provides better coverage for un-modelled or non-linear factors. The non-linearity in the first two seconds shown in the figures was initiated by slowing the opening factor of the valve (see section 6.4.8 for further information on valve characteristics). This feature can also be used at any point where stability can be relaxed for tracking ability, that is anywhere except around the set point, thus is useful for complicated container shapes or slow valves.

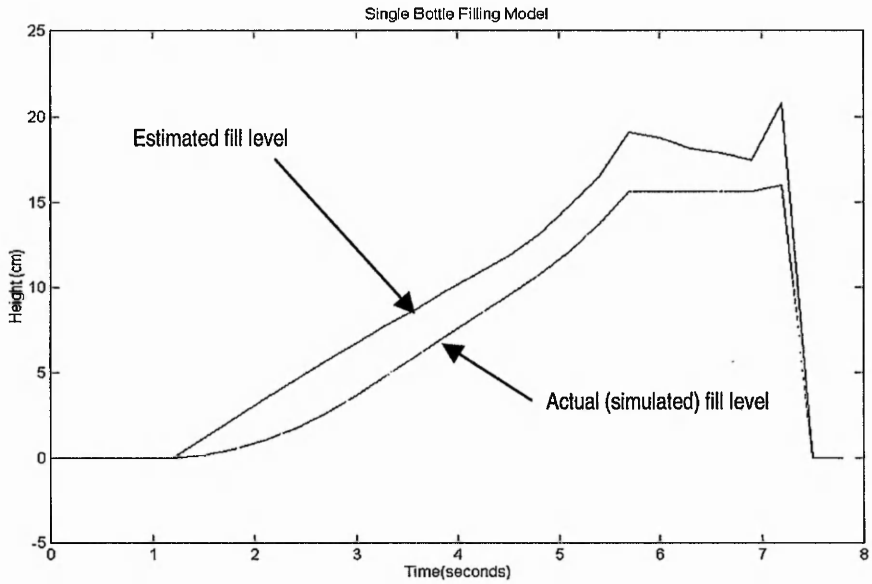


Figure 6.23 Level Measurement Estimate with Kalman Filter Error Covariance, P = 0

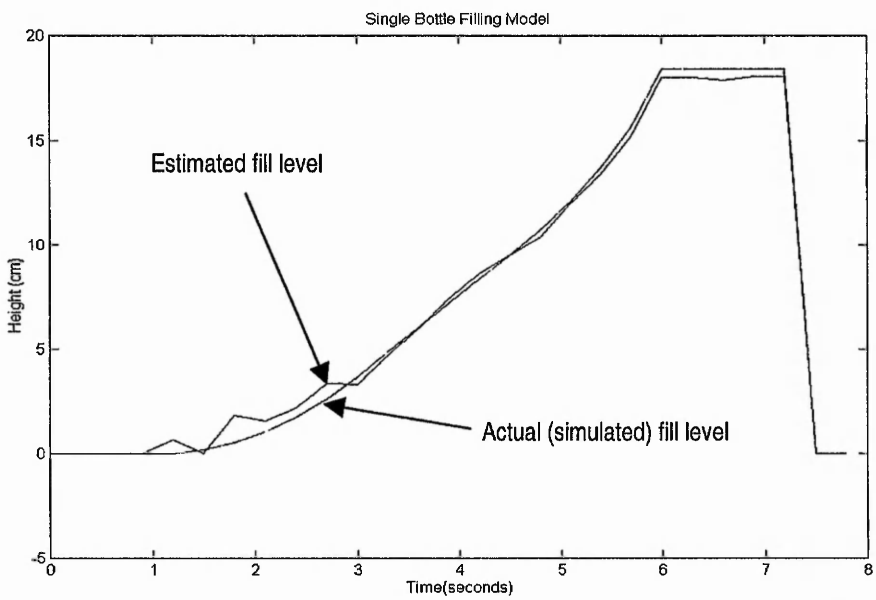


Figure 6.24 Level Measurement Estimate with Kalman Filter Error Covariance, P = 99999

6.4.5 Bottle Shape

With the noise model proven to be representative of both the process and stable for use with the Kalman filter, it is necessary to change to a slightly more complex vessel model. Using a simple cylinder for the main body and a linear cone for a neck, a shape was created for the bottle model to be both simple and representative of the types likely to be encountered on an actual filling plant.

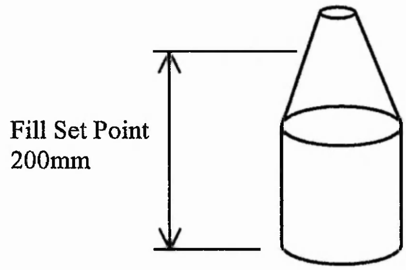


Figure 6.25 Simple Bottle Shape with Chosen Fill Height for Plant Simulations

The ideal fill height was chosen to be 200mm and the crossover between cylinder and cone to be 100mm (Figure 6.25). This allows for a mainly linear height-volume relationship with slight non-linear curvature approaching the set-point. The diameter of the cylindrical vessel was set to 70mm. This model will be used as the basis for all further simulations, including those found in the multiple-valve model discussed in depth in the next chapter.

6.4.6 Noise levels

The noise levels measured in the ultrasound experiment (Section 6.2.2) can be considered to be one of the more ideal measurement cases. With room temperature and standard atmospheric conditions, the noise levels witnessed are due mostly to the agitation of the liquid surface. With weaker signals from attenuating gases (such as Carbon dioxide) and variations in temperature causing scattering it can be reasonable to assume greater levels of measurement deviation from the true fill height will be seen. However, it can also be expected that the noise in the height measurement will exhibit similar distribution properties to those already examined. In any case, the Gamma distribution offers a wide range of variability from altering the two shape parameters α and β, such that any future experimental work undertaken beyond the study presented here, the simulation can be modified to mimic the conditions witnessed under sub-optimal conditions.

A number of tests were carried out to investigate the performance of the system under a range of noise levels.

6.4.6.1 Assumptions

For the tests conducted here it is assumed that the shape of the gamma distribution is identical to that used previously and that scaling of the distribution will be representative of increased levels of noise. The aim will be to show that the Kalman filter remains stable under more difficult conditions and also to categorise the resultant estimation errors.

As before it will be necessary to convert the standard deviation of the height to a variance in volume. However, by using the bottle shape from Section 6.4.5 the relationship of height and volume is not constant, such that the variance in volume will decrease as the fill height reaches the top of the vessel given the same disturbance in measurements. Nevertheless, by assuming the worst case scenario, which occurs lower down in the vessel, the Kalman filter is not unduly affected. This is because if measurements are better than expected then the filter estimates improve accordingly, even if the variance is overestimated. This is preferable to under-estimating the variance (i.e. matching the variance around the set point), such that estimates are unnecessarily noisy at the lower portion of the vessel.

The flow rate and fill time was set to mimic the operation of a filling plant with a flow rate of 150 cm^3/s and fill time of around 4 to 6 seconds. The filter iteration time was set at a reasonable 0.05 seconds and the error covariance matrix initialised to 0.

6.4.6.2 Results

Five noise levels were tested with approximate height standard deviations of 0.1, 0.5, 1, 2 and 5 cm. These are equivalent to variances in volume of 15, 370, 1481, 5924 and 37026 cm^3 for a bottle radius of 3.5 cm. The model variance is set at $0.1cm^3$, which over-estimates the true value of 0, as the model is exactly representative of the process (both use the same equations). However, by increasing the error, the estimates are destabilised in a manner similar to that which can be expected in an actual application.

Four parameters are shown to demonstrate the performance of the filter, height (measurement and estimate), estimate height error distribution and percentage error in volume estimates.

With a standard deviation in the measured height of 0.1cm, the simulation represents a better scenario that what was achieved through experimentation. This can be seen in Figure 6.26, where the original measurements can be seen to provide a reasonably accurate representation of the filling process. Figure 6.27 shows that the Kalman filter offers only a small improvement at this noise level, height standard deviation is reduced from ~0.07cm of the original measurements to ~0.02cm in the new estimates. It should be noted that because of the random generation of the noise, the simulation does not always reproduce the requested deviation value (0.07cm in this case, compared to the requested 0.1cm), which is also true of real processes, in that a measured deviation may not always represent the process. These estimates represent a maximum volume error of generally less than ±0.2% (Figure 6.28). The large errors witnessed at the 30^{th} and 31^{st} second, which are products of the long tail of the gamma distribution, are quickly corrected, such that all final volumes are better than 0.1% of the total.

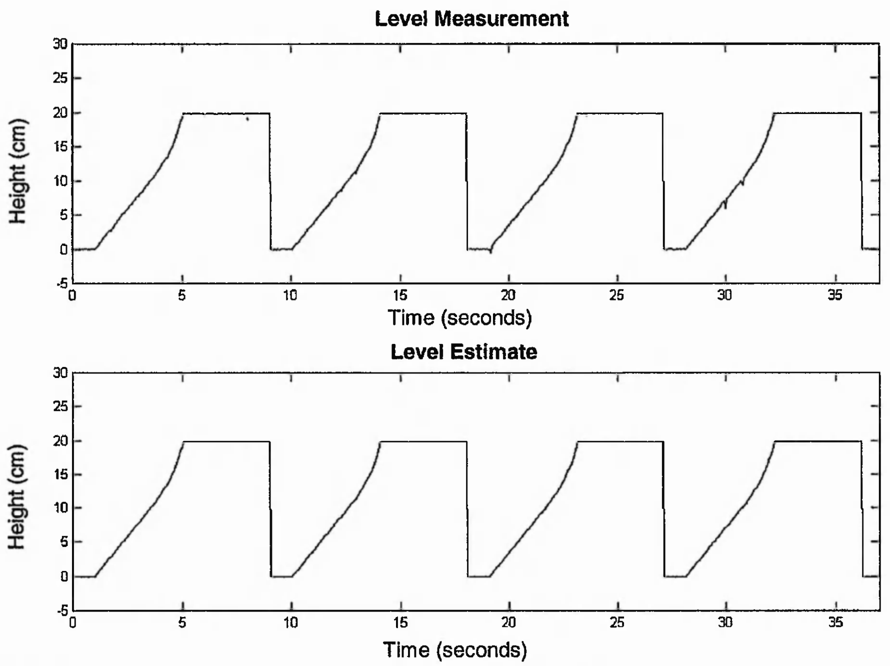


Figure 6.26 Level Measurement with Standard Deviation of 0.1cm

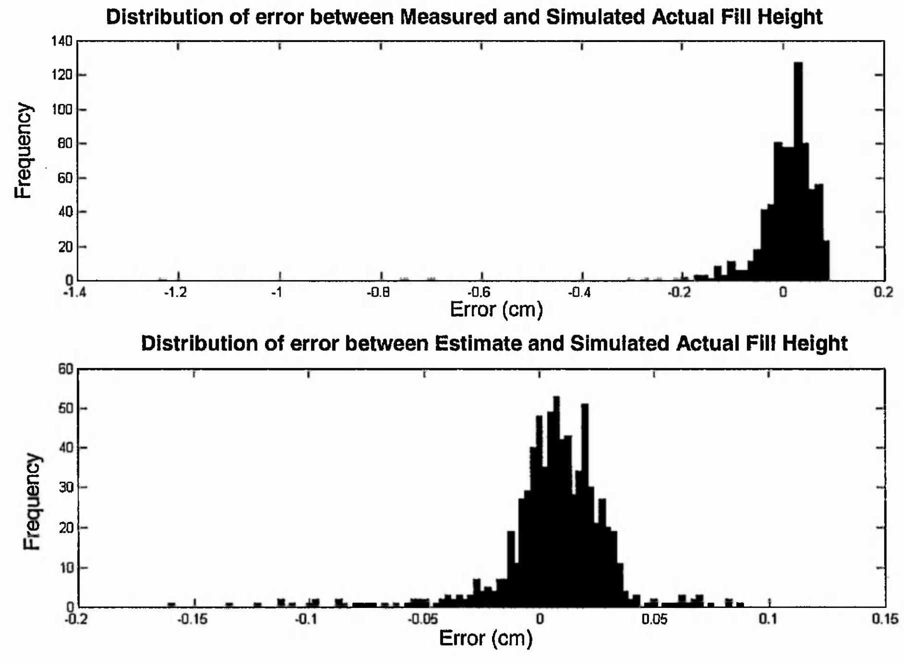


Figure 6.27 Error Distribution of Measurement and Estimate

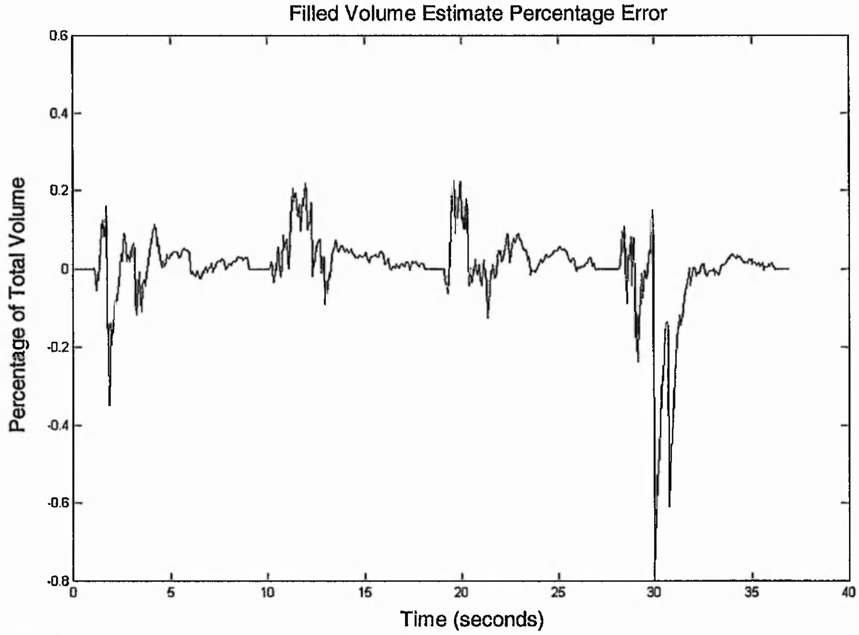


Figure 6.28 Volumetric Error Based on Filter Estimates

A standard deviation of 0.5cm is similar to that found from the ultrasound tests and represents are reasonable expectation of noise within the real process. Figure 6.29 shows more disturbance in the measurements than before, but the estimates remain stable. Standard deviation is reduced from 0.57cm to 0.23cm (figure 6.30), which is a similar order of magnitude reduction as seen in the previous example. Maximum percentage error in volume estimates has increased to a range of ±1.5%, but final estimates are less than ±0.5% (figure 6.31).

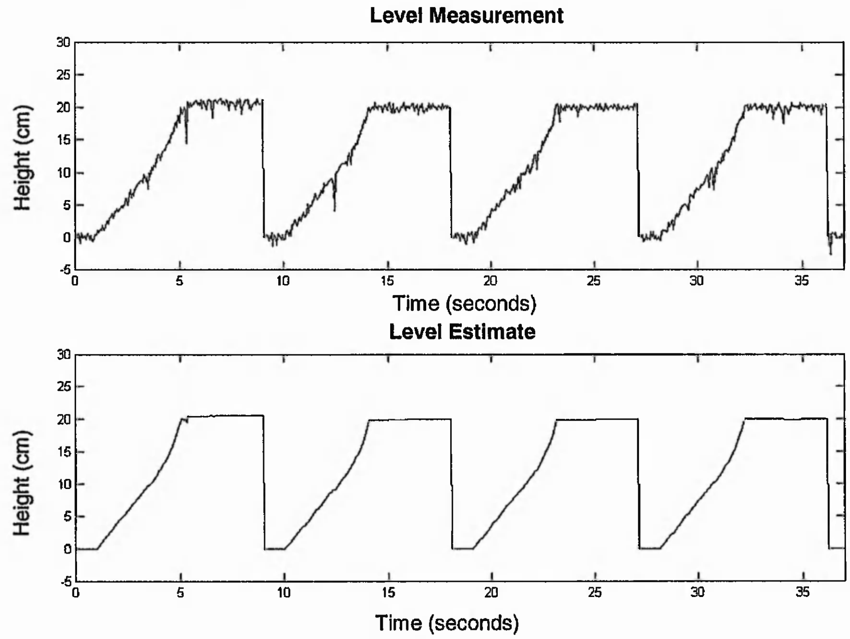


Figure 6.29 Level Measurement with Standard Deviation of 0.5cm

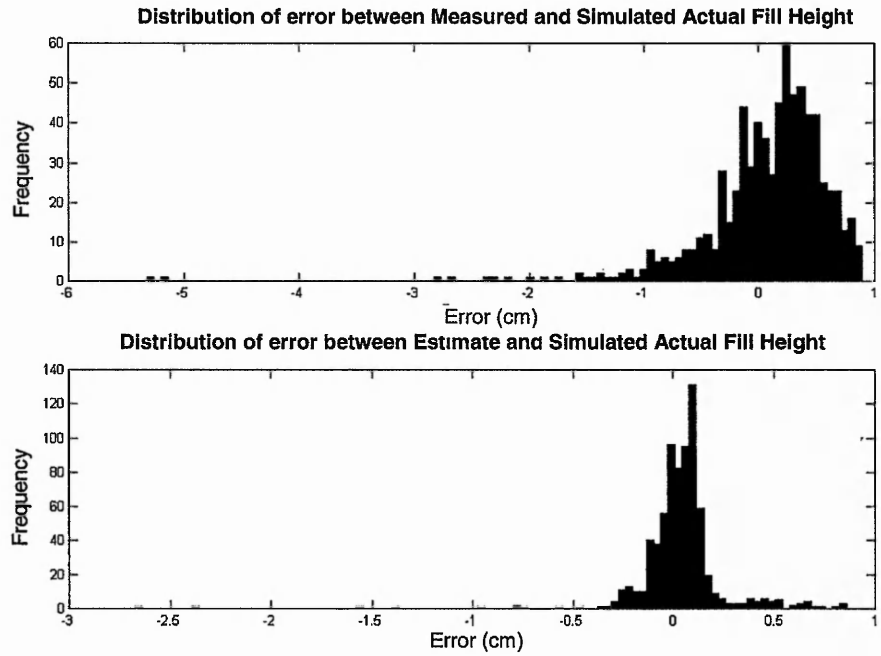


Figure 6.30 Error Distributions of Measurements and Estimates

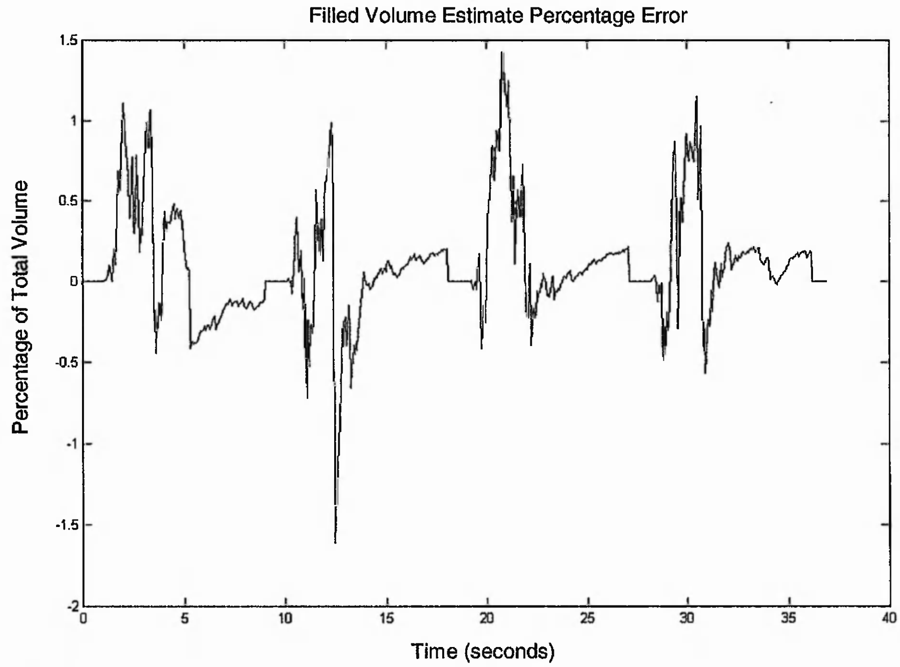


Figure 6.31 Volumetric Errors from Kalman Filter Estimates

With a specified standard deviation of 1cm the measurement signal is beginning to show significant amounts of noise (Figure 6.30). However, the estimate remains stable reducing the calculated standard deviation from 1.1cm to

0.4cm (Figure 6.33). Volumetric errors, as shown in Figure 6.34, do not increase significantly from the previous example.

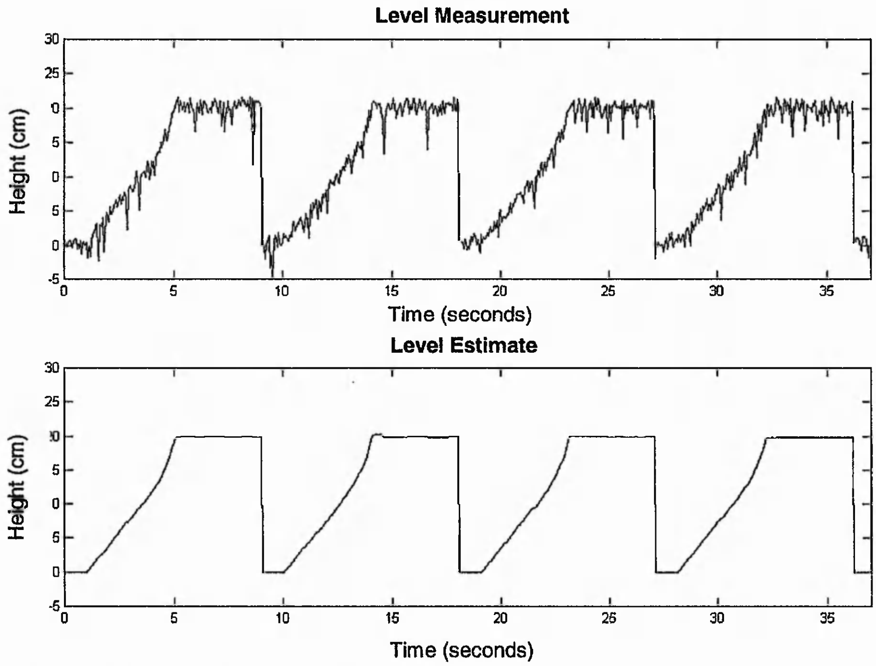


Figure 6.32 Level Measurement with Standard Deviation of 1cm

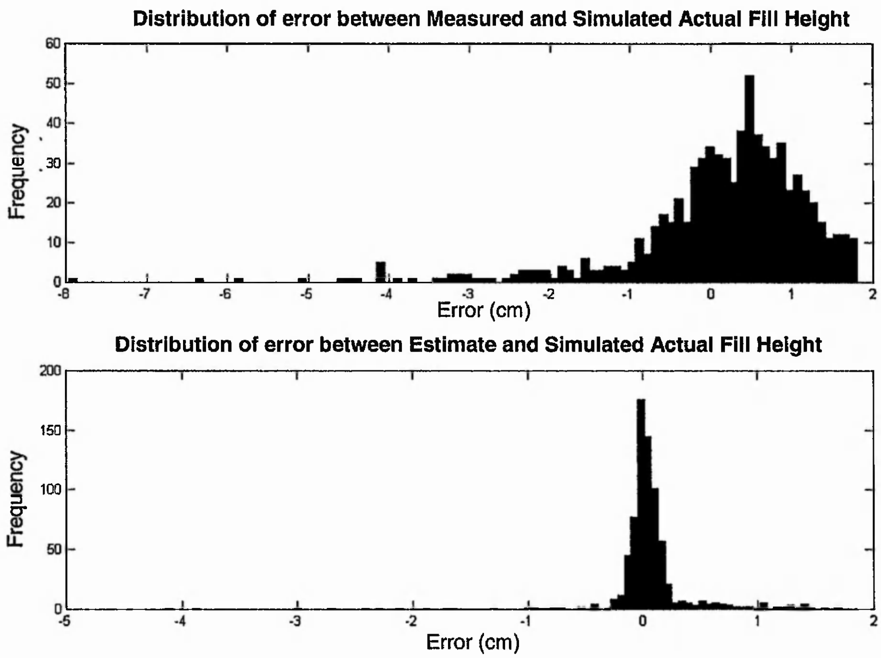


Figure 6.33 Error Distribution of Measurements and Estimates

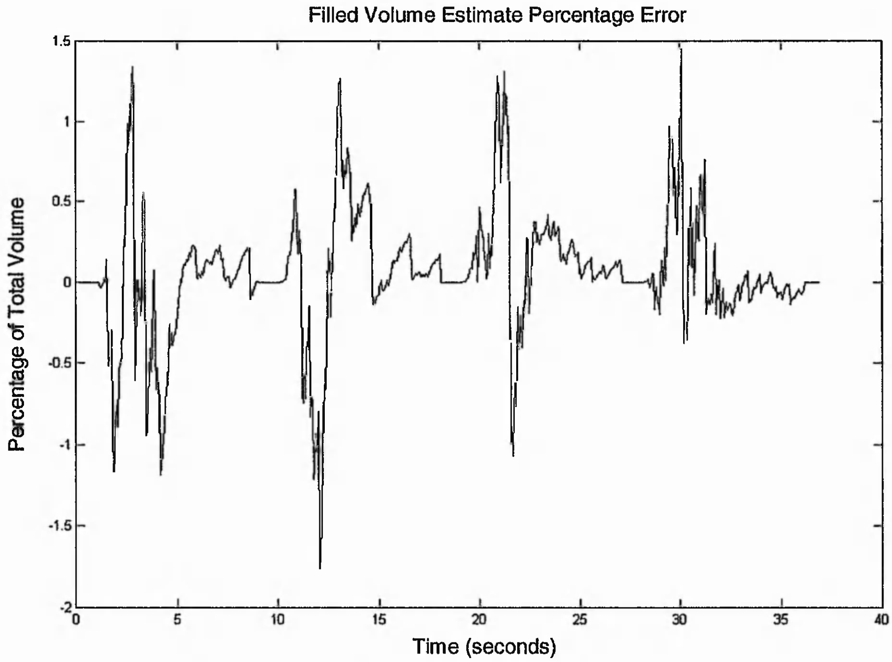


Figure 6.34 Volumetric Errors from Kalman Filter Estimates

As shown in Figure 6.35, a standard deviation of around 2cm begins to show significant disruption to the measured level. A feature of the gamma distribution is the long tail to one side; this is now becoming more prominent in the data with large dropouts at regular intervals. Nevertheless, estimates remain stable having a standard deviation of 0.8 compared to 2.4cm of the measurements.

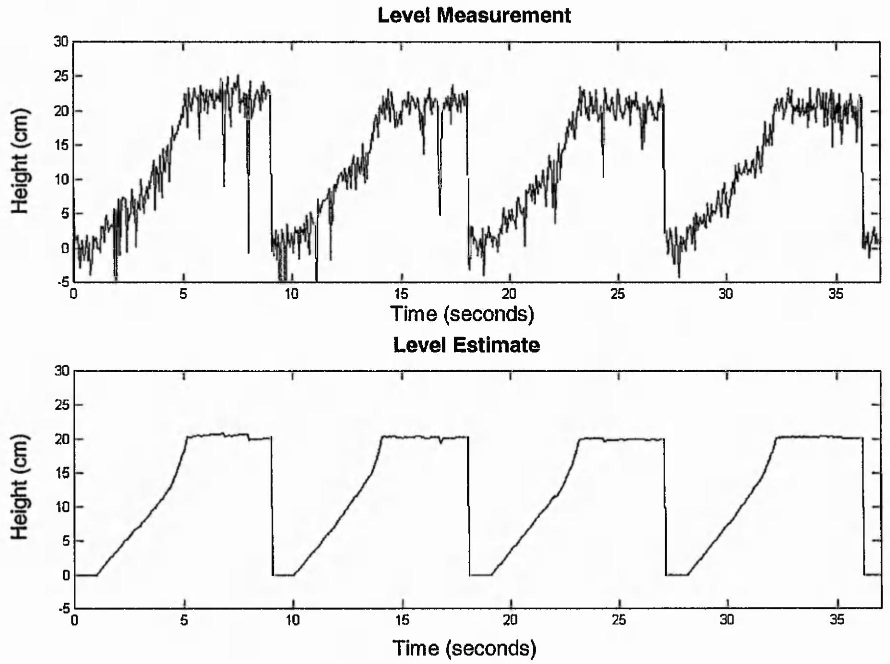


Figure 6.35 Level Measurements with Standard Deviation of 2cm

The shape of the estimate distribution looks less Gaussian in this example (Figure 6.36), although it should be noted that on different runs the distribution shape tends to vary from good Gaussian representation on some occasions and less clear shape on others. The volumetric error, however, shows significant variation from some large measurement errors in the first fill cycle (Figure 6.37).

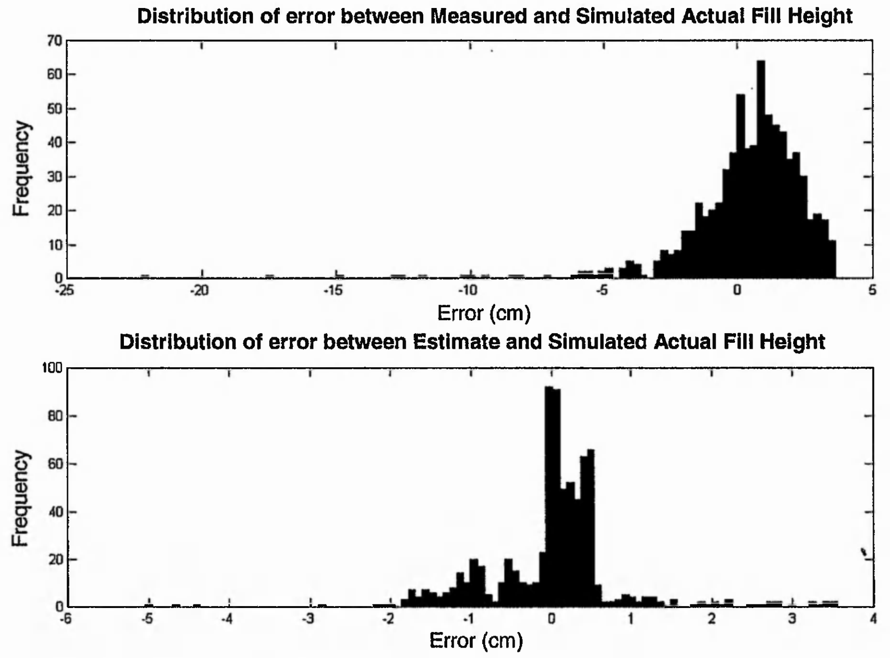


Figure 6.36 Error Distribution of Measurement and Estimates

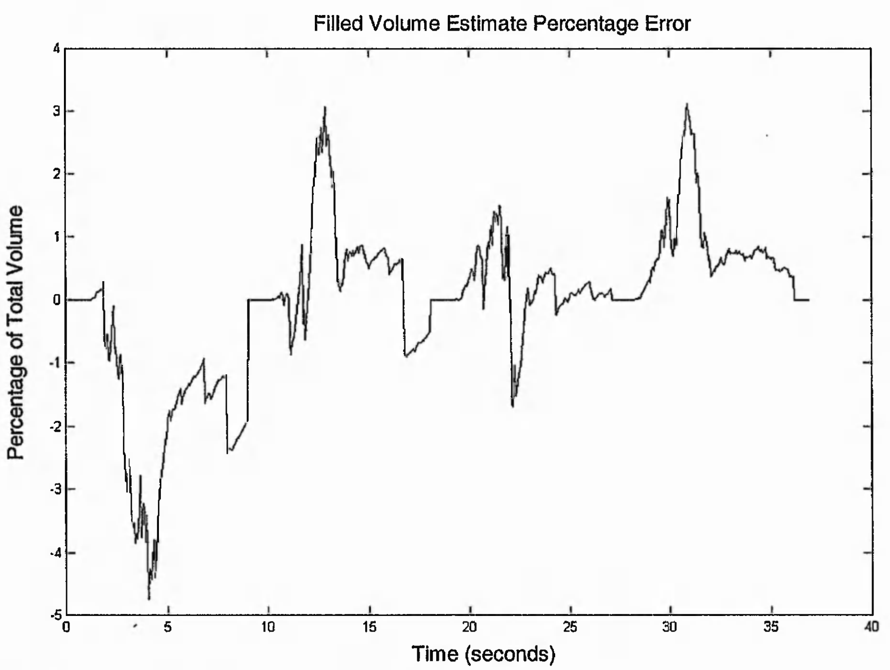


Figure 6.37 Volumetric Errors from Kalman Filter Estimates

In specifying a standard deviation of 5cm the measurements are now so significantly disturbed that control action on the pure signal would be impractical (Figure 6.38). The estimate remains stable for three out of four fills, however, fill no.3 shows a large overfill this can be attributed to a momentarily large underestimate in measurement around the critical set point. This has caused the controller to re-open the valve, which at this stage of filling causes a large increase in height because of the narrow diameter in the neck. Overall the performance is good with a reduction in standard deviation from 5.3cm to 2.4cm, with much of the spread caused by a few instants of large estimate errors, i.e. fill no.3 (Figure 6.39). As can be seen from the volumetric error in Figure 6.38, the estimates have not deteriorated significantly from previous runs, except for fill 3. This demonstrates stronger influence of the internal model and that the greatest risk at this noise level is not the general distribution, but occasional large errors at inconvenient moments.

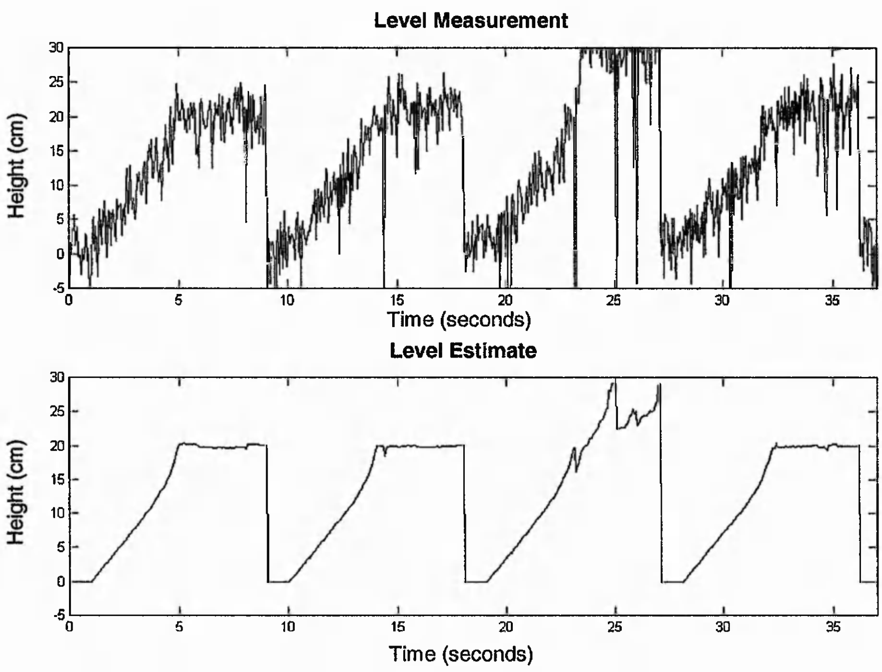


Figure 6.38 Level Measurement with a Standard Deviation of 5cm

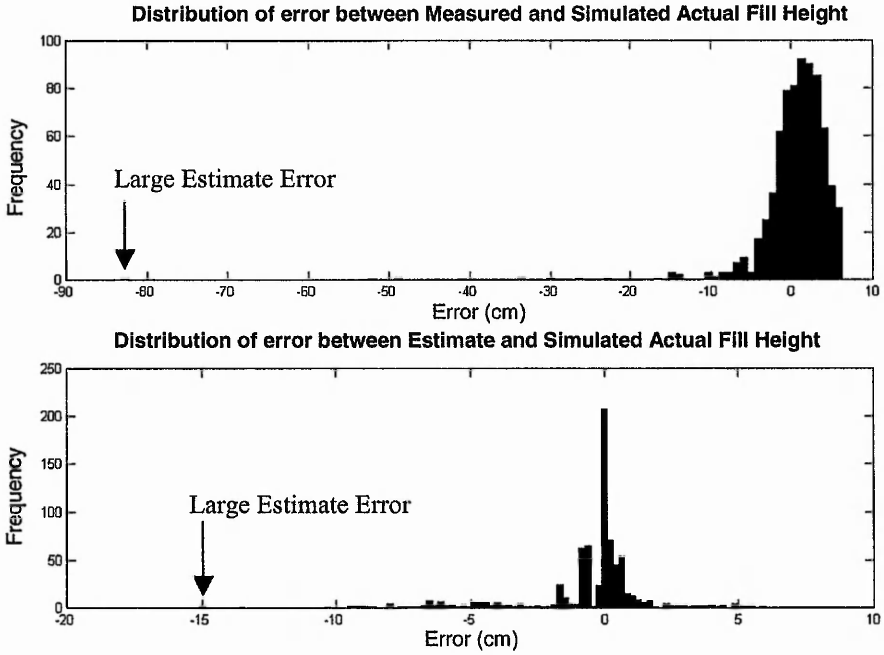


Figure 6.39 Error Distributions of Measurement and Estimates

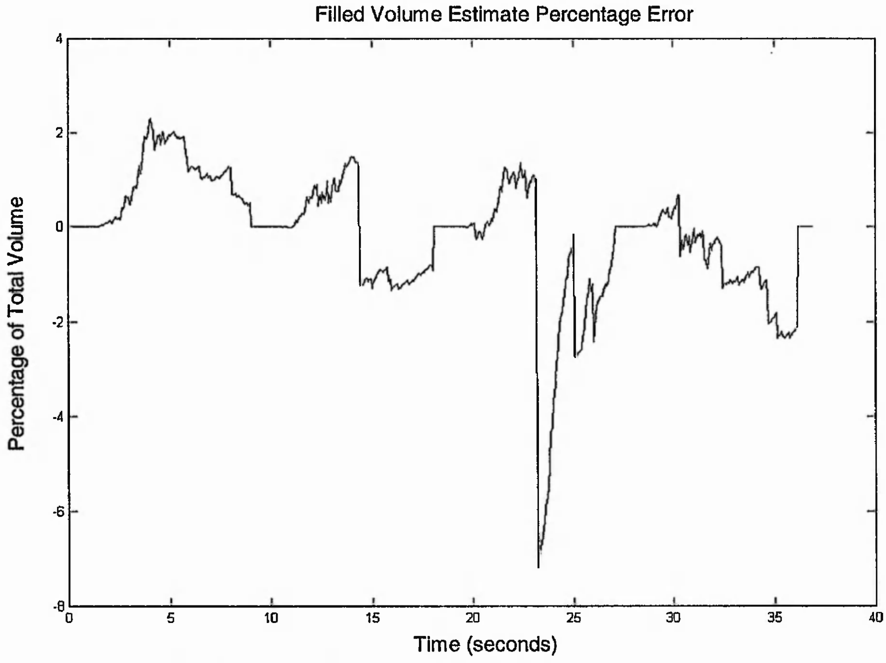


Figure 6.40 Volumetric Errors from Kalman Filter Estimates

6.4.6.3 Remarks

By plotting the original and estimate standard deviation it can be shown that the relationship appears approximately linear (Figure 6.41). This demonstrates that the amount of noise reduction from the Kalman filter is determined to a large extent on the setting of error covariance matrices *R* and *Q* (see Section 3.4) and that its performance under a set of predetermined conditions can in large be predicted. However, from the evidence shown in the examples demonstrated here it can be seen that the greatest threat to stability is occasional large disturbances as shown in figure 6.38.

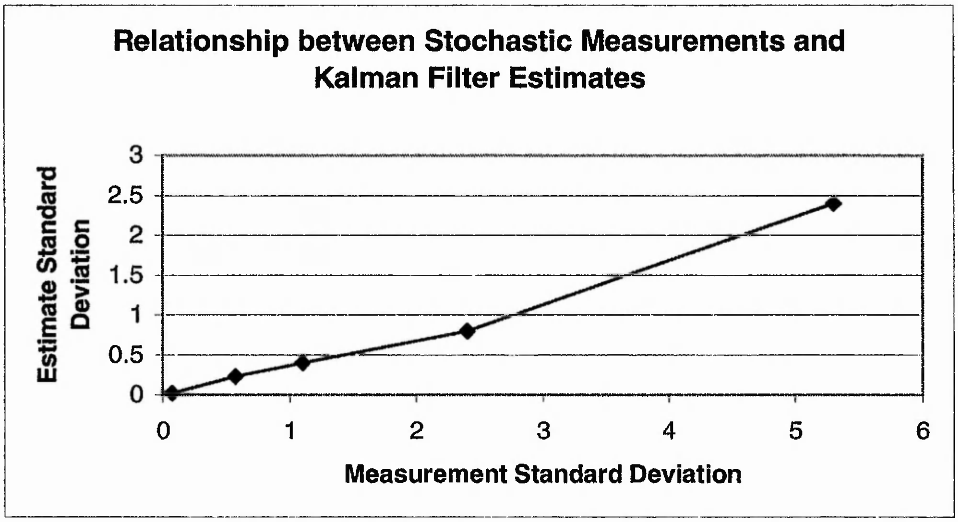


Figure 6.41 Kalman Filter Performance as shown by Error Distribution

A simple solution to this problem would be to include a pre-filter stage that rejects signals whose change is larger than is achievable from the physics of the plant, especially easy to determine if the reading appears to originate from a distance greater than the size of the bottle. A simple rejection algorithm may suffice or a more complex such as that proposed by Pena and Guttman (see Section 3.3), although further investigation would be required as the approach is primarily designed for Gaussian noise.

6.4.7 Valve Opening Characteristics

In the previous tests, the maximum flow rate was instantaneously achieved upon opening of the valve. This is not the characteristic that can be expected from an actual plant where valve opening and flow establishment will mean an increasing flow rate from zero to maximum flow rate.

6.4.7.1 Assumptions

In section 6.4.4 it was already shown that it is possible to model an opening characteristic by the choice of an initialisation value for covariance error, P. Here it will be shown how a range of opening rates can be modelled by intentional destabilising of the filter estimate by increasing the model noise, S. Hence, the internal model does not necessarily have to be truly representative of the process to obtain acceptably stable results.

Within the simulation the model of the bottle within the Kalman filter is identical to the process model, in the sense they are derived from the same source. To this extent the modelling error, S, should be set to zero. However, when an opening characteristic is added, it remains un-modelled within the filter, such that the value of S must be increased and thus reducing confidence in the model.

The valve opening rate is chosen by specifying a value for the percentage opening per second, where the percentage is represented by a decimal such that 1 is 100%, 0.5 is 50% etc. Thus, an input value of 1 means the valve opens 100% in 1 second, and a value of 1000 which is used as the default value in the model for the valve opens 100% in 0.001 seconds – effectively instantaneous. For the tests here, an opening factor of 1 is used to allow the demonstration of variation to the Kalman filter variable S. Once the valve is fully open, the maximum flow rate is $150cm^3/s$. The following values of S are tested; 0.001, 0.01 and 0.1 with P at 0. The measurement noise was set to an approximate standard deviation of 0.5cm with a gamma distribution and filter iteration time at 0.05 seconds.

6.4.7.2 Results

Figure 6.42 shows the fill height characteristic; the valve opening characteristic can be seen between seconds one and two as a slight curvature in the gradient of the fill height line.

/

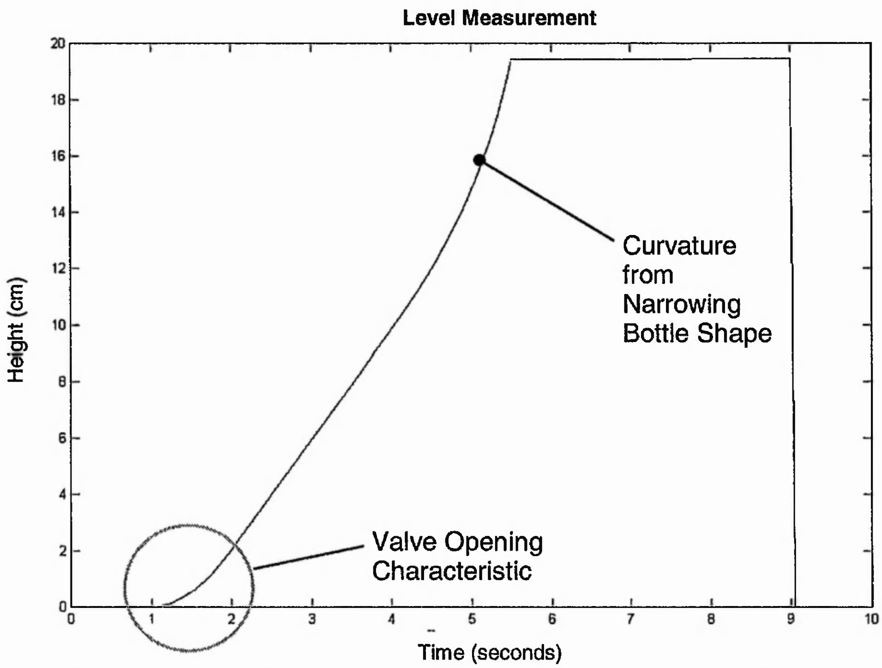


Figure 6.42 Simulated Fill Height from a Slow Opening Valve Reducing Initial Flow Rate

With a S value of 0.001, Figure 6.43 shows the significant and continual bias to overestimation of the height of the liquid. The immediate transient response is a result of valve opening characteristic, but this initial error is not corrected during the fill. The sawtooth pattern is attributed to the reopening of the valve as the estimate drops below the setpoint. The model expects a sudden increase in level from a flow rate equivalent to a fully open valve. However, the slow response of the valve means additional liquid reaches the bottle at a much slower rate than anticipated, thus an oscillatory saw wave is created as estimates move between model predictions with the valve closed and the measurements with the valve open.

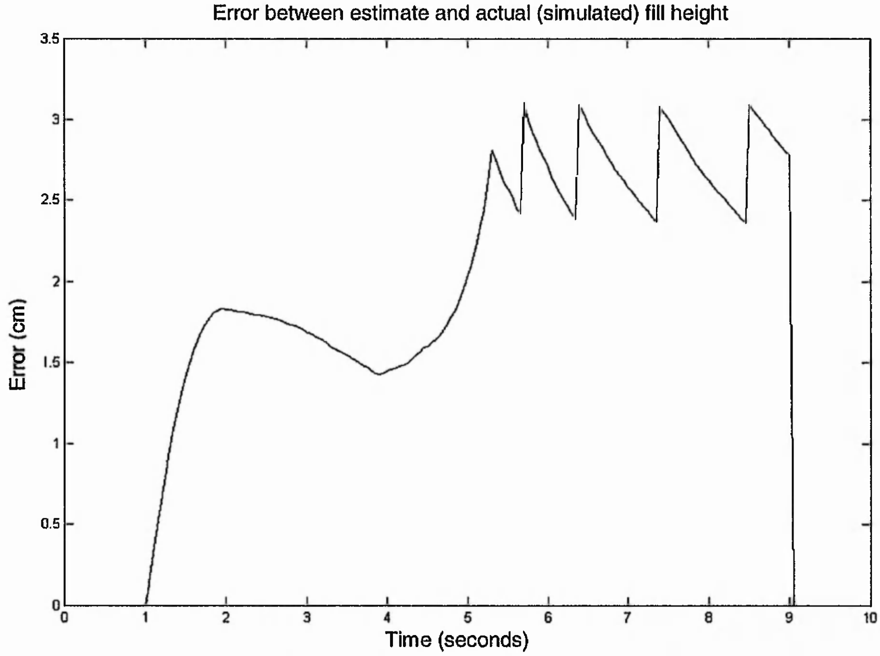


Figure 6.43 Estimation Error from Un-modelled Slow Valve Opening, where $S = 0.001$

Increasing the S value to 0.01 shows a distinct improvement (Figure 6.44). The transient error remains and a positive bias in the estimates is present throughout the cycle. However, the oscillatory behaviour has now been eliminated. It can be seen a small amount of noise is entering the signal after the 3rd second showing less reliance on the internal filter model at this time.

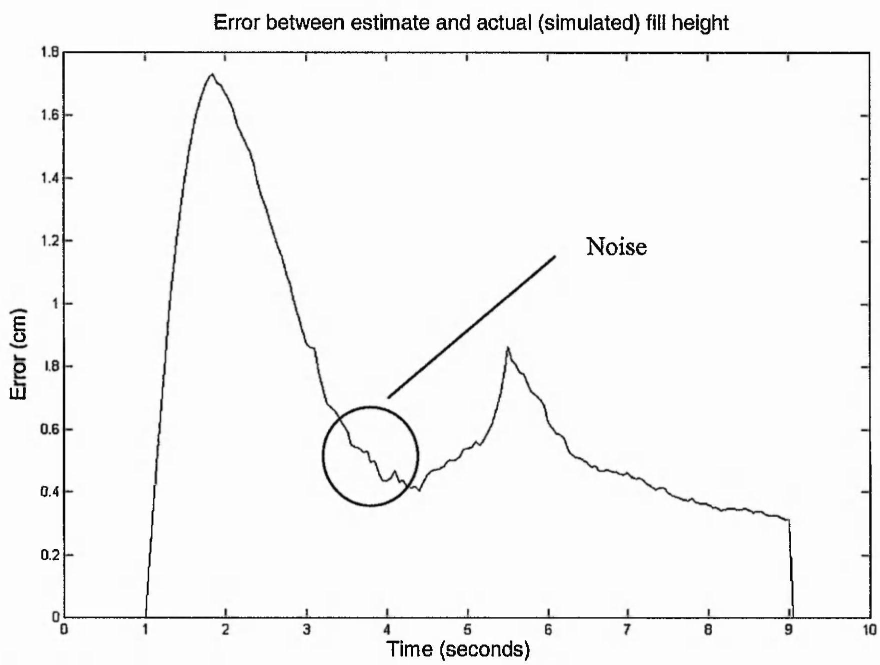


Figure 6.44 Estimation Error from Un-modelled Slow Valve Opening, where $S = 0.01$

Figure 6.45 shows the result of a S value of 0.1. A distinct improvement can be witnessed in the long term performance of the estimates. A significant transient disturbance remains, however the error is quickly corrected to the point where little bias remains, although larger levels of noise are present within the estimate.

With a satisfactory steady state performance it is possible to reduce the transient performance using the initialisation techniques described in section 6.4.4. By setting the measurement error covariance, P to 99999 the transient disturbance is reduced by more than half (Figure 6.46).

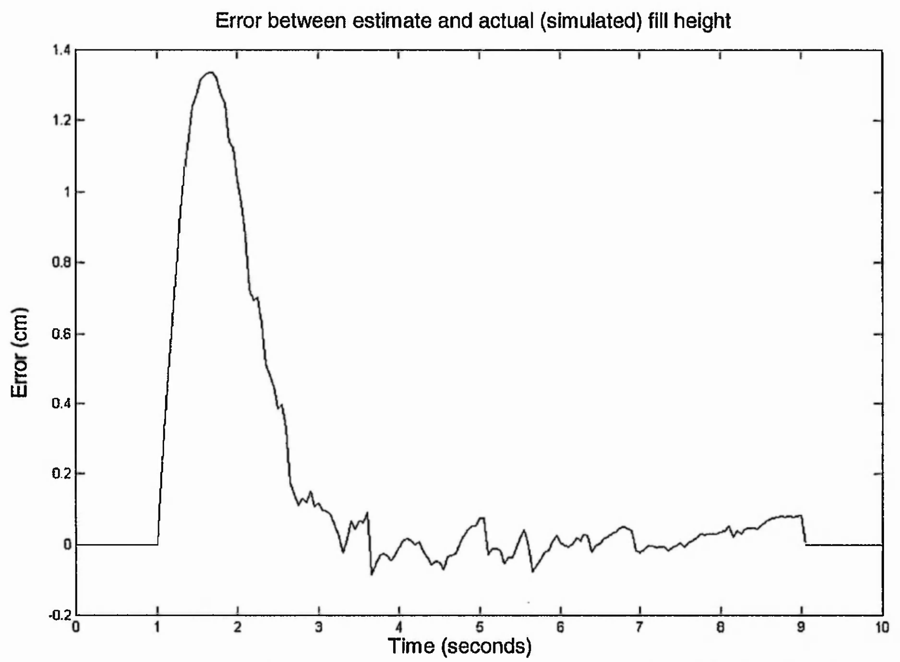


Figure 6.45 Estimation Error from Un-modelled Slow Valve Opening, where $S = 0.1$

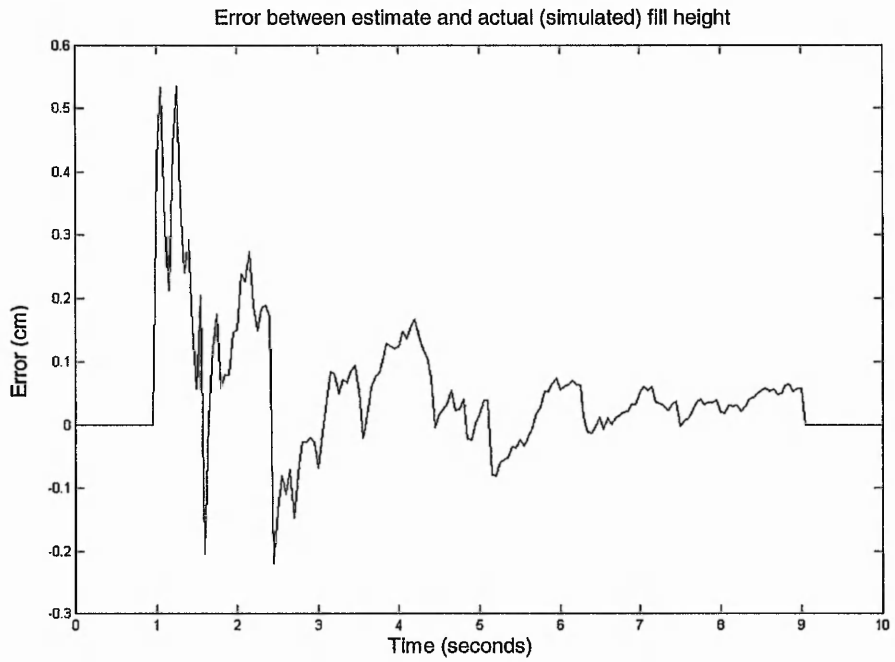


Figure 6.46 Estimation Error from Un-modelled Slow Valve Opening, $S = 0.1$, $P = 99999$

6.4.7.3 Remarks

It can be seen from the examples shown that although unmodelled non-linear characteristics will disturb the filter estimate, by selecting a larger value of S recovery characteristics can be significantly improved. In addition, by changing initialisation values of P, transient characteristics can be reduced. Unfortunately, both these techniques sacrifice smoothness of estimates for better parameter tracking. Nevertheless, these techniques represent an acceptable compromise within the measurement situation when steady state errors can be reduced from 3cm (Figure 6.43) to less than 0.1cm (Figure 6.46). For convenience, the filter uses a default S value of 0.1, which improves overall tracking performance.

6.4.8 Valve Closing Characteristics

In many filling operations, valve closure is very fast, such that it can be assumed that flow ceases instantaneously. However, for the proposed strategy to show completeness, it is necessary to demonstrate the effects of slow closure on height estimates. It will be shown that this is a significantly bigger problem than opening characteristics, to the extent that if closure was slower than modelled then the strategy has a severe limitation in the present configuration.

6.4.8.1 Assumptions

The same basic filter settings are used as before, with the valve opening rate set at 1000 and a S value of 0.1. A single valve closure rate of 1 (100% per second) is shown to demonstrate the characteristic.

The example shown is an over exaggeration of possible performance; it is used to demonstrate the behaviour of the Kalman filter rather than the valve.

6.4.8.2 Results

Figure 6.47 shows the actual and estimated fill height. The estimates track the fill height successfully until the set point is reached. Although the valve is given the signal to close after 4 seconds of filling (t = 5), liquid continues to fill the vessel to the point of unacceptable overfill – even overflow if this were an actual bottle.

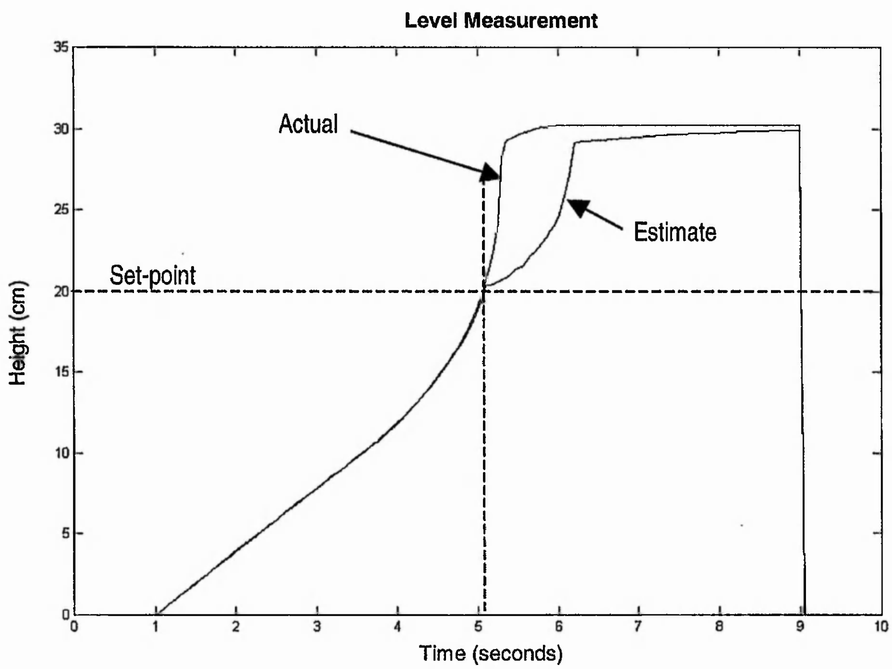


Figure 6.47 Estimation Error Generated from Un-modelled Valve Closing Characteristic

If the S value is increased to 100, then the estimate can be made to track the fill level more accurately (figure 6.48). However, the significant overfill remains, and in situations where noise was more prominent control would become even more impractical as the high value of S means the internal filter model is barely used. This indicates the control method is unsatisfactory in an example where slow closure characteristics of the valve are unmodelled and cannot be simply overcome by modifying filter parameters as shown for slow opening characteristics. Nevertheless, the application in hand is unlikely to encounter slow closure valves due to the spring-return design of standard filling valves. Furthermore, in situations where closure characteristics are significant in valve performance then they would have to be implicitly modelled.

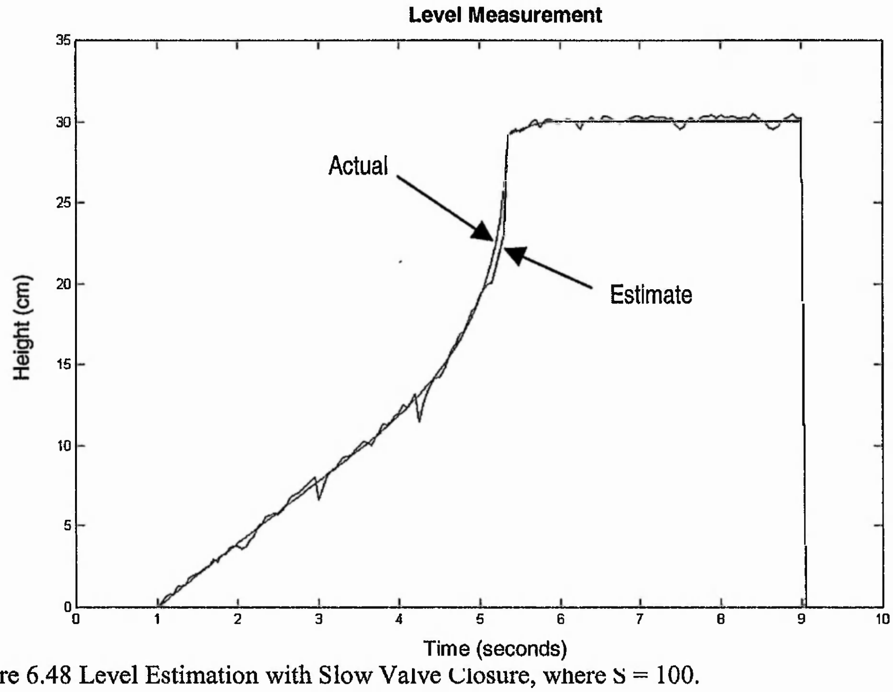


Figure 6.48 Level Estimation with Slow Valve Closure, where S = 100.

6.4.8.3 Remarks

The example demonstrates limitations of both the control method and Kalman filter. The controller uses a predictive algorithm using estimates from the filter (Section 5.4), therefore is heavily reliant on the filter's internal model. To this end, if significant valve closure characteristics remain un-modelled the control will fail in meeting the desired set-point.

The example shown is extreme and realistically closure rates can be expected to be appreciably faster. Nevertheless, it demonstrates that some factors have to be modelled, rather than relying solely on the Kalman filter characteristics.

6.5 Summary

In this chapter the measurement of liquid level in a single vessel has been shown. Experiments have been conducted on a custom built test rig which attempt to categorise the types of signals expected from ultrasonic measurements and an attempt to describe the noise is made. A gamma distribution is shown to provide a good approximation to the noise statistics.

A series of simulation tests incorporating different filling conditions have been conducted and it has been shown that some process characteristics can be compensated for without implicit modelling (e.g. valve opening), which allows for linearity to be maintained within the Kalman filter model. However, it has also been shown that under some conditions, like valve closure characteristics, the filter does not provide an adequate solution alone and that in some cases the controller must also be modified.

Nevertheless, with the degree of stability shown in the tracking of filling level, the knowledge gained thus far can be incorporated into a design for multiple controllers necessary for a full carousel filling control system. As will be seen, this will attempt to remove limitations witnessed in previous incarnations of the filling controller, such as lack of flow rate measurement, by using intelligent inference in the form of fuzzy logic and is described in depth in the next chapter.

Chapter 7

Monitoring of Multiple Valve Carousels

7.1 Introduction

The single valve approach, as described in the previous chapters, has shown potential benefits for the control of bottling plant. However, limitations exist within this approach. The most significant being the lack of flow measurement to improve the Kalman estimates in the presence of plant parameter variation.

Chapter 2 briefly discussed the possibility of using ultrasonic Doppler shift to deduce the flow rate of the filling process by measuring the speed of the rising surface. The shape of the bottle determines the relationship between flow rate and speed of the surface; where constant vessel radius creates a fixed relationship and a varying radius will continuously change the relationship. If knowledge of both the bottle shape and exact height of the current fill level were known, then the flow rate could be determined for all instances of the process. However, the exact height is unknown, only an approximation can be made with the Kalman filter, which cannot be used to provide input for flow estimates if the flow estimates are intended for the internal filter model due to the high risk of positive feedback. Nevertheless, within the bottle shape regions exist where the height does not affect the perceived flow rate, namely, in cylindrical sections where the radius is unchanging.

Many bottle and container designs contain a cylindrical section, as it is often the location of labelling, and this would be the best location for Doppler shift measurements. However, although the exact height is not required it is necessary to know whether filling is within the cylindrical section. This will require height information from the Kalman filter and intelligent processing eliminating the risk of positive feedback mechanisms.

Within a single valve controller, updates to the flow model could only be made during part of the filling cycle where liquid levels are within any cylindrical sections, thus limiting the accuracy. In actual filling plant, however, many valves are used in unison on a carousel, and because of this, at any stage of the filling process at least one bottle can be expected to be within the linear section (assuming this region is not uncommonly small). As many filling plants use a single header to feed all the valves, global changes in flow rate should be visible at all valves, therefore, if measurements could be combined intelligently then updates to the flow models could be made throughout the filling process.

With this in mind, a system has been developed using the techniques of fuzzy logic, as described in Chapter 4, to create a controller capable of parallel sampling Doppler shifts and intelligently inferring both individual and global flow rate changes. This information can then be used to feedback into the Kalman filters of the individual valve controllers, hence improving process control accuracy and long-term robustness of the strategy.

In this chapter, a Fuzzy Flow Estimator will be described in detail, and in the next chapter shown to perform within a complete simulation of a multiple valve bottling plant under the influence of flow disturbances. Prior to this, experimental data will be presented that confirms the feasibility of using Doppler shift as a means of determining flow rate.

7.2 Doppler Shift of Ultrasound

7.2.1 Background

The relationship between speed of a moving object and the subsequent frequency shift of waveforms reflecting off of it is given in Equation 7.1. The equation in this form can be used to find the theoretical frequency shift or be used to model the shift within the simulation (see section 5.5.1 for usage).

$$f_e = f_o + \frac{2Vf_o}{c} \tag{7.1}$$

where,

f_e = *Doppler shifted frequency*

f_o= original ultrasound frequency

c = speed of sound

V= speed of moving object

The physical equipment required to measure the Doppler shift of the ultrasonic echo is identical to that used in the previously described liquid level measurements (Section 6.2.2). The additional requirement is for processing of the waveforms to extract frequency information from pulsed or continuous signals. As already described in Section 2.10, Fast Fourier Transforms (FFT) can provide the post-processing analysis required to determine frequency shifts. However, it must be remembered that limitations in the accuracy (frequency resolution = sampling frequency/buffer size) and processing speed of software-based FFT means the application to real-time control would be impractical. Therefore, hardware-based frequency measurement would be required for actual implementation on a filling plant.

7.2.2 Feasibility Experiment

7.2.2.1 Aim

An experiment has been undertaken to test the potential of using the airborne Doppler shift of ultrasound as a means of determining flow rate. The aim of the experiment was to assess the ease with which this method could be used and highlight obvious problems. However, the refining of the process has not been tackled, as it is outside the remit of this particular project.

7.2.2.2 Method

Using the same flow rate settings as used for the experiments outlined in Section 6.2, a timing test revealed a height of 147mm was reached in an average of 12 seconds, which equates an average surface speed of 0.01229 m/s . This result corresponds closely with a value which can be obtained from the pressure sensor data (section 6.2.1.3), giving a speed of 0.01239 m/s . From Equation 7.1 it can be calculated that for an ultrasonic frequency of 156KHz, the Doppler shift can be expected to be an 11.2Hz increase in the received echo frequency. Measuring the average surface speed during emptying revealed an expected decrease in the received frequency of 11.9Hz. These values are small compared to the original (0.007%) and will test the limits of the FFT approach.

The hardware (as described in Section 6.2.2) has a range of fixed sampling rates. For the previous liquid level experiments this was set at the maximum of 6.25MHz. However, through preliminary investigation using FFT it was found that the listed specification of the card did not match the sampling rate achievable in practice. By using a fixed frequency reference signal fed through the sensors and to the ultrasonic A/D card, FFT revealed the true sampling frequency to be 5.86MHz. However, the lower selectable sampling rates of 3.125MHz and 1.56MHz were found to be accurate.

To maximise FFT accuracy it is necessary to use the lowest possible sampling frequency with the longest possible sampling buffer. As a compromise between FFT accuracy and sampling speed, a frequency of 1.56MHz was selected.

Although the frequency shift can theoretically be found from either pulsed or continuous signals, in order to maximise the obtainable accuracy within this feasibility experiment, continuous signals have been used. This allows the length of the sample buffer to be artificially lengthened by 'cloning' the individual data files and joining them together to form a new longer single file. It is necessary to carefully combine the waveforms between the files at zero crossing points, as discontinuities or mismatches reduce accuracy back to that which is achievable from the original single file. As sampling rate is 1.56Mhz, any small errors of one or two sample points in the 'cloning' process should produce noise in frequencies much higher than the 156KHz of the measured signal and thus the method will achieve the aim of improving the sensitivity of the FFT analysis. A sample buffer length of 16k (16384 samples) was used as the maximum practically achievable from the A/D card.

An approaching surface (filling) and a receding surface (emptying) was used to measure Doppler shift and 40 ultrasound measurements were made for each direction (the maximum achievable within the sampling rate performance of the equipment used, which included PC software for some signal processing). As noise is such a large factor in level measurement, it can be expected to interfere with the Doppler measurements taken here too. Of greatest concern is surface agitation, which contributes to continuously changing surface speed making underlying speed difficult to pinpoint. However, by looking at the mean speed of all 40 measurements for each direction should reveal an overall average value, or a close indication thereof.

7.2.2.3 Results

The results are categorised in to five sets of data (Tables 7.1-7.5), each one representing double the effective sample length of the previous and hence, double the resolution. The first set is for the raw file length of 16384 samples. Therefore, the base resolution is ~95Hz, which although too large to detect the Doppler shift from a single point, the level of noise in the measured signal ensures a distribution whose mean should approach that of the true value. The last set has an effective resolution of 6Hz.

For each set the unshifted frequency value was measured using the same mean sampling method used to determine the Doppler shift, thus maintaining consistency.

Table 7.1 Doppler Shift of Ultrasound with 95Hz FFT Resolution

Set1 (95Hz)	Mean Frequency (Hz)	Difference
Unshifted	155980	N/A
Filling	155988	**8**
Emptying	155974	**-6**

Table 7.2 Doppler Shift of Ultrasound with 48Hz FFT Resolution

Set2 (48Hz)	Mean Frequency (Hz)	Difference
Unshifted	155977	N/A
Filling	155991	**14**
Emptying	155962	**-15**

Table 7.3 Doppler Shift of Ultrasound with 24Hz FFT Resolution

Set3 (24Hz)	Mean Frequency (Hz)	Difference
Unshifted	155979	N/A
Filling	155993	**14**
Emptying	155964	**-15**

Table 7.4 Doppler Shift of Ultrasound with 12Hz FFT Resolution

Set4 (12Hz)	Mean Frequency (Hz)	Difference
Unshifted	155978	N/A
Filling	155991	**13**
Emptying	155964	**-14**

Table 7.5 Doppler Shift of Ultrasound with 6Hz FFT Resolution

Set5 (6Hz)	Mean Frequency (Hz)	Difference
Unshifted	155979	N/A
Filling	155990	**11**
Emptying	155968	**-11**

It can be seen in all cases that approaching and receding liquid gave the appropriate direction of frequency shift and of the expected order of magnitude (theoretical value ~±12Hz). It was intended to increase the resolution further, but the computer programs used in the analysis continually failed when processing the next stage, where the effective sample length was 512k. The process time for the last set of data (Table 7.5) was three days on a Pentium® 166MHz PC, which shows the impracticalities of using this analysis approach for real-time applications.

7.2.2.4 Remarks

Within the limitations of the FFT approach these results have shown that Doppler shift of air transmission ultrasound can detect the movement and speed of a liquid surface. However, it is also evident that to obtain the required accuracy for flow rate tracking dedicated hardware would be required. The aim of this experiment was to demonstrate the movement of the surface only and therefore, no attempt was made to resolve flow variations. Nevertheless, the results demonstrate the opportunity available for further study and provide the foundation for the development of a multiple valve approach to be discussed within this chapter.

7.3 Multiple Valve Carousels

As briefly mentioned in Chapter 1, filling plant for most products are based around a carousel of valves to maximise process efficiency. The size of the carousel can vary between plant, but a common feature on which they are all based, is that the individual valves are independent of each other. This allows very effective scaling up or down of the plant without significant changes to the filling characteristics. Although the valves themselves are separate from each other, the liquid comes from a single header tank for all valves. This means, all other things being equal, the flow rate should be identical from all valves, or at the very least changes in the header pressure should be detectable from any open valve.

The carousel layout effectively means that at any given instant the bottle filling process is represented in its entirety by the states of filling of individual valves. That is, from initial valve opening to final shut off, snapshots of the whole process are available from carousel without necessarily tracking a complete fill of a single bottle. If these snapshots could be combined then process control speed would be significantly improved by reducing the time individual valve controllers react to changes in plant parameters. Furthermore, by comparing measurements from the singular to those taken from the many, possibilities arise allowing the differentiation of local effects such as sticking or worn valves, from global effects such as pressure loss in the header tank. However, within the limitations of the discussed measurement noise a degree of intelligence is required to maximise these benefits.

7.4 Expanding the Simulation

The simulation for a single valve has already been shown and discussed in the previous chapter. The results showed a great deal of promise in measurement of an individual fill height. However, without a control for a multiple valve carousel the research would have little value to the bottling industry. To this end, it was necessary to expand the simulation such that it could contain many valves arranged in a virtual carousel, thus allowing exploration of the possibilities highlighted in the previous section. Namely, can input information from a number of valves be combined in a robust strategy for improving both the performance of the individual valves and the plant as a whole.

Following the modular approach taken throughout the software development, the simulation was multiplexed by adding matrices for many of the variables, whilst keeping the same algorithms and functions. For example, the single variable h becomes an array $h[n]$, where n can be any integer between 1 and the number of valves modelled. The maximum number of simulated valves was set at sixty-four, with any number below that, selectable from the input text file. By no means does sixty-four valves represent a theoretical limit to the modelling – any number could be used. However, limitations arise in attempts to display the results from so many individual valves. As will be seen, for the most part only six valves will be used to demonstrate the capabilities of the proposed strategy. An advantage of the strategy, as will be discussed later, is that it becomes more effective the greater number of valves used, hence using only six valves will be almost a worst case scenario.

Whatever number of valves are chosen it is necessary to determine the timing of the valves within the carousel, which will be governed by the time it takes to fill one bottle and the interval between successive fills on the same valve. This was achieved by a simple algorithm (Eq. 7.2), which offsets the timing of individual valves against the master timing clock. Such that initially only the first valve starts at zero time, the others have successively negative values, which translates into the time it will take them to reach the first fill.

$$tfill[n]=-((individual\_fill\_time+fill\_interval)\cdot(n-1))/number\_of\_valves \quad (7.2)$$

where,

tfill is the individual valve timing offset

n is the valve number (1 to total of valves)

For six valves the following data set can be created.

Table 7.6 Valve Timing Offsets for a Carousel Filling Arrangement

Valve number (n)	**tfill [fill time = 5] [interval = 1]**	**tfill [fill time = 6] [interval = 1]**
1	0	0
2	-1	-1.1667
3	-2	-2.3333
4	-3	-3.5
5	-4	-4.6667
6	-5	-5.8333

Essentially, the speed of the carousel is set by sum of the fill time and the fill interval, which equates to a single revolution. Figure 7.1 shows how these figures transfer into the spatial relationship of valves and the carousel.

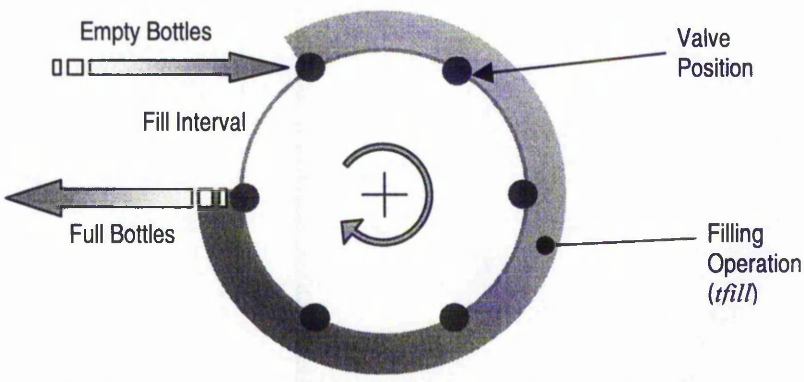


Figure 7.1 Filling Carousel (with 5 seconds fill and 1 second interval)

All the features of the single bottle simulation are present within the multiple bottle model, therefore not only is the new overall control strategy explored but also its integration into the Kalman filter and individual valve controller. The source code for the plant simulation can be found in Appendix A.

7.5 Fuzzy Flow Estimator

It was shown on the previous chapter that although good estimation of fill height can be achieved with a Kalman filter the presence of flow variations was not directly modelled. In circumstances such as slowly opening valves the filter can be made to overcome discrepancies between measurements and the internal filter model by modifying filter parameters. However, the internal model was based on a fixed flow rate and thus would be vulnerable to other variations in flow rate during the filling process, such as changes in pressure at the header tank. As already described at the beginning of this chapter, possibilities exist within the multiple valve carousel for gathering information about the flow rate otherwise unavailable in the single valve scenario.

7.5.1 Problem Definition

A need exists to improve control by incorporating flow measurement or estimates from a number of valves within a filling carousel. However, to calculate flow rate purely from Doppler measurements requires knowing the fill height for shapes other than cylinders. Therefore, there is no independence between flow rate and fill height, and instability could arise from using flow rate derived from the Kalman filter's own estimates of height, leading to a high risk of positive feedback. It is therefore necessary to find independence.

Many bottle shapes contain an area that can be described by a cylinder. Within this cylinder, speed of the rising liquid can be directly correlated to the flow rate as the diameter of the bottle is unchanging. A means is therefore required to differentiate when Doppler measurements can be robustly taken (i.e when fill level is within the cylindrical region) and then incorporate all the available data into useful information for the individual valves.

Fuzzy Logic techniques would be well suited to the role of inference within this application for three main reasons.

- MIMO systems can be handled with ease
- Outputs from Doppler measurements are likely show instability; therefore robust inference is required
- Combining many estimates for a single measure of plant performance has already been shown to good effect in other applications (Jeffries et al, 1997).

7.5.2 Specifications

For a strategy aimed at solving the flow rate estimation problem a number of performance criteria represent a preferred range of attributes:

- Provide feedback to individual valve controllers for latest overall pressure.
- Be able to differentiate between flow variations of individual valves and overall pressure, thus not sacrificing local performance for a global average.
- Allow variations in flow to be used by controllers unable to perform own measurements because factors such as initial turbulence of the surface or the shape of the bottle.
- Provide condition monitoring information to the operators, therefore allowing parameters outside direct control to be adjusted to the current performance of the plant.
- Allow support for failed Ultrasound sensors by re-construction from other measurements.

The last point can be accommodated in combination with the Kalman filter by setting the filter parameters, R, P and S, such that the internal model alone is used. Although less than ideal, feedback from other parts of the system should maintain adequate control as a temporary measure until maintenance can be carried out. Moreover, the original mechanical control is likely to remain as a failsafe system, so performance can be expected to meet current standards even in the presence of failed sensors.

7.5.3 Estimator Overview

It is reasonable to expect that the flow rate, Q will not be identical in all valves and to use flow rate directly to establish an overall flow variations would lead to an over-simplification of plant performance. However, the rate of change of flow can be reasonably expected to vary similarly over all valves from a global disturbance, independent to a large degree on the actual flow rate of individual valves. Therefore, if the fuzzy estimator is to be able to cope with a wide variety of conditions, such as valve sticking or residue build-up, it is necessary to remove actual flow rate from inference calculations; thus, rate of change of flow will be used as the primary parameter. Within the computer algorithms, the rate of change of flow, $\dot{Q}$ is never used directly, instead the change in flow, ΔQ is the preferred parameter. As $\dot{Q}$ is used only to update flow rate and not as separate variable the following equation, Eq 7.4, can be reduced for efficiency to Eq 7.5.

$$Q_{k+1} = Q_k + \dot{Q}_k \cdot t \tag{7.4}$$

where

k	is the current time step
$k+1$	is the next time step
t	is the update interval

but,

$$\dot{Q}_k = \frac{dQ_k}{dt}$$

which, for a discrete system, can be expressed as,

$$\dot{Q}_k = \frac{Q_k - Q_{k-1}}{t_k - t_{k-1}}$$

and because,

$$t = t_k - t_{k-1} \quad \text{and} \quad \Delta Q_k = Q_k - Q_{k-1}$$

Eq. 7.4 can be rewritten as,

$$Q_{k+1} = Q_k + \Delta Q_k \tag{7.5}$$

Therefore the need to differentiate with respect to time is redundant. This provides an efficient means of processing within the computer but does remove the value of $\dot{Q}$ existing in any real sense within the algorithms, despite this being the parameter that is described as the best means of combining flow information from a number of valves. However, it exists within the invisible heart of the calculations undertaken.

Two 'modules' of fuzzy logic are required; one to determine the overall flow rate, the other to balance direct Doppler measurements taken at the individual valves and overall estimates from other valves. The first will be called Flow Consolidation, the second Flow Measure. It will be shown that the proposed arrangement is capable of simultaneously tracking both local and global changes in flow rate in a robust manner suitable for filling plant control.

7.5.4 Fuzzy Input Set Definition

Within the bottle shape, an area of cylindrical proportions is required to give independence from the fill height estimate, if the flow rate is to be used for improving the model within the Kalman filter. However, it is still necessary to know when this cylindrical region has been reached.

For a simple wine bottle style, three areas can be defined (Figure 7.2). First, the region at the base of the bottle where turbulence and surface agitation are sufficient to prevent a true surface forming and hence, Doppler measurements of surface speed are unreliable. Second, the main body the bottle, where the diameter is unchanging and measurements are the most reliable. Third, the neck of bottle where the decreasing diameter prevents flow rate measurements as the level accelerates. These three areas can be used as templates for fuzzy sets.

The crossover region between the sets accommodates the uncertainty in level from the height estimates, and as height is not used directly no positive feedback can be created. Membership to the BODY set defines the confidence, μ of flow measurements, where, only a central region provides ideal conditions and hence 100% confidence in Doppler readings. On either side of this central region confidence diminishes linearly into the NECK or BASE regions.

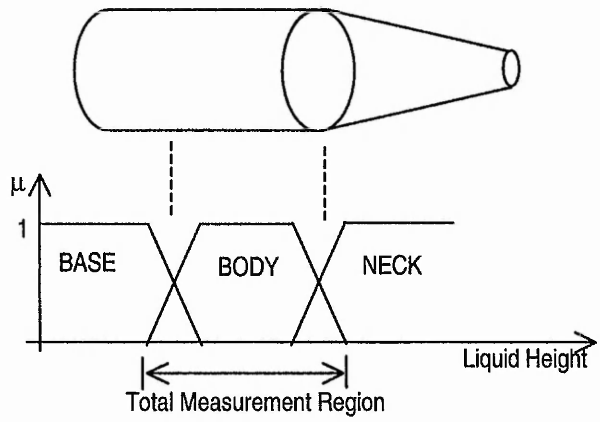


Figure 7.2 Relationship of Fuzzy Input Sets to Bottle Shape

7.5.5 Flow Consolidator Module

Flow Consolidation combines rate of change of flow measurements from all valves by using the membership of the BODY set to give a confidence to each measurement. Those flow rates deduced from Doppler measurements made outside the BODY region will be ignored, and those taken in the centre of the region will be given prime consideration.

The action of the module can be characterised by the following rule:

If Fill_Height is BODY then Measurement_Confidence is HIGH

This rule is repeated for all valves at every iteration producing a matrix of inference results.

Figure 7.3 gives a graphical representation of this module's operation

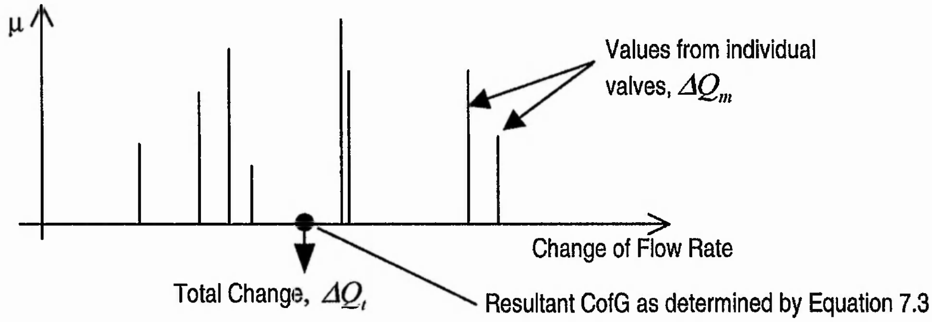


Figure 7.3 Fuzzy Consolidator Graphical Representation

The lines on the diagram are effectively the fuzzy output sets. The position on the x-axis is the change in flow as measured at the individual bottles. The height of the line is the confidence with which the measurement was made, i.e. membership to the fuzzy set BODY. Two properties differentiate these output sets to those often used in fuzzy systems. Firstly, the shape, or lack thereof, means the truncation value is linearly and directly related to the weight used in defuzzifying the consequent set space, thus satisfying the concerns presented in Section 4.4.2. Secondly, the location of the set changes depending on the value measured. These properties mean that using the modified centroid equation (Eq 4.2 – section 4.4.2), an overall flow rate change can be found in a predictable manner Eq. 7.3.

$$\Delta Q_t = \frac{\sum \mu_i \Delta Q_m}{\sum \mu_i} \tag{7.3}$$

where,

ΔQ_t is the total change in flow rate

ΔQ_m is the individual measured change in flow rate

μ_i is the membership or confidence of the individual values

The value ΔQ_t will be used in two ways. Firstly, to feedback into the Flow Measure module (discussed next) and secondly, to provide condition monitoring information on plant performance.

7.5.6 Flow Measure Module

Flow Measure uses the membership to BASE or NECK to weight global estimates of flow rate change against direct measurements with weighting from the BODY region. Outside the BODY region, global estimates are exclusively used to update flow, within the BODY region only direct measurements are used. This can be summarised in rule form:

If Fill_Height is BODY then Flow_Measurement is DIRECT

If Fill_Height is NECK or Fill_Height is BASE then Flow_Measurement is ESTIMATE

These two rules are applied to each valve in turn after Flow Consolidation has been performed. The final output from the Flow Measure section can be used to update the flow rate inside the Kalman filter.

Figure 7.4 demonstrates the process graphically.

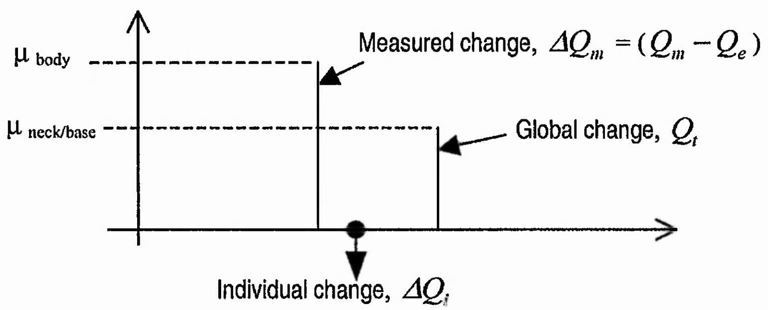


Figure 7.4 Fuzzy Measure Graphical Representation

The modified centroid equation (Eq 4.2) is used again here to find the value, ΔQ_i, which is used to update the value of flow rate held within the Kalman filter model, Q_e. The measured value for the flow rate change, ΔQ_m, is found by finding the differences between the current measured flow rate and the estimate held with the Kalman filter from the previous iteration.

The value of Q_e can be used to output to condition monitoring elements indicating the performance of the individual valves.

7.5.7 Fuzzy Flow Estimator Schematic

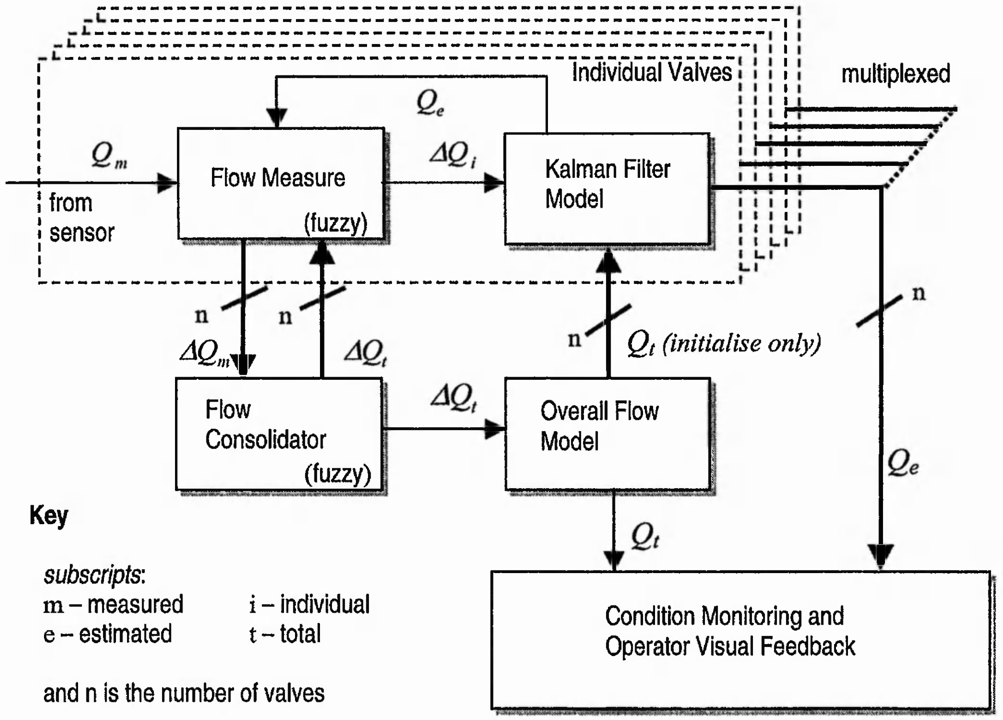


Figure 7.5 Schematic of Fuzzy Flow Estimator showing interaction of fuzzy modules with Kalman Filter and external components such as Condition Monitoring

The diagram (Figure 7.5) shows the two main fuzzy components and their interaction with the Kalman filter and condition monitoring systems (which are discussed briefly in Section 9.3.2). The number of Flow Measure modules is equal to the number of valves in the system, i.e. one per Kalman filter and valve controller. There is only one Flow Consolidator module, as it combines the output from individual valves. In addition, output from this module is multiplexed back to Flow Measure modules.

The overall flow model contains the current maximum flow rate or pressure available, which is used to initialise Kalman Filter models and also prepare the variable for display or further monitoring.

The complete operation of Fuzzy Flow Estimator can be summarised as follows: The change in flow rate is obtained from all valves in the carousel and 'consolidated' to form an estimate of the global change in flow rate. This is used to update an overall model for flow rate and more importantly to compare to an individual measurement in the Flow Measure module. The importance of the global change to the individual valve is assessed against the validity of local flow measurements by using the fuzzy inference based on the sets described in Figure 7.2. The flow rate is then updated in the internal model of the Kalman filter accordingly and hence, hopefully improving fill height estimation.

The role of the Flow Consolidator Module is to extract global trends from the readings of all valves on the carousel. On the other hand, the Flow Measure Module attempts to differentiate local changes to those seen globally. However, it is only able to do this when Doppler shift measurements are taken from within the reliable cylindrical section of the bottle; therefore, limitations exist within this proposed strategy. Nevertheless, as will be shown in the next Chapter, the Fuzzy Flow Estimator is capable of managing a number of different flow rate scenarios, which would have otherwise led to poor estimation of the fill height without updates to the flow rate models within the individual Kalman filters. Furthermore, it will be shown that the Estimator provides stability and robustness even when encountering localised changes in flow rate that are initially undetected because they occur outside the ideal measurement region. Thus, the Fuzzy Flow Estimator will be shown to be able to cope with conditions not explicitly defined within its algorithms, but like the Kalman filter, has an implicit stability which is well suited to role requested of it here.

7.6 Summary

This chapter aimed to outline a method by which the technologies developed in Chapter 5 for the measurement or control of a single valve can be fully exploited on the full-scale plant present within the bottling industry. It was proposed that Doppler shift of the ultrasound signal (already exploited for time-of-flight liquid height measurement) could be used to determine the flow rate, which is an unmeasured parameter in the single valve case. A simple experiment was conducted, which found a Doppler shift could be detected, but also that further work would be required to develop a system for practical real-time use.

Nevertheless, the frequency shift was a real, measurable phenomenon that could be used as a basis for further development of monitoring and control strategy.

A Fuzzy Flow Estimator was developed from the fuzzy logic modifications proposed and discussed in Section 4.4. It seeks to track flow rates over many filling valves whilst maintaining high levels of stability – an inherent and useful feature of Fuzzy Logic systems. The Estimator uses changes in flow rate to update internal model within the already demonstrated Kalman Filter, and hence improve liquid level measurement accuracy. The performance of the proposed system for Multiple Valve Carousels is demonstrated in the next chapter.

8.1 Introduction

The single valve scenario and measurement problems associated with using ultrasound were dealt with previously in Chapter 6, this chapter aims to show the performance of the Fuzzy Flow Estimation system within a multiple valve situation. A range of flow conditions from both global and local disturbances will be shown and the results analysed.

The noise model used is the Gamma distribution set at a level where σ is ~0.5 as measured in Section 6.2.2.3, except for the first example case where the noise level is increased for clarity. The maximum fill time for an individual bottle will be capped at 5 seconds and a fill interval of 1 second will be used. Hence, carousel timing is 6 seconds per revolution.

The fuzzy sets for describing the measuring regions (Section 7.5.4) of the bottle shape described in Section 7.4.5, are defined as shown in Figure 8.1.

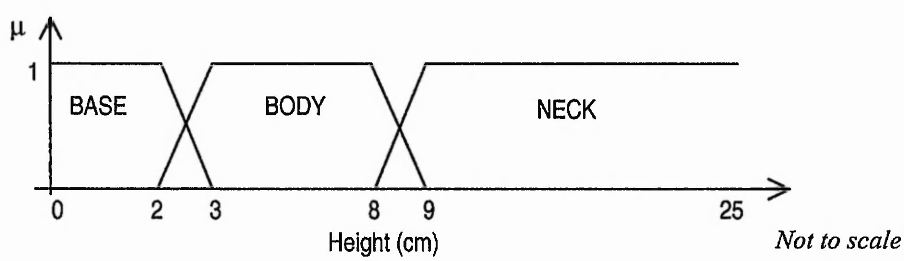


Figure 8.1 Fuzzy Input Sets for Multiple Valve Carousel Simulation

8.2 Fixed Flow Model

Before the performance of the Fuzzy Flow Estimator is shown it is necessary to demonstrate the basic operation of the multiple valve simulation. As already described in Section 7.4, the timing of the individual valves is offset comparable to its position on the carousel, this can be demonstrated by looking at the flow rates of six valves over a 30 second period, as shown in Figure 8.2. For clarity, the maximum flow rate in this example is assumed to be instantaneously achieved (i.e. valve opens 100% in 0.001 seconds).

Figure 8.3 shows the fill level on the six valves, two aspects of performance should be noted. Firstly, the height measurement noise (gamma distribution, $\sigma = 1$) shows that each valve is a unique entity within the simulation and not merely a transposition of a single model. Secondly, the measurements from individual valves can be subject occasional large 'drop-outs' caused here by the extended tail of the gamma distribution and can be associated to the physical mechanisms of ultrasound scattering and signal amplitude variations as discussed previously in Section 6.2.2.

As the flow rate is unaltered throughout the thirty second run the Fuzzy Flow Estimator has no influence on Kalman filter performance at this stage, although it is present transparently linking flow rate parameters within the system. The processing of this six-valve simulation takes approximately ten seconds on a PentiumII® 233MHz.

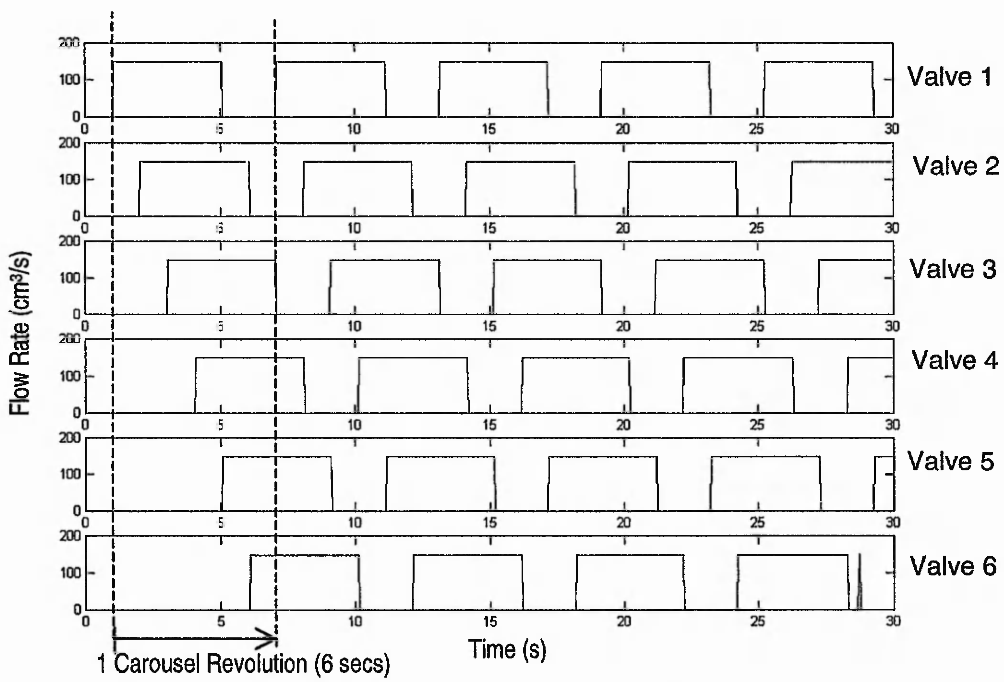


Figure 8.2 Simulated Flow Rates of Six Valves arranged in a Carousel

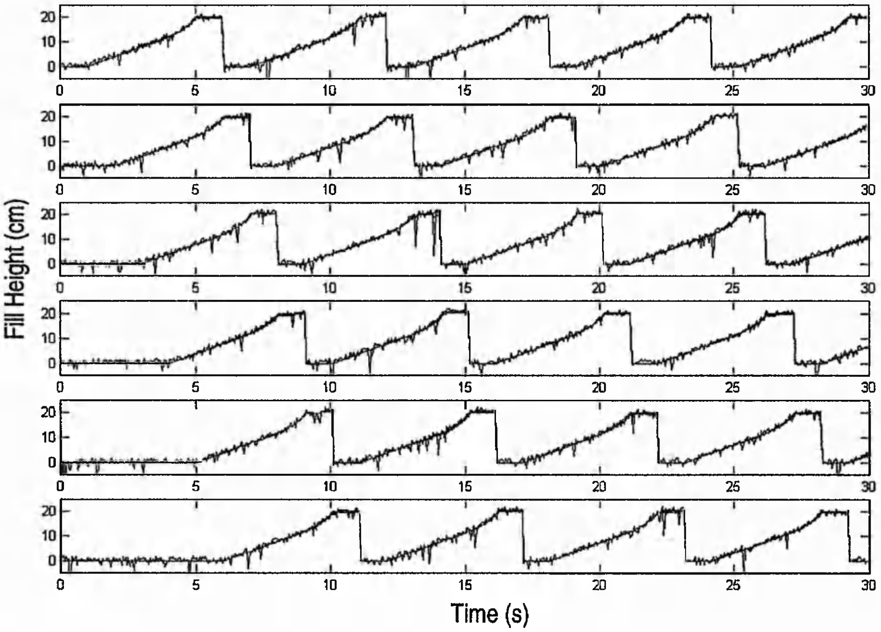


Figure 8.3 Simulated Fill Level (with added Gamma Noise) of Six Valves arranged in a Carousel

8.3 Global Disturbances

A disturbance that can be described as global is an effect that is visible from all measurement points. In this bottling filling application, the origin of such a disturbance to the flow rate can be assumed to be the single header tank, which is responsible for providing pressure for all valves. The reason can be attributed to either changes in demand (from the valves) or changes in the feed (from product mixing tanks, carbonation or pasteurisation stages).

The rate of change of these variations can be expected to be reasonably slow (compared to local disturbances) because of high system inertia. Such that, either a sinusoidal-like variation about some pre-defined plant set-point or long-term drift can be expected; although, it should be noted that these are approximations to possible problems rather than a result of direct measurement. The intention of the following simulation runs is to highlight the abilities of the Fuzzy Flow Estimator to manage flow variations under a range of conditions, as opposed to revealing the performance characteristics of any given filling plant specifically. The aim of the strategy is to provide flexibility and robustness, such that any subsequently discovered features of plant operations can be incorporated with confidence.

8.3.1 Sinusoidal Variation

A sinusoidal variation was introduced to the global flow rate with a peak amplitude of 10 cm^3/s (6.7% of maximum flow rate). The frequency is demonstrated at three levels, ~2 cycles per fill, ~1 cycle per fill and ~1/2 cycle per fill, and as before six valves are modelled. Figure 8.4 shows the effect of variation over the six modelled valves, the disturbance can be seen as gentle wave occurring at the maximum flow rate – compare with the flat profile in the fixed flow model in Figure 8.2. It can be seen that peaks in flow rate occur at the same moment over all valves and hence represents a global disturbance. A number of spikes can be seen towards the end of a number of the fill cycles; this is due to the valve reopening to meet the set point after reassessment of the fill level during steady-state measurement of the surface after the valve closes initially.

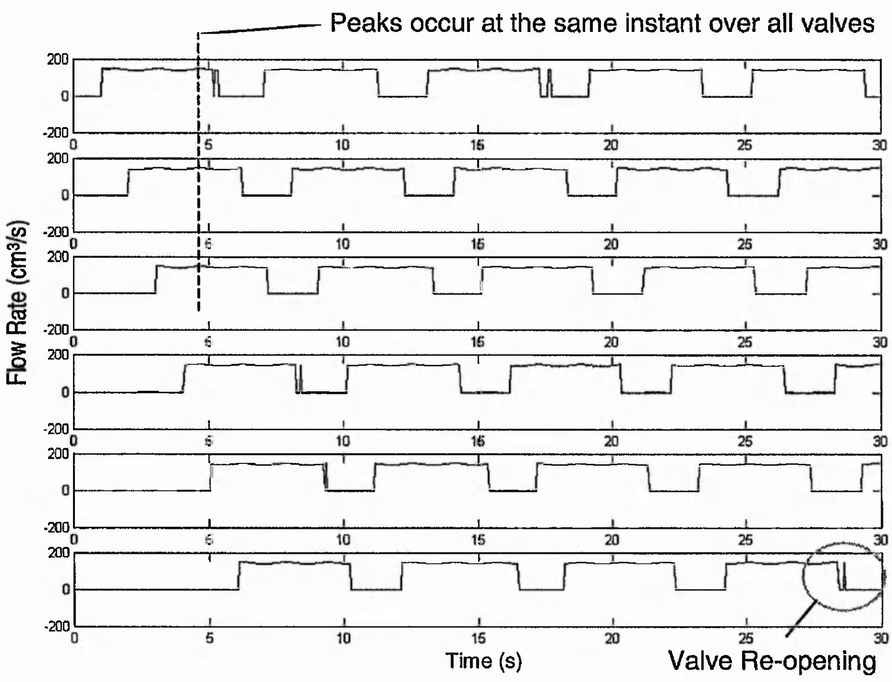


Figure 8.4 Result of Global Sinusoidal Variation to Flow Rate in a Valve Carousel

Taking the difference between the Fuzzy Flow estimation and the actual flow rate entering the bottle, (which is the ideal flow rate the simulation generates within the plant model in response to the operating conditions), a plot can be generated of the estimation 'error'. It can be seen in Figure 8.5 that flow rate is tracked with zero error except for two circumstances. Firstly, upon initialisation of the estimator a complete cycle of the carousel must be undertaken before all valves contribute to the measurement of flow. Hence, the sinusoidal variation can be witnessed where the model assumes static flow rate.

Secondly, between successive fills, flow rate for a given valve is zero, (as the valve is closed), but the estimator assumes continual variation. However, over the actual filling cycle the estimation error is zero.

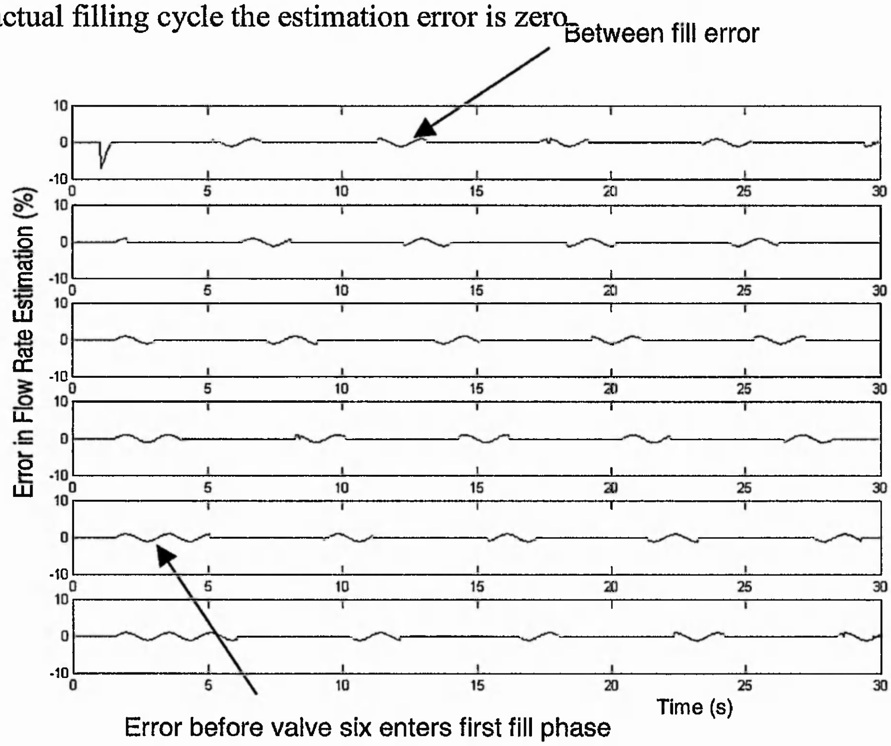


Figure 8.5 Flow Rate Estimation Errors (2 cycles per fill sinusoidal variation)

Figures 8.6 and 8.7 show the errors from ~1 cycle per fill and ~1/2 cycle per fill examples. These also show that during filling flow rate errors are effectively tracked. Noticeable on both the examples is a larger step error at the very beginning, which can be attributed to the fact that no flow rates are measured until the first bottle is approximately one-quarter full, i.e. in the linear filling region.

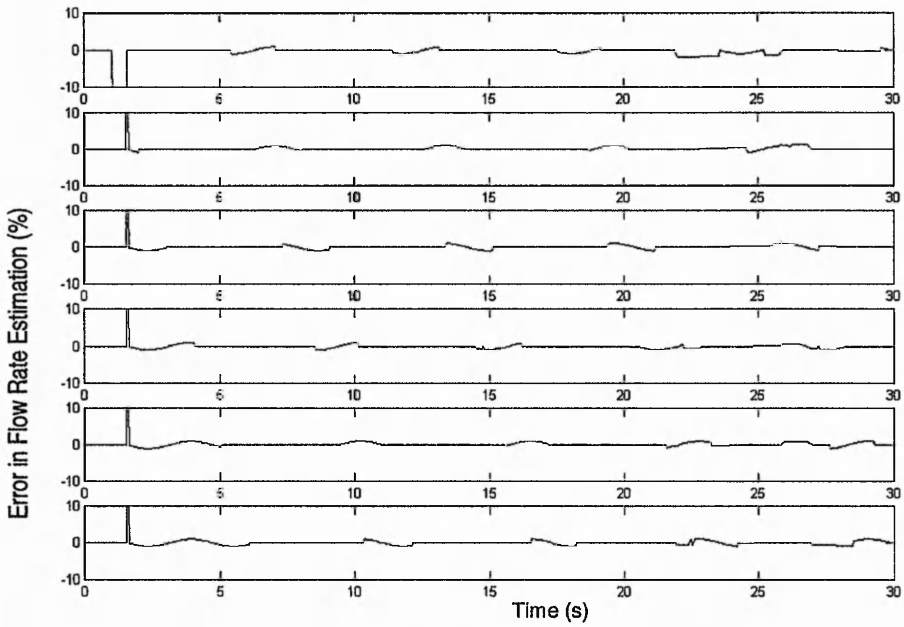


Figure 8.6 Flow Rate Estimation Errors (1 cycle per fill sinusoidal variation)

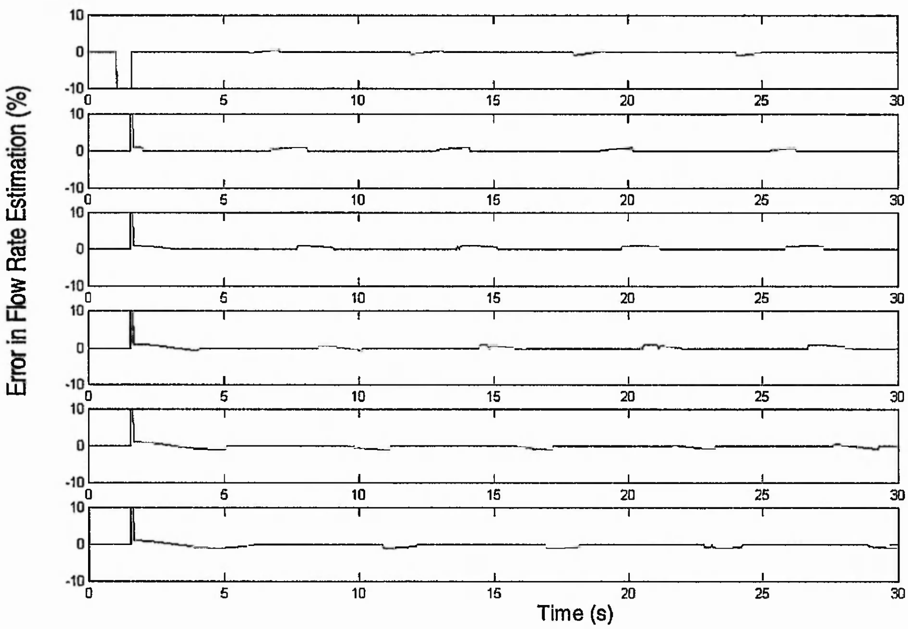


Figure 8.7 Flow Rate Estimation Errors (1/2 cycle per fill sinusoidal variation)

8.3.2 Long term drift

During the operation of a bottle filling plant a gradual variation in flow rate can be anticipated, for example, from the plant operator changing pressures in order to ensure satisfactory carbonation or product mixtures. Figure 8.8 shows a scenario where flow rate drops continuously and uniformly from 150 cm^3/s to 115 cm^3/s over the period of 30 seconds from an unspecified event.

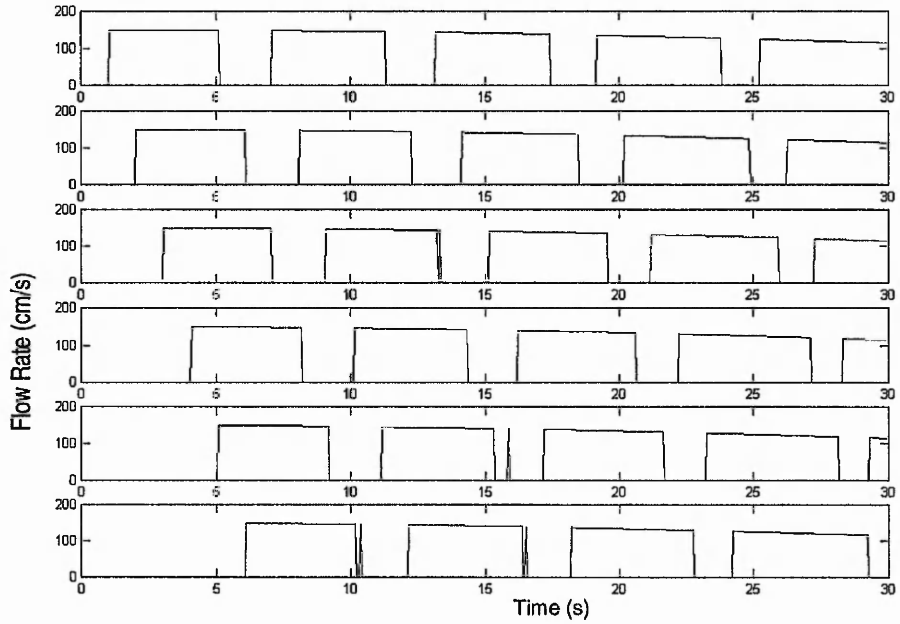


Figure 8.8 Flow Rate affected by Continual Long-Term Reduction.

In this example, the change in flow rate is relatively slow compared to previous examples such that the tracking error is so appreciably close to zero that to show the graph of errors would not reveal any further information.

8.3.3 Remarks

The above examples show that, for any appreciable change in the overall flow rate, the Fuzzy Flow Estimator can easily track variations. In those cases where errors did occur such as the sinusoidal error seen between successive fills in Section 8.3.1, it can be attributed to the output parameter not reflecting operation of the software logic rather than an inherent failure within the estimator. Within software development it is often necessary to utilise variables for efficiency and convenience, therefore values can sometimes be misleading when they are not being actively used. In this case, it was deemed acceptable to allow flow rate to vary between successive fills because the values were not being used for fill height estimation and provided a robust means of storing flow variations within the main estimator logic without resorting to additional program code.

However, it also has to be remembered that in these examples, it was assumed that flow rate was measured accurately and without noise, which as Section 6.2 describes, is unlikely in practical situations. Nevertheless, it has been demonstrated the feedback mechanisms of the Fuzzy Flow Estimator are stable and robust when dealing with global disturbances demonstrated above. It is now necessary to study the operation of the Estimator under variations in flow present only at single valves.

8.4 Local Disturbances

Local changes in flow rate can be attributed to factors such as sticking valves (partial opening) or product build-up causing a long-term reduction in the valve orifice, and hence reduced flow. However, unlike global variations, detection of these effects can only be achieved at the valve which is experiencing problems, and as a matter of course cannot be allowed to detrimentally affect flow measurements from other valves.

8.4.1 Step Change

In the following example a single valve is shown to 'stick' partially open allowing only 50% of the maximum flow rate through. The valve becomes 'unstuck' during the next fill and maximum flow is resumed. All other valves operate normally and no global disturbances are present at this stage.

Figure 8.9 shows that the third valve in the carousel exhibits reduced flow for a period of ten seconds (9^{th} to 19^{th} second). Both the simulated actual and estimated flow rates (dashed line) are shown.

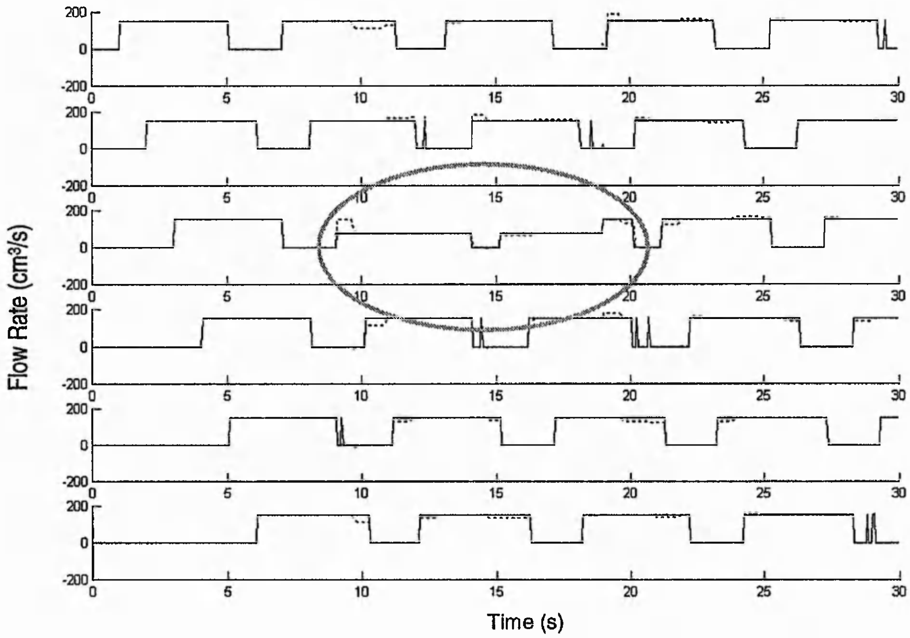


Figure 8.9 Effect of a Step Change on Flow Rate of a Single Valve within a Six Valve Carousel

The estimation error is isolated in Figure 8.10. It can be seen that before the step change at the 9th second errors are zero. The first error is present in valve 3, which is where the change has occurred, but this is corrected once the measurement region is reached for this bottle. Transitory errors can be seen on all the other bottles at the beginning of each fill. This is an interaction of valve errors caused by the feedback mechanisms of the Fuzzy Flow estimator, where flow rates have to be inferred from other bottles if measurements are unobtainable. If one valve or bottle is experiencing problems, valve 3 in this example, when the flow estimates are used from the related sensor then a small degree of cascade error is witnessed in other estimates, but these are corrected once individual measurement regions are reached. After normal flow is resumed on all valves, errors are still present, but can be seen to reduce with each successive fill.

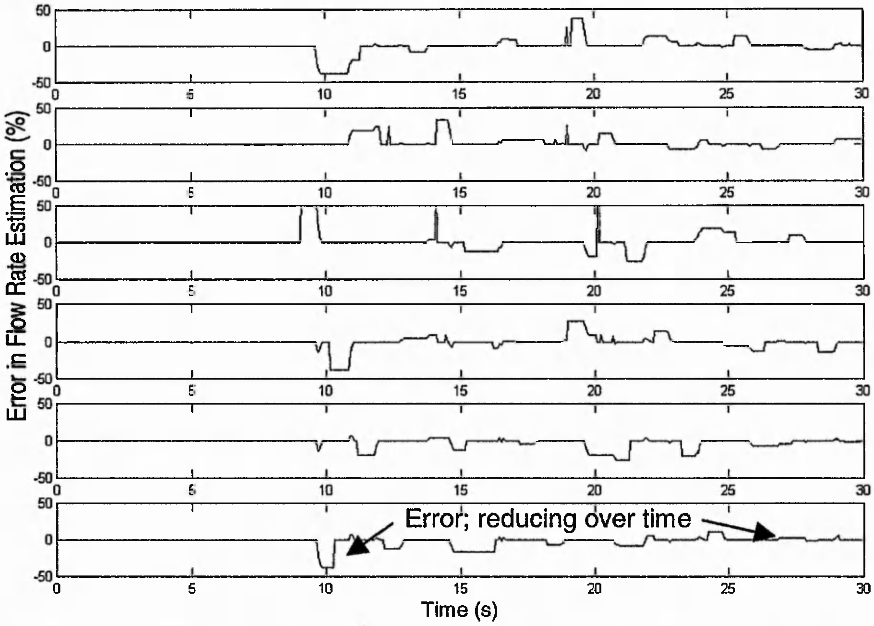


Figure 8.10 Flow Rate Estimation Error from Step Change reduction on a single valve

From the previous two figures it has been shown that the flow rate is tracked the majority of the time. Nevertheless, the observed transitory errors on all the valves could lead to long term bias in estimates if this were the sole means of determining the filled volume. However, these flow rate estimates are used in combination with the Kalman filter as described previously. It should be remembered that the system or model error (Kalman filter parameter, $S = 0.1$) was selected to reduce the dominance of the internal model. In this case, flow rate estimates that are fed into the filter's model are not used with 100% confidence, which leads to no overall bias in volume estimates or height estimates. In addition, the 50% change on valve 3 is tracked resulting in all volume estimation errors below 10% during the disturbance, and minimal once full performance is restored (Figure 8.11).

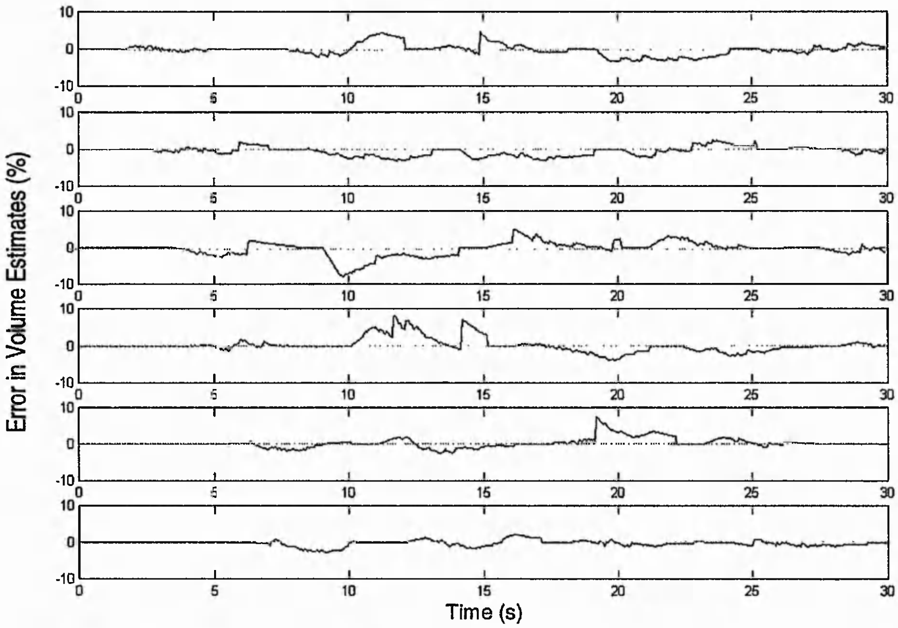


Figure 8.11 Volumetric Estimation Errors during flow rate step change on valve 3

If these volumetric errors are compared to a case where no flow rate disturbances occur, i.e. ideal operating conditions (Figure 8.12), it can be seen that error amplitude is not significantly increased from normal variation caused by inaccuracies of fill height measurement.

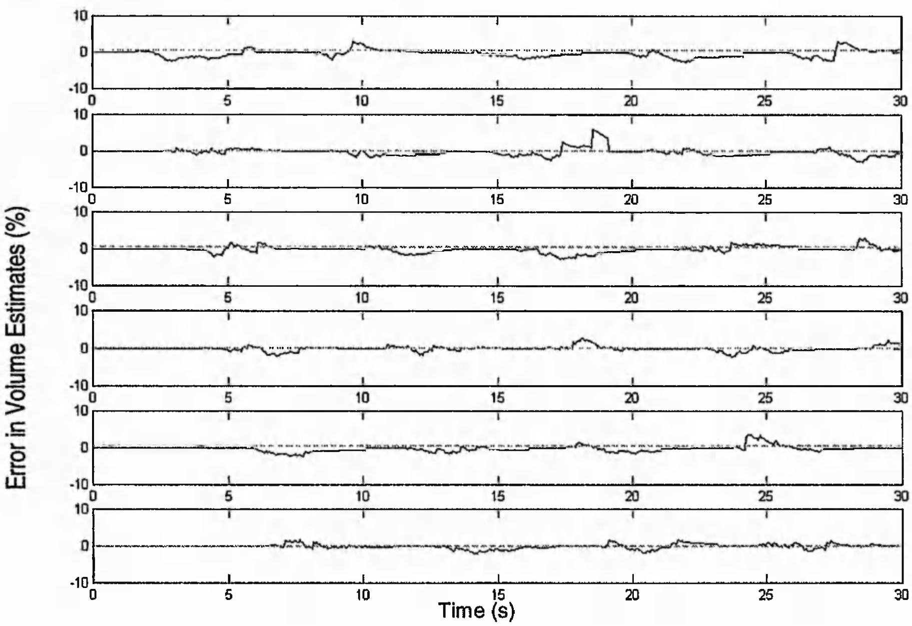


Figure 8.12 Volumetric Estimation Errors generated under constant flow rate

8.4.2 Remarks

The step change represents the most significant disturbance that could lead to destabilisation of the controller, and other variations, such as long-term drift are easier to contain within the strategy.

Although flow rates have not tracked with 100% precision, within the complete strategy, which incorporates the Kalman filter discussed in Chapter 5, a stable means of maintaining filling monitoring accuracy is provided. This is despite a sudden unexpected change in plant performance, namely, the 50% reduction in flow rate from a single valve for a period of 10 seconds.

8.5 Combined Global and Local Disturbances

The most challenging situations for the Fuzzy Flow estimator is in cases where both local and global disturbances in flow rate are occurring simultaneously; this is a scenario that is most likely to be encountered within an actual filling plant. The cases which represent the more challenging tracking problem is a local step change in the presence of a sinusoidal global variation. Two examples are used to demonstrate the performance under these conditions. The first example simply combines the two disturbances with the same system parameters as previously used. A second example will demonstrate how all valves can have independent flow rates and still maintain the same data integrity. This is representative of the actual process, where it cannot be expected for all valves to perform identically.

8.5.1 Identical Maximum Flow Rates

The sinusoidal disturbance is set at approximately one wavelength per filling cycle with an amplitude of $20cm^3/s$ (13.3% of maximum), which is twice the level from examples in section 8.3.1. The step change is identical to the previous example in section 8.4.1. All other system parameters are also identical to previous examples. Figure 8.13 shows how the flow rate varies over the six valves. The solid line is the actual (simulated) flow rate and the dashed line is the estimate, which is only visible when the value differs from the actual.

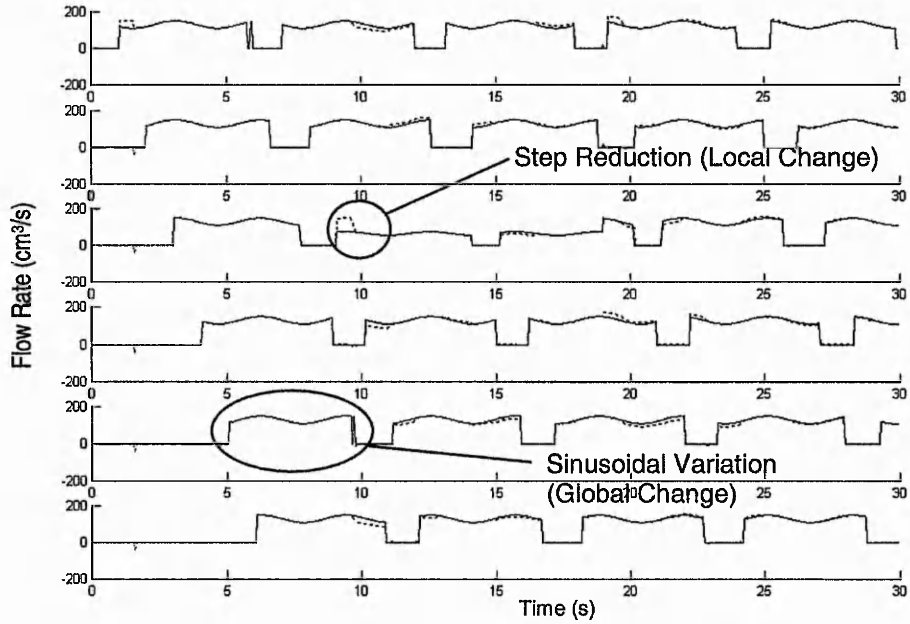


Figure 8.13 Flow Rates for Six Valves undergoing Global Sinusoidal and Local Step Change Variation

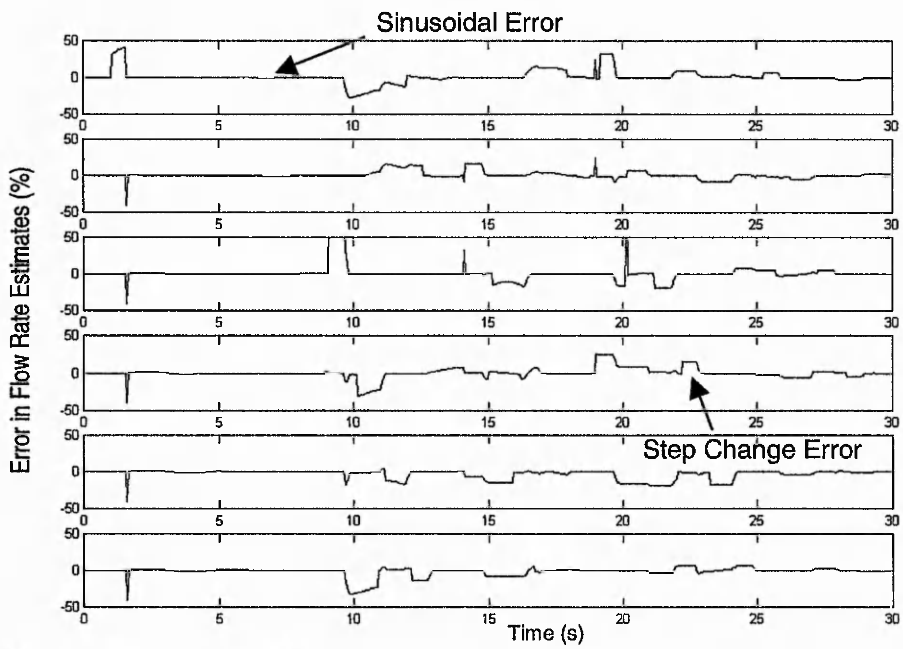


Figure 8.14 Estimation Errors from Combined Sinusoidal Variation and Step Change to Flow Rate

Figure 8.14 shows estimation error characteristics representative of both global (slight sinusoidal errors) and local disturbances (large transitory errors). However, these remain largely independent of each other, except for some simple addition. Nevertheless, the combination of disturbances does not destabilise the performance of Fuzzy Flow Estimator. It can be seen from the volumetric error plots, Figure 8.15, that estimation of the volume (by combination of flow rate and Kalman filter height estimates) shows no more significant estimation error than for local disturbances only (see Figure 8.11). In fact in this example, errors are marginally smaller, which is a property of the random nature of the generated measurement noise rather than any particular improvement via the developed strategy.

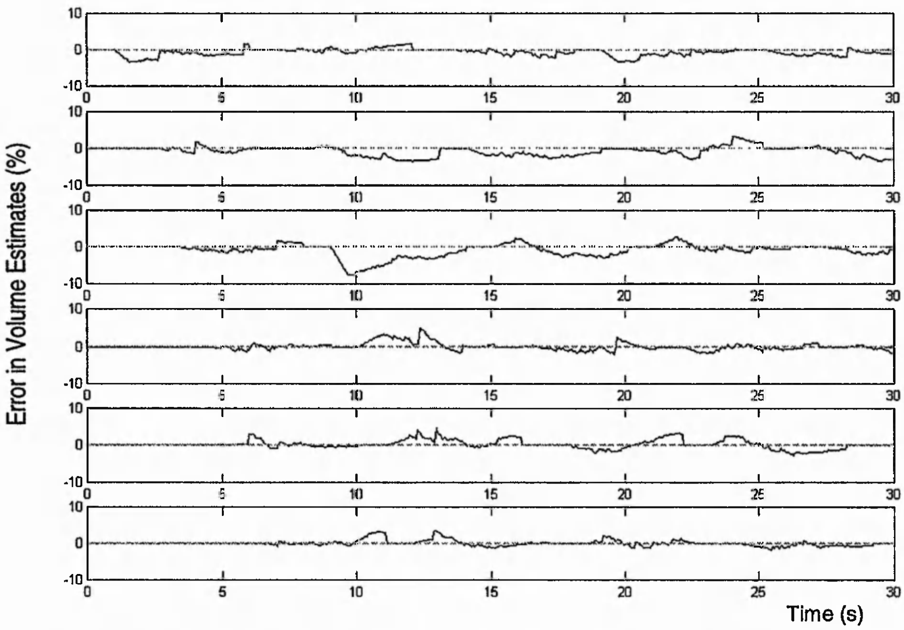


Figure 8.15 Volumetric Estimation Errors from Combined Global & Local Disturbances

8.5.2 Different Maximum Flow Rates

In this example it will be demonstrated that the strategy does not rely on having identical flow rates for all valves. This scenario could be representative of performance variations of an actual plant where valves have drifted from optimal condition because of long term wear-and-tear. The flow rates of the six valves when fully open are reduced from the maximum obtainable by the amount shown in Table 8.1.

Valve	Percentage of Maximum
1	70%
2	80%
3	100% (intermittent 50%)
4	90%
5	60%
6	100%

Table 8.1 Fixed Flow Rate Offsets for a Six Valve Carousel

In order to allow bottles to fill completely with the proposed reductions in flow rate, the rotational speed of the virtual carousel has to be reduced; hence, the time allowed for the filling process is increased from 6 to 8 seconds. The enforced global and local disturbances remain the same as the previous example.

Figure 8.16 shows the variation in flow over the six valves. It can be seen that the estimate (dashed line) shows more deviation from the actual (simulated) flow rate than in the previous example where maximum flow rates of all valves were identical; nevertheless, deviations remain relatively small.

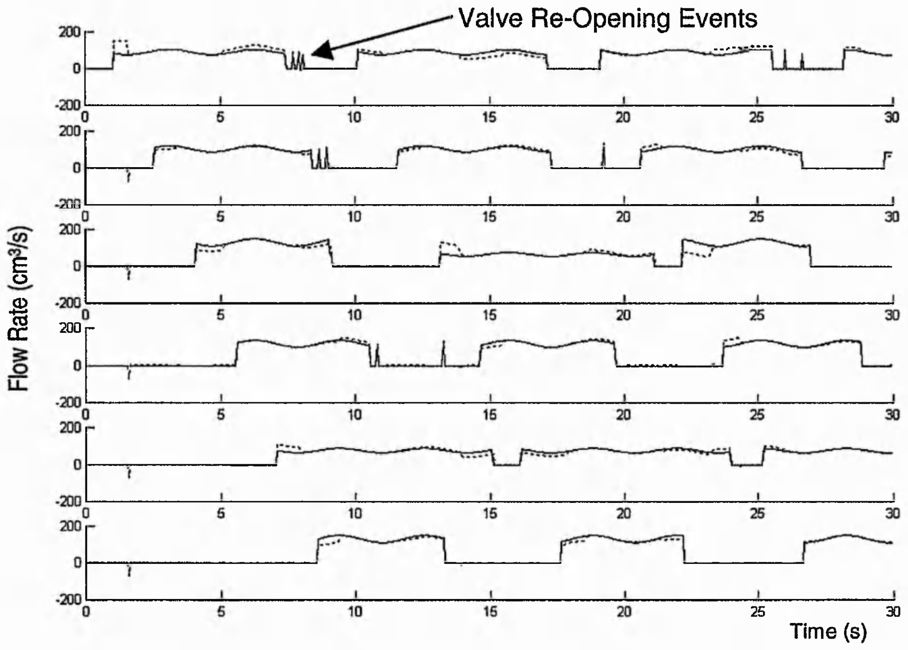


Figure 8.16 Sinusoidal Variation, Step Change and Differing Maximum Flow Rates

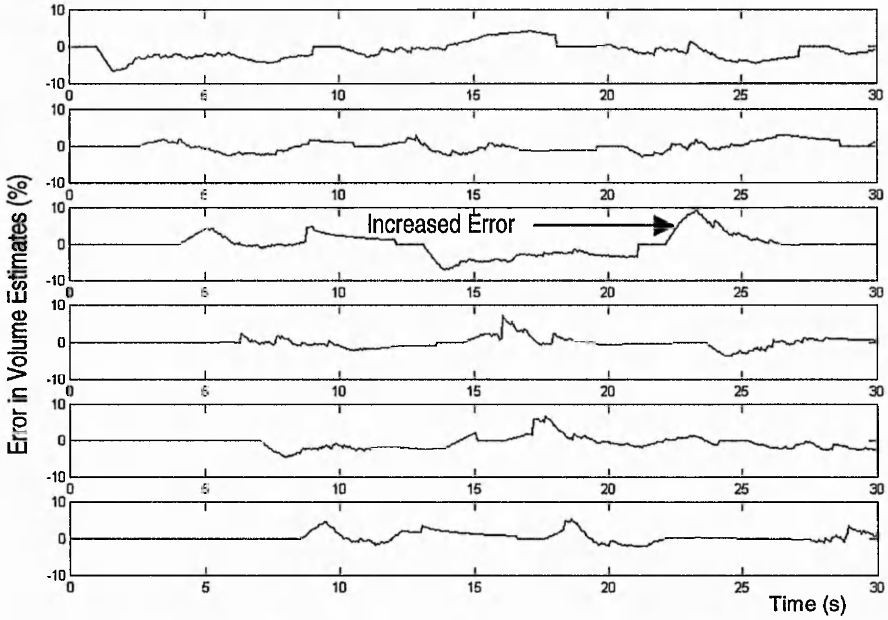


Figure 8.17 Volumetric Estimation Errors from Sinusoidal Variation, Step Change and Differing Maximum Flow Rates

The volumetric errors (Figure 8.17) are increased in some locations, most notable on the trace for valve 3, but overall performance remains stable, albeit with increased deviation from the previous example (Figure 8.15). Figure 8.18 shows the comparison of estimated and ideal filled volume. The estimate (dashed line) can be seen in the context of the true value and it can be noted that a close match is presented for much of the time.

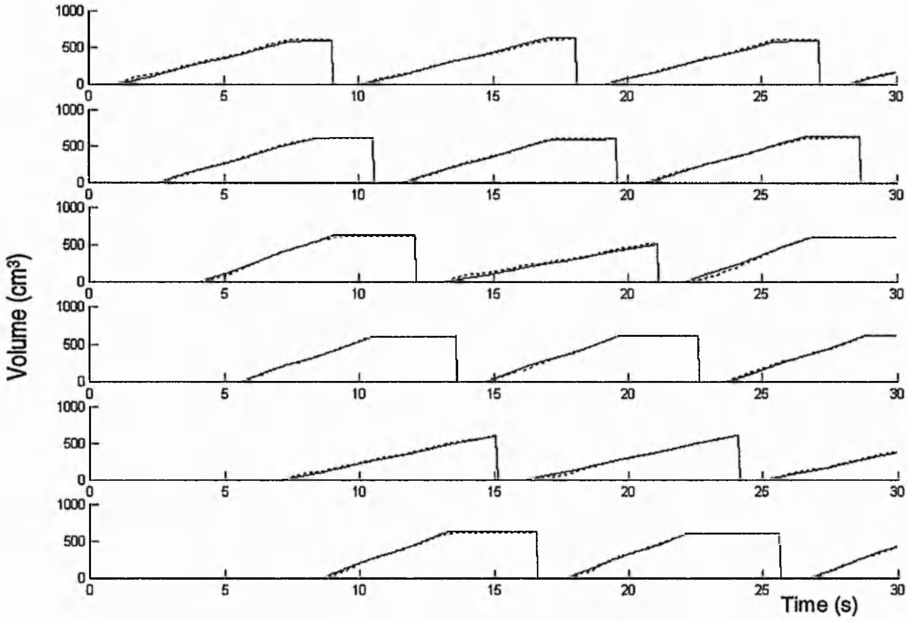


Figure 8.18 Fill Level Estimates (dashed) and Actual (solid) where Sinusoidal Variation, Step Change and Differing Maximum Flow Rates are present

8.5.3 Remarks

These examples have demonstrated that the Fuzzy Flow Estimator is inherently stable under conditions where flow rates are affected by both local and global disturbances. It has been shown that, for much of the time, volumetric errors are significantly below 10%, with only occasional spikes. In addition, the final example demonstrates the independence of each valve measurement from others, such that individual flow rate characteristics can be accommodated.

8.6 Full Plant Model

This final example uses the maximum capabilities of the simulation, namely, sixty-four valves in the carousel, and demonstrates how increased modelling stability is created when using more valves. For clarity, only the results from the first 6 valves are presented. However, this amply demonstrates the improvements in estimation performance that could be theoretically achieved with this level of plant complexity.

The conditions for the model are essentially identical to previous simulations, including local and global disturbances. Refer to Figure 6.18 for an example of the types of flow rate changes used. The main difference between this and previous examples is that the time between adjacent fills is compressed, because for the same carousel rotational speed more bottles have to be filled.

Figure 8.19 shows the volumetric estimation errors from just six valves. In comparison, Figure 8.20 shows 6 of 64 traces for volume errors with the same parameters as before, except an increased number of valves. The amplitude of the errors is visibly reduced in all cases. A small number of transient events remain, however, they also show some reduction (e.g. valve 5 around 22 seconds).

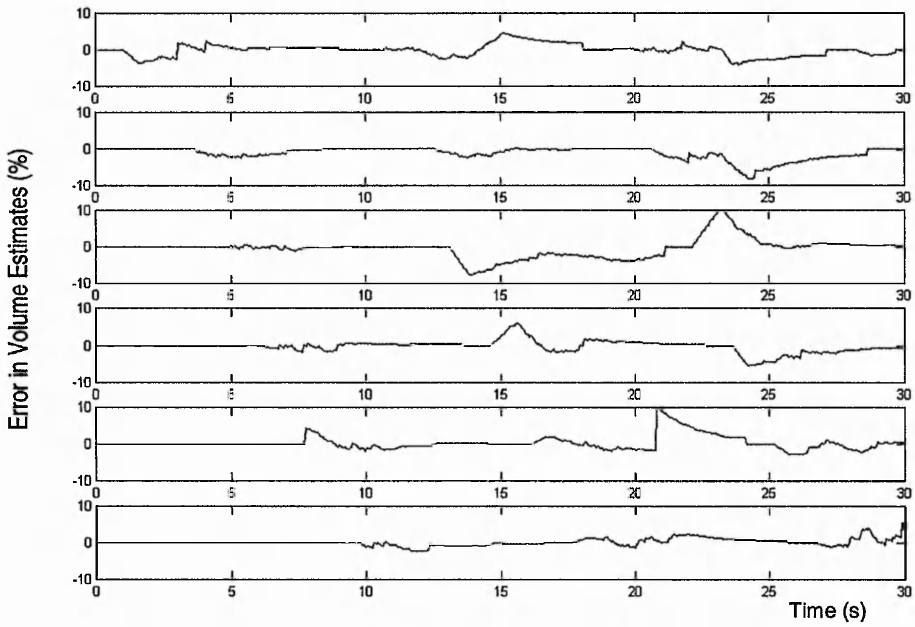


Figure 8.19 Volumetric Estimation Errors for a Six Valve Carousel

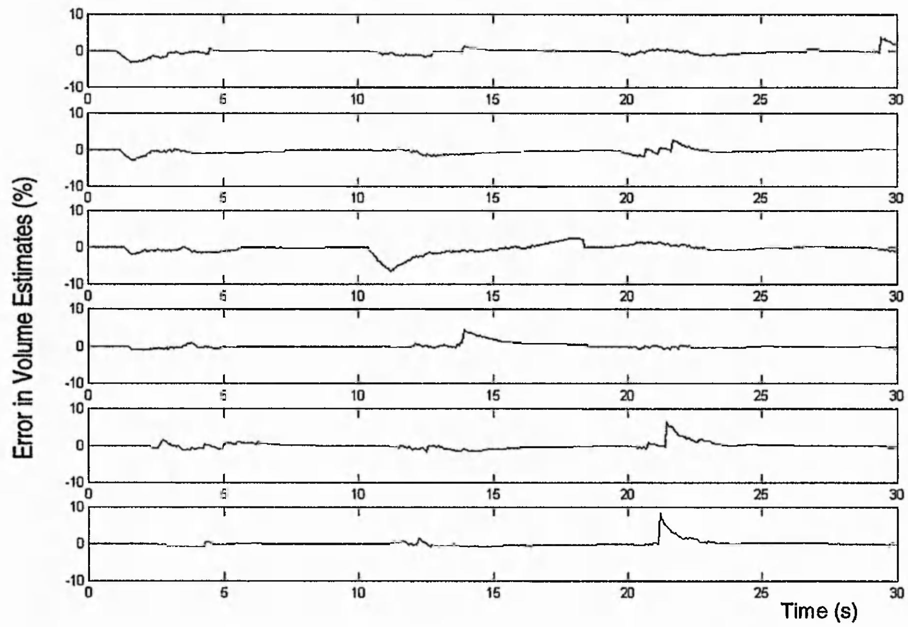


Figure 8.20 Volumetric Estimation Errors for a Sixty-Four Valve Carousel

More conclusive evidence of increased stability is found by looking at the flow rate estimate errors. Figure 8.21 shows the six valve example where transitory errors are present in all valves resulting from the 50% step change in flow occurring in valve 3 (see also Section 8.4.1 and Figure 8.14).

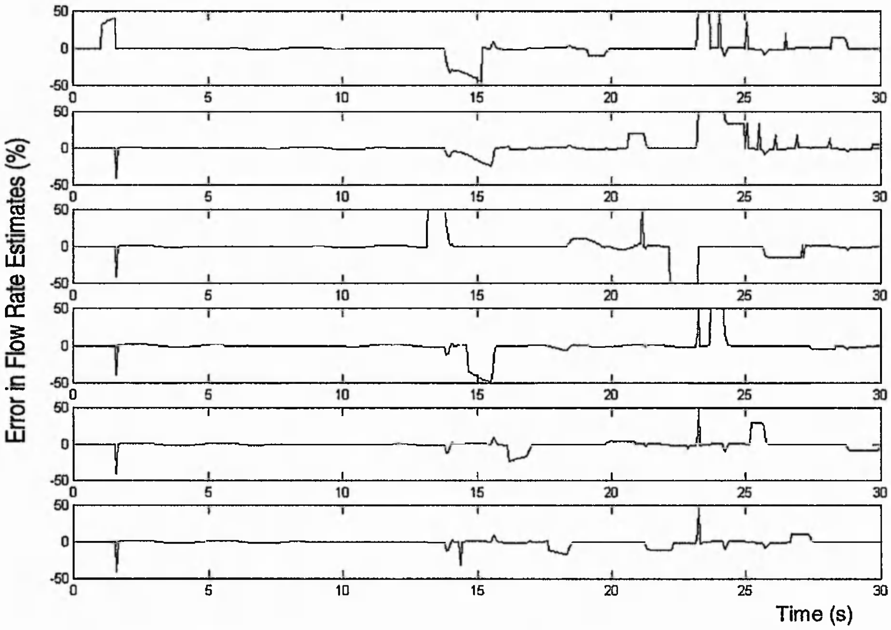


Figure 8.21 Flow Rate Estimation Errors for a Six Valve Carousel with 50% Step Change in Flow Rate on Valve 3

A more serene picture is found from the 64-valve simulation (Figure 8.22), where disturbances on valves, apart from the third, are kept to a minimum. A transient error is still present initially, which is a consequence of initialising flow rate models on the first fill.

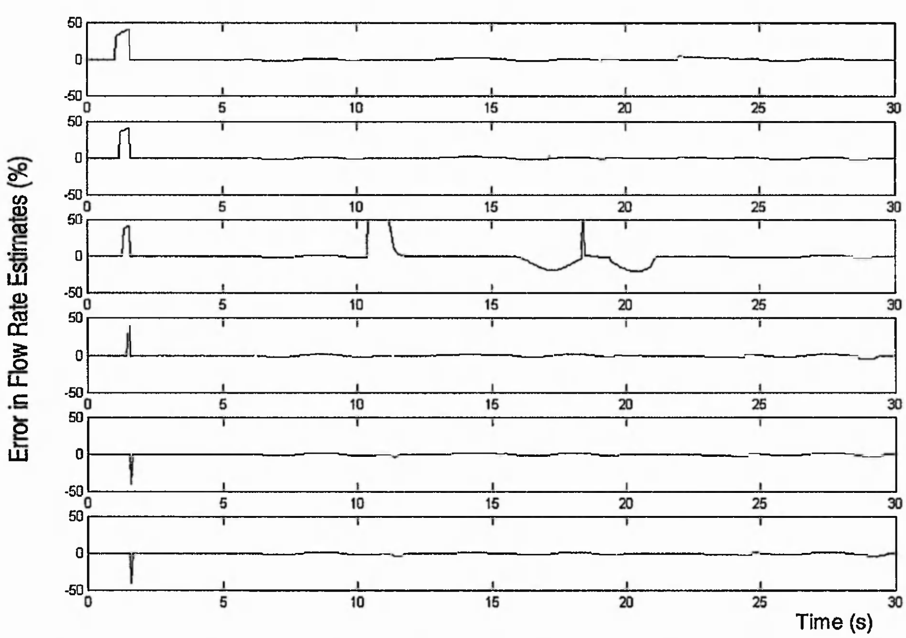


Figure 8.22 Flow Rate Estimation Errors from a Sixty-Four Valve Carousel with 50% Step Change in Flow Rate on Valve 3

8.6.1 Remarks

This example has demonstrated that stability of the complete system is enhanced when the number of valves is increased. This suggests that scalability is inherent within the strategy and that large filling systems will benefit most from the proposed approach.

8.7 Summary

This chapter has demonstrated the combination of plant dynamics, Kalman filters and Fuzzy Logic estimation in a comprehensive liquid level tracking system. Using an expanded version of the previously developed plant simulation for single valves, several examples were given on the performance of the Fuzzy Flow Estimator proposed in Chapter 7. The simulation results have indicated that the proposed approach offered useful characteristics that would be beneficial for the tracking of both local and global disturbances of flow rate. Furthermore, these flow rates can be exploited by the Kalman filter to improve height or volume estimation at the individual valves, which provides a solid foundation for liquid level control in a bottling environment.

Chapter 9
Discussion and Conclusions

9.1 Discussion

9.1.1 Measurement of a Single Vessel

It has been described that the measurement of liquid levels using ultrasound, although offering several benefits such as placement and size, would be affected substantially by the medium through which the signal passes, i.e. the gas above the liquid surface. Upon investigation, a possible solution was offered in the form of estimation theory and more specifically the Kalman filter. However, from the reviewed work it was made clear that additional effort would be required to maximise the performance of the filter within the filling application because of non-linear factors, like bottle shape and non-Gaussian noise.

To test and optimise the Kalman filter directly on filling plant or test rig would be, at best, impractical. In order to have full control of every parameter it was necessary to model the process within a simulation, using experimental data to provide a high level of realism. A modular approach was taken to the software development to allow continual improvement of the models, as more information on the process becomes available from areas investigated outside this particular study, which represents part of larger whole.

A series of simple ultrasound experiments were conducted that allowed the noise present in the signal to be analysed to reveal that a Gamma distribution represented a close model of the disturbances. Although, the experiment did not attempt to reproduce the extremes of measurement that can be expected within the actual filling process, (i.e. Carbon dioxide, raised temperatures etc), based on the observation of the factors attributed to the degradation of accuracy, the gamma distribution appears to be a suitable choice. By modifying the two parameters α and β, the shape of the distribution can be altered to create different probability spreads.

However, the Kalman filter is primarily designed for linear systems with Gaussian noise of zero bias. It was demonstrated that, with the particular gamma distribution chosen to compare favourably to experimental results, the performance of the filter is not severely degraded. This can be justified through two lines of reasoning. Firstly, the shape of the distribution is not overly dissimilar from a standard Gaussian distribution. Secondly, the filter relies to a large extent on the internal process model for the production of state estimates. Such that measured values falling outside the expected range are largely replaced by model values within the estimates. A major benefit of using the Kalman filter within this type of application is the inherent robustness, which, as shown by the simulation runs with increasing levels of noise, can produce stable estimates even in the presence of large degrees of noise. Furthermore, factors such as valve opening characteristics can be accounted for without explicit modelling by giving different initialisation values of the covariance error.

The filter may be seen as a relatively complex approach to noise reduction, compared to simple damping or averaging techniques. However, once created, the filter can successfully operate under a range of conditions with very little additional overheads in time or effort. The experimental results obtained from the filling of a simple bottle have shown that, although not always optimal, the fill level estimates produced offer a useful starting point for the implementation of a monitoring or control strategy.

The requirements of the filling application mean that the controller for the valves is relatively simple, as the valves only have open or closed states, albeit with associated characteristics. A simple predictor was used to provide actual control by utilising next iteration estimates from the Kalman filter. A limitation of the current simulation is the inability of the controller to be de-coupled from the iteration time of the filter and the rest of the program. This means achievement of the desired set point is more reliant in many cases to the iteration time step of the program rather than any monitoring accuracy that is attained. This is especially relevant in models of bottles with narrowing necks where the liquid level accelerates as it approaches the set-point. This means the largest difference in height between successive iterations occurs around the place where greatest accuracy is required, thus overall fill level suffers. Nevertheless, by concentrating on the monitoring and measurement aspects of the problem, the work hopes to make clear that when the controller is de-coupled from the iteration time step, performance can be expected to meet the expectations of the application. It has been shown that the volumetric error is often less than 1% and largely below 2% even in cases where measurement accuracy is severely affected by noise (Section 6.4.6).

Overall, it has been demonstrated that a basic Kalman filter can provide a foundation on which to build a measurement or control strategy for the filling process. However, limitations exist that could reduce performance significantly, namely variation in flow rate, which was largely ignored in the initial study.

9.1.2 Multiple Vessel Carousels

The lack of flow rate measurement could be overcome by monitoring of the Doppler shift of ultrasonic echoes, which can be used to determine the speed of a moving liquid surface. This could then be further processed to find the associated flow rate. The simple investigation conducted showed that, although a shift in the ultrsound frequency could be detected, a significant amount of processing was required in the form of Fast-Fourier Transforms (FFT), to an extent that real-time implementation could not be realised within the configuration tested. The FFT algorithm was chosen because it was readily available feature of the Matlab® software already being used in the research and would provide a quick and simple means of confirming the Doppler shift phenomenon. However, it was found to be computationally inefficient and not particularly accurate with this application (due to sample size, see Eq. 2.5). However, if the processing could be performed directly by dedicated hardware, such as a frequency counter, then processing times could be reduced. For experimental purposes, a proprietary counter would be suitable, but for practical implementation a custom built dedicated system would be more appropriate. This would inevitably be part of a complete signal processing unit (e.g. Digital Signal Processor – DSP) that would perform much of the processing currently undertaken in software. The personal computer is a useful platform for testing and development, but does not represent a practical solution for real-time applications, as robustness is not guaranteed within the standard operation system. Despite these current limitations, the investigation has revealed that Doppler shift is a potential means of obtaining flow rate information on the process without further sensors being required. Under this assumption a strategy has been developed that utilises some aspects of fuzzy logic to provide robust monitoring, which allows flow rate to be incorporated into Kalman filter internal models.

The greatest benefit of the Doppler Shift measurement was observed to be in multiple vessel scenarios, i.e. valve carousels, where information from many simultaneous filling events could further improve tracking performance.

A number of modifications to a basic fuzzy logic approach were considered and these have been used to produce two fuzzy processors that link with each other to track both local and global changes in flow rate. By dividing a prospective vessel into three parts – BASE, BODY and NECK – intelligent inference can be made on the suitability of measurements obtained and hence, a confidence level given. This helped to reduce uncertainty and increase robustness, as only purely cylindrical sections are suitable for flow rate measurement by the Doppler method in this implementation.

The simulation results demonstrate a range of different operational conditions that the system may encounter. Six valves were tested in a carousel formation and subjected to changes in flow rate from effects local to individual valves or global factors influencing all valves. The results have shown that robustness and stability are satisfactory, with the most disturbing influence being a local step change on a single valve. Nevertheless, the error for the calculated volume was almost entirely below 10% and often much smaller at less than 2%. Furthermore, an example using sixty-four valves showed the inherent stability of the developed tracking system and how it would provide a suitable foundation for controlling a multiple vessel filling process.

9.1.3 Overall

The aim of this investigation was to develop an intelligent control system for container filling plant, and to achieve this, a strong emphasis has been placed on the development and testing of algorithms that would provide a level of robustness suitable for an industrial setting. This is especially important in the presence of noise or disturbances within the measurement signals, to which ultrasound is susceptible. Consequently, less emphasis has been placed on ultrasound experiments, which is the undertaking of another project. Moreover, the developed strategies have not been solely aimed at ultrasound applications, but are indeed applicable to a wider range of sensor problems, especially with regard to the parallel parameter tracking of the Fuzzy Flow Estimator.

The developed fuzzy approach is, to the best of this authors knowledge, novel, and has an influence greater than the confines of this investigation. It has been demonstrated that not only can fuzzy logic techniques be used to create robust techniques but can also be modified to form an accurate mathematical representation of a given problem. In this instance, flow monitoring is the application, but the approach could be adapted for use on general condition monitoring systems with multiple parameters or systems which require a degree of data fusion to combine many received measurements into a more useful parameter.

Despite the success of many areas of this investigation, some improvements could be made to reduce limitations currently present within the simulation. One limitation of the modelling is that the iteration time of the whole model is tied together, such that the timing of the controller, Kalman filter and fuzzy systems is inextricably tied together. Although this is adequate for monitoring and tracking demonstrations (the underlying equations are mainly continuous) for control purposes the lack of independence prevents fine timing required to achieve satisfactory set points. In many ways, this is a limitation of many digital simulations, namely, the inability to model the events between time steps, which is why experimental evidence is often required to verify computational results. However, this investigation has concentrated on the areas that are modelled reliably, (i.e. monitoring and tracking), and hopefully demonstrated that in this application the greatest problems lie within this area as opposed to complex control characteristics that need to be overcome.

Nevertheless, for further study on the container filling problem, a simulation capable of independent timing between processing elements would be useful.

This investigation has relied to a large extent on simulation with some practical experimentation to provide insights into physical processes such as measurement noise. To progress closer to a practical system for 'real world' implementation, it is clear that more detailed experimentation would be required, especially in the area of Doppler shift measurement and processing. This would give greater confidence to the proposed strategy and also provide further insight into the physical constraints of measuring liquid levels by means of ultrasound sensors in the filling environment.

9.2 Benefits to Industry

A clear advantage of the strategy developed here is that it can be implemented incrementally. Firstly providing monitoring only before closing the loop and proving complete control. This would allow confidence in the approach to be nurtured within the bottling industry, who are unlikely to willingly invest completely and whole-heartedly in an, as yet, unproven system.

The following points, though, can be made.

- Ultrasound can be a viable tool for assessing liquid levels, especially under the guidance of an intelligent filtering system, such as the Kalman Filter.
- The monitoring aspects alone would provide valuable information on plant performance and would no doubt allow efficiency to be increased.
- Robustness is an inherent feature of the developed approach both in the Kalman filter and the Fuzzy Flow Estimator, thus, a high confidence can be given on the long-term performance of the system.

An advantage of the incremental approach is that data could be obtained from the implementation of a monitoring system, which would then allow refinement of control algorithms, such that upon introduction, performance of the system will necessarily be refined to an actual process, rather than sole reliance on simulation.

9.3 Areas for Further Study

9.3.1 Ultrasound

The focus of this investigation was firmly placed within the control and monitoring aspects of the filling process. Nevertheless, to take the strategy closer to implementation it will be necessary to conduct further work directly on the characteristics of ultrasound. Four areas of study have been indentified; pressure, CO_2, surface agitation and temperature (thermals). The following factors and questions are of particular interest.

Pressure

- Verify effects of pressure variations on waveforms with intended transducer
- Quantify damping effects on transducer face, effects of resonance and power requirements

CO_2

- Investigation into conditions where CO_2 is present
- In what percentage is CO_2 found?
- Is water vapour present?
- Can problems be easily avoided?

Surface Agitation

- Investigate various modes of agitation
- Use wavetank to help quantify effects such as wave height and scattering (Figure 9.1)
- What is the effect if the surface is not normal to transducers?
- Chaotic turbulence

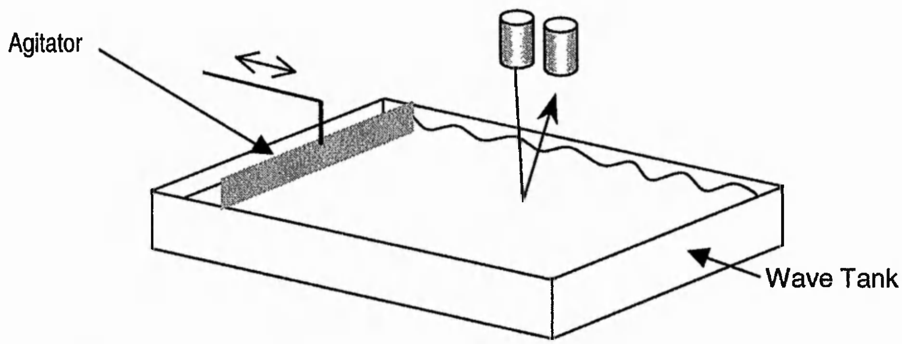


Figure 9.1 Suggested Agitation Experimental Rig

Temperature

- Effects of hot or warm liquid below cool atmosphere, i.e. thermal scattering and refraction (Figure 9.2)

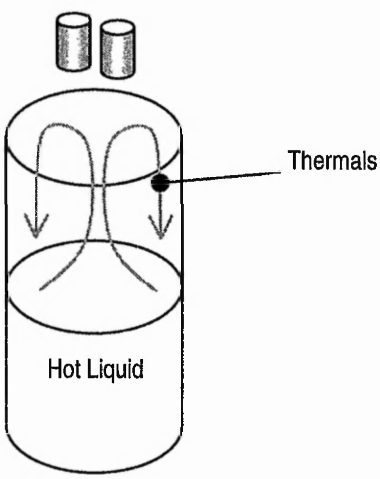


Figure 9.2 Suggested Thermal Scattering/Refraction Experimental Rig

Doppler

- How does noise influence measurements of flow rate

The above issues relate to the environmental conditions within which the ultrasonic measurements will be made. Of equal importance is the behaviour of the transducer and associated electronic hardware. These can be classified as static and dynamic factors.

Static

- Sensitivity
- Accuracy, uncertainty, precision and bias
- Threshold, resolution, dead band and hysteresis
- Linearity
- Analogue Drift

These can be measured with calibration

Dynamic

- Time constant, response time and rise time
- Overshoot, settling time, damped frequency
- Frequency response

These can be measured by applying a known change.

Many of the listed characteristics will be specific to one particular transducer configuration, and thus can only been found when final prototypes of the system have been designed. However, the developed simulation allows for continual modifications to be made. Thus, optimisations can be made at any stage of development.

9.3.2 Condition Monitoring

Two types of condition monitoring could be considered for this application, a relatively simple interpretation of the parameter values or a more complex fault diagnosis system based on heuristics. However, neither approach is implemented, as it has to be applied in real-time to be appreciated properly. Instead, some details of possible designs are shown.

9.3.2.1 Parameter Interpretation and Display

The simplest system of visual feedback for the operator is simple graph or text display of the parameter value. However, when many parameters need to be checked it is important that attention is drawn to values approaching or exceeding performance tolerances. To this end a simple system can be used giving display indicators on the computer screen changing from green, through yellow, to red, depending on the value.

Computers base their colour representation on three colours, red, green and blue, from which all other colours can be derived. The amount of the three colours in an 8-bit graphics mode is controlled by values from 0 to 255, with 255 being the maximum.

By using fuzzy set-style descriptors it is possible to create indicator display elements, which vary smoothly over a parameter's range as shown in Figure 9.3. It is also possible to change the colour range, to either increase or decrease sensitivity to variation by moving the mid-point (figures 9.4 and 9.5). Therefore, tolerance characteristics can be taken into account without changing the value of the parameter itself. Definition of the colour range can be achieved using the Fuzzy Logic modules already written and hence, makes a simple addition to the developed system.

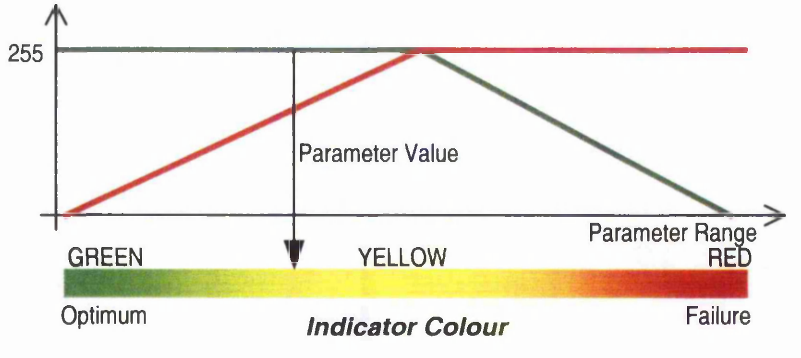


Figure 9.3 Fuzzy Parameter Display

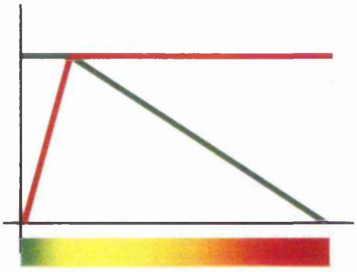


Figure 9.4 Small Safe Zone

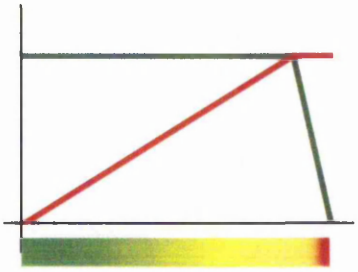


Figure 9.5 Large Safe Zone

9.3.2.2 Failure Case Analysis

The controller for the filling plant contains a large of amount of modelling to achieve the performance required for this application. This benefits the condition monitoring aspect, as values can be compared to ideals without significant additional work in creating process models. Moreover, with the computational algorithm approach, it is simple to extract values from anywhere in the control chain.

Fuzzy logic or an expert system could be used to develop a rule base containing the common failure cases, in combination with the visual display proposal from the previous section, the operator could be warned of impending or actual failures.

To maximise robustness, feedback could be initiated from the condition monitoring system back to the controller and modelling elements to protect against unwanted behaviour in the presence of failing or degrading performance.

The failure cases that could be typically expected are categorised below (Figure 9.6). They are grouped into four distinct areas, valve, sensor, controller and communications.

Valve		
Actuation failure		
	Intermittent action	
	No movement	
		Failure to open
		Failure to close
	Reduced speed	
		Opening
		Closing
Product build-up		
	Poor seal	
	Reduced flow rate	
Sensor		
No signal		
Bias		
Intermittent Signal		
Communications		
No data		
Controller		
Control Failure		

Figure 9.6 Failure Cases

9.4 Conclusions

The aim of this investigation was to develop an intelligent monitoring and control system for bottle filling plant and to complement the use of ultrasound as a primary sensor. To this end the following achievements have been made,

- The potential of ultrasound has been maximised by use of a Kalman filter to reduce noise present in the measured parameter.
- Practical experiments have revealed that a gamma distribution is a good statistical description of the noise within some measurement environments
- Simulation results have shown that the approach is a viable method for improving measurement performance and thus, control is necessarily more robust.
- The simulation has been developed to a stage where up to sixty-four valves can be modelled in a carousel arrangement
- Further information, in the form of Doppler shift of the ultrasound echo, can be extracted to allow for information on the flow rates within individual vessels
- A fuzzy logic-based system has shown how the strategy can benefit from the flow information and thus, is further improved within the multiple valve environment.
- The developed strategy has been demonstrated to be both stable and robust.

In addition to the objectives met, two innovative and novel areas have been investigated; these are,

- The use of a Kalman filter within a liquid level context
- A novel fuzzy logic system has been devised, which is not only capable of robust performance, but also offers a high degree of mathematical accuracy for the input-output relationship.

9.5 Final Remarks of the Author

The work that has been presented here represents three years of investigation, which has at many times pleasantly surprised even the author as to the solutions finally presenting themselves, like the Fuzzy Flow Estimator. It is with only a limited degree a foresight that these projects are undertaken and often the final solution can remain subtly hidden until the last moment. However, sometimes, as in this case, the solution, (which is only one of a possible many), unfolds with a degree of congruency that is unexpected. From vague ideas on the usefulness of parallel sampling within the multiple valve environment, to a solution utilising fuzzy techniques initially developed before the commencement of this investigation, the work has constantly rewarded innovative approaches.

It is hoped that this work, especially the fuzzy systems, can be put to further use now this investigation has come to a close, and perhaps even outside the confines of the bottle filling problem.

In summary, the research undertaken has not only supported ultrasound as a measurement tool and provided a context within which it can be used to improve bottle filling performance, but also has advanced core research in areas of monitoring and parameter tracking for a much wider scope.

Chapter 10
References

Asher, R.C., 1997, *Ultrasonic Sensors for Chemical and Process Plant*, Institute of Physics Publishing, Bristol and Philadelphia.

Blitz, 1971, Ultrasonics: Methods and Applications, Butterworths

Chao, C-T. and Teng, C-C., 1997, "A PD-like self-tuning fuzzy controller without steady-state error", *Fuzzy Sets and Systems*, **87**, pp.141-154.

Cho, C.H., 1982, *Measurement and Control of Liquid Level*, Instrument Society of America – Independent Learning Module.

Cox, E, 1994, The Fuzzy Systems Handbook, Academic Press, London.

Daum F.E. , 1988, New Exact Nonlinear Filters, *Bayesian Analysis of Time Series and Dynamic Models,* ed. Spall,J.C. Marcel Dekker INC

Devroye, L, 1986, Non-uniform Random Variate Generation, Springer-Verlag, New York.

Dwyer, 1989, Contributions to Non-Gaussian Signal Processing, *Topics in Non-Gaussian Processing*, Springer-Verlag, New York

Feng, G., Cao, S.G., Rees, N.W. and Chak, C.K., 1997, "Design of fuzzy systems with guaranteed stability", *Fuzzy Sets and Systems*, **85**, pp. 1-10.

Graham, B and Newell, R, 1988, Fuzzy Identification and Control of Liquid Level Rig, *Fuzzy Sets and Systems* **26**: 255-273

Graham and Newell, 1989, Fuzzy Adaptive Control of a First Order Process, *Fuzzy Sets and Systems*, **31**: 47-65

Graebe, S.F. and Bohlin, T., 1993, "Identification of Non-linear Stochastic Grey Box Models – Theory, Implementation and Experiences" *Adaptive Systems in Control and Signal Processing, 4th IFAC symposium*, Grenoble, France, 1992. Selected Papers pp. 47-52.

Grimble, M.J. and Johnson, M.A., 1988, *Optimal Control and Stochastic Estimation: Theory and Applications Volume 2*, A Wiley-Interscience Publication, John Wiley & sons Ltd, Chichester.

Holst, J., Holst, U., Madsen, H. and Melgaard, H., 1993, "Validation of Grey Box Models", *Adaptive Systems in Control and Signal Processing, 4th IFAC symposium*, Grenoble, France, 1992. Selected Papers pp. 53-60.

Hsieh, C.S. and Chen, F.C., 1995, "Optimal solution of the Two-Stage Kalman Estimator", *Proc. 34th CDC*, New Orleans.

Hull J.B., Peers N., Whalley R. & Zeng Z., (1994) Fluid Level Monitoring by Ultrasound for Process Control during Filling and Retail Food Containers, *Applications of Multivariable System Techniques, AMST 94*, (pp 329-345) Bradford, March 1994.

Hull, J.B., Henthorn, K.S., Muumbo, A.M., 1995, "Controlling Waste in the Food Processing Industry", *Advances in Materials and Processing Technologies, AMPT95*, Dublin, pp31-39.

Jeffries M, E Lai, Plantenberg D.H. and Hull J.B. 1997. "A Fuzzy Approach to the Condition Monitoring of a Packaging Plant". *Proc. 6th International Scientific Conference on Achievements in the Mechanical & Materials Engineering AMME '97.* The Silesian Technical University, Gliwice, Poland, 28 November - 3 December 1997, 95-98

Jeffries M, Lai, E, Plantenberg, D.H., J.B., Hull, 1998, "Real-time Implementation of Fuzzy Condition Monitoring System for Predicting Process Breakdown", *Proceeding of Engineering Intelligent System*, Tenerife.

Julier, S.J. and Uhlmann, J.K., 1997, "A New Extension of the Kalman Filter to Nonlinear Systems", *Signal Processing, Sensor Fusion and Target Recognition VI*, SPIE Vol **3068**, pp.182-193.

Kalman, R.E., 1960. "A New Approach to Linear Filtering and Prediction Problems", *Journal of Basic Engineering*, **82**, pp. 35-45.

Kalman, R.E., and Bucy, R.C., 1961 "New Results in Linear Filtering and Prediction Theory", *Journal of Basic Engineering*, **83**, pp. 95-108.

Kandiah, S, 1996, Fuzzy-based Predictive Control of Chemical Processes, PhD Thesis, Dept. of Automatic Control and Systems Engineering, University of Sheffield.

Keller, J.Y. and Darouach, M., 1997, "Optimal Two-Stage Kalman Filter in the Presence of Random Bias", *Automatica*, Vol. **33**, No.9, pp. 1745-1748

Kinsler, L.E., Frey, A.R., Coppens, A.B. and Sanders, J.V. 1982, *Fundamentals of Ultrasonics*, Wiley.

Kosko, B, 1992, Neural Networks and Fuzzy Systems, Prentice-Hall

Li, R. and Chu, D., 1997, "Stability of Kalman Filter for Time-Varying Systems with Correlated Noise", *International Journal of Adaptive Control and Signal Processing*, **11**, pp. 475-487.

Maryak, J.L, Spall, J.C. and Heydon, B.D., 1997, "Use of Kalman Filter for Inference in State-Space Models with Unknown Noise Distributions", *Proceedings of the American Control Conference,* Albuquerque, New Mexico, June 1997, pp. 2127-2132.

Massey, B.S., 1989, *Mechanics of Fluids*, Sixth Edition, Chapman & Hall

Mattiat (editor) O.E., 1971, Ultrasonic Transducer Materials, Plenum Press, New York

Peers N., (1992) Fluid Level Monitoring in Retail Food Industry, M.Sc. Thesis, University of Bradford.

Pena, D and Guttman I, 1988, Bayesian Approach to Robustifying the Kalman Filter, *Bayesian Analysis of Time Series and Dynamic Models,* ed. Spall,J.C. Marcel Dekker INC

Postlewaite, B, 1994, A Model-based Fuzzy Controller, TRANS. IchemE 72(a) 38-46

Ridgway, J.S., Henthorn, K.S., and Hull, J.B. 1997, "Controlling of Overfilling in Food Processing" *Ultrasound in Food Processing*, M.J. Povey, ed., Chapman & Hall

Rondeau, L., Ruelas, R., Levrat, L. and Lamotte, M., 1997, "A defuzzification method respecting the fuzzification", *Fuzzy Sets and Systems*, **86**, pp. 311-320.

Siouris, G.M., 1996, *An Engineering Approach to Optimal Control and Estimation Theory*, A Wiley- Interscience Publication, John Wiley & Sons, INC
Siouris, 1996

So, W.C. and Tse, C.K., 1996, "Derivation of Fuzzy Controller for DC/DC Converters using Neural Networks", *Journal of Electrical & Electronic, Australia – IEAust & IREE Society*, 16, No. 2, pp. 85-90.

Takagi, T and Sugeno, M, 1985, Fuzzy Identification of Systems and its application to modelling and control. IEEE TRANS. on Systems, Management and Cybernetics 15(I):116-132

Terrell, T, J, and Shark, L-K, 1996, Digital Signal Processing – A students guide, Macmillan Press, London.

Tonda, S.G. and Huignard, J-P, 1997, *Signal & Data Processing of Small Targets*, SPIE, Vol **3163**, pp. 204-213

Tufts D.W., Kirsteins, P.F. Swaszek, P.F., Efron, A.J., Melissinos, C.D., 1989, Detection of Signals in the Presence of Strong, Signal-like Interference and Impulse Noise. *Topics in Non-Gaussian Processing*, Springer-Verlag, New York

Verbruggen, H.B. and Brujin P.M., 1997, "Fuzzy Control and Conventional Control: What is (and can be) the real contribution of fuzzy systems", *Fuzzy Sets and Systems*, **90**, pp.151-160.

Whalley, R. and Mitchell, D. 1997, "The identification of engineering system parameters" *Proceedings of Institute of Mechanical Engineers*, **211**, part I, pp. 1-13.

Appendix A

C source code

A.1 Single Vessel

```
/* filler.c
Martyn Jeffries
August 1998
*/
#include "filler.h"

void main(void)
{
double h,               //measured liquid height
                        h_state,            //state estimate of height
                        speed,      //speed of moving liquid - estimate
                        h_pred,        //prediction of next fill point from model
                        freq;       //frequency shift in ultrasound signal
double T=0.0;

unsigned char vlv=0,                //valve control OPEN|CLOSED (1 or 0)
                              new_bottle=0;       //reset process for next
bottle (0 equals new bottle)
double N=0;

        randomize();
        puts("Single Bottle Fill Simulation");
        puts("Calculation Engine");
        puts("Copyright Martyn Jeffries 1998\n\n");
        if ((loaddata(&N)!=NULL) && (loadmodel()!=NULL))
        {
                do
                {
                        sensor(vlv,new_bottle,&h,&freq);
                        //update environment and measurements
                        filter(h,freq,vlv,new_bottle,&h_state,&speed,&h_pred);
                        //take measurements and make state estimates
                        control(h_state,h_pred,speed,new_bottle,&vlv);
                        //implement control actions for next iteration
                        exec(&new_bottle);
                        //executive/supervisory control
                        //prepare variables for saving
                        output.speed=speed;
                        output.h=h;
                        output.h_state=h_state;
                        output.vlv=vlv;
                        output.filling=new_bottle;
                        savedata(T);
                        //printf("%f\n",T);
                        T+=input.ts;
                        //increment global time
                }
                while(T<N);
                puts("\nProgram End");
        }
}
/*********************************************************************************/
/* sensor.c
Martyn Jeffries
August 1998*/

#include "sensor.h"

void sensor(unsigned char vlv,unsigned char new_bottle,double* h,double* freq)
{
```

```
static double h_actual;
double speed;

        if (new_bottle==0)
        {
                plant(vlv,&h_actual,&speed,0); //reset symbolic model
                noise(h_actual,speed,h,freq,0);       //reset noise model
        }
        else
        {
                plant(vlv,&h_actual,&speed,1);     //find actual fill height and
actual speed
                noise(h_actual,speed,h,freq,1);    //compensate for noise effects
        }
   output.h_actual=h_actual;
}
/*********************************************************************************/
//filter.c
//Kalman Filter Equations
//Martyn Jeffries
//August 1998

#include "filter.h"

/**************************************************************/
/*              Filter Control                 Martyn Jeffries               August
1998  */
/**************************************************************/
void filter(double h,double freq,unsigned char vlv,unsigned char
new_bottle,double* h_state,double* speed,double* h_pred)
{ double z=0;           //measurement
  static double
                     Pe,     //covariance estmate
                     Pu,     //covariance update
                     R,
                     Q,
                     K,              //kalman gain
                     xe,     //state estimate
                     xu,     //state update
                     xp;   //state prediction not used by kalman filter but by
controller

        if (new_bottle==0)
        {
                 R=input.R;
                 Q=input.Q;
                 Pe=input.Pe;
                 xe=input.xe;
       xp=input.xe;
                 xu=xe;
        }
        else
        {
                if (vlv==1)    //use bottle model
                {
                        pre_process(h,&z);                    //measurement
                        kalman_gain(Pe,R,&K);
                        state_update(K,xe,z,&xu);
                        cov_update(K,Pe,&Pu);
                        state_estimate(xu,&xe);
                        xp=xu;                  //next estimate if valve is closed
                        cov_estimate(xe,xu,Pu,Q,&Pe);
                }
                else          //assume filling has stopped
                {
                        pre_process(h,&z);                    //measurement
                        kalman_gain(Pe,R,&K);
                        state_update(K,xp,z,&xu);
                        cov_update(K,Pe,&Pu);
                        state_estimate(xu,&xe);               //prediction if valve
opens
                        xp=xu;
        //state estimate
                        Pe=Pu+Q;
```

```
                }

        }
        output.vol_e=xu;                          //volume estimate
        output.vol_a=z;                           //measured volume
        post_process(xu,h_state);                 //height estimate
        post_process(xe,h_pred);                  //estimate for next iteration
        *speed=34300*(freq-input.us_freq)/((double)2*input.us_freq);
        output.K=K;
        output.Pu=Pu;
}
/*************************************************************/
void pre_process(double x,double* v)
{
        int i=1;

        while(x > model.data[0][i]) i++;

        *v=(x-model.data[0][i-1])*(model.data[1][i]-model.data[1][i-
1])/(model.data[0][i]-model.data[0][i-1])+model.data[1][i-1];
}
/*************************************************************/
void post_process(double v,double* x)
{
        int i=1;

        while(v > model.data[1][i]) i++;

        *x=(v-model.data[1][i-1])*(model.data[0][i]-model.data[0][i-
1])/(model.data[1][i]-model.data[1][i-1])+model.data[0][i-1];


}
/*************************************************************/
/*State Estimate                    Martyn Jeffries              August 1998
*/
/*              X(k+1)=phi*X(k)
                                                   */
/*************************************************************/
void state_estimate(double xu,double* xe)
{
        phi(xu,xe);
}
/*************************************************************/
/*State Update                Martyn Jeffries              August 1998   */
/*              X(k)=X'(k)+K(k)(z(k)-H(k)X'(k))
                                    */
/*************************************************************/
void state_update(double K,double xe,double z,double* xu)
{
        *xu=xe+K*(z-xe);
}
/*************************************************************/
/*Covariance Estimate  Martyn Jeffries        August 1998   */
/*              P'(k+1)=phi^2*P(k)+Q(k)
                                              */
/*************************************************************/
void cov_estimate(double xe,double xu,double Pu,double Q,double* Pe)
{
        *Pe=((xe/xu)*(xe/xu))*Pu+Q;
}
/*************************************************************/
/*Covariance Update           Martyn Jeffries              August 1998   */
/*              P(k)=(I-K(k)H(k))P'(k)
                                      */
/*************************************************************/
void cov_update(double K,double Pe,double* Pu)
{
        *Pu=(1-K)*Pe;
}
/*************************************************************/
/*Kalman Gains          Martyn Jeffries        August 1998   */
/*              K(k)=P'(k)/(P'(k)-R(k))
                                              */
```

```
/**************************************************************/
void kalman_gain(double Pe,double R,double* K)
{
	*K=Pe/(Pe+R);
}
/*********************************************************************************/
/* plant.c
Martyn Jeffries
August 1998*/

#include "plant.h"
void plant(unsigned char vlv,double* h_actual,double* speed,char reset)//symbolic
bottle model
{
static double volume;
double flowrate;

	if (reset==0) //set filled volume to zero etc..
	{
		volume=0;
	*h_actual=0;
	}
	else		//find fill height at current time step
	{
		valve(&flowrate,vlv); //calculate flowrate from state of valve
		volume+=(double)flowrate*(double)input.ts;
		bottle(volume,h_actual,speed); //how full is bottle?
	}
	output.vol=volume;
}
/*********************************************************************************/
/* noise.c
Martyn Jeffries
August 1998*/

#include "noise.h"

void noise(double h_actual,double speed,double* h,double* freq,char reset)
{double x,v1,v2,s=0;
double a,b,bb,c,u,v,w,y,xg,z;
unsigned char accept;

	if(reset==0)	//reset for time-correlated noise
	{
			if (input.gamma!=1)
			{
			//normally distributed noise, std dev = input.std
				do
				{
					v1=2*((double)rand()/RAND_MAX )-1;
					v2=2*((double)rand()/RAND_MAX )-1;
					s=v1*v1+v2*v2;
				}
				while (s>=1 || s==0);
				x=v2*sqrt(-2*log(s)/s);

				*h=h_actual+x*input.std;
	}
			else
			{
			//gamma distribution
				a=8;
				b=-1;
				bb=a-1;
				c=3*a-0.75;
				accept=0;
				do
				{
					u=(double)rand()/RAND_MAX;
					v=(double)rand()/RAND_MAX;
					w=u*(1-u);
					y=sqrt(c/w)*(u-0.5);
					xg=bb+y;
					if (xg >= 0)
```

```
                                        {
                                                z=64*w*w*w*v*v;
                                                if (z <= (2*y*y)/xg) accept=1;
                                                if (accept==0)
                                                {
                                                        if (log10(z) <=
2*(bb*log10(xg/bb)-y)) accept=1;
                                                }
                                        }
                                }
                                while (accept==0);
                                *h=h_actual+(((xg*b)+10)*0.1-0.1)*input.std;
                        }
//              *h=h_actual+((((double)(rand() % 1000))/1000)-0.5)*input.std;
                *freq=input.us_freq;
        }
        else
        {

//              *h=h_actual+((((double)(rand() % 1000))/1000)-0.5)*input.var;
//white noise (width = var)

                //measurement data loss
                if (((double)(rand() % 10000)/10000) >= input.loss)
                {
                        if (input.gamma!=1)
                        {//normally distributed noise, std dev = input.std
                                do
                                {
                                        v1=2*((double)rand()/RAND_MAX )-1;
                                        v2=2*((double)rand()/RAND_MAX )-1;
                                        s=v1*v1+v2*v2;
                                }
                                while (s>=1 || s==0);
                                x=v2*sqrt(-2*log(s)/s);

                                *h=h_actual+x*input.std;
                 }
                 else
                 {//gamma distribution
                                a=8;
                                b=-1;
                                bb=a-1;
                                c=3*a-0.75;
                                accept=0;
                                do
                                {
                                        u=(double)rand()/RAND_MAX;
                                        v=(double)rand()/RAND_MAX;
                                        w=u*(1-u);
                                        y=sqrt(c/w)*(u-0.5);
                                        xg=bb+y;
                                        if (xg >= 0)
                                        {
                                                z=64*w*w*w*v*v;
                                                if (z <= (2*y*y)/xg) accept=1;
                                                if (accept==0)
                                                {
                                                        if (log10(z) <=
2*(bb*log10(xg/bb)-y)) accept=1;
                                                }
                                        }
                                }
                                while (accept==0);
                                *h=h_actual+(((xg*b)+10)*0.1-0.1)*input.std;
        }

                        //doppler shift. - no noise

        *freq=(double)input.us_freq+((2*(speed/100)*input.us_freq)/(double)343);
                }
                 else // no data
                 {
                        *h=0;           //find better value for this
```

```
                        *freq=input.us_freq;    //using zero cause div by 0 errors
later

                }
        }
}
/*********************************************************************************/
/* symb.c
Martyn Jeffries
August 1998*/

#include "symb.h"


void bottle(double volume,double* h_actual,double* speed)
{double last_h;
 int i=1;

        last_h=*h_actual;

        while(volume > model.data[1][i]) i++;
        *h_actual=(volume-model.data[1][i-1])*(model.data[0][i]-model.data[0][i-
1])/
                                        (model.data[1][i]-model.data[1][i-
1])+model.data[0][i-1];

        *speed=(*h_actual-last_h)/input.ts;
}
/*********************************************************************************/
/* valve.c
MArtyn Jeffries
August 1998*/

#include "valve.h"

void valve(double* flowrate,unsigned char vlv)
{
static float opening;

          if(vlv==0) //valve closing
          {
                  if (opening > 0.0) opening-=input.vc_speed*input.ts;
                  //characteristic - linear
                  if (opening < 0.0) opening=0.0;
          }
          else        //valve opening
          {
                  if (opening < 1.0) opening+=input.vo_speed*input.ts;
                  //characteristic - linear
                  if (opening > 1.0) opening=1.0;
          }
          *flowrate=(double)opening*(double)input.max_flowrate;     //linear model

}
/*******************************************************************************/
/* numr.c
Martyn Jeffries
August 1998*/

#include "numr.h"

void phi(double xu,double* xe)
{       // volume model
        *xe=xu+input.flow_rate*input.ts;
}
/* control.c
Martyn Jeffries
August 1998*/

#include "control.h"

void control(double h_state,double h_pred,double speed,unsigned char
new_bottle,unsigned char* vlv)
{
```

```
        if(new_bottle==0) //reset valve
        {
                *vlv=0;
        }
        else
        {
                //close valve if fill level will exceed or equal filling set point
                //in next time step, else open.
                //if(h_state>input.f_point) *vlv=0;
                if (fabs(input.f_point - h_state) < fabs(input.f_point - h_pred))
*vlv=0;
                //if (h_state+(speed*input.ts)>=input.f_point) *vlv=0;
                else *vlv=1;

        }
}

void exec(unsigned char* new_bottle)//overall control for plant - linked to multi
bottle setup
{
static double tfill; //time elapsed in individual bottle fill

        tfill+=input.ts;                                        //increment filling
time

        if (*new_bottle==0 && tfill > 1.0)      //wait for empty bottle for 1 second
        {
                 *new_bottle=1;         //start filling cycle
                 tfill=0.0;               //reset fill time for fresh bottle
        }

        if (tfill >= input.tbottle)                             //end filling cycle
        {
                *new_bottle=0;
                tfill=0.0;
        }

}
/********************************************************************************/
//loaddata.c
//load initial model data
//Martyn Jeffries
//august 1998

#include "loaddata.h"

char loaddata(double *N)
{
FILE *fptr;
//int n,i; //array size

        if((fptr=fopen("c:/user/developm/1bottle/data/input.txt","r")) == NULL)
        {
                fprintf(stderr,"No input data!\n");
                *N=0;
                return(NULL);
        }
        else
        {
                puts("Loading Input Parameters");
                //printf("Single Bottle Simulation\nReading from file...\n");
                find("RUN_TIME",fptr);
                read_float(fptr,N);
                find("TIME_STEP",fptr);
                read_float(fptr,&input.ts);
                //maximum fill time
                find("TIME_B",fptr);
                read_float(fptr,&input.tbottle);

                //filling set point
                find("FILL_P",fptr);
                read_float(fptr,&input.f_point);

                //plant max flow rate
```

```
		find("MAXI_F",fptr);
		read_float(fptr,&input.max_flowrate);

		//ultrasound frequency
		find("US_FREQ",fptr);
		read_integer(fptr,&input.us_freq);

		//use gamma?
		find("GAMMA",fptr);
		read_integer(fptr,&input.gamma);

		//actual noise variance
		find("NOISY",fptr);
		read_float(fptr,&input.std);

		//percentage data loss
		find("LOSS",fptr);
		read_float(fptr,&input.loss);

		//valve closing speed
		find("CLOSE",fptr);
		read_float(fptr,&input.vc_speed);

		//valve opening speed
		find("OPEN",fptr);
		read_float(fptr,&input.vo_speed);

		//measurement noise variance
		find("MEAS_R",fptr);
		read_float(fptr,&input.R);

		//system noise variance
		find("SYST_Q",fptr);
		read_float(fptr,&input.Q);

		//initial estimate of state varaibles
		find("INIT_X",fptr);
		read_float(fptr,&input.xe);

		//initial estimate of error covariance matrix
		find("INIT_P",fptr);
		read_float(fptr,&input.Pe);

		//flow rate estimate
		find("FLOW_EST",fptr);
		read_float(fptr,&input.flow_rate);

		fclose(fptr);

		//clear output file
		fptr=fopen("c:/user/developm/1bottle/data/output.txt","w");
		fclose(fptr);
		return(1);
	}
}
/*******************************************************************/
char loadmodel(void)
{
FILE *fptr;
int n;

	if((fptr=fopen("c:/user/developm/1bottle/data/model.txt","r")) == NULL)
	{
		fprintf(stderr,"No model data!\n");
		return(NULL);
	}
	else
	{
		puts("Loading Model Data");
		find("ELEM",fptr);
		read_integer(fptr,&model.element);
		find("DATA",fptr);
		for(n=0;n<model.element;n++)
		{
```

```
                read_two_floats(fptr,&model.data[0][n],&model.data[1][n]);
            }
            fclose(fptr);
            return(1);
        }
}

/*****************************************************************/
int find(char *string,FILE *fptr)
{
 char buffer[BUF_SIZE];

        do
        {
                fgets(buffer,BUF_SIZE,fptr);
        }
        while((!feof(fptr)) && (strncmp(string,buffer,strlen(string))));
        return(feof(fptr));
}
/************************************************************************/
void read_string(FILE *fptr,char *destin,int length)
{
        char buffer[BUF_SIZE];

        fgets(buffer,BUF_SIZE,fptr);
        strncpy(destin,buffer,length);
}
/************************************************************************/
void read_integer(FILE *fptr,int *value)
{
        char buffer[BUF_SIZE];

        fgets(buffer,BUF_SIZE,fptr);
        sscanf(buffer,"%i",value);
}
/************************************************************************/
void read_float(FILE *fptr,double *value)
{
        char buffer[BUF_SIZE];

        fgets(buffer,BUF_SIZE,fptr);
        sscanf(buffer,"%lf",value);
}
/*************************************************************************/
void read_two_floats(FILE *fptr,double *value1,double *value2)
{
        char buffer[BUF_SIZE];

        fgets(buffer,BUF_SIZE,fptr);
        sscanf(buffer,"%lf %lf",value1,value2);
}
/************************************************************************/
//savedata.c
//Save model data
//Martyn Jeffries
//August 1998

#include "savedata.h"


void savedata(double T)
{
FILE *fptr;
char buffer[250];
fptr=fopen("c:/user/developm/1bottle/data/output.txt","a");
sprintf(buffer,"%f\t%f\t%f\t%f\t%f\t%f\t%d\t%d\t%f\t%f\t%f\t%f\n",T,output.h_actua
l,output.h,output.h_state,


        output.K,output.Pu,output.vlv,output.filling,output.vol,output.vol_e,

                output.vol_a,output.speed);
fputs(buffer,fptr);
fclose(fptr);
```

```
}
/***********************************************************************************/
```

A.2 Multiple Vessel

```
/* filler.c
Martyn Jeffries
January 1999
*/
#include "filler.h"

void main(void)
{
                        double h[64],  //measured liquid height
                        h_state[64],   //state estimate of height
                        h_actual[64],  //actual fill height of liquid
                        speed[64],     //speed of moving liquid - estimate
                        h_pred[64],    //prediction of next fill point from model
                        freq[64],      //frequency shift in ultrasound signal
                        dQm[64],       //rate of change of measured flow
                        Qt,            //overall available flow rate
                        Qtc,           //calculated overall flow
                        Qe[64],        //estimated flow rate
                        Qa[64],        //available flow rate
                        dQi[64],       //rate of change of individual flow rate
                        dQt;           //overall rate of change flow rate
                        double T=0.0;         //time
                        unsigned char  vlv[64],//valve control OPEN|CLOSED (1 or 0)
                                       new_bottle[64];//reset process for next
bottle (0 equals new bottle)
                        double N=0;    //overall end count
                        int i;         //loop variable

        randomize();
        puts("Multi-Bottle Fill Simulation");
        puts("Calculation Engine");
        puts("Copyright Martyn Jeffries 1998,1999\n\n");
        if ((loaddata(&N)!=NULL) && (loadmodel()!=NULL) && (fuzzy_data()!=NULL))
        {
                puts("\nProcessing...");

                exec(0,0,0); //initialise process
                Qt=input.max_flowrate;
                Qtc=input.max_flowrate;
                for(i=0;i<input.nv;i++)        //set initial values
                {
                        Qe[i]=Qt;
                        Qa[i]=Qt;
                }
                do
                {
                        Qt-=sin(2*T)*2;              //modulate flow rate
                        for(i=0;i<input.nv;i++) //iterate through all valves
                        {
                                //update environment and measurements
                                sensor(vlv[i],new_bottle[i],Qt,&h[i],
                                        &freq[i],&h_actual[i],i,T);
                                speed[i]=34300*(freq[i]-input.us_freq)
                                                /((double)2*input.us_freq);
                                if (vlv[i]==0) Qe[i]=0;
                                else Qe[i]=Qa[i];

        dQm[i]=(3.14159265359*input.b_radius*input.b_radius*speed[i])-Qe[i];
                                        //speed to change of flow
                        }
                        consolidate(h_state,dQm,&dQt);//fuzzy flow consolidator
                        Qtc+=dQt;                     //update overall flow
                        for(i=0;i<input.nv;i++)
                        {
                                measure(h_state[i],dQm[i],dQt,&dQi[i]);
                                //individual fuzzy flow calculation
                                Qe[i]+=dQi[i];
                                Qa[i]+=dQi[i];
                                filter(h[i],vlv[i],Qa[i],new_bottle[i],
                                        &h_state[i],&h_pred[i],i);
```

```
                                        //take measurements and make state estimates
                                        control(h_state[i],h_pred[i],new_bottle[i],&vlv[i]);
                                        //implement control actions for next iteration
                                        exec(&new_bottle[i],1,i);
                                        //individual executive/supervisory control
                                        //copy variables for saving
                                        output.h_actual[i]=h_actual[i];
                                        output.dQi[i]=dQi[i];
                                        output.dQm[i]=dQm[i];
                                        output.dQt=dQt;
                                        output.Qe[i]=Qe[i];
                                        output.h[i]=h[i];
                                        output.h_state[i]=h_state[i];
                                        output.vlv[i]=vlv[i];
                                        output.filling[i]=new_bottle[i];
                                        //then save
                                        savedata(T,i);
                                }
                                //further processing for overall flow characteristcs
                                //condition monitoring

                                T+=input.ts;  //increment global time
                        }
                        while(T<N);
                        puts("\nProgram End");
                }
}
/*******************************************************************************/
/* fuzzy.c
Martyn Jeffries
October 1998*/

#include "fuzzy.h"


void consolidate(double h[64],double dQm[64],double* dQt)
{char i;
 double mem_table[64];
 double sumtop=0.0,sumbot=0.0;
 for(i=0;i<input.nv;i++)
 {
        mem_table[i]=get_membership(&ud[0].set[1],h[i]);  //membership of set BODY
        sumtop+=mem_table[i]*dQm[i];
        sumbot+=mem_table[i];
 }
 if (sumbot>0) *dQt=sumtop/(double)sumbot;
 else *dQt=0;

}
void measure(double h,double dQm,double dQt,double* dQi)
{
        double mem_table[3];
        mem_table[0]=get_membership(&ud[0].set[0],h);
        mem_table[1]=get_membership(&ud[0].set[1],h);
        mem_table[2]=get_membership(&ud[0].set[2],h);

        *dQi=(mem_table[1]*dQm+mem_table[0]*dQt+mem_table[2]*dQt)/(mem_table[0]+mem
_table[1]+mem_table[2]);

}

double get_membership(fuzzy_set *fs,double input_value)
{
         double membership=0.0;

//       printf("input value %f\n",input_value);
         if(input_value <= fs->value[0])
         {
                membership=fs->mem[0];
                return(membership);
         }
         if(input_value >= fs->value[3])
         {
                membership=fs->mem[3];
```

```
		return(membership);
	}
	if((input_value > fs->value[0])&&(input_value < fs->value[1]))
	{
		membership=((input_value-fs->value[0])*((fs->mem[1]-fs->mem[0])
							/(fs->value[1]-fs-
>value[0])))+fs->mem[0];
		return(membership);
	}
	if((input_value >= fs->value[1])&&(input_value < fs->value[2]))
	{
		membership=((input_value-fs->value[1])*((fs->mem[2]-fs->mem[1])
							/(fs->value[2]-fs-
>value[1])))+fs->mem[1];
		return(membership);
	}
	if((input_value >= fs->value[2])&&(input_value < fs->value[3]))
	{
		membership=((input_value-fs->value[2])*((fs->mem[3]-fs->mem[2])
							/(fs->value[3]-fs-
>value[2])))+fs->mem[2];
		return(membership);
	}
	return(0);
}
/*****************************************************************************/
//filter.c
//Kalman Filter Equations
//Martyn Jeffries
//August 1998

#include "filter.h"

/***************************************************************/
/*		Filter Control			Martyn Jeffries			August
1998  */
/***************************************************************/
void filter(double h,unsigned char vlv,double Qe,unsigned char new_bottle,
				double* h_state,double* h_pred,int i)
{ double z=0;		//measurement
  static double
			Pe[64],		//covariance estmate
			Pu[64], //covariance update
			R[64],
			Q[64],
			K[64],		//kalman gain
			xe[64], //state estimate
			xu[64], //state update
			xp[64];	//state prediction not used by kalman filter but
by controller

	if (new_bottle==0)
	{
		R[i]=input.R;
		Q[i]=input.Q;
		Pe[i]=input.Pe;
		xe[i]=input.xe;
		xp[i]=input.xe;
		xu[i]=xe[i];
	}
	else
	{
		if (vlv==1)	//use bottle model
		{
			pre_process(h,&z);			//measurement
			kalman_gain(Pe[i],R[i],&K[i]);
			state_update(K[i],xe[i],z,&xu[i]);
			cov_update(K[i],Pe[i],&Pu[i]);
			state_estimate(xu[i],&xe[i],Qe);
			xp[i]=xu[i];			//next estimate if valve is
closed
			cov_estimate(xe[i],xu[i],Pu[i],Q[i],&Pe[i]);
		}
		else		//assume filling has stopped
```

```
                {
                        pre_process(h,&z);                      //measurement
                        kalman_gain(Pe[i],R[i],&K[i]);
                        state_update(K[i],xp[i],z,&xu[i]);
                        cov_update(K[i],Pe[i],&Pu[i]);
                        state_estimate(xu[i],&xe[i],Qe);
        //prediction if valve opens
                        xp[i]=xu[i];
        //state estimate
                        Pe[i]=Pu[i]+Q[i];
                }

        }
        output.vol_e[i]=xu[i];                  //volume estimate
        output.vol_a[i]=z;                                              //measured
volume
        post_process(xu[i],h_state);                    //height estimate
        post_process(xe[i],h_pred);             //estimate for next iteration
        output.K[i]=K[i];
        output.Pu[i]=Pu[i];
}
/****************************************************************/
void pre_process(double x,double* v)
{
        int i=1;

        while(x > model.data[0][i]) i++;

        *v=(x-model.data[0][i-1])*(model.data[1][i]-model.data[1][i-
1])/(model.data[0][i]-model.data[0][i-1])+model.data[1][i-1];
}
/****************************************************************/
void post_process(double v,double* x)
{
        int i=1;

        while(v > model.data[1][i]) i++;

        *x=(v-model.data[1][i-1])*(model.data[0][i]-model.data[0][i-
1])/(model.data[1][i]-model.data[1][i-1])+model.data[0][i-1];


}
/****************************************************************/
/*              State Estimate                  Martyn Jeffries                 August
1998    */
/*              X(k+1)=phi*X(k)
                                                        */
/****************************************************************/
void state_estimate(double xu,double* xe,double Qe)
{
        phi(xu,xe,Qe);
}
/****************************************************************/
/*              State Update                    Martyn Jeffries                 August
1998  */
/*              X(k)=X'(k)+K(k)(z(k)-H(k)X'(k))
                                */
/****************************************************************/
void state_update(double K,double xe,double z,double* xu)
{
        *xu=xe+K*(z-xe);
}
/****************************************************************/
/*              Covariance Estimate     Martyn Jeffries                 August 1998   */
/*              P'(k+1)=phi^2*P(k)+Q(k)
                                                */
/****************************************************************/
void cov_estimate(double xe,double xu,double Pu,double Q,double* Pe)
{
        *Pe=((xe/xu)*(xe/xu))*Pu+Q;
}
/****************************************************************/
```

```
/*              Covariance Update              Martyn Jeffries               August
1998  */
/*              P(k)=(I-K(k)H(k))P'(k)
                                *
/**************************************************************/
void cov_update(double K,double Pe,double* Pu)
{
        *Pu=(1-K)*Pe;
}
/**************************************************************/
/*              Kalman Gains                   Martyn Jeffries               August
1998  */
/*              K(k)=P'(k)/(P'(k)-R(k))
                                        */
/**************************************************************/
void kalman_gain(double Pe,double R,double* K)
{
        *K=Pe/(Pe+R);
}
/********************** this is the end ***********************/
/* control.c
Martyn Jeffries
August 1998*/

#include "control.h"

void control(double h_state,double h_pred,unsigned char new_bottle,unsigned char*
vlv)
{
        if(new_bottle==0) //reset valve
        {
                *vlv=0;
        }
        else
        {
                //close valve if fill level will exceed or equal filling set point
                //in next time step, else open.
                //if(h_state>input.f_point) *vlv=0;
                if (fabs(input.f_point - h_state) < fabs(input.f_point - h_pred))
*vlv=0;
                //if (h_state+(speed*input.ts)>=input.f_point) *vlv=0;
                else *vlv=1;

        }
}

void exec(unsigned char* new_bottle,unsigned char init,unsigned char i)//overall
control for plant - linked to multi bottle setup
{
static double tfill[64]; //time elapsed in individual bottle fill
unsigned char n;

        if (init == 0)  //initialise timing parameters
        {
                for (n=0;n<input.nv;n++)
                {
                        tfill[n]=-((input.tbottle+1)*n)/(double)input.nv;
                }

        }
        else
        {
                tfill[i]+=input.ts;
        //increment filling time

                if (*new_bottle==0 && tfill[i] > 1.0)  //wait for empty bottle for
1 second
                {
                         *new_bottle=1;                                  //start
filling cycle
                         tfill[i]=0.0;                               //reset fill
time for fresh bottle
                }
```

```
		if (tfill[i] >= input.tbottle)		//end filling cycle
		{
			*new_bottle=0;
			tfill[i]=0.0;
		}
	}
}
/* plant.c
Martyn Jeffries
August 1998*/

#include "plant.h"
void plant(unsigned char vlv,double Qa,double* h_actual,double* speed,int i,double
T,char reset)//symbolic bottle model
{
static double volume[64];
double flowrate;

	if (reset==0) //set filled volume to zero etc..
	{
		flowrate=0;
		volume[i]=0;
		*h_actual=0;
	}
	else		//find fill height at current time step
	{
		valve(&flowrate,Qa,vlv,i,T); //calculate flowrate from state of
valve
		volume[i]+=(double)flowrate*(double)input.ts;
		bottle(volume[i],h_actual,speed); //how full is bottle?
	}
	output.vol[i]=volume[i];
   output.Qa[i]=flowrate;
}
/* noise.c
Martyn Jeffries
August 1998*/

#include "noise.h"

void noise(double h_actual,double speed,double* h,double* freq,char reset)
{double x,v1,v2,s=0;
double a,b,bb,c,u,v,w,y,xg,z;
unsigned char accept;

	if(reset==0)	//reset for time-correlated noise
	{
			if (input.gamma!=1)
			{
			//normally distributed noise, std dev = input.std
				do
				{
					v1=2*((double)rand()/RAND_MAX )-1;
					v2=2*((double)rand()/RAND_MAX )-1;
					s=v1*v1+v2*v2;
				}
				while (s>=1 || s==0);
				x=v2*sqrt(-2*log(s)/s);

				*h=h_actual+x*input.std;
	 }
			else
			{
			//gamma distribution
				a=8;
				b=-1;
				bb=a-1;
				c=3*a-0.75;
				accept=0;
				do
				{
					u=(double)rand()/RAND_MAX;
					v=(double)rand()/RAND_MAX;
					w=u*(1-u);
```

```
                                        y=sqrt(c/w)*(u-0.5);
                                        xg=bb+y;
                                        if (xg >= 0)
                                        {
                                                z=64*w*w*w*v*v;
                                                if (z <= (2*y*y)/xg) accept=1;
                                                if (accept==0)
                                                {
                                                        if (log10(z) <=
2*(bb*log10(xg/bb)-y)) accept=1;
                                                }
                                        }
                                }
                                while (accept==0);
                                *h=h_actual+(((xg*b)+10)*0.1-0.1)*input.std;
                        }
 //             *h=h_actual+((((double)(rand() % 1000))/1000)-0.5)*input.std;
                *freq=input.us_freq;
        }
        else
        {

//              *h=h_actual+((((double)(rand() % 1000))/1000)-0.5)*input.var;
//white noise (width = var)

                //measurement data loss
                if (((double)(rand() % 10000)/10000) >= input.loss)
                {
                        if (input.gamma!=1)
                        {//normally distributed noise, std dev = input.std
                                do
                                {
                                        v1=2*((double)rand()/RAND_MAX )-1;
                                        v2=2*((double)rand()/RAND_MAX )-1;
                                        s=v1*v1+v2*v2;
                                }
                                while (s>=1 || s==0);
                                x=v2*sqrt(-2*log(s)/s);

                                *h=h_actual+x*input.std;
                 }
                 else
                 {//gamma distribution
                                a=8;
                                b=-1;
                                bb=a-1;
                                c=3*a-0.75;
                                accept=0;
                                do
                                {
                                        u=(double)rand()/RAND_MAX;
                                        v=(double)rand()/RAND_MAX;
                                        w=u*(1-u);
                                        y=sqrt(c/w)*(u-0.5);
                                        xg=bb+y;
                                        if (xg >= 0)
                                        {
                                                z=64*w*w*w*v*v;
                                                if (z <= (2*y*y)/xg) accept=1;
                                                if (accept==0)
                                                {
                                                        if (log10(z) <=
2*(bb*log10(xg/bb)-y)) accept=1;
                                                }
                                        }
                                }
                                while (accept==0);
                                *h=h_actual+(((xg*b)+10)*0.1-0.1)*input.std;
        }

                        //doppler shift. - no noise

        *freq=(double)input.us_freq+((2*(speed/100)*input.us_freq)/(double)343);
                }
```

```
                else // no data
                {
                        *h=0;           //find better value for this
                        *freq=input.us_freq;     //using zero cause div by 0 errors
later

                }
        }
}
/* sensor.c
Martyn Jeffries
August 1998*/

#include "sensor.h"

void sensor(unsigned char vlv,unsigned char new_bottle,double Qa,double* h,double*
freq,double* h_actual,int i,double T)
{
double speed;

        if (new_bottle == 0)
        {
                plant(vlv,Qa,h_actual,&speed,i,T,0); //reset symbolic model
                noise(*h_actual,speed,h,freq,0);      //reset noise model
        }
        else
        {
                plant(vlv,Qa,h_actual,&speed,i,T,1);      //find actual fill height
and actual speed
                noise(*h_actual,speed,h,freq,1);     //compensate for noise effects
        }

}
/* valve.c
MArtyn Jeffries
August 1998*/

#include "valve.h"

void valve(double* flowrate,double Qa,unsigned char vlv,int i,double T)
{
static float opening[64];

          if(vlv==0) //valve closing
          {
                  if (opening[i] > 0.0) opening[i]-=input.vc_speed*input.ts;
//characteristic - linear
                  if (opening[i] < 0.0) opening[i]=0.0;
          }
          else         //valve opening
          {
                  if (opening[i] < 1.0) opening[i]+=input.vo_speed*input.ts;
//characteristic - linear
                  if (opening[i] > 1.0) opening[i]=1.0;
          }
          if ((i==2) && (T>9) && (T<22))
        *flowrate=(double)opening[i]*(double)Qa*0.5; //linear model
          else
          {
                  *flowrate=(double)opening[i]*(double)Qa;    //linear model
          }
}
/* numr.c
Martyn Jeffries
August 1998*/

#include "numr.h"

void phi(double xu,double* xe,double Qe)
{       // volume model
        //*xe=xu+input.flow_rate*input.ts;
        *xe=xu+Qe*input.ts;
}
/* symb.c
```

```
Martyn Jeffries
August 1998*/

#include "symb.h"


void bottle(double volume,double* h_actual,double* speed)
{double last_h;
 int i=1;

        last_h=*h_actual;

        while(volume > model.data[1][i]) i++;
        *h_actual=(volume-model.data[1][i-1])*(model.data[0][i]-model.data[0][i-
1])/
                                                (model.data[1][i]-model.data[1][i-
1])+model.data[0][i-1];

        *speed=(*h_actual-last_h)/input.ts;
}
//loaddata.c
//load initial model data
//Martyn Jeffries
//august 1998

#include "loaddata.h"

char loaddata(double *N)
{
FILE *fptr;
int i;
char buffer[250];
//int n,i; //array size

        if((fptr=fopen("c:/user/developm/mbottle/data/input.txt","r")) == NULL)
        {
                fprintf(stderr,"No input data!\n");
                *N=0;
                return(NULL);
        }
        else
        {
                puts("Loading Input Parameters");
                //printf("Single Bottle Simulation\nReading from file...\n");
                find("RUN_TIME",fptr);
                read_float(fptr,N);
                find("TIME_STEP",fptr);
                read_float(fptr,&input.ts);
                //number of filling heads max = 64
                find("NUMB_HEAD",fptr);
                read_integer(fptr,&input.nv);
                //maximum fill time
                find("TIME_B",fptr);
                read_float(fptr,&input.tbottle);

                //filling set point
                find("FILL_P",fptr);
                read_float(fptr,&input.f_point);

                //plant max flow rate
                find("MAXI_F",fptr);
                read_float(fptr,&input.max_flowrate);

                //bottle radius
                find("RADI_B",fptr);
      read_float(fptr,&input.b_radius);

                //ultrasound frequency
                find("US_FREQ",fptr);
                read_integer(fptr,&input.us_freq);

                //gamma distribution
                find("GAMMA",fptr);
                read_integer(fptr,&input.gamma);
```

```
		//actual noise variance
		find("NOISY",fptr);
		read_float(fptr,&input.std);

		//percentage data loss
		find("LOSS",fptr);
		read_float(fptr,&input.loss);

		//valve closing speed
		find("CLOSE",fptr);
		read_float(fptr,&input.vc_speed);

		//valve opening speed
		find("OPEN",fptr);
		read_float(fptr,&input.vo_speed);

		//measurement noise variance
		find("MEAS_R",fptr);
		read_float(fptr,&input.R);

		//system noise variance
		find("SYST_Q",fptr);
		read_float(fptr,&input.Q);

		//initial estimate of state varaibles
		find("INIT_X",fptr);
		read_float(fptr,&input.xe);

		//initial estimate of error covariance matrix
		find("INIT_P",fptr);
		read_float(fptr,&input.Pe);

		//flow rate estimate
		find("FLOW_EST",fptr);
		read_float(fptr,&input.flow_rate);

		fclose(fptr);

		//clear output files
		for (i=1;i<=input.nv;i++)
		{
	sprintf(buffer,"c:/user/developm/mbottle/data/output%d.txt",i);
			fptr=fopen(buffer,"w");
			fclose(fptr);
		}
	//	fptr=fopen("c:/user/developm/mbottle/data/output.txt","w");
	//	fclose(fptr);
		return(1);
	}
}
/*****************************************************************/
char loadmodel(void)
{
FILE *fptr;
int n;

	if((fptr=fopen("c:/user/developm/mbottle/data/model.txt","r")) == NULL)
	{
		fprintf(stderr,"No model data!\n");
		return(NULL);
	}
	else
	{
		puts("Loading Model Data");
		find("ELEM",fptr);
		read_integer(fptr,&model.element);
		find("DATA",fptr);
		for(n=0;n<model.element;n++)
		{
			read_two_floats(fptr,&model.data[0][n],&model.data[1][n]);
		}
		fclose(fptr);
```

```
                return(1);
        }
}
/*****************************************************************/
char fuzzy_data(void)
{
        FILE *fptr;
        int no_of_uni,i,k,j;

        if((fptr=fopen("c:/user/developm/mbottle/data/input.fuz","r")) == NULL)
        {
                fprintf(stderr,"No fuzzy data file!\n");
                return(NULL);
        }
        else
        {
                puts("Reading fuzzy sets from file...");
                if(find("FZZY",fptr))
                {
                        fprintf(stderr,"File Error:wrong format\n");
                        return(NULL);
                }
                else
                {
                        read_integer(fptr,&no_of_uni);
                        for(i=0;i<no_of_uni;i++)
                        {
                                find("UNIV",fptr);
                                read_string(fptr,ud[i].var_name,16);
                                fprintf(stderr,"Variable Name %s\n
Sets:",ud[i].var_name);
                                read_integer(fptr,&ud[i].num_sets);
                                for(j=0;j<ud[i].num_sets;j++)
                                {
                                        find("SET",fptr);
                                        read_string(fptr,ud[i].set[j].name,16);
                                        fprintf(stderr," %s",ud[i].set[j].name);
                                        for(k=0;k<4;k++)
                                        {

        read_two_floats(fptr,&ud[i].set[j].value[k],&ud[i].set[j].mem[k]);
                                        }
                                }
                        }
                }
                fclose(fptr);
                return(1);
        }
}
/*****************************************************************/
/*                              Core File reading functions
                                  */
/*****************************************************************/
int find(char *string,FILE *fptr)
{
 char buffer[BUF_SIZE];

        do
        {
                fgets(buffer,BUF_SIZE,fptr);
        }
        while((!feof(fptr)) && (strncmp(string,buffer,strlen(string))));
        return(feof(fptr));
}
/************************************************************************/
void read_string(FILE *fptr,char *destin,int length)
{
        char buffer[BUF_SIZE];

        fgets(buffer,BUF_SIZE,fptr);
        strncpy(destin,buffer,length);
}
/************************************************************************/
void read_integer(FILE *fptr,int *value)
```

```
{
        char buffer[BUF_SIZE];

        fgets(buffer,BUF_SIZE,fptr);
        sscanf(buffer,"%i",value);
}
/************************************************************************/
void read_float(FILE *fptr,double *value)
{
        char buffer[BUF_SIZE];

        fgets(buffer,BUF_SIZE,fptr);
        sscanf(buffer,"%lf",value);
}
/****************************************************************************/
void read_two_floats(FILE *fptr,double *value1,double *value2)
{
        char buffer[BUF_SIZE];

        fgets(buffer,BUF_SIZE,fptr);
        sscanf(buffer,"%lf %lf",value1,value2);
}
/**************************************************************************/
//savedata.c
//Save model data
//Martyn Jeffries
//August 1998

#include "savedata.h"


void savedata(double T,int i)
{
FILE *fptr;
char buffer[250];


        sprintf(buffer,"c:/user/developm/mbottle/data/output%d.txt",i+1);
        fptr=fopen(buffer,"a");
        sprintf(buffer,"%f\t%f\t%f\t%f\t%f\t%f\t%d\t%d\t%f\t%f\t%f\t%f\t%f\t%f\t%f\
t%f\n",

        T,output.h_actual[i],output.h[i],output.h_state[i],

        output.K[i],output.Pu[i],output.vlv[i],output.filling[i],output.vol[i],

        output.vol_e[i],output.vol_a[i],output.dQm[i],output.dQi[i],output.dQt,
                                                output.Qa[i],output.Qe[i]);
        fputs(buffer,fptr);
        fclose(fptr);

}
/****************************************************************************/
```

Appendix B

Publications

A novel approach to reduce packaging process downtime
D.H. Plantenberg, E. Lai, M. Jeffries – AMPT '97, Portugal

A fuzzy approach to the condition monitoring of a packaging plant
M. Jeffries, E. Lai, D.H. Plantenberg, J.B. Hull– AMME '97, Poland – extended abstract

A fuzzy approach to the condition monitoring of a packaging plant
M. Jeffries, E. Lai, D.H. Plantenberg, J.B. Hull– Special Edition of Elsevier's Journal of Material Processing Technology, pub. 1999

Real-time implementation of a fuzzy condition monitoring system for predicting process breakdown
M. Jeffries, E. Lai, D.H. Plantenberg, J.B. Hull – EIS '98, Tenerife, Spain

An extended Kalman Filter to assimilate stochastic factors of ultrasound measurement in container filling operations
M. Jeffries, E. Lai, J.B. Hull– MID '98, Nottingham, UK

A new approach to process control for bottling plant.
M. Jeffries, E. Lai, J. B. Hull – submitted to Special Edition of Elsevier's Journal of Material Processing Technology, pub. 1999

A NOVEL APPROACH TO REDUCE PACKAGING PROCESS DOWN-TIME

D.H. Plantenberg, E. Lai and M. Jeffries
The Nottingham Trent University, Department of Mechanical and Manufacturing Engineering,
Nottingham NG1 4BU, United Kingdom. Fax: +44 115 948 6506, email: e.lai@domme.ntu.ac.uk

Abstract: Semi-automated systems are widely used for packaging products, which require the support of experienced mechanics to perform tasks such as tool changes and machine adjustment. This arrangement can lead to rising production overheads owing to the need to allocate additional resources to cover emergencies. The present study seeks to develop a method by which the task of machine adjustment can be delegated to an unskilled operator in a "controlled" manner.

An experimental set-up, which consisted of three independently driven rotating shafts and an reciprocating device, has been designed to mimic the movement of the key components in a packaging machine. A novel data acquisition unit has been developed to capture simultaneously the signals produced by three rotary encoders and a proximity sensor. The assimilated signals enable the state of the system to be ascertained. Any non-compliance of pre-determined operating limits can be detected before the end of each cycle of operation. Furthermore, the condition monitoring system can also provide information concerning the performance history of the machine.

The experimental results have demonstrated a high degree of repeatability and accuracy. This enables an innovative method to be devised to reduce process down-time through a supervised delegation of tasks. The method offers three distinct advantages: (i) to alert an operator of an imminent machine failure due to misalignment; (ii) to authorise and guides the operator to carry out appropriate machine adjustment; (iii) to record all the events for managerial reference.

Keywords: condition monitoring, packaging, data acquisition, down-time.

1. Introduction

The UK manufacturing industry has, over the past two decades, undergone radical changes both in the use of production methods and in working practices. The implementation of methods such as 'Just-In-Time (JIT)' and 'Single-Minute-Exchange-Die (SMED)', has helped to raise the awareness of the need to improve all aspects of the manufacturing process. Substantial productivity gain can be achieved through the automation of processes and the rationalisation of staffing levels. Greater dependence on machinery has meant that any production strategy becomes less effective if the performance of machines in the production chain can not be relied upon. Consequently there has been a drive, particularly amongst larger companies, to develop and implement preventative maintenance programmes in order to reduce the possibility of unexpected equipment breakdowns, thereby optimising plant availability and reducing maintenance cost. This approach requires the continuous monitoring of the 'health' of machinery and a diagnostic capability to allow the prediction of the onset of failures, usually involving the processing and interpretation of a large amount of data generated by sensors [1,2]. To this end, investigations have been conducted to assess the feasibility of using artificial neural networks to assist with the development of condition monitoring systems for various applications [3-5]. Packaging of products, often conceived as a relatively simple operation and where open-loop control is a common practice, is one area that has attracted less investment in new technology. Automation in such an operation often refers to the automatic execution of tasks with little or no measurement of parameters for feedback control or parameter adjustment. A mistake can result in the loss of 'finished' products due to damage; this becomes especially relevant if the cost of additional quality checks, after problems have been identified, exceeds the value of the product. A typical packaging machine relies mainly on two types of motion to perform its task: rotation and translation; these manifest in the form of pushers, conveyor belts and other manipulators. These motions are transmitted to the required parts from the power sources through various mechanical parts: gear trains, rack and pinions, belts, cams and followers. It is important that all these parts perform in synchronisation in order to achieve the desirable outcome.

Problems arise when misalignment is present in the system, the cause of which can come from several sources. For example, the unintentional variations in the set-up of a machine and wear and tear on the mechanical components amongst others. There is also the tendency to try to achieve higher yields by operating the machine closer to its operational limits. This will put more stress on the synchronised systems, which in turn may start to show the strain by exaggerating any misalignment, leading to an eventual breakdown of the packaging process. To minimise the machine down-time and hence the loss of production, skilled mechanics are employed whose expertise is used to identify, track and eliminate faults. Though commonly used, this approach has a number of drawbacks: increased maintenance overheads, non-availability of 'experts' and 'expertise', human variance and production inflexibility. By focusing on the maintenance problems exhibited by a typical industrial packaging machine, this project seeks to develop a method for predicting potential causes of misalignment. This method will help to facilitate condition based maintenance, either to provide assistance to the mechanic and hence reduce set-up variance, or to delegate system adjustment tasks to unskilled operators in a controlled manner.

2. Experimental Set-up

The present study is a collaborative project between an international company and the Nottingham Trent University. The industrial packaging machine being investigated has three rotating shafts (the "cams") and a reciprocating mechanism (the "rake"). The movements of individual components must be synchronised through repeated adjustment in order to achieve an optimum operation. To minimise disruption to the packaging process during the development and testing of the Real-Time Condition Monitoring (RTCM) system, an experimental set-up has been designed to mimic the movements of the key components of the packaging machine (Figure 1). The rig consists of three main parts: Instrumentation; Hardware interface and data processing; Software analysis and display.

2.1 Instrumentation

The main problem of machine down-time is believed to be attributable to the lack of useful information governing the general machine set-up and operating window. Current remedial actions often involve the subjective adjustment of machine parameters until a workable solution is found, but this approach has a number of drawbacks: time consuming, inconsistent and expensive. For condition monitoring new instrumentation has to be introduced to the packaging machine; rotational movements of individual shafts are to be measured by shaft encoders, while translational motion of the rake is to be measured by an inductive proximity sensor which has a high degree of consistency and reliability. Shaft encoders are well suited for measuring repetitive rotational movements and are available in a variety of different resolutions and mounting types. For the experimental rig, rotational motions are provided by three stepper motors each of which is coupled to an encoder, and the reciprocating motion is generated by a cam and follower driven by a fourth stepper motor.

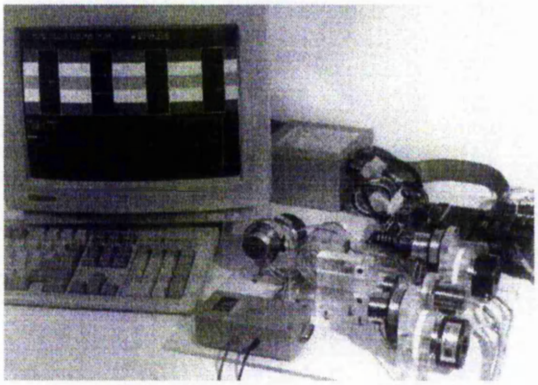


Figure 1 Experimental Set-up

2.2 Hardware Interface And Data Processing

The next stage involves the quantification of the offset and tolerance band for each shaft, as well as the exact position of the reciprocating device, but the shaft encoders and the proximity sensor produce two different types of output signal: digital and analogue. Figure 2 shows the design of the data acquisition interface. The encoders produce output in the form of two square waves with a 90° phase shift, allowing directional as well as positional information to be obtained. Conversion of the encoder output to a numerical form, representing the angular position, is performed by Quadruple decoders.

The proximity sensor provides a standard 4-20 mA output which is transformed into the numerical form by means of an analogue-to-digital (ADC) converter. For practical reasons, the following two constraints need to be overcome. Firstly, although the condition monitoring system is designed to be operated with a personal computer, the latter can not be relied upon to provide a steady timing between sampled data points. This is because a personal computer is not designed as a dedicated system for real-time monitoring of external devices; any "timing jitter" during the acquisition of data will distort the information contained therein. Secondly, individual rotating shafts in the industrial packaging machine may run at variable speeds and hence a fixed reference may not be available. Thus any varying time shift that exists between sampled data points will make extraction of synchronisation information extremely difficult. Parallel sampling offers the only workable solution if the above problems are to be eliminated. A novel technique has been developed which involves the use of the timer within the ADC to generate a synchronisation signal to enable parallel sampling to take place. The sampled data is then stored in buffers ready to be processed by the personal computer. The RTCM interface has the following key features;

- 8 Encoder inputs (square wave)
- 8 Analogue input channels
(expandable to 16)
- 24 Digital I/O lines (optional)
- Parallel sampling of analogue & digital data
- 8 ms scan-time (based on 486DX2/66)
- Up to 120 process cycles per minute

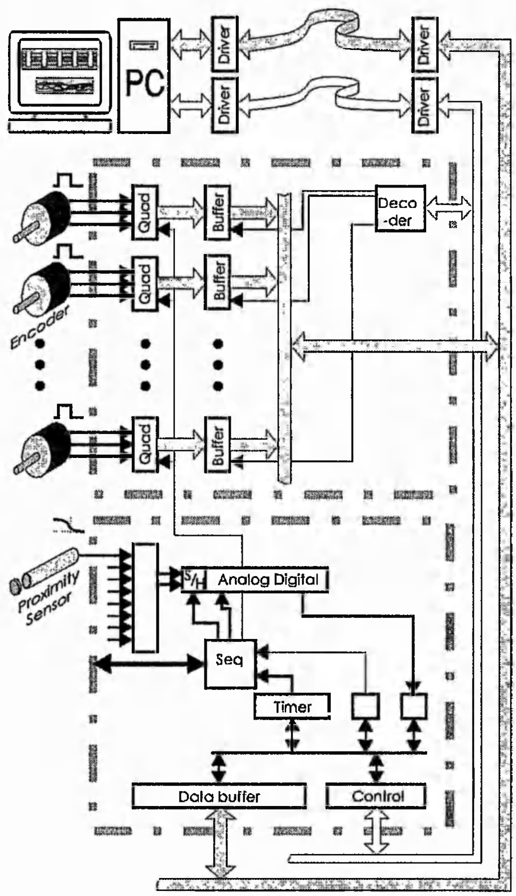


Figure 2 Data Acquisition Interface

2.3 Software Analysis And Display

The data acquired by the hardware must be processed and displayed within one machine cycle so that an operational decision can be made. Figure 3 shows how an offset between two transducers, Δ , is determined, with encoder 0 chosen as the reference. The rotational offset, Δ_{10} , between encoders 1 and 0 is the difference of their angular positions, while the translational offset, Δ_{p0} , is a measure of the position of the rake relative to the angular position of encoder 0.

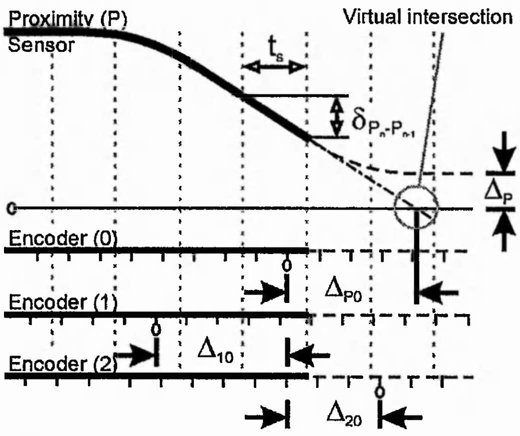


Figure 3 Determination of relative offsets

The software converts the signals stored in the buffer into relative phase shifts from which the minimum and maximum values are identified. It checks if the predetermined operational limits have been exceeded and displays the relevant information on a VDU. Since the interface can acquire data simultaneously from all transducers at the rate of 125 samples per second (i.e. at a sampling time of 8 milliseconds), a 'time slicing' technique has been used to overcome the limitations imposed by the graphic display. The software also has the capability to cater for different set-up conditions and operational windows. The RTCM software has the following key features.

- Cyclic representation of machine status
- Graphical display or tabular format
- Data rationalisation and interpretation algorithms
- Logging of raw or assimilated data
- Real-time system with sampling priority

3. Results

Predicting the onset of failures caused by the misalignment between moving parts is the principal goal of the present investigation, but the case study is also intended to provide design information on various aspects pertaining to the development of a low cost PC-based real-time condition monitoring system for packaging operation. These include:

- reviewing a range of packaging processes;
- identifying the key methods for power transmission;
- selecting low maintenance and reliable transducers;
- designing an electronic interface for parallel data sampling;
- developing efficient data analysis methods;
- designing an ergonomic information display.

In common with other manufacturing processes, the packaging operation has a set of limits (tolerance) within which efficient performance can be attained. The determination of these tolerances holds the key to predicting the onset of failures and the case study has shown that detailed and reliable information is needed in three areas: absolute displacement of individual transducers, relative offset of rotary elements (synchronisation) and correlation of rotational and translational movements. Figure 4 shows the image of a screen displaying the results of the immediate past cycle of machine operation. For convenience, the term "cam" refers to a shaft and the "rake" refers to the reciprocating device. The first three columns (from left to right) on the top half of the image provide information on the relative offsets between cam 0 and the rake, cams 0 and 1, cams 0 and 2, while the fourth column gives the proximity of the rake. Colour bands (or in this case varying grey levels) are used to signal the degree of misalignment. The positions of the markers signify the status of the operation: normal (region 1); machine adjustments are needed (region 2); shut down is recommended (region 3). The relative offsets of rotary elements vary between minimum and maximum values, as displayed in columns 2 and 3. By measuring the absolute displacement of the rake using the proximity sensor, it is possible to detect positional undershoot or overshoot of the rake during the filling function. The most important part of the data analysis is to develop a simple, but reliable, method of determining the relationship between the rotating shafts and the reciprocating rake. The synchronisation of these two mechanical systems ultimately determines the success of the packaging operation. By linking both systems to a datum (Figure 3), the relative offsets can be determined for each machine cycle. The monitoring system is capable of measuring angular displacement to a resolution of 0.018° and linear displacement to an accuracy of ±0.01 mm, thus allowing accurate determination of the synchronisation characteristics. It is not only important to show the current state of the machine operation, but to predict the likelihood of failure. This can be achieved by showing the trends of the relevant parameters. The bottom half of the screen (Figure 4) charts the trends of individual transducers over the past 500 cycles. A method analogous to the Statistics Process Control (SPC) could be used to help an operator to identify changes in performance over time.

4. Discussion

The aim of the project is to develop a method, through real-time condition monitoring, for improving the productivity of a packaging operation by reducing its down-time. A detailed investigation of the problem, taking into account the requirements specified by the industrial partner, has led to the conclusion that a deductive approach is needed to predict the onset of failures based on the gathered "intelligence". To this end, shaft encoders and an inductive proximity sensor have been chosen as the appropriate instruments for their accuracy and reliability. Although the resolution achievable with these devices may appear to be too good for tackling the misalignment problems associated with industrial packaging machines, the case study is also intended to provide a framework within which a generic solution can be developed for monitoring other processes and machines. Indeed tests carried out, but not presented here, have suggested that the current set-up may be used to identify the inherent errors present in gearboxes, from which the standard operating variations can be determined.

For practical reasons all basic functions of the RTCM system including data logging, analysis and information display must be completed within one cycle of machine operation. This requirement presented an interesting challenge to the research team because of the relative small timeframe within which operational decisions have to be made. A data acquisition interface capable of sampling analogue and digital data simultaneously has been developed to facilitate the determination of synchronisation and relative offsets at individual sampled data points.

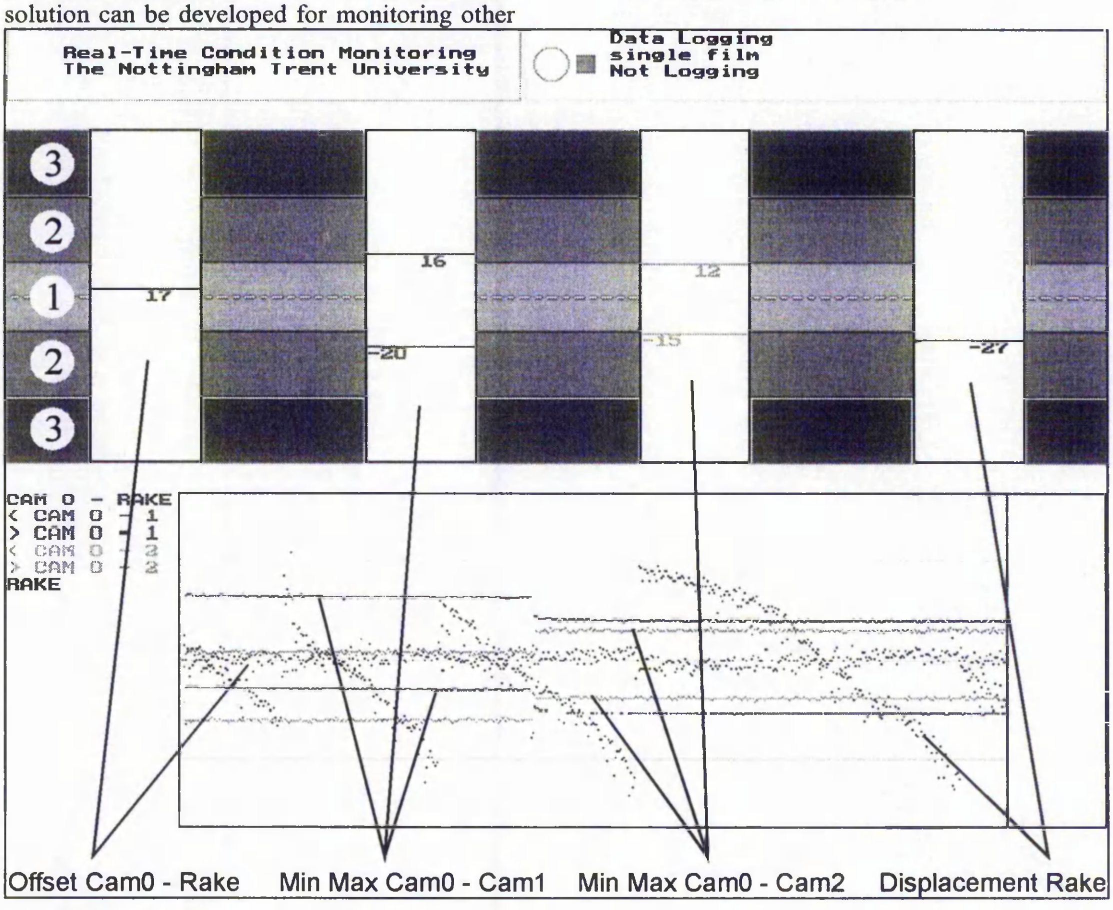


Figure 4 The RTCM display screen

The correlation between the rotating shafts and the reciprocating mechanism was obtained by means of a gradient technique which has shown to be independent of the sensitivity of the proximity sensor. By combining the historical information governing the mechanical behaviour of individual components given in the trend chart, with the knowledge of the degree of misalignment, if any, between them, the RTCM system is capable of discerning the state of a packaging machine.

The capability demonstrated by the monitoring system permits a cost effective maintenance strategy to be developed through the principle of ownership of responsibility. A comparison of the proposed strategy with the current practice is shown in figure 5. The iterative process of identifying faults, deciding on the corrective actions and carrying out parameters adjustment is to be replaced by the two stage process. Whenever a fault is identified or an initial machine set-up is needed, the RTCM system will decide on the corrective actions and this will be communicated to a person authorised to carry out the adjustment in the form of graphical instructions.

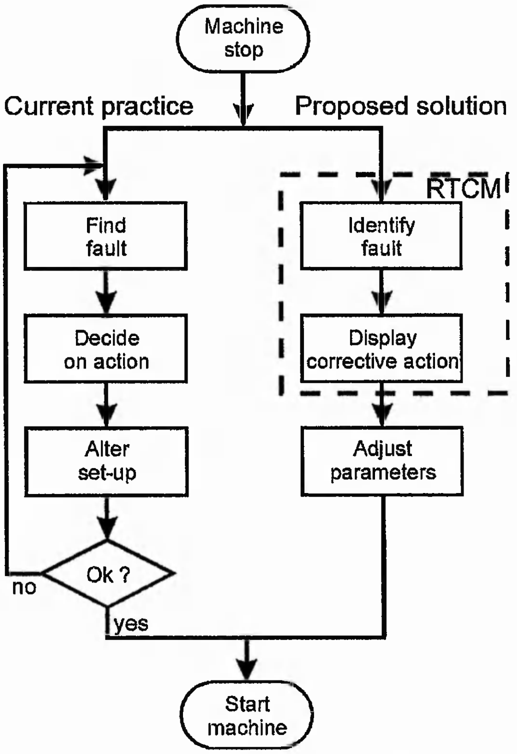


Figure 5 A simplified maintenance procedure

Figure 6 shows an example where the status of all three cams (i.e. shafts) are displayed along with suggested corrective actions. In this case cams 0 and 2 are misaligned but not cam 1. For each cam, the degree of offset is represented by the shaded segment of the outer circle, while the colour of the inner circle indicates whether correct adjustment has been completed. In the case of cam 0, corrective actions should be taken to gradually reduce the shaded segment by turning the shaft clockwise until it has completely disappeared and the inner circle changes its colour, as shown in cam 1. By the same analogy, corrective actions for cam 2 should be anticlockwise. As the RTCM system monitors the operation continuously, all events are logged automatically for operational control and production statistics. The proposed methodology was tested on the experimental set up and encouraging results were obtained.

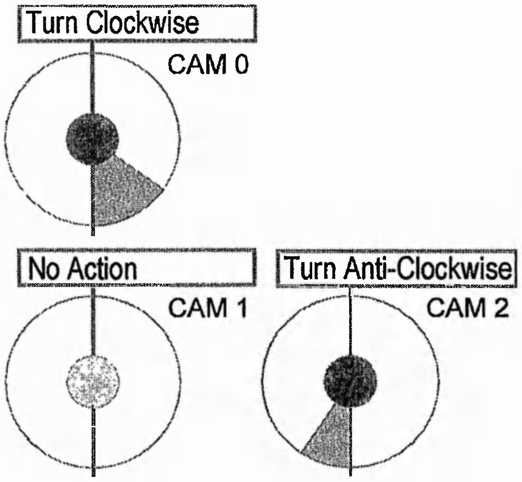


Figure 6 Computer-guided machine set-up

5. Conclusions

The success of a real-time condition monitoring system for an industrial application will be judged by a number factors including technical merits, ease of installation and implementation, cost effectiveness and reliability. The causes of breakdowns in packaging machines are reasonably well understood by skilled mechanics, but the interaction between moving parts within the machine has meant that "operating windows" can only be identified by a time consuming trial and error process. In terms of investment in new technology, packaging operation is likely to remain the poorer cousin of its manufacturing counter parts.

With this in mind, a low cost system has been developed to enable the prediction of the onset of failures through continuous monitoring of the "health" of a packaging machine. By means of an experimental set up, the capability of the system was ascertained and the methodology validated. Once installed and commissioned, the system can help to further reduce process down-time through a supervised delegation of machine adjustment tasks. The proposed method offers three distinct advantages: (i) to alert an operator of an imminent machine failure due to misalignment; (ii) to authorise and guide the operator to carry out appropriate machine adjustment; (iii) to record all the events for operational control and production statistics.

References

[1] Moon C. 1995. Condition Monitoring puts the squeeze on plant downtime *Noise and Vibration Worldwide,* ***Vol.26****, 9, Institute of Physics Publishing Ltd Bristol, 12-14.*

[2] Poon H L. 1991. A Knowledge-based Condition Monitoring System For Electrical Machines. *Computers In Industry,* ***Vol.16****, 2, 159-168.*

[3] Lai E, Plantenberg D, Grant Z R and Hull J B. 1996. Condition Monitoring of plants and equipment with the aid of neural networks *Proc. Symposium on Mechanics in Design, University of Toronto, Canada, 6-9 May, 515-522.*

[4] Aluindigue I E et al. 1993. *Monitoring and diagnosis of rolling element bearings using artificial neural networks. IEEE Trans. on Industrial Electronics,* ***Vol.40****, 2, 209-216.*

[5] Chow M, Magnum P M and Yee S O. 1991. A neural network approach to real time condition monitoring of induction motors. *IEEE Trans. on Industrial Electronics,* ***Vol.38****, 6, 448-453.*

A FUZZY APPROACH TO THE CONDITION MONITORING OF A PACKAGING PLANT

M Jeffries, E Lai, D H Plantenberg, J B Hull

Department of Mechanical and Manufacturing Engineering, The Nottingham Trent University, Nottingham NG1 4BU, United Kingdom. Fax: +44 115 9486506; email: e.lai@domme.ntu.ac.uk

ABSTRACT

The packaging of manufactured products attracts relatively little attention from research or benefits from the opportunities available through condition monitoring. The technical requirements of packaging can easily be underestimated in the overall production process. A typical packaging machine has translational, reciprocating and rotational motions guiding and manipulating the product. The contributing factors to packaging problems include machine set-up variations and synchronisation problems, due to the misalignment of key components such as cams or transmission systems. These misalignments can cause jamming within the machine leading to unnecessary damage or loss of product, resulting in the need for additional quality checks, the cost of which may exceed the value of mass-produced high premium products.

A widely used method of reducing waste relies upon the knowledge of an experienced mechanic to set up the machine before production runs and to correct misalignments should problems occur. Whilst this method offers a workable solution, there are a number of inherent problems. Relevant practical experience may take a long time to acquire, which can lead to a shortage of expertise. For economic reasons, there is also a tendency for manufacturers to reduce production overheads through rationalisation of maintenance staffing levels and the formation of multi-discipline teams. Instead of allocating specific responsibilities to individuals, members of a team are expected to provide emergency cover for a number of machines over the factory floor, in addition to their routine maintenance tasks. This can present problems in scheduling maintenance personnel should a machine require regular adjustments or a number of machines breakdown at the same time. Even with the mechanic present, the current practice of machine adjustment consists of a certain amount of hit-and-miss judgement and therefore the time taken to diagnose and remedy faults can significantly vary even for similar problems. Larger companies often implement Real Time Condition Monitoring (RTCM) systems as a means of tackling this problem. Although condition monitoring can produce the information on performance parameters, expertise is required to decode this intelligence. Furthermore, the cost of analysing the large volume of data logged from even the simplest machine can be prohibitively high. Production overheads are necessarily raised in both these scenarios, either by the employment of highly skilled labour or the resultant downtime from their lack of availability.

This study aims to develop an efficient, hybrid method for capturing the machine information, by means of fuzzy logic to decode the intelligence supplied from an RTCM system. It also intends to provide an insight into the operation of the machine from a management perspective and the operator's point of view. The fuzzy diagnostic system is required to consolidate a number of input variables into a single output value representing the current state of the machine. This is to be achieved by introducing a modification to a standard fuzzy logic correlation technique.

Input variables measured by the RTCM system, including synchronisation offsets between moving parts, are classified into five fuzzy sets: LARGE-NEGATIVE, SMALL-NEGATIVE, NOMINAL, SMALL-POSITIVE and LARGE-POSITIVE. The output domain has three sets: SAFE, CAUTION and CRITICAL, representing the possible states of a typical packaging machine. A rule base links the input variables to form a single defuzzified output value. The sensitivity of the individual rules is governed by the distribution of the input sets.

A typical rule base example is

if ENC0-ENC1 LOWER LIMIT *is* NOMINAL *then* MACHINE *is* SAFE

if (ENC0-ENC1 LOWER LIMIT *is* SMALL -VE)
or (ENC0-ENC1 LOWER LIMIT *is* SMALL +VE) *then* MACHINE *is* CAUTION

if (ENC0-ENC1 LOWER LIMIT *is* LARGE -VE)
or (ENC0-ENC1 LOWER LIMIT *is* LARGE +VE) *then* MACHINE *is* CRITICAL

where ENC0-ENC1 LOWER LIMIT is the minimum relative offset between two encoders during one machine cycle.

The structure of the rules is kept simple, to allow for a generic solution to be generated for all similar processes. The effectiveness of the fuzzy logic reasoning process will depend on the calibration of the input variables, where the input fuzzy sets are to be tuned to match intended operational tolerance windows.

In order to maintain low computational overheads in real-time condition monitoring systems, a variant of the commonly used min-max to centroid method has been developed to correlate the outcomes of all rules into a matrix of output sets. When using min-max inference, information from some rules is lost - only the maximum 'cut' is used. In this new 'matrix' variant all information is stored before defuzzification in the matrix, so that no outcomes of the rules are dismissed. A variation on a standard centroid method is then used to produce the defuzzified result. A key feature of the 'matrix' approach is to cause subtle changes to the shape of the fuzzy decision surface. This results in the profile becoming flattened around the areas associated with the centre of the output sets, while allowing the surface to still pass through the anchor points at the corners. Application of the new 'min-matrix' correlation technique in conjunction with the modified centroid method will help to reduce computational overheads.

To evaluate the proposed methodology, a purposely-built condition monitoring experimental test rig has been designed to mimic the operation of a film packaging machine, including the main rotational and reciprocating motions. The rig consists of three constituent parts.

- Instrumentation - three optical shaft encoders and a proximity sensor are used, each of which is driven by a stepper motor.

- Hardware interface and data processing - a novel data acquisition interface has been developed, capable of the parallel sampling of analogue and digital signals.

- Software analysis and display - a personal computer is used to log, assimilate and display the data. Data analysis is to be carried out by the fuzzy diagnostic software.

The test rig allows for the introduction of controlled errors into individual measured parameters. The parameters of the packaging machine have associated tolerances within which satisfactory performance of the machine is assured - often referred to as operational windows. Although the instrumentation can provide intelligence associated with individual parameters, the interaction between moving elements that determines the overall system state is more difficult to decipher. The use of operational windows does provide a means of quantifying performance but two main problems need to be addressed. Firstly, a mechanism must be devised to ascertain the position of individual windows. Secondly, the functional relationships between individual operating windows need to be established. As these effects cannot be easily described through mathematical models, fuzzy logic appears to offer a simple and robust approach enabling heuristic information to be represented in a structured manner. The developed fuzzy condition monitoring framework allows for the easy transfer from the current application to other similar problems, with minimal alteration to the software. The operational windows of the machine can be mapped onto the fuzzy domain allowing individual fuzzy sets to become the corresponding tolerance bands within each window. The fuzzy set is well suited to this role, showing degrees of membership rather than a simple true or false association. By describing an operating window in a fuzzy manner, even though the true value is discrete, a close approximation can be produced.

The min-matrix correlation method is capable of combining multiple, but not necessarily similar, variables into a single output value. This in turn will provide a useful means of tracking machine performance through the monitoring of a single value, by tuning fuzzy sets to match estimated operational windows. Preliminary results have shown that the min-matrix correlation technique introduced within a fuzzy RTCM system can produce useful information for diagnosing the state of a packaging machine and provide a methodical approach for close estimation of operational windows. The work has therefore lent support to the use of fuzzy condition monitoring as a reliable and inexpensive means of reducing wastage and maintenance overheads in the packaging industry.

A FUZZY APPROACH TO THE CONDITION MONITORING OF A PACKAGING PLANT

M Jeffries, E Lai, D H Plantenberg, J B Hull

Department of Mechanical and Manufacturing Engineering, The Nottingham Trent University, Nottingham NG1 4BU, United Kingdom. Fax +44 115 9486506
email: martyn.jeffries@ntu.ac.uk, e.lai@domme.ntu.ac.uk

Abstract.

The packaging of manufactured products attracts relatively little attention from research or benefits from the opportunities available through condition monitoring. The technical requirements of packaging can easily be underestimated in the overall production process.
A widely used method of reducing waste relies upon the knowledge of an experienced mechanic to set up the machine before production runs and to correct misalignments should problems occur. This approach has the following drawbacks: non-availability of expertise, the relevant practical experience takes time to acquire and the reduced staffing levels present in the current economic climate.
This study aims to develop an efficient, hybrid method for capturing the machine information, by means of fuzzy logic to decode the intelligence supplied from an RTCM system. Developed from the min-max fuzzy inference method, the present approach replaces the maximum selection technique for output set truncation, by a matrix of truncation values derived from all rule outcomes. A suitably generic method has been employed allowing the developed techniques to be utilised for a wide range of manufacturing problems.
The work has lent support to the use of fuzzy condition monitoring as a reliable and inexpensive means of reducing wastage and maintenance overheads in the packaging industry.

Keywords: Fuzzy Logic, Condition Monitoring, Packaging

1. INTRODUCTION

The role of condition monitoring has been exploited in many aspects of manufacturing industry, but its use in the final stage packaging is only now being investigated [1]. The technical requirements of packaging can easily be underestimated in the overall production process as any problems occurring at this stage can lead to unnecessary damage or loss of product.
In industry, the availability of a skilled mechanic becomes a crucial element when attempting to achieve performance or efficiency targets. The set-up and maintenance of a machine can be a time-consuming operation, which requires iterative steps, including some trial and error, to achieve near-optimal set-up. The problems arise when condition monitoring is employed as a means to develop an understanding of the relationship of machine elements that leads to desirable performance. The tracking of individual variables is not problematic, but to deduce the overall machine condition is more difficult. The mechanic can perform this task, to some extent, by using a set of rules learnt through experience. A similar approach is available by means of fuzzy logic, which has a proven ability to convey heuristic information governing non-deterministic problems using a rule base. Fuzzy logic may even allow an insight into the performance of the machine from a management perspective or operator's point of view.

Fuzzy logic has enjoyed much attention from control researchers in their attempts to capture the heuristic and often non-discrete nature of real-world problems. It has proved invaluable in its ability to provide simple and robust solutions to problems considered difficult to tame with conventional control theory. However, there may be one drawback, although some may see it as a benefit, is the ease and simplicity with which a fledgling fuzzy logic 'toolbox' can be developed without thoroughly understanding the mechanisms contained therein. The ability to select methods within a fuzzy toolbox, for reasons of computational ease or apparent suitability, can create algorithms that do not necessarily have a solid theoretical basis linking the separate components - correlation, defuzzification etc. This has been recognised as a problem by Rondeau et al. [2] and is an important consideration in the choice of the fuzzy method used for a specific application. The criterion for selection of the method should not be based purely on the need to match the output to the expected result. But rather, to understand the algorithms involved, by ensuring correct manipulation of the data and 'respecting' the intention of the given application. This approach will provide a solid foundation from which an optimal fuzzy system can be developed, and hence the attainment of the desired output. This means that fuzzy logic does not become the quick fix for those people uninterested in control theory, thereby giving greater credibility to the field of research in the industrial sector.

Investigations into the use of Artificial Neural Networks (ANN) in condition monitoring [3-5] have shown the benefits of using artificial intelligence techniques in the diagnosis of machine condition. A hybrid approach utilising fuzzy logic in conjunction with ANNs has also been explored [6]. Within the packaging process, the monitoring takes on another aspect. The focus is not only on individual elements, such as tool wear [7], but the complex interaction between transmission systems and the various types of motion used to convey product through the machine. In this context, the application of fuzzy logic must be considered in its approach to solving the problems encountered. A thorough understanding of the processing involved must be a pre-requisite to achieve the goal of successful condition monitoring.
This paper outlines the theoretical aspects of a proposed condition monitoring system using fuzzy logic. The fuzzy system is to be discussed in its relevance to the problem in hand. Experimental validation of the proposed method has been carried out on a purposely-built condition monitoring rig and some results are presented here.

2. FUZZY OPERATIONAL WINDOWS

The determination of state for a packaging machine or any other type of machine can be a difficult process. A machine typically has several different types of motion (rotational, translational, reciprocating) performing a variety of tasks. The parameters governing these motions are susceptible to machine set-up variations and relational offsets causing synchronisation problems. A method for quantifying machine parameters is to use so called 'operational windows'. An operational window is a way of analysing how the actual value of a variable relates to a pre-determined datum. The term 'error' is used to describe a measure of the difference between the pre-determined datum and the actual value of the parameter. In theory, a machine running at absolute optimum would have no errors, and all values would be at the centre of their respective operational windows (Figure 1)
The width of the operational window is the tolerance within which a parameter can lie without causing the machine to fail. The window is subdivided to differentiate smaller errors, which tend to reduce machine performance, from larger errors risking process breakdown.

Although operational windows do provide a means of quantifying machine performance, their determination and use can suffer from two problems. Firstly, a mechanism must be devised to ascertain the positions of individual windows. Secondly, the functional relationship between these windows needs to be established. As these relationships cannot be easily described through mathematical models, fuzzy logic appears to offer a simple and robust approach enabling the heuristic information to be represented in a structured manner. The operational window of a parameter can be mapped onto the fuzzy domain, thus allowing individual fuzzy sets to become the tolerance bands within each window (Figure 2).

Input parameters can therefore be classified with five fuzzy sets: LARGE-NEGATIVE, SMALL-NEGATIVE, NOMINAL, SMALL-POSITIVE and LARGE-POSITIVE. The fuzzy set is well suited to this role, because it shows varying degrees of membership rather than a simple true or false association. By describing an operating window in a fuzzy manner, although the true value is discrete, a close approximation can be produced by utilising the flexibility and robustness of the fuzzy logic method.
In describing individual operational windows within the fuzzy domain, the opportunity to combine information using the fuzzy rule base becomes apparent. If operating windows are to be combined, the result must also be an operating window representing the overall machine state. A typical machine can be described by three basic states:

Safe, all parameters are close to nominal error and there is minimal risk of process failure.

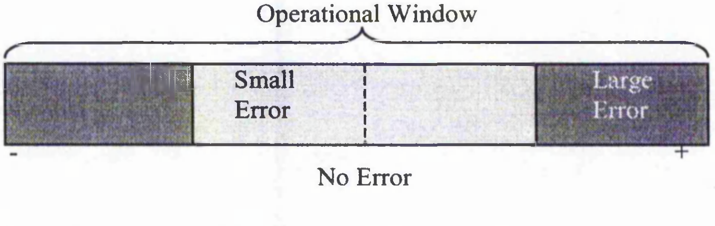


Figure 1. A schematic diagram of an operating window

Caution, some parameters are showing small errors and there is a possible risk of process failure.

Critical, one or more parameters have exceeded acceptable tolerance limits and process failure is imminent.

These states can be classified in the output fuzzy domain as three sets - SAFE, CAUTION, CRITICAL.

The rule base, to link input to output sets, is kept simple to allow a generic approach to be formulated (Figure 3). The sensitivity of the rules is governed by the distribution of the input sets. The effectiveness of the fuzzy logic process will depend on the calibration of the input variables, where input fuzzy sets are tuned to match intended operational tolerance windows.

The implementation of the rule base within the fuzzy logic framework must be properly considered. The choice of correlation/defuzzification method is an important aspect in 'respecting' the information present in the input parameters. To ensure this, a method has been developed to encapsulate the needs of this application.

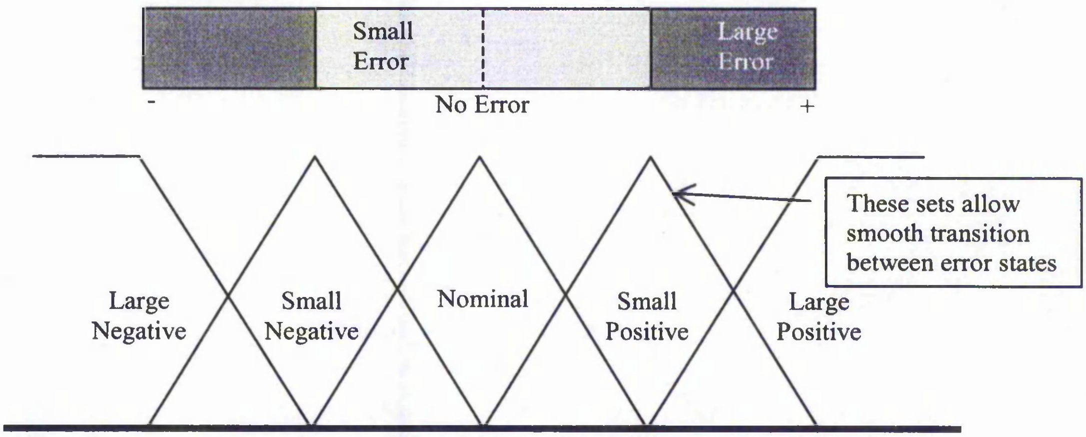


Figure 2. The mapping of an operating window onto the fuzzy domain.

RULE BASE

if PARAMETER_1 *is* NOMINAL *then* MACHINE *is* SAFE

if (PARAMETER_1 *is* SMALL -VE) *or* (PARAMETER_1 *is* SMALL +VE) *then* MACHINE *is* CAUTION

if (PARAMETER_1 is LARGE -VE) *or* (PARAMETER_1 *is* LARGE +VE) *then* MACHINE *is* CRITICAL

where PARAMETER_1 is, for example, the relative offset between two rotational elements during one machine cycle.

Figure 3. An example of the rule base structure.

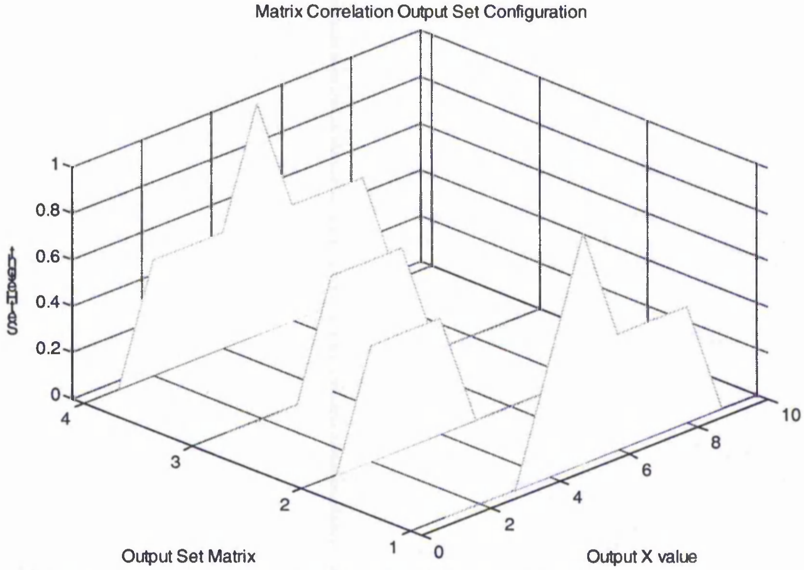


Figure 4. Three-dimensional representation of stored truncation values.

3. MATRIX INFERENCE

A commonly used method for fuzzy inference, is the min-max method. However, this simple but effective technique does have a drawback when used in problems related to risk assessment type problems, such as monitoring and diagnosis of the state of a machine. In this application, it is desired that all rules contribute towards the final defuzzified result to ensure the derivation of a good estimation of machine state. With the min-max method, only rules presenting a maximum truth-value for a given output set are contributing to the final outcome. A method for alleviating this problem is to use the fuzzy additive approach, where the results of the rules are summed to determine the output set truncation. However, this approach is deemed unsatisfactory, as rules are prevented from contributing further if the total calculated truncation value exceeds the height of the output set. Therefore, a new approach, entitled the 'matrix' method, has been devised to meet the requirements of this application. Developed from the min-max method, the new approach replaces the maximum selection technique for output set truncation, by a matrix of truncation values derived from all rule outcomes. By utilising the matrix values during defuzzification the fuzzy output domain, in effect, becomes three dimensional - as each possible cut is retained (Figure 4).
A centroid method (Eq. 1) can be used to defuzzify the matrix.

$$c_o = \frac{\int x\mu_o(x)dx}{\int \mu_o(x)dx} \tag{1}$$

where,
x *is the position on the x-axis of an output set*
$\mu_o(x)$ *is the height of any given set at point x.*
c_o *is the total centroid.*

This can be made discrete using Eq. 2, for computational ease.

$$c_o = \frac{\sum_{i=1}^{N} c_i A_i}{\sum_{i=1}^{N} A_i} \tag{2}$$

where,
A_i *is the area of a set.*
c_i *is the individual set centroid*
N *refers to the total number of output sets.*

As can be seen from Eq. 2 the areas of the fuzzy sets are critical to the determination of the final defuzzified result. This is an important factor in shaping the way a fuzzy logic system links input and output parameters, by means of centroid defuzzification. This in turn leads to the creation of fuzzy decision surface. The flexibility of the centroid method proves useful in accommodating the new matrix approach, as two fuzzy sets can effectively occupy the same x-position and contribute equally to the defuzzified result.
A property of the present method is that the relationship between a given rule outcome and the consequent output set has been altered to provide a more suitable approach for this application, compared to that offered by other inference techniques. As mentioned earlier, the min-max method suffers from a disadvantage of not utilising all rule outcomes for determining output set truncation. As emphasis moves between rules, changes are only visible when the current maximum truncation value has been exceeded, even if the new rule outcome is almost equal in weighting. This leads to 'surprise' changes in the defuzzified output. In the fuzzy additive approach, truncation values are added but a non-linear result is formed, as the relationship between a rule outcome and the area created by truncation of the output set is not equal for all cases. For example, if two rules modify the same output set and both produce a result of 0.5, the consequent set would be of height 1.0. However, the second rule has only one-third the additive effect of a single rule, due to the triangular shape of the set defining the area. Where in fact the consequent set might be desired to have twice the area, which would double the effective weight of the set during the process of centroid defuzzification. The matrix method can overcome both of these problems. Firstly, all rules are considered however small their contribution. Secondly, rules producing equal results have equal influence on the final defuzzified value. The effect, therefore, is to subtly change the topology of the fuzzy decision surface linking input to output values.
A comparison between min-max and matrix methods is shown in Figure 5. Here the effects of two input variables, given an arbitrary scale of 0-50, were tested using the rule base developed specifically for the condition monitoring application. Unlike the min-max method, the matrix method shows a flattening of the areas associated with the centre of the output sets, creating smoother transitions through these points. It has also been found that computational overheads are not increased by this new method, as the computationally expensive maximum comparison is no longer required.

4. APPLICATION AND RESULTS

To evaluate the proposed methodology, a purposely-built condition monitoring experimental test rig has been designed to mimic the operation of a film packaging machine (Figure 6), including the main rotational and reciprocating motions.
The rig consists of three constituent parts.

- Instrumentation - three optical shaft encoders and a proximity sensor are used, each of which is driven by a stepper motor.
- Hardware interface and data processing - a novel data acquisition interface has been developed, capable of the parallel sampling of analogue and digital signals.

- Software analysis and display - a personal computer is used to log, assimilate and display the data. Data analysis is to be carried out by the fuzzy diagnostic software.

Instrumentation used on the test rig is identical to that intended for use on the actual packaging machine. The stepper motor drive components represent four of the main machine elements, the performance of which is critical in determining machine condition. Three of the motors are used to provide rotational motion, which simulate parts of the conveyor system that moves product through the machine. The fourth is used to create a reciprocating motion, by means of a cam and follower, to model the action of a 'rake' used to bring the carton boxes in-line with the product. This component is critical to the success of the packaging action, because misalignment here will guarantee failure. The rig permits the introduction of controlled errors into individual simulated components.

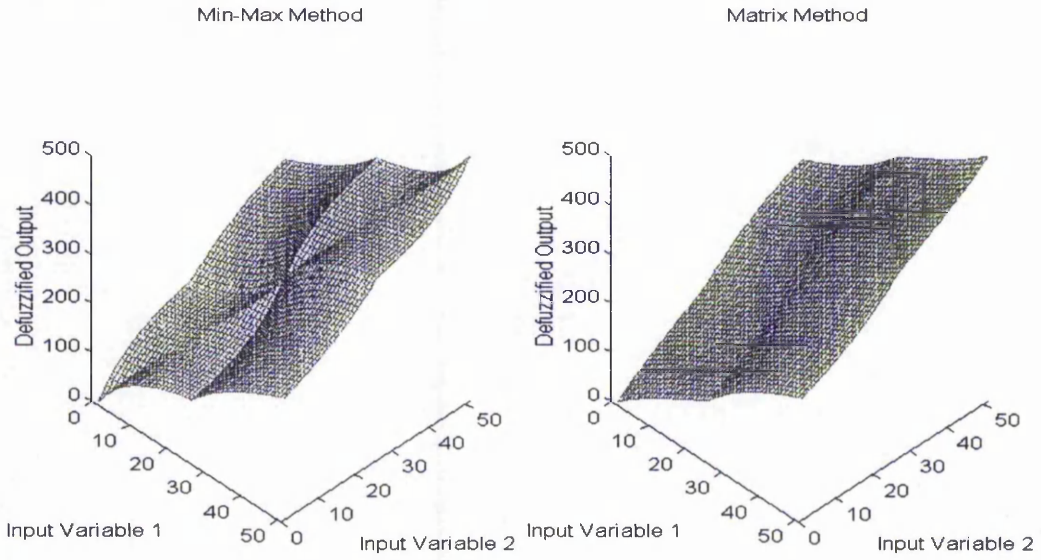


Figure 5. A comparison of the fuzzy decision surface generated by Min-max and Matrix methods.

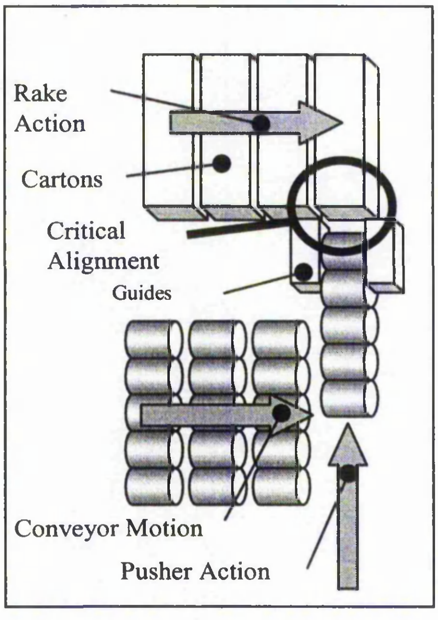


Figure 6. Representation of the packaging machine

The real-time condition monitoring system acquires signals from the optical encoders and the proximity sensor and, through processing, forms six observable parameters. All are considered over one machine cycle, which is the time taken to fill a single box.

Preliminary investigations have been carried out to test the general performance of the developed fuzzy logic approach, with respect to its ease of implementation and process applicability. Initial results were obtained by specifying arbitrary input sets to simulate operational windows. It should also be noted that by tuning the input sets, the fuzzy processing has been made more sensitive to changes in absolute position of the rake, measured by the proximity sensor, reflecting its importance in ensuring successful packaging operations.

Figures 7, 8 & 9 show three events over a 100-cycle section from a test run of the condition monitoring rig. The first event, covering the first ten cycles, is a large offset between two encoders (Figure 7). The proximity sensor is showing a small error at this stage (Figure 8). The output from the fuzzy logic (Figure 9) registers a high value of approximately 0.7, indicating an increased likelihood of process failure. Over the next 80 cycles, a minimal offset situation is recorded; while the proximity sensor indicates an increasing error on an upward trend. Over this period, the fuzzy result (Figure 9) can be seen to track the small variations manifested in the proximity sensor readings, whilst noting an offset error spike between the encoders just before machine cycle 70. Towards the end of the test run a sharp change in encoder offset is introduced, when the proximity sensor is already registering a large error. This results in a large increase in the output from the fuzzy logic diagnosis. The flatness of the profile is an indication that the tracked parameters have exceeded the limits of their operating windows. Hence, if this data represented a scenario of the operational conditions of an actual packaging machine, it could be concluded that a high probability of a critical process failure would occur at this time.

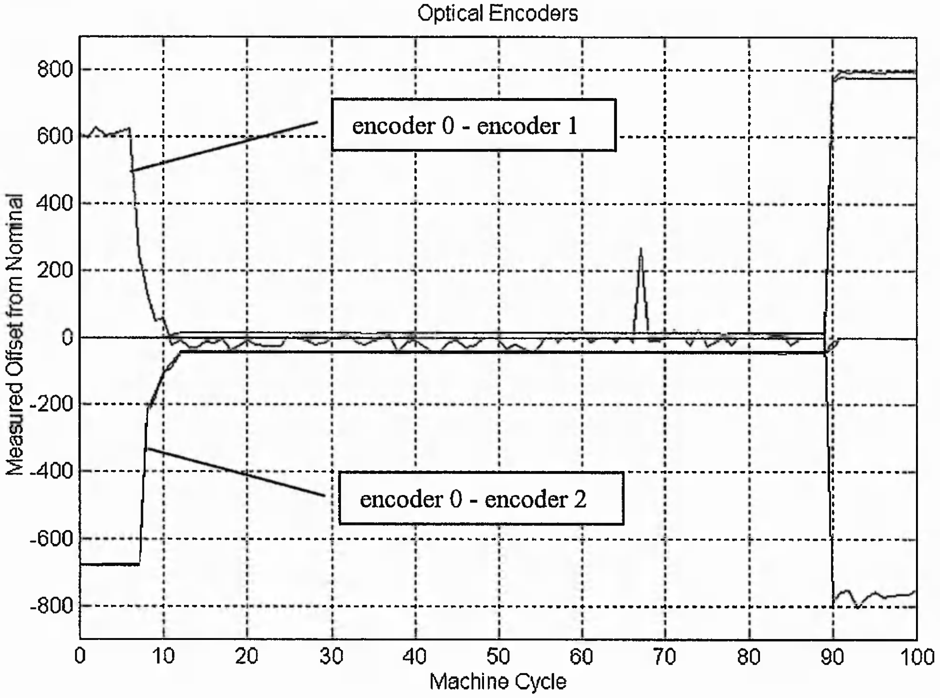


Figure 7. Optical Encoder Measurements.

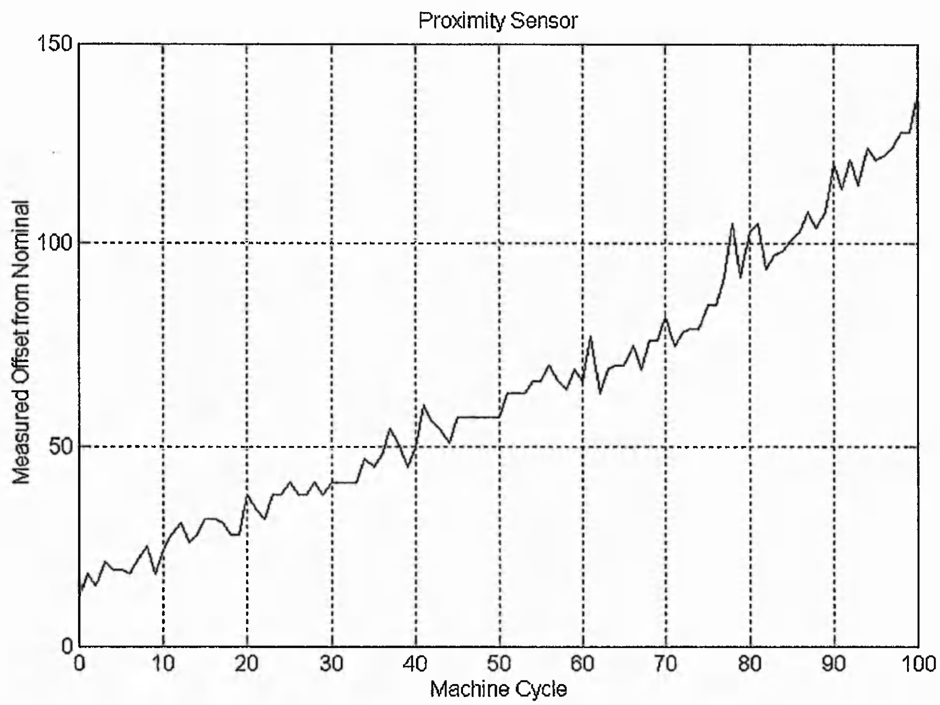


Figure 8. Proximity Sensor Measurements.

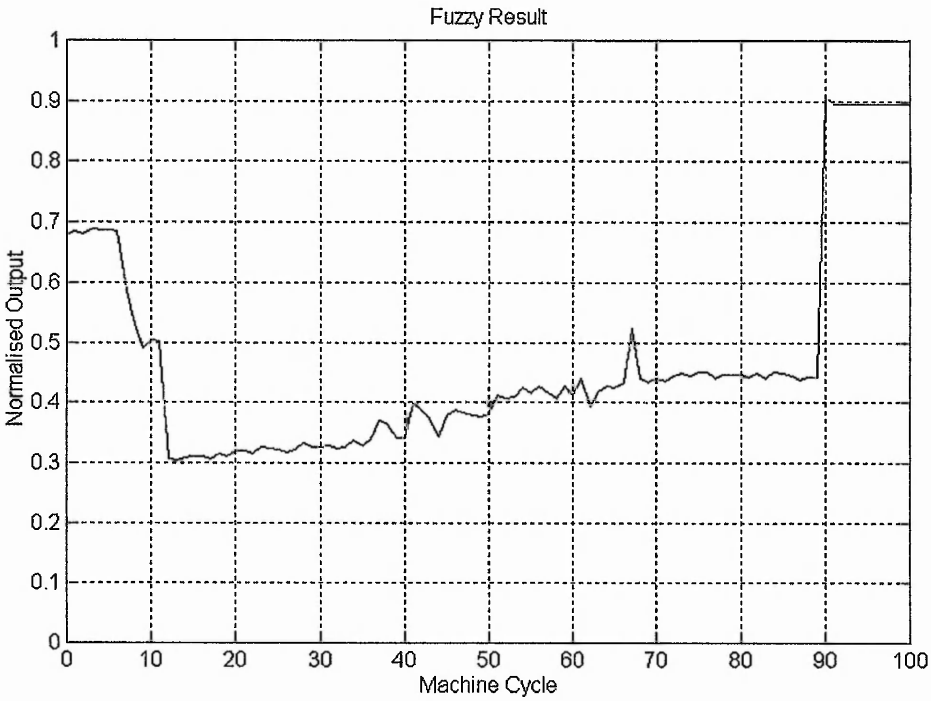


Figure 9. The calculated fuzzy result classifying machine state.

5. CONCLUSIONS

The matrix inference method is capable of combining multiple, but not necessarily similar, variables into a single output value. This in turn will provide a useful means of tracking machine performance through the monitoring of a single value, by tuning fuzzy sets to match estimated operational windows. A suitably generic method has been employed in the formation of the rule base and the linking of operational windows to the fuzzy domain. This will allow the developed techniques to be utilised for a wide range of manufacturing problems.

Preliminary results have shown that the matrix inference technique introduced within a fuzzy real time condition monitoring system can produce useful information for diagnosing the state of a packaging machine. Furthermore, fuzzy logic can provide a methodical approach for close estimation of operational windows. The work has therefore lent support to the use of fuzzy condition monitoring as a reliable and inexpensive means of reducing wastage and maintenance overheads in the packaging industry.

6. REFERENCES

[1] Plantenberg D H, Lai E and Jeffries M. 1997, A Novel Approach to Reduce Packaging Process Down-Time *Advances in Materials and Processing Technologies, AMPT '97* **Vol. 2**, 630-635.

[2] Rondeau L, Ruelas R, Levrat L, Lamotte M. 1997, A defuzzification method respecting the fuzzification *Fuzzy Sets and Systems* **86**, 311-320

[3] Chow M, Magnum P M and Yee S O. 1991, A neural network approach to real-time condition monitoring of induction motors. *IEEE Trans. On Industrial Electronics*, **Vol. 38**, 6, 448-453.

[4] Aluindigue I E et al. 1993, Monitoring and diagnosis of rolling element bearings using artificial neural networks. *IEEE Trans. On Industrial Electronics*, **Vol. 40**, 2, 209-216.

[5] Lai E, Plantenberg D H, Grant Z R, Hull J B. 1996, Condition Monitoring of plant and equipment with the aid of neural networks. *Proc. Symposium on Mechanics in Design, University Of Toronto, Canada*, **6-9 May**, 515-522.

[6] Li S and Elbestawi M A. 1996, Tool Condition Monitoring in Machining by Fuzzy Neural Networks,

Journal of Dynamic Systems, Measurement, and Control, Transactions of the ASME, **Vol. 118**, 665-672.

[7] Das S, Bandyopadhyay P P, Chattopadhyay A B, 1997, Neural-networks-based tool wear monitoring in turning medium carbon steel using a coated carbide tool, *Journal of Materials Processing Technology*, **Vol.63**, No.1-3, pp.187-192

REAL-TIME IMPLEMENTATION OF A FUZZY CONDITION MONITORING SYSTEM FOR PREDICTING PROCESS BREAKDOWN

M. Jeffries, E. Lai, D.H. Plantenberg and J.B. Hull
The Nottingham Trent University, Department of Mechanical and Manufacturing Engineering, Nottingham NG1 4BU, United Kingdom. Fax +44 115 948 6506, email: martyn.jeffries@ntu.ac.uk or e.lai@domme.ntu.ac.uk

Abstract

For a packaging process, machine set-up/adjustment and corrective maintenance are the two main contributors to recurrent production overheads. Whilst the diagnostic capability of condition monitoring may enable preventative maintenance work to be carried out when required, machine set-up/adjustment is a time consuming trial and error process normally carried out by experienced mechanics. Incorrect machine set up can lead to misalignment between moving parts and hence products jamming during packaging operation. The present study seeks to develop a fuzzy condition monitoring system to enable an expert decision to be made on the probability of an imminent process breakdown brought about by components misalignment.

An experimental rig has been designed to mimic the rotational and translational movements of a packaging machine. It consists of three shaft encoders, a proximity sensor, a novel data acquisition interface capable of parallel sampling, and a personal computer. The state of the machine is to be ascertained by means of fuzzy logic based on the analysis of the measurements of six input variables. The input sets represent the operational windows of the measured variables and expert knowledge is represented by rules. A new matrix inference technique has been developed to allow the determination of a single output value upon which an assessment can be made on the likelihood of an imminent breakdown.

By adjusting the weighting of the rules and the sensitivity of the input variables, the experimental results have shown that the fuzzy system is capable of modelling the dynamic behaviour of a packaging machine, which can have several set up variations. Furthermore, the novel data acquisition interface and software enables all performance calculations, decision making and information display to be completed within one cycle of machine operation. The fuzzy condition monitoring system thus developed is generic and can be applied to other manufacturing processes.

Keywords: condition monitoring, fuzzy logic, real-time, packaging, and downtime.

1 Introduction

The investigation of condition monitoring within the manufacturing industry has been largely confined to tool-wear monitoring [1] and direct machinery measurement such as gearboxes or bearings. The monitoring of overall machine condition, especially in the field of packaging has only recently begun [2]. This is despite the relatively high recurrent production overheads resulting from routine adjustment and corrective maintenance of packaging machines. To obtain satisfactory performance, current remedial actions utilise the knowledge and skills of an experienced mechanic. However, machine set-up/adjustment can still be a time consuming process involving a certain amount of trial-and-error. The types of machines being considered often contain rotational, reciprocating and translational motions. Two major factors need to be overcome when attempting to reduce maintenance-related overheads: (i) the lack of information describing the state of a process and (ii) the problem solving skills of an experienced mechanic usually take a long time to acquire. It may be considered that fuzzy logic, in combination with a real-time condition monitoring (RTCM), can offer a solution to helping resolve these problems, by encapsulating the heuristic qualities of the problem solving skills used by an experienced mechanic.

Artificial neural networks (ANNs) have proved valuable in the traditional areas of condition monitoring [3-9], and fuzzy logic in combination with ANNs has also been explored [10-11]. Nevertheless, the benefits of applying fuzzy logic alone in real-time condition monitoring have been under-exploited. While fuzzy logic has proved popular in the control field for solving 'difficult-to-quantify' problems, the danger of careless implementation has also been highlighted [12]. The need to observe good theoretical practice when implementing fuzzy logic systems can often be ignored with little adverse effect, due to their generally robust nature. However, long-term reliability and quality assurance is difficult to maintain should such a method be employed. When implementing a fuzzy condition monitoring within a real-time environment, there is an expectation that the software developed is more reliable than the machine being measured. It is therefore necessary to fully understand all processes involved in data acquisition, correlation, and display. Fuzzy logic can then be used for its abilities to manage difficult-to-quantify variables and its stability under unforeseen conditions.

Due consideration must also be given to ensure that any software within the fuzzy monitoring system is designed from the outset with real-time implementation in mind. The use of techniques such as multi-tasking or parallel processing have to be considered in ensuring data integrity, especially if synchronisation information is contained within the measured parameters [13].

This paper discusses the techniques and problems associated with implementation of a fuzzy logic within a real-time condition monitoring system. Particular emphasis is given to the real-time monitoring of packaging machines. The results of experimental evaluation of a prototype fuzzy condition monitoring system are also presented.

2 Application

The main aspects of a packaging machine are rotary motions of conveyor systems and reciprocating motions of actuating objects, such as a rake or a pusher. A pre-requisite to achieving good monitoring performance is the measurement of synchronous offsets, which necessitates the need for parallel sampling of analogue and digital signals from different sensors - a feature not readily available from commercial interface or monitoring products. It was therefore necessary to develop a complete package incorporating hardware and software aspects capable of meeting these requirements. A standard PC (486 class) has been used for software processing.

A purposely-designed test rig has been built to enable the validation of the prototype system prior to industrial implementation. Four main constituent motions of the packaging machine - three rotational, one reciprocating - are simulated using stepper motors (Figure 1), with the reciprocating motion achieved through the use of a cam-and-follower. Instrumentation includes three optical shaft encoders and an induction proximity sensor, identical to those intended for use on the packaging machine.

In addition to the developed instrumentation system, investigations into the benefits of fuzzy logic have been undertaken. As machine parameters often interact with each other in a non-linear manner, it is difficult to determine an overall machine state from several measured parameters of the monitoring using standard algorithmic techniques. However, fuzzy logic appears to offer a solution by encapsulating heuristic information on machine condition, in a similar manner to the method currently employed by a mechanic to diagnose problems. By representing operational tolerance windows of the machine parameters (Figure 2) in the form of fuzzy sets, the errors associated with individual parameters can be quantified and transformed into the fuzzy domain.

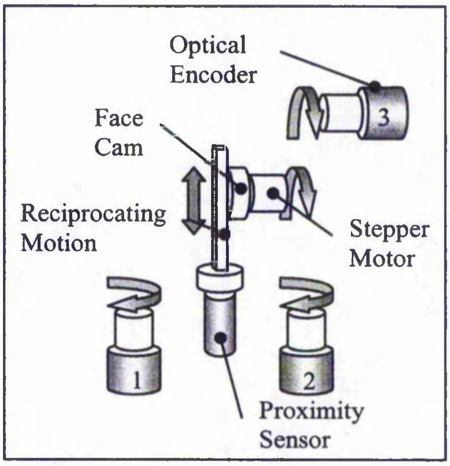


Figure 1. Test rig schematic

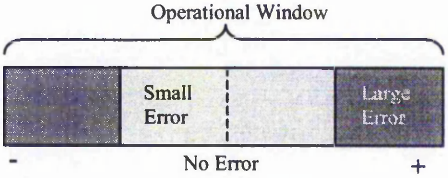


Figure 2. An operational window

This permits the use of the fuzzy rule base to consolidate the input variables into a single (defuzzified) value representing overall machine state (Figure 3). A new 'matrix' method [14] has been developed to overcome some of the shortcomings associated with other inference techniques (such as min-max, additive), when tackling this type of risk assessment problem. These shortcomings include the lack of ability to consider all rules when formulating the defuzzified result and to provide predictable influence of each rule. Nevertheless, implementation of the proposed fuzzy condition monitoring system requires due consideration for application within a real-time operating environment.

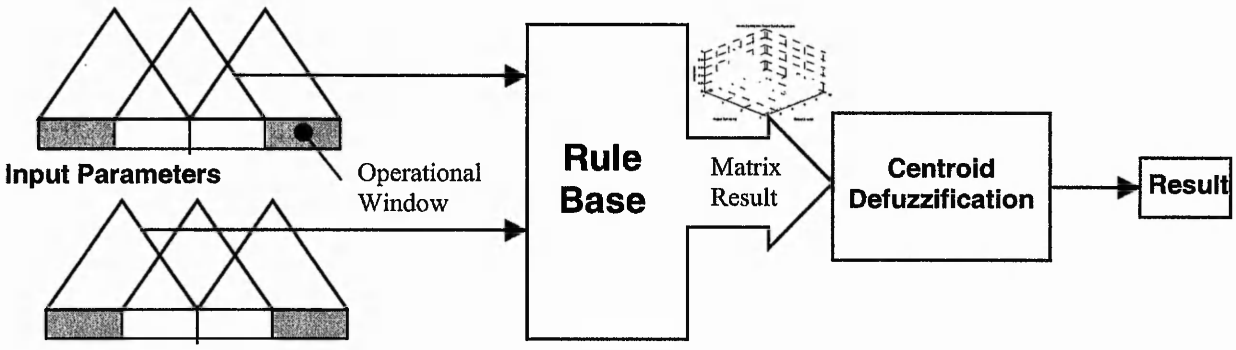


Figure 3. The Fuzzy Approach

3 Real Time Implementation

In software simulations, lengthy calculation time between computational iterations begets only inconvenience. However, in real-time systems, highest priority is given to sampled data and its integrity must be maintained at all costs. When data is sampled from the outside world, it is imperative that no data is lost at this sampling stage, as this may invalidate any calculations, assumptions or predictions made.

Three features are commonly present in condition monitoring systems: sampling, logging, and data display. By adding inference in the form of fuzzy logic to a system, computational requirements are necessarily increased. Using the programming 'FOR' loop to control software processes, the PC is restrained within tasks until completion. Consequently, data integrity is compromised by the increasing demands of these tasks. Parallel processing is one way of resolving this problem by allowing separate processors to deal with separate parts of the process. Each task can have it's own processor and memory, thus preventing data loss. One major drawback in this approach can be prohibitively expensive both in development time and in cost. A more realistic approach for producing a cost-effective solution is multi-tasking; this is where a single processor is shared amongst a number of concurrent processes, in such a way that no one process completely takes over. This is achieved by making each process aware of its current state and the next planned state so that control can be relinquished without loss of data.

In many multi-tasking systems, it is possible for a processor to move between tasks freely. Control between tasks is implemented by a technique known as time slicing/sharing. Time Slicing (TS) allocates a small amount of CPU time to each process as required so that all are allowed to progress; this is obviously much slower than each task having its own processor. Multi-tasking of a PC usually takes place at the Operating System (OS) level and is usually outside the grasp of an average C programmer. Nevertheless, through adaptation, the same techniques can be applied at a higher operational level, i.e. in a condition monitoring system software.

The packaging process being measured operates in a cyclic fashion; information is logged and displayed once per cycle after being collated for the whole of the previous cycle. Data sampling has highest priority in this system and must be performed to ensure that sampling rate is never compromised. The rest of the program must utilise the remaining unused processor time. Unfortunately, computationally intensive processes such as graphics or time-consuming data logging cannot be completed between samples. Time slicing splits these 'troublesome' processes into small task packets and executes them within available time slots (Figure 4). This method is applicable as graphics display or data logging does not have to be completed between the sampling intervals, updates of once per machine cycle will meet the requirements of this process.

Fuzzy logic is not particularly computing intensive, but as the size of the rule base increases the program performance will suffer. In a real time system, it would not be possible to perform all of the fuzzy logic tasks within the sampling time (assuming the sample rate is high - measured in milliseconds). In the application considered here, the sample rate should be set high at 8ms, a necessity in fast process control or condition monitoring. If the fuzzy logic inference is required to make a decision every sample then the TS architecture offers no improvements in performance. However, if it is necessary to collate many data points before invoking the fuzzy logic then TS can offer a solution. As many manufacturing and industrial processes are cyclic in nature as opposed to continuous processes such as chemical production, this method can be considered suitably generic for a wide range of industrial applications.

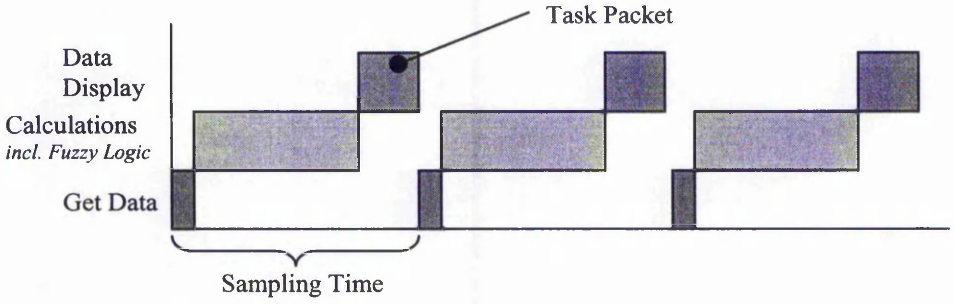


Figure 4. Time Slicing into task packets

The difficulties presented by the TS approach are not how well the task has been performed, but to do so simply, elegantly and robustly. There are some important criteria for consideration. A function must be self-aware of its own progress/state, therefore, it must be self-contained, so that it can be called from anywhere else within the program without any external knowledge of the progress of execution. This eliminates the need for a complex web of variables and pointers describing the possible states of software functions, hence providing a simple but robust solution. This feature will also allow for a certain amount of nestability (i.e. TS functions calling other TS functions), which is important for maintaining good program structure. It should also be noted that the computational effort of deducing the next process state must not be so time consuming as to negate the advantages of time slicing. The biggest obstacle presented by the TS method is developing the software in such a way that functions can be exited from upon part completion and then returned to in the next available CPU time slot to continue execution. The solution is to effectively change the program execution dynamically. The path through TS architecture depends on the progress of individually called TS functions, without using pre-defined loops. The effect of this method is to create a dynamic software architecture. Although the functional programming structure is easily recognisable, the program execution path becomes more convoluted but more efficient in the task at hand.

4 Results

The time slicing method for software development was implemented and tested on the condition monitoring test rig. By associating fuzzy input variables with possible machine parameters, a good appraisal of system performance was obtained. The parameters identified were relative offsets between the three encoders, absolute value of the proximity sensor, and a derived relationship between the proximity sensor and the encoders. These were considered good indicators for deducing overall machine condition. The sensitivity of the fuzzy inference engine to the variations of individual parameters was achieved by setting the fuzzy input domains (operational windows) to an estimation of critical tolerance. Hence, the values obtained from the proximity sensor were sensitised to follow small changes in alignment of the reciprocating device.

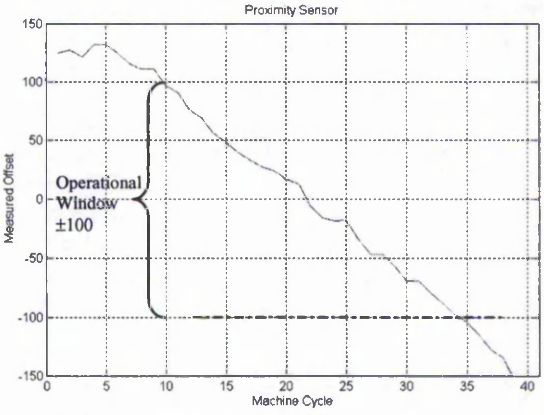


Figure 5. Proximity Sensor Measurements

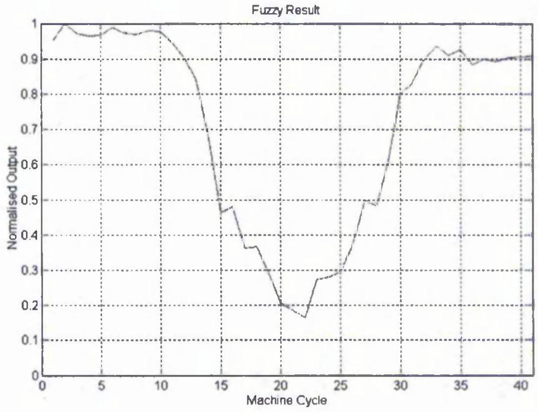


Figure 6. Defuzzified Result

Figures 5 & 6 show the effect on the fuzzy outcome from changes in the proximity sensor from an experimental run of the test rig for 41 simulated machine cycles. The synchronisation error of the optical encoders remained constant during these cycles. The operational window of the proximity sensor was set at ±100 (arbitrary units, proportional to a measured distance of approximately 0.2mm); intersected at cycle 10 and cycle 34 (Figure 5).

As can be seen, the fuzzy result (Figure 6) is unchanged whilst measured values are outside the operational window, cycles 0-9 and 35-41. Once measurements are received within the operational window, the fuzzy result is able to track the changes, smoothly and representatively. Variations in the fuzzy result contour are indicative of measurement noise present in the instrumentation.

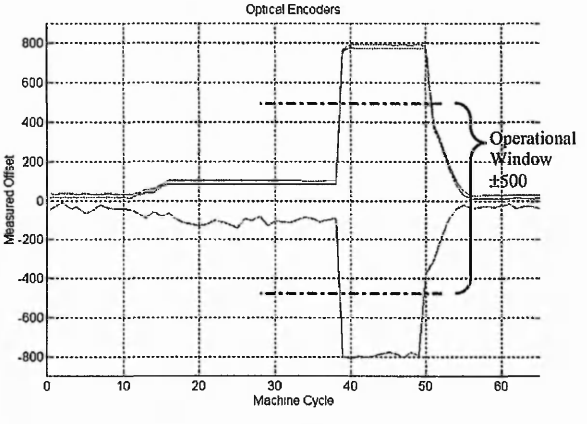


Figure 7. Optical Encoder Offsets

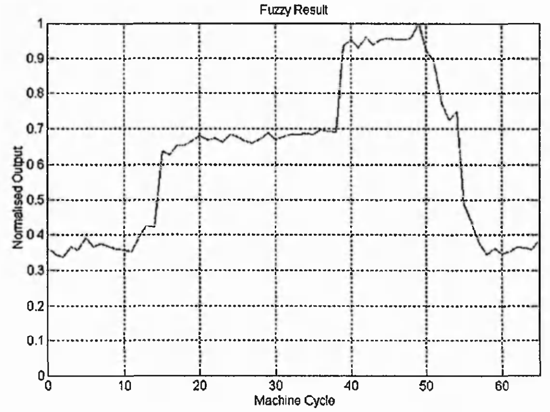


Figure 8. Defuzzified result

Figures 7 & 8 show the effect of changes of offset between optical encoders. Within the test run of 65 cycles, a synchronisation error is introduced into encoder 1. This can be seen in the three traces of Figure 7 at cycle 10; the error also increasing again at cycle 38. The positive traces are the offsets between encoder 1 and encoder 2 and between encoder 1 and encoder 3; the negative trace is the calculated estimate of the offset between the proximity sensor and encoder 1. The operational window of the encoders was set at ±500.

Figure 8 shows the fuzzy result is able to track changes in the offsets and is not confused by the positive and negative values of input parameters. At cycle 38, the encoder offset exceeds the operational window and therefore the fuzzy response flattens off. However in practical situations, the result would have already triggered an action as the machine condition value has moved above its safe steady state value (0.4 in this example). It should be noted that beyond the operational window the actual value is irrelevant, as the parameter concerned is already in failure.

Based on these results, the matrix inference technique appears to provide an accurate and useful means of determining overall machine condition. This therefore provides the groundwork to the next stage of research - interpreting these results to allow suggestions to be made on corrective actions.

5 Conclusions

When considering solutions for manufacturing problems, cost-effective condition monitoring has a large appeal. This is especially relevant when the processes under examination are not of high premium, and expensive technological solutions do not provide a reasonable price/performance ratio to the process in question. Therefore, the scope for 'off-the-shelf' technology is broad, covering many manufacturing sectors with packaging being one example. By employing a standard PC, utilising a custom-built interface and with in-house developed software, the work presented here shows a method for implementing a cost-effective solution to many condition monitoring applications.

However, if sample by sample consideration is required then fast, dedicated processors with optimised code are the only answer. Time slicing offers a cost-effective solution for assimilating cyclic information only. This method does not allow a slow processor to outperform faster ones. Nevertheless, it does produce a sound methodology for the design of a real-time condition monitoring system, allowing the incorporation of additional fuzzy logic elements without sacrificing the overall performance.

The fuzzy logic has been implemented within a real-time condition monitoring system. In utilising a matrix inference technique for determining rule output, the results confirm that this approach is valid for risk assessment problems.

The fuzzy condition monitoring system discussed here provides the stepping stone to the development of a fully intelligent condition monitoring system for packaging and other processes. This approach presents itself as a robust and inexpensive means of tackling the problems of wastage and maintenance management in the modern manufacturing environment.

6 References

[1] Byrne G, Dornfeld D, Inasaki I, Ketteler G, Konig W, Teti R. 1995, Tool condition monitoring (TCM) - the status of research and industrial application. *CIRP Annals - Manufacturing Technology*, **Vol.44**, No.2, pp.541-567

[2] Plantenberg D H, Lai E and Jeffries M. 1997, A Novel Approach to Reduce Packaging Process Down-Time *Advances in Materials and Processing Technologies, AMPT '97* **Vol. 2**, 630-635.

[3] Chow M, Magnum P M and Yee S O. 1991, A neural network approach to real-time condition monitoring of 7induction motors. *IEEE Trans. On Industrial Electronics,* **Vol. 38,** 6, 448-453.

[4] Aluindigue I E et al. 1993, Monitoring and diagnosis of rolling element bearings using artificial neural networks. *IEEE Trans. On Industrial Electronics*, **Vol. 40**, 2, 209-216.

[5] Lai E, Plantenberg D H, Grant Z R, Hull J B. 1996, Condition Monitoring of plants and equipment with the aid of neural networks. *Proc. Symposium on Mechaincs in Design, University Of Toronto, Canada*, **6-9 May**, 515-522.

[6] Hong G S, RahmanM, Zhou Q. 1996. Using neural network for tool condition monitoring based on wavelet decomposition *International Journal of Machine Tools & Manufacture*, **Vol.36**, No.5, pp.551-566

[7] Javed M A, Hope A D, Littlefair G, Adradi D, Smith G T, Rao B K N, 1996, On-line tool condition monitoring using artificial neural networks *Insight: Non-Destructive Testing and Condition Monitoring*, **Vol.38,** No.5, pp.351-354

[8] Das S, Roy R, Chattopadhyay AB, 1996, Evaluation of wear of turning carbide inserts using neural networks *International Journal of Machine Tools & Manufacture,* **Vol.36**, No.7, pp.789-797

[9] Das S, Bandyopadhyay P P, Chattopadhyay A B, 1997, Neural-networks-based tool wear monitoring in turning medium carbon steel using a coated carbide tool, *Journal of Materials Processing Technology*, **Vol.63**, No.1-3, pp.187-192

[10] Li S and Elbestawi M A. 1996, Tool Condition Monitoring in Machining by Fuzzy Neural Networks, *Journal of Dynamic Systems, Measurement, and Control, Transactions of the ASME,* **Vol. 118**, 665-672.

[11] Li S, Elbestawi M A, 1996, Fuzzy clustering for automated tool condition monitoring in machining, *Mechanical Systems & Signal Processing*, **Vol.10**, No.5, pp.533-550

[12] Rondeau L, Ruelas R, Levrat L, Lamotte M. 1997, A defuzzification method respecting the fuzzification *Fuzzy Sets and Systems* **86,** 311-320

[13] Pressman R S, 1987. Software Engineering - A Practitioners Approach 2nd Edition, McGraw-Hill International Editions, p367-402

[14] Jeffries M, Lai E, Plantenberg D H, Hull J B. 1997, A Fuzzy Approach to the Condition Monitoring of a Packaging Plant. Accepted for presentation at *6th International Scientific Conference "Achievements in Mechanical & Materials Engineering" November 28th - December3rd 1997, Poland/Hungary.*

AN EXTENDED KALMAN FILTER TO ASSIMILATE STOCHASTIC FACTORS OF ULTRASOUND MEASUREMENT IN CONTAINER FILLING OPERATIONS

M. Jeffries, E. Lai, J.B. Hull
Department of Mechanical and Manufacturing Engineering, The Nottingham Trent University, UK. Email: martyn.jeffries@ntu.ac.uk or eugene.lai@ntu.ac.uk

ABSTRACT

Recently, ultrasound has begun to show its potential in providing a non-invasive and reliable technique for measuring fluid level in container filling operations. Whilst the theoretical aspects are well established, the design of ultrasound-based control systems for regulating bottle filling operations is severely constrained by a number of obstacles. These include transducer design parameters and the characteristics of the carrier medium, which can interact in a highly non-linear manner. This inevitably leads to uncertainties in measured values from the stochastic effects of the measuring environment. Unless these effects can be quantified, a model-based adaptive control methodology can not be developed.

By utilising an extended Kalman filter, the stochastic factors can be accommodated and non-linearities encompassed, permitting a model for the filling process to be attained. This provides the foundation for the development of an effective adaptive control strategy for bottle filling operations, thus overcoming the shortfalls of current mechanical methods.

INTRODUCTION

The bottling industry has relied on mechanical means to facilitate control of liquid levels ever since its inception. However, as the need to reduce waste becomes an increasingly dominant force in the modern manufacturing world, the search for a reliable and accurate method to control container filling has been brought to the fore. Although modern bottling plants are sophisticated in mechanical design, the employed principles no longer provide sufficient scope for the optimisation necessary to meet waste reduction targets. However, the introduction of modern control theory and new technologies could allow for further improvements to be made (Hull et al, 1995).

The wastage from bottling plant arises largely within the initial 'start-up' period of the machine. This can be attributed to a number of factors, these predominantly being filling level and product composition, i.e. satisfactory carbonation, pasteurisation etc. Within current operating procedure, the main control effort is focused on ensuring satisfactory product, with only inaccurate bulk measurement techniques for fill level comparison. Therefore, additional control installed at the filling valve would improve delivery efficiency, allowing possibilities of reducing the time duration required to achieve satisfactory plant steady state (Ridgway et al, 1996). This would require dynamic measurement of the filling level, a challenging task in view of the fast process time (< 4 seconds per bottle/can) and the absolute requirement for non-contamination of the food products.

Griffin et al (1997) has highlighted the applicability of a low-power ultrasound technique for bottle filling. The approach affords the scope for non-contact measurement of liquid level using air-transmission transducers placed within the filling valve. Although theoretical aspects are largely understood, a number of practical obstacles would have to be overcome for satisfactory implementation. The main obstacle is the electronic and mechanical design of the ultrasound transducer system. Utilising air transmission necessitates the use of relatively low frequency of 156KHz to overcome the high attenuation characteristic of gaseous media. This imparts a need on the electronics to measure the pulse-echo timing to a high accuracy. Accuracy to one wavelength of the ultrasound at this frequency would give a distance measurement to within 2.2mm (assuming the speed of sound is 343 m/s). Despite the ability of modern electronics to achieve and exceed this kind of accuracy, a number of physical obstacles conspire to undermine this target - these can be attributed to the environment within which measurements will be made.

The bottle filling environment is relatively hostile to ultrasound measurement techniques due to several essential process requirements. The bottles intended for carbonated products have a pressurised carbon dioxide environment above the liquid surface to prevent effervescence of liquified carbon dioxide during filling. This will have inevitable impact on the ultrasound, and would require investigation into the resultant properties of this atmosphere. Other non-carbonated products have to be flash pasteurised (rapidly heated) immediately before bottling in order to remove possible bacterial contamination. However, the resultant thermals rising within the bottle from the warm liquid can cause diffraction and scattering of the ultrasound waves, thus reducing accuracy and certainty of measurements carried out therein. In both situations turbulence at the liquid surface during filling will disrupt the sound reflecting surface, further degrading measurements. All these stochastic influences will have to be compensated for during real-time measurements of the filling process.

This paper seeks to demonstrate the potential of utilising an extended Kalman filter incorporating a non-linear bottle model to allow a control strategy to be developed. This method will encompass the stochastic factors inherently present within the ultrasound technique and hence provide a reliable and effective means of improving the efficiency of current mechanically based bottle filling plant.

MATHEMATICAL DESCRIPTION OF BOTTLE SHAPE

In order to use a Kalman filter as a successful measurement filter, it is necessary to have some a priori knowledge of the process under scrutiny. This includes the expected noise statistics, and a predictor model for estimation of the next process state. Although on first inspection the model appears only to be a simple single variable (fill height) system, the complex shape of the bottle causes the fill height to vary in a non-linear manner - even for a fixed volume flow rate. This has two effects on the proposed method: firstly, a mathematical description of bottle shape, or at least for the critical neck section, must be found; secondly, an extended Kalman filter must be used to linearise this inevitable non-linear description.

Bottle shape is extremely varied, both across the industry and within a single manufacturer. Nevertheless, the overall shape can be categorised into a number of simple primitives (Figure 1.). The first and simplest is the cylinder, creating a purely linear association between volume and fill level. The next primitive - the cone - has three variants, linear conical, convex conical and concave conical. These can be used to describe the neck region on a large number of bottles, however the relationship between volume and fill height is non-linear.

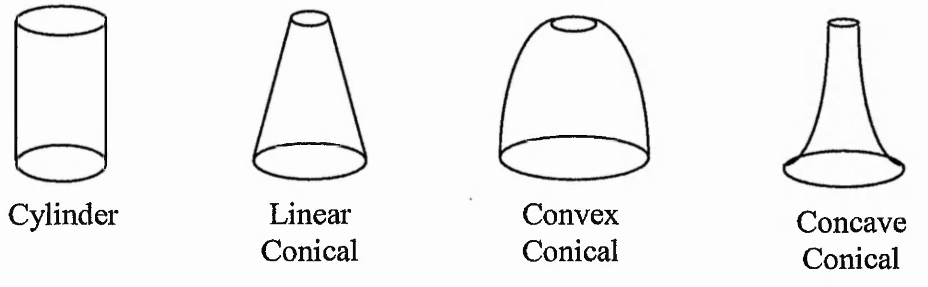


Figure 1.The basic primitives of bottle shape

By forming an approximate mathematical description of the bottle curvature, a volumetric representation can be gained by performing a volume of revolution integration about the vertical axis; this creates a Characteristic Equation (CE) for each shape. This equation relates volume to fill height and is independent of bottle size. For example, the most common shape used for bottle necks is the linear conical, which can be described by the following characteristic equation,

$$\frac{x^3}{3h^2} - \frac{x^2}{h} + x - \frac{V}{\pi r^2} = 0 \qquad (1)$$

where

x,	*fill height*
V,	*filled volume*
r,	*radius of cone*

h, *height of cone*

This can be rearranged to show fill height in terms of the filled volume. (Eq. 2)

$$x = \left[\frac{3Vh^2}{\pi r^2} - h^3 \right]^{\frac{1}{3}} + h \qquad (2)$$

As can be seen the result is in a non-linear form even for a relatively simple shape (Figure 1b). If the bottle characteristic equations are to be used to model the filling process, then an extended Kalman filter, which can accommodate non-linear processes, is a necessity to provide any realistic or reliable means of extracting a control signal from the stochastic environment.

EXTENDED KALMAN FILTER (EKF)

Since Kalman (1960) and Kalman-Bucy (1961), there has been an accessible solution to the problem of state estimation from noisy measurements. The Kalman filter is easy to implement on digital computers and is well recognised as the optimal method for use with linear systems (Grimble and Johnson, 1988). The discrete variant of the filter can be easily extended to allow the application in non-linear systems. By linearising the non-linear internal model across the current time step, an equivalent linear function can be generated allowing estimation of the next state. However, extra care must be taken as the resultant filter is only an approximation to optimality and stability is not guaranteed.

The discrete Kalman filter can be described by five equations (Eq. 3-7), using matrix (state space) notation.

Kalman Gain Update

$$K_k = \frac{P_k^- H_k^T}{(H_k P_k^- H_k^T + R_k)} \qquad (3)$$

State Update

$$\hat{x}_k = \hat{x}_k^- + K_k (z_k - H_k \hat{x}_k^-) \qquad (4)$$

Covariance Update

$$P_k = (I - K_k H_k) P_k^- \qquad (5)$$

State Estimation

$$\hat{x}_{k+1} = \Phi \hat{x}_k \qquad (6)$$

Covariance Estimation

$$P_{k+1}^- = \Phi P_k \Phi^T + Q_k \qquad (7)$$

where,

R *measurement error matrix - expected noise from measurement represented as a variance about a zero mean.*
Q *plant/system error matrix - as R except noise from plant or system*
P *error covariance matrix - difference between prediction and actual states*
$\hat{x}$ *state estimate matrix - state variables*
Φ *state transition matrix - the process model*
K *Kalman gain matrix - balances measurements to predictions based on P*
H *observation matrix - transforms state variables to observation variables. If state variables are directly measured then H=I (identity matrix).*

The superscript $^-$ represents the matrix in the previous instant to that denoted in the subscript

Computing the equations in the order presented here constitutes the discrete Kalman filter algorithm. The EKF is identical, except for the state transition matrix, Φ which becomes a linear representation of a non-linear process at the current time instant. In linear systems, Φ is time-invariant, whereas within non-linear systems it is critically time-dependent. This increases the computational overheads as it can not be pre-determined - a common procedure for linear systems.

MODELLING THE BOTTLING PROCESS

For initial analysis, modelling of the entire bottle shape is not necessary. By examining only the neck, the extended Kalman filter can be tested with the non-linear characteristic equations already formulated (Eqs. 1 & 2). Furthermore, the 'cylinder' (Figure 1.) can represent the body of most bottles. This linear form will present no difficulties for the Kalman filter and will prove valuable when full implementation on dynamic filling is required - allowing a degree of stabilisation of the filter parameters before the critical non-linear neck region where the fill level is ratified.

Due to current mechanical restrictions in the real bottling plant, additional measurement data, apart from fill height, cannot be obtained. This reduces the system state space to a single dimension. Should investigations into the control method lead to the conclusion that measurement of additional variables (e.g. flow rate) may help to improve the accuracy of the model, the state space approach allows easy integration of new parameters, without undermining the established work.

Three criteria need to be met in order to use the Kalman filter: knowledge of the measurement noise variance (matrix *R*), knowledge of the plant noise variance(matrix *Q*) and a system model for the state transition matrix. The characteristic equation (CE) can be used to generate a system model by defining the change in fill height (Eq. 8), then with substitution of the shape CE and using $V_{k-1}=V_k-\Delta V$, to reach Eq. 9.

$$\Delta x = x_k - x_{k-1} \tag{8}$$

$$\Delta x = x - h + \frac{1}{\pi^{1/3} r^{2/3}} \left[\pi r^2 (h-x)^3 + 3h^2 \Delta V \right]^{1/3} \tag{9}$$

Eq. 9 defines the transition of the state variable x (Δx), in terms of the current fill height, x , ΔV - change in volume (flow rate) and the shape definition parameters h & r. The equation is highly non-linear and justifies the selection of an extended Kalman filter.

In simulation study, the system model will be identical to the 'real system', as they should both use the best mathematical description of the plant or process. This also means Q should have a variance of zero, allowing the Kalman filter absolute confidence in the internal model (Kalman gain matrix becomes equal to zero). The difference between R and Q decides whether the Kalman filter 'trusts' the internal model or the external measurements. If R is higher than Q then more weight is given to the internal model, if Q is larger than R then the measurements are given more consideration. In this application R will be much larger than Q, leading to decreased use of measurements to identify system states. Q will increase if the bottle shape equations do not adequately represent the true shape, thus the expected fill height. This will come especially prevalent for very complex shapes such as those found in some moulded plastic bottles.

RESULTS

The combined influence of all factors leading to noise in the measurement signal can be approximated to white noise. This can easily be represented in the modelling of the system and does not require further modifications to the Kalman filter. To fully test the capabilities of the filter, values of R were chosen to represent situations of high levels of white noise content in the measurement signal. A low, but non-zero, value of Q was used to increase the influence of measurement in state identification in the presence of an exact state transition model - attainable due to both kalman model and the simulation using the same characteristic equations to describe the process.

Implemented in the C programming language, the simulation creates a data log file that can be further processed or displayed by programs such as Matlab®. The shape chosen for simulation was a 250mm high cone filled to 200mm. Measurement noise was tested at two levels, ±10mm (Figure 2a) and ±25mm (Figure 3a). Filter iteration time was chosen to be 0.2 seconds and a volume flow rate calculated to fill to the required height in 120 iterations.

Figures 2a and 3a contain three plots each: characteristic equation model, measurement (model plus noise) and Kalman estimate. Although the measurement is clearly visible due to the presence of white noise, the difference between CE model value and the Kalman estimate is barely distinguishable. This is due in part to the success of the EKF and also to the scale needed to represent the 200mm full range. The error can be seen in more details in figures 2b and 3b. In both simulations, the noise evident in state estimation error is drastically reduced from the original measurement signal. Analysis of the Kalman gains after 120 iterations reveals low values of around 0.02 to 0.04, indicating prodigious use of the internal model for calculating the fill height. This is to be expected with a high value of R;

nevertheless, these simulations prove the effectiveness of the EKF at linearising the non-linear bottle characteristic equation.

It can be noted that the average estimate error is lower for approximately the first 40-60 iterations, an unexpected result considering the prospect of improved estimation as the Kalman filter converges to steady state values. However, the linearised model of the non-linear equations is considerably more representative of the process during the first 40 iterations, less so as the curvature increases. Indicating that as rate of curvature in the characteristic equation increases, instabilities of the estimates can be expected, although not significant enough to undermine the proposed approach.

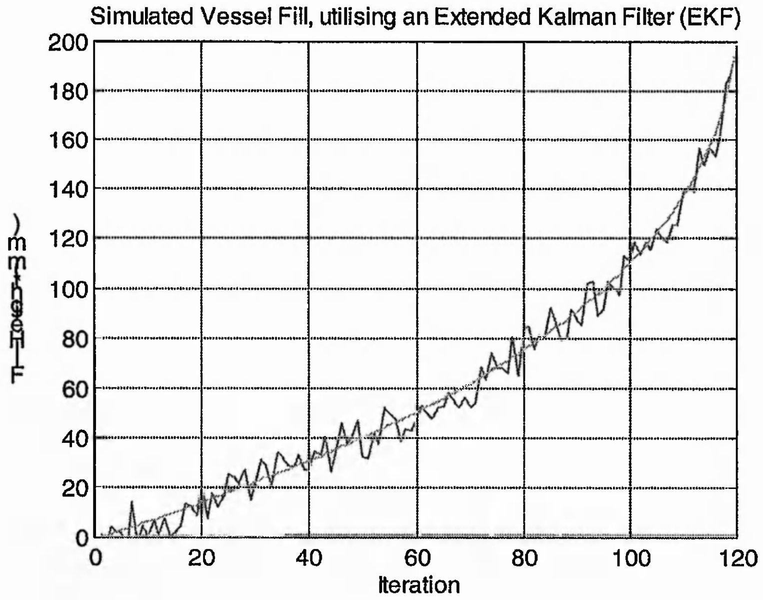


Figure 2a. Simulation results with white noise at ±10mm

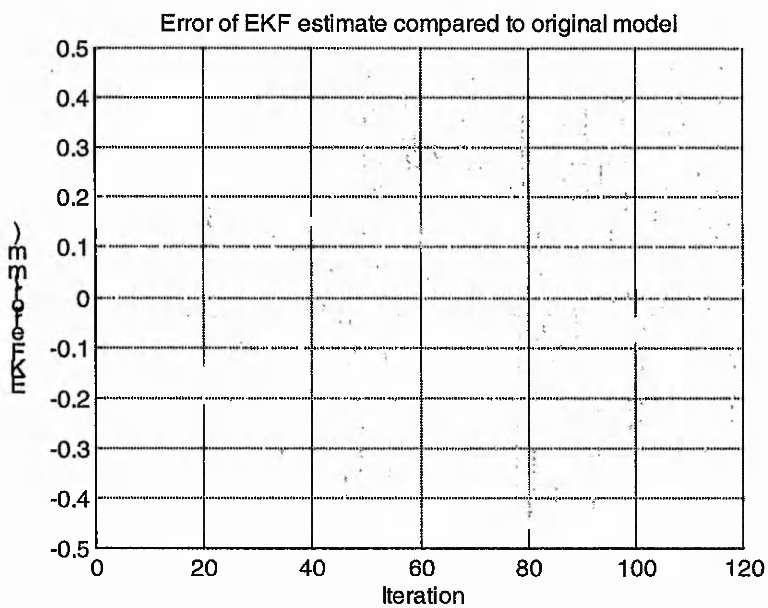


Figure 2b. State estimation error (noise ±10mm).

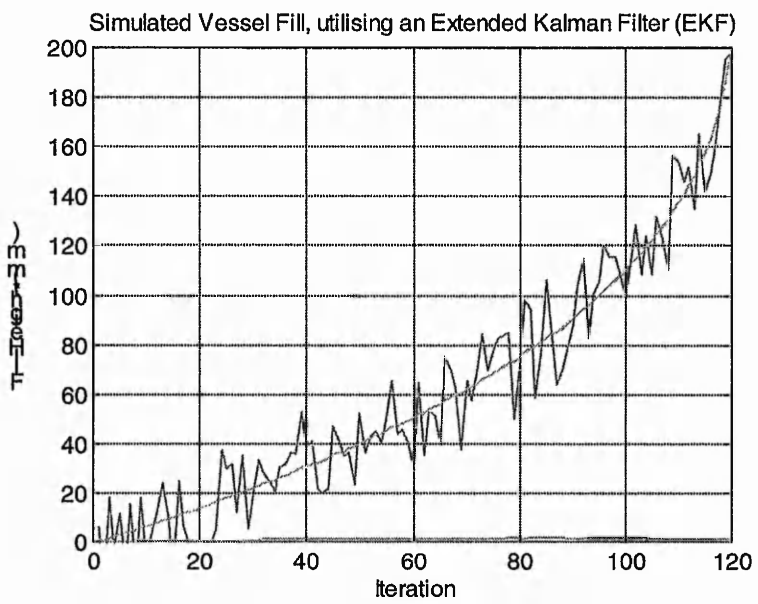


Figure 3a. Simulation results with white noise at ±25mm

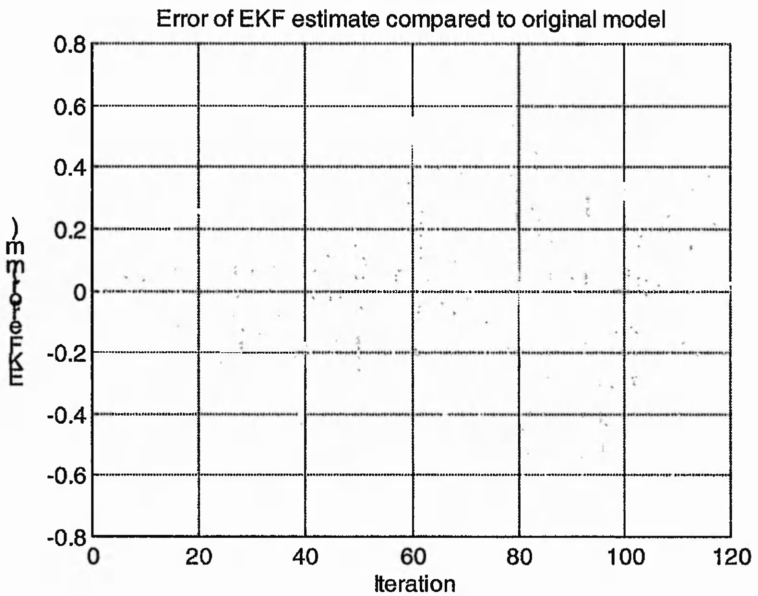


Figure 3b. State estimation error (noise ±25mm).

CONCLUSIONS

The application of ultrasound to the problems of fill level control in modern bottle filling plant has the potential to make significant impact into waste management in the industry. However, the development of the complementary control strategy will be the key to its success. Although a relatively simple process, the bottling plant introduces a number factors causing stochastic effects on measurement signals. The use of a Kalman filter is well-tested solution to the need for accurate parameter identification in a real time stochastic environment.

The non-linear description of bottle shape introduces the necessity to implement an Extended Kalman Filter. This allows linearisation of the equations over the short term, permitting the use of the standard 5-stage Kalman filter algorithm. However, the stability of such filters is not guaranteed and must be tested against the non-linear system in question.

Using a non-linear characteristic equation to describe a common shape of bottle neck, the developed filter was tested using a simulated bottle fill. Introducing two levels of white noise into the measurement signal, the EKF output was compared to the original model. The results show that noise is drastically reduced giving a substantially more reliable value for fill height. This will provide a sound basis for the development of a complete adaptive control strategy for container filling operations.

The outcome of this initial investigation verifies the hypothesis of using an extended Kalman filter to improve the capabilities of control strategies intent on using ultrasound level measurement within noisy environments. The benefits providing increased scope for reducing the current wastage witnessed by the bottling industry.

REFERENCES

Griffin, S., Hull, J.B. and Lai, E., 1997, "Development of a novel ultrasound monitoring system for container filling operations" *Proceedings of Achievements in Mechanical & Materials Engineering (AMME) '97*, Gliwice-Wisla, Poland, pp. 79-82.

Grimble, M.J. and Johnson, M.A., 1988, *Optimal Control and Stochastic Estimation: Theory and Applications Volume 2*, A Wiley-Interscience Publication, John Wiley & sons Ltd, Chichester.

Hull, J.B., Henthorn, K.S., Muumbo, A.M., 1995, "Controlling Waste in the Food Processing Industry", *Advances in Materials and Processing Technologies, AMPT95*, Dublin, pp31-39.

Kalman, R.E., 1960. "A New Approach to Linear Filtering and Prediction Problems", *Journal of Basic Engineering*, **82**, pp. 35-45.

Kalman, R.E., and Bucy, R.C., 1961 "New Results in Linear Filtering and Prediction Theory", *Journal of Basic Engineering*, **83**, pp. 95-108.

Ridgway, J.S., Henthorn, K.S., and Hull, J.B. 1997, "Controlling of Overfilling in Food Processing" *Ultrasound in Food Processing*, M.J. Povey, ed., Chapman & Hall

A New Approach to Process Control for Bottling Plant

M Jeffries, E Lai and J B Hull

Department of Mechanical and Manufacturing Engineering, The Nottingham Trent University, Nottingham NG1 4BU, United Kingdom. Fax +44 115 9486506

Abstract

The bottling industry has relied on mechanical means to facilitate control of liquid levels ever since its inception. However, as the need to reduce waste becomes an increasingly dominant force in the modern manufacturing world, the search for a reliable and accurate method for the control of container filling has been brought to the fore. Although modern bottling plants are sophisticated in mechanical design, the employed principles no longer provide sufficient scope for the optimisation necessary to meet waste reduction targets. However, the introduction of modern control theory and new technologies could allow for further improvements to be made. Using a Kalman Filter to reduce limitations in the sensor technology (due to less-than-ideal placement) will allow measurement of the liquid level during filling of individual bottles. In addition, an overall controller for a multi-headed filling plant using fuzzy logic will allow for a robust approach, which will build on the advantages gained from using ultrasound technology to obtain not only liquid levels but also flow rates by means of Doppler shift of the returning signal.

The approach has great potential to improve the filling process, with implementation on a range of different configurations possible by utilising the rule-based inference engine of the fuzzy logic to provide overall control.

Keywords: Liquid Level Control, Kalman Filter, Fuzzy Logic, Ultrasound.

1. Introduction

Many areas of manufacturing and processing are currently focusing on optimisation and control technologies in order to improve performance and efficiency. In an increasingly competitive market place, an industry showing a clear need and strong desire to improve is that of the bottle filling of beverages. They are constrained by regulations which ensure bottles are filled to within a specified accuracy of the intentional content, but due to the mainly mechanical systems employed, sensitivity is low and consistent overfilling is currently the most reliable method of meeting these regulations. However, overfilling produces large amount of unnecessary waste product from the addition of small amounts over many bottles, despite its seeming benefits to the consumer, this inevitably drives up the costs. Although the mechanical systems employed are reasonably sophisticated in their design, after many years of evolutionary modifications little scope remains for further improvement. The most common current method for establishing fill height and hence content volume is that of the pressure balance shut off valve. Utilising a tube projecting vertically from the base of the valve and terminating at the desired fill level, filling ceases when the carbon dioxide used to pressurise the system can no longer escape up the tube when liquid covers the entry port; the valve closes due to the pressure change. The setting of fill height is achieved using different lengths of tube. Some systems, notably those used for cans, use a floatation ball at the end of the tube, which again stops the pressure balancing once level is achieved.

Bottling plants do contain areas of sophisticated control, namely in product composition (i.e. satisfactory carbonation, pasteurisation etc.), but control of level is reduced to bulk measurement and rejection systems for unsatisfactory bottles – closed loop control is not common. In addition, a large amount of wastage is witnessed within the 'start-up' period of the plant, when technicians attempt to achieve the correct balance of pressure and flow rates to meet fill level targets. Without direct measurement, the meeting of performance tolerances can take up to thirty minutes.

However, the introduction of modern control theory and new technologies could allow for further improvements to be made [1]. Installing additional closed loop control would improve delivery efficiency, allowing possibilities of reducing the time duration required to achieve satisfactory plant steady state. Although placement of sensors outside the valve has been shown to improve performance [2], evidence of process harmonics is observable in this approach. This indicates better performance could be achieved by placement of sensors within the valve or bottle, replacing or supporting the current pressure tube method – a challenging task in view of the fast process time (~ 5 seconds per bottle/can) and the absolute requirement for non-contamination of the food products. This indicates the need for an 'intelligent' control strategy and innovative use of sensor technology.

This paper seeks to demonstrate such an approach. Although the main emphasis is on the control aspects of the problem, consideration is also given to the potential of ultrasound sensor technologies to provide the necessary information to ascertain fill height dynamically.

2. Measurement

The solutions available for liquid level measurement are numerous and varied, using physical parameter changes such as force, pressure, electrical effects or indirect measurements using microwaves, infrared, nuclear, thermal and ultrasound [3]. Nevertheless, the suitability of the various methods is restricted in this application by the design of the filling process. Currently, measurements are often made at a position away from the filling valve, giving an indication on overall fill level and allowing rejection of under-filled bottles, but the information does not feedback to the filling valve. A non-contact method is preferred because the bottles are moving past the sensor at relatively high speeds. It is clear that to improve the bottling process closed-loop control of some kind must be used, and placing the sensor as close to the actual filling point will reduce any process lag. Given the flexibility to design a plant around a sensor technology almost all the previously mentioned methods are viable. However applying a new approach to an existing plant will result in both less sensor choice and a reduction of the capabilities of the sensors. This will necessitate the control system accommodating imperfections in the sensor technology.

Unlike some processes where a variety of sensors can be placed to provide a wide range of information, the filling process will be limited to a single sensor for level measurements due to space restrictions. This further reduces the effectiveness of state estimation because plant parameters such as pressure and flow rate have a significant effect on the fill. However, using ultrasound technology a possibility is afforded the control system designer – the ability to measure two parameters from a single sensor. By not only measuring the time of flight to obtain the liquid level, but also looking at the Doppler shift of the returning signal, it is possible to infer the speed of the moving liquid surface and hence to some extent the flow rate of liquid into the bottle (Figure 1). Nevertheless, a degree of intelligence is required by the control system due to unavoidable factors such as bottle shape and turbulence within the bottle. Moreover, the presence of pressurised carbon dioxide within the bottling environment reduces the effectiveness of the ultrasound due to the signal attenuation characteristics [4]. This will necessitate 'filtering' the measurement signal before control decisions can be made.

3. Filtering

A well-tested solution to the problem of imperfect or noisy measurements is the Kalman Filter [5,6]. By filtering measurements before the control system has chance to make decisions, inadequacies in the accuracy of sensors can be de-coupled from control actions.

The Kalman Filter requires two key ingredients: a model of the process for comparison against incoming measurements and statistical knowledge of parameter errors including those within the model. The errors are represented in covariance matrices and are used to verify the original measurements or the model predictions as most representative of the process under scrutiny. A Kalman gain matrix balances the two sources to provide a single robust output. Manipulation of the internal filter states is achieved through five equations which when implemented on a digital computer characterise a discrete Kalman Filter algorithm (Eq^s. 1-5).

Although the Filter is recognised as being the optimal solution for linear stochastic systems [7], in real situations, areas of non-linearity can be expected. To overcome limitations, the Filter is 'extended' to form the Extended Kalman Filter (EKF); this modification allows a linear approximation for the non-linear process to be formed at the current iteration time step. The resultant filter is only an approximation to optimality and stability cannot be guaranteed. However, ensuring a small iteration time step will give the best representation of any non-linear system.

For the bottle filling operation, the internal filter model was chosen to represent flow rate and volume rather than fill height. To model fill height directly would result in non-linear transform for all shapes except cylinders, whereas a volume/flow rate model would be largely linear and independent of bottle shape. It would also allow more careful modelling of flow variations due to valve and plant characteristics. However to achieve this, two additional elements were required to transform height measurements to volume estimates and back again. These elements require an accurate model of the bottle shape and this is achieved by using pre-processed data look-up table of the relationship between height and volume. In addition, statistical errors in level measurement from the ultrasound sensor must be converted to represent changes in volume. However, as this relationship is not constant over the range of fill level, the representation of noise will not be optimal. Nevertheless, accurate assessment of measurement noise will be difficult at best due to the number of uncontrollable environmental factors such as CO_2 content or surface turbulence. Therefore performance should not be overly reduced from the relatively small variations in modelling errors.

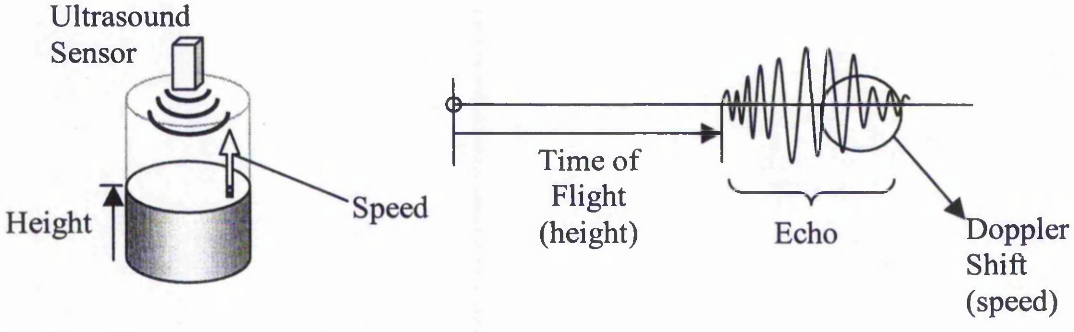


Figure 1. Approach to ultrasonic measurement of liquid level

Kalman Gain Update

$$K_k = \frac{P_k^- H_k^T}{(H_k P_k^- H_k^T + R_k)} \quad (1)$$

State Update

$$\hat{x} = \hat{x}_k^- + K_k(z_k - H_k \hat{x}_k^-) \quad (2)$$

Covariance Update

$$P_k = (I - K_k H_k) P_k^- \quad (3)$$

State Estimation

$$\hat{x}_{k+1} = \Phi \hat{x}_k \quad (4)$$

Covariance Estimation

$$P_{k+1}^- = \Phi P_k \Phi^T + Q_k \quad (5)$$

where,

- R *measurement error matrix – expected noise from measurement represented as a variance about a zero mean.*
- Q *plant/system error matrix – as R except noise from plant or system*
- P *error covariance matrix – difference between prediction and actual states*
- $\hat{x}$ *state estimate matrix – state variables*
- z *original measurement matrix – filter input*
- Φ *state transition matrix – the process model*
- K *Kalman gain matrix – balances measurements to predictions based on P*
- H *observation matrix – transforms state variables to observation variables. If state variables are directly measured then H=I (identity matrix).*

The superscript ⁻ represents the matrix in the previous iteration to that denoted in the subscript

4. Control

The majority of bottle filling plant do not fill one bottle at a time, but instead rely on a carousel of filling heads, allowing many bottles to be filled simultaneously in a continuous process. For example, in the soft drinks industry, carousels of 30 or more heads are commonly employed. At any given moment, no two bottles are at the same stage of filling, and hence, any controller must accommodate a large number of inputs and a variety of states. Plus, the process is fast with a fill time of around 4 to 5 seconds and thus control decisions have to be made in a short time scale.

Due to the stochastic nature of input variables, any overall controller must be robust in the presence of destabilising parameter variation. Also some degree of feedback to the Kalman Filter could allow improvement of the filter's internal model. Using traditional control methodology the process is difficult, especially with input variables showing cyclic significance due to the position on the carousel. However, Fuzzy Logic has the potential to not only simplify the system, but also provide a range of additional features, like in-built condition monitoring using techniques previously explored [8].

A significant factor in the bottling situation is the reliability of measurements. Given a fixed variance the Kalman Filter alone would be enough to compensate for sensor inaccuracies, however, the suitability of measurements changes depending on the fill level. For example, as filling begins, the turbulence at the bottom of the receiving vessel will be so great so as to prevent a surface forming for measurements; despite this some measurements can be expected. The reliability of these measurements, whether they are time of flight or Doppler shifts, can be brought into question as all measurements are taken under the assumption that a surface is being monitored.

Fuzzy Logic has become a popular tool for creating systems with the ability to quantify variables that have non-discrete states. This allows for increased robustness across transitions between states and a continuous description over the whole variable range. By using Fuzzy Logic sets to describe the stages of filling, the confidence of the measurements can be estimated. Using three sets: base, body and neck; the stages of filling can be represented robustly despite the lack of a fixed datum point (Figure 2.).

In the multiple filling head situations this approach is even more useful. Measurements from the main body of the bottle can be considered most reliable due to the straight sides (of many bottles) providing a fixed relationship between height and volume. Hence, flow rate can be estimated from the Doppler shift of the returning ultrasound signal, without the need to know the exact height, only the diameter of the vessel. The Fuzzy Logic inference can be used to feed this value to the controllers of valves whose own measurements are less reliable because of the current position of the fill level. This will allow continuous updating of flow rates for the whole plant even if only a percentage of the sensors are currently providing reliable measurements (Figure 3).

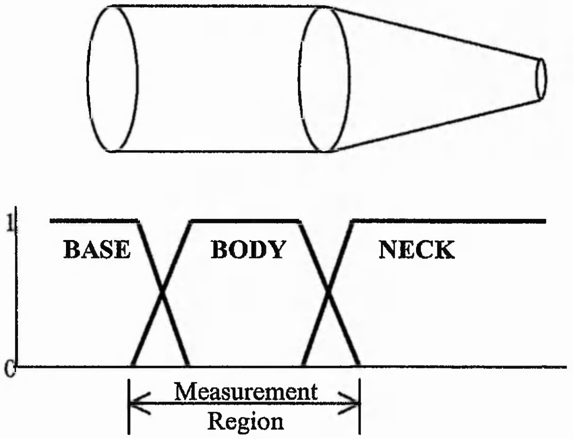


Figure 2. Fuzzy definition of bottle shape for measurement validation

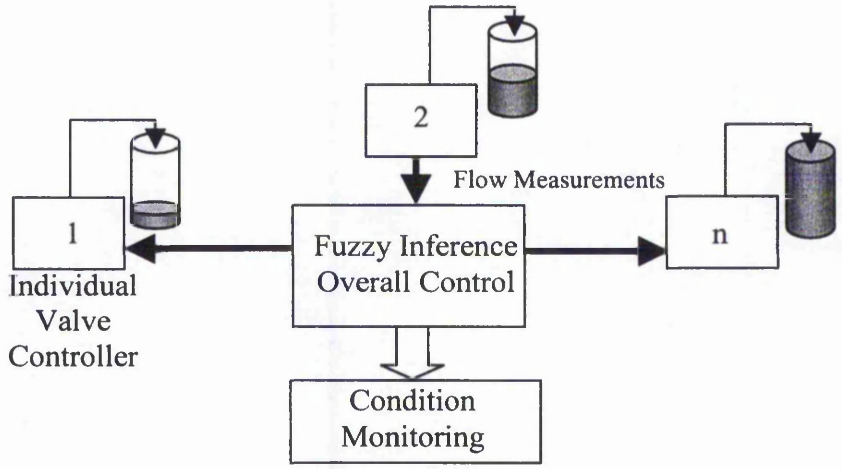


Figure 3. Schematic representation of Fuzzy Logic elements in multi-bottle control situation

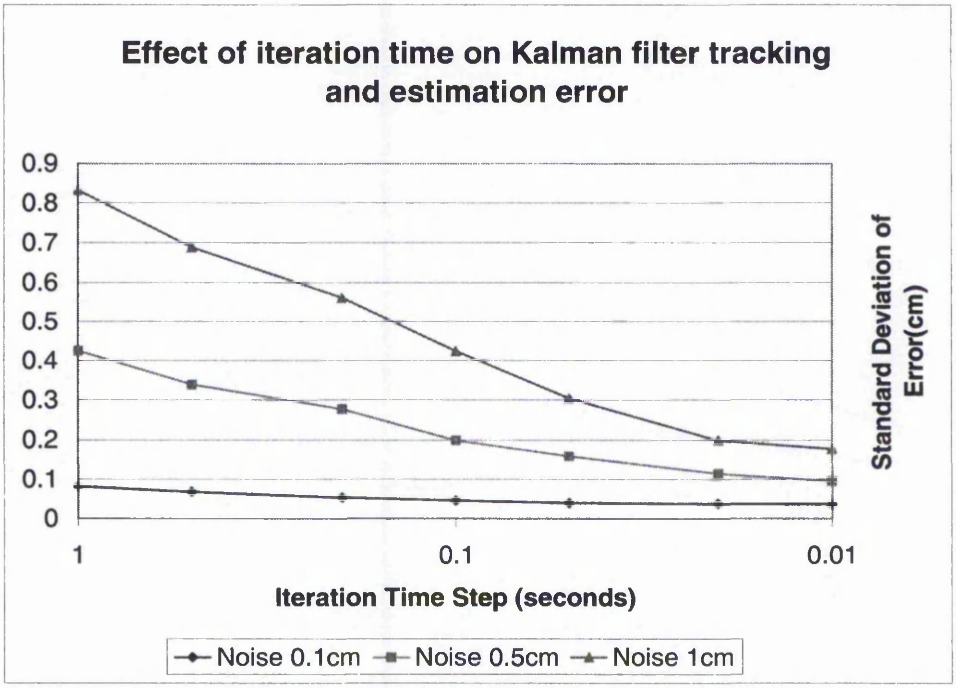


Figure 4. The influence of iteration time on the performance of a Kalman filter

5. Results of Modelling and Simulation

To test the proposed strategy before implementation on an actual filling assembly, a computer-based model has been developed to simulate the main components of such a filling plant. Using the C programming language a system has been developed in a fashion which allows continuous improvements from experimentation to be incorporated in a simple approach.

Measurement noise is simulated using Gaussian white noise, which can be set to represent a variety of conditions; from very poor measurements to almost perfect. The Kalman Filter is implemented in its extended form to provide the widest coverage of filling characteristics. Valves are given opening and closing characteristics, typically with faster closing than opening to represent the mechanical performance of actual valves. A standardised bottle is used in the model: 250mm in height and approximately 0.5 litres in volume. The shape is the commonly found wine/beer bottle style. The performance of the Fuzzy Logic will not be discussed here.

As previously mentioned, the iteration time of the Kalman Filter has a significant impact of the success of the parameter estimation. To have a very fast iteration time is computationally expensive and also the smallest iteration time is limited to the transit time of the ultrasound (~1.5ms @ 0.5m). To better understand the Filter performance, a series of tests were carried out to show the improvements in noise reduction at number of different iteration times and three different initial measurement noise levels. The results, as shown in Figure 4., show that the greatest improvements occur initially around the 1 to 0.1 second region, with flattening of the curve after this, therefore reducing iteration below 0.01 seconds would not provide sufficiently better performance to warrant faster iteration time. Hence, the ultrasound transit time cannot be considered a limiting factor on system performance.

Another important factor on the performance of the Kalman Filter is the initialising of internal parameters, such as, the gain matrix or the covariance errors. This is especially important when the filter has to provide estimates in the cyclic environment of multiple bottle fills. Although values can be chosen that will closely match the converged values of the Filter, the effect of valve opening characteristics and the establishing of flow within the system means the internal model is unlikely to be able to sufficiently model this initial period of filling. An error at this stage can cause a permanent bias to estimates from the initial overestimate of flow rate (Figure 5). However, by purposely destabilising internal values by including an overly large covariance error, the filter can be forced to ignore the internal model until a degree of stabilisation has occurred. Although this means that parameter estimates from the filter are unreliable at the beginning, this will be addressed by the overseeing Fuzzy Logic controller, and by careful balancing with the iteration time, the filter will track the actual fill height further up the bottle, despite un-modelled valve characteristics. The effect of better initialisation can be seen in Figure 6.

6. Conclusions

The bottling industry needs a control system to improve the efficiency of the filling process. However, with no option of redesigning plant at this stage, a retrofitted system will have to cope with the problems associated with the lack of complete control over plant parameters.

Using a Kalman Filter to reduce limitations in the sensor technology (due to less-than-ideal placement) will allow measurement of the liquid level during filling of individual bottles. This is in contrast to the present approach, which places sensors at some point after filling has complete. Hence, the time to plant steady state is significantly reduced. It has been demonstrated that the Filter is a realistic approach in a real-time environment with a suitable iteration time. It has also been shown that features such as valve characteristics can be accommodated without necessarily using explicit models.

An overall controller for a multi-headed filling plant using fuzzy logic will allow for a robust approach, which will build on the advantages gained from using ultrasound technology to obtain two measurements for the effort of one and the performance gains of a Kalman Filter. By using fuzzy sets to describe the stages of filling the confidence of measurements can be ascertained, this will be most useful for estimating flow rates from Doppler shifts, thus improving height estimates. It will also be possible to include condition monitoring into the system without significant further investment.

The approach has great potential to improve the filling process, with implementation on a range of different configurations possible by utilising the rule-based inference engine of the fuzzy logic to provide overall control. Moreover, the application is not limited to beverage bottle filling, but many other fluid level applications such as in the chemical or medical industries. In addition, the control and filtering approach is applicable to other sensor technologies and because of the relative simplicity of Fuzzy Logic, modifications for other applications is made reasonably straightforward. Nevertheless, the greatest short to medium term prospect will be improvements to bottle filling and the reduction of waste necessary to support the industry's continuing push for greater efficiency.

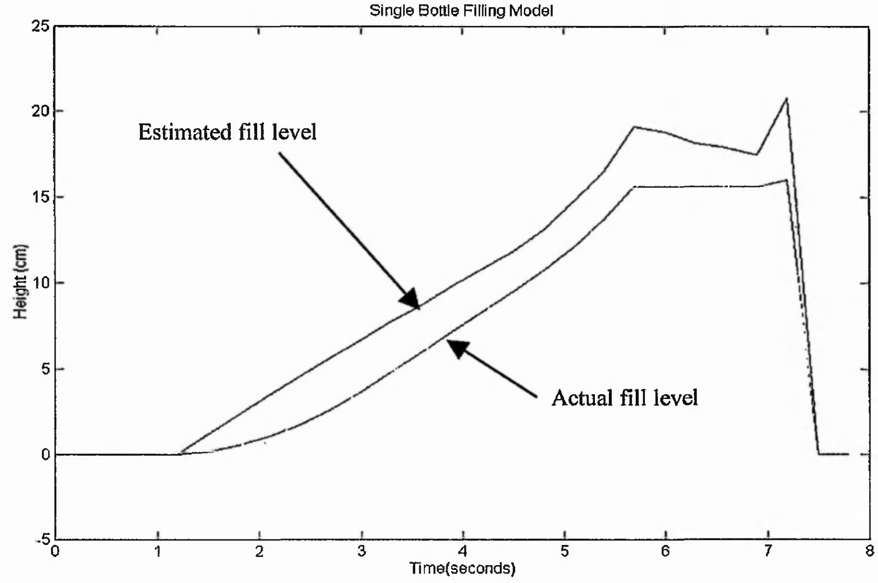


Figure 5. The creation of offset due to sub-optimal initialisation of filter parameters

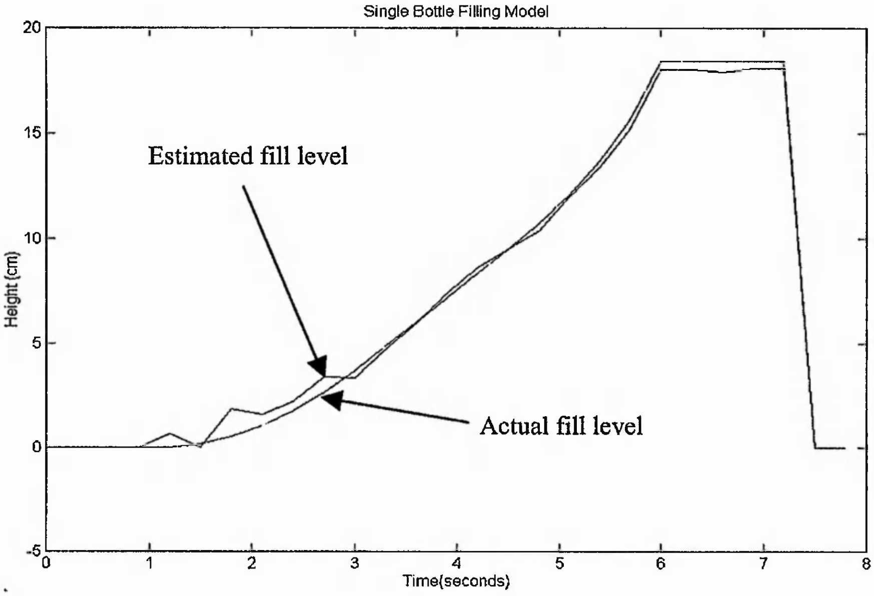


Figure 6. Improved performance from forced destabilisation of filter parameters at initialisation

6. References

[1] Hull, J.B., Henthorn, K.S., Muumbo, A.M., 1995, "Controlling Waste in the Food Processing Industry", *Advances in Materials and Processing Technologies, AMPT95*, Dublin, pp31-39.

[2] Ridgway, J.S., Henthorn, K.S., and Hull, J.B. 1997, "Controlling of Overfilling on Food Processing" *Ultrasound in Food Systems*, M.J. Povey, ed., Chapman & Hall.

[3] Cho, C.H., 1982, *Measurement and Control of Liquid Level*, Instrument Society of America – Independent Learning Module.

[4] Asher, R.C., 1997, *Ultrasonic Sensors for Chemical and Process Plant*, Institute of Physics Publishing, Bristol and Philadelphia.

[5] Kalman, R.E., 1960. "A New Approach to Linear Filtering and Prediction Problems", *Journal of Basic Engineering*, **82**, pp. 35-45.

[6] Kalman, R.E., and Bucy, R.C., 1961 "New Results in Linear Filtering and Prediction Theory", *Journal of Basic Engineering*, **83**, pp. 95-108.

[7] Grimble, M.J. and Johnson, M.A., 1988, *Optimal Control and Stochastic Estimation: Theory and Applications Volume 2*, A Wiley-Interscience Publication, John Wiley & sons Ltd, Chichester.

[8] Jeffries M, E Lai, Plantenberg D.H. and Hull J.B. 1997. "A Fuzzy Approach to the Condition Monitoring of a Packaging Plant". *Proc. 6th International Scientific Conference on Achievements in the Mechanical & Materials Engineering AMME '97.* The Silesian Technical University, Gliwice, Poland, 28 November - 3 December 1997, 95-98