A CONTROL LOOP PERFORMANCE MONITOR

By

SAMUEL ODEI OWUSU

Bachelor of Science in Chemical Engineering
University of Science and technology
Kumasi, Ghana
1994

Master of Science in Chemical Engineering
North Carolina Agriculture and Technical State University
Greensboro, North Carolina
2001

Submitted to the Faculty of the
Graduate College of the
Oklahoma State University
in partial fulfillment of
the requirements for
the Degree of
DOCTOR OF PHILOSOPHY
May, 2006

A CONTROL LOOP PERFORMANCE MONITOR

Thesis Approved:

Dr. R. R. Rhinehart

Thesis Advisor

Dr. J. R. Whiteley

Dr. R. G. Ingalls

Dr. E. Misawa

Dr. G. X. Chen

Dr. A. Gordon Emslie

Dean of Graduate College

ACKNOWLEDGMENTS

It is with great pleasure that I write to acknowledge my gratitude to those that have made an impact on my educational life. First, my unqualified gratitude goes to Dr. R. R. Rhinehart, my research advisor whose in-depth knowledge, ideas, guidance and support has helped steered this project to this conclusion. I am also grateful for his continuous financial assistance. I found in him an advisor for his constant guidance and support, a father for always standing towards me in loco parentis, and a true friend for always being available to meet with me for discussions. I would like to extend a special appreciation and gratitude to my advisory committee members; Dr. J. R. Whiteley of the School of Chemical Engineering, Dr. E. Misawa of the Department of Mechanical and Aerospace Engineering, Dr. R. G. Ingalls of the School of Industrial Engineering and Management and Dr. G. X. Chen of the Fractionation Research Incorporated (FRI). I am truly grateful to you all for taking time of your busy schedules to review my work and to guide me in completing this research.

I would also like to gratefully acknowledge the Measurement Control and Engineering Center (MCEC) for providing financial support and industrial guidance for this work. Thanks also to the Oklahoma State University (OSU) and also to the College of Engineering and Architecture (CEAT) for providing me with a conducive academic environment for my research work. To the school of Chemical Engineering, I say thanks for the awesome academic programs and the knowledge you have instilled. I am proud to

have been a student in this august department and for a fruitful academic period here at OSU.

For camaraderie, I shall always be grateful to my colleagues, Ming Su and Preetica Kumar for the useful discussions and humorous times we shared together in the laboratory.

My special gratitude goes my true love and better half Millicent for her love, encouragement, and precious support in all my academic study. Thanks for being there for me always. Indeed, it takes but two to make a pair.

I would also like to acknowledge my parents back home in Ghana, David and Elizabeth Owusu, for their constant prayers and spiritual support and for seeing me through a wonderful education right from my childhood. My pastor Leroy Hawkins and his better half Leta of Stillwater Hosanna Assembly of God for the fellowship we shared together and their prayers. My parents in the United States Don and Alvetta McCabe for making us feel at home and adding us to their family. To all in Hosanna, thank you for your prayers.

Lastly, I would also like to extend my love to my daughter Kimberley whose presence around me always glows like a candle and brightens my life. Thanks for being in our life Kimberley.

TABLE OF CONTENTS

Chapter Page

Chapter Page

LIST OF TABLES

Table Page

LIST OF FIGURES

Figure Page

Figure Page

Figure Page

Figure Page

Figure Page

Figure Page

Figure Page

Figure Page

CHAPTER 1

INTRODUCTION

Control loop performance monitoring and assessment is becoming the basis of operational excellence in the chemical and allied industry and a typically large process operation in most of these industries consists of hundreds of control loops, often operating under varying conditions. Maintenance of these loops is generally the responsibility of either a lead operator, control engineer, or an instrument technician; but other responsibilities, coupled with the tediousness of consistently monitoring a large number of loops, often result in control problems being overlooked for long periods of time (Hugo, 2000). Moreover, recent corporate acquisitions coupled with downsizing in human resource needs have led to a situation where very few technical personal are left to operate or monitor too many corporate assets.

For process safety, product quality, and profitable manufacturing practice, good control performance is a necessary requirement. In most chemical and allied industries today, real time detection and diagnosis of faults have become an integral part of process design (Ralston, *et al.*, 2001). Tatara, *et al.,* 2002 have indicated that among the various methods for detecting changes in an industrial process, statistical methods generally predominate due to the accuracy involved in using sampled data for decision analysis. A primary difficulty of controller performance analysis is the number of loops in a process.

Hugo, (2000) also revealed that anywhere between 66-88% of industrial process controllers do not perform as well as they should. Also, according to Desborough, *et al.*, 2001 only about a third of industrial controllers provide an acceptable level of performance, in spite of the performance measures developed in the past 10 years. In fact, surveys available suggest that a vast majority of industrial control loops perform far less than optimal.

In a recent article, Merritt, 2003 discussed a study by Honeywell Process Solutions in which the company, using its controller performance software, Loop Scout, reported performance results of over 100,000 process control loops at 350 manufacturing facilities. Their results indicated that of all the process control loops that were analyzed, nearly 49% were found to be performing poorly, about 32% were rated as having common oscillation problems, 16% had valve stiction problems and only about 4.4% of the loops had been retuned within the last two years. Using statistical analysis, Honeywell further determined that almost 63% of all the control loops have poor performance ratings. This further buttresses the claim by previous researchers that only about a third if not fewer of all control loops have good performance ratings.

Among the reasons for the dismal performance of process controllers are poor controller tuning, deficiencies in the control structure, valve malfunctions, nonlinearities and poor process design. Even when a loop performs well at the time of commissioning, its performance deteriorates over time due to changing operating conditions - a good controller becomes a bad one.

In any control scheme, a deviation from setpoint is a function of both controller performance and the plant disturbance spectrum, and it is a general requirement that any

controller performance assessment technique should have at least the following basic attributes (Hugo, 2000):

1. Be independent of disturbance or setpoint spectrums.
2. Able to be automated.
3. Require minimum specification of process dynamics.
4. Be sensitive to detuning or process model mismatch.

It is well established that it is not just adequate to describe the performance of a control system with simple statistics like the mean and variance of manipulated and controlled variables. While these are important performance measures, a comprehensive approach (Harris, *et al*., 2001) for controller performance monitoring usually includes the following:

1. Determination of the best performance capability of the control system
2. Development of suitable statistics for monitoring the performance of the existing system, and
3. Development of methods for diagnosing the underlying causes for changes in performance of the control system.

With the easy availability of plant data today, it has become more appropriate and useful to develop tools and procedures for assessing the performance of control loops and possibly to determine the causes of poor performance.

The past 20-30 years has seen very dramatic changes in the process industries due to the significant improvement in process control techniques and strategies. That industry needs an efficient means to monitor the goodness of performance of process controllers is evident in the number independent approaches being pursued. According

to Merritt, (2003) it is generally not easy to locate poor performing control loops; and while control loops do not fail like pumps, their continuous deterioration toward poor performance erodes operating profit. Though some plants live with situations like this, unfortunately it is not a good manufacturing practice.

Numerous control techniques have been developed to improve and or enhance controller performance and increase economic output. Nevertheless, as indicated earlier, estimates of the percentage of industrial process controllers with performance problems are surprisingly high and efficient techniques to detect and arrest poor controller performances have not been completely explored to make the situation any better. Moreover, current applications available for detecting poor controller performance are either too expensive, cumbersome to use or are themselves fraught with inherent difficulties.

Currently, millions of dollars are lost in industry because faults are not detected and identified on time. In the United States of America alone, it is estimated that petrochemical industries lose almost $20 billion annually due to poor monitoring and control of abnormal situations (Venkatasubramanian, 2003). It is further estimated that this cost is much more when similar situations in other industries such as pharmaceutical, specialty chemicals, power, etc. are included.

Present commercial mechanisms available for monitoring controller performance are done using data from the plant “historian”. This implies that, in real-time, there is no simple approach, by which the goodness of a process controller can be assessed. The outcome of this work seeks to contribute in addressing this problem.

This work is an extension of the work by Li *et al*., 2004 who used the run length of the actuating errors between consecutive zero crossings coupled with the chi-squared goodness-of-fit statistics to develop a performance monitoring technique. In a private communication, Dr. R. G. Ingalls reviewing Li's work suggested the use of Markov chains. Dr. Ingalls and Dr. Avery also submitted an MCEC proposal in which they demonstrated the Markov chain approach. This work was born out of those ideas and seeks to improve on Li's work by coupling the Markov chain technique with binomial statistics. The work was sponsored by the measurement control and engineering center (MCEC), a consortium of industrial companies and academic experts whose primary objective, among others is to carry out research in technological areas related to process control in response to and in support of the needs of its industrial sponsors.

This work hypothesizes the development of a novel method that can inform an operator to a high level of certainty and in real time if a controller is healthy (i.e., performing optimal) or not.

1. The use of Markov chain analysis in monitoring controller performance: a major embodiment in this work is expected to introduce efficiency into controller monitoring techniques. In general, the Markov chains provide a solution for determining the dynamic behavior of a system in occupying states that it can occupy. As a result, the use of Markov chains is expected to provide an easy and simple benchmark for checking the health of process controllers, inferential sensors, and in-fact any system with process output that can be compared to a setpoint or a model output to a process output. It is expected that it will utilize

minimal computer storage space and time and, yet, will be efficient in monitoring the goodness of performance of process controllers.

2. The binomial distribution is probably the best known and most commonly used of the discrete probability distributions (NIST). It arises any time an event with exactly two outcomes is repeated over and over again. It gives the likelihood of finding the number of successes or failures in a given number of observations.

CHAPTER 2

2.0 CONTROL LOOP PERFORMANCE MONITORING

2.1 Literature Review

A common benchmark for controller performance assessment is the minimum variance control (MVC). The MVC and similar techniques have received much attention as a popular standard that greatly reduces the amount of process knowledge required for control performance evaluation. Based on the work of Harris, *et al*., (1989) the MVC, also referred to as the Harris index (HI), compares the ratio of the variance of the actuating error signal to the minimum variance ideally achievable by a perfect controller. It is denoted as:

$$HI = \frac{(Current\ Error\ Variance)}{(Minimum\ \ Acheivable\ Variance)} \tag{2.1}$$

Where Error = Setpoint – Control Variable;

In this format the HI indicates perfect control when HI =1 and bad or degrading control when HI is large. Another form of the minimum variance index also introduced by Harris is the closed loop performance assessment (CLPA). The CLPA is normalized between 0 and 1 as:

$$\text{CPLA} = 1 - \frac{1}{\text{HI}} \tag{2.2}$$

In this format, 0 denotes perfect control and 1 poorest control. The advantage of the Harris index is that it does well in indicating loops that have oscillation problems. In general, when loops have oscillations, the error variance about the mean is large and therefore the HI will be large and CPLA will approach 1.

Unfortunately, the HI may consider loops that are sluggish to be just fine because when loops are sluggish their error variance is small and the HI will approach 1. Hence, the HI may not detect loops that have been detuned to the point of sluggishness. Moreover, a controller not even running in automatic may give a small value of the HI since without the valve moving much, the variance is low. Because small values are generally good, the HI index may not flag such sluggish loops. Thus, in a typical plant, the Harris Index will easily identify loops that are oscillating because of valve hardware problems or over aggressive tuning. On the other hand, a typical operator response to oscillating loops is to detune it. If it is detuned to the point of sluggishness, the HI may not catch this. This is a limitation in the minimum variance control.

In addition, the delay associated with the process must be known in order to describe the process model from which the variance and perfect control can be evaluated. However, processes change during routine operation and so do the delay associated with them and online estimation of the process delay is cumbersome to estimate.

Further, "perfect" control from a controlled variable (CV) measure means excessive manipulated variable (MV) action. Such perfect control is not desired. Therefore a CPLA value of zero is never attained. Moreover, neither the HI nor CPLA value has absolute meaning. A 0.3 may be good for one loop while a 0.5 may be the perfect balance for another.

Hägglund, (1995) has proposed a method in the time domain, which considers the integrated error (IE) between all zero-crossings of the signal. If the IE is large enough, a counter is increased. If this counter exceeds a certain number then an oscillation is indicated. Nevertheless, the question here is the quantification of what "large enough" is. In other to quantify what large enough means, the author used the ultimate frequency of the loop in question. Alternatively, one may use the integral time of the controller assuming the controller is optimally tuned. This method is very appealing in that it is able to quantify the size of the oscillation. However, it assumes that the loop oscillates at its ultimate frequency which may not be entirely true, for instance in the case of stiction. Moreover, the ultimate frequency is not always available and the integral time may be a bad indicator of the ultimate period. These constitute a major limitation in this technique.

Ko and Edgar, (1998) suggested an index that computes the ratio of the actual variance and the minimum achievable variance using a PI-controller. However, this approach assumes that a process model is available. Since processes continuously change response dynamics, any model used would need to be continuously updated.

Other mathematically rigorous controller performance assessment criteria such as the integral absolute error (IAE), integral squared error (ISE), integral time-weighted absolute error (ITAE), and integral time-weighted squared error, (ITSE), have all received much attention. However, the problem with these assessment methods is that, they are very tedious to implement and rely on models for processes, sensors, and final control elements, which may not be exactly known (Stephanopoulos, 1984). In addition, they are scale dependent with process specific values, which must be determined

Rhinehart, (1995) developed an automated, goodness of control performance monitor using the r-statistic, which could indicate when a constraint or a performance limit was violated. The r-statistic was defined as the ratio of the expected variance of the deviation of the controlled variable from the setpoint to one half of the expected variance of the deviation of two consecutive process measurements. It then compares the current r-statistic value with some critical values to indicate performance changes. This technique however, just like other performance monitoring techniques compared a single index value to a trigger value to judge the performance.

Mathematically, the r-statistic was defined as

$$r_w = \frac{\sigma_{y1}^2}{\sigma_{y2}^2} \qquad 2.3$$

Where the variance σ_{y1}^2 is obtained from $\sigma_{y1}^2 = \frac{1}{N-1}\sum_{i=1}^{N} e_1^2(i)$, and $e_1^2(i)$ is the deviation of controlled variable sample *i* from setpoint value. The variance σ_{y2}^2 is obtained from $\sigma_{y2}^2 = \frac{1}{2(N-1)}\sum_{i=1}^{N} e_2^2(i)$, where $e_2(i)$ is the distance between two consecutive samples. A performance problem is indicated if both variance estimates differ such that the ratio is larger than $r_w = 3$. Rhinehart (private communication) realized that it did not consider the distribution of index values and, therefore, the r-statistic was not accepted as a good statistical approach for performance assessment.

A relative variance index (RVI) was proposed by Bezergianni *et al*., (2000). The RVI technique compares the closed-loop performance with the minimum-variance control and open-loop control. They defined the RVI as:

$$RVI = \frac{\sigma_{OL}^2 - \sigma_y^2}{\sigma_{OL}^2 - \sigma_{MV}^2} \tag{2.4}$$

Where, σ_{OL}^2 is the output variance if the controller is removed (i.e. placed in open loop). The RVI is equal to zero if the current performance σ_y^2 is equal to the open-loop performance σ_{OL}^2. It is equal to one if the current performance σ_y^2 is equal to the minimum-variance control σ_{MV}^2. The limitation in this technique is that the process, controller and noise models must all be known.

Kadali, *et al*., (2002) proposed the use of Linear Quadratic Gaussian (LQG) benchmark as a more appropriate tool for assessing the performance of controllers. However, calculation of the LQG benchmark requires a complete knowledge of the process model, which is often a demanding requirement or simple not possible for on-line assessment.

A problem with all the above methods is that they consider integrated metric over a fixed window or time scale. In any, one can easily create higher or lower frequency upsets which would not be discovered. While they can detect some upsets, they will miss others. The progression of Rinehart's studies reveals that a fully functional monitor must observe the process at multiple time scales.

Li, *et al*., (2003) recently proposed the use of the chi-squared goodness-of-fit statistic to compare the distribution of a performance index (run length) within a window of data to a reference run length distribution in order to determine the performance of a controller. A statistically significant change in any section of the distribution, not just an average value is indicative of a significant change in controller performance. The

technique uses only routine plant data and is suited for online application. Although it uses the generally robust chi-squared test, the theoretical foundation was not exact.

The technique proposed and developed in this work also uses run length distribution of data. However, the metric used, state transition probability is based on binomially distributed variables, and the analysis using Markov Chains coupled with Binomial Statistics is not only ideal but also more satisfying. The original idea was proposed by Ms. Sherri S. Avery now Dr. Avery, and Dr. R G. Ingalls (2001) in their analysis of Li's work. The objective of this work is set out below.

2.2 Objective

The main objective of this project is to develop a new and novel method, with a high degree of accuracy based on Markov analysis and binomial statistics to automatically identify poor performance of process controllers, inferential sensors and any model of process output.

The procedures based on this objective are:

1. Measure deviations from setpoint ($y_{sp} - y$)
2. Model a characteristic of the errors (runs) as states in a Markov chain
3. Determine transition probabilities between states
4. Use binomial statistics to detect significant changes in performance from an operator chosen good period

2.3 The Concept of Markov Chains

Markov chains is named after the Russian mathematician Andrey Andreyevich Markov or “Markoff” (June 14 1856 - July 20 1922). Markoff's name “isn't spelled consistently” in English language mathematical literature. Some authors prefer "Markov", "Markov number", "Markov numbers", and "Markov equation". The question that has often arisen is should “Markoff” be spelled -off or -ov? In general, "Markoff" is more often used when discussing his, work in number theory (e.g., Markoff numbers); while "Markov" tends to be used when discussing his work in probability (e.g. Markov chains). When in doubt though experts recommend just writing Андрей А. Марков .

Markov's early work was mainly in number theory and analysis, continued fractions, limits of integrals, approximation theory and the convergence of series. For instance, a “Markoff” number is number that appears in a positive integer solution to the equation $x^2 + y^2 + z^2 = 3xyz$ (known as the “Markoff” equation). For example (1,1,1), (1,1,2) (1,2,5), and (2,169,985), etc, are Markoff numbers and these numbers form part of the vertices of a tree like structure (http://www.minortriad.com/mref.html#spelling).

Nevertheless, Markov is particularly remembered for his study of Markov chains, sequences of random variables in which the future state of a variable is determined by the present state of the variable but is independent of the way in which the present state arose from its predecessors. This work launched the theory of stochastic processes.

Markov chains find widespread use in many areas of science and technology such as Polymers, Biology, Physics, Astronomy, Astrophysics, Chemistry, Operations Research, Economics, Communications, Computer Networks, and others. Markov chains have many advantages. Among these are their ability to be used in the representation of

physical systems in a unified description via state vector and one-step transition probability matrix. They also provide simple solutions to complicated problems by way of Markov chain finite difference equations where exact solutions to models are not available. However, despite their many advantages, the application of Markov chains in modeling chemical and control engineering systems has been relatively meager to non-existent.

Simply put, Markov chains enable one to predict the future state of a system given knowledge of its present state, ignoring its history. It provides a solution for determining the dynamic behavior of a system in occupying various locations (states) that it can occupy (Tamir, 1998). In a stochastic sense, a Markov chain is a probabilistic model describing the state transition of a system where the immediate future state depends only on the present state and not on the manner the system arrived at this particular present state. This is called the "memoryless" property and it can be stated mathematically as

$$P\left(X_{n+1}=j\,\middle|\,X_n=i_n;X_{n-1}=i_{n-1};\ldots X_0=i_0\right)=P\left(X_{n+1}=j\,\middle|\,X_n=i_n\right)\quad\forall j,n,i_n \tag{2.5}$$

Where X denotes a token whose location in time is changing from one place to another

P is the probability of the token X moving from one state to another

j is the next state to be visited at the next transition (n + 1)

i denotes a state that has been visited already

n is an index indicating the location of a token as it moves from one state to another

Thus, Equation (2.5) can be explained as the probability of the token X, occupying the state j at the $(n+1)^{th}$ transition given that it is presently in the state $i_{(n)}$ at the n^{th} transition and just prior to that was in the state $i_{(n-1)}$ at the $(n-1)^{th}$ transition and

before that also, it was in the state $i_{(n-2)}$ at the $(n-2)^{th}$ transition and so on and so forth is the same as the probability of the token occupying the state j at the $(n+1)^{th}$ transition given that presently it is in the state i_n at the n^{th} transition.

The basic concepts of Markov chains are; "system" and "state transition". A system is the set of all possible states (i.e. positions or locations) that a token can occupy. The system (also referred to as the state space (SS)), (Tamir, 1998), is designated by SS = $\{S_{\pm1}, S_{\pm2}, S_{\pm3}, S_{\pm4}, ..., S_{\pm k}\}$ where S_k, denotes the state. A state is (or must be) real. It is a location that can be occupied by the token. States are exclusive of one another, that is, no two states can occur or be occupied simultaneously.

The movement of the token from one state to another in the chain is referred to as transition. Transitions occur within a system from state to state and are referred to as state transitions. Thus, a state transition is the transfer of an event or observation of an event from one state to another.

If the state space is finite (i.e. SS = $\{\pm1, \pm2, \ldots, \pm N\}$, where N is a fixed number), then the chain is said to be a finite Markov chain. On the hand, if the chain has no defined bounds, as in birth-death processes (where states take on all non-negative integer values), then the chain is referred to as an infinite Markov chain. Graphically, the allowable transitions link states, often resulting in a chain like appearance. Transitions are governed by the probability of the system to occupy or visit a state or not to occupy it. In his writings, William Shakespeare captured this literally in his poetic piece "The Hamlet" as: "To be (in a state) or not to be (in a state), that is the question".

Often, the basic aim of Markov chain analysis is to answer questions such as:

1. What is the unconditional probability that at some step n, the system is occupying

or will occupy some state given that the first occupation of this state has occurred already?

2. What is the probability of going from state j to state k in n steps
3. Is there a steady state behavior?
4. If the Markov chain terminates, when it reaches a state k, defined as absorbing or dead state, then what is the expected mean time to reach k (hence terminate the movement of the chain) given that the chain started in some particular state j.

However, this work does not intend to provide answers to all these questions but rather to explore Step 2 for the one-step transition probability of a system and utilize it for further analysis.

A Markov chain may be time homogenous or non-homogenous. It is time homogenous or simply homogenous if its dynamics depend on the time interval but not the time itself.

For instance, the one step transition probability function for a time dependent process can be written as

$$P_{i,j}(t,t+1) = \mathrm{P}\left\{X_{t+1} = j \middle| X_t = i\right\} \quad \forall j,i,t \tag{2.6}$$

Equation (2.6) describes the phenomena that, given that a token *X,* is in a state i at time *t*, what is the probability that it will visit the state *j* at the next sampling instant (*t +1).* Such a process is non-homogenous or time-dependent.

On the other hand, given that:

$$P_{i,j} = \mathrm{P}\left\{X_{n+1} = j \middle| X_n = i\right\} \qquad \forall j,i,n, \tag{2.7}$$

Then it can be noticed that the Equation (2.7) now describes a situation where given that the token *X,* visited or occupied the state i at the nth transition, then what is the probability that it will visit the state *j* at the next transition (n+1) irrespective of time.

Thus, for a time homogenous Markov chain, the probability of transition in a single step from one given state to another depends on the two states and not on time itself. Put another way, $P_{i,j}$ is time homogenous or stationary if it satisfies:

$P_{i,j} =$ Function (time interval between state i and state j), and non-homogenous if it satisfies:

$P_{i,j}(t,t+1) =$ Function (time to start from state i and time when will visit state j)

Let the n-step transition probability function for a stationary Markov chain be denoted by $P_{i,j}^{n}$, where $P_{i,j}^{n}$ is a function that gives the probability of a token going to a state *j* in exactly n-steps given that it occupied or is currently occupying the state *i* on the on the k^{th} transition. Thus:

$$P_{i,j}^{n} = P\left\{X_{k+n} = j \middle| X_k = i\right\} \qquad \forall_{i,j,k,n} \qquad (2.8)$$

If n = 1, then the one-step stationary transition probability $P_{i,j}^{1}$, generally represented as $P_{i,j}$ is given by: $P_{i,j} = P\left\{X_{k+1} = j \middle| X_k = i\right\}$ $\forall_{i,j,k}$

The discrete Chapman-Kolmogorov (C-K) equation provides a method for calculating the transition probability of a token in moving from state *i* to *j* in exactly n steps. The C-K equation is given as:

$$P_{i,j}^{n+m} = \sum_{k=1}^{z} P_{i,k}{}^{n} * P_{k,j}^{m} \qquad \forall_{i,j,k,m,n} \qquad (2.9)$$

Where *z* is the total number of states and n, m denote the number of transitions. Equation (2.9) represents the probability that, starting in *i*, a token will go to state *j* in (m + n) transitions through a path that takes it to state *k*, at the n^{th} transition before arriving at the state *j* after m additional transitions. If **P** denotes the matrix of the one-step transition probabilities $P_{i,j}$, and $\mathbf{P^{(n)}}$ denotes the matrix of the n-step transition probabilities $P_{i,j}^{n}$, then Equation (2.9) can be expressed in matrix form as:

$$\mathbf{P^{(n+m)} = P^{(n)} * P^{(m)}} \tag{2.10}$$

Thus, for instance $\mathbf{P^{(5)} = P^{(3+2)} = P^{(3)} * P^{(2)}}$ and by induction, $\mathbf{P^{(n)} = P^{(n-1+1)} = P^{(n-1)} * P}$

In other words, the n-step transition probabilities of a process may be obtained by multiplying the one-step transition probability matrix **P** by itself n times.

2.3.1 Classification of states and their behavior

All states of a Markov chain fall into distinct types according to their limiting behavior. Suppose that a system is initially in a given state. If the ultimate occupation of the state is certain again at some time later, then the state is said to be recurrent with a probability of unity. In other words, let, $f_{jj} = \text{Prob}\left(\text{ever visit state j} \middle| \text{start from j}\right)$, then state *j* is recurrent if $f_{jj} = 1$. If a state is recurrent, then it is said to be positive recurrent if starting in i, the expected number of steps (transitions) until the process returns to state *i* again is finite. Otherwise, it is null-recurrent (Ross, 2003). If the ultimate return to the state has probability less than unity, then the state is called transient with $f_{jj} < 1$. Thus, the current return to the state is uncertain.

A state *k* is accessible from state j if there exists some positive integer n, such that $P_{jk}(n) > 0$. If a state *j* is accessible from another state *k* in a finite number of transitions,

then the two are said to communicate. If state *j* is accessible from state *k* and state *k* is accessible from state *i*, then *j* is accessible from *i*.

One of the most important concepts of Markov chains is irreducibility. A Markov chain is said to be irreducible if all states communicate. That is, $P_{ij} > 0$. In other words, every state can be reached from every other state. An irreducible section of a Markov chain is referred to as a class. Thus, from above, if a set of states communicate, then they form a class. In addition, since a class is irreducible, it implies that communication is a class property.

If the occupation of a state is such that the transition probability where $P_{jj}=\text{Prob}\left(\text{occupying state j after 1 transition} \middle| \text{start from j}\right)=1$, then the state is said to be an absorbing (dead or trapping) state. In other words, once the system occupies this state, it remains there forever. A typical example of this phenomenon may be identified in a control loop when a valve is saturated (i.e. a fully opened or filly closed valve) for an extended period.

Suppose a chain starts in a state S_j. Subsequent occupations of S_j can only occur at steps (times) 1ν, 2ν, 3ν, 4ν,… where ν is an integer. If ν > 1 and the chain is finite, then S_j is periodic. If ν = 1 and the chain is finite, then S_j is aperiodic. A finite irreducible, positive recurrent, aperiodic Markov chain is called ergodic. Ergodicity is an important concept that helps in estimating the limiting or stationary distribution of stochastic processes modeled with a Markov chain.

For an irreducible ergodic Markov chain, the $\lim_{n\to\infty} p_{ij}^n$, exist and is independent of the initial state *i*. Thus, if $\pi_j = \lim_{n\to\infty} p_{ij}^n \quad \forall_{i,j,n} \geq 1$, then π_j is the unique non-negative solution of:

$$\pi_j = \sum_{i=1}^{\infty} \pi_i p_{ij} \tag{2.11}$$

and

$$\sum_{i=1}^{\infty} \pi_j = 1 \tag{2.12}$$

Where, π_j is the limiting probability describing the fraction of instances that a process will be in a state *j*. As an illustration, consider the Markov chain where:

$$P_{i,j} = \begin{Vmatrix} 0.5 & 0.4 & 0.1 \\ 0.3 & 0.4 & 0.3 \\ 0.2 & 0.3 & 0.5 \end{Vmatrix}$$

If the process is ergodic, then the stationary distribution π must exist and must satisfy Equations (2.11) and (2.12):

$$\begin{aligned} For\ j = 1: &\quad \pi_1 = 0.5\pi_1 + 0.3\pi_2 + 0.2\pi_3 \\ j = 2: &\quad \pi_2 = 0.4\pi_1 + 0.4\pi_2 + 0.3\pi_3 \\ j = 3: &\quad \pi_3 = 0.2\pi_1 + 0.3\pi_2 + 0.5\pi_3 \\ &\pi_1 + \pi_2 + \pi_3 = 1 \end{aligned}$$

Solving gives $\pi_1 = 21/62, \quad \pi_2 = 23/62, \quad \pi_3 = 18/62$

Thus, the asymptotic probability of being in the state of 1, is 21/62, in state 2, is 23/62 and in state 3, is 18/62.

2.4 The Binomial Distribution

The binomial probability distribution applies only to sampling that satisfies the conditions of a binomial experiment. A binomial experiment is one that exhibits or possesses certain key properties (shown below). It is defined under the conditions where a random experiment consisting of 'n' repeated trials is performed such that:

1. The number of trials 'n' is fixed.

2. Each experimental unit results in only two possible outcomes. Of the two characteristic events or outcomes, the one of interest is often referred to as success and the other failure.
3. The probability of success on each trial, denoted as p, remains constant.
4. The outcome for any one experimental unit is independent of the outcome for any other experiment unit.
5. The random variable x, counts the number of "successes" in n trials

In the limiting case when n = 1, a binomial random variable is often referred to as a Bernoulli random variable. In such a case the binomial experiment is referred to as a Bernoulli trial. Also, it is worth mentioning that a more general distribution, which includes the binomial as a special case is the multinomial distribution. For a binomial experiment, a random variable x say, is used to denote the number of trials that result in a success or an event of interest. This random variable has a binomial distribution with parameters p ($0 \leq p \leq 1$), and n = 1, 2, 3, 4, …

Consider for instance, the tossing of a fair coin repeatedly for n times. At each toss, there could be a head or a tail. The chance of getting a head or a tail at each toss is the same at any instance. The event that a head or tail was obtained in the previous trial has no influence on the outcome at the next tossing. Such a process that has all the above properties can be described as a binomial experiment.

For any binomial experiment, given that, there are x successful outcomes in n total trials, the probability distribution of x in such an experiment is called the binomial probability distribution. The probability of getting exactly x successes in n trials is given by the binomial relation:

$$P_x(\mathrm{x,p,n}) = \begin{cases} \binom{\mathrm{n}}{\mathrm{x}} p^x (1-p)^{n-x} & x = 1, 2, 3, \ldots, n \\ 0 & \textit{Otherwise} \end{cases} \tag{2.13}$$

where the notation $\binom{\mathrm{n}}{\mathrm{x}} = \frac{n!}{(n-x)!\,x!}$, denotes the total number of different sequences of trials that contain *x* success and (*n*-*x*) failures. Thus, the total number of different sequences that contain *x* successes and (*n*-*x*) failures times the probability of each sequence equals probability P(*x*), of obtaining a desired outcome *x* number of times in *n* total trials.

The probability expression above leads to a very useful relation called the binomial expansion. For constants *a* and *b* the binomial expansion is given by:

$$(a+b)^n = \sum_{k=0}^{n} \binom{n}{k} a^k b^{n-k} \tag{2.14}$$

Let a = *p* and b = 1-p, then the equation becomes

$$\sum_{k=0}^{n} \binom{n}{k} p^k (1-p)^{n-k} = \left(p + (1-p)\right)^n = 1 \tag{2.15}$$

Thus, the sum of the probabilities for a binomial random variable is always equal to one.

2.4.1 Moments of the Binomial Distribution

The moments of the binomial distribution depend only on the parameters *p*, the proportion or probability of success and *n*, the number of trials. The first moment (also referred to as the mean or expectation) of the random variable *x* is given by:

$$E(x) = \sum_{x=0}^{n} xP(x) \tag{2.16}$$

$$E(x) = \sum_{x=0}^{n} x \binom{n}{x} p^x (1-p)^{n-x} \tag{2.17}$$

$$E(x) = \sum_{x=0}^{n} \frac{xn(n-1)!\, p^x (1-p)^{n-x}}{x(n-x)!(x-1)!} \tag{2.18}$$

Simplifying

$$E(x) = np \sum_{x=1}^{n-1} \frac{(n-1)!\, p^{x-1} (1-p)^{((n-1)-(x-1))}}{((n-1)-(x-1))!(x-1)!} \tag{2.19}$$

$$E(x) = np \sum_{x=1}^{n-1} \binom{n-1}{x-1} p^{(x-1)} (1-p)^{((n-1)-(x-1))} \tag{2.20}$$

Let $k = x$-1, and $N = n$-1, then

$$E(x) = np \sum_{k=0}^{N} \binom{N}{k} p^k (1-p)^{(N-k)} \tag{2.21}$$

Recall from Equation (2.14), for two variables a and b,

$(a+b)^N = \sum_{k=0}^{N} \binom{N}{k} a^k b^{N-k}$, Where, $a = p$ and $b = 1$-p

$$E(x) = np(p+1-p)^N \tag{2.22}$$

$$E(x) = \mu_x = np \tag{2.23}$$

Thus the expected value or mean of the binomial random variable x, is given by the product of the number of trials and the probability of the event of success.

The second central moment about the mean namely the variance of x, is given by $\sigma^2 = E(x^2) - (E(x))^2$, but E($x$) is given by Equation (2.23), and E(x^2) is given by

$$E(x^2) = \sum_{x=0}^{n} x^2 p(x) \tag{2.24}$$

$$E\left(x^2\right)=\sum_{x=0}^{n}x^2\binom{n}{x}p^x\left(1-p\right)^{n-x} \tag{2.25}$$

$$E\left(x^2\right)=\sum_{x=0}^{n}\left[x(x-1)+x\right]\binom{n}{x}p^x\left(1-p\right)^{n-x} \tag{2.26}$$

$$E\left(x^2\right)=\sum_{x=0}^{n}\left[x(x-1)\right]\binom{n}{x}p^x\left(1-p\right)^{n-x}+\sum_{x=0}^{n}x\binom{n}{x}p^x\left(1-p\right)^{n-x} \tag{2.27}$$

$$E\left(x^2\right)=n(n-1)p^2\sum_{x=2}^{n-2}\binom{n-2}{x-2}p^{x-2}\left(1-p\right)^{((n-2)-(x-2))}+E\left(x\right) \tag{2.28}$$

Let $k = x\text{-}2$, and $N = n\text{-}2$, then

$$E\left(x^2\right)=n(n-1)p^2\sum_{k=0}^{N}\binom{N}{k}p^k\left(1-p\right)^{N-k}+E\left(x\right) \tag{2.29}$$

$$E\left(x^2\right)=n(n-1)p^2\left(p+1-p\right)^{n-2}+np \tag{2.30}$$

$$E\left(x^2\right)=\left(np\right)^2-np^2+np \tag{2.31}$$

$$E\left(x^2\right)=\left(np\right)^2+np\left(1-p\right) \tag{2.32}$$

Recall that $\sigma^2=E\left(x^2\right)-\left(E\left(x\right)\right)^2$

$$\sigma^2=\left(np\right)^2+np\left(1-p\right)-\left(np\right)^2 \tag{2.33}$$

$$\sigma^2=np\left(1-p\right) \tag{2.34}$$

The third central moment about the mean of the binomial random variable gives the skewness of the distribution. The skewness gives the degree of symmetry of the probability distribution. If a distribution is normal, the value of skewness is zero. If it is skewed to the left, it has a negative value for skewness and if it is skewed to the right, it has a positive value for skewness. The skewness of a binomial distribution can be

derived by going through similar steps as before, but it is given here without derivation as:

$$S = \frac{(1-2p)}{\sqrt{np(1-p)}} = \frac{1-2p}{\sigma} \tag{2.35}$$

Similarly, the fourth central moment about the mean of a binomial random variable gives the Kurtosis of the distribution. Kurtosis is a measure of the peakedness of the probability distribution. It can be shown that Kurtosis of a binomial distribution is given by:

$$K = \frac{1-6p(1-p)}{np(1-p)} = \frac{1-6p(1-p)}{\sigma^2} \tag{2.36}$$

2.5 The Normal Distribution

Perhaps, the most widely used distribution for modeling random experiments is the normal distribution. It is also known as the bell-shaped curve, and Gaussian distribution, and can be developed by considering the basic model for a binomial random variable as the number of trials becomes large (Montgomery, *et al*., 1998; Devore, 1995).

In general a random variable x is said to have a normal distribution with parameters μ and σ^2 where $-\infty < \mu < \infty$ and $\sigma^2 > 0$, if the probability density function (pdf) is given by

$$f(x) = \frac{1}{\sqrt{2\pi\sigma}} e^{-(x-\mu)^2/(2\sigma^2)} \qquad -\infty < x < \infty \tag{2.37}$$

Thus, $P(a \le x \le b) = \int_a^b f(x)dx = \int_a^b \frac{1}{\sqrt{2\pi\sigma}} e^{-(x-\mu)^2/(2\sigma^2)} dx$ (2.38)

The statement that x is normally distributed with parameters μ and σ^2 is often abbreviated as $x \sim N\left(\mu, \sigma^2\right)$. Equation (2.38) cannot be solved with analytical integration techniques. However, when the parameter values $\mu = 0$ and $\sigma^2 = 1$, the normal distribution is referred to as the standard normal distribution with random variable often denoted z. The pdf of z is given by

$$f\left(z\right) = \frac{1}{\sqrt{2\pi}} e^{-(z)^2/(2)} \qquad -\infty < z < \infty \qquad (2.39)$$

Probabilities involving the random variable x are therefore determined by standardizing. The standardizing shifts the mean from μ to zero, and then scales the variable so that the standard deviation is unity rather than σ. In general, $z = \frac{x - \mu}{\sigma}$ and by standardizing, any probability involving the random variable x, can be expressed as a probability involving the standard normal random variable z. Thus, Equation (2.39) is often evaluated numerically after standardizing the random variable x using given values of the mean μ and standard deviation σ.

The cumulative density function (cdf) of Z is $P(Z \leq z) = \int_{-\infty}^{z} f\left(z\right) dz$ and is often denoted by $\Phi(z)$. In general,

$$\phi(z) = \int_{-\infty}^{z} f\left(z\right) dz \qquad (2.40)$$

2.6 Sample Proportions

For a given a binomial random variable x, if n is the number of trials of an experiment and $\hat{p}$ denotes the sample proportion of successes, where success identifies

that the outcome of an event has some specified property of interest. The proportion of successes is given by $\hat{p} = \frac{x}{n}$. Thus,

$$E\left(\hat{P}\right) = E\left(\frac{x}{n}\right) = \frac{1}{n}E\left(x\right) = \frac{np}{n} = p \tag{2.41}$$

The above Equation indicates that provided *x* is a binomial random variable with population parameters *n* and *p*, then value of the parameter *p* that maximizes the chance of success is given by the sample proportion $\hat{p} = \frac{x}{n}$. In other words, $\hat{p} = \frac{x}{n}$ is an unbiased estimator of the population mean, *p*. From Equation (2.34), the variance of the distribution of the number of successes *x*, is given by $\sigma_X^2 = E\left(x^2\right) - \left(E\left(x\right)\right)^2 = n(p(1-p)$. However, for the proportion of successes $\hat{p}$, the variance of the distribution is given by:

$$\sigma_{\hat{p}}^2 = E\left(\hat{p}^2\right) - \left(E\left(\hat{p}\right)\right)^2 = E\left(\left(\frac{x}{n}\right)^2\right) - \left(E\left(\frac{x}{n}\right)\right)^2 \tag{2.42}$$

$$\sigma_{\hat{p}}^2 = \frac{1}{n^2}\left(E\left(x^2\right) - \left(E\left(x\right)\right)^2\right) = \frac{np\left(1-p\right)}{n^2} = \frac{p\left(1-p\right)}{n} \tag{2.43}$$

Often, the population mean *p*, is a parameter that is not known. Hence, from Equation (2.41-), since $\hat{p}$ is an unbiased estimator of *p*, the variance of a sample proportion can be estimated as:

$$\sigma_{\hat{p}}^2 = E\left(\hat{p}^2\right) = \frac{\hat{p}\left(1-\hat{p}\right)}{n} \tag{2.44}$$

2.7 Test of Hypothesis

A statistical hypothesis, or hypothesis, is a claim about the value(s) of one or more population characteristic(s). In any hypothesis testing, two contradictory statements are often proposed. The objective is then to decide which of the two statements about the population is more likely or correct based on sample information. The statements are often formulated so that one of the claims is initially favored. The initially favored claim will not be rejected in favor of the alternative claim unless sample evidence confidently contradicts it and provides strong support for the alternative assertion. The claim initially favored and believed to be more likely is referred to as the null hypothesis. It is often denoted by H_0. The other claim is referred to as the alternative or research hypothesis and is often denoted by H_a (Devore, 1995; Montgomery, *et al*., 1998; Mendenhall, *et al*., 1982)

2.8 Errors in Hypothesis Testing

The decision to accept or reject a claim on the null hypothesis is based on information often contained in a sample drawn from the population of interest. The sample values are used to compute a single number corresponding to a point on a line. This number, referred to as the test statistic, is usually used by experimenters in making statistical decisions. The entire range of values that the test statistic may assume is divided into two sets, or regions. One corresponds to the rejection region and other the acceptance region. If the test statistic computed from a particular sample assumes a value in the rejection region (also called the critical region), then the null hypothesis is rejected in favor of the alternative hypothesis. On the other hand, if the test statistic falls in the

acceptance region, then the null hypothesis is accepted (but not proven truth). The circumstances leading to either of these decisions vary are discussed below. In statistical tests, it is possible to make an error that involves rejecting the null hypothesis when in fact it is true and should not be rejected. The probability of making such an error is referred to as a Type-I error. It is also often referred to as the significance level and is usually denoted by α.

$$\alpha = P\left(Type-I\ error\right) = P\left(Reject\ H_0 \middle| H_0\ is\ true\right) \tag{2.45}$$

Another type of error involves accepting the null hypothesis (i.e. failing to reject the null hypothesis) when it is false and the alternative hypothesis is true. The probability of making such an error is referred to as a Type-II error and is often denoted by β.

$$\beta = P\left(Type-II\ error\right) = P\left(Accept\ H_0 \middle| H_0\ is\ false\right) \tag{2.46}$$

In typical applications, a Type-I error, results in a false alarm while a Type-II error results in a missed alarm. Table 2.1 is a summary of the decisions generally made in Hypothesis testing.

Table 2.1 Decisions in Hypothesis Testing

Decision	**H_0 is true**	**H_0 is False**
Fail to Reject H_0	Correct decision	Type-II error β
Reject H_0	Type-I error α	Correct decision Power = 1-β

Where Power is the probability of rejecting H_0, if H_0 is false.

Figure 2.1 is a graphical illustration of the occurrence of a Type-I error. The shaded region between the lower control limit (i.e. critical value below which distribution

is out of control, (LCL)) and the upper control limit (i.e. the critical value above which the distribution is out of control, (UCL)) gives the range or area within which a test statistic must lie before it is accepted.

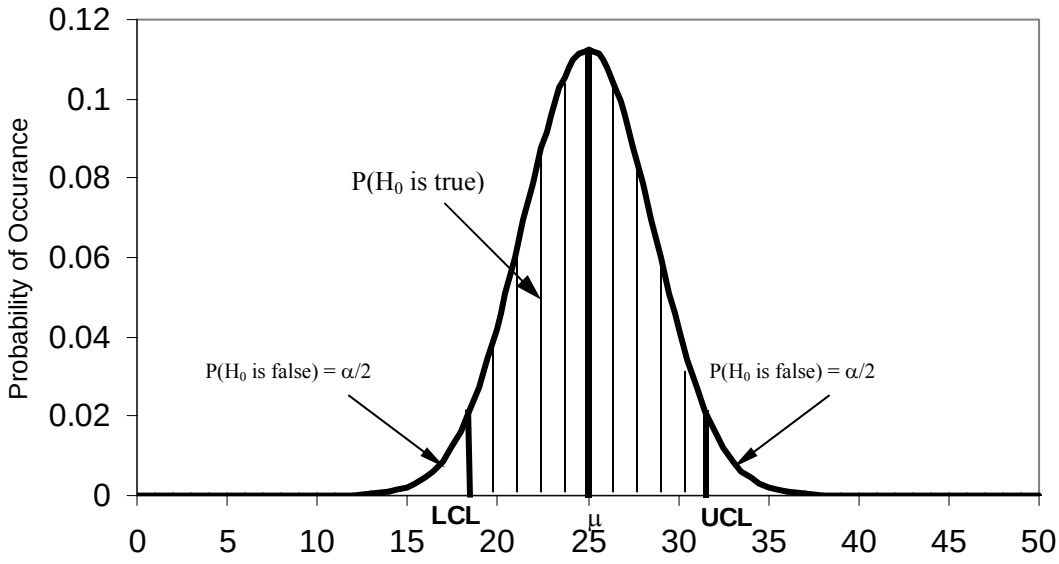


Figure 2.1 Probability Distribution of *x* successes in n trials (n = 50, P_o = 0.5)

If a test statistic falls in this range and it is truly within the range, then a correct decision will be made by accepting the null hypothesis. However, it a possible that a test statistic that lies in the shaded region could be ignored or rejected based on a claim that it lies outside the shaded region. The probability of erroneously rejecting the true outcome of an hypothesis is the value of a Type-I error. From Figure 2.1 the probability of making a Type-I error is given by the sum of the areas in the tails outside the shaded region.

Furthermore, during experimental analysis, the mean value the distribution changes. If the change is significantly different from the hypothesized value μ, it implies

that the new mean value comes from a significantly different distribution. It is essential that a decision be made to reject this new mean value. Using Figure 2.2, let graph A be the hypothesized distribution.

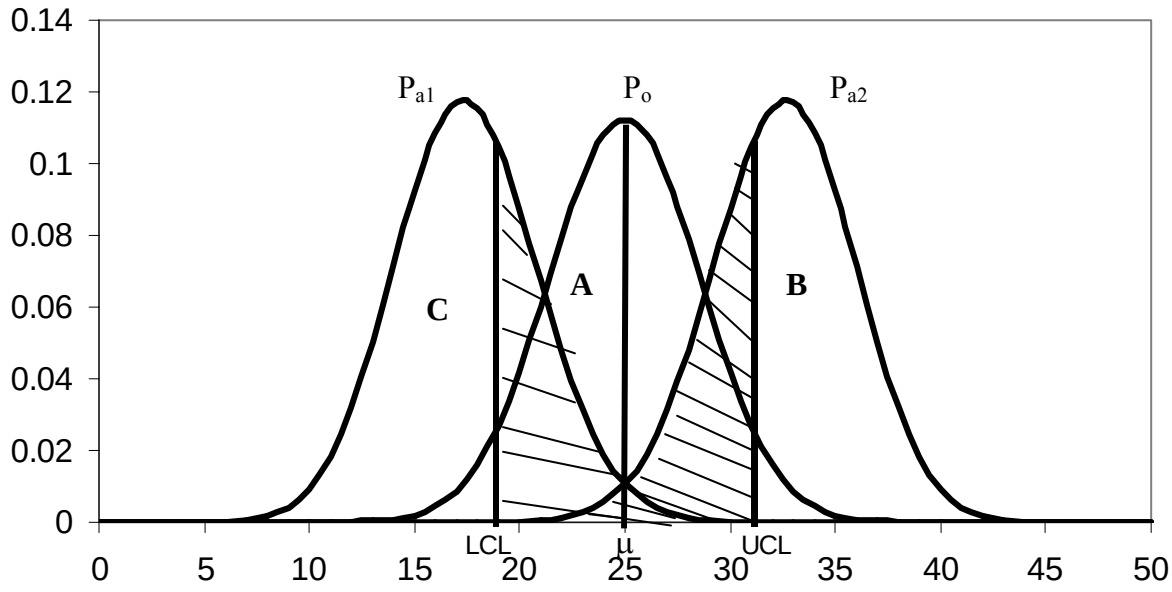


Figure 2.2 Probability Distribution of X successes in n trials (n = 50, P_o = 0.5, P_{a1} = 0.35, P_{a2} = 0.65)

However, if a value belongs to the distribution shown by C it is possible that some of the test statistics that belong to C may also appear in distribution A and what is a false or spurious value may get accepted. The same situation may occur between distribution A and B. The frequency or rate of not detecting false values lead to the occurrence of Type-II errors. The probability of a Type-II error is given by the area of the shaded region in each of the distributions A or B.

The goodness of a statistical test of an hypothesis is measured by the probabilities of making a Type-I or a Type-II error (Mendenhall, 1982).

- The size of the critical regions, and consequently the probability of a Type-I error α, can always be reduced by appropriate selection of the critical values.
- The Type-I and Type-II errors are related. A decrease in the probability of one type of error always results in an increase in the probability of the other, provided the sample size does not change.
- An increase in sample size will generally reduce both α and β provided that, the critical values are held constant.
- When the null hypothesis is false, β increases as the true value of the parameter approaches the hypothesized value in the null hypothesis. The value of β decreases as the difference between the true mean and the hypothesized value increases.

Very often, the probability of a Type-I error is controlled by the researcher or analyst when the critical values are selected. Thus, it is usually easy to set a Type-I error rate or probability α, at or near some desired value. On the other hand, the probability of a Type-II error rate β, is not constant. It depends on the true value of the population parameter P_0, the sample size n, the new population variance, and the extent to which the null hypothesis H_0 is false. The probability of correctly rejecting a false null hypothesis is referred to as Power and is given by

$$Power = P\left(Reject\ H_0 \,|\, H_0\ is\ false\right) = 1 - \beta \tag{2.47}$$

The Power of a statistical test is a very descriptive and concise measure of the sensitivity of the test (i.e. the ability of the test to detect differences). If a test of hypothesis is designed to detect difference on both sides of the true mean μ_0, then it is called a two-tailed test. In such a test, the conventional approach is that the critical

region is split into two parts with equal probability placed in each tail of the distribution of the test statistic [Montgomery, et al., 1998]. Such tests are formulated as:

$$H_0 : \mu = \mu_0$$
$$H_a : \mu \neq \mu_0 \tag{2.48}$$

If the hypothesis-testing is on only one side of the true population mean μ_0, then it is referred to as a one sided alternative hypothesis formulated as:

$$H_0 : \mu = \mu_0$$
$$H_a : \mu \geq \mu_0 \tag{2.49}$$

or

$$H_0 : \mu = \mu_0$$
$$H_a : \mu \leq \mu_0 \tag{2.50}$$

2.9 Analysis of Type-I errors Using Binomial Distribution

As indicated earlier, a Type-I error occurs when a true null hypothesis is rejected. Figure 2.3 illustrates the occurrence of a Type-I error

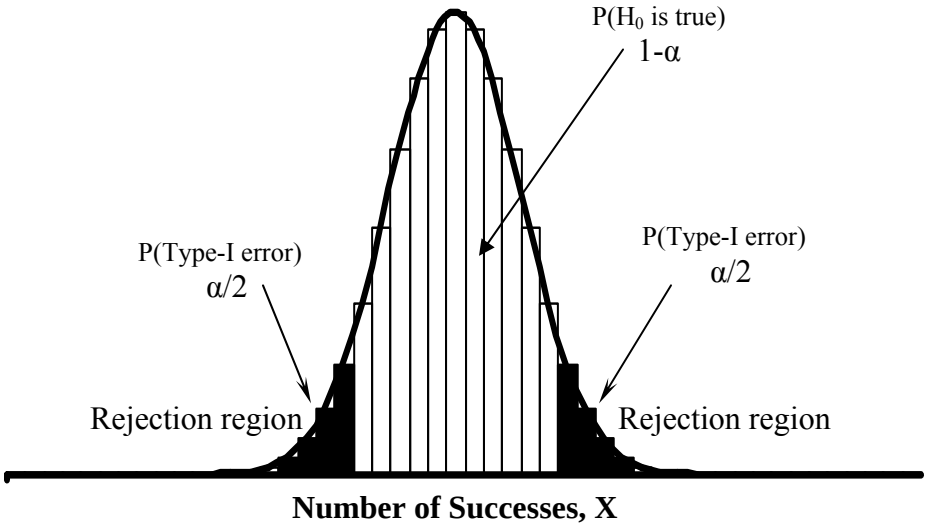


Figure 2.3. Probability Distribution Showing a Type-I Error Occurrence

In general, for the binomial distribution, Equation (2.13), where

$$P_x\left(\mathrm{x,p,n}\right)=\begin{cases}\binom{\mathrm{n}}{\mathrm{x}}p^x\left(1-p\right)^{n-x} & x=1,2,3,...,n\\ 0 & \textit{Otherwise}\end{cases}$$

can be used to estimate the probability of a Type-I error. The probability of rejecting the null hypothesis when it is true is given by α, and for a two-tailed test, each side has equal probability of $\alpha/2$. The left side rejection region is given by:

$$\frac{\alpha}{2}=P\left(x\le x_L\right)=\sum_{x=0}^{x_L}P\left(x=0,1,2...x_L\right) \tag{2.51}$$

$$\frac{\alpha}{2}=\sum_{x=0}^{x_L}\binom{n}{x}P_0^x\left(1-P_0\right)^{n-x} \tag{2.52}$$

where X_L is the number of discrete samples that define the rejection region to the left of Figure 2.3. The right side rejection is also given by:

$$\frac{\alpha}{2}=P\left(x\ge x_H\right)=\sum_{x=x_H}^{n}P\left(X=x_H,x_H+1,x_H+2...n\right) \tag{2.53}$$

$$\frac{\alpha}{2}=\sum_{x=x_L}^{n}\binom{n}{x}P_0^x\left(1-P_0\right)^{n-x} \tag{2.54}$$

Either of Equation (2.52) and (2.54) can be used to estimate the probability of a Type-I error in either tail of the probability distribution knowing the reference probability P_0 and the number of samples *x*, that indicate the particular region.

2.10 Analysis of Type-II errors Using Binomial Distribution

A Type-II error occurs when a false null hypothesis is not rejected. This is illustrated in Figure 2.4 where the probability of a Type-II error, β, is represented by the

diamond shaded area under the probability distribution curve when $P = P_a \neq P_0$. From Figure 2.4,

$$\beta = P\left(x_L \le x \le x_H \,\middle|\, P = P_a\right) = \sum_{x=0}^{x_H} P\left(x = 0,1,2,x_H\right) - \sum_{x=0}^{x_L} P\left(x = 0,1,2,x_L\right) \tag{2.55}$$

Using Equation 2.52 in Equation 2.55 gives,

$$\beta = \sum_{x=0}^{x_H} \binom{n}{x} P_a^x \left(1 - P_a\right)^{n-x} - \sum_{x=0}^{x_L} \binom{n}{x} P_a^x \left(1 - P_a\right)^{n-x} \tag{2.56}$$

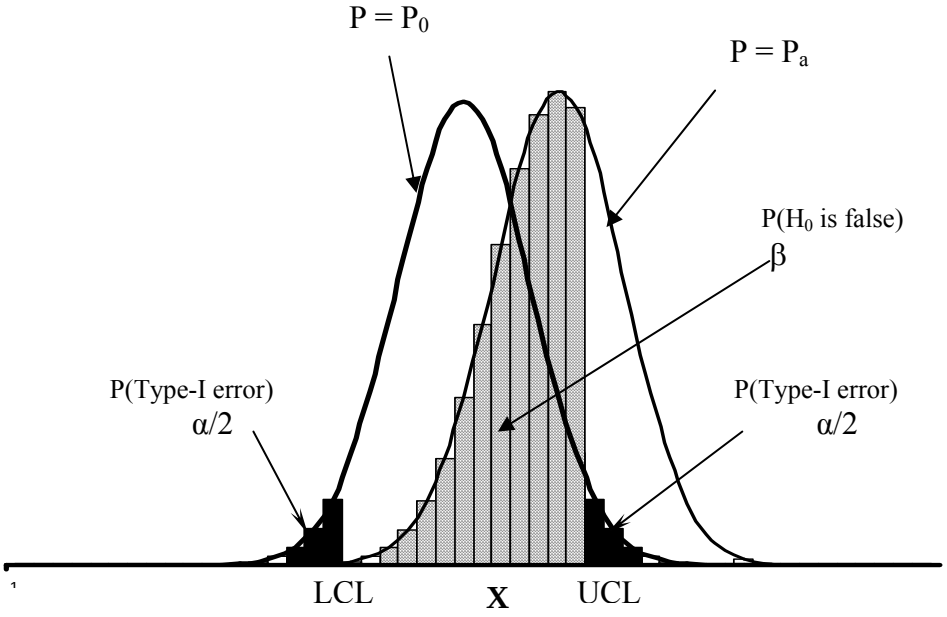


Figure 2.4. Probability Distribution Showing a Type-II Error Occurrence

2.11 Normal Approximation to the Binomial Distribution

Although the binomial distribution is discrete and the normal distribution is continuous, the normal distribution can be used as an approximation to the discrete binomial probability distribution when the product of the number of trials and the proportion of successes (*np*) as well as the product of the number of trials and the

proportion of failures ($n(1-p)$) is greater than or equal to five. If $np < 5$ and $n(1-p) < 5$, then the binomial distribution is too skewed for the normal distribution to give accurate approximations (Devore, 1995). Given the hypothesis,

H_0: $P = P_0$ $\qquad$ H_a: $P \neq P_0$

When the null hypothesis is H_0, is true:

$$CL = P_0 \pm Z_{\alpha/2}\sqrt{\frac{P_0\left(1-P_0\right)}{n}} \tag{2.57}$$

Where CL denotes the confidence limits, $Z_{\alpha/2}$ denotes the boundaries marking the critical region for a two-tailed test and P_0 represents the reference probability. The upper confidence limit (UCL) is given by

$$UCL = P_0 + Z_{\alpha/2}\sqrt{\frac{P_0\left(1-P_0\right)}{n}} \tag{2.58}$$

While the lower confidence limit (LCL) is

$$LCL = P_0 - Z_{\alpha/2}\sqrt{\frac{P_0\left(1-P_0\right)}{n}} \tag{2.59}$$

When the alternative hypothesis H_a is true, the UCL is again given by

$$UCL = P_a + Z_{\beta}\sqrt{\frac{P_a\left(1-P_a\right)}{n}} \tag{2.60}$$

Where Z_{β} denotes the critical boundaries on the left and right of the distribution

$$LCL = P_a - Z_{\beta}\sqrt{\frac{P_a\left(1-P_a\right)}{n}} \tag{2.61}$$

Solving Equations (2.58) and (2.60) and then Equations (2.59) and (2.61) gives

$$Z_{\beta+} = \frac{P_0 - P_a + Z_{\alpha/2}\sqrt{\frac{P_0(1-P_0)}{n}}}{\sqrt{\frac{P_a(1-P_a)}{n}}} \tag{2.62}$$

$$Z_{\beta-} = \frac{P_0 - P_a - Z_{\alpha/2}\sqrt{\frac{P_0(1-P_0)}{n}}}{\sqrt{\frac{P_a(1-P_a)}{n}}} \tag{2.63}$$

$$\beta = \phi\left(Z_{\beta+}\right) - \phi\left(Z_{\beta-}\right) \tag{2.64}$$

$$\beta = \phi\left[\frac{P_0 - P_a + Z_{\alpha/2}\sqrt{\frac{P_0(1-P_0)}{n}}}{\sqrt{\frac{P_a(1-P_a)}{n}}}\right] - \phi\left[\frac{P_0 - P_a - Z_{\alpha/2}\sqrt{\frac{P_0(1-P_0)}{n}}}{\sqrt{\frac{P_a(1-P_a)}{n}}}\right] \tag{2.65}$$

Where, $\phi(z)$ is given by Equation (2.40), after substituting Equation (2.39), to get

$$\phi(z) = \frac{1}{\sqrt{2\pi}}\int_{-\infty}^{z} \exp^{-(U^2/2)}\, dU \tag{2.66}$$

To solve Equation 2.66, let $t = \frac{U}{\sqrt{2}}$, then $\sqrt{2}dt = dU$ and $t^2 = \frac{U^2}{2}$. In addition, the upper limit changes to $\frac{z}{\sqrt{2}}$, while the lower limit becomes $\frac{-\infty}{\sqrt{2}} = -\infty$. Substituting in Equation 2.66),

$$\phi(z) = \frac{1}{\sqrt{2\pi}}\int_{-\infty}^{z/\sqrt{2}} exp^{-(t^2)}\sqrt{2}dt \tag{2.67}$$

$$\phi(z) = \frac{1}{\sqrt{\pi}}\int_{-\infty}^{z/\sqrt{2}} exp^{-(t^2)}dt \tag{2.68}$$

$$\phi(z) = \frac{\sqrt{2}}{\sqrt{2\pi}}\int_{-\infty}^{0} exp^{-(t^2)}dt \quad + \quad \frac{2}{2\sqrt{\pi}}\int_{0}^{z/\sqrt{2}} exp^{-(t^2)}dt \tag{2.69}$$

$$\phi(z)=\frac{1}{\sqrt{2\pi}}\int_{-\infty}^{0} exp^{-(t^2)}\sqrt{2}dt \quad + \quad \frac{2}{2\sqrt{\pi}}\int_{0}^{z/\sqrt{2}} exp^{-(t^2)}dt \tag{2.70}$$

The first integral term of Equation (2.70), can be simplified by reason of symmetry of the normal distribution as

$$\frac{1}{\sqrt{2\pi}}\int_{-\infty}^{0} exp^{-(t^2)}\sqrt{2}dt=\frac{1}{\sqrt{2\pi}}\int_{0}^{\infty} exp^{-(t^2)}\sqrt{2}dt=\frac{1}{2} \tag{2.71}$$

The second integral term can also be simplified using the error function (*erf*) approximation as

$$\frac{1}{2}\left(\frac{2}{\sqrt{\pi}}\int_{0}^{z/\sqrt{2}} exp^{-(t^2)}dt\right)=\frac{1}{2}erf\left(z/\sqrt{2}\right) \tag{2.72}$$

Hence,

$$\phi(z)=\frac{1}{2}\left(1+erf\left(z/\sqrt{2}\right)\right) \tag{2.73}$$

Equation (2.73) can thus be used to find a solution for Equation (2.65) knowing P_0, P_a, and n. Conversely, Equation (2.73) can be used to estimate the critical boundaries given the values of $\phi(z)$ as:

$$z=erfinv\left(2\phi(z)-1\right)\sqrt{2} \tag{2.74}$$

It is worth mentioning that Equation (2.66) can also be approximated using the series relation in Equation (2.75).

$$\phi(z)=\left\{\frac{1}{2}+\frac{1}{\sqrt{2\pi}}\left[\sum_{k=0}^{\infty}\frac{(-1)^k z^{(2k+1)}}{(2k+1)2^k k!}\right]\right\} \tag{2.75}$$

CHAPTER 3

3.0 Development of the Health Monitor

3.1 Model States as a Markov Chain

When a controller is in operation, the actuating errors (Setpoint minus Controlled Variable) are generated sequentially as shown in Figure 3.1.

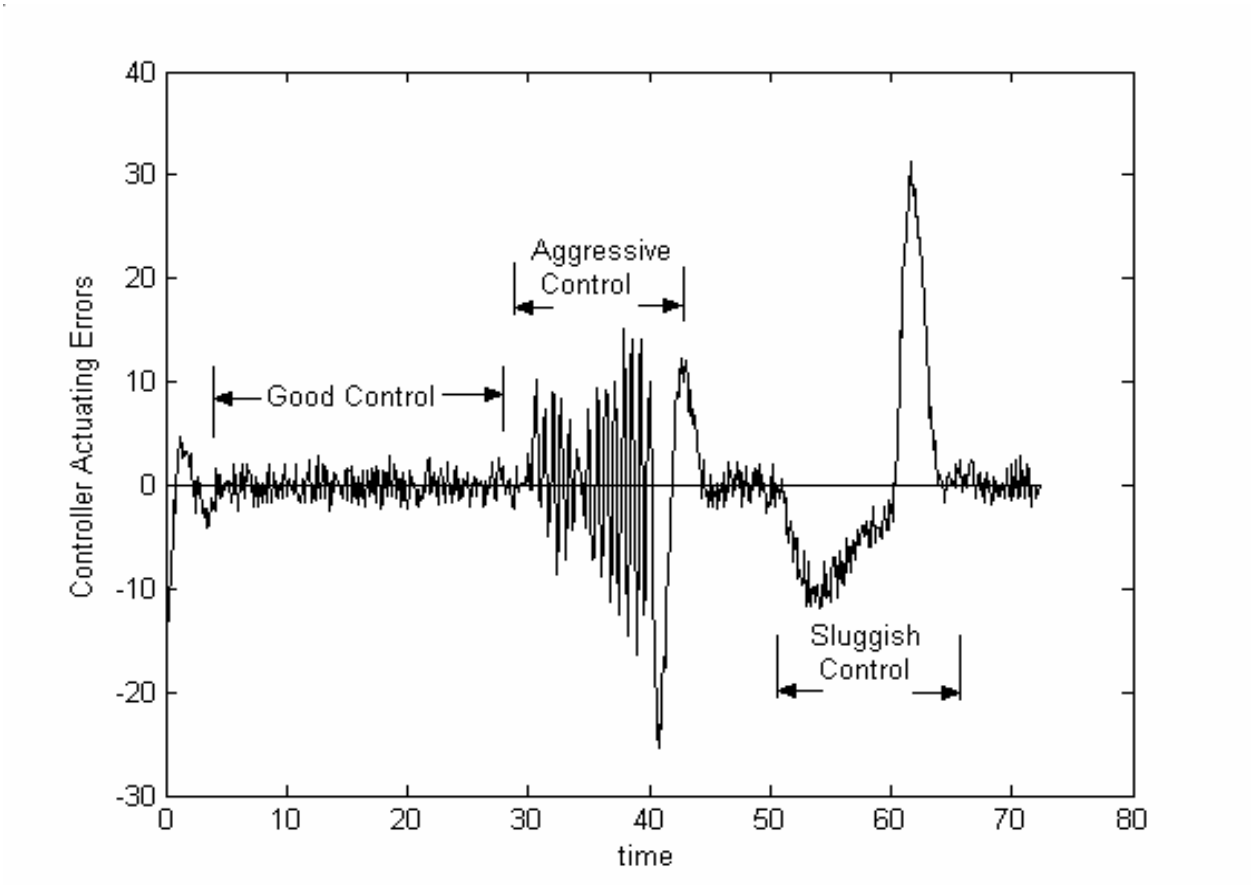


Figure 3.1 Actuating Error Signals in a Time series of Controlled Data

Labeled in Figure 3.1 is a period of good control when a controller is able to desirably manipulate a controlled variable in order to make the process stable and minimize deviations of the process output variable from the process setpoint. The Figure also illustrates a period when a controller is aggressive, resulting in increased oscillations in the process output. Lastly, a period of sluggish control is labeled. Not shown however, are other examples of constraint encounters, stiction, or continuous disturbances, all of which are possible nuisances that can occur in a control loop and are analyzed by the proposed method in this work.

The actuating errors (Setpoint-Controlled Variable) as they occur are labeled showing their run length in Figure 3.2. Errors above the mean value of zero are labeled as positive and those below are labeled as negative. If the actuating error persists on one side of the mean, the run length numbering continues on that side with the appropriate sign. However, anytime a zero crossing occurs, it signifies a sign change, and the run numbering also changes from positive to negative (or vice versa) and begins again with either +1 or –1 as appropriate. If the error has a value of zero, by definition in this work, it is not a zero crossing, and the run length continues to increase, with the run number bearing the same sign as the prior run. A run characterized by a positive (+) or negative (-) sign is referred to as a state. Shown in Figure 3.2, for illustration, the maximum state number is 4. Hence, the number of states will vary between -4 and +4 inclusive but excluding zero. This means that runs of 5, 6, or more, remain in a state of either +4 or -4 depending on the sign characterizing the run. Thus, when the actuating error visits a state that is higher than the maximum number, the run length increases but the state does not

change. Another characteristic of the data in Figure 3.2 is that each transition is binomially distributed.

For instance, given that an actuating error run is in a state of $+i$, it only has 2 (i.e. binomial) transition options for the next observation. It can either move from the state of $+i$ to the state of +i+1 or make a zero crossing to a state of -1. If it is in a state of $-i$, it also has 2 options of either moving to a state of -i-1, or making a zero crossing to a state of +1.

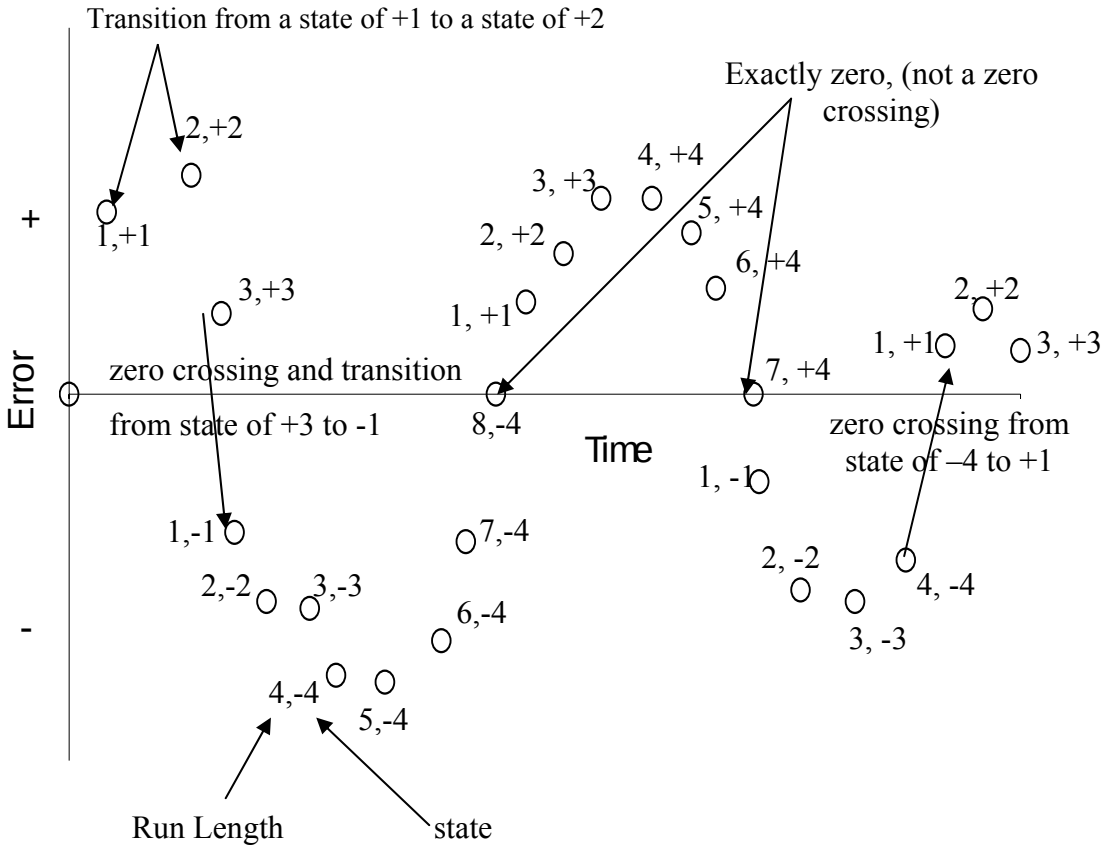


Figure 3.2 **Controller Run length, Zero Crossing, State and Transition State (O Represents an Actuating Error Value** (The Adjacent Number is the State)

Thus once the actuating error has visited a particular state, its next transition is not governed by how it got to this state. It is entirely dependent on the present state only. In other words, whether the next transition will result in a zero crossing or not has nothing to do with visits to a prior state or states. However, the next transition is entirely dependent on this present location of the data. This is the Markov property, which forms the broad basis for this work. The directed paths in Figure 3.2, which indicate the state transitions, are modeled with k total states ($k = 8$) also referred to as the run length. In general, the run length can be any number, but for practical purposes, it is appropriate to limit states. Any run length higher than the extreme state (+E) remains in the extreme state. The Markov chain Model is shown diagrammatically in Figure 3.3.

In general, let n_i denote the total number of samples (transitions) that have ever visited (entered) the state of i, (i could be $\pm$). For all interior states, this number of samples also denotes the total number of samples in that state. The only exceptions to this involve the states of ±1 and the extreme state of ±E. For this model, except for the extreme states, once a state is occupied or visited, the data cannot occupy that state again at the next immediate transition. Although, a state can be visited or occupied infinitely often, future visits to all interior states do not occur immediately at the next transition but rather at some time in the future. The only exception to this is the extreme state.

All states are positive recurrent and the entire model once defined forms a class. This is because all states communicate and the chain is irreducible as can be seen in Figure 3.3.

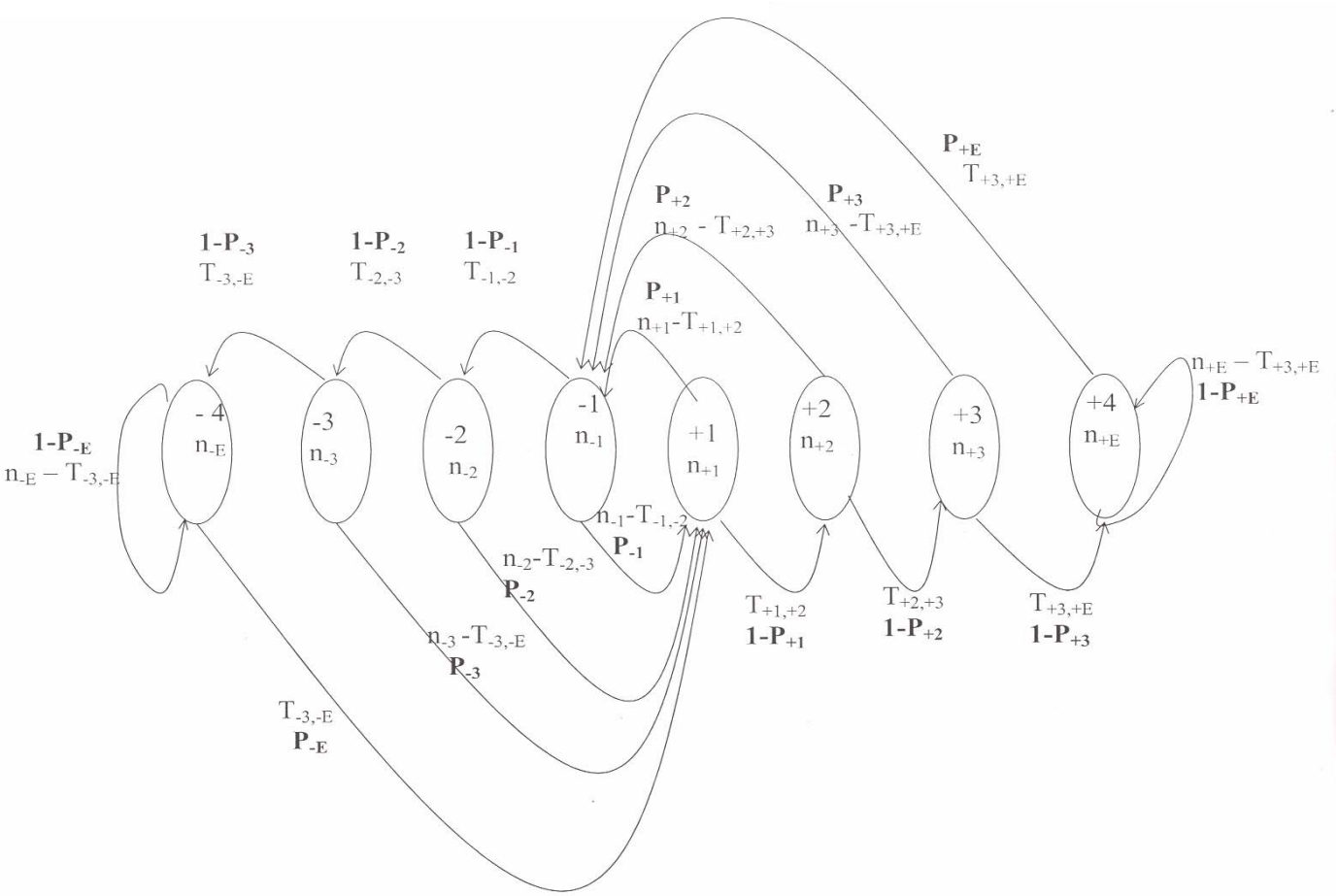


Figure 3.3 Illustration of Modeling of States Using Markov Chains to Determine the Transition Probabilities

Once states have been modeled as a Markov chain, the states transition probabilities can be determined as follows. Let $T_{i,j}$ denote the number of samples that do not make a zero crossing but rather leave the state of i and enter the state j, (i.e. j not equal ±1). Also, let $n_{\pm i}$ denote the number of samples that have ever visited the state of $\pm i$ respectively. Furthermore, for the general case in Figure 3.3, let P_{+i} denote the transition probability of the number of samples exiting the state of +i and entering the state of -1. Then,

$$P(+i) = \frac{n_{(+i)} - T_{+i,+i+1}}{n_{(+i)}} \tag{3.1}$$

Similarly, if P_{-i} denotes the probability of transition from a state of -i to a state of +1, then

$$P(-i) = \frac{n_{(-i)} - T_{-i,-i-1}}{n_{(-i)}} \tag{3.2}$$

Unlike all interior states, state transitions from the extreme states differ. Whereas transitions from all interior states either make a zero crossing or move to the next absolute higher state, transitions from the extreme states either make a zero crossing or re-enter themselves. Let ±E denote the positive or negative extreme states, and P_E denote the proportion of samples that leave the extreme and make a zero crossing to the state of ±1 respectively. Then, the transition probability from the state +E to the state -1 say is given by:

$$P(+E) = \frac{T_{+E-1,+E}}{n_{+E}} \tag{3.3}$$

Similarly, the transition probability from the state of –E to the state +1 is given by:

$$P(-E) = \frac{T_{-E-1,-E}}{n_{-E}} \tag{3.4}$$

The transition probabilities associated with each state can be illustrated in an array as shown in Table 3.1. From Table 3.1 and using row 3 for instance, if the actuating error is in a state of +1, then it can only visit the state of +2 with a probability 1-P(+1) or make a zero crossing to the state of -1 with a probability of P(+1). Similarly, considering row 8 for instance, if the actuating error is in a state of -2, then it has only two options of either visiting the state of -3 with a probability of 1-P(-2) or making a zero crossing to visit the state of +1 with a probability P(-2).

Table 3.1 Markov Transition Probability Matrix (Shown Only for 8 Total States)

		To State							
		+1	**+2**	**+3**	**+4**	**-1**	**-2**	**-3**	**-4**
From State	**+1**	0	1-P(+1)	0	0	P(+1)	0	0	0
	+2	0	0	1-P(+2)	0	P(+2)	0	0	0
	+3	0	0	0	1-P(+3)	P(+3)	0	0	0
	+4	0	0	0	1-P(+4)	P(+4)	0	0	0
	-1	P(-1)	0	0	0	0	1-P(-1)	0	0
	-2	P(-2)	0	0	0	0	0	1-P(-2)	0
	-3	P(-3)	0	0	0	0	0	0	1-P(-3)
	-4	P(-4)	0	0	0	0	0	0	1-P(-4)

Where the transitions probabilities are zero in Table 3.1 indicate that such visit are not allowed. For instance, an actuating error in a state of $\pm i$ cannot visit itself if i is an interior state. Neither can it visit a state $(+i\ +2)$ or $(-i\ -2)$ say, in just one step. Transitions are either from a state of $+i$ to a state of $(+i\ +1)$ or from $-i$ to $(-i\ -1)$ or make a zero crossing. It is only in the extreme state that transitions occur from the state of $\pm E$

to the state $\pm E$ with respective probabilities of either $1-P(+E)$ or $1-P(-E)$ or make a zero crossing with probabilities of either $P(+E)$ or $P(-E)$.

Appendix H provides a brief description on the deference between state space modeling of a time series data and state modeling of a data in a Markov chain.

3.2 Overall Structure of the Health Monitor

In this section, the general development of the monitor is discussed. In order to obtain the information used for hypothesis testing and hence making decisions, data must be collected for a period during which controller performance is judged by operators or engineers to be acceptable. The data collected during this period will be referred to as reference data. Once the reference data has been collected, it is used to determine the transition probabilities associated with each state, the number of samples that need to visit a particular state if control performance is good, a window of data that need to be used for statistical comparison and control limits associated with each state by the following procedure:

3.2.1 Estimate the Sampling Ratio

The health monitor is initialized to run with 8 total states (4 on both the "+" and "-" runs) at a sampling frequency equal to controllers sampling frequency. This means that the ratio of number of data from the controller to the number of data sampled by the monitor is unity. This ratio will be referred to as the sampling ratio. Hence, a sampling ratio of unity denotes that the controller and the health monitor both sample data at the same rate or frequency. For instance, a sampling ratio of three means that, for every

three controller actuating error samples, the health monitor samples one actuating error to analyze. The relation between sampling ratio, controller sampling time interval and monitor sampling time interval is discussed below.

Let f_c denote the controller sampling frequency, in units of number of controller samples/time. Also, let f_H denote the sampling frequency of the health monitor also in units of number of health monitor samples/time. Moreover, let SR denote the sampling ratio. Then,

$$SR = \frac{Number\ of\ Controller\ Samples / Time}{Number\ of\ HealthMonitor\ Samples / Time} \tag{3.5}$$

$$SR = \frac{f_c}{f_H} \tag{3.6}$$

But $$f_c = \frac{1}{Controller\ Sampling\ Time\ Interval} = \frac{1}{\Delta T_c} \tag{3.7}$$

And $$f_H = \frac{1}{Health\ Monitor\ Sampling\ Time\ Interval} = \frac{1}{\Delta T_H} \tag{3.8}$$

Hence, $$SR = \frac{1/\Delta T_c}{1/\Delta T_H} = \frac{1}{\Delta T_c} * \frac{\Delta T_H}{1} \tag{3.9}$$

Or

$$\Delta T_H = \Delta T_c * SR \tag{3.10}$$

Thus, the health monitor sampling time interval is given by the product of the controller sampling time interval and the sampling ratio. Alternatively, the sampling frequency of the health monitor is given by:

$$f_H = \frac{f_c}{SR} \tag{3.11}$$

The monitor is automated to analyze any length of data until all the transition probabilities lie between a desired range such that they are not too close to zero or unity. For the analysis in this work, the range is chosen to lie between 0.25 and 0.75 inclusive. If at the end of the first iteration, any transition probability is outside this range, the monitor automatically adjusts the sampling ratio from 1 to 2 and then determines the transition probabilities again. This process iteratively continues, until all the state transition probabilities for the initial 8 total states lie within the pre-chosen range. The reason for choosing this range initially is to avoid a situation where control limits, which are to be determined, lie too close to unity or zero because the reference probabilities were already that close. Once all transition probabilities lie within the chosen range, the monitor then adjusts the total number of states until at most 10 percent of the entire data lie in the extreme states. This is explained below.

3.2.2 Estimate the Number of States

In general, when a controller is performing optimally well, its actuating errors should be characterized by frequent zero crossings such that as time progresses the mean error is approximately close to zero. However, in practice this does not actually happen unless the noise effect in the data is purely white, which is independent of each other and identically distributed (IID). This means that ideally, if every thing was perfect for a well-tuned control loop, then the distribution of the actuating error samples will be approximately Gaussian, normal and independently distributed (NID). This will cause lower states to be populated with more samples than higher states as shown in Figure 3.4.

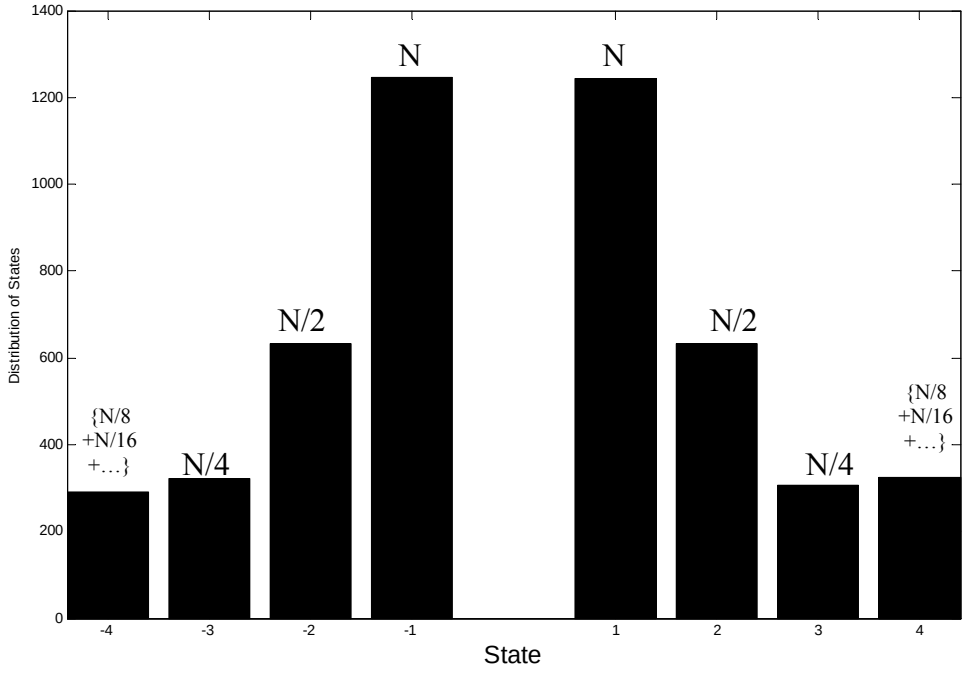


Figure 3.4 Distributions of States for Historical Good Data

This implies that for eight (8) total states, if N denotes the total number of samples in each of the lowest states (i.e. +1 and -1 states), then the states of +2, +3 say, will each have approximately $N/2$ and $N/4$ samples respectively. This is because, since the data is IID, there is no correlation between them. Hence, each transition has equal chance of about 50% to either make a zero crossing or visit the next absolute higher state. This means that the number of samples in adjacent higher states will differ from their adjoining lower state by a factor of 0.5. All run lengths equal to or exceeding the extreme state stay in that state and since the state of +4 is an extreme state, all sample sizes from $N/8$, $N/16$, $N/32$, etc., will all be cumulative parts of the samples visiting the

state of +4. Consequently, if n_{+E} denotes the total number of samples in the extreme state of +4, then

$$n_{+E} = n_{+4} + n_{+5} + n_{+6} + \ldots + n_{+\theta}$$

Where, θ denotes the largest run length that would have occurred if states were not limited to $\pm E$

$$n_{+E} = \frac{N}{8} + \frac{N}{16} + \frac{N}{32} + \ldots \tag{3.12}$$

If N_{+p} denotes the total number of samples in all the positive states, then

$$N_{+p} = n_{+1} + n_{+2} + n_{+3} + n_{+E}$$

$$N_{+p} = (N) + \left(\frac{N}{2}\right) + \left(\frac{N}{4}\right) + \left(\frac{N}{8} + \frac{N}{16} + \ldots\right)$$

Thus, the total number of samples in the extreme state can be estimated as the sum of a geometric series as:

$$n_{+E} = \lim_{\theta \to \infty} \left(n_4 + n_5 + n_6 + \ldots + n_\theta\right) \tag{3.13}$$

$$n_{+E} = \lim_{\theta \to \infty} \left[\frac{N}{8}\left(1 + \frac{1}{2} + \frac{1}{4} + \frac{1}{8} + \ldots\right)\right] \tag{3.14}$$

$$n_{+E} = \lim_{\theta \to \infty} \left[\frac{N}{8}\left(1 + \frac{1}{2} + \frac{1}{4} + \frac{1}{8} \ldots\right)\right] = \frac{N}{8}\left(\frac{1 - \left(1/2\right)^{\theta}}{\left(1 - 1/2\right)}\right) \tag{3.15}$$

$$n_{+E} = \frac{N}{4} \tag{3.16}$$

The number of samples in the other extreme sate of –E will be approximately same as Equation (3.16). Now, for the entire positive states' side of the chain starting from +1, +2, … , +E-1, and +E, it can be shown that:

$$N_{+p} = \left(n_{+1}\right) + \left(n_{+2}\right) + \left(n_{+3}\right) + \left(n_4 + n_{+5} + n_{+6} + \ldots n_{+\theta}\right)$$

$$N_{+p} = \left(n_1 + n_2 + n_3 + \left(n_{+E}\right)\right) \tag{3.17}$$

$$N_{+p} = \left(N + \frac{N}{2} + \frac{N}{4} + \frac{N}{4}\right) \tag{3.18}$$

$$N_{+p} = 2N \tag{3.19}$$

Notice that, the actuating error samples are geometrically distributed on each half of Figure 3.4 (i.e. from -1 to -4 and +1 to +4) for the 8-state case under discussion. Since the distribution of samples in one half is identical to the other half, the total number of samples for the entire chain can be approximated as:

$$N_{total} = 2N_p = 4N \tag{3.22}$$

Hence, if "rp" denotes the ratio of the number of samples in any of the extreme states to the total number of samples in the entire chain, then "rp" is given by:

$$rp = \frac{N/4}{4N} = \frac{1}{16} \cong 0.0625 = 6.25\% \tag{3.23}$$

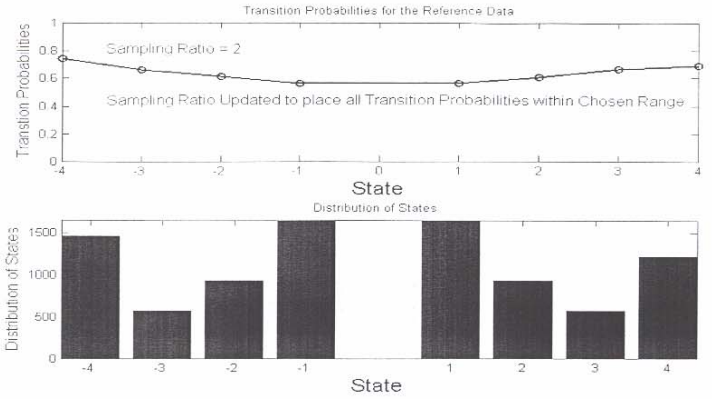
Thus, ideally about 6.25 percent of the total number of samples analyzed will lie in each of the extreme states in an 8-state model, assuming that the distribution of actuating error samples was independent. Intuitively then, a value of 10 percent is proposed as an approximation of the fraction of the number of samples that should lie in any of the extreme states.

Therefore, during analysis of the reference data, if after using the initial total number of states for analysis, it is determined that more than 10 percent of the total data is in the extreme states, the total number of states is updated by 2 (one positive and one

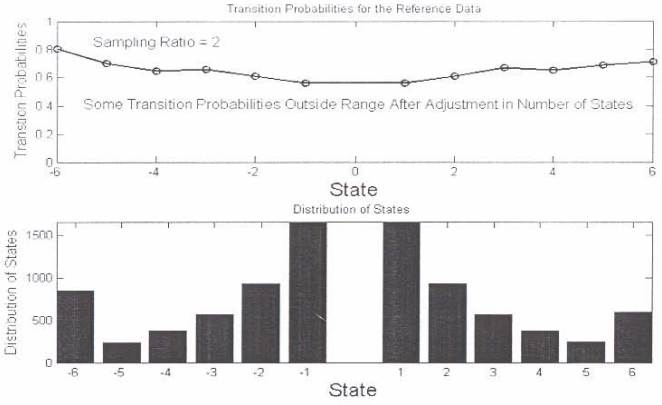
negative state are added) and the analysis repeated. This process is repeated until the monitor finds the number of states that satisfy the ten percent condition.

After a set of states is determined such that not more than 10% of the data lie in the extreme state, it is often possible that some of the transition probabilities associated with any newly added states may lie outside the chosen range. Consequently, the monitor updates the sampling ratio and analyzes the data again so that all the transition probabilities lie within the chosen range. This procedure is illustrated in Figure (3.5a to d). For instance, Figure (3.5a) illustrates the nominal analysis of the reference data starting with a sampling ratio of unity and 8 total states. As can be seen, after the initial data was analyzed, some of the transition probabilities were greater than the desired upper limit 0.75. Hence, the sampling ratio was updated from unity to 2 and the entire process repeated. After analysis of the data using the new sampling ratio, all the transition probabilities fell below 0.75 as shown in Figure (3.5b). Nevertheless, the extreme states are too populated with data, therefore the monitor adjusts the number of states from 8 to a total of 10 (1 new positive state and 1 new negative state on each side) and repeats the analysis using this new number of states but keeping the sampling ratio at the prior value. This process is repeated until no more than 10% of the total number of samples being analyzed is in the extreme states. For this example, 12 total states (± 6) were determined by the monitor to be enough in order to achieve the user-desired conditions. This is shown in Figure (3.5c). It can however be noticed in Figure (3.5c) that the transition probabilities in the extreme states of "-6" is outside the desired range. Therefore, the monitor adjusts the sampling ratio again until all the probabilities fall between 0.25 and 0.75 inclusive as shown in Figure (3.5d).

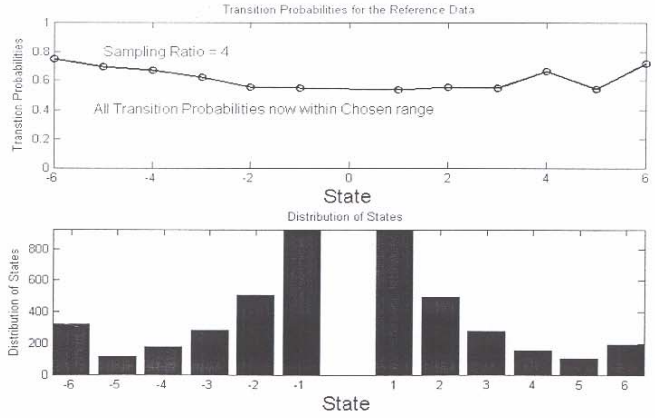
a). Nominal Analysis



b). Increasing Sampling Ratio to get Transitions between 0.75 and 0.25



c). Adding states to get ≤ 10% in Extremes



d). Readjusting Sampling Ratio

Figure 3.5 Stages in the Analysis of the Reference Data

3.2.3 Identify States Containing the Least Number of Reference Samples

Generally, after the reference data is analyzed, the minimum number of samples would lie either in the extreme state or in the penultimate state. It is important for statistical analysis that number of samples in the state having the least number of samples be significantly large enough in order to minimize statistical errors, namely Type-1 and Type-II errors. Essentially, after determining the transition probabilities using the reference data, the monitor observes all the states to determine the state (on each half of the chain) that contains the least number of samples. It is possible for the minimum number of samples to be located in the extreme state on one half while on the other half, it will occur in the penultimate state. It is also possible that the minimum number of samples will be located in the penultimate state in both halves or in the extreme in both halves. Either way all that the monitor needs to know is the state in which this minimum number of samples is located. Once this state is known, the monitor calculates the number of samples that need to be in that states in order to meet the user specified criteria on Type-I and Type-II errors. Given that the level of significance associated with the state that is identified to contain the minimum number of samples is α_k, since statistical errors can occur in either tail of the distribution, a two-tailed test analysis is conducted. This implies that for each tail the level of significance will be given by $\frac{\alpha_k}{2}$. This is illustrated in Figure 3.6.

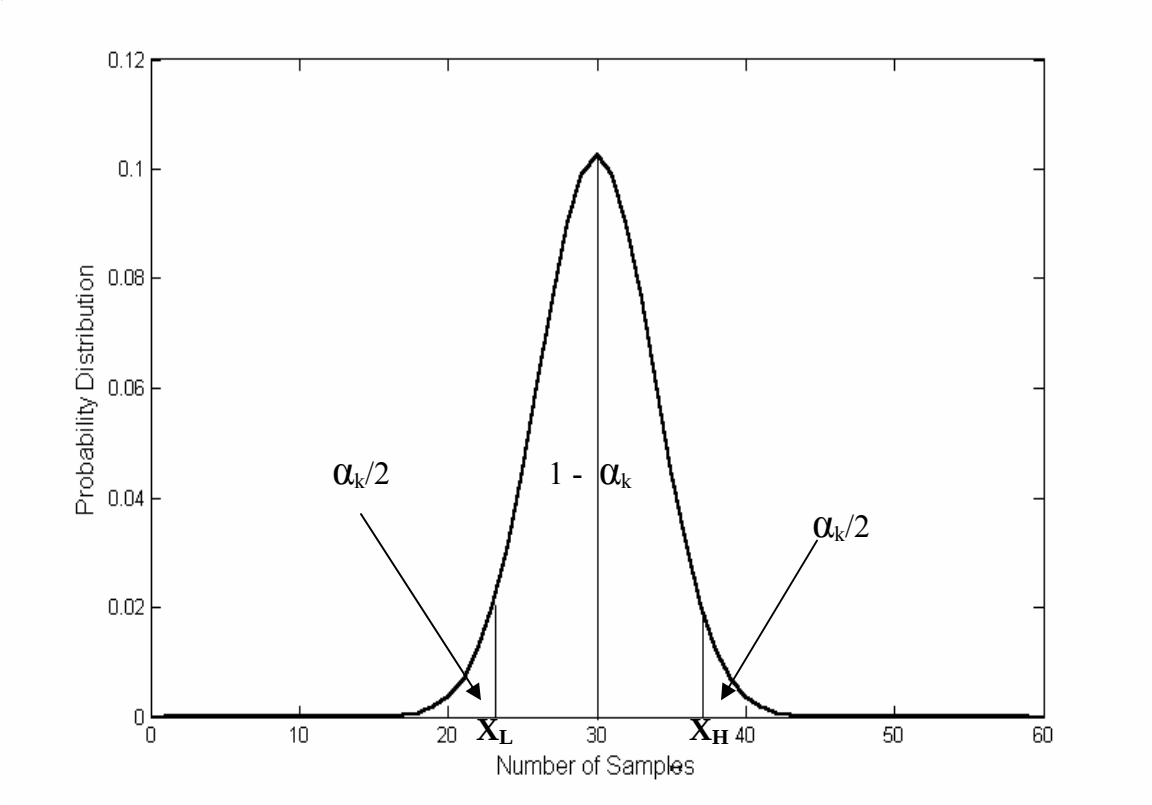


Figure 3.6 Analysis of Type-I Error Rate for State with Least Number of Reference Data

Hence, the number of samples X_L, required to obtain the cumulative probability $\frac{\alpha_k}{2}$ in the left hand tail, is given by the Equation (2.52) as:

$$\frac{\alpha_k}{2} = \sum_{x=0}^{x_L} \binom{n}{x} P_0^x \left(1 - P_0\right)^{n-x} \tag{3.24}$$

Similarly, the number of samples X_H, that give the cumulative probability $\frac{\alpha_k}{2}$, in the right hand tail is given by Equation (2.54) as:

$$\frac{\alpha_k}{2} = \sum_{x=x_H}^{n} \binom{n}{x} P_0^x \left(1 - P_0\right)^{n-x} \tag{3.25}$$

Equation (3.25) can be written in another form as

$$1-\frac{\alpha_k}{2}=\sum_{x=0}^{X_H}\binom{n}{x}P_0^x\left(1-P_0\right)^{n-x} \tag{3.26}$$

Thus if α_k (Type-I error rate for a particular state), and the number of samples, X_L and X_H required to give α_k are all known, then Equations (3.24) and (3.26) can be used to determine the number of samples "n" to place in a particular state in order to obtain a desired error rate. There are however, four unknown variables namely the type one error rate associated with a particular state α_k, the number of samples X_L, that give the lower control limit, the number of samples X_H, that give the upper control limit, and total number of samples n, in that state). Nevertheless, there are only two Equations making solution of Equations (3.24) and (3.26) impossible at first sight. One way to get around this is to select a desired Type-II error rate when future transition probabilities deviate by some amount and develop and a third equation involving both the desired Type-I and Type-II error rates. For the examples in this work, a desired Type-II error rate equal to the overall Type-I error is set when the Transition probability differ from the reference value by 90 percent.

Let P_a denote some future transition probability (different from the reference transition probability P_0), and which is desired to be detected with a Type-II error rate specified. Also, let β denote the Type-II error rate when P is equal to P_a. Then from Figure 3.7, the shaded area gives the Probability of a Type-II error.

With reference to Equation 2.56, the Type-II error rate for the chosen state can be estimated using

$$\beta_k = \sum_{x=0}^{x_H} \binom{n}{x} P_a^x \left(1-P_a\right)^{n-x} - \sum_{x=0}^{x_L} \binom{n}{x} P_a^x \left(1-P_a\right)^{n-x} \tag{3.27}$$

With β known, there are now three nonlinear Equations and four unknown variables (X_L, X_H, α_k, and n). Thus, there is one more variable than the number of Equations. A straight form solution can still not be determined at this stage. In addition, at this stage, the Type-I error rate associated with each state is not known. However, it can be estimated if the overall Type-I error rate for the entire experiment is known. The Type-I error rate for each state is determined below.

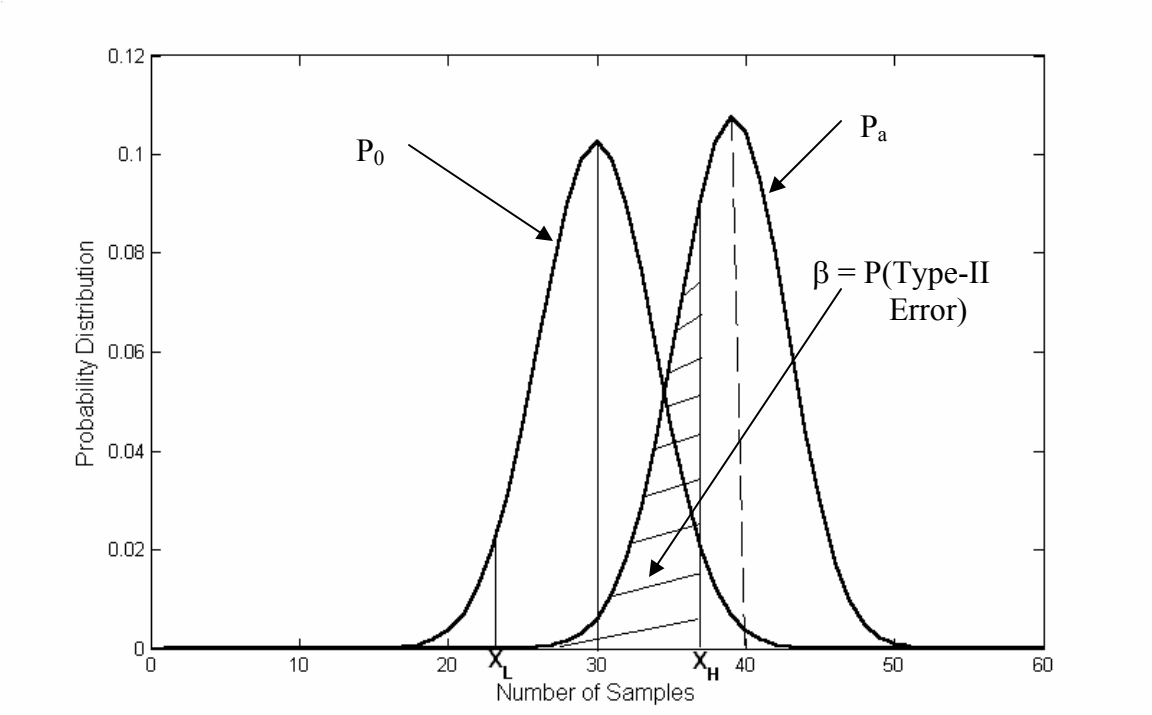


Figure 3.7 Analysis of Type-II Error for State with Least Number of Reference Data

Let α_T, denote the entire Type-I error rate desired by the engineers or operators for the entire test. However, each window or chain contains 'N_S' total states where

$\left(N_s = \sum_{k=-E}^{+E} k \right)$ and each state transition is associated with a level of significance denoted by α_k. The health monitor flags if any one transition is in any of critical regions of the probability distribution associated with any state as shown in Figure 3.7.

Given the null hypothesis: H_0: $P = P_0$ and H_a: $P \neq P_0$

Where, H_0 is the null hypothesis that all future state transition probabilities are equal to the reference transition probability P_0. Let $P(T_i)$ = probability of the i^{th} transition that is not in the extreme region. Then;

$P(H_0)$ = P(T_{-E} is not extreme and T_{-E+1} is not extreme and ••• T_{-1} is not extreme and T_{+1} is not extreme ••• and T_{+E} is not extreme).

$P(H_0)$ = P(T_{-E} is not extreme)* P(T_{-E+1} is not extreme)*•••* P(T_{-1} is not extreme)* P(T_{+1} is not extreme)* •••* P(T_{+E-1} is not extreme)* P(T_{+E} is not extreme)

But $P(H_0) = 1 - \alpha_T$

Hence,

$1 - \alpha_T$ = [1-P(T_{-E} is extreme)]*…* [1-P(T_{-1} is extreme)]*[1-P(T_{+1} is extreme)]* •••* [1-P(T_{+6} is extreme)]

But, $P(T_{-k}$ is extreme) = α_k, Therefore,

$$1-\alpha_T = [1-\alpha_{-E}]*...*[1-\alpha_{-1}]*[1-\alpha_{+1}]*...*[1-\alpha_{+E}] \qquad (3.28)$$

For the binomial distribution, a necessary requirement is that: each trial must be independent, each trial must result in only two possible outcomes, the probability of success on each trial must remain constant and all trials must be identical. Hence, if all

future transitions are identical, the Type-I error rate for each of the "k" states can be assumed identical. Hence,

$$1-\alpha_T = (1-\alpha_k)*(1-\alpha_k)*(1-\alpha_k)*\ldots \tag{3.29}$$

$$1-\alpha_T = (1-\alpha_k)^{N_s} \tag{3.30}$$

Solving for α_k gives:

$$\alpha_k = 1-\sqrt[N_s]{(1-\alpha_T)} \tag{3.31}$$

Where, N_S denotes the total number of states and k denotes each individual state. Equation (3.31) can be used to determine the Type-I error rate associated with each state, given the overall Type-I error rate for the entire test. It is worthwhile to mention here that Equation (3.31) can be further approximated using a Taylor series expansion to give the Bonferroni approximation that is used in multiple hypotheses testing in experimental analysis. From Equation (3.31), let

$$\alpha_k = 1-f(\alpha_T) \tag{3.32a}$$

Where $f(\alpha_T) = \sqrt[N_s]{(1-\alpha_T)}$ (3.32b)

Using, a Taylor series approximation and expanding Equation (3.32b) about a reference value α_0, $f(\alpha_T)$ can be written as

$$f(\alpha_T) = f(\alpha_0) + (\alpha_T-\alpha_0) f'(\alpha_0) + \frac{(\alpha_T-\alpha_0)^2}{2} f''(\alpha_0) + \ldots \tag{3.33}$$

Ignoring the second order and higher terms and letting $\alpha_0 = 0$, Equation (3.33) can be simplified as

$$f(\alpha_T) = f(\alpha_0) + (\alpha_T-\alpha_0) f'(\alpha_0) \tag{3.34}$$

But, from Equation (3.32b), the first derivative of $f(\alpha_T)\big|_{\alpha_T=\alpha_o}$

$$f'(\alpha_T)\Big|_{\alpha_T=\alpha_o} = -\frac{(1-\alpha_0)^{(1/N_s-1)}}{N_s} \tag{3.35}$$

Setting $\alpha_0 = 0$ in Equation (3.35)

$$f'(\alpha_o) = -\frac{1}{N_S} \tag{3.36a}$$

Also,

$$f(\alpha_0) = \sqrt[N_S]{(1-0)} = 1 \tag{3.36b}$$

Substituting Equations (3.36a) and (3.36b) in Equation (3.34) gives

$$f(\alpha_T) = 1 + (\alpha_T - 0)\left(-\frac{1}{N_S}\right) = \left(1 - \frac{\alpha_T}{N_S}\right) \tag{3.37}$$

Substituting Equation (3.37) in Equation (3.32a) gives

$$\alpha_k = 1 - \left(1 - \frac{(\alpha_T)}{N_s}\right) \tag{3.38a}$$

Equation 3.38a simplifies to

$$\alpha_k = \frac{\alpha_T}{N_s} \tag{3.38b}$$

Equation 3.38b is the well-known Bonferroni approximation. It must be mentioned that since the Taylor series is truncated after the second order term, it means that the general error involved in the approximation in Equation (3.38b) is of the order $O\left(|\alpha_T - \alpha_0|\right)^{\lambda+1}$ where λ is the order of the highest derivative used in the Taylor series estimate (i.e. $\lambda = 1$). However, the approximation gets better as the number of states N_S becomes large and the error rate α_T, gets smaller. Consequently, for this work, the exact form of Equation (3.31) is used in all the calculations of α_k for the monitor rather than

Equation (3.38b). Table 3.2 illustrates the difference between the exact calculation for determining α_k and the Bonferroni approximation (α_B) for determining same, for various values of the overall Type-I error rate α_T, and different values of total number of states N_s. It can be noticed that for a given α_T, the difference between α_k and α_B, decreases as N_s increases with the relative error also increasing. Also, for a given N_s, as α_T decreases, both α_k and α_B decrease and the relative error also decreases. In all cases, it is seen that the difference between α_k and α_B becomes negligibly small as N_s increase and α_T decrease. In order to avoid ambiguities as to where the approximation is best, α_k is used for all estimates in this work.

Table 3.2 Values of the Estimated Type-I Error Rate compared with Bonferroni Approximation

	α_T =0.1			α_T =0.05			α_T =0.01			α_T =0.0030		
Ns	α_k*10^3	α_B *10^3	Error, %	α_k*10^3	α_B *10^3	Error, %	α_k*10^3	α_B *10^3	Error, %	α_k*10^3	α_B *10^3	Error, %
8	13.084	12.500	4.4614	6.391	6.250	2.2085	1.255503	1.250000	0.43832	0.375493	0.375000	0.13132
10	10.481	10.000	4.5869	5.116	5.000	2.2712	1.004529	1.000000	0.45083	0.300406	0.300000	0.13507
12	8.742	8.333	4.6705	4.265	4.167	2.3129	0.837177	0.833333	0.45917	0.250344	0.250000	0.13757
14	7.498	7.143	4.7302	3.657	3.571	2.3427	0.717624	0.714286	0.46512	0.214585	0.14286	0.13936
16	6.563	6.250	4.7749	3.201	3.125	2.3650	0.627949	0.625000	0.46958	0.187764	0.187500	0.14070
18	5.836	5.556	4.8097	2.846	2.778	2.3824	0.558196	0.555556	0.47306	0.166903	0.166667	0.14174
20	5.254	5.000	4.8376	2.561	2.500	2.3963	0.502391	0.500000	0.47584	0.150214	0.150000	0.14257
22	4.778	4.545	4.8603	2.329	2.273	2.4077	0.456729	0.454545	0.47811	0.136559	0.136364	0.14326
24	4.380	4.167	4.8793	2.135	2.083	2.4172	0.418676	0.416667	0.48000	0.125180	0.125000	0.14382
26	4.044	3.846	4.8953	1.971	1.923	2.4252	0.386477	0.384615	0.48161	0.115551	0.115385	0.14431
28	3.756	3.571	4.9091	1.830	1.786	2.4321	0.358876	0.357143	0.48298	0.107298	0.107143	0.14472
30	3.506	3.333	4.9210	1.708	1.667	2.4380	0.334955	0.333333	0.48417	0.100145	0.100000	0.14508
32	3.287	3.125	4.9314	1.602	1.563	2.4432	0.314024	0.312500	0.48521	0.093886	0.093750	0.14539

Ns = Number of states

α_k = Type-I error rate for state k using Equation (3.31)

α_B = Type-I error rate for state k using Equation (3.38b)

α_T = Overall Type-I error rate for the test

% Error determined as $\left(\frac{\alpha_k - \alpha_B}{\alpha_k}\right)$*100 %

Once α_k is known, Equations (3.24), (3.26) can be solved simultaneously by first guessing a value for value for "n" and solving for X_L and X_H. Once X_L and X_H are known, use them together with the value of "n" to find a solution for Equation (3.27). If the solution agrees with the pre-chosen value of β, then the value of "n" is the desired minimum to place in the chosen state in order to achieve the set error rates. if not, then update the value of "n" by one and repeat the entire process. The algorithm for this process is shown in the flow chart in Figure 3.8

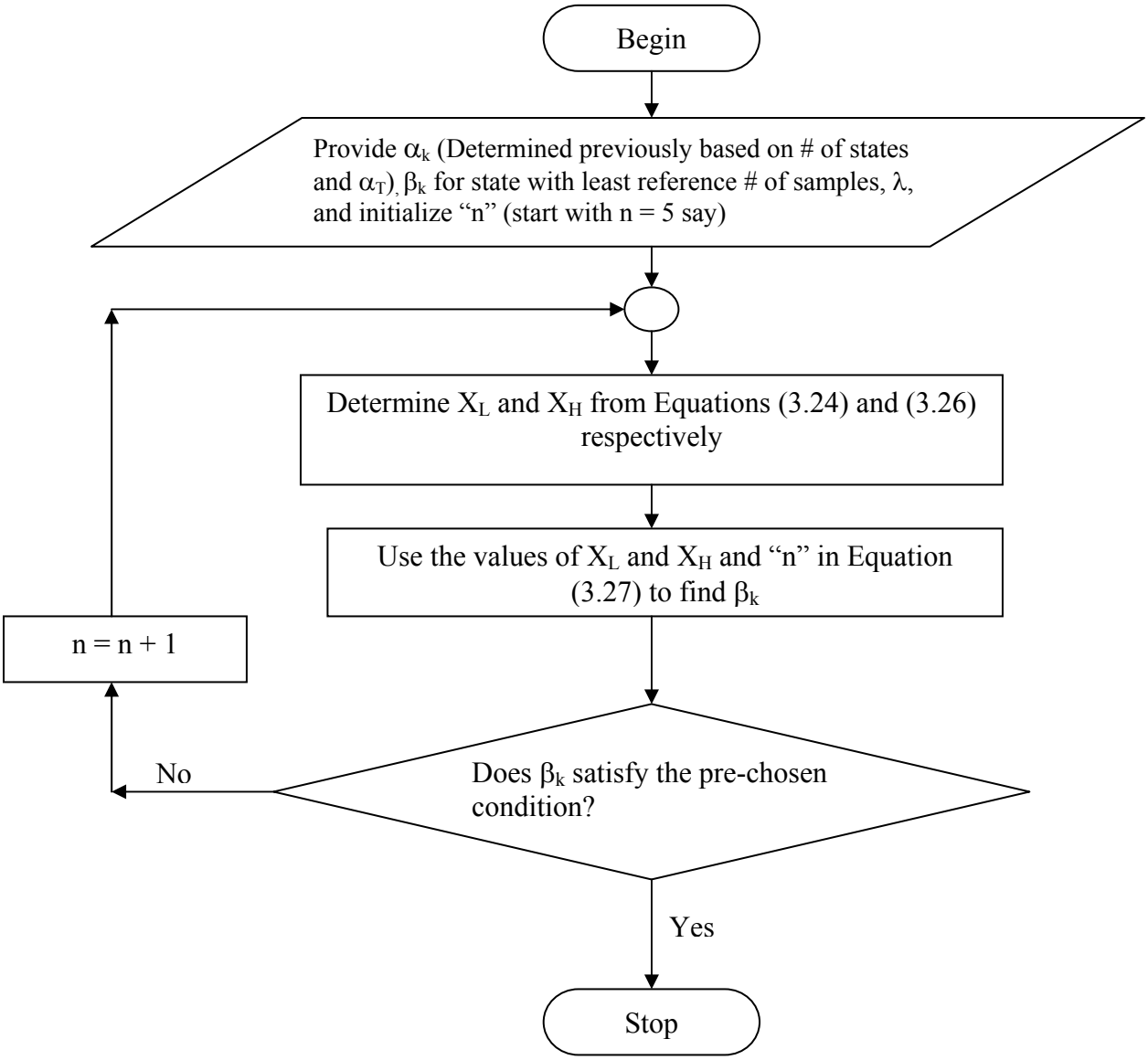


Figure 3.8 Flowchart for Determining the Number of Samples to Place in the State Haven the Least Number of Reference Data in order to balance Type-I and Type-II error

3.2.4 Estimate the Window Length

Once the number of samples "n" for the state with least number of reference data has been determined, the number of samples that need to be placed in all the other states are determined based on "n". The window length is estimated by considering each half of the chain separately (see Figure 3.3). Consider first the positive transition states and recalling that for all interior states:

$P(+i) = \frac{n_{(+i)} - T_{+i,+i+1}}{n_{(+i)}}$, but for all interior states, $T_{+i,+i+1} = n_{+i+1}$, Hence,

$$P(+i) = \frac{n_{(+i)} - n_{+i+1}}{n_{(+i)}} \tag{3.39}$$

For the penultimate state, let $n_{+E'}$ denote the number of samples leaving that state and moving into the extreme state, then $T_{+E-1,+E} = n_{+E'}$. Hence,

$$P_{(+E-1)} = \frac{n_{+E-1} - n_{+E'}}{n_{+E-1}} \tag{3.40}$$

Therefore, for the extreme state,

$$P_{(+E)} = \frac{n_{+E'}}{n_{+E}} \tag{3.41}$$

Case 1,

Assuming that the least number of reference data occurs in the extreme state (+E), then from Equation 3.40,

$$P_{(+E-1)} n_{(+E-1)} = n_{(+E-1)} - n_{+E'} \tag{3.42}$$

$$n_{(+E-1)} \left(1 - P_{(+E-1)}\right) = n_{+E'} \tag{3.43}$$

$$n_{(+E-1)} = \frac{n_{+E'}}{\left(1 - P_{(+E-1)}\right)} \tag{3.44}$$

From Equation (3.41), $n_{+E'} = n_{+E} P_{+E}$, substituting this in 3.44 gives

$$n_{(+E-1)} = \frac{n_{+E} P_{+E}}{\left(1 - P_{(+E-1)}\right)} \tag{3.45}$$

So, for the case where the least number of reference data is in the positive extreme state, set $n_{+E} = n$ (where "n" has been determined as per the algorithm in Figure 3.8). Then use Equation (3.45) to find n_{+E-1}. Thus,

$$n_{+E} = n \tag{3.46}$$

$$n_{+E-1} = \frac{nP_{+E}}{\left(1 - P_{+E-1}\right)} \tag{3.47}$$

From Equation (3.39), $P(+i) = \frac{n_{(+i)} - n_{+i+1}}{n_{(+i)}}$, Hence, $P_{(+E-2)} = \frac{n_{(+E-2)} - n_{(+E-1)}}{n_{(+E-2)}}$, Rearranging gives

$$n_{(+E-2)} = \frac{n_{(+E-1)}}{\left(1 - P_{(+E-2)}\right)}$$

$$n_{(+E-2)} = \frac{nP_{(+E)}}{\left(1 - P_{(+E-2)}\right)\left(1 - P_{(+E-1)}\right)} \tag{3.48}$$

Similarly,

$$n_{(+E-3)} = \frac{n_{(+E-2)}}{\left(1 - P_{(+E-3)}\right)}$$

$$n_{(+E-3)} = \frac{nP_{(+E)}}{\left(1 - P_{(+E-1)}\right)\left(1 - P_{(+E-2)}\right)\left(1 - P_{(+E-3)}\right)} \tag{3.49}$$

Continuing,

$$n_{(+3)} = \frac{n_{(+4)}}{\left(1 - P_{(+3)}\right)}$$

$$n_{(+3)} = \frac{nP_{(+E)}}{\left(1 - P_{(+E-1)}\right)\left(1 - P_{(+E-2)}\right)\left(1 - P_{(+E-3)}\right)...\left(1 - P_{(+3)}\right)} \tag{3.50}$$

Moreover,

$$n_{(+2)} = \frac{n_{(+3)}}{\left(1 - P_{(+2)}\right)}$$

$$n_{(+2)} = \frac{nP_{(+E)}}{\left(1 - P_{(+E-1)}\right)\left(1 - P_{(+E-2)}\right)\left(1 - P_{(+E-3)}\right)...\left(1 - P_{(+3)}\right)\left(1 - P_{(+2)}\right)} \tag{3.51}$$

Furthermore,

$$n_{(+1)} = \frac{n_{(+2)}}{\left(1 - P_{(+1)}\right)}$$

$$n_{(+1)} = \frac{nP_{(+E)}}{\left(1 - P_{(+E-1)}\right)\left(1 - P_{(+E-2)}\right)\left(1 - P_{(+E-3)}\right)...\left(1 - P_{(+3)}\right)\left(1 - P_{(+2)}\right)\left(1 - P_{(+1)}\right)} \tag{3.52}$$

Thus, on the positive states side of the chain for this case, the window length, WdL^{+} is given by:

$$WdL^{+} = n_{+1} + n_{+2} + n_{+3} + ... + n_{+E}$$

$$WdL^{+} = \frac{nP_{(+E)}}{\left(1-P_{(+E-1)}\right)\left(1-P_{(+E-2)}\right)\left(1-P_{(+E-3)}\right)\ldots\left(1-P_{(+3)}\right)\left(1-P_{(+2)}\right)\left(1-P_{(+1)}\right)} + \frac{nP_{(+E)}}{\left(1-P_{(+E-1)}\right)\left(1-P_{(+E-2)}\right)\left(1-P_{(+E-3)}\right)\ldots\left(1-P_{(+3)}\right)\left(1-P_{(+2)}\right)} + \frac{nP_{(+E)}}{\left(1-P_{(+E-1)}\right)\left(1-P_{(+E-2)}\right)\left(1-P_{(+E-3)}\right)\ldots\left(1-P_{(+3)}\right)} + \bullet\bullet\bullet + \frac{nP_{(+E)}}{\left(1-P_{(+E-1)}\right)} + n_{+E} \tag{3.53}$$

Factoring out the first term

$$WdL^{+} = \frac{nP_{(+E)}}{\left(1-P_{(+E-1)}\right)\left(1-P_{(+E-2)}\right)\left(1-P_{(+E-3)}\right)\ldots\left(1-P_{(+3)}\right)\left(1-P_{(+2)}\right)\left(1-P_{(+1)}\right)} * \left[\begin{array}{l} 1+\left\{\left(1-P_{(+1)}\right)\right\}+\left\{\left(1-P_{(+1)}\right)\left(1-P_{(+2)}\right)\right\}+\left\{\left(1-P_{(+1)}\right)\left(1-P_{(+2)}\right)\left(1-P_{(+3)}\right)\right\}+ \\ \bullet\bullet\bullet+\left\{\left(1-P_{(+1)}\right)\left(1-P_{(+2)}\right)\left(1-P_{(+3)}\right)\bullet\bullet\bullet\left(1-P_{(+E-5)}\right)\left(1-P_{(+E-4)}\right)\right\}+ \\ \left\{\left(1-P_{(+1)}\right)\left(1-P_{(+2)}\right)\left(1-P_{(+3)}\right)\bullet\bullet\bullet\left(1-P_{(+E-4)}\right)\left(1-P_{(+E-3)}\right)\right\}+ \\ \left\{\left(1-P_{(+1)}\right)\left(1-P_{(+2)}\right)\left(1-P_{(+3)}\right)\bullet\bullet\bullet\left(1-P_{(+E-3)}\right)\left(1-P_{(+E-2)}\right)\right\} \end{array}\right] + n_{+E} \tag{3.54}$$

Let q(i) = 1-P(i), then

$$WdL^{+} = \frac{nP_{(+E)}}{q_{(+1)}q_{(+2)}q_{(+3)}q_{(+E-3)}q_{(+E-2)}q_{(+E-3)}} * \left[\begin{array}{l} 1+q_{(+1)}+q_{(+1)}q_{(+2)}+q_{(+1)}q_{(+2)}q_{(+3)}+ \\ \bullet\bullet\bullet+\left\{q_{(+1)}q_{(+2)}\bullet\bullet\bullet q_{(+E-4)}\right\}+\left\{q_{(+1)}q_{(+2)}\bullet\bullet\bullet q_{(+E-3)}\right\}+n_{+E} \\ +\left\{q_{(+1)}q_{(+2)}\bullet\bullet\bullet q_{(+E-2)}\right\} \end{array}\right] \tag{3.55}$$

Equation 3.55 can be further simplified to give

$$WdL^{+} = \frac{nP_{(+E)}}{\prod_{i=1}^{+E-1} q_{(i)}}\left[1 + \sum_{k=1}^{+E-1}\left(\prod_{j=1}^{+E-1} q_{(j)}\right)\right] + n_{+E} \tag{3.56}$$

On the negative state side of the chain, if the least number of samples occurs in the extreme state then the number of samples to place in each state can be deduced by similarly reasoning as above. Again, set n_{-E} equal to "n" after going through the algorithm in Figure 3.8 for the negative side of the chain. It can be shown that the total number of samples to place in the negative states, WdL^{-} is given by

$$WdL_{n} = \frac{nP_{(-E)}}{\prod_{i=1}^{-E+1} q_{(i)}}\left[1 + \sum_{k=1}^{-E+1}\left(\prod_{j=1}^{-E+1} q_{(j)}\right)\right] + n_{-E} \tag{3.57}$$

Summing up Equations 3.56 and 3.57 gives the total window length WdL_T as

$$WdL_T = \left[WdL_P\right] + \left[WdL_n\right]$$

$$WdL_T = \left[\frac{nP_{(+E)}}{\prod_{i=1}^{+E-1} q_{(i)}}\left[1 + \sum_{k=1}^{+E-1}\left(\prod_{j=1}^{+E-1} q_{(j)}\right)\right] + n_{+E}\right] + \left[\frac{nP_{(-E)}}{\prod_{i=1}^{-E+1} q_{(i)}}\left[1 + \sum_{k=1}^{-E+1}\left(\prod_{j=1}^{-E+1} q_{(j)}\right)\right] + n_{-E}\right] \tag{3.58}$$

Case 2

If the least number of reference data occurs in any of the penultimate states, then use the algorithm in Figure 3.8 to place the number of samples "n" that minimizes the rate of Type-I and Type-II error in the penultimate state. Thus for the positive side of the chain say, set $n_{+E-1} = n$. Then re-arranging Equation (3.45), gives

$$n_{(+E)} = \frac{n_{+E-1}\left(1 - P_{+E-1}\right)}{P_{+E}} \tag{3.59}$$

Again, from Equation (3.39), $P(+i) = \frac{n_{(+i)} - n_{+i+1}}{n_{(+i)}}$, Hence, $P_{(+E-2)} = \frac{n_{(+E-2)} - n_{(+E-1)}}{n_{(+E-2)}}$,

Rearranging gives,

$$n_{(+E-2)} = \frac{n_{(+E-1)}}{\left(1 - P_{(+E-2)}\right)} \tag{3.60}$$

Similarly,

$$n_{(+E-3)} = \frac{n_{(+E-2)}}{\left(1 - P_{(+E-3)}\right)}$$

$$n_{(+E-3)} = \frac{n_{(+E-1)}}{\left(1 - P_{(+E-3)}\right)\left(1 - P_{(+E-2)}\right)} \tag{3.61}$$

For the state of +E-4,

$$n_{(+E-4)} = \frac{n_{(+E-1)}}{\left(1 - P_{(+E-4)}\right)\left(1 - P_{(+E-3)}\right)\left(1 - P_{(+E-2)}\right)} \tag{3.62}$$

Continuing in a similar fashion down to the state +3 say,

$$n_{(+3)} = \frac{n_{(+E-1)}}{\left(1 - P_{(+3)}\right)\left(1 - P_{(+4)}\right)\bullet\bullet\bullet\left(1 - P_{(+E-3)}\right)\left(1 - P_{(+E-2)}\right)} \tag{3.63}$$

For the state of +2

$$n_{(+2)} = \frac{n_{(+E-1)}}{\left(1 - P_{(+2)}\right)\left(1 - P_{(+3)}\right)\bullet\bullet\bullet\left(1 - P_{(+E-3)}\right)\left(1 - P_{(+E-2)}\right)} \tag{3.64}$$

And for the state of +1,

$$n_{(+1)} = \frac{n_{(+E-1)}}{\left(1 - P_{(+1)}\right)\left(1 - P_{(+2)}\right)\bullet\bullet\bullet\left(1 - P_{(+E-3)}\right)\left(1 - P_{(+E-2)}\right)} \tag{3.65}$$

Summing all the numbers in each state for the positive side of the chain,

$$\mathrm{WdL}^{+} = n_{+1} + n_{+2} + n_{+3} + \ldots n_{+E-1} + n_{+E} \tag{3.66}$$

$$\begin{aligned} WdL^{+} = {} & \frac{n_{(+E-1)}}{\left(1-P_{(+E-2)}\right)\left(1-P_{(+E-3)}\right)\ldots\left(1-P_{(+3)}\right)\left(1-P_{(+2)}\right)\left(1-P_{(+1)}\right)} + \\ & \frac{n_{(+E-1)}}{\left(1-P_{(+E-2)}\right)\left(1-P_{(+E-3)}\right)\ldots\left(1-P_{(+3)}\right)\left(1-P_{(+2)}\right)} + \\ & \frac{n_{(+E-1)}}{\left(1-P_{(+E-2)}\right)\left(1-P_{(+E-3)}\right)\ldots\left(1-P_{(+4)}\right)\left(1-P_{(+3)}\right)} + \bullet\bullet\bullet + \\ & n_{(+E-1)} + \frac{n_{(+E-1)}\left(1-P_{(+E-1)}\right)}{P_{+E}} \end{aligned} \tag{3.67}$$

$$\begin{aligned} WdL_P = {} & \frac{n_{(+E-1)}}{\left(1-P_{(+E-2)}\right)\left(1-P_{(+E-3)}\right)\ldots\left(1-P_{(+3)}\right)\left(1-P_{(+2)}\right)\left(1-P_{(+1)}\right)} * \\ & \left[\begin{array}{l} 1+\left(1-P_{+1}\right)+\left(1-P_{+1}\right)\left(1-P_{+2}\right)+\left(1-P_{+1}\right)\left(1-P_{+2}\right)\left(1-P_{+3}\right)+\bullet\bullet\bullet+ \\ \left(1-P_{+1}\right)\left(1-P_{+2}\right)\ldots\left(1-P_{(+E-5)}\right)\left(1-P_{(+E-4)}\right)+\left(1-P_{+1}\right)\left(1-P_{+2}\right)\ldots \\ \left(1-P_{(+E-4)}\right)\left(1-P_{(+E-3)}\right)+\left(1-P_{+1}\right)\left(1-P_{+2}\right)\ldots\left(1-P_{(+E-3)}\right)\left(1-P_{(+E-2)}\right) \end{array}\right] \\ & + \frac{n_{+E-1}\left(1-P_{(+E-1)}\right)}{P_{+E}} \end{aligned} \tag{3.68}$$

$$WdL_P = \frac{n_{(+E-1)}}{\prod_{i=1}^{+E-2} q_i}\left[1+\sum_{j=1}^{+E-2}\left(\prod_{k=1}^{+E-2} q_k\right)\right] + \frac{n_{+E-1}q_{(+E-1)}}{\left(1-q_{+E}\right)} \tag{3.69}$$

Equation (3.69) gives the number of samples that will make up the window length on the positive side of the chain provided the least number of reference samples occurred in the positive penultimate states. If on the negative side of the chain, the least number of reference samples occurs in the penultimate state, then it can also be shown that the window length on that side is given by

$$WdL_P = \frac{n_{(-E+1)}}{\prod_{i=1}^{-E+2} q_i}\left[1 + \sum_{j=1}^{-E+2}\left(\prod_{k=1}^{-E+2} q_k\right)\right] + \frac{n_{-E+1}q_{(-E+1)}}{(1-q_{-E})} \tag{3.70}$$

Again, summing up Equations (3.69) and (3.70) gives the total window length WdL_T as

$$WdL_T = [WdL_P] + [WdL_n]$$

$$WdL_T = \left[\frac{n_{(+E-1)}}{\prod_{i=1}^{+E-2} q_i}\left[1 + \sum_{j=1}^{+E-2}\left(\prod_{k=1}^{+E-2} q_k\right)\right] + \frac{n_{+E-1}q_{(+E-1)}}{(1-q_{+E})}\right] + \left[\frac{n_{(-E+1)}}{\prod_{i=1}^{-E+2} q_i}\left[1 + \sum_{j=1}^{-E+2}\left(\prod_{k=1}^{-E+2} q_k\right)\right] + \frac{n_{-E+1}q_{(-E+1)}}{(1-q_{-E})}\right] \tag{3.71}$$

In summary, after analysis of the reference data, the monitor is designed to identify the two states (one on each half of the chain) that contain the least number of reference data. After that, it uses the appropriate relation from among the four Equations, namely, Equations (3.56), (3.57), (3.69) and (3.70) to estimate the widow length of samples for each half of the chain and then sum the two up to estimate the desired window length for the entire monitor.

While the analysis above uses the information from the least populated states from each half of the chain, it is also possible to use information from the overall least populated state in the entire chain to obtain the number of samples that meets the desired Type-I (α) and Type-II (β) error rates for that state. Then, use that number of samples to obtain the number of samples that need to be in all other states. This approach however, leads to excessive number of samples in other states, making α and β for all other states more conservative than specified.

3.2.5 Estimate the Control Limits for Each State

Once the number samples in each state has been estimated, the upper and lower control limits for each state must be determined. At the start of the analysis, the user provides the overall Type-I error α_T. As was discussed in Section 3.2.3, the monitor uses this information together with the total number of states to estimate the Type-I error rate that will be associated with transitions to and from each state as per Equation (3.31). Having estimated the Type-I error rate for each state, the upper and lower control limits for each state are estimated as follows.

Let X, denote number of transitions within the control limits (CL) given the null hypothesis H_0. Also, let N_k = Total number of transitions into a state, then using the binomial Equation:

$$P\left(X_k \middle| N_k\right) = \frac{N_k!}{X_k!(N_k - X_k)!} P\left(T_k\right)^{X_k} \left(1 - P\left(T_k\right)\right)^{(N_k - X_k)} \tag{3.71}$$

Where "k" denotes a particular state and $0 \le X_k \le N_k$.

Figure 3.9 is a graphical representation of how the control limits are determined. The lower control limit for which $\frac{\alpha_k}{2} = \sum_{X_k=0}^{LCL_k} P\left(X_k \middle| N_k\right)$, is determined by finding the cumulative sum of $P\left(X_k \middle| N_k\right)$ until two cumulative density values denoted by C_N and C_O as shown in Figure 3.9 bound the lower control value $\frac{\alpha_k}{2}$. Once that is established, the lower control limit, LCL_k, for a particular state "k" is given by interpolation as:

$$LCL_k = \frac{X_k}{N_k} = \frac{X_O + \left[\frac{(\alpha_k/2 - C_O)}{(C_N - C_O)}\right]}{N_k} \tag{3.72}$$

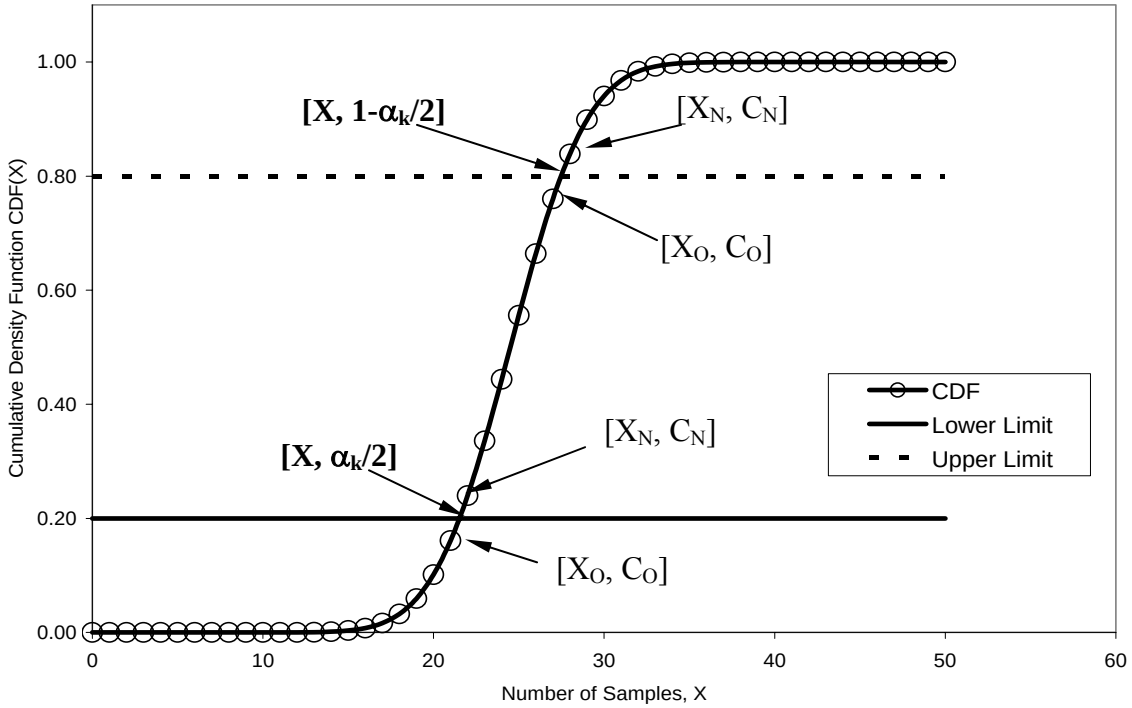


Figure 3.9 Binomial Cumulative Density Function for Calculating Control Limits

Similarly, to find the upper control limits for which $1-\alpha_k/2 = \sum_{X_k=0}^{UCL_k} P\left(X_k|N_k\right)$, the cumulative sum of $P\left(X_k|N_k\right)$ is determined until two cumulative values, denoted earlier by C_N and C_O again bracket the upper control value $1-\alpha_k/2$. Once that is established, the upper control limit for a particular is estimated by interpolation as:

$$UCL_k = \frac{X_k}{N_k} = \frac{X_O + \left[\frac{\left((1-\alpha_i/2) - C_O\right)}{\left(C_N - C_O\right)}\right]}{N_k} \tag{3.73}$$

3.2.6 Estimate Type-II Error Rate and Power for all States and the Entire Monitor

With control limits estimated for each state in the chain, the Type-II error rate that will occur for each state transition that is different from the reference transition probability can be easily estimated. Using Figure 3.5 for illustration, the Type-II error rate (the rate of missed alarms) associated with state is estimated using Equation (3.27) as

$$\beta_k = \sum_{x=0}^{x_H} \binom{n}{x} P_a^x \left(1-P_a\right)^{n-x} - \sum_{0}^{x_L} \binom{n}{x} P_a^x \left(1-P_a\right)^{n-x} \text{, where } P_a \neq P_o$$

However, it is important to know the total number of missed alarms (i.e. overall Type-II error rate) for the entire monitor. Recall the Null hypothesis H_0: $P = P_0$ and the alternative hypothesis H_a: $P \neq P_0$. Let β_T denote the overall Type-II error rate. Then,

Type-II Error Rate = P(Transition probabilities are different from the reference value but monitor does not flag)

But, Overall the Type-II Error Rate = β_T

Conversely, The overall Power, $P_W = 1- \beta_T$

P_W = P(Transitions are significantly different from reference value P_0, and monitor flags)

= 1-βT

Thus, P_W = P(H_a is true and monitor flags)

P_W = P(P_{-E} is different from P_0 and monitor flags and P_{-E+1} is different from P_0 and Monitor flags … and P_{-1} is different from P_0 and monitor flags and P_{+1} is Different from P_0 and monitor flags and … and P_{+E-1} is different from P_0 and Monitor flags and P_{+E} is different from P_0 and monitor flags)

P_W = P(P_{-E} is different from P_0 and monitor flags)*P(P_{-E+1} is different from P_0 and

Monitor flags)*…* P(P_{-1} is different from P_0 and monitor flags)*P(P_{+1} is Different from P_0 and monitor flags)* …*P(P_{+E-1} is different from P_0 and Monitor flags)*P(P_{+E} is different from P_0 and monitor flags)

P_W = [1-P(P_{-E} is different from P_0 and monitor does not flag)]*[1-P(P_{-E+1} is different from P_0 and monitor does not flag)]*…* [1-P(P_{-1} is different from P_0 and monitor does not flag)]* [1-P(P_{+1} is different from P_0 and monitor does not flag)]*…*[1-P(P_{+E-1} is different from P_0 and monitor does not flag)*[1-P(P_{+E} is different from P_0 and monitor does not flag)]

$$P_w = (1-\beta_{-E})(1-\beta_{-E+1})*...*(1-\beta_{-1})(1-\beta_{+1})*...*(1-\beta_{+E-1})(1-\beta_{+E}) \quad (3.74)$$

$$1-\beta_T = (1-\beta_{-E})(1-\beta_{-E+1})*...*(1-\beta_{-1})(1-\beta_{+1})*...*(1-\beta_{+E-1})(1-\beta_{+E}) \quad (3.75)$$

Hence,

$$\beta_T = 1-\prod_{k=1}^{L}(1-\beta_k) \quad (3.76)$$

Equation (3.76) can be used to estimate the overall Type-II error rate and hence the overall power for the entire test.

$$P_W = 1-\beta_T \quad (3.77)$$

If the power of the test (P_{wk}) associated with each state transition is desired, it can also be estimated as $P_{wk} = 1-\beta_k$ where β_k is known from Equation (3.27).

In using Equation (3.76) to estimate the Type-II error rate, it must be mentioned that future transition probabilities (P_a) associated with each state may differ from their reference values (P_o) by different amounts at particular instances. Consequently, there is no simple concept as the Type-II error rate for an entire test because it depends on the deviation. For this work, a limiting assumption is made where the transition probabilities

P_a associated with each state are assumed to differ from the reference value P_0 associated with that state by equal amounts at some instance. Although this is an ideal assumption, it is not a limitation in any way and it provides a good basis for one to assess the rate at which alarms will be missed if future transition probabilities were to differ by the fractional amounts λ indicated in Table 3.3.

$$P_a = \begin{cases} \lambda P_0 & P < P_0 \\ \mathrm{P}_0 + \lambda\left(1 - P_0\right) & P > P_0 \end{cases} \tag{3.78}$$

Table 3.3 List of Type-II Error Rates

			λ										
State	n_s	P_0	0.1	0.3	0.5	0.7	0.8	0.9	0.95	0.97	0.98	0.99	1
-4	25	0.47241	0.99564	0.94335	0.67987	0.1933	0.041962	0.001599	3.87E-05	2.15E-06	2.06E-07	3.50E-09	0
-3	28	0.47516	0.9955	0.93497	0.63027	0.13972	0.022837	0.000488	6.24E-06	2.12E-07	1.37E-08	1.18E-10	0
-2	55	0.50789	0.9928	0.83581	0.26374	0.004814	5.37E-05	5.57E-09	2.16E-13	8.78E-17	1.61E-19	2.94E-24	0
-1	108	0.50924	0.98579	0.54433	0.014871	3.34E-07	4.64E-12	1.28E-21	5.40E-32	6.67E-40	2.87E-46	2.94E-57	0
1	102	0.50804	0.99063	0.6289	0.029438	2.11E-06	8.36E-11	1.58E-19	4.96E-29	2.75E-36	3.94E-42	3.18E-52	0
2	52	0.48418	0.99261	0.8275	0.24949	0.004191	4.44E-05	4.37E-09	1.64E-13	6.61E-17	1.20E-19	2.18E-24	0
3	25	0.52614	0.9988	0.98448	0.87652	0.484	0.208	0.028737	0.002676	0.000407	8.70E-05	5.89E-06	0
4	27	0.50613	0.99538	0.94442	0.68824	0.20155	0.044711	0.001747	4.29E-05	2.40E-06	2.30E-07	3.93E-09	0
$\beta_T = 1-\prod_{k=1}^{L}(1-\beta_k)$			1	1	0.99759	0.71665	0.29178	0.032456	0.002764	0.000412	8.75E-05	5.90E-06	0
Power = $1-\beta_T$			0	1.52E-08	0.002407	0.28335	0.70822	0.96754	0.99724	0.99959	0.99991	0.99999	1

In Table 3.3, the values of λ, the fractional amounts by which future transition probabilities P_a, differ from the reference value P_0 are given in row 2 columns 4 to 14. The values in the Table are from 0.1 to 1.0. The values in rows 3 to 10 of column 1 are the states, rows 3 to 10 of column 2 are the number of samples in the corresponding state in column 1 and column 3 rows 3 to 10 are the reference transition probabilities associated with the state. The data in rows 3 to 10 and columns 4 to 14 represent the Type-II error rate β for each state. For instance, in row 4 column 4, the value of 0.9955 indicates that for the state of –3, if during test analysis transition probabilities differ from the reference by 10% (0.10), then there is a 99.55% chance that it will not be detected. Similarly, if for that same state, transition probabilities differ from the reference by 90% (0.9), then, there is a 0.0488% chance that it will not be detected Thus as the extent of deviation from the reference probability increase, the chances of failing to detect that the values are significantly different from the reference value decreases. In other words, the ability to detect that a data is significantly different from the reference value P_0 increases as the deviation of P_a from P_0 increases. The same explanation holds for all the values in that range. In row 11, columns 4 to 14, the values represent the composite or cumulative Type-II error rate for all the states while the values in the last row and columns 4 to 14 represent the entire power. On the last row in column 10, the value of 99.724% (0.99724) means that if future transition probabilities associated with all the states were deviated from the reference values by 95% (λ=0.95), then there is 99.724% chance that transition probability values will be detected as belonging to a different distribution and the likelihood of not missing an alarm is high.

In summary, before using the monitor for test analysis, user must define a good control period and collect data for some duration. User must also provide the Type-I error rate α, the Type-II error rate β, and deviation (λ) from the reference transition probability (P_0) at which β is desired. The initialization of the monitor uses the data from the good period, an initial sampling ratio (SR) and initial number of states (N_s) both of which can be adjusted to meet two criteria:

1. That, all transition probability lie between 0.25 and 0.75 (for this work) and
2. That no more than 10% of the data lie in the extreme states

The initialization then uses α, β, λ, to determine a window length and then an upper and lower control limit on each reference transition probability. The flow chart in Figure 3.10 illustrates the procedure described above.

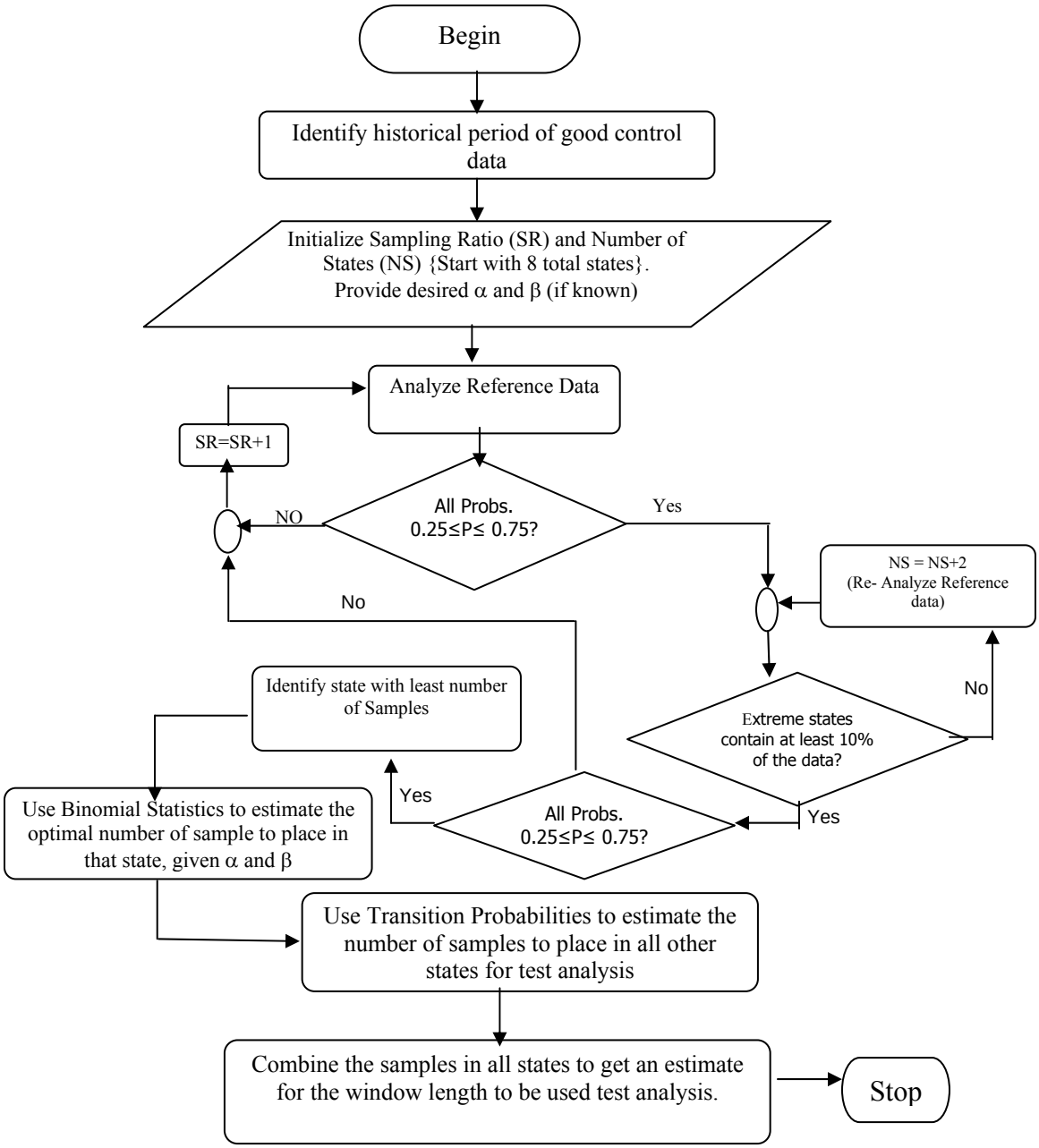


Figure 3.10 Flow Chart for Determining Ideal Window length

3.3 Moving Window Statistics

Now, in the moving window of data, the total count of samplings forming the window that was used in calculating the control limits must be maintained. However, the indexing (where index refers to the various cells containing state data values) and calculation of transition probabilities for the entire window at each sampling is a computational burden. To minimize this, the array in Table 3.4 and the "clock" data structure illustrated in Figure 3.11 with a pointer are used. State measurements are stored in an array as in Table 3.4.

Table 3.4 Array of State Measurements and Their Indexes

Index	1	2	3	4	5	6	7	8	9	10	.	.	.	N-2	N-1	N
State	+1	-1	-2	+1	-1	-2	1+	+2	3	+4	.	.	.	-1	+1	+2

During test analysis, data is indexed and stored in the array as shown in Table 3.4. As the data is been stored in the array, a record of the cumulative count of each state data is kept in another array, to be discussed shortly. Once the window length of data (N) is collected, then at each new sampling, the oldest data in the window must be replaced with new data. There must be N samples in the window at each sampling. Before replacing the oldest data however, its state value is read. Then decrease the cumulative state (count) of this oldest data by one. Next, determine the state of the current data and store its state value at the current location (which is same as the location where the oldest

data is to be replaced). Increment the cumulative state (count) of this new state by one. At the next sampling, repeat the entire process

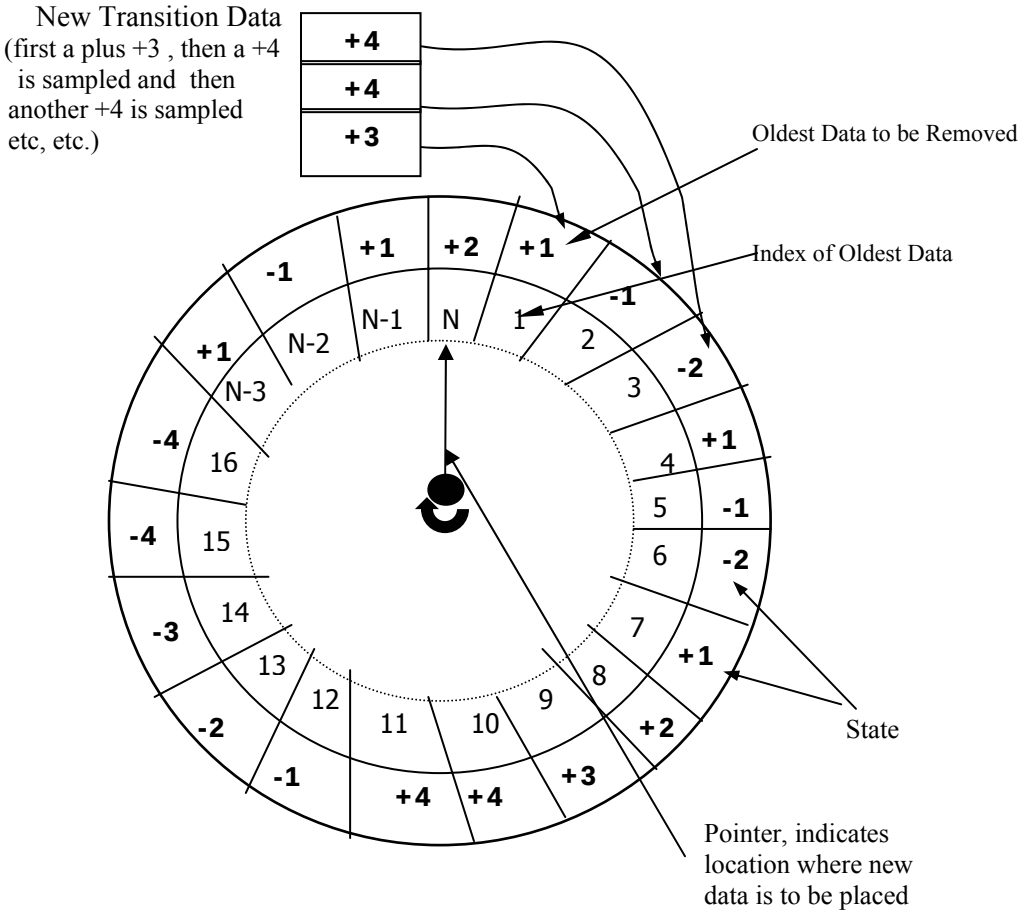


Figure 3.11 Statistics in a Moving Window

Looked at from another perspective, the array can be viewed as the face of a clock if the two ends of Table 3.4 are swung together. In Figure 3.11, the pointer indicates the current location where the newest data that marks the end of the window length is to be placed in the array. In practice, data is stored as the pointer moves clockwise until it gets to the end of the window (estimated previously). Before the next new data is stored after

the window length is exceeded, the pointer is indexed to first read the state value of the oldest data (in this example, the oldest data now just after the Nth data was sampled is in location 1), and decrease the cumulative count (i.e. number of samples) of this oldest data by one. Then determine the state of the current sampled data, store the state value at this location and increment the cumulative count of this new state by one. An example of the cumulative count data is shown in Table 3.5.

Table 3.5 Array of Cumulative State Measurements

State ->	-4	-3	-2	-1	+1	+2	+3	+4
Cumulative Count (just after Sampling a Window Length of data)	71	123	229	455	454	257	140	84
Updated Cumulative Count (After Sampling +3 data)	71	123	229	455	**453**	257	**141**	84
Updated Cumulative Count (After Sampling +4 data)	71	123	229	**454**	453	257	141	**85**
Updated Cumulative Count (After Sampling another +4 data)	71	123	**228**	454	453	257	141	**86**

$$p(+2) = \frac{Count(+2) - Count(+3)}{Count(+2)} = \frac{257 - 140}{257} = 0.4553 \tag{3.79}$$

New Transition probability $$p(+2) = \frac{257 - 141}{257} = 0.4514 \tag{3.80}$$

For instance in Figure 3.11, the monitor first checks the state of the data stored at the very beginning of the window, which is the oldest data at the beginning of the window (in the illustrated case, +1). Therefore, the monitor removes one data from the cumulative count of +1 state. If the new state is +3 say, then the monitor now replaces the old +1 data with the new data +3 data and adds one to the cumulative count of +3 state. At the next transition, the oldest state at the beginning of the window is the –1 data shown in the clock with index location 2. Therefore, the monitor subtracts one from the

cumulative count of -1. The new transition could be a state of +4 or a zero crossing to a state of –1. Assuming it is +4, this new transition data will replace the oldest data in the location indexed as 2, and one added to the cumulative count of +4 state. The process then continues in a similar fashion during monitoring. In this way, a window of fixed length is always used for statistical comparison with the control limits at each sampling.

As indicated earlier, Table 3.5 shows the cumulative count data of all states in the moving window. The first Row shows states in the window. The second to fifth Rows show the cumulative count (i.e. number of samples) in each state. From the data shown in Row 2 of Table 3.5, the transition probability of a sample in occupying the state of +3 given that it was in the state of +2 previously is given by Equation (3.79). After the window length is exceeded, the oldest data is identified and the cumulative count of that state decreased by one and then the oldest data is replaced with the new data. For the example under discussion, the cumulative count of the +1 state is decreased by one from 454 to 453. The new data is +3, hence the cumulative count of the +3 is increased from 140 to 141. For illustration, this updated data is shown in the third row of Table 3.5.

Once the data in the array is updated, the new transition probability of a sample in occupying the state of +3 given that it was in the state of +2 previously is given by Equation (3.80). Notice that, the +3 state is a penultimate state. Assuming that prior to sampling the new transition data shown in Figure 3.11, 20 samples had left the penultimate state and visited the extreme state of +4. Therefore, the probability of making a zero crossing given that the token is now occupying the state of +3 will be:

$$p(+3) = \frac{Count(+3) - Count(+4 \,|\, +3 \; prior)}{Count(+3)} = \frac{140 - 20}{140} = 0.8571 \tag{3.81}$$

And for the +4 (i.e. extreme) state:

$$p(+4) = \frac{Count(+4\ to -1)}{Count(+4)} = \frac{Count(+4|+3\ prior)}{Count(+4)} = \frac{20}{84} = 0.2381 \tag{3.82}$$

After sampling the new data (i.e. the +3 data) and updating the cumulative count, the new transition probability for making a zero crossing from a +3 state will be:

$$p(+3) = \frac{141-20}{141} = 0.8582, \tag{3.83}$$

And for the +4 state, $p(+4) = \frac{20}{84} = 0.2381$ (3.84)

At the next sampling, the entire process is repeated. Suppose that the new sampled data after the +3 data in Figure 3.11 is a +4 data. Now the number of samples that have visited the state of +4 just after visiting the +3 state has increased by 1 to 21. The oldest data in the window now is a -1 data as shown in Table 3.4. First, decrease the cumulative count of the –1 data by one, and then replace -1 with the new +4 data and then increase the cumulative count of the +4 data by one as shown in Row 4 of Table 3.5. Update all transition probabilities. For instance,

$$p(+3) = \frac{Count(+3) - Count(+4|+3\ prior)}{Count(+4)} = \frac{141-21}{141} = 0.8511 \tag{3.85}$$

and $p(+4) = \frac{21}{85} = 0.2471$ (3.86)

Furthermore, assume that the next sampled data is again a +4 state (i.e. token revisits +4 state, starting in +4 state). Now the number of samples in the +4 state has increased by one to 86. The oldest data now is –2 state data (See Table 3.4). Decrease the cumulative count of the –2 state by 1, replace the -2 data with the new +4 data and increase the cumulative count of the new state (+4) data by 1 as shown in Row 5 of Table 3.5. Update all transition probabilities. For the state of +4 say, this will be given by:

$$p(+4) = \frac{21}{86} = 0.2442 \tag{3.87}$$

If during operation, the controller is adjusted in anyway, then it may be appropriate for the operator to collect new reference data and begin the analysis afresh.

3.4 Grace period

During monitoring, in the moving window of data, comparisons are made at each transition, between the probabilities and the control limits. If the transition probabilities lie outside the control limits, it indicates the new data has a significantly different behavior. This could be the result of a setpoint change or a disturbance. The controller needs time to adjust to any such disturbance. Hence, a grace period equal to the closed loop settling time (i.e. time from setpoint change to the time that the process variable response has settled within a certain percentage band of the final setpoint value, (usually 2 to 5%, ISA, 1998) plus the window length (CLST + WL) is allowed during which a violation counter is invoked. It is worth mentioning that the CLST is not a fixed a value. Every loop once tuned has it own settling time. The user must therefore provide this value in determining the allowable grace period. If after the grace period is exceeded, the controller has still not been able to adjust to the disturbance, then the monitor should raise a flag. If the controller is able to adjust to the disturbance within the grace period, the violation counter is reset to zero.

3.5 Conducting Test Analysis

Once the reference data is analyzed and a window length (WL) has been estimated, the monitor can now be activated to perform test analysis.

During test analysis, sample the test data, and at each sampling, calculate the transition probability. Continue sampling until the sample size is equivalent to the window length estimated. Once the sample size is equivalent to the window length, compare transition probabilities with the control limits. If any control limit is violated, the monitor initializes a violation counter. If the violation persists, then the counter is increased by one until it exceeds the grace period by one sample. If that happens, then a flag is raised retroactively (by CLST + WL samples) to indicate the point of first recognition of a problem in the loop that the controller is unable to resolve. Otherwise, if the transition probabilities fall within the control limit, then the violation counter is reset to zero and the analysis continues. The algorithm is illustrated in Figure 3.12

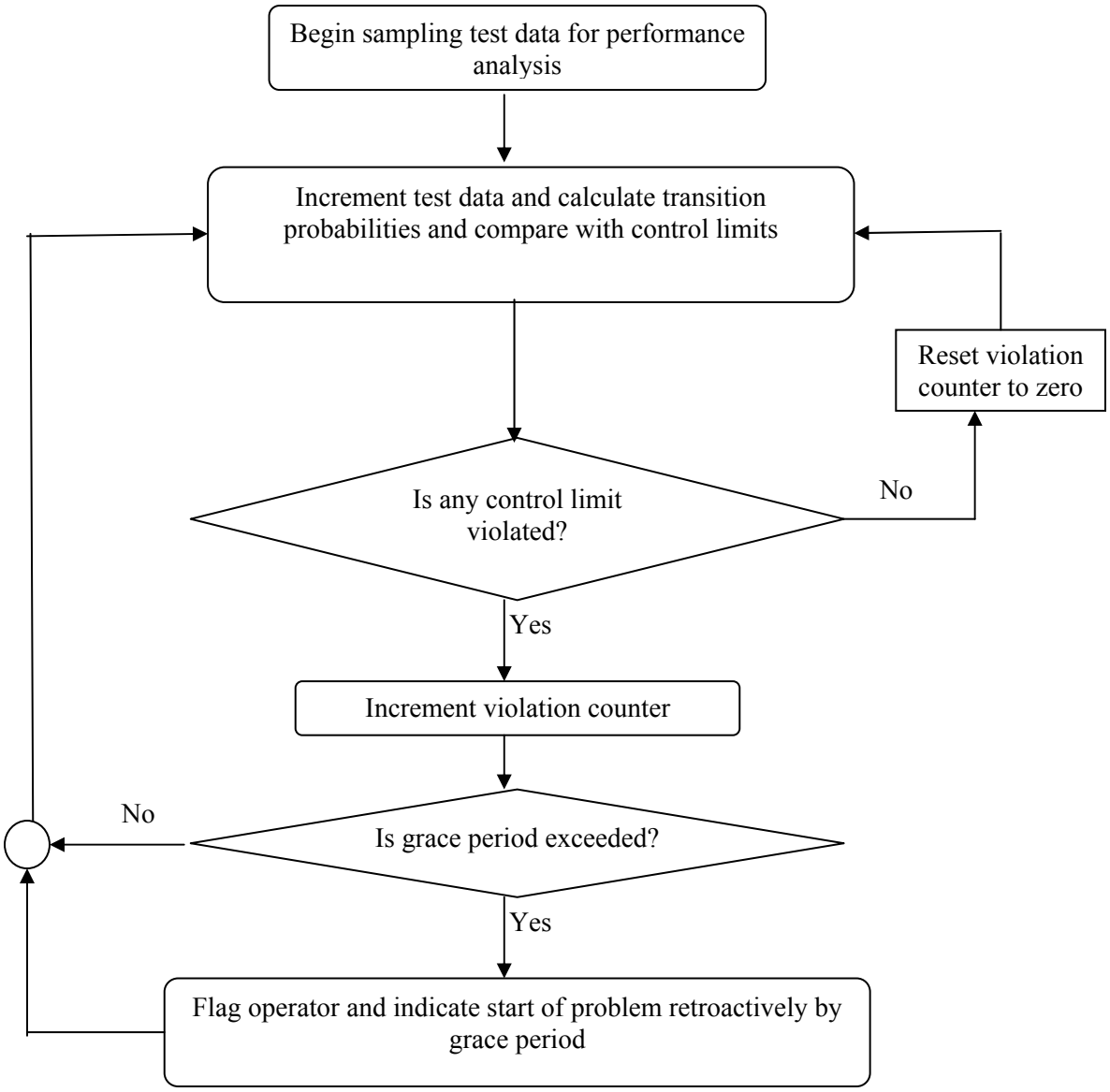


Figure 3.12 Flow Chart for Test Analysis

CHAPTER 4

4.0 EVALUATION OF THE HEALTH MONITOR

In this section the performance output of the health monitor on computer simulations, unit operation experimental data as well as application on industrial data are discussed.

4.1 Implementation Procedure

The procedure followed in implementing the health monitor is as follows:

1. Tune the controller at some desired operating point.
2. After tuning, observe the closed loop settling time from step changes in the setpoint.
3. When control is identified to be desirably good, initialize the health monitor to collect data if running online or store data elsewhere for offline use later. Collect good data over a period of time.
4. Provide initial values for the sampling ratio, number of states, Type-I error rate, the desired Type-II error for state with least number of reference good data, and the difference from the reference transition probability at which this Type-II error rate is desired.
5. Analyze the data using the procedures summarized in Chapter 3 to obtain the window length and control limits.
6. Set grace period equal to the closed loop settling time plus the data window length.

7. Once this information available, begin sampling data for controller performance analysis.

4.2 Computer Simulation Evaluation

In this section, results of the performance output from computer simulations using the health monitor are discussed. Results of various control schemes ranging from PID, IMC and MPC are shown and discussed.

4.2.1 First-Order Plus Time Delay Process

The performance of the monitor was tested a first-order plus time delay (FOPTD) level control process. A schematic diagram of the process is shown in Figure 4.1. The transfer function for the process is $G_p = K_p e^{-\theta s} \big/ \left(t_p s + 1\right)$. The process is controlled with a PI controller, which is tuned using the ITAE controller tuning rules. The process parameters are given in Appendix A.

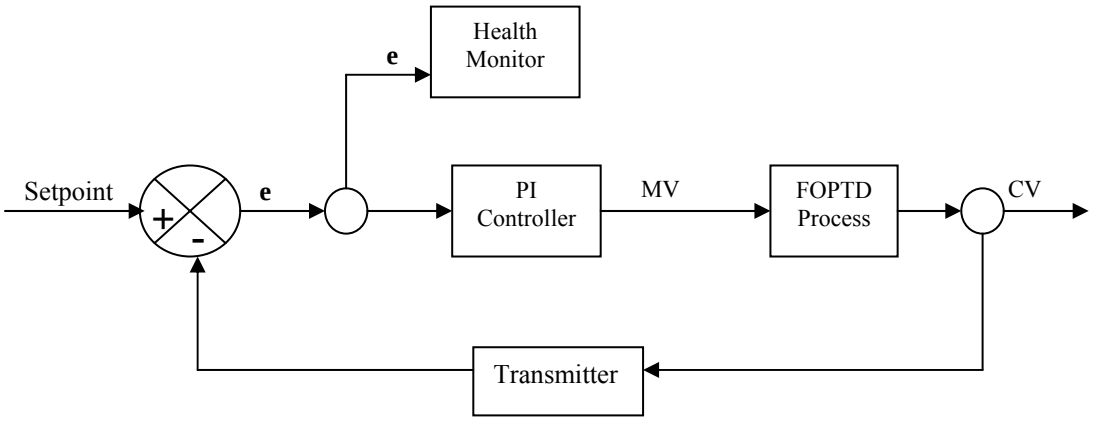


Figure 4.1 Schematic Diagram of a First-Order Plus Time Delay Process (e = Actuating Error, CV = Controlled variable, MV = Manipulated Variable)

Tuning was done using the ITAE controller tuning method (See Appendix A). After performing step changes, the closed loop settling time was estimated to be about 50 samplings. Data collected during a period of good control was analyzed and is shown in Figure 4.2. The sampling time interval for the controller was 0.25 units. Based on the data analyzed, the algorithm determined that a sampling ratio of 1 (i.e. health monitor sampling time interval = 0.25 units) and a window length of 526 samples weres ideal for test analysis. This implies that the grace period during test analysis will be (50 + 526) samples. This implies that during test analysis, it will take approximately, $\left(\frac{0.25 \text{ time units}}{\text{monitor sample}}\right) * (576 \text{ monitor samples})$ time units for the monitor to flag if a problem was detected in loop.

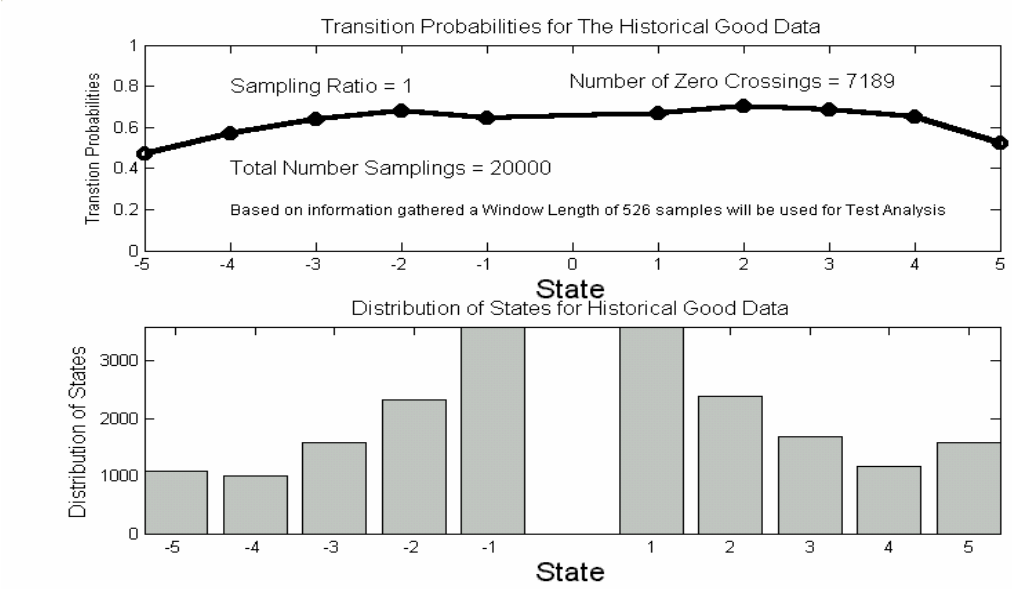


Figure 4.2 Distribution of States and Transition Probabilities from Reference Good Data (Window length = 526 samples; Sampling Ratio = 1)

If time units is in seconds say, then this means the monitor will flag in approximately 2.4 minutes after detecting degrading performance in the loop. The monitor was initialized with 8 total states (± 4 on each side). However, the algorithm determined that 10 (± 5 on each side) total states were needed in order to meet the requirement of having no more than 10 % of the data in the extreme states. Also, although the algorithm is developed to estimate the probability of making a zero crossing (i.e. $p_{\pm i}$), the first chart in Figure 4.2 reflects the probability of exiting from one state and visiting the next absolute higher state ($1-p_{\pm i}$).

After the analysis, and before estimating the window length, the algorithm selects the state with the least number of samples on each half of the Figure 4.2 and determines the number of samples to place in those states in order to meet the requirement on Type-I (α) and Type-II error (β). For this example, it can be observed from Figure 4.2 that the states that have the least number of samples are the -4 state on the negative side and the +4 state on the positive side. Only one of the two states (in this example -4 state) is chosen and discussed in Figure 4.3. The number of samples that need to visit this state was determined to be 27, given choices of α and β, and the how far way from the reference transition probability that this β is desired. Since this is a two-tailed test, there will be two confidence limits (i.e. a lower confidence limit and an upper confidence limit). The lower confidence limit is denoted by X_L, which indicates the minimum number of samples that need to leave the state and visit the next absolute higher state in order to reduce the Type-I error to the desired level. The upper confidence limit is denoted by X_H and indicates maximum number of samples that need to leave a state and visit the next higher state in order to reduce the Type-I error rate to the desired level. For

the illustrated situation in Figure 4.2, $X_L = 6$ and $X_H = 22$. Hence, in a given window (to be determined), out of a total of 27 samples that need to visit the state of -4 under ideal conditions, if fewer than 6 samples leave this state and visit the next absolute higher state (-5) or more than 22 samples leave this state and visit the next absolute higher, then a violation will occur.

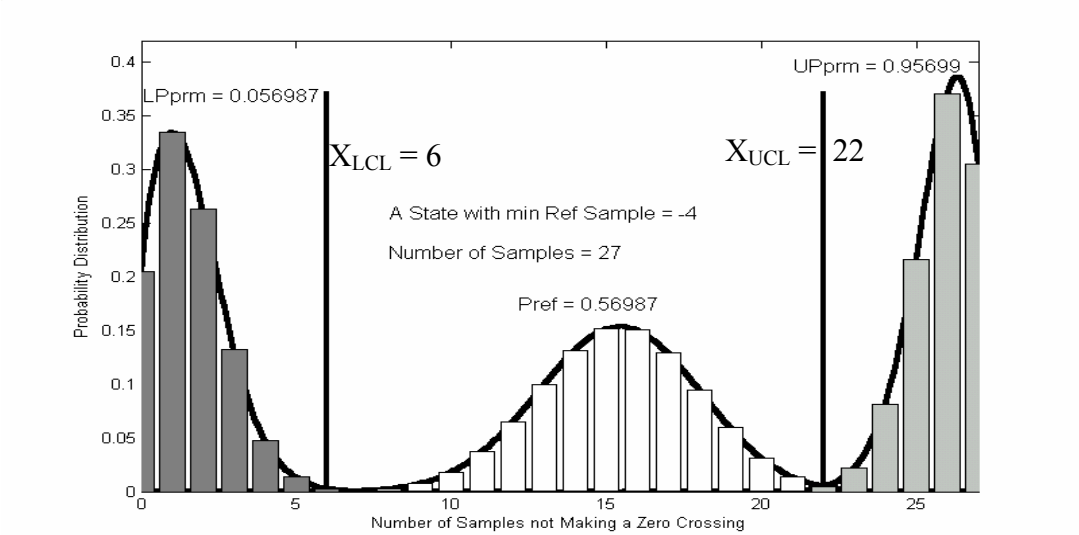


Figure 4.3 Analysis of Reference Data for State with Least Number of Samples. (Distribution of Samples Leaving State and Entering the Next Absolute Higher State Given the Reference Probability, α, β and λ)

Furthermore, Figure 4.3 illustrates that if future transition probabilities are the same as the reference value, then for example, 15% of the time, say, about 15 or 16 samples will leave the state of -4 and visit next absolute higher state. Also if future transition probabilities were to change significantly to say 0.95699 (i.e. $P_{ref} + \lambda(1-P_{ref})$), where for this state, $P_{ref} = 0.56987$ and $\lambda = 0.9$ in this example), then Figure 4.3 indicates that about 22% of the time, 25 samples will leave that state of -4 and visit the state of -5

and when it happens this will be outside the desired upper control limit of 22 and so a violation will occur. Similarly, if future transition probabilities were to change significantly to say 0.056987 (i.e. $P_{ref}(1 - \lambda)$, where for this state, $P_{ref} = 0.56987$ and $\lambda = 0.9$ in this example), then about 26% of the time only 3 samples will leave and visit the state of -5 and since this is outside the lower control limit of 6, a violation will occur. The chance of missing a violation as can be seen from the Figure 4.3 is negligibly small as desired.

After the analysis on this state is complete, it will be nice to know the probability distribution of the data in other states in order to observe how the Type-I and Type-II errors are minimized. As a result, for this work, after the numbers of samples in all other states are determined, the algorithm selects one state at random and determines the statistical errors that are associated with it based on the choices made. For this example, the algorithm selected the state of +3. The probability distribution for this state is shown in Figure 4.4. The explanation of Figure 4.4 is similar to that given for Figure 4.3. Notice that the algorithm estimated, based on the reference transition probability that, a total of 45 samples need to visit the state of +3.

In addition, if future transition where to differ from the reference value to say 0.068761 (i.e. $P_{ref}(1 - \lambda)$, where for this state, $P_{ref} = 0.68761$ and $\lambda = 0.9$), then 5 % of the time say, about 6 samples can be expected to leave the state of +3 and visit the state of +4. Furthermore, if probabilities differ from the reference value to about 0.96876 (i.e. $P_{ref} + \lambda(1-P_{ref})$), where for this state, $P_{ref} = 0.68761$ and $\lambda = 0.9$), then about 13% of the time 42 samples may be expected to leave the state of +3 and visit the state of +4. In both case, these number of visits are outside the control limits and so it will mean there is

an observed violation of the control limits. Once all samples that need to visit all the states during a period of good control are known, the algorithm estimates the window length necessary for performance monitoring.

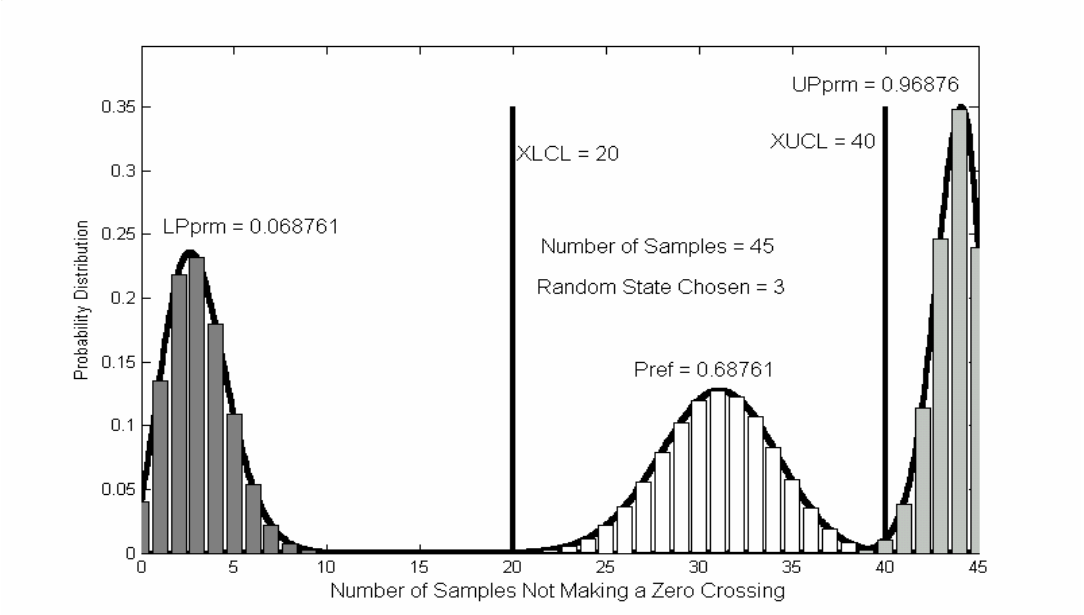


Figure 4.4 **Analysis of Reference Data for Randomly Selected State.** (Distribution of Samples Leaving State and Entering the Next Absolute Higher State Given the Reference Probability, α, β and λ)

Once the window length is estimated, a power curve for the test is plotted based on the assumption that at some instance, all state transition probabilities are equally deviated from their reference probabilities by some equal value ranging between 0 and 100%. This is shown in Figure 4.5 using a smooth graphical function option to mask break points at the discrete allowable abscissa values. It must be mentioned here that the smooth background curve in Figures 4.3 and 4.4 are not normal distribution curves. They are binomial distributions curves and were obtained using cubic spline interpolation between discrete points of the binomial distributions data and then plotting both a line

and a bar chart on the same graph. This was done for validation purposes and that if the bar chart was correct, the smooth fit should pass through about the center of each bar. Future plots were done same way.

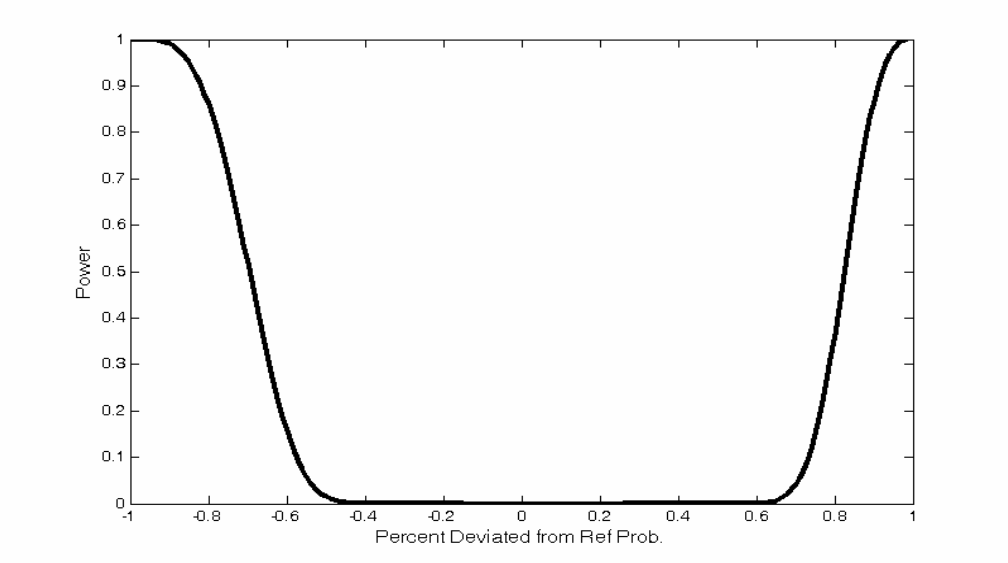


Figure 4.5 Power Curve (Given α = 1% and β= 1% λ = 90%)

As indicated earlier, this pre-monitoring information provides useful statistically expected outcomes that enable a user determine the power and reliability of the test. A high power value when future transitions probabilities are significantly different from the reference value is an indication that when the monitor flags signaling a violation, then it is indeed an indication that something is not right in the loop and that the distribution of new sampled data is different from the reference distribution.

4.2.2 Performance Monitor Demonstration

The performance of the monitor is illustrated in Figure 4.6, which has seven different plots. Starting at the top of the Figure 4.6, the first plot shows the controlled

variable and setpoint as a function of time. The second plot shows the actuating error as a function of time. The third shows the manipulated variable as a function of time. The fourth plot shows the violation counter and flag as a function of time. The last three plots labeled "A","B" and "C", are state transition probabilities and control limits as a function of state at particular points in time and illustrate how the transition probabilities compare with the control limits.

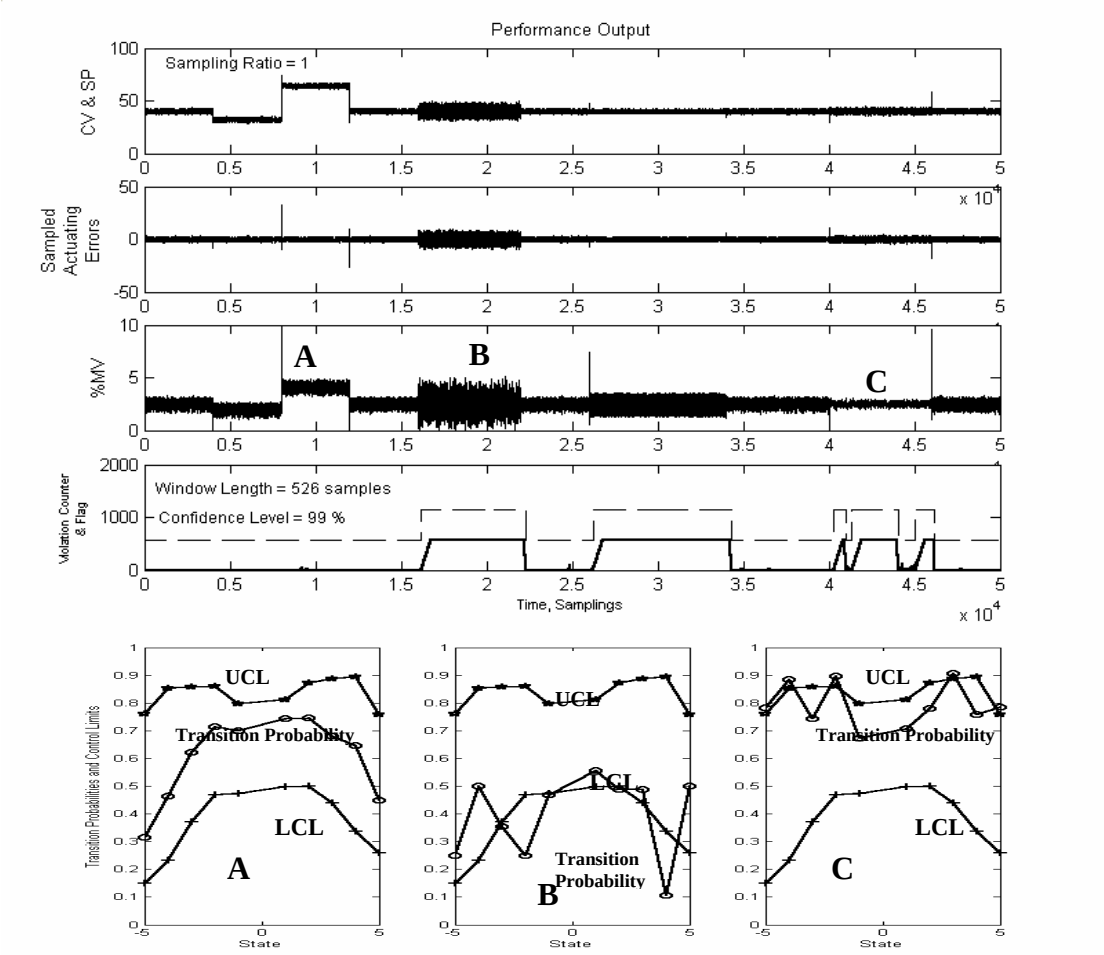


Figure 4.6 **Control Loop Performance Output** (Sampling Period = 0.25 Time Units, Sampling Ratio = 1; Window length = 526 Samples, Startup Period = 0 Samples, Grace Period 576 Samples, Violation Counter Trigger = Length of Grace Period +1, Overall Level of Significance (α_T) = 1%). Performance Output (A = No nuisances in the loop B = Stiction in Valve, C = Controller made sluggish by decreasing gain)

At sample numbers 4000, 8000 and 12000, setpoint changes were made. The controller was able to place the controlled variable at setpoint rapidly, so the run length distribution did not violate a control limit and the violation counter did not start counting.

Between sampling 16,000 and sampling 22000, a stiction effect was invoked in the control valve. The controller had difficulty keeping the controlled variable at setpoint and so the violation counter started counting once control limits were violated. After the grace period was exceeded and the violations were still present in the loop, the monitor raised a flag retroactively to indicate the start of the problem. Stiction was then removed and the flag turned off.

Between sampling 26,000 and 34,000, the controller was made too aggressive by increasing the controller gain Kc by a factor of 2. The monitor detected this and started the violation counter. After the grace period was reached and the controller had still not recovered, a flag was raised indicating something was seriously wrong either with the controller or within the control loop. At sampling 34,000, the controller gain was reset to its original value. The monitor immediately detected that; and, within about one window length of data sampling, the monitor stopped flagging.

Between sampling 40,000 and 46,000, the controller was made sluggish by changing Kc by a factor of 1/3. Again, the monitor detected that the controller was not performing well. Notice that between sampling 41,000 to about 42,500, the monitor initialized the violation counter a couple of times and reset it to zero. This is indicative of the fact that the controller was attempting to place the controlled variable at the desired

setpoint and to keep the loop in normal performance. The on-off counting might be due to the response of the monitor to the deteriorating performance in the controller.

However, once performance deteriorated completely, the monitor started the violation counter again and after the grace period was exceeded and the violation was still present, the monitor flagged continuously for the rest of the duration of sluggish control. The controller was restored to normal mode at sampling 46,000. When the monitor detected a return to good control the flagging stopped.

Other times when the counter started counting might also be due the fact that there is a 1% chance (α_T = 1%) of the null hypothesis H_0 being rejected when in fact it should not (Type-I error). But, in all such instances for instance around sampling 25,000, the flag was not raised because good control was recovered within the grace period, and the monitor reset the violation counter to zero.

The determination of values of X_L and X_H, the lower and upper control limits, are often done using normal statistics. Rigorously, the binomial statistics should be used. This investigation found the two to have minor distinguishable differences and while calculations with the normal statistical assumptions are more convenient, the more rigorous binomial statistics is used entirely in this work. A comparison analysis between using the exact binomial approach to estimate window length and using the normal approximation to the binomial relation (presented in Chapter 3) is discussed in Appendix F.

4.2.3 A Second-Order Plus Time Delay Process (SOPTD)

In this section, a discussion of the application of the health monitor on a SOPTD non-interacting process is presented. The SOPTD process is represented by the transfer function $G_p = K_p e^{-\theta s} / \left((t_{p1} s + 1)(t_{p2} s + 1) \right)$. A list of the process parameters is given in Appendix B. Figure 4.7 is a schematic illustration of a PID control loop with the health monitor in tandem with a controller and sampling the actuating errors.

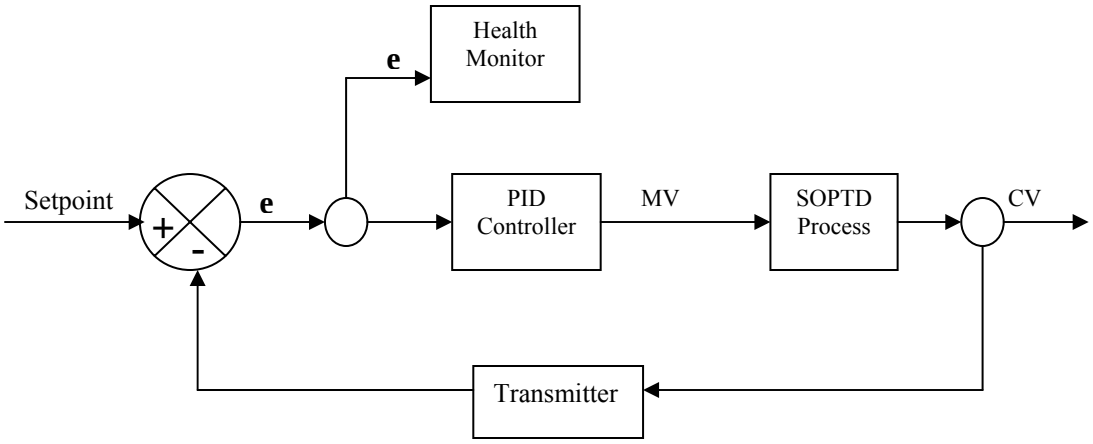


Figure 4.7 Schematic Diagram of a Second Order Plus Time Delay Process (e = Actuating Error, CV = Controlled variable, MV = Manipulated Variable)

After tuning (using the Cohen-Coon controller tuning technique) and performing step changes, the closed loop settling time was estimated to be about 50 samplings. Data collected during a period of good control was analyzed and is shown in Figure 4.8. The sampling time interval for the controller was 0.25 time units. Based on the data analyzed, the algorithm determined that a sampling ratio of 2 (i.e. health monitor sampling time interval = 0.5 time units) and a window length of 409 samples would be ideal for test

analysis. This implies that the grace period during test analysis will be 459 samples (50 + 409 = 459). Thus, during test analysis, it will take approximately,

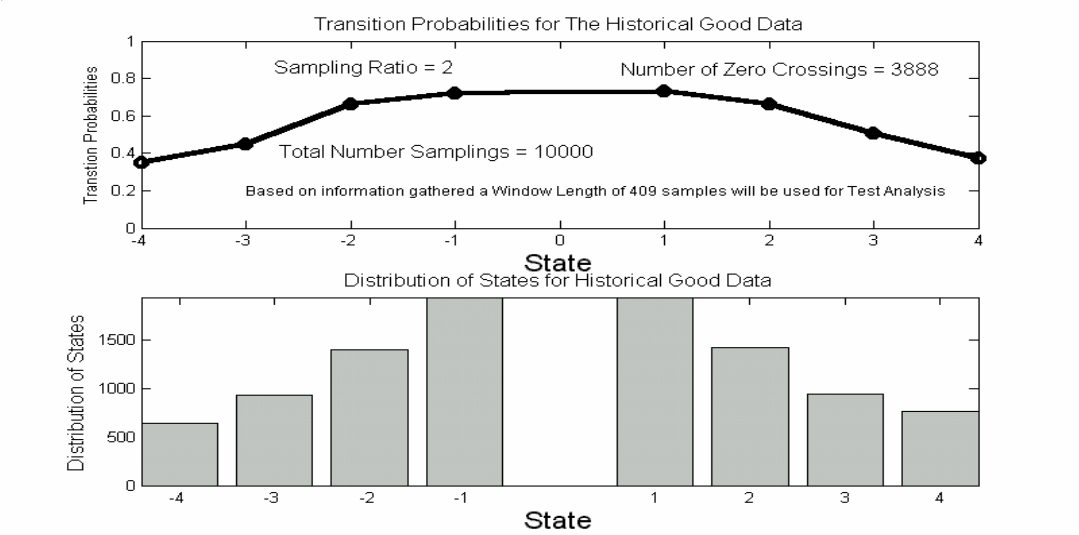


Figure 4.8 Distribution of States and Transition Probabilities from Reference Good Data (Window length = 409 Samples; Sampling Ratio = 2)

$\left(\frac{0.5 \text{ time units}}{\text{monitor sample}}\right) * (459 \text{ monitor samples})$ time units for the monitor to flag if a problem was detected in the loop. Assuming time units to be in seconds say, then this means the monitor will flag in approximately 4 minutes after detecting degrading performance in the loop. Violation counting, however, starts instantaneously.

In this example, the monitor was initialized with 6 total states (± 3 on each side, just to compare with the prior analysis in Chapter 3 where it was determined using random errors that it was adequate to initialize the monitor with 8 total states). However, the monitor determined that 8 (± 4 on each side) total states were needed in order to meet the requirement of having no more than 10 % of the data in the extreme states. Other

simulations performed, also revealed that 8 total or more states was always necessary in order to meet the user specified requirement in this work. After the analysis, and before estimating the window length, the monitor selects the state with least number of samples on each half of the Figure 4.8 and determines the number of samples to place in those states in order to meet the requirement on Type-I (α) and Type-II (β) error. For this example, it can be observed from Figure 4.8 that the states that have the least number of sample are the + 4 state on the positive side and the –4 state on the negative side. Only one of the two states (in this example -4 state) is chosen and Type-II error analysis revealed in Figure 4.9.

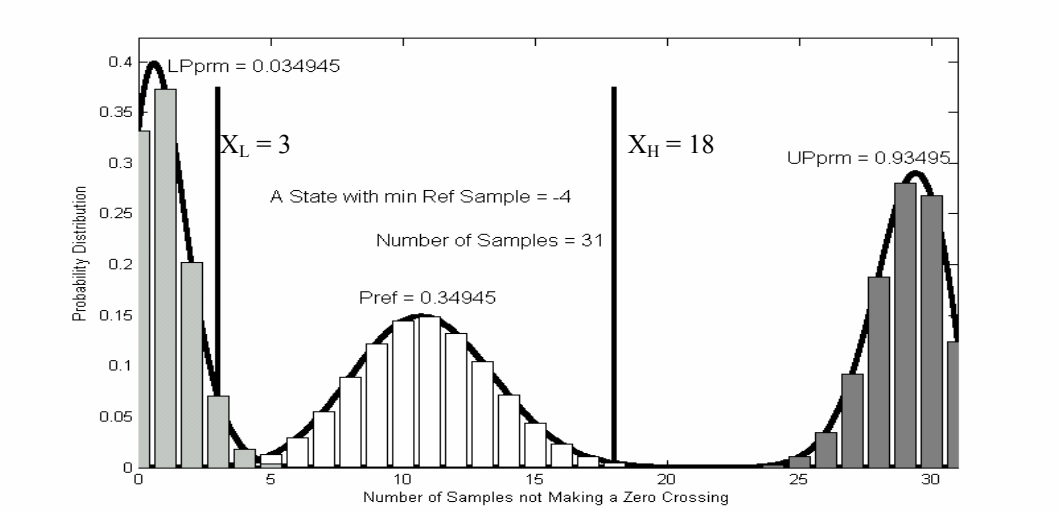


Figure 4.9 **Analysis of Reference Data for State with Least Number of Samples**. (Distribution of Samples Leaving State and Entering the Next Absolute Higher State Given the Reference Probability, α, β and λ)

The algorithm estimated the number of samples that need to visit this state (i.e. -4 state) to be 31 given choices of α and β and the how far way from the reference transition

probability (λ) that this β is desired. X_L indicates the minimum number of samples that need to leave a state and visit the next absolute higher in order to avoid a Type-I error while X_H indicates maximum number of samples that need to leave a state and visit the next absolute higher in order to avoid same.

For the illustrated case, $X_L = 3$ and $X_H = 18$. Thus, out of a total of 31 samples that need to visit the state of -4 in a given window (to be determined), a violation will occur if fewer than 3 samples leave the state and visit the next absolute higher state or more than 18 samples leave the state and visit the next absolute higher state. Moreover, Figure 4.9 shows that if future transition probabilities are not from the reference transition probability of 0.34945, then for example, 15% of the time say, about 11 samples will leave the state of -4 and visit the next absolute higher state. However, since the -4 state is an extreme state, it means about 15 % of the time 11 samples will leave the state of -4 and re-enter that state again. Also, if future transition probabilities were to change significantly to say 0.93495 (i.e. $P_{ref} + \lambda(1-P_{ref})$, where for this state, $P_{ref} = 0.34945$ and $\lambda = 0.9$ in this example), then it indicates that about 20% of the time, 28 samples will leave that state of -4 and revisit it again and when it happens this will be outside the desired upper control limit of 18 and so a violation will occur. Similarly, if future transition probabilities were to change significantly to say 0.034945 (i.e. $P_{ref}(1 - \lambda)$, where for this state, $P_{ref} = 0.34945$ and $\lambda = 0.9$), then 20% of the time only 2 samples may be expected to leave and revisit the state of -4 and since this is outside the lower control limit of 3, a violation will occur. The chance of missing a violation as can be seen from the Figure 4.9 is negligibly small as desired.

After the analysis on this state is complete and the number of samples in all the other states are determined, the algorithm selects one state at random and determines the statistical errors that are associated with it based on the choices made. For this example, the algorithm selected the state of +3 and the probability distribution is shown in Figure 4.10.

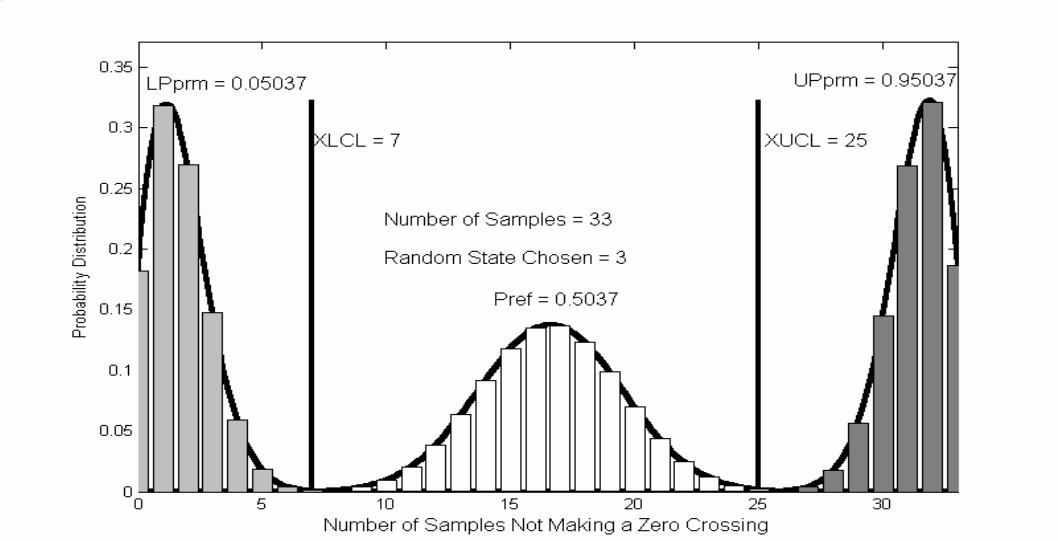


Figure 4.10 **Analysis of Reference Data for State Chosen at Random** (Distribution of Samples Leaving State and Entering the Next Absolute Higher State Given the Reference Probability, α, β and λ)

Again, the explanation of Figure 4.10 is similar to that given for Figure 4.9. The algorithm determined, based on the reference transition probability that, a total of 33 samples need to visit the state of +3. In addition, if future transition where to differ from the reference value to say 0.05037 (i.e. $P_{ref}(1 - \lambda)$, where for this state, $P_{ref} = 0.5037$ and $\lambda = 0.9$ in this example), then about 6% of the time say, about 4 samples may be expected to leave the state of +3 and visit the state of +4. Furthermore, if probabilities differ to about 0.95037 (i.e. $P_{ref} + \lambda(1-P_{ref})$, where for this state, $P_{ref} = 0.5037$ and $\lambda = 0.9$

in this example), then about 6% of the time 29 samples may be expected to leave that of +3 and visit the state of +4. In both case, these number of visits are outside the control limits and so it will mean a violation of the control limits. Once all samples that need to visit all the states during a period of good control are known, the monitor estimates the window length necessary for performance monitoring.

Once the window length is estimated, a power curve for the entire test is plotted based on the assumption that at some instance, all state transition probabilities are equally deviated from their reference probabilities by some equal value ranging between 0 and 100%. This is shown in Figure 4.11.

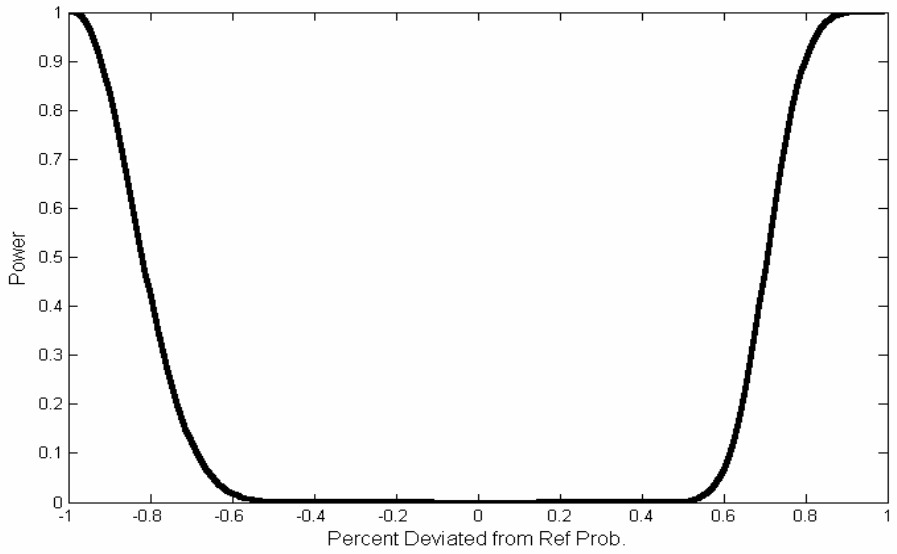


Figure 4.11 Power Curve (Given α = 1% and β= 1% λ = 90%)

4.2.4 Performance Monitor Demonstration

The performance of the monitor is illustrated in Figure 4.12 with seven plots. The first plot shows the controlled variable and setpoint as a function of time. The second plot shows the actuating error as a function of time.

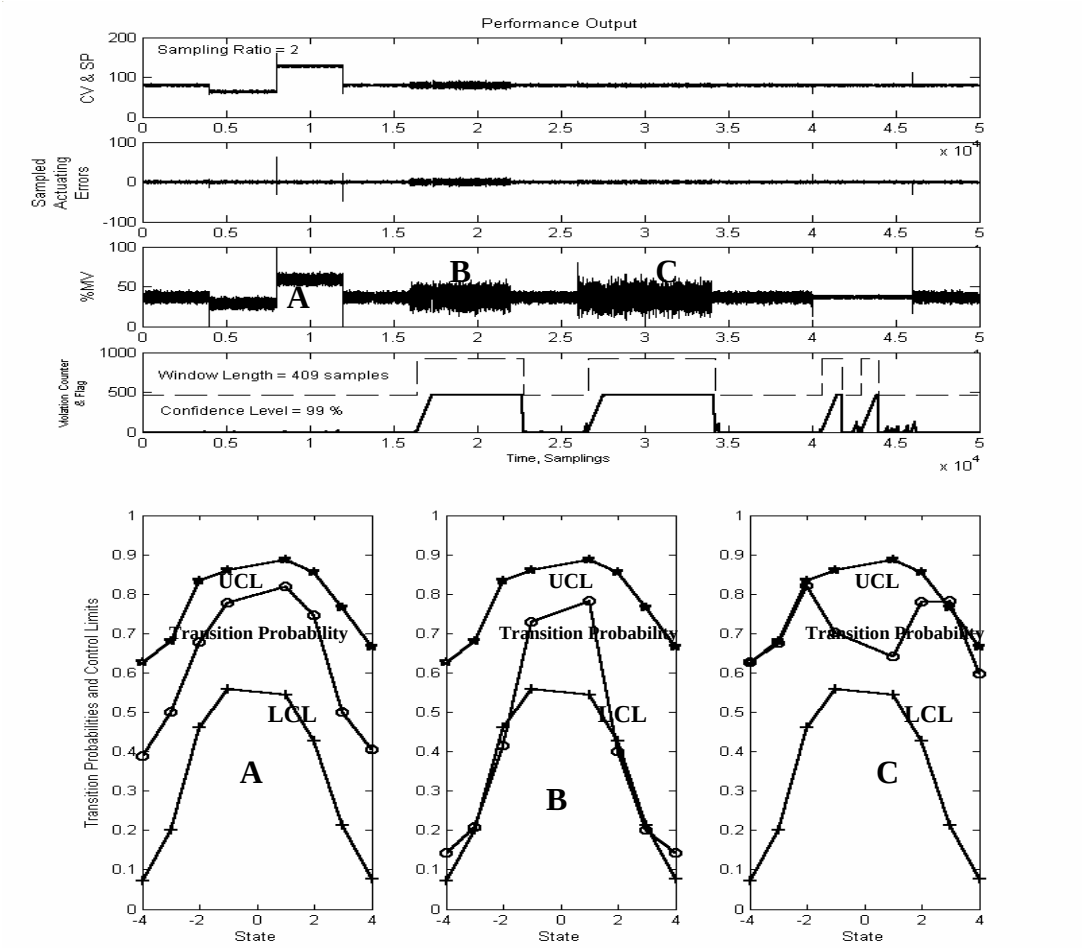


Figure 4.12 **Control Loop Performance Output** (Sampling Period = 0.25 Time Unit, Sampling Ratio = 2; Window length = 409 Samples, Startup Period = 0 Samples, Grace Period 459 Samples, Violation Counter Trigger = Length of Grace Period +1, Overall Level of Significance (α_T) = 1%). Performance Output (A = No nuisances in the loop B = Stiction in Control Valve, C = Controller made aggressive by increasing gain)

The third shows the manipulated variable as a function. The fourth plot shows the violation counter and flag as a function of time. The last three plots labeled "A","B" and "C", are state transition probabilities and control limits as a function of state at particular points in time and illustrate how the transition probabilities compare with the control limits.

Starting at the top of the page and referring to the first plot, at samples 4000, 8000 and 12000, setpoint changes were made. The controller was able to rapidly place the controlled variable at setpoint, so the run length distribution did not violate a control limit and the violation counter did not start counting as shown on the fourth plot.

Between sampling 16,000 and sampling 22000, a stiction effect was invoked in the control valve. The controller had difficulty placing the controlled variable at setpoint and so the violation counter started counting when control limits were first violated. After the grace period was exceeded and the violations were still present in the loop, the monitor raised a flag retroactively to indicate the start of the problem. The flag went off after the stiction was removed.

Between sampling 26,000 and 34,000, the controller was made too aggressive by increasing the controller gain Kc by a factor of 2. The monitor detected this and started the violation counter. After the grace period was reached and the controller had still not recovered from the aggressive condition, a flag was raised indicating something was wrong either with the controller or within the control loop. At sampling 34,000, the controller gain was reset to its original value. The monitor immediately detected that, and, within about one window length of data sampling, the monitor stopped flagging.

Between sampling 40,000 and 46,000, the controller was made sluggish by changing Kc by a factor of 1/3. Again, the monitor detected that the controller was not performing well. Notice that between sampling 41,000 to about 42,500, the monitor initialized the violation counter a couple of times and rest it to zero. This is indicative of the fact that the controller was attempting to place the controlled variable at the desired setpoint and to keep the loop in normal performance. However, once performance deteriorated completely, the monitor started the violation counter again and after the grace period was exceeded and the violation was still present, the monitor flagged continuously for the rest of the duration of sluggish control. The on-off behavior of the violation counter around sampling 45,000 might be due to the monitor's response to the deteriorating performance in the controller. The controller was restored to normal mode at sampling 46,000. When the monitor detected a return to good control the flagging stopped. Other times when the counter started counting might also be due the fact that there is a 1% chance (α_T = 1%) of the null hypothesis H_0 being rejected when in fact it should not (Type-I error). But, any such instance was short-lived and a flag was not raised because good control was achieved within the grace period, and the monitor reset the counter to zero.

4.3 Application on Unit Operations Experimental Data

4.3.1 Description of Experimental Unit

The experimental unit is equipped with three control valves in all. Two of the valves are airflow control valves (a large one and a small one). The third valve is a water flow control valve. The unit consists of a vertical column through which water and air flow upward, co-currently creating a two-phase flow. The unit is designed with individual PID-Type control loops to manipulate both water and the airflow control valves to sustain either flow rate or pressure drop. The unit is equipped with CamileTG 2000 software for remote process control and data acquisition. A schematic diagram of the experimental unit is shown in Figure 4.13.

The control loops in the two-phase flow unit can be configured in different schemes for studies. For instance, pressure transducers at the top and bottom of the column enable the pressure drop in the column to be determined and controlled by manipulating air or water flow rate. The different control schemes that were studied are discussed below.

The procedure used was; first to tune the controller using the process reaction technique. Once control was judged optimum after tuning, collect data at a desired sampling rate. The data-logging rate used for this work was 1sample/100 milliseconds.

After collecting good data, introduce disturbances in the loop such as setpoint changes, flow rate changes, controller gain changes, etc. Collect data from the experimental unit and use the health monitor to analyze the data with the view to testing the ability of the monitor in detecting durations when there were control problems.

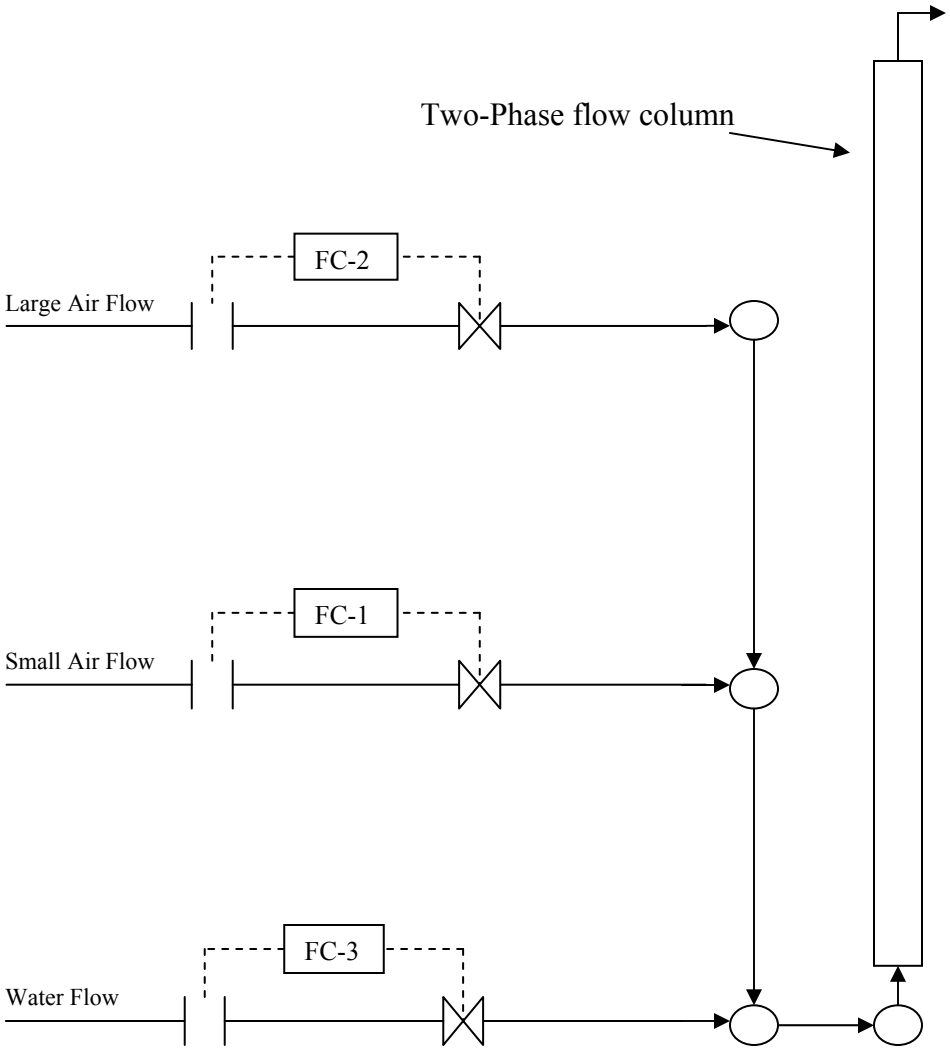


Figure 4.13 Two Phase Flow Experimental Unit with a Water Flow Control Loop and two Air flow Control Loops

4.3.2 Performance Monitoring for Pressure Drop Control by Manipulating Signal to Water Flow Control Valve

The first experimented control scheme studied involved controlling the pressure drop in the column by manipulating the water flow rate. Once the controller was tuned, and control found to be good, data was collected for a period. During this good period of control, the closed loop settling time was determined through setpoint changes to be about 50 samples. The good or reference data is analyzed as shown in Figure 4.14. After analysis of the reference data, the algorithm estimated that a sampling ratio of 2 would be ideal for test analysis.

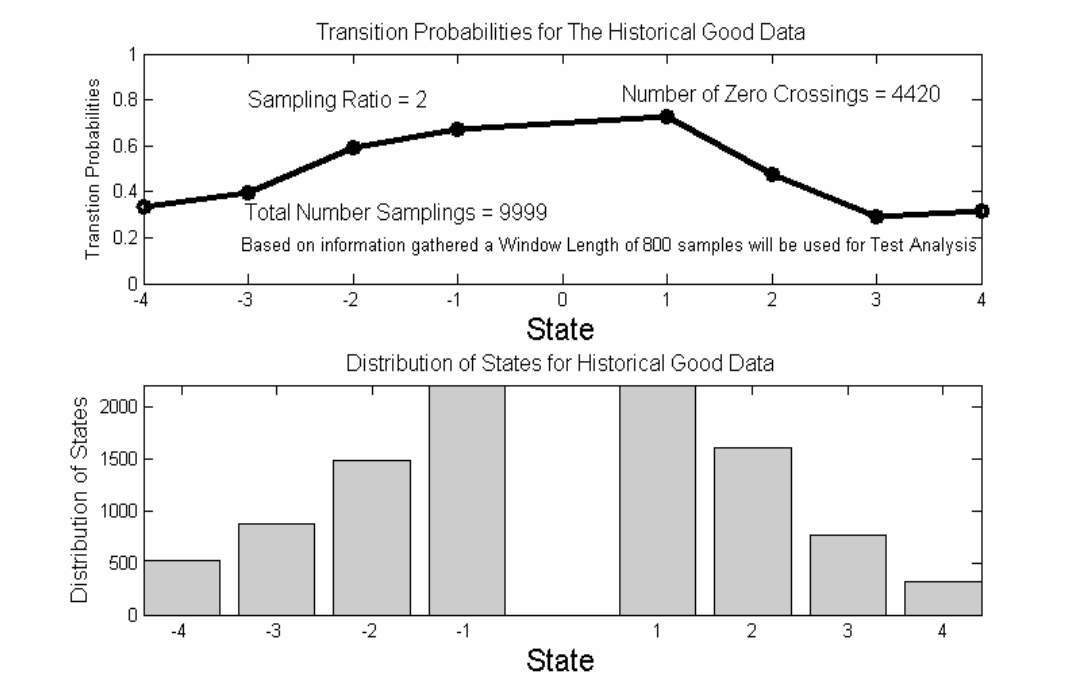


Figure 4.14 Distribution of States and Transition Probabilities from Reference Good Data (Window length = 800 samples; Sampling Ratio = 2, α_T = 1%, β = 1%, λ = 0.9)

Based on the logging rate of 0.1 seconds for the unit, it implies that the monitor will sample data from the unit every 0.2 seconds i.e. $\left(\frac{2 \text{ controller samples}}{1 \text{ Monitor sample}}\right)*\left(\frac{0.1 \text{ Seconds}}{1 \text{ controller sample}}\right)$. In addition, the window length for statistical comparisons was determined by the monitor to be 800 samples. This in effect means that at a sampling rate of 0.2 seconds, the monitor will require $\left(\frac{0.2 \text{ seconds}}{\text{monitor sample}}\right)*(800 \text{ monitor samples})$, or 160 Seconds (i.e. approx. 2.67 min) to sample one window length of data from the process. Moreover, during performance monitoring, if a problem occurred in the loop it will take about this much time plus the closed loop settling time before the monitor will flag. The confidence level $(1-\alpha)*100\%$ used was 99% and the Type-II error rate (β) used for the state having the least number of reference samples was set at 1% when future transition probabilities during testing differ by 0.9 (λ) from the reference value. Using information from analysis of the reference data, test data collected was analyzed as shown in Figure 4.15. The Figure shows 7 plots. The first plot shows the controlled variable and setpoint as a function of time. The second plot shows the actuating errors as a function of time. The third plot shows the manipulated variables as a function of time. The fourth plot shows the violation counter and flagging as a function of time. The last three plots show instances of the transition probabilities compared with the control limits as a function of state.

The point marked "A" shows a period when there were no control limit violations. Around sampling 14000, marked "B", no external disturbances were imposed on the loop but control limits were violated and the violation counter started counting. After the

grace period was exceeded and control limits were still been violated, the monitor raised a flag to indicate that there was a problem in the loop.

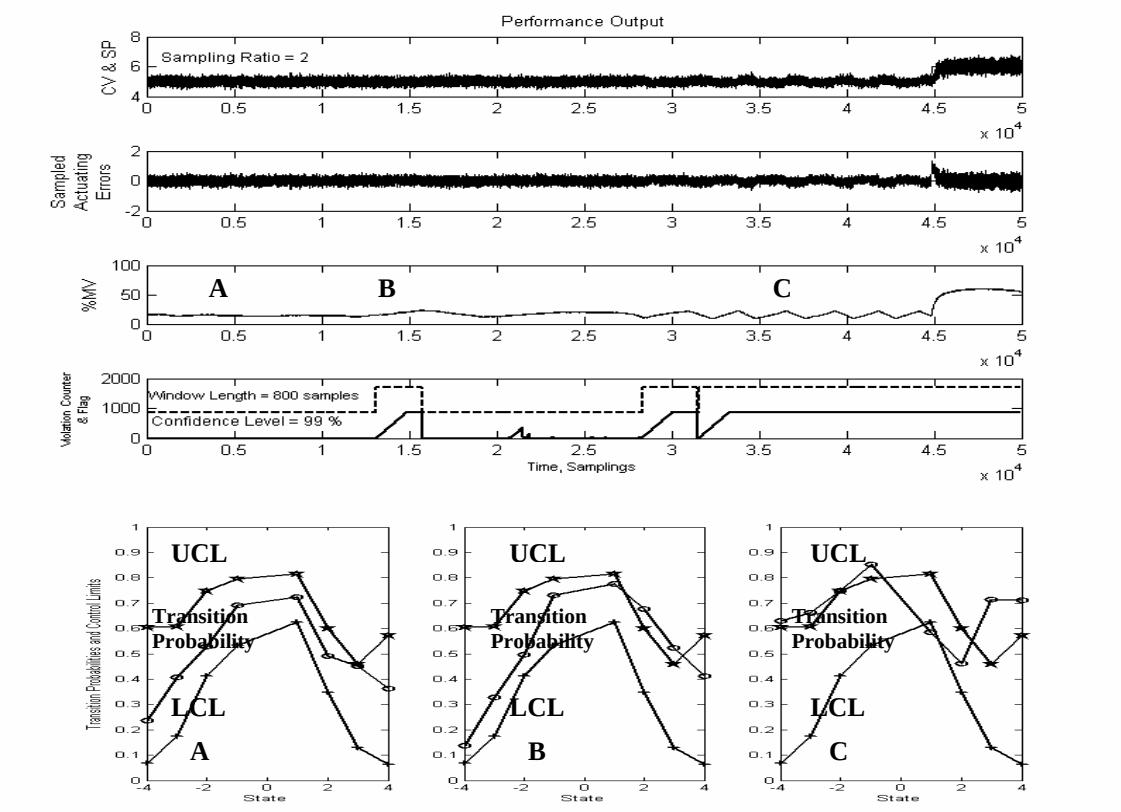


Figure 4.15 Control Loop Performance Output for Pressure Drop Control by Manipulating signal to Water Flow Control Valve (Sampling Period = 0.1s, Sampling Ratio = 2; Window length = 800 Samples, Startup Period = 0 Samples, Grace Period 850 Samples, Violation Counter Trigger = Length of Grace Period +1, Overall Level of Significance (α_T) = 1%). Performance Output (A = No nuisances in the loop B = Oscillations in Loop, C = Stiction in Loop)

Shortly there after, the oscillation that seemed to have caused the violations appeared to go off as depicted in the 3rd plot. When the monitor detected this, flagging stopped, and the violation counter was reset to zero as shown in the 4th plot. Around

sampling 21000, the violation counter started counting again but the violations detected were short lived and so the counter was reset to zero again. Then around sampling 29000 and in the region marked "C", the monitor detected violations again and so started counting again. After the grace period was exceeded and the violations were still present in the loop, the monitor raised a flag again. The flagging was sustained for as along as the violations were present. The saw tooth behavior in the manipulated variable plot (plot number 3) and the square wave behavior in the controlled variable plot (plot number 1) are typical signals to expect for a sticky valve. The monitors ability to detect a sticky valve and flag during the entire period of stiction was a good indication of its efficiency to detect degrading control in a loop.

4.3.3 Effect of Varying the Overall Type-I Error Rate on Window Length and Performance Monitoring

The overall Type-I error rate α_T, was varied and the Type-II error rate kept same in order to study how changing α_T affects the widow length and the overall performance of the monitor. The analysis was done offline on the same data as that used for Figure 4.15. The confidence level was set at 95% (i.e. $\alpha_T = 5\%$, the reference data was analyzed again and the result is shown is Figure 4.16. The algorithm estimated a window length of 618 samples but the sampling ratio SR, was same as previous (SR = 2).

The Type-II error rate used for the state having the least number of reference samples was kept unchanged at 1% when future transition probabilities during testing differ by 0.9 from the reference value. Notice that despite the fact that the monitor

estimated a smaller window length for testing, it detected all instance of loop disturbance and nuances within the regions marked “A” “B”, and “C” respectively in Figure 4.17.

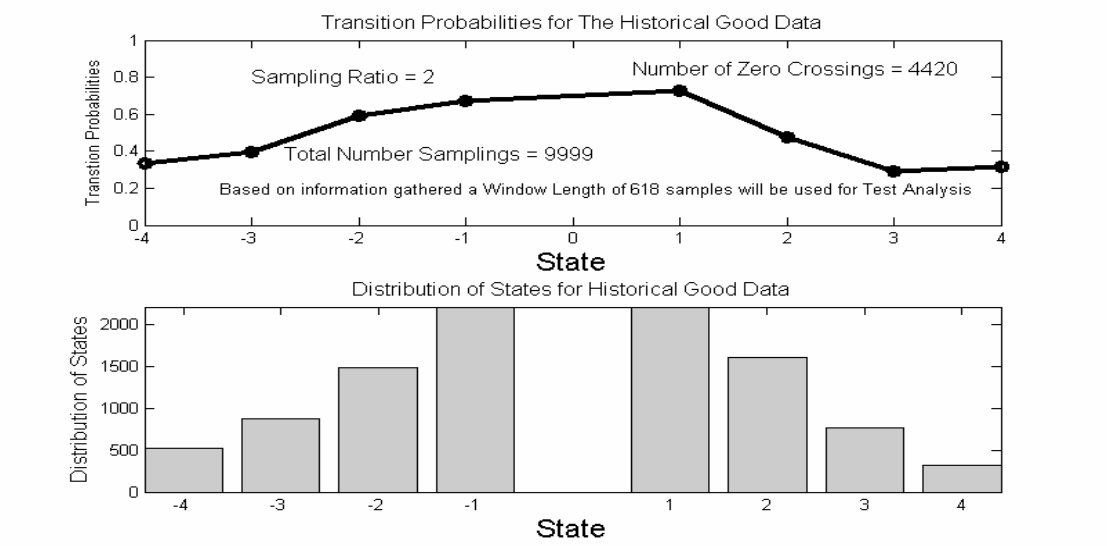


Figure 4.16 Distribution of States and Transition Probabilities from Reference Good Data (Window length = 800 samples; Sampling Ratio = 2, α_T = 5%, β = 1%, λ = 0.9)

However, between sampling 15000 and about 28500, the monitor initialized the violation counter, started counting and reset to zero more often than was observed when α was 1%. This is indicative of the user choice of the α_T. With α_T set at 5%, it means for k = 8 total states, $\alpha_k = \left(1-\sqrt[k]{1-\alpha_T}\right)$ = 0.639% for each state. Consequently, approximately 0.639% of the time, there will be a false alarm associated with transitions from one state to the other.

Previously, when α_T was set at 1%, it meant for k = 8 total states, $\alpha_k = \left(1-\sqrt[k]{1-\alpha_T}\right)$ = 0.1255%. In comparison, it means that when α_T is increased from 1% to 5%, the monitor starts counting about 5 times more frequently. In Figure 4.18, α_T was

changed to 10% and β kept at 1%, again the algorithm estimated a window length of 572 samples even smaller than when α_T was 5% or 1%. However, the test analysis showed numerous instances of spurious alarms due obviously to the choice of α_T.

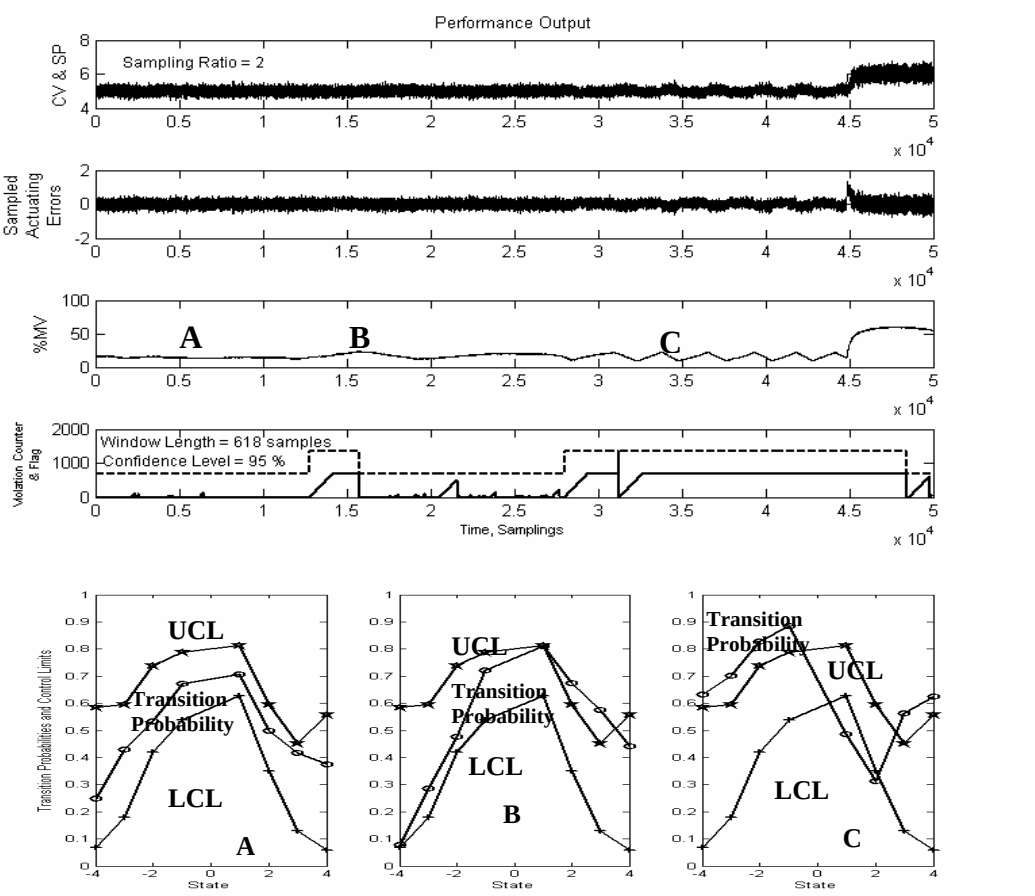


Figure 4.17 Control Loop Performance Output for Pressure Drop Control by Manipulating signal to Water Flow Control Valve (Sampling Period = 0.1s, Sampling Ratio = 2; Window length = 618 Samples, Startup Period = 0 Samples, Grace Period 668 Samples, Violation Counter Trigger = Length of Grace Period +1, Overall Level of Significance (α_T = 5%). Performance Output (A = No nuisances in the loop B = Oscillations in Loop, C = Stiction in Loop)

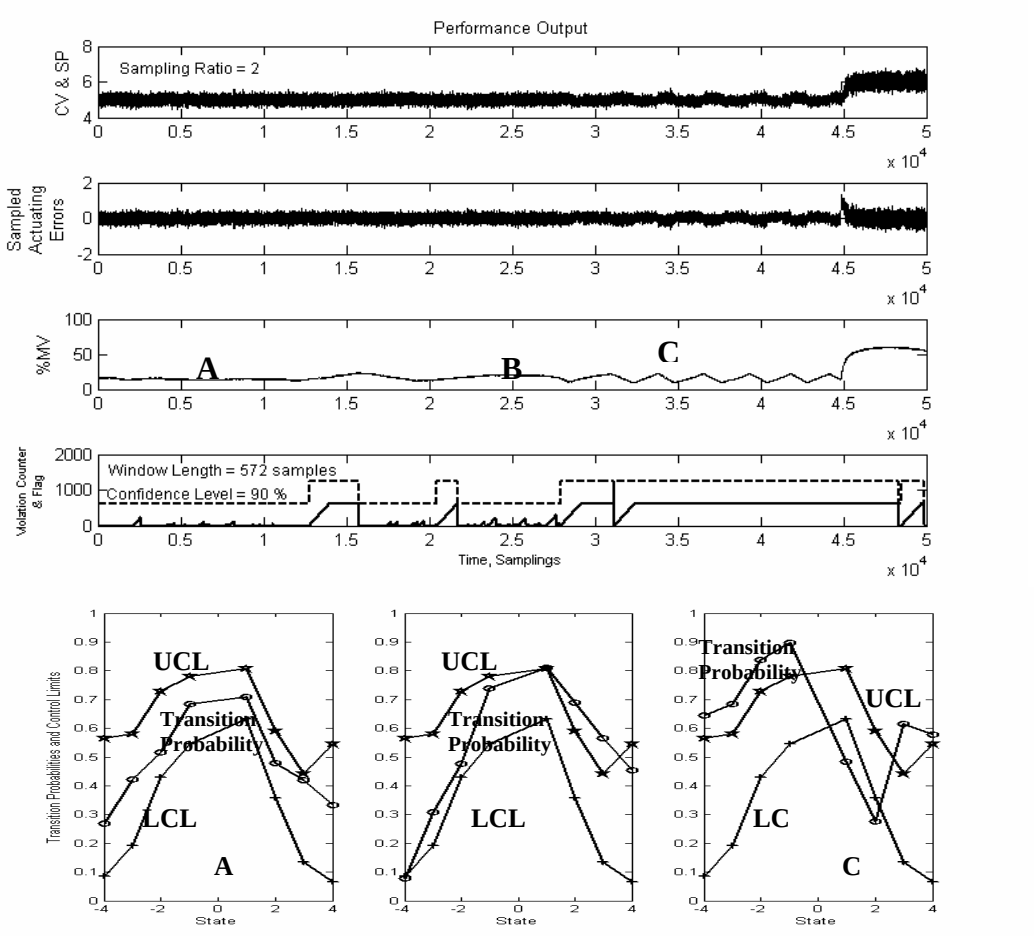


Figure 4.18 Control Loop Performance Output for Pressure Drop Control by Manipulating signal to Water Flow Control Valve (Sampling Period = 0.1s, Sampling Ratio = 2; Window length = 572 Samples, Startup Period = 0 Samples, Grace Period 612 Samples, Violation Counter Trigger = Length of Grace Period +1, Overall Level of Significance (α_T = 10%). Performance Output (A = No nuisances in the loop B = Oscillations in Loop, C = Stiction in Loop

4.3.4 Effect of Varying the Type-II Error Rate on Window Length and Performance Monitoring

The Type-II error rate β, was varied keeping the Type-I error rate same in order to study how changing β, affects the window length and the overall performance of the monitor. Using a confidence level of 99%, the reference data was analyzed again and the result is shown is Figure 4.19. The monitor estimated a window length of 562 samples but the sampling ratio SR, was again same as previous (SR = 2). The Type-II error rate used for the state having the least number of reference samples was changed from 1% to 10% when future transition probabilities during testing differ by 0.9 from the reference value.

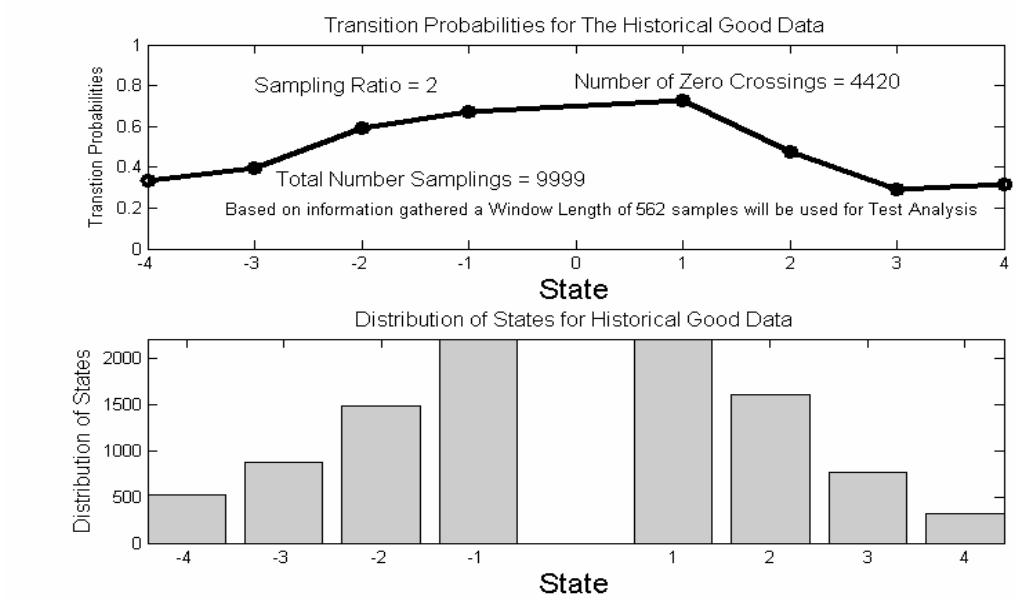


Figure 4.19 Distribution of States and Transition Probabilities from Reference Good Data (Window length = 800 samples; Sampling Ratio = 2, α_T = 1%, β = 10%, λ = 0.9)

The monitor detected all instance of loop upsets within the regions marked “A” “B”, and “C” respectively of Figure 4.20. However, notice from Figures 4.15, 4.17 and 4.18 that the flagging of the violations in region “B” stopped at about sampling 16000. Nevertheless, in Figure 4.20 flagging stopped at almost 15000. The reasons may be due to the short window length or that the monitor might be missing some alarms that should been detected. Also, notice that the violation in region “C” appeared to go off although it can be seen from plot 3 of Figure 4.20 that the stiction effect was still present in the loop. This could be due to a missed alarm. These two instances of missed alarm are related to the Type-II error set by the user. Changing β from 1 to 10% reduces the Power of the hypothesis test resulting in frequent missed alarms. This means that it is always essential for the user to provide a reasonably good value of α_T and β, in order to reduce the number of missed and false alarms. For loops where tight control is not desired, this requirement can be relaxed but the user should expect frequent false and or missed alarms.

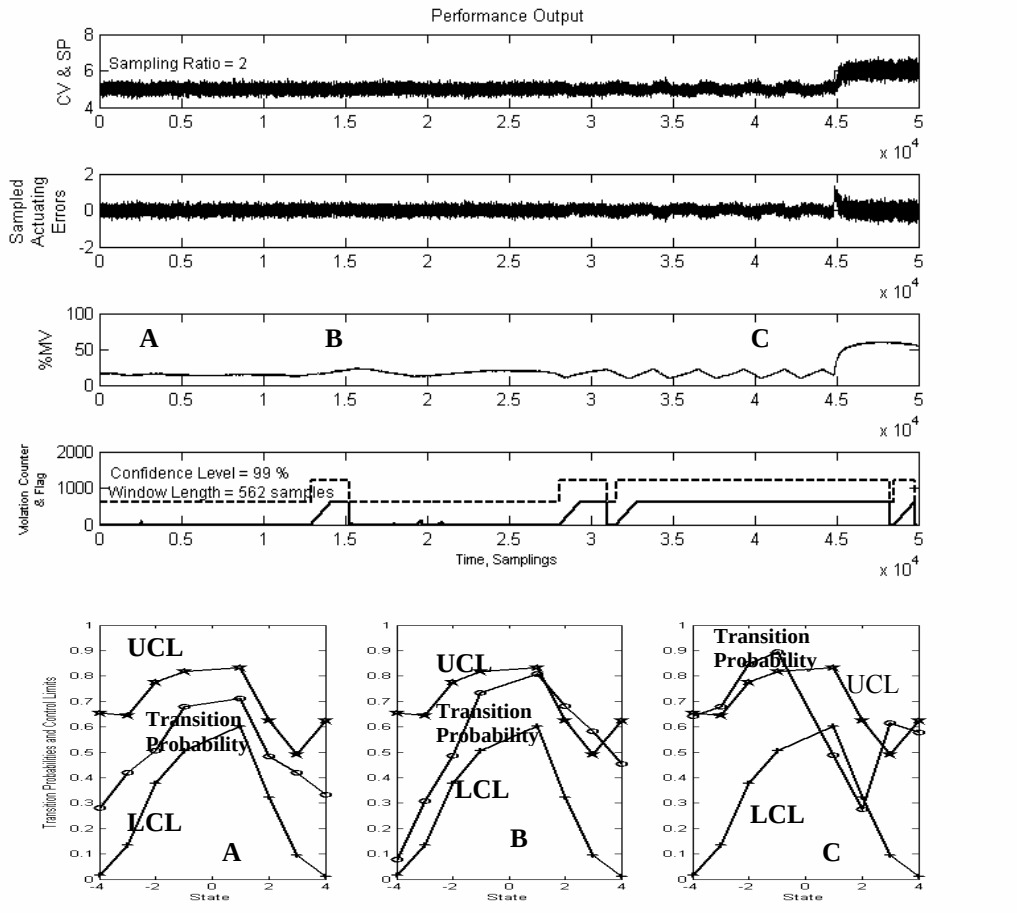


Figure 4.20 Control Loop Performance Output for Pressure Drop Control by Manipulating signal to Water Flow Control Valve (Sampling Period = 0.1s, Sampling Ratio = 2; Window length = 562 Samples, Startup Period = 0 Samples, Grace Period 612 Samples, Violation Counter Trigger = Length of Grace Period +1, Overall Level of Significance (α_T) = 1%). Performance Output (A = No nuisances in the loop B = Oscillations in Loop, C = Stiction in Loop)

4.3.5 Performance Monitoring for Pressure Drop Control by Manipulating Signal to Air Flow Control Valve

The second control scheme studied involved controlling the pressure drop in the two-phase flow column by manipulating the airflow rate. Again, the controllers were first tuned and when control was determined to be good, data was collected for sometime.

The reference data was analyzed and is shown in Figure 4.21. After analysis of the reference data the algorithm estimated that a sampling ratio of 3 will be ideal for test analysis. Based on a control loop data logging rate of 10 samples every second for the unit, it implies that the monitor will sample data from the loop every 0.3 seconds. In addition, the window length for statistical comparisons was determined by the monitor to be 1723 samples. This means that the monitor will sample a window length of data in approximately 8.6 min. The confidence level used was 99% and the Type-II error rate used for the state having the least number of reference samples was set at 1% when future transition probabilities during testing differ by 0.9 from the reference value.

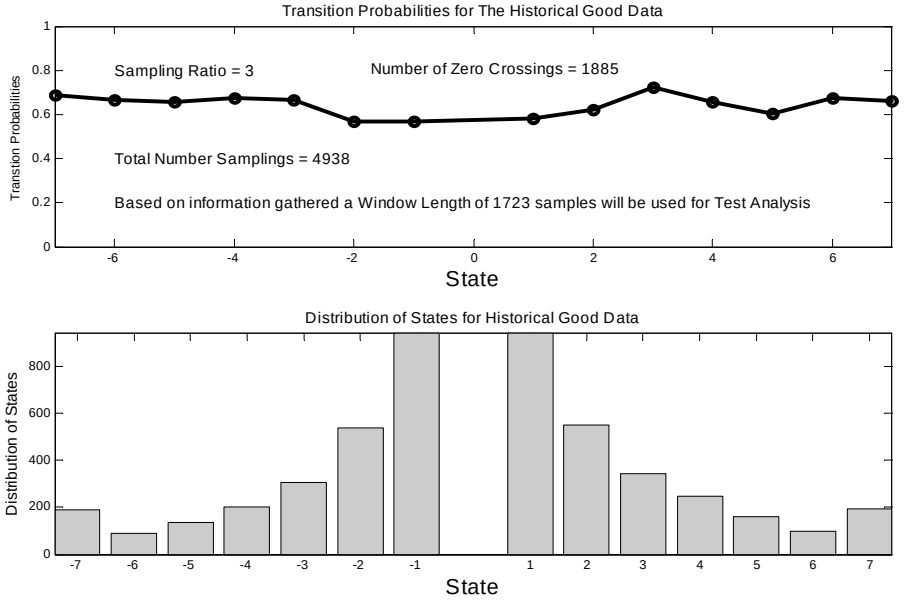


Figure 4.21 Distribution of States and Transition Probabilities from Reference Good Data (Window length = 1723 samples; Sampling Ratio = 3, α_T = 1%, β = 1%, λ = 0.9)

Using information from analysis of the reference data, test data collected was analyzed as shown in Figure 4.22. At sampling 3000, in the region marked "A", the controller gain was increased by 3 fold. The monitor detected that new data collected had different distribution from the reference data and so the violation counter started counting after control limits were violated. At about sampling 8000, when the controller gain was reset to it's nominal value, the violations went off and the monitor reset the counter to zero. Around sampling 18000, in the region marked "B", the controller was detuned by a factor of 0.5. The violation counter did not start counting because it seemed that the controller was not detuned to the to the point of sluggishness and so the monitor determined that the data collected during that period was not significantly from the reference data so no violations were registered. Around, sampling 55000 in the region marked "C", the water flow rate was intentionally increased: In response, notice the manipulated variable reduce in an attempt to adjust the pressure drop to the setpoint and the variation in the manipulated variable and controlled variable. The monitor detected the control degradation in the loop and started the violation counter. Again, when the water flow rate was reset to the nominal value the violations went off and the violation counter was reset to zero.

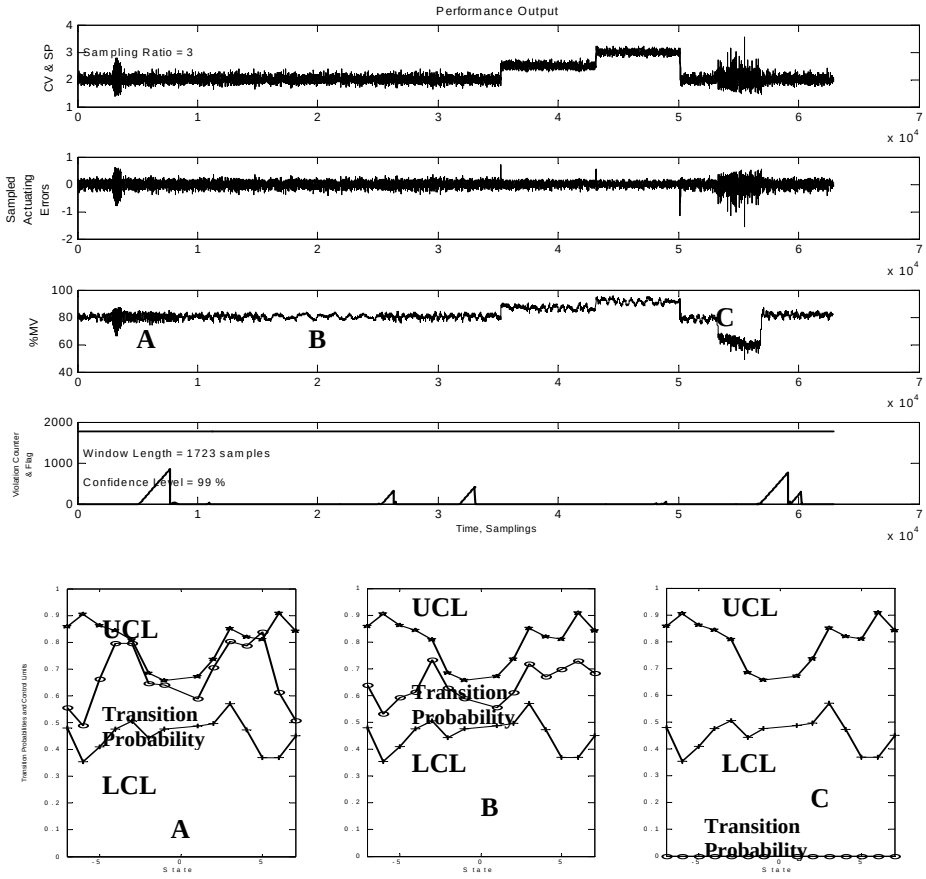


Figure 4.22 Control Loop Performance Output for Pressure Drop Control by Manipulating signal to Air Flow Control Valve (Sampling Period = 0.1s, Sampling Ratio = 3; Window length = 1723 Samples, Startup Period = 0 Samples, Grace Period 1773 Samples, Violation Counter Trigger = Length of Grace Period +1, Overall Level of Significance (α_T) = 1%). Performance Output (A = Controller gain increased, B = Controller gain reduced, C = Water flow Rate Increased)

4.3.6 Performance Monitoring for Water Flow Control by Manipulating Signal to Water Flow Control Valve

The next control scheme studied involved controlling the water flow rate by manipulating the signal to the water flow control valve. The controller was first tuned around a nominal setpoint of 10 kg/hr. After control was determined to be good, data was collected for a period. The reference data is analyzed as shown in Figure 4.23. After

analysis of the data, the monitor estimated a sampling ratio of 4. Based on a data logging rate of 10 samples every second for the unit, it implies that the monitor will sample data from the unit every 0.4 seconds. In addition, the window length for statistical comparisons was determined by the monitor to be 1227 samples so it will take approximately 8.16 minutes to sample a window length of data. The confidence level used was 99% and the Type-II error rate used for the state having the least number of reference samples was set at 1% when future transition probabilities during testing differ by 0.9 from the reference value.

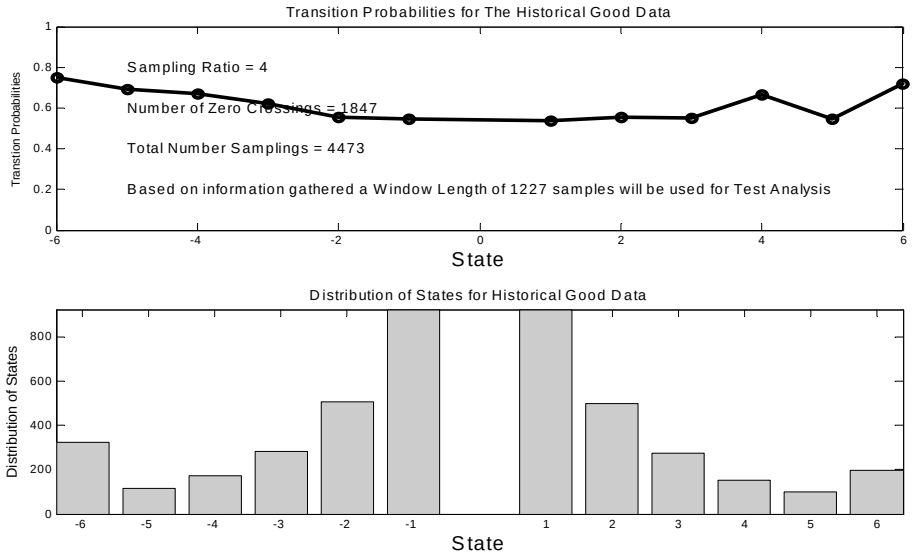


Figure 4.23 Distribution of States and Transition Probabilities from Reference Good Data (Window length = 1227 samples; Sampling Ratio = 4, α_T = 1%, β = 1%, λ = 0.9)

Using information from analysis of the reference data, test data collected was analyzed as shown in Figure 4.24. At sampling 8000 and within the region marked "A" there was a setpoint change from 10 kg/hr to 20 kg/hr. The controller was able place the

controlled variable at the desired setpoint rapidly and so no control limits were violated and the violation counter did not start. At sampling 17000, the setpoint was changed again from 20 to 25 kg/hr. At this new setpoint, oscillations were detected in the loop and the monitor detected this and started the violation counter.

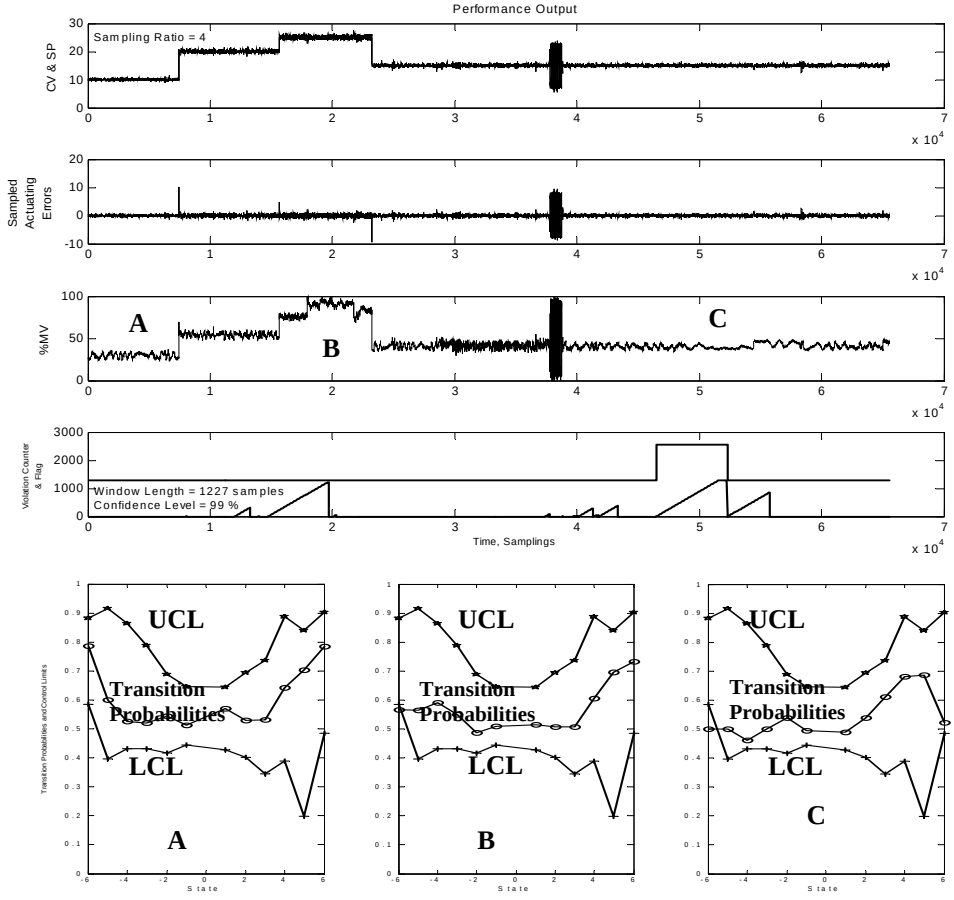


Figure 4.24 **Control Loop Performance Output for Flow Rate Control by Manipulating signal to Water Flow Control Valve** (Sampling Period = 0.1s, Sampling Ratio = 4; Window length = 1227 Samples, Startup Period = 0 Samples, Grace Period 1277 Samples, Violation Counter Trigger = Length of Grace Period +1, Overall Level of Significance (α_T) = 1%). Performance Output (A = Controller gain increased, B = Controller gain reduced, C = Water flow Rate Increased)

It appears that this new setpoint was far removed from the nominal value used to tune the controller, and it was very near the maximum flow rate allowable for the valve. The monitor detected that data sampled during the period had a different distribution from the reference period and so once control limits were violated the monitor started the violation counter. Around sampling 20000, within the region marked "B", the setpoint was reset to 12 kg/hr. However, just prior resting the setpoint, the monitor reset the violation counter to zero indicating that the controller might have been able to eliminate the oscillations encountered due to the setpoint change. At sampling 39000 in Figure 4.24, the controller was made aggressive but this aggressive period also did not exceed the grace period. Even if counting had been continuous during the high gain period, it would have stopped prior to setting the flag. Perhaps the high gain was not high enough to set the counter. At sampling 50000, and in the region marked "C", the controller was detuned by a factor of 0.3. The monitor detected a change in the distribution of the data it was sampling, started the violation counter when control limits were violated and after the grace period was exceeded, a flag was raised retroactively to indicate the start of the problem. Other instance when the violation counter started counting when it was not supposed to may be attributed to the Type-I error rate of 1% used for the test.

4.3.7 Application of the Health Monitor on Qing Li's Data (Monitoring the Performance of Water Flow Control Loop by Manipulating Signal to Water Flow Control Valve)

As indicated earlier, this work is an extension of Li's work (2002). Li developed a health monitor using run length distribution of actuating errors between two consecutive zero crossings and detected changes in sample distribution using the Chi-square goodness

of fit test. Li tested his monitor on unit operation data he collected at that time. This version of the Health monitor also was tested on Li's data in order to compare the performance of his monitor and the monitor developed in this work.

Li collected data from the two-phase flow experimental unit by controlling the water flow rate while manipulating the signal to the water flow control valve. Data logging rate was 10 samples every seconds. He tuned the controller around a nominal operating point of 35 kg/hr. After tuning, he determined the closed loop settling to be about 50 samples. He collected data during which time he did not introduce any nuance(s) in the loop. The reference data was analyzed using the monitor developed in this work and the result is shown in Figure 4.25. The sampling ratio was estimated to be 3 and the window length 1334 samples. With a sampling ratio of 3, it means the monitor samples 10 data points every 3 seconds. Thus, it will take about 6.67 minutes to collect a window of data and about same time for a flag to be raised during testing if a problem is detected.

Using information from the good control period, a test data collected by Li in which he made several upsets in the loop was analyzed. The results are shown in Figure 4.26. Between sampling 2100 and 4000, the setpoint was reduced from 35 to about 20 kg/hr, in gradual steps, but the controller was able to place the controlled variable at the setpoint at each change and so no violations were registered during that interval and the monitor did not start the violation counter. Around sampling 4800, the setpoint was changed again to 15 kg/hr. This resulted in oscillations in the loop. The violation counter started counting once control limits were violated and after the grace period was exceeded and the

problem was still in the loop the monitor raised a flag. For as along as the setpoint was kept at 15 kg/hr and below the monitor continued to flag.

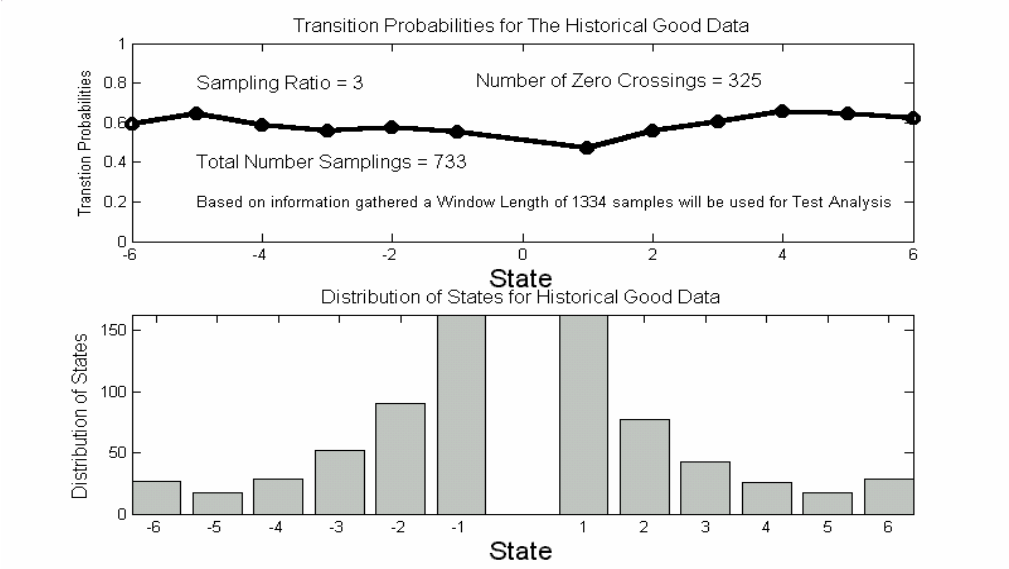


Figure 4.25 Distribution of States and Transition Probabilities from Reference Good Data (Window length = 1334 samples; Sampling Ratio = 3, α_T = 1%, β = 1%, λ = 0.9)

However, around sampling 14000 when the flow rate was reset to the nominal operating point, the oscillations stopped and monitor reset the flag and violation counter to zero.

In comparison with the run length based monitor, it was noted that the run length based approached used a shorter window length of 129 samplings for statistical analysis as compared with 1334 samplings needed by this approach. Both techniques are able to detect problems when they should. Nevertheless, the Chi-Square approach was noted to flag on and off even when nuances and instabilities are still known to be present in a loop

(during the entire 6000 to 12000 sampling period) while the technique used in this work flags on until the nuances detected (caused by low setpoint changes) are removed.

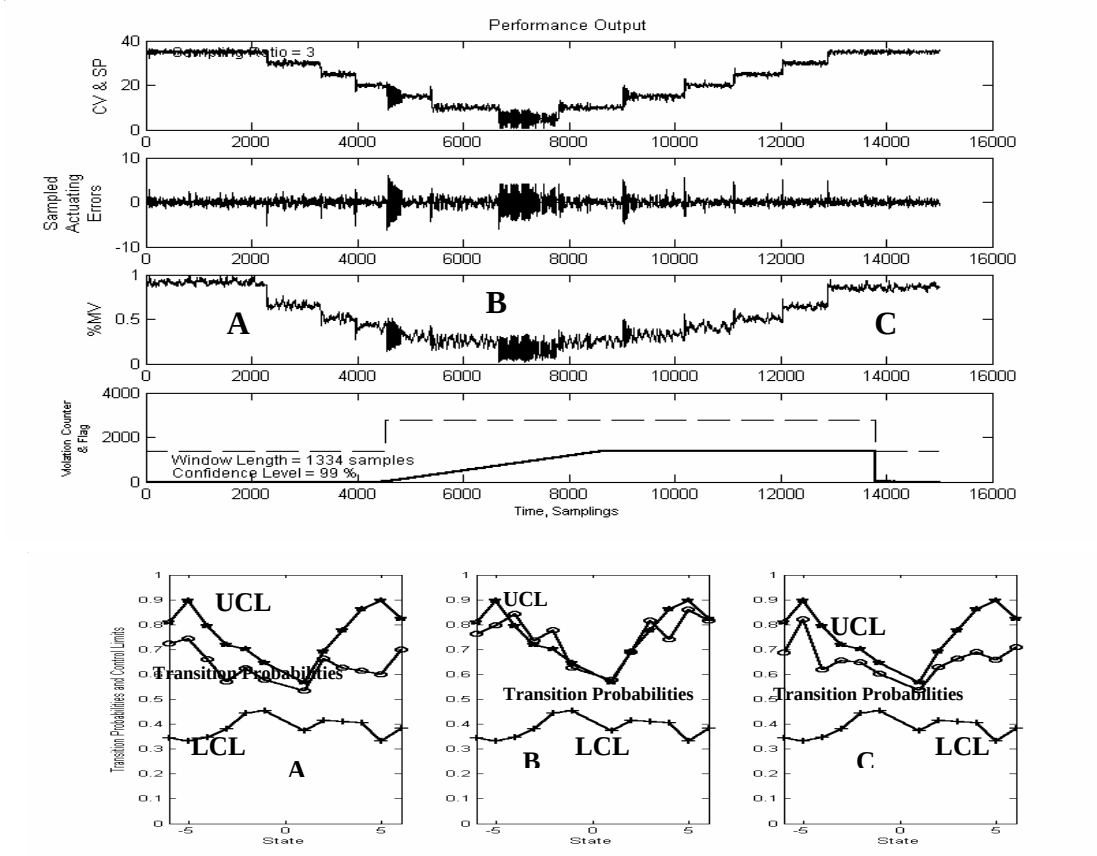


Figure 4.26 Control Loop Performance Output for Flow Rate Control by Manipulating signal to Water Flow Control Valve (Sampling Period = 0.1s, sampling ratio = 3; Window length = 1334 Samples, Startup Period = 0 Samples, Grace Period 1384 Samples, Violation Counter Trigger = Length of Grace Period +1, Overall Level of Significance (α_T) = 1%). Performance Output (A = Controller gain increased, B = Controller Gain Reduced, C = Water flow Rate Increased)

4.3.7 Application on Industrial Data

One of the industrial sponsors for this project (ExxonMobil) provided us with data from their plant to test this monitor offline. The unit is shown schematically in Figure 4.27. It consists of a cascaded exothermic reaction process for the manufacturing polypropylene. The reaction temperature is controlled by cooling the feed to the process using cooling water supplied via a heat exchanger arrangement. The reaction temperature which is primary process variable (PV_p) is transmitted to the primary controller (TC1) via the transmitter (TT1). The desired process reaction setpoint (SP_p) is provided via TC1 which compares SP_p and PV_p and then writes a secondary setpoint to the secondary controller (TC2). The secondary controller is connected to the process stream flowing to the reactor via transmitter TT2.

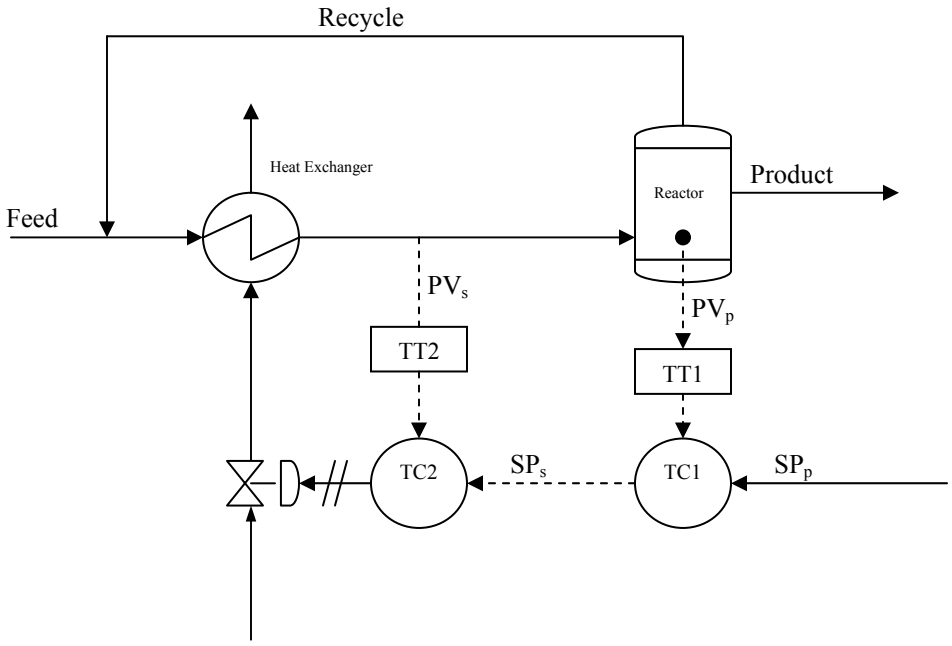


Figure 4.27 Cascade Polymer Processing Unit: Reactor Temperature, PVp is Controlled by Cooling the Feed Temperature, PVs

TC2 sends signals to the valve, which adjust the flow rate of cooling water used in manipulating the feed temperature. The reference data was analyzed separately for the primary and secondary loops. The data logging was done at 1 sample every 5 seconds. The reference analysis for the primary loop data is shown in Figure 4.28. The algorithm estimated a sampling ratio of 34 and a window length of 833 samples. With this sampling ratio, it implies that the monitor will require approximately 39.34 hrs, $\left(141610\text{ s; i.e. }\left(\frac{5\text{ s}}{1\text{ controller sample}}\right)\left(\frac{34\text{ controller samples}}{1\text{ monitor sample}}\right)*(833\text{ monitor samples})\right)$ to sample a window length of data. Based on this, the sampling ratio was judged to be too large. A further investigation reveals that the data has strong autocorrelation, which perhaps may be contributing to the large sampling ratio. In Appendix D, an attempt is made to investigate how the sampling ratio can be reduced by adjusting the extent of autocorrelation present in the data.

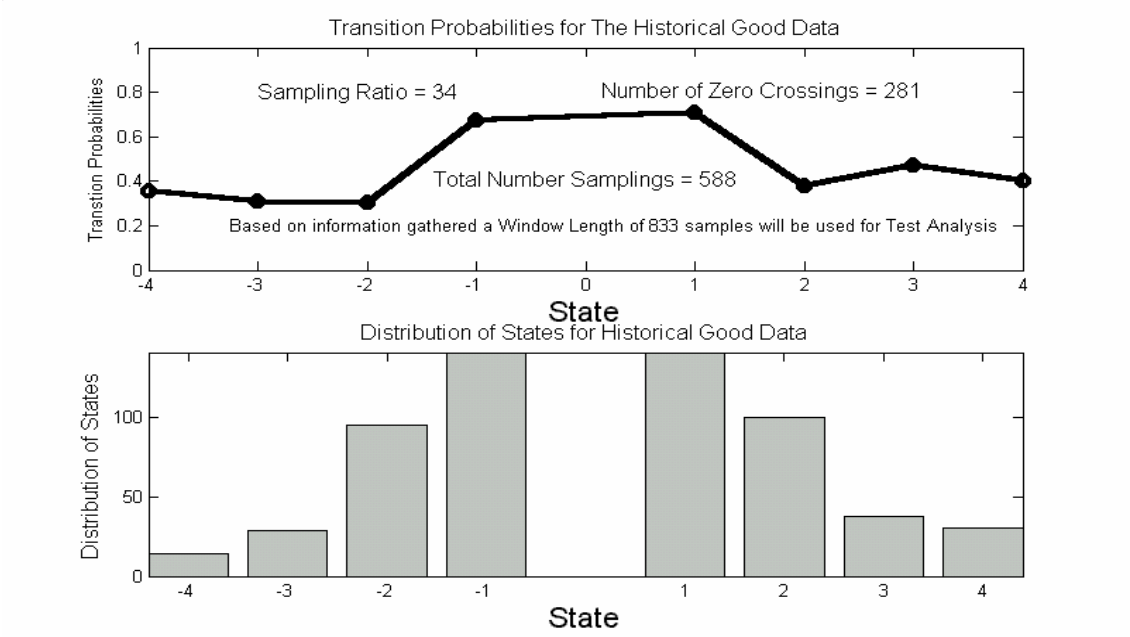


Figure 4.28 Distribution of States and Transition Probabilities from Reference Good Data (Window length = 833 samples; Sampling Ratio = 34, α_T = 1%, β = 1%, λ = 0.9)

The test data was then analyzed as shown in Figure 4.29 for the primary control loop. At sampling 15000, according to notes accompanying the data, there was an increase in production rate resulting in an overshoot in temperature as shown on the first plot in Figure 4.29. In response, the feed temperature dropped as shown in the third plot to regulate the reactor temperature. The monitor did not show the nuance because the controller was able to stabilize the disturbance in the loop and place the controlled variable back at the setpoint.

Around sampling 30000, the notes indicated that a pump used to supply cooling water was switched to a backup unit. This led to oscillations in both primary and secondary loops. In response, a control engineer detuned the gain on the secondary controller.

It appeared that the oscillations went off after the secondary controller was detuned. However, at sampling 40000, the monitor started the violation counter again but reset it back to zero after a short while. This was probably an indication that there were still some residual oscillations present in the loop. According to the notes, about 90 % through the data, the integral time on the secondary controller (TC2) was changed in order to reduce oscillations in the process variable.

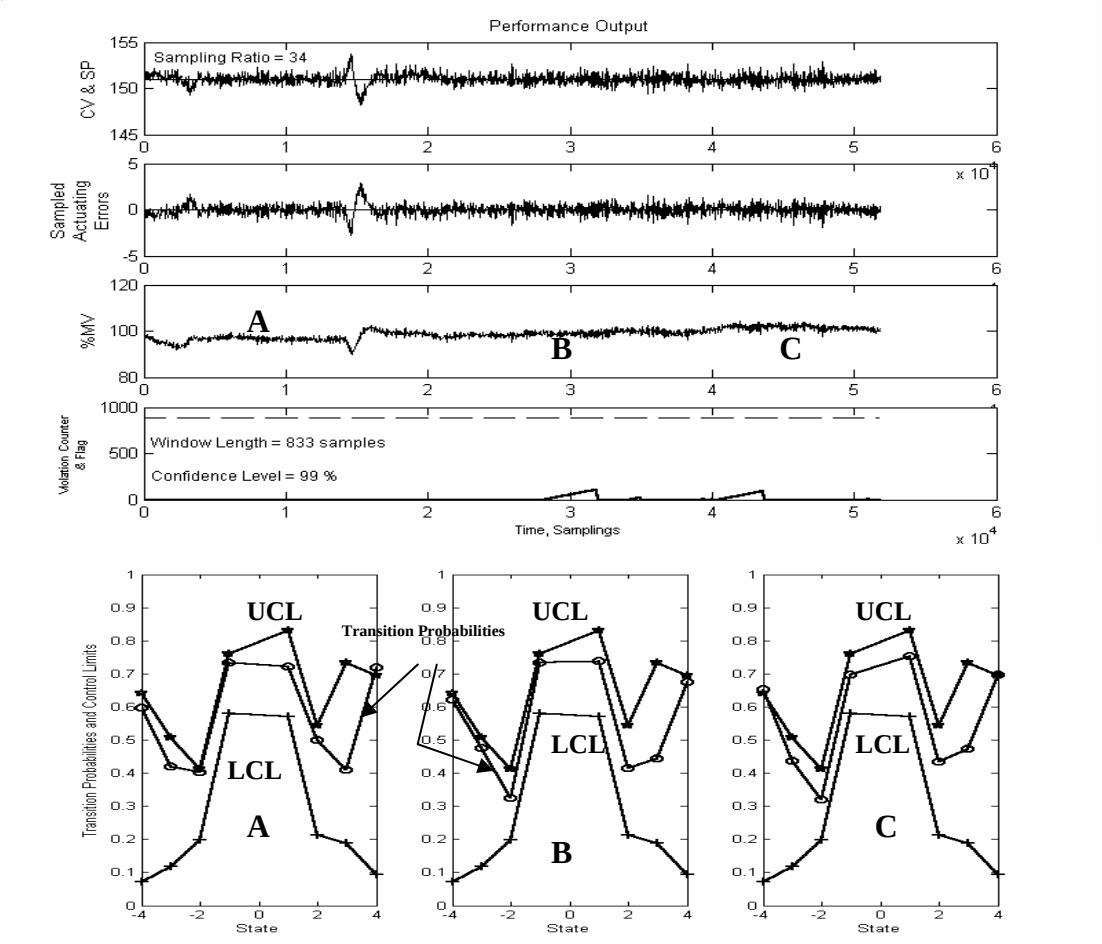


Figure 4.29 Control Loop Performance Output for Primary Loop Temparaure: ExxonMobil Data (Sampling Period = 5s, Sampling Ratio = 34; Window Length = 833 Samples, Startup Period = 0 Samples, Grace Period 883 Samples, Violation Counter Trigger = Length of Grace Period +1, Overall Level of Significance (α_T) = 1%). Performance Output (A = No Nuance in Loop, B = Oscillations in Loop, C = Oscillations in Loop)

Figure 4.30 shows analysis of the reference data for the secondary loop. The monitor estimated a sampling ratio of 11 and a window length of 513 samples, which was used for the test analysis in Figure 4.31. Notice that from sampling 30000 when the pump was switched through to about the end of the data when there were oscillations in the loop, the monitor flagged consistently after the grace period was exceeded. This reveals that the monitor is able flag when it is supposed to and does not flag when it is not supposed to.

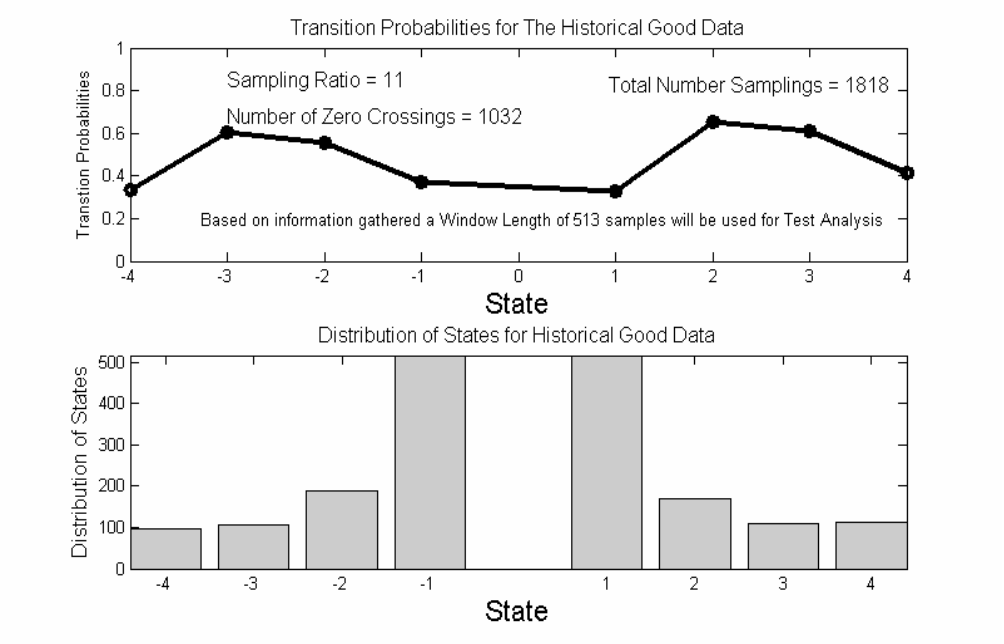


Figure 4.30 Distribution of States and Transition Probabilities from Reference Good Data (Window length = 513 samples; Sampling Ratio = 11, α_T = 1%, β = 1%, λ = 0.9)

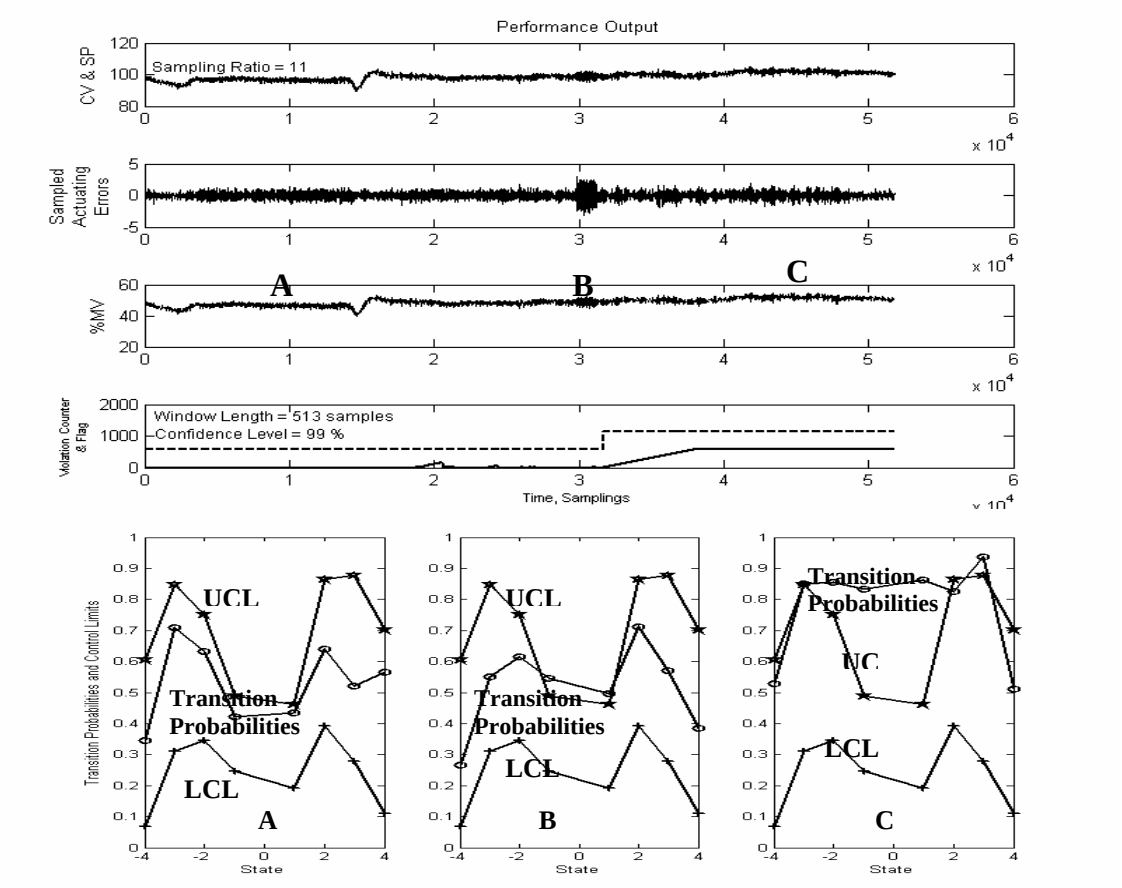


Figure 4.31 Control Loop Performance Output for Secondary Loop Temparaure: ExxonMobil Data (Sampling Period = 5s, sampling ratio = 11; Window Length = 513 Samples, Startup Period = 0 Samples, Grace Period 563 Samples, Violation Counter Trigger = Length of Grace Period +1, Overall Level of Significance (α_T) = 1%). Performance Output (A = No Nuance in Loop, B = Oscillations in Loop, C = Oscillations in Loop)

4.4 Model Based Control (MBC)

4.4.1 Simulation Using Internal Model Control

This section discusses a simulated application of the health monitor on a process controlled using Internal Model Control (IMC) technique. Details of the process description are provided in Appendix C. Figure 4.32 is a schematic illustration of an IMC control loop with the health monitor in tandem with an IMC controller and sampling the actuating errors.

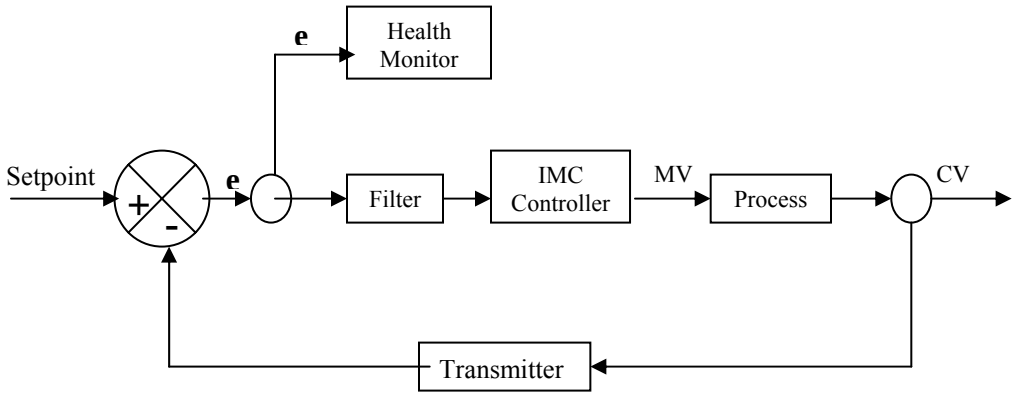
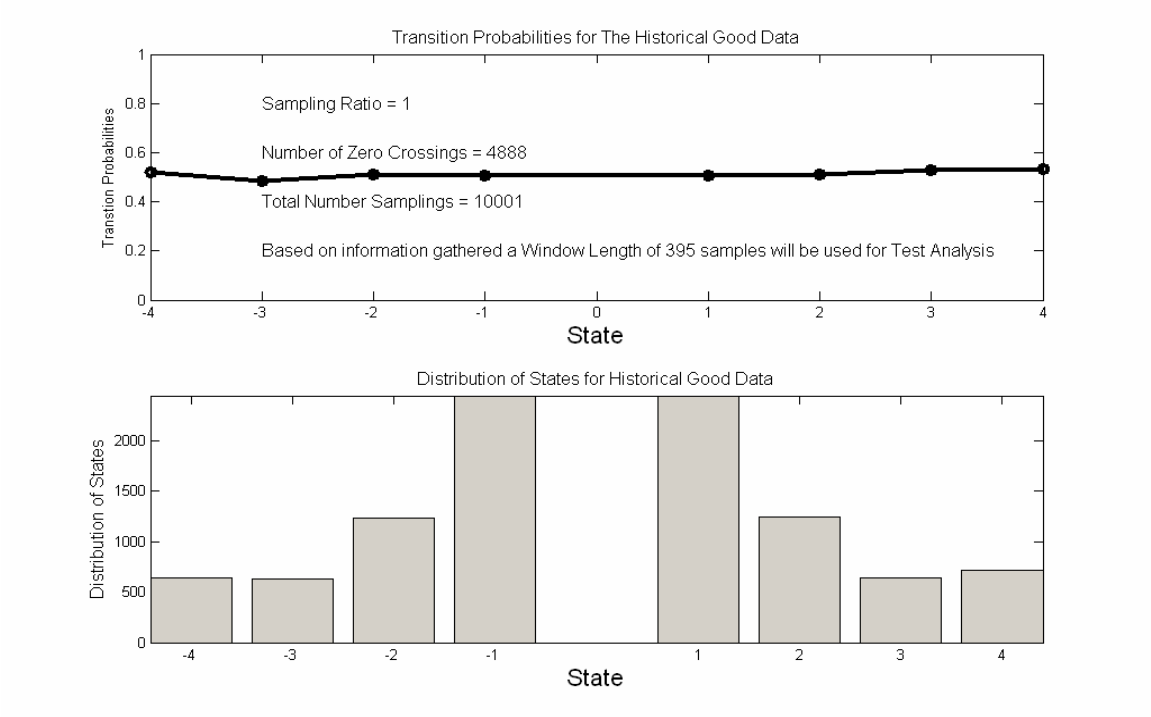


Figure 4.32 Schematic Diagram of a Process Controlled with IMC Technique (e = Actuating Error, CV = Controlled variable, MV = Manipulated Variable

Data collected during a period of good control was analyzed and is shown in Figure 4.33. The sampling time interval for the controller was 0.25 time units. Based on the data analyzed, the algorithm estimated a sampling ratio of 1 (i.e. health monitor sampling time interval = 0.25 time units) and a window length of 395 samples to be ideal for test analysis. With a closed loop settling time of about 50 samples, it implies that the grace period during test analysis will be 445 samples and that during test analysis, it will take approximately, (0.25 time units/monitor samples)*(445 monitor samples) time units

for the monitor to flag if a problem was detected in the loop. Assuming time units to be in seconds, it means it will take about 1.85 minutes before the monitor will flag. Violation counting however starts instantaneously.

Figure 4.33 Distribution of States and Transition Probabilities from Reference Good Data (Window length = 395 Samples; Sampling Ratio = 1)



The monitor was initialized with 8 total states (± 4 on each side), and the algorithm estimated that it was enough to meet the requirement of having no more than 10 % of the data in the extreme states. After the analysis, and before estimating the window length, the algorithm selects the state with least number of samples on each half of the Figure 4.33 and determines the number of samples to place in those states in order to meet the requirement on Type-I and Type-II error. For this example, it can be

observed from Figure 4.33 that the states that have the least number of samples are the +3 state on the positive side and the –3 state on the negative side. Only one of the two states (in this example +3 state) is chosen and revealed in Figure 4.34.

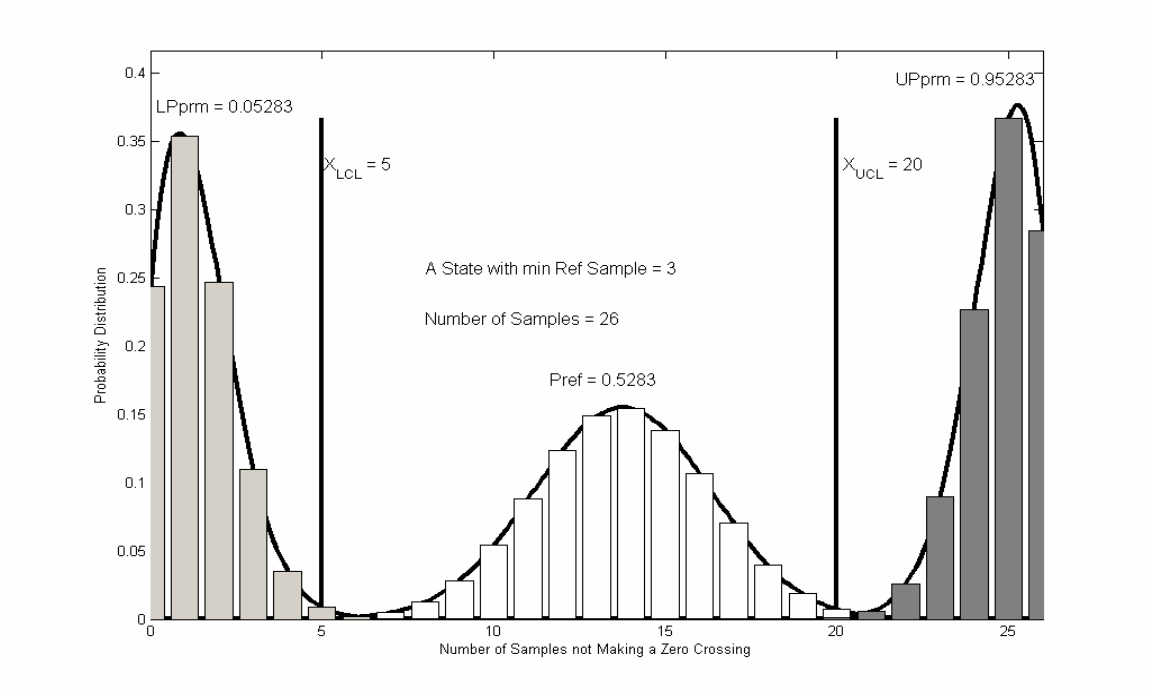


Figure 4.34 Analysis of Reference Data for State with Least Number of Samples (Distribution of Samples Leaving State and Entering the Next Absolute Higher State Given the Reference Probability, α, β and λ)

The algorithm estimated the number of samples that need to visit this state (+3 state) to be 26 given choices of α and β and the how far way from the reference transition probability that this β is desired. X_L indicates the minimum number of samples that need to leave the state and visit the next absolute higher in order to reduce the Type-I error rate to the desired level, while X_H indicates maximum number of samples that need to leave a state and visit the next absolute higher in order to minimize the Type-I error rate to the

desired level as well. For the illustrated case, $X_L = 5$ and $X_H = 20$. Thus out of a total of 26 samples that need to visit the state of +3 in a given window (to be determined), a violation will occur if fewer than 5 samples leave the state and visit the next absolute higher state (i.e. +4) or more than 20 samples leave the state and visit the next absolute higher state. Furthermore, Figure 4.34 illustrates that if future transition probabilities are not different from the reference value of 0.5283, then for example, 15% of the time, say, about 14 samples will leave the state of +3 and visit next absolute higher state (i.e. +4). In addition, if future transition probabilities were to change significantly to say 0.95283, then about 23% of the time, 24 samples will leave that state of +3 and visit the state of +4 and when it happens this will be outside the desired upper control limit of 20 and so a violation will occur. Similarly, if future transition probabilities were to change significantly to say 0.05283, then about 24% of the time only 2 samples will leave the state of +3 and visit the state of +4 and since this is outside the lower control limit of 5, a violation will occur. The chance of missing a violation as can be seen from the Figure 4.34 is negligibly small, as desired.

After the analysis on this state is complete and the number of samples in all the other states are determined, the algorithm selects one state at random and determines the statistical errors that are associated with it based on the choices made. For this example, the algorithm selected the state of +1 and based on the reference transition probability, it determined that 97 samples need to visit the state of +1 (see Figure 4.35). In addition, if future transition where to differ from the reference value to say 0.050757, then about 10% of the time say, about 7 samples may be expected to leave the state of +1 and visit the state of +2.

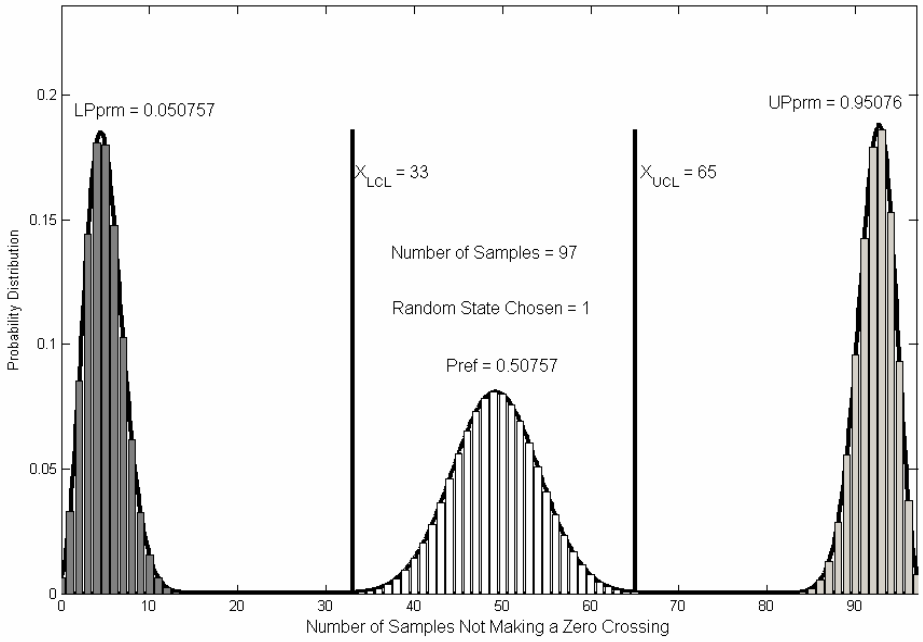


Figure 4.35 **Analysis of Reference Data for State Chosen at Random** (Distribution of Samples Leaving State and Entering the Next Absolute Higher State Given the Reference Probability, α, β and λ)

Furthermore, if probabilities differ to about 0.95076, then about 15% of the time 90 samples may be expected to leave that of +1 and visit the state of +2. In both case, these number of visits are outside the control limits 33 on the lower side and 65 on the upper side and so it will mean a violation of the control limits. Once all samples that need to visit all the states during a period of good control are known, the monitor estimates the window length necessary for performance monitoring. Once the window length is estimated, a power curve for the entire test is plotted. This is shown in Figure 4.36.

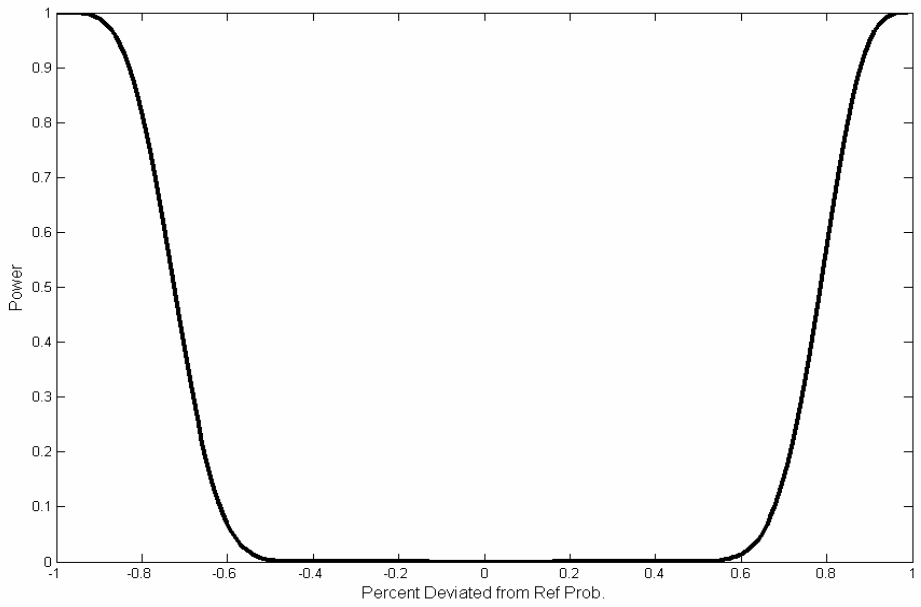


Figure 4.36 Power Curve (Given α = 1% and β= 1% λ = 90%)

4.4.2 Performance Evaluation of Health Monitor on the IMC Process

The performance of the health monitor is illustrated in Figure 4.37. At sampling 4000, 8000 and 12000, set point changes were made. However, the controller was able to rapidly place the controlled variable at the new setpoint after each change and so no control limits were violated during these periods

Between sampling 16,000 and sampling 22000 and in the region labeled "A" in Figure 4.37, the filter time-constant was altered by a factor 0.2. This made the controller aggressive resulting in degrading performance. Control limits were violated, and so the monitor started the violation counter. After the grace period was exceeded and the violations were still present in the loop, the monitor raised a flag retroactively for the entire duration when the filter time-constant was altered. After the filter time-constant

was restored to the original value, the controller was able to restore stability in the loop and the violations were removed. Once the monitor detected a return to good control, it stopped flagging and reset the violation counter to zero.

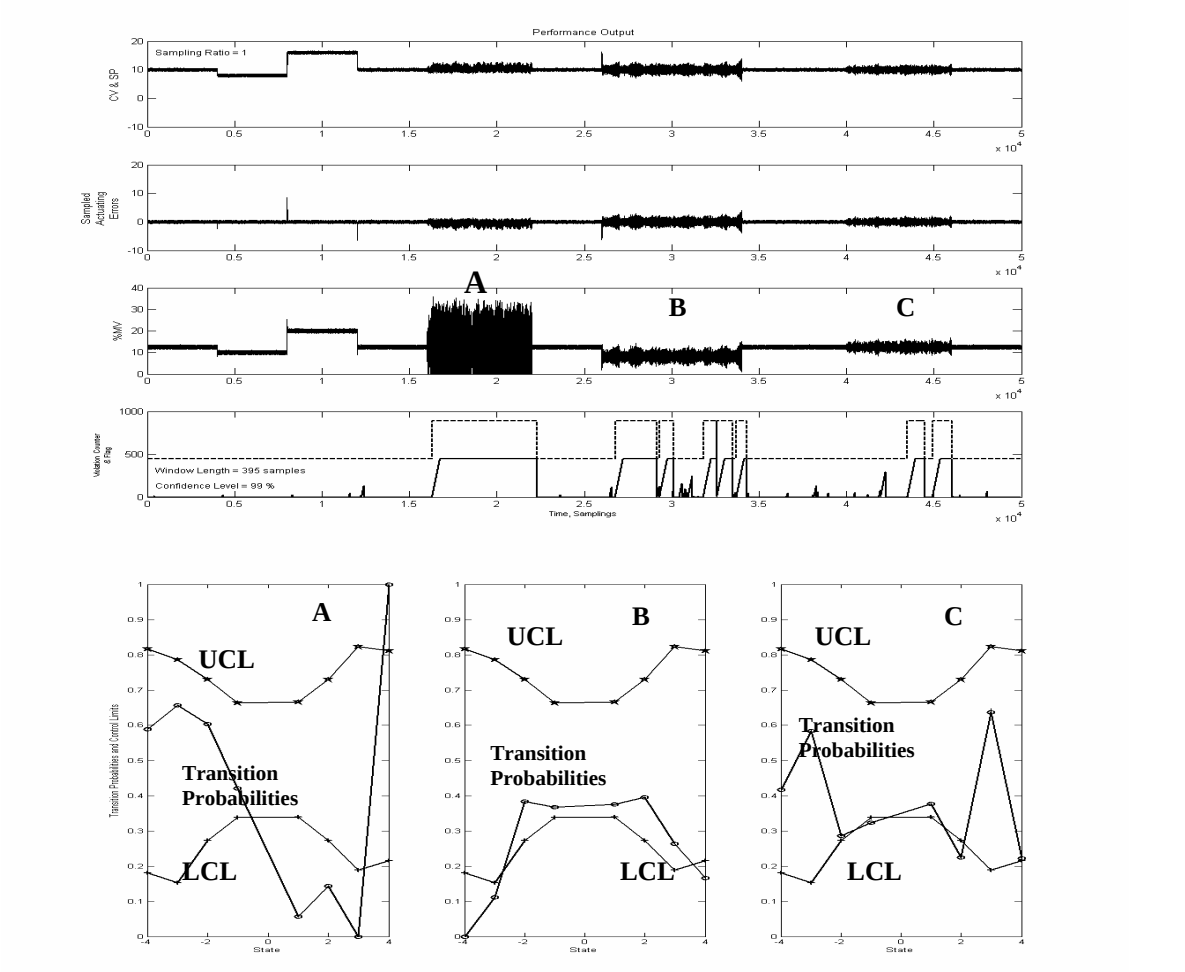


Figure 4.37 **Control Loop Performance Output** (Sampling Interval = 0.25 Time Units, Sampling Ratio = 1; Window length = 395 Samples, Startup Period = 0 Samples, Grace Period 445 Samples, Violation Counter Trigger = Length of Grace Period +1, Overall Level of Significance (α_T) = 1%). Performance Output (A = Filter Time Constant Decreased, B = Process Gain Increased, C = Process Gain Decreased)

Between sampling 26,000 and 34,000, labeled “B” in Figure 4.37, the process gain Kp, was increased by a factor of 10 to see if the monitor would be able to detect any changes in the performance of the controller, given a different process parameter than the one used for tuning. Notice the change in performance during the period in the region labeled “B”. Once control limits were continuously violated beyond the grace period, a flag was raised to indicate the degrading controller performance. Notice that at some instances, the monitor reset the violation counter to zero due perhaps to the fact that the controller was attempting to restore good performance and the monitor detected that, but once performance deteriorated again the counter was invoked to start counting and a flag was raised once the grace period was exceeded. At sampling 34,000, the process gain was reset to its original value. Flagging stopped after the monitor sampled about a window length of data and detected return to good control.

Between sampling 40,000 and 46,000, labeled “C”, the process time constant was decreased by a factor of 0.1. Again, notice the change in the controllers performance during that period. Initially it appears that the controllers’ performance did not deteriorate much as to violate control limits except for occasional inception of the violation counter and resetting back to zero. However, between sampling 43500 and about 44800, control limits were violated and so the monitor started counting and flagged after the grace period was exceeded and the violations were still present in the loop. After sampling 44800, the violation counter was reset to zero. It seems that controller performance was viewed by the monitor to be acceptable around that period but shortly after that, around sampling 45000 control limits were violated again. The monitor detected the control limit violations, started counting and flagged retroactively again after

the grace period was exceeded and the violations were still present in the loop. The process gain was restored to normal mode at sampling 46,000. The monitor detected the change in loop performance and stopped flagging. It must be mentioned that, it is possible that instances when the violation counter was reset to zero when there were still nuances in the loop could also be due to alarms being missed (Type-II errors).

4.4.3 Simulation Using Model Predictive Control

This last simulation discussed in this work involves the application of the health monitor on a process controlled using Model Predictive Control (MPC) technique. Figure 4.38 is a schematic illustration of an MPC control loop. The dynamics of the process is assumed to be inverse acting and second-order plus time delay. It is represented by $G_p = \frac{(-\lambda s+1)e^{-\theta s}}{(t_{p1}s+1)(t_{p2}s+1)}$. A step response model used for determining the dynamic matrix was obtained after introducing a step change in the controller output signal and saving the process output over the entire prediction horizon. Details of the process parameters and descriptions are provided in Appendix E.

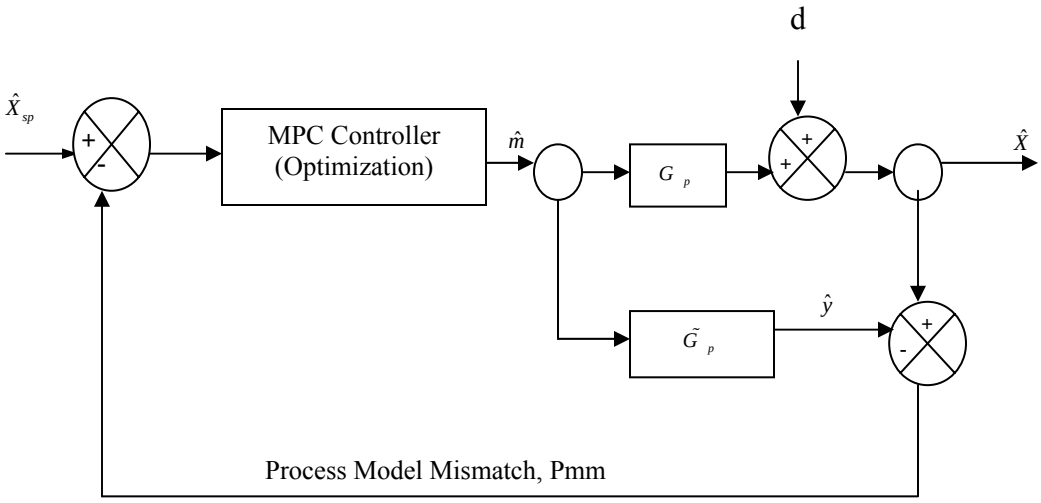


Figure 4.38 Schematic Diagram of a Process Controlled with MPC Technique (e = Actuating Error = Process Model Mismatch (Residuals), X = Controlled Variable, m = Manipulated Variable, $\tilde{G}_p$ = Step Response Model, G_p = Process, $\hat{X}_{sp}$ = Desired Trajectory (Setpoint), d = Disturbance)

Data collected during a period of good control was analyzed and is shown in Figure 4.39. The sampling time interval for the controller was 0.25 time unit. The closed loop settling time (CLST) was estimated through step test to be about 30 samples. The algorithm estimated that a sampling ratio of 1 (i.e. health monitor sampling time interval = 0.25 time unit) and a window length of 405 samples to be ideal for test analysis.

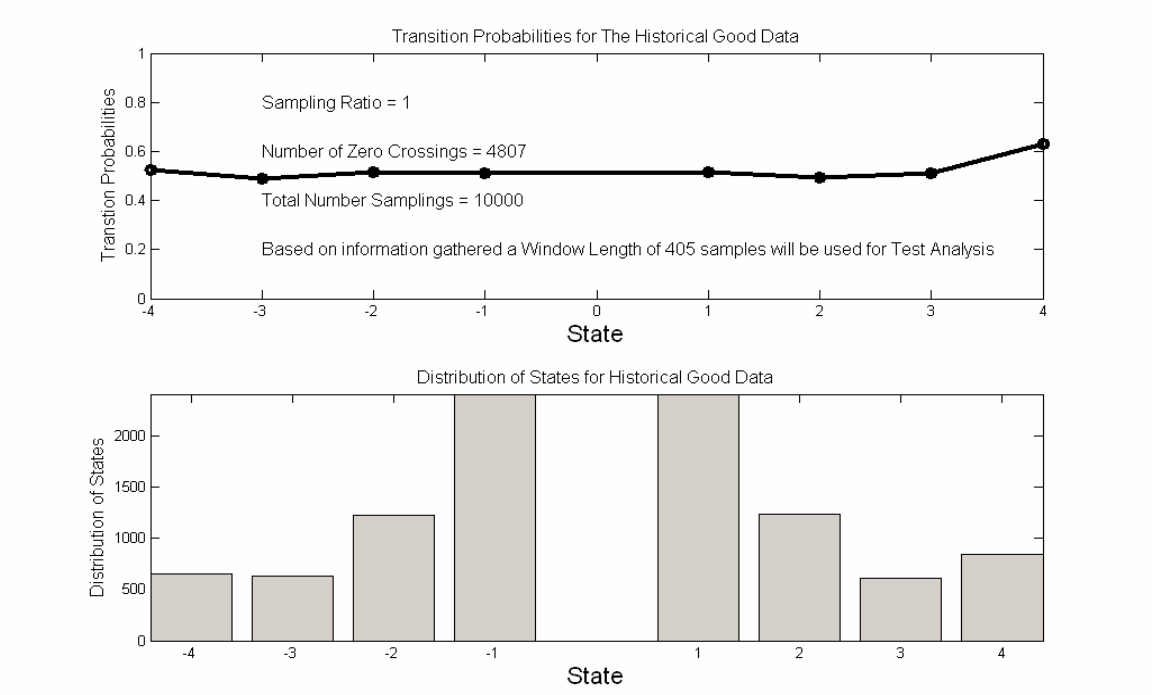


Figure 4.39 Distribution of States and Transition Probabilities from Reference Good Data (Window length = 405 Samples; Sampling Ratio = 1)

The monitor was initialized with 8 total states (± 4 on each side), and the algorithm determined that it was enough to meet the requirement of having no more than 10 % of the data in the extreme states. Prior to estimating the window length, the algorithm selects the state with least number of samples on each half of the Figure 4.39

and determines the number of samples to place in those states in order to meet the requirement on Type-I and Type-II error. For this example, it can be observed from Figure 4.39 that the states that have the least number of samples are the +3 state on the positive side and the –3 state on the negative side. Only one of the two states (in this example +3 state) is chosen and revealed in Figure 4.40. The algorithm estimated the number of samples that need to visit this state (+3 state) to be 26 given choices of α and β and the how far way from the reference transition probability that this β is desired. For this state, number of samples denoting the lower control limit is $X_L = 5$ and the number of samples denoting the upper control limit is $X_H = 20$. Thus, out of 26 samples

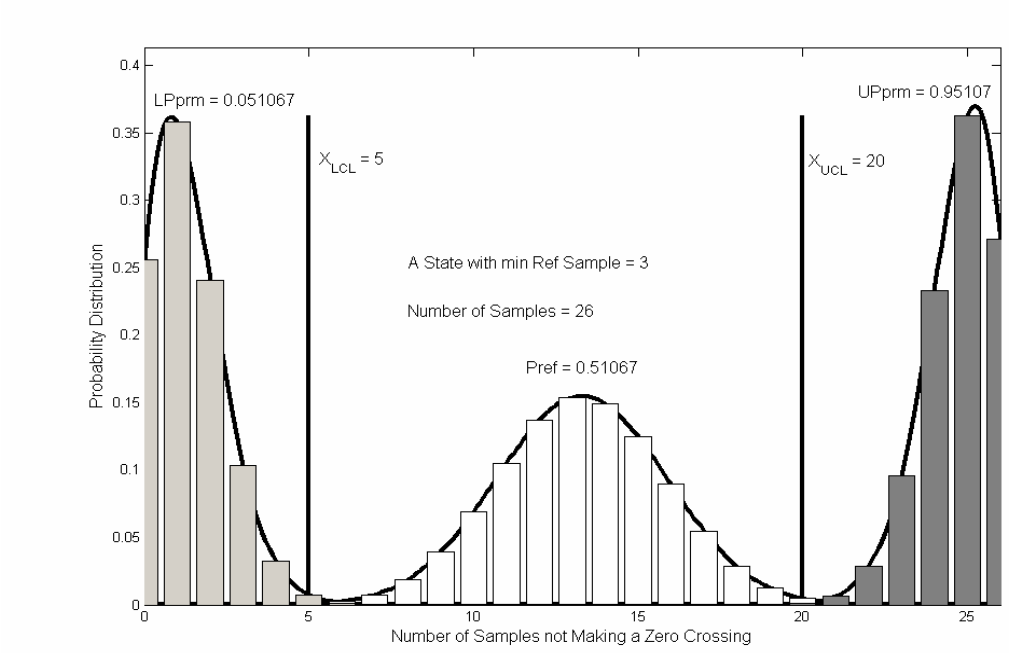


Figure 4.40 Analysis of Reference Data for State with Least Number of Samples (Distribution of Samples Leaving State and Entering the Next Absolute Higher State Given the Reference Probability, α, β and λ)

that need to visit the state of +3 in a given window (to be determined), a violation will occur if fewer than 5 samples leave the state and visit the next absolute higher state (i.e. -4) or more than 20 samples leave the state and visit the next absolute higher state.

Furthermore, Figure 4.40 illustrates that if future transition probabilities are not different from the reference value of 0.51067, then for example, 15% of the time, say, about 13 samples will leave the state of +3 and visit the state of +4. Also, if future transition probabilities were to change significantly to say 0.95107 then about 23% of the time, 24 samples may be expected to leave the state of +3 and visit the state +4 and when it happens, this will be outside the desired upper control limit of 20 samples and so a violation will occur. Moreover, if future transition probabilities were to change significantly to say 0.051067, then about 24% of the time only 2 samples may be expected to leave the state of +3 and visit the state of +4 and since this is outside the lower control limit of 5 samples, a violation will occur. Notice from Figure 4.40 that the chance of missing a violation is negligibly small, as desired.

After the analysis on this state is complete and the number of samples in all the other states are determined, the algorithm selects one state at random and determines the statistical errors that are associated with it based on the choices made. For this example, the algorithm selected the state of +1 and based on the reference transition probability, determined that 99 samples need to visit the state of +1 as shown in Figure 4.41. Figure 4.41 reveals that if future transition probability where to differ from the reference value to say 0.051331, then about 15% of the time say, about 6 samples may be expected to leave the state of +1 and visit the state of +2. Furthermore, if transition probabilities differ to about 0.95133, then about 10% of the time 92 samples may be expected to leave

the state of +1 and visit the state of +2. In both cases, these numbers of visits are outside the control limits 34 on the lower side and 66 on the upper side and so it will mean a violation of the control limits. Once all samples that need to visit all the states during a period of good control are known, the algorithm estimates the window length necessary for performance monitoring. Once the window length is estimated, a power curve for the entire test is plotted. This is shown in Figure 4.42.

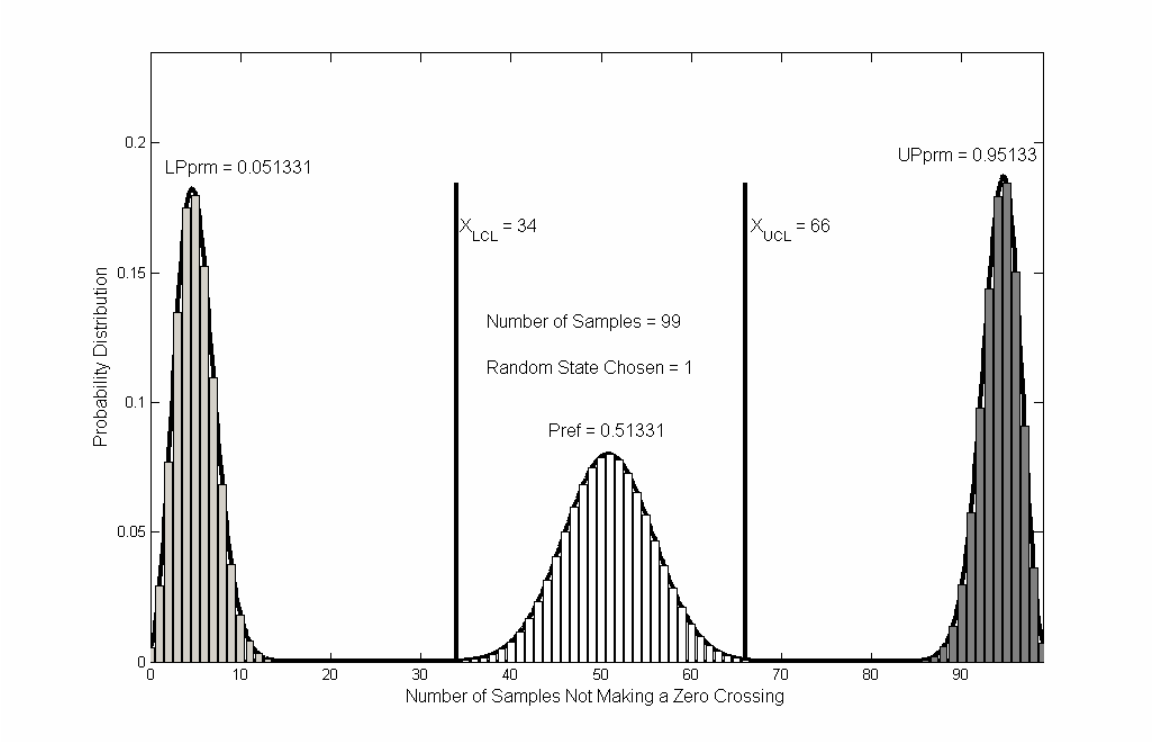


Figure 4.41 **Analysis of Reference Data for State Chosen at Random** (Distribution of Samples Leaving State and Entering the Next Absolute Higher State Given the Reference Probability, α, β and λ)

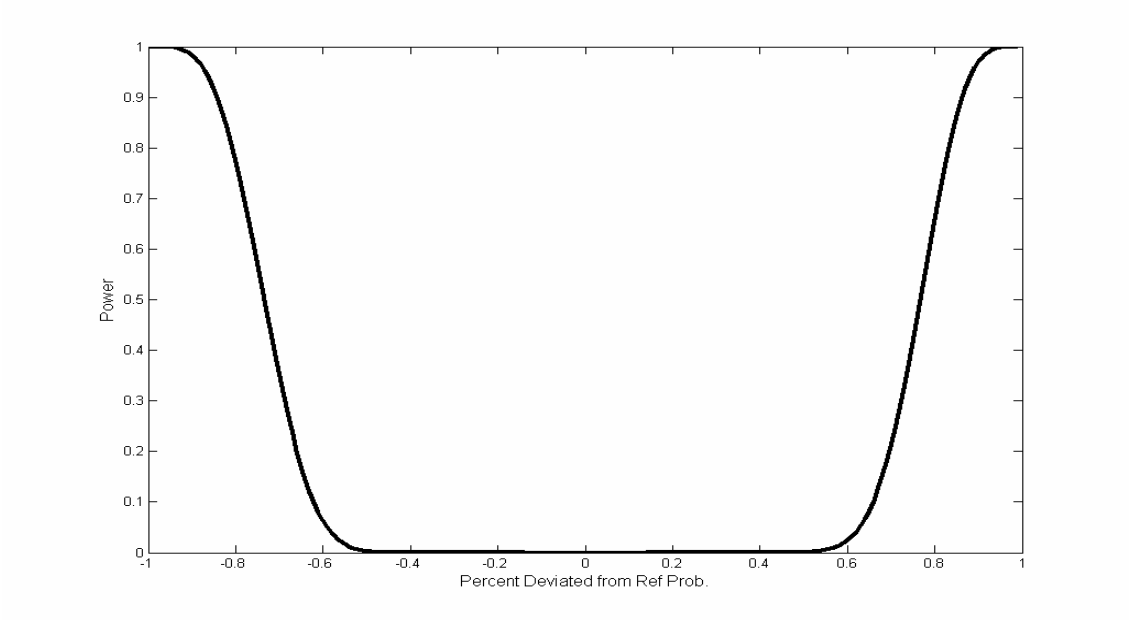


Figure 4.42 Power Curve (Given α = 1% and β= 1% λ = 90%)

4.4.4 Performance Evaluation of Health Monitor on the MPC Process

The performance of the health monitor is illustrated in Figure 4.43. At sampling 2000, 4000 and 8000, setpoint changes were made. The original Setpoint used during reference data sampling was 50. At sampling 2000, it was reduced to 30 and then doubled at sampling 4000 to 60 and then finally restored back to 50 at sampling 8000. Notice that the setpoint changes at sampling 2000 and 4000 did result in control limit violations and so the violation counter was not invoked but the setpoint change at sample 8000 destabilized the loop some how resulting in control limit violations, but this was no significant problem for the MPC controller to overcome. The controller placed

controlled variable at setpoint rapidly within the grace period and the violations went off, and the counter was reset to zero

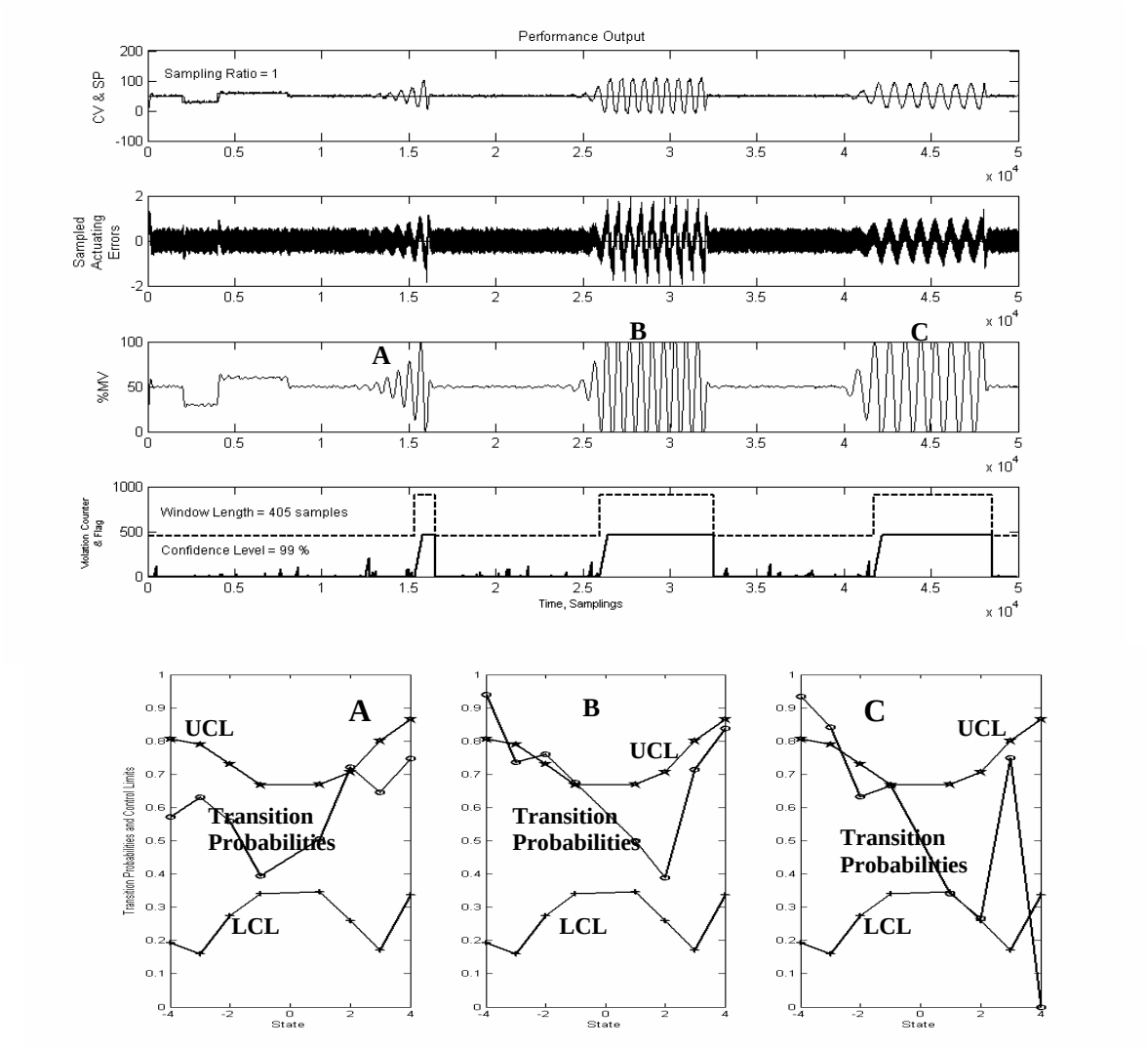


Figure 4.43 **Control Loop Performance Output for MPC** (Sampling Interval = 0.25 Time unit, Sampling Ratio = 1; Window Length = 405 Samples, Startup Period = 0 Samples, Grace Period 435 Samples, Violation Counter Trigger = Length of Grace Period +1, Overall Level of Significance (α_T) = 1%). Performance Output (A = Change in Process Delay, B = Change in Process Delay and Process Zeros, C = Change in Process Delay and Process Poles)

Between sampling 12,000 and sampling 16,000 and in the region labeled “A” in Figure 4.43, the delay associated with the process was changed from the original value of 2 to 15. Notice the resulting oscillations in the loop. This resulted in control limit violations. The monitor started the violation counter and after the grace period was exceeded, a flag was raised retroactively to indicate that the controller was having difficulty stabilizing conditions in the loop. At sampling 16,000, the delay was restored to the original value of 2. The violation counter was reset to zero and the flagging went off.

Between sampling 24,000 and 32,000, in the region labeled “B” in Figure 4.43, the process zero was changed from 1/3 to 1/12 and the delay again changed from 2 to 15. Notice the ensuing oscillations in that region. The monitor detected this and initialized the violation. After the grace period was exceeded and the violations were still present, the monitor flagged retroactively to indicate the start of the problem. At sampling 32,000, the delay was restored to 2 and process zero reset to the original value of 1/3. Again, the violation counter was reset to zero and the flagging went off.

Between sampling 40,000 and 48,000, labeled “C”, one of the process poles was changed from the original value -1/2 to –1/30 and the delay changed to 15 again. This again resulted in oscillations. The monitor detected the violations and after the nuances had persisted in the loop beyond the grace period, a flag was raised. At sampling 48,000 when the process pole and delay were restored to their original values of –1/2 and 2, notice that the violation counter was reset to zero and the flagging went off.

Summary

A simple technique to detect and flag degrading control loop performance has been developed. The technique uses normal plant operating data and a few user desired parameters. These include, the desired Type-I and Type-II error rates and the how far away from a reference transition probability that the Type-II error rate is desired. The algorithm is automated, but it must be initialized with a sampling ratio (SR) and a certain number of states (NS). The monitor requires data from a reference period when control was judged to be good by operators or engineers.

In the selection of the reference data, it is necessary to ensure that it includes incidents that are expected to be seen and not flagged as "bad" events.

The monitor uses this data to determine the transition probabilities and control limits. Ideally, it will be desired that good control will be maintained throughout plant operation. Nevertheless, process and plant conditions change and control is no longer good as was desired. The monitor continually determines the transition probabilities of this data at each sampling instant and if future probabilities differ significantly from that determined during the reference period as to violate any control limit, then a violation counter is initialized. If the violations persist beyond a grace period then a flag is raised retroactively to indicate the start of the problem.

For this work, the overall Type-I error rate (α) used was 1%. The Type-II error rate (β) was also set at 1%. The measure of how far way from the reference probability (P_{ref}), that the Type-II error rate (λ) was desired, was set at 90%. The algorithm was initialized with a sampling ratio of 1 and initial total number of states of 8.

The monitor was evaluated on simulation data, unit operation laboratory ((UOL) experimental data, and industrial data from one of the industrial sponsors for the project, ExxonMobil.

The results show that the monitor is able to detect poor and degrading control performance. For instance, it was able to detect instances of valve stiction, and oscillations caused by external events, changing process conditions such as gain increases or decreases, time delays and poles and zeros. Moreover, even in instances where changes in process conditions resulting in degrading control performance are not visible to the normal human perception, the monitor is able to detect when the distribution of data being analyzed is significantly different from that analyzed during the reference period and inform operators. The monitor flags continuously for as long as a problem detected persists. It does not flag when process conditions are good.

In this work, the amplitude of a signal is ignored in characterizing the actuating errors. The question has arisen as to whether this is a limitation or not. It must be stated that if an incident changes the amplitude of a signal such that the Autoregressive Moving Average (ARMA) description is not affected, then perhaps the monitor may be unable to detect such changes. On the other hand, if an event such as the process gain, process time-constant, controller gain and controller time-constant (See Appendix G), changes the amplitude of a signal, it will affect the natural frequency of oscillation of the signal and hence the natural period of oscillation. This will lead to differences in the period of oscillation of the data sampled in the reference stage and the period of oscillation of the data sampled during testing or performance monitoring. For such cases, the health monitor will detect the changes and flag when it has to.

CHAPTER 5

5.0 Conclusion and Recommendations

A simple and practicable technique capable of flagging poor and degrading control loop performance has been developed. The conclusions are summarized in section 5.1 and a few recommendations for future work are presented in section 5.2.

5.1 Conclusions

A control loop monitoring software is hereby proposed. The proposed technique is able to detect and flag poor and degrading control loop performance. The method uses only routine plant operating data.

It does not require *a-priori* knowledge of the process (such as model or process deadtimes and delays) nor the controller. It only requires the process setpoint and a representative data of the controlled variable during a period of good control. For model-based control such as model predictive control (MPC), it uses the residuals or process model mismatch for analysis.

The method herein proposed provides a helpful autonomous tool for technicians and engineers in monitoring control loop performance more efficiently. This will help reduce or avoid erroneous human decision.

Simulations, unit operations, and industrial data testing, reveal that the health monitor can automatically detect and flag common control loop problems such as

oscillations, aggressive and sluggish control, and other constraints that degrade process conditions.

The technique is simple and easy to implement. It only requires a few user specified parameters. Once the required number of user-parameters is specified, the monitor automatically determines all other parameters needed for performance monitoring.

The monitor flags when during period of degrading or poor control. It does not flag when control loop performance is good.

The technique has been developed to balance user desired Type-I and Type-II error rates in estimating the window length. As a result, when the violation counter is initialized, it is a likely indication of degrading control loop performance and may require operators to pay attention or investigate what the cause might be. Furthermore, when a grace period is exceeded and a flag is raised, then it is more than likely that something has gone wrong within the loop requiring instant operator attention or response.

Although, the technique has been developed using single input single output (SISO) system, this does not in anyway impair its application or extension to multiple input multiple output (MIMO) systems. It is easily extensible to all applications where a target value can be compared to an output value.

The technique is easy to understand and implement. It is non-intrusive in that it does temper with control loop activities. It only samples data from the controller. Nevertheless, it is able to detect oscillations that are even not perceptible to the human eye. It flags on when it is supposed to flag, and off when it is not supposed to. This makes it sensitive and efficient.

5.2 Recommendations

Although the method flags on poor or degrading control it does not indicate the location of the problem that caused the nuance in the loop. The objective of this work did not include identifying fault or problem location once the monitor detects a problem. However, it will be ideal to extend the application of the monitor to enable it report to operators the location of a problem once it detects bad or degrading control loop performance. Such a utility will enhance process safety and reduce time spent by operators in attempting to identify the source of a problem in a complex plant with numerous interacting loops.

The technique has been tested extensively on unit operations and simulation data and found to be internally consistent. Though the monitor was tested on one set of industrial data, it will be nice to test it on more external data from industry in order to claim external consistency as well.

This work utilized two arbitrary conditions:

1. Desiring that transition probabilities lie between 0.25 and 0.75 to avoid getting controls limits that are zero, unity or very nearly so.
2. Desiring that at most 10% of the data lie in the extreme state during analysis of the reference data.

While these conditions appeared to work fine for all the analysis explored in this study, no fundamental proof has been established for their usage. It is recommended that further work be carried out to investigate a fundamental basis for theses choices or otherwise.

The normal approximation to the binomial distribution is often used as a simple and easy way for estimating distributions involving binomially distributed variables. This work discovered differences in the use of the two distributions. For rigorous usage and all applications involving this work, the binomial distribution is recommended.

It is recommended that further work be carried to investigate how amplitude changes that do not affect the natural frequency oscillation of a signal will affect the performance of the monitor.

An attempt was made to identify the effect of autocorrelation on sampling ratio in this work. While it was noticed that autocorrelation does have an impact on the sampling ratio, the results may not be adequately conclusive. It will be helpful to explore this effect further to identify how strong effects of autocorrelation actually affect the performance of the monitor.

REFERENCES

Ingalls R., "Private Communication", 2002

Huang, B., Shah, S. L., Kwok, E. K., "Good, Bad, or optimal? Performance Assessment of Multivariable Processes", Automatica Vol. 33. No. 6, pp 1175-1183 1997

Shah S. L., Patwardhan, R., Huang, B., "Multivariate Controller performance Analysis Methods, Approach, Applications and Challenges", Proceedings of the 6th International Conference on Chemical Process, AIChE Symposium series No. 326 Vol. 98, pp190-207, Tucson, Arizona, 2001

Desborough L., Miller R., "Increasing Customer value of Industrial Control Performance Monitoring - Honeywell's Experience", Proceedings of the 6th International Conference on Chemical Process, AIChE Symposium series No. 326 Vol. 98, pp190-207, Tucson, Arizona, 2001

Schäfer J., Cinar A., "Multivariable MPC system performance assessment, monitoring and diagnosis", Journal of Process Control, 14, pp 1-17, 2004

Anderson T.W., Sclove S. L., "Introductory Statistical Analysis", Houghton Mifflin Company, Boston, USA, pp 382, 1974

Hugo, J, A., "Process Controller Performance Monitoring and Assessment", Control Arts Inc., 2000

Ross M. S., "Introduction to Probability Models", Academic Press, 8th Ed., NY, 2003

Montgomery D. C., Runger G. C., "Applied Statistics and Probability for Engineers", John Wiley and Sons Inc., NY, 1994

Devore J. L., "Probability and Statistics for Engineering and the Sciences", 4th Ed., Duxbury Press, NY, 1995

Mendenhall W., Reinmuth J. E., " Statistics for Management and Economics", 4th Ed., Duxbury Press, Boston, 1982

N.I.S.T., "http://www.itl.nist.gov/div898/handbook/", 2005

Box G., Lugeño A., “Statistical Control By Monitoring and feedback Adjustment”, John Wiley and Sons Inc., NY, 1997

Merrit R. “ Some Process Plants are out of Control”, Control for the Process Industries, pp 30, November 2003

Box G. E. P, Jenkins G. M., Reinsel G. C., “Time Series Analysis Forecasting and Control”, 3rd, Prentice Hall, New Jersey, 1994

Horch A., “Condition Monitoring of Control Loops”, Stockholm, 2000

Keller P., “Six Sigma Demystified: A self-Teaching Guide” , McGraw-Hill, New York, 2005

Shumway R. H., Stoffer D. S., “Time Series Analysis and Its Applications”, Springer-Verlag, New York, 2000

Wei W. W. S., “Time Series Analysis Univariate and Multivariate Methods”, Addison-Wesley Publishing Company, Inc., New York, 1989

Lee P.L., Newel R. B., Cameron I. T., “ Process Control Management”, Blakie Academic And Professional” New York, 1998

Coleman B., Babu J., “ Techniquies of Model Based Control”, Prentice Hall, 2002

Li, Q., Whiteley, J. R., and Rhinehart, R. R., “An Automated Performance Monitor for Process Controllers”, Control Engineering Practice, 2004

Rhinehart, R. R., “A Watch Dog for Controller Performance Monitoring”, Proceedings of the 1995 American Control Conference. Seattle, WA. 1995

Ralston, P., Depuy, G., and Graham, J. H., “Computer-Based Monitoring and Fault Diagnosis: A Chemical Process case study”, ISA Transactions, 40, pp 85-98, 2001

Tamir, A., “Applications of Markov Chains in Chemical Engineering”, Elsevier Science B.V., NY 1998

Tatara, E., and Cinar, A., “An Intelligent System for Multivariate Statistical Process Monitoring and Diagnosis”, ISA Transactions, 41, 225-270, 2002

Harris, T. J., Seppala, C. T., and Jofriet, P. J., “Recent Developments in Controller Performance Monitoring and Assessment Techniques” AIChE Symposium Series 326, 1998

Kadali, R., and Huang, B., “Controller Performance Analysis with LQG Benchmark Obtained Under Closed Loop Conditions”, ISA Transactions, 41, pp 512-532, 2002

Mann, P. S. “Introductory statistics”, 2nd, John Wiley & Sons, 1995

Serborg D. E., Edgar T. F., Mellichamp D.A., “Process Dynamics and Control”, 2nd, John Wiley and Sons, Inc., New York, 2004

Erickson K. T., Hendrick J. L., “ Plantide Process Control”, John Wiley and Sons, Inc., New York, 1999

Bethea R. M., Rhinehart R. R., “Applied Engineering Statistics”, Marcel Dekker, Inc., 1991

Freund R. J., Wilson W. J., “Statistical Methods” 2nd, Academic Press, New York, 2003

O’dwyer A., “ PI ne PID Controller Tunning Rules”, World Scientific, New Jersey, 2003

Ingham J.,Dunn I., J., Heinzle E., Prenosil J. E., “Chemical Engineering Dynamics Modeling with PC Simulation”, New York, 1994

Babatunde A. O., Ray W. H., “Process Dynamics, Modeling and Control”, Oxford University Press, New York 1994

Levine W. S., “The Control Handbook”, CRC Press, 1996

Duarte-Mernoud M. A., Prieto R. A., “ performance Index for Quality response of Dynamic Systems”, ISA Transactions 43, pp 133-151, 2004

Brockwell P. J., Davis R. A., “Introduction to Time Series and Forecasting”, Springer Verlag, New York, 2002

Häggstrom O., “Finite Markov Chains and Algorithmic Applications”, Cambridge University Press, U.K., 2002

Norris J.R., “Markov Chains”, Cambridge University Press, U.K, 1997

Morari M., Zafiriou E., “Robust Process Control”, Prentice Hall, New Jersey, 1989

Stephanopoulos G., “Chemical Process Control: An Introduction to Theory and Practice”, Prentice Hall, 1984

Gerry J., “Process Monitoring and Loop Prioritization can Reap Big Payback and Benefit Process Plants”, Paper Presented: ISA 2002, Chicago, IL, 2002

Gerry J., “Establishing a Basis for Performance”, White Paper, www.expertune.com, 2004

Kinney T., “Choosing Performance Assessments”, White Paper, www.expertune.com, 2004

Finner H., Roters M., “ Multiple Hypothesis Testing and Expected Number of Type I Errors”, Annals of Statistics, 30(1), pp 220-238, 2002

Ko B., Edgar T. F., “Performance Assessment of Multivariable Feedback Control Systems”, Automatica, 37, pp 899-905, 2001

Ruel M., “Stiction: The Hidden Menace”, White Paper, http://www.expertune.com, 2000

Gerry J., “Performance Measurement - The Rest of the Story”, White Paper, www.expertune.com, 2004

Ruel M., “A Simple Method to Determine Control Valve Performance and its Impacts on Control Loop Performance”, White Paper, http://www.topcontrol.com, 2000

McNabb C. A., Qin J S., “Projection Based MIMO Control Performance monitoring: I-Covariance Monitoring in State Space”, Journal of Process Control, 13, pp739-459, 2003

Qin S. J., “Control Performance Monitoring – A Review and Assessment”, Computers and Chemical Engineering, 23, pp 173 – 186, 1998

Doraiswami R., Jiang J., “Performance Monitoring in Expert Control Systems”, Automatica, 25(6), pp 799-811, 1989

Knegtering B., Brombacher A. C., “A Method to Prevent Excessive Numbers of Markov States in Markov Models for Quantitative Safety and Reliability”, ISA Transactions, 39, pp 363-369, 2000

Venkatasubramanian V., “http://molecule.ecn.purdue.edu/~lips/research.html”, 2003

http://www.minortriad.com/mref.html#spelling, 2006

http://www.minortriad.com/markoff.html, 2006

The New Encyclopedia Britannica, 7, 15ed, Chicago, 1994

APPENDIX A

Controller Tuning for First-Order-Plus-Time Delay (FOPTD) Process

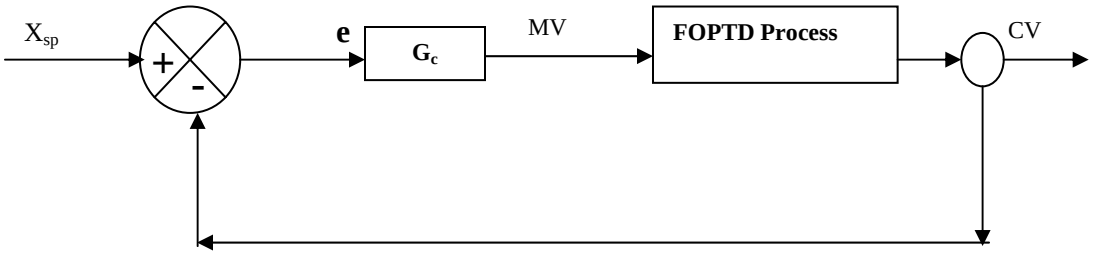


Figure A1 Schematic Diagram of a PID Control Loop for FOPTD Process

Transfer function for processes = $G_p = \frac{K_p e^{-\theta s}}{(t_p s + 1)}$

Kp = Process Gian

t_p = Process time constant

θ = Transport (Time) delay

The following process parameters are used:

$K_{p1} = 15$

$t_{p1} = 2 \text{ min}$

$\theta_1 = 0.1 \text{ min}$

Tuning is done using the ITAE PI controller tuning relations below for the controller gain Kc, the Integral time constant t_I respectively as:

$$K_c = \left(\frac{0.859}{K_p} \right) \left(\frac{t_p}{\theta} \right)^{0.977}$$

$$t_I = \left(\frac{t_p}{0.674} \right) \left(\frac{\theta}{t_p} \right)^{0.680}$$

The controller output is given by

$$U = U_O + K_c \left(e(t) + \frac{1}{t_I} \int_0^t e(t)\, dt \right)$$

Where U_O is the controller bias

APPENDIX B

Controller Tuning for Second Order-Plus-Time Delay (SOPTD) Process

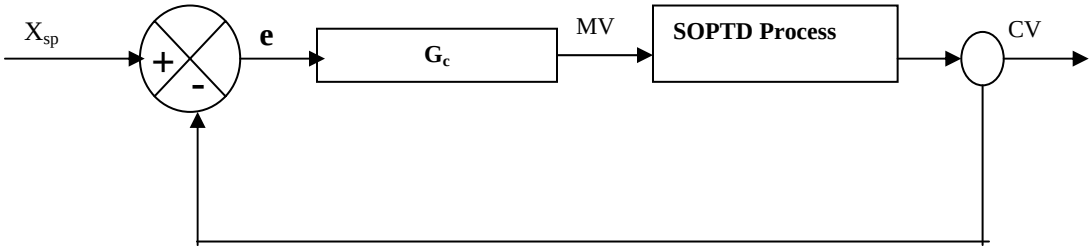


Figure B1 Schematic Diagram of Control Loop for SOPTD Process

$G_p = G_{p1}G_{p2}$ = Combined transfer functions for processes 1 and 2

$G_{p1} = \frac{K_{p1}e^{-\theta_1 s}}{(t_{p1}s+1)}$ = Transfer function for processes 1

$G_{p2} = \frac{K_{p2}e^{-\theta_2 s}}{(t_{p2}s+1)}$ = Transfer function for processes 2

Combining the two transfer functions gives:

$$G_p = \frac{K_{p1}K_{p2}e^{-(\theta_1+\theta_2)s}}{(t_{p1}s+1)(t_{p2}s+1)}$$

Tuning is done using the Cohen-Coon PID controller tuning relations below for the controller gain Kc, the integral time constant, t_I, and the derivative time constant, t_D, respectively as:

$$K_c = \frac{t_p}{K_p \theta}\left(\frac{4}{3} + \frac{\theta}{4 t_p}\right); \qquad t_I = \theta\left(\frac{32 + 6\theta/t_p}{13 + 8\theta/t_p}\right); \qquad t_D = \frac{4\theta}{\left(11 + 2\theta/t_p\right)}$$

The controller output is given by

$$U = U_O + K_c\left(e(t) + \frac{1}{t_I}\int_0^t e(t)\,dt + t_D\frac{de}{dt}\right)$$

Where U_O is the controller bias

The following process parameters are used:

$K_{p1} = 0.5$; $t_{p1} = 1.5\ \text{min}$; $\theta_1 = 0.5\ \text{min}$

$K_{p2} = 1.0$; $t_{p2} = 1.2\ \text{min}$; $\theta_2 = 0.3\ \text{min}$

APPENDIX C

Internal Model Control (IMC) Structure

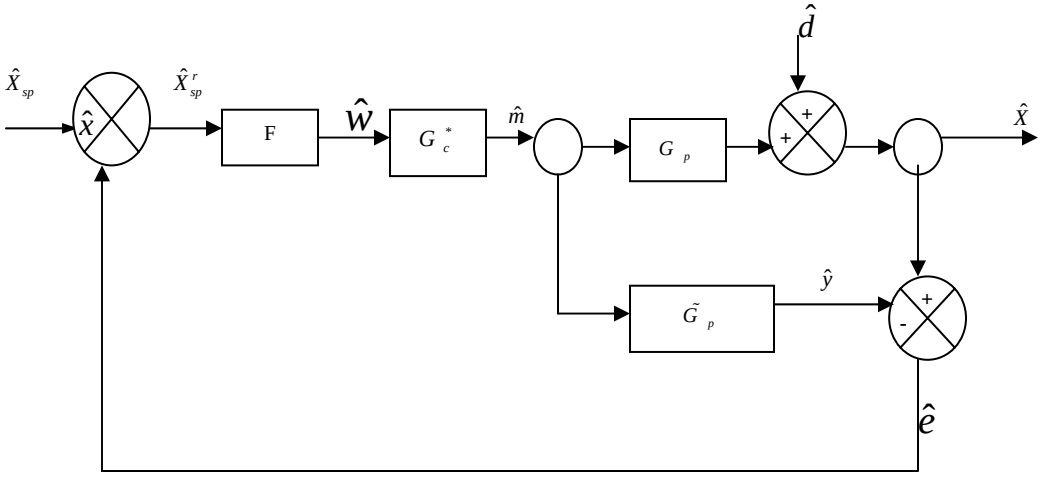


Figure C1 Schematic Diagram of Control Loop with a Process Model (IMC)

$\tilde{G}_p$ = Process

G^*_c = Internal Model Controller

$\tilde{G}_p$ = Process Model = $\tilde{G}_+ \tilde{G}_-$

$\tilde{G}_+$ = Non-invertable part of process model (Contains all time delays and right-half plane poles

$\tilde{G}_-$ = Invertable part of process model

F = Filter = $\dfrac{1}{(t_w s + 1)^n}$

n = filter order

τ_w = filter time-constant

$\hat{e}$ = Mismatch between process and process model = $\hat{X} - \hat{y}$

$\hat{d}$ = Disturbance to the process

$\hat{X}_{sp}$ = External setpoint to the process

$\hat{X}^r_{sp}$ = Corrected setpoint to the process = $\hat{X}_{sp} - \hat{e}$

$\hat{w}$ = Filter output

$\hat{m}$ = Controller Output

Figure C1 can be rearranged to get the equivalent structure in Figure C2 for the entire structure in the area marked with the broken lines represents the IMC controller shown in simplified form in Figure C3.

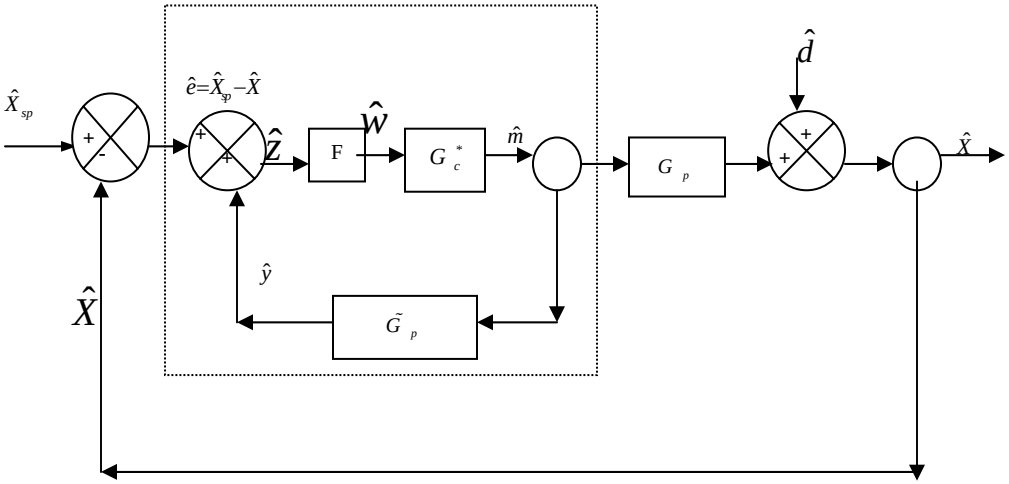


Figure C2 Equivalent Structure of Figure C1

From Figure C2,

$$\hat{e} = \hat{X}_{sp} - \hat{X}$$
$$\hat{y} = \hat{G}_p \hat{m}$$
$$\hat{z} = \hat{e} + \hat{y}$$
$$\hat{w} = F\hat{z}$$
$$\hat{m} = G_C^* \hat{w}$$

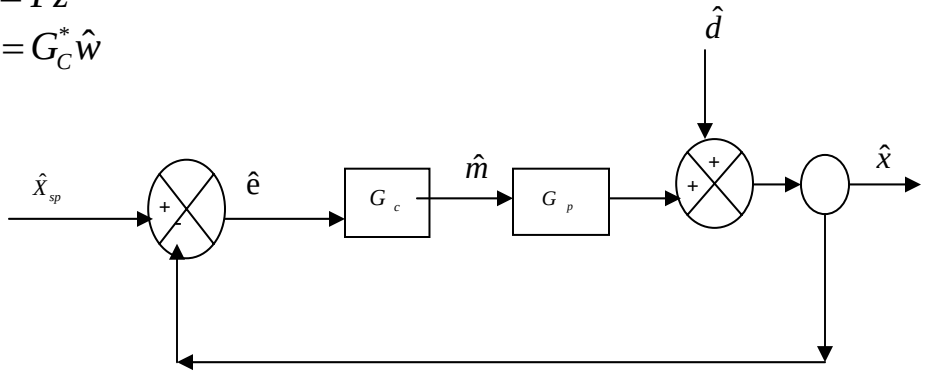


Figure C3 Simplified Structure of Figure C1

G_C^* is chosen as the reciprocal of the invertable part of G_p (i.e. $1/G_-$). Hence given $G_p = \frac{K_p e^{-\theta s}}{(t_p s + 1)}$, it implies that $G_C^* = \frac{1}{G_-}$ where $G_- = \frac{K_p}{(t_p s + 1)}$ and $G_+ = e^{-\theta s}$

Hence $G_C^* = \frac{(t_p s + 1)}{K_p}$

Tunable parameters include: t_p, K_p, t_w and n.

Since G_c^* is typically improper, the filter order, n is chosen so that $F * G_c^*$ is proper. The following process parameters are chosen from lecture notes by Dr. R. R. Rhinehart. $t_p = 4.3 \text{ min}$; $K_p = 0.8\%/psi$; $\theta = 2.1 \text{min}$. Fro these, Rhinehart recommends t_w of 3 min (i. e. $(t_w + \theta)/2$), and "n" is chosen to be 1.

$$G_- = \frac{0.8}{(4.3s+1)}; \; G_c^* = \frac{(4.3s+1)}{0.8}; \qquad G_+ = e^{-2.1s}; \qquad F = \frac{1}{(3s+1)}$$

A simple algorithm for the IMC loop therefore is:

$$e = X_{sp} - X$$

$$y_{new} = \left(\frac{\Delta T}{4.3}\right) * 0.8 * \left[m\,(t - 2.1)\right] + \left(1 - \frac{\Delta T}{4.3}\right) y_{old}$$

$$y_{old} = y_{new}$$

$$z = e + y_{new}$$

$$w_{new} = \left(\frac{\Delta T}{3}\right) z + \left(1 - \frac{\Delta T}{3}\right) w_{old}$$

$$m = 1.25w + 5.375\left(\frac{w_{new} - w_{old}}{\Delta T}\right)$$

$$w_{old} = w_{new}$$

Output m

APPENDIX D

AUTO-CORRELATION AND PARTIAL AUTO-CORRELATION ANALYSIS

Given a time series of data, it is often useful to investigate if there is a relationship or correlation between the data points. Investigation of the correlation between different values or signals within a data set is referred to as autocorrelation. In this excursion of the work, it is desired to study how the presence of autocorrelation affects sampling ratio and number of states used for performance monitoring. A study of autocorrelation often involves plotting a correlogram (i.e. a set of graphs showing the correlations in the data as a function of the lag). In this work, the intent here is to investigate if the noise (i.e. actuating errors in this work) pattern:

1. Is just random or not
2. If not, is there a defined pattern or model that describes the errors
3. And, if there is a defined pattern present in the actuating errors,
    a. How does the pattern affect the sampling ratio
    b. Can the pattern be removed and
    c. How does removing the pattern affect sampling ratio

In general, the extent to which two random variables vary together (co-vary) is measured by their covariance. If the variation is about different values in the same data set then it is referred to as autocovariance. It is established that the autocovariance between two points does not depend on time itself but on the difference between 2 times.

In signal processing and time series analysis this difference is generally referred to as lag (Box *et al*., 1994; Shumway *et al*., 2000). The autocovariance enables a calculation of the autocorrelation function, which enables one to determine if the noise present in a data set is just random or if a pattern exists. A discussion of how to get the correlogram is presented below

Given time series of data:

$$x_t = \alpha + \phi_1 x_{t-1} + \phi_2 x_{t-2} + \phi_3 x_{t-3} + \ldots + \phi_p x_{t-p} + w_t \quad \text{(D1)}$$

Where $\alpha = \mu\left(1 - \phi_1 - \phi_2 - \phi_3 - \ldots - \phi_p\right)$ (D2)

μ is the sample mean and ϕ_i denotes the coefficient of x_{t-i} and w_t is white Gaussian noise. The covariance between samples or observations at time t and t + h is denoted by:

$$Cov\left(x_t, x_{t+h}\right) = \gamma\left(t, t+h\right) = \gamma\left(h\right) = E\left(x_t - \overline{x}_t\right)\left(x_{t+h} - \overline{x}_{t+h}\right) \quad \text{(D3)}$$

Where $E\left(x\right) = \frac{1}{N}\sum_{i=1}^{N} x$ and $\overline{x} = \mu$ is the sample mean. Hence:

$$\gamma\left(h\right) = \frac{1}{N}\sum_{t=1}^{N}\left(x_t - \overline{x}_t\right)\left(x_{t+h} - \overline{x}_{t+h}\right) \quad \text{(D4)}$$

If the mean $\overline{x}$, variance σ_w and model coefficients ϕ_i are approximately constant over time, then the series is said to be stationary. Thus $\overline{x}_t = \overline{x}_{t+h}$

$$\gamma\left(h\right) = E\left(x_t - \overline{x}_t\right)\left(x_{t+h} - \overline{x}_t\right) \quad \text{(D5)}$$

$$\gamma\left(h\right) = E\left(x_t \cdot x_{t+h} - \overline{x}x_t - \overline{x}x_{t+h} + \overline{x}^2\right) \quad \text{(D6)}$$

$$\gamma\left(h\right) = E\left(x_t \cdot x_{t+h}\right) - E\left(\overline{x}x_t\right) - E\left(\overline{x}x_{t+h}\right) + E\left(\overline{x}^2\right) \quad \text{(D7)}$$

$$\gamma\left(h\right) = E\left(x_t \cdot x_{t+h}\right) - \overline{x}E\left(x_t\right) - \overline{x}E\left(x_{t+h}\right) + E\left(\overline{x}^2\right) \quad \text{(D8)}$$

$$\gamma(h) = E(x_t \cdot x_{t+h}) - \bar{x}^2 = E(x_t \cdot x_{t+h}) - E(x_t)^2 \tag{D9}$$

$$\gamma(h) = \frac{1}{N}\sum_{t=1}^{N-h}(x_t \cdot x_{t+h}) - \bar{x}^2 \tag{D10}$$

if h = 0, then $\gamma(t,t) = E(x_t^2) - E(x_t)^2 = \frac{1}{N}\sum_{t=1}^{N}(x_t^2) - \bar{x}^2$

$$\gamma(0) = \frac{1}{N}\sum_{t=1}^{N}(x_t^2) - \bar{x}^2 \tag{D11}$$

In general, for a given data set, the sample variance is estimated as:

$$\hat{\sigma}_w^2 = \frac{1}{N-1}\sum_{t=1}^{N}(x_t - \bar{x})^2 \tag{D12}$$

$$\hat{\sigma}_w^2 = \frac{1}{N-1}\left(\sum_{t=1}^{N}x_t^2 - 2\bar{x}\sum_{t=1}^{N}x_t + \sum_{t=1}^{N}\bar{x}^2\right) \tag{D13}$$

$$\hat{\sigma}_w^2 = \frac{1}{N-1}\left(\sum_{t=1}^{N}x_t^2 - 2N\bar{x}^2 + N\bar{x}^2\right) \tag{D14}$$

$$\hat{\sigma}_w^2 = \frac{1}{N-1}\left(\sum_{t=1}^{N}x_t^2\right) - \frac{N}{N-1}\bar{x}^2 \tag{D15}$$

But $\lim_{N\to\infty}\left(\frac{N}{N-1}\right) = 1$ and $\lim_{N\to\infty}\left(\frac{1}{N-1}\right) = \frac{1}{N}$, hence

$$\hat{\sigma}_w^2 = \frac{1}{N}\left(\sum_{t=1}^{N}x_t^2\right) - \bar{x}^2 \tag{D16}$$

Thus, Equation (D11) and (D16) are same. Hence $\gamma(0) = Variance(x) = \sigma_w^2$

Thus, in general the autocovariance is given by:

$$\gamma(h) = \begin{cases} \sigma_w^2 & h = 0 \\ \frac{1}{N}\sum_{t=1}^{N-h}(x_t \cdot x_{t+h}) - \bar{x}^2 & h \geq 1 \end{cases} \tag{D17}$$

It is essentially the correlation coefficient between a value x and a time shifted version of itself. Let ρ denote the autocorrelation at lag h. Then

$$\rho(h) = \frac{\gamma(h)}{\gamma(0)} \tag{D18}$$

$$\rho(h) = \frac{\frac{1}{N}\left(\sum_{t=1}^{N-h}(x_t - \bar{x})\cdot(x_{t+h} - \bar{x})\right)}{\frac{1}{N}\left(\sum_{t=1}^{N-h}(x_t - \bar{x})^2\right)} = \frac{\sum_{t=1}^{N-h}(x_t - \bar{x})\cdot(x_{t+h} - \bar{x})}{\sum_{t=1}^{N-h}(x_t - \bar{x})^2} \tag{D19}$$

Both the autocovariance and autocorrelation are even functions. That is:

$\gamma(h) = \gamma(-h)$ and $\rho(h) = \rho(-h)$

It is known that the Autocorrelation function (ACF) of a time series provides a significant amount of information about the order of the dependence when the process is a moving average (MA) Shumway *et al*., (2000). The general MA process can be represented as:

$$x_t = \mu + \varphi_1 w_{t-1} + \varphi_2 w_{t-2} + \varphi_3 w_{t-3} + \ldots + \varphi_q w_{t-q}$$

Where w_t is independently and identically distributed with mean zero and variance σ^2_w $\left(i.e. w_t \approx IID\left(0, \sigma_w^2\right)\right)$

$$x_t = \mu + \sum_{k=0}^{q} \varphi_k w_{t-k} \tag{D20}$$

$$x_{t+h} = \mu + \varphi_1 w_{t+h-1} + \varphi_2 w_{t+h-2} + \varphi_3 w_{t+h-3} + \ldots + \varphi_q w_{t+h-q}$$

$$x_{t+h} = \mu + \sum_{j=0}^{q} \varphi_j w_{t+h-j} \tag{D21}$$

Where q is the maximum lag beyond which there is no significant correlation between observations and h is any lag between zero and q inclusive. The covariance is given by

$$\gamma(h) = E\left(x_{t+h} - \mu\right)\left(x_t - \mu\right) \tag{D22}$$

$$\gamma(h) = E\left(\sum_{j=0}^{q} \varphi_j w_{t+h-j}\right)\left(\sum_{k=0}^{q} \varphi_k w_{t-k}\right) \tag{D23}$$

$$\gamma(h) = E\left(\varphi_0 w_{t+h} + \varphi_1 w_{t+h-1} + \ldots + \varphi_q w_{t+h-q}\right)\left(\varphi_0 w_t + \varphi_1 w_{t-1} + \ldots + \varphi_q w_{t-q}\right) \tag{D24}$$

In general,

$$E\left(w_i, w_j\right) = \begin{cases} \sigma_w^2 & i = j \\ 0 & i \neq j \end{cases} \tag{D25}$$

Hence from Equation (D24)

$$\gamma(h) = \begin{cases} \sigma_w^2 \sum_{i=0}^{q-h} \left(\varphi_i \varphi_{i+h}\right) & 0 \leq h \leq q \\ 0 & h > q \end{cases} \tag{D26}$$

Conventionally, φ_0 is often assumed to be 1. Hence from Equation (D26),

$$\gamma(h) = \begin{cases} \sigma_w^2\left(1 + \varphi_1^2 + \varphi_2^2 + \varphi_3^2 \ldots + \varphi_q^2\right) & h = 0 \\ \sigma_w^2\left(\varphi_1 + \varphi_1\varphi_2 + \varphi_2\varphi_3 \ldots + \varphi_{q-1}\varphi_q\right) & h = 1 \\ \sigma_w^2\left(\varphi_2 + \varphi_1\varphi_3 + \varphi_2\varphi_4 \ldots + \varphi_{q-2}\varphi_q\right) & h = 2 \\ \vdots & \\ \sigma_w^2\left(\varphi_h + \varphi_1\varphi_{h+1} + \varphi_2\varphi_{h+2} \ldots + \varphi_{q-h}\varphi_q\right) & h = q \\ 0 & h > q \end{cases} \tag{D27}$$

Using $\rho(h) = \dfrac{\gamma(h)}{\gamma(0)}$

$$\rho(h)=\begin{cases}\dfrac{\sum_{j=0}^{q-h}\varphi_j\varphi_{j+h}}{1+\sum_{k=0}^{q}\varphi_k^2} & 1\le h\le q\\ 0 & h>q\end{cases} \tag{D28}$$

Thus in general, if a series can be modeled as an MA process, then the ACF cuts off ($\rho(h)\cong 0$) after lag *h*. Thus if the ACF plot in the correlogram cuts off after lag h, the process can be represented by an MA model. However, if the series is autoregressive (AR) where:

$x_t=\mu+\phi_1x_{t-1}+\phi_2x_{t-2}+\phi_3x_{t-3}+...+\phi_px_{t-p}+w_t$, or a combination of autoregressive and moving average (i.e. ARMA), then the ACF reveals very little about the order of dependence between the sample data points.

For instance, consider the series

$$x_t=\mu+\phi_1x_{t-1}+w_t \tag{D29}$$

$$x_t=\mu+\phi\left(\phi x_{t-2}+w_{t-1}\right)+w_t$$

$$x_t=\mu+\phi^2x_{t-2}+\phi w_{t-1}+w_t$$

$$x_t=\mu+\phi^2\left(\phi x_{t-3}+w_{t-2}\right)+\phi w_{t-1}+w_t$$

$$x_t=\mu+\phi^3x_{t-3}+\phi^2w_{t-2}+\phi w_{t-1}+w_t$$

$$\vdots$$

$$x_t=\mu+\phi^kx_{t-k}+\sum_{j=0}^{k-1}\phi^jw_{t-j} \tag{D30}$$

In the limit as k$\to\infty$, and assuming $|\phi|<1$, it implies

$$x_t=\mu+\sum_{j=0}^{k-1}\phi^jw_{t-j} \tag{D31}$$

and

$$x_{t+h} = \mu + \sum_{j=0}^{k-1} \phi^j w_{t+h-j} \tag{D32}$$

The Autocovariance is given by:

$$\gamma(h) = E\left(x_{t+h} - \mu\right)\left(x_t - \mu\right)$$

$$\gamma(h) = E\left(\sum_{j=0}^{k-1} \phi^j w_{t+h-j}\right)\left(\sum_{i=0}^{k-1} \phi^i w_{t-i}\right) \tag{D33}$$

Using the property from Equation D25 and simplifying gives

$$\gamma(h) = \sigma_w^2 \sum_{j=0}^{\infty} \phi^j \phi^{j+h} \tag{D34}$$

$$\gamma(h) = \sigma_w^2 \phi^h \sum_{j=0}^{\infty} \phi^{2j} \tag{D36}$$

But $\sum_{j=0}^{\infty} \phi^{2j}$ forms a geometric series, the sum of which is given by

$s_n = \frac{\left(1-\phi^{2n}\right)}{\left(1-\phi^2\right)}$ Which in the limit as n$\rightarrow\infty$, simplifies to give $s_n = \frac{1}{\left(1-\phi^2\right)}$. Equation (D36) can thus be written as

$$\gamma(h) = \frac{\sigma_w^2 \phi^h}{\left(1-\phi^2\right)}$$

Using $\rho(h) = \frac{\gamma(h)}{\gamma(0)}$

$$\rho(h) = \frac{\sigma_w^2 \phi^h}{\left(1-\phi^2\right)} \Bigg/ \frac{\sigma_w^2 \phi^0}{\left(1-\phi^2\right)}$$

$$\rho(h) = \phi^h \qquad h \geq 0 \tag{D37}$$

Furthermore,

$$\rho(h-1) = \phi^{h-1} \tag{D38}$$

Dividing Equation (D37) by Equation (D38), and simplifying, it can shown that $\rho(h)$ satisfies the recursive relations:

$$\rho(h) = \phi\rho(h-1) \qquad h \geq 1 \tag{D39}$$

From Equation (D38) and (D39), it can be noticed that the ACF of an autoregressive process does not cut off after lag h like an MA process. Thus, it is worthwhile to pursue a function that will behave like the ACF (of an MA series) but will represent an AR series. Such a function is referred to as partial autocorrelation function (PACF) and is the correlation between time-indexed variables x_t and x_{t+h} with the effect of $x_{t\pm1,}x_{t\pm2,}x_{t\pm3,}x_{t\pm4,}...x_{t\pm h-1}$ removed (Box *et al*., 1994; Shumway *et al*., 2000). Let $\hat{x}_t$ denote the best linear predictor of x_t based on $\{x_{t-1}, x_{t-2}, x_{t-3}, ..., x_{t-h+1}\}$. That is:

$$\hat{x}_t = \beta_1 x_{t-1} + \beta_2 x_{t-2} + \beta_3 x_{t-3} + ... + \beta_{h-1} x_1 \tag{D40}$$

Where the β's are chosen to minimize the mean square error $E(x_t - \bar{x})^2$.

Also, let $\hat{x}_{t-h}$ denote the best linear predictor or regression of x_{t-h} based on the future values of $\{x_{t-1}, x_{t-2}, x_{t-3}, ..., x_{t-h+1}\}$. That is:

$$\hat{x}_{t-h} = \beta_1 x_{t-h+1} + \beta_2 x_{t-h+2} + \beta_3 x_{t-h+3} + ... + \beta_{h-1} x_1 \tag{D41}$$

Where the β's are again chosen to minimize the mean square error $E(x_{t-h} - \bar{x})^2$. The partial autocorrelation of x_t, ϕ_{hh} is defined as the correlation between the x_t and x_{t-h} with the dependence chain between them removed and is denoted as:

$$\phi_{hh} = cor\left(x_t - \hat{x}_t, x_{t-h} - \hat{x}_{t-h}\right) \quad h > 1 \tag{D42}$$

And

$$\phi_{11} = cor\left(x_t, x_{t-1}\right) = \rho(1) \qquad h = 1 \tag{D43}$$

While Equations (D42) and (D43) illustrate the general idea behind the PACF, the determination of the function coefficients is often done by following the derivation of autocovariance function, then the autocorrelation function from which the PACF values ϕ_{hh} are estimated. Precisely, the estimate $\hat{\phi}_h$ of the last coefficient of the ACF provides the estimated PACF coefficient value ϕ_{hh}. This is discussed below.

Consider the general stationary series,

$$x_t = \phi_1 x_{t-1} + \phi_2 x_{t-2} + \phi_3 x_{t-3} + \ldots + \phi_p x_{t-p} + w_t \tag{D44}$$

Where w_t is independently identically distributed, IID (0, σ^2_w), the autocovariance is expressed as:

$$\gamma(h) = E\left(x_t \cdot x_{t+h}\right) \tag{D45}$$

$$\gamma(h) = E\left(\left(\phi_1 x_{t-1} + \phi_2 x_{t-2} + \phi_3 x_{t-3} + \ldots + \phi_p x_{t-p} + w_t\right) \cdot x_{t+h}\right) \tag{D46}$$

$$\gamma(h) = E\left(\phi_1 x_{t-1} \cdot x_{t+h}\right) + E\left(\phi_2 x_{t-2} \cdot x_{t+h}\right) + \ldots + E\left(\phi_p x_{t-p} \cdot x_{t+h}\right) + E\left(w_t \cdot x_{t+h}\right) \tag{D47}$$

Since w_t is IID

$$\gamma(h) = \phi_1 \gamma(h-1) + \phi_2 \gamma(h-2) + \ldots + \phi_p \gamma(h-p) \tag{D48}$$

Dividing through by $\gamma(0)$ gives

$$\rho(h) = \phi_1 \rho(h-1) + \phi_2 \rho(h-2) + \ldots + \phi_p \rho(h-p) \tag{D49}$$

Equations (D48 and (D49) are the autocovariance and autocorrelation functions written in difference equation forms. Let ϕ_{hj} denote the j^{th} coefficient in an autoregressive

representation of order h so that ϕ_{hh} is the last coefficient. Then according to Box *et al*., (1994) ϕ_{hj} satisfy the set of equations:

$$\rho(j) = \phi_{h,1}\rho(j-1) + \phi_{h,2}\rho(j-2) + ... + \phi_{h,h-1}\rho(j-h+1) + \phi_{h,h}\rho(j-h) \qquad \text{(D50)}$$

Equation (D50) can be arranged in a matrix form as:

$$\begin{bmatrix} 1 & & & & & & \\ \rho_1 & 1 & & & & & \\ \rho_2 & \rho_1 & 1 & & & & \\ . & & & & & & \\ . & & & & & & \\ . & & & & & & \\ \rho_{h-1} & \rho_{h-2} & \rho_{h-3} & \rho_{h-4} & . & . & . & 1 \end{bmatrix} \begin{bmatrix} \phi_{h1} \\ \phi_{h2} \\ \phi_{h3} \\ . \\ . \\ . \\ \phi_{hh} \end{bmatrix} = \begin{bmatrix} \rho_1 \\ \rho_2 \\ \rho_3 \\ . \\ . \\ . \\ \rho_h \end{bmatrix} \qquad \text{(D51)}$$

The matrix can be represented as:

$\Gamma\Phi = R_\rho$ from which $\Phi = \Gamma^{-1}R_\rho$. The values of $\rho_{i,s}$ (the autocorrelation function values) are all known and so the coefficient matrix Φ_{hh} can be solved to get the partial autocorrelation function values. For a large matrix, the solution of Equation (D51) may pose a computational burden. So the algorithm below due to Durbin-Levinson (Box *et al*., 1994; Shumway *et al*., 2000) below is used to determine Φ_{kk}. Where for index n≥ 1,

$$\phi_{nn} = \frac{\rho(n) - \sum_{k=1}^{n-1} \phi_{n-1,k}\rho(n-k)}{1 - \sum \phi_{n-1,k}\rho(k)}$$

And for $n \geq 2$ (D52)

$$\phi_{nk} = \phi_{n-1,k} - \phi_{nn}\phi_{n-1,n-k} \qquad \vee\ k = 1,2,3...n-1$$

Specifically, the PACF are useful in identifying the order of an autogressive model which essentially reveals the extent to which the data points are correlated with each other and the how far back in the data the correlation exist. The PACF of an AR(P)

series is zero at lags p+1 and greater. In this way, the PACF of an AR process resembles the ACF of an MA process. In general, the PACF is often used in conjunction with the ACF in order to describe the data adequately. Table A.1 is a summary of the behavior of ACF and PACF for a causal time series data (i.e. a series for which future data points depend only on previous or past points).

Table D1 Behavior of the ACF and PACF for Causal an Invertible ARMA Series (P is the lag for an AR process and q is the lag for an MA Process)

Function	AR(*p*)	MA(*q*)	ARMA(*p, q*)
ACF	Tails off (Decreases gradually to zero, but may oscillate)	Cuts off after lag q	Tails of
PACF	Cuts off after lag p (Is Significantly zero)	Tails off	Tails off

(Shumway *et al*., 2000)

Often, if the sample ACF of a process indicates that an AR model may be appropriate to describe the data, then the sample PACF plot is examined to help identify the order. In using the plot, it is useful to always look for lags where the autocorrelations essentially become significantly close to zero. In doing so, a confidence interval equivalent to $\left(\pm z/\sqrt{n} \right)$, where "z" is the ordinate of the standard normal distribution and "n" is the sample size, is often placed on the chart and all correlations below this limit are deemed to be not significant. In general, the most commonly used confidence limit is the 95 % level and it is also used in this work to determine a cut off for the ACF and PACF. The 95 % confidence level has an approximate z-value band of ±1.96. However, a value

of $\pm 2/\sqrt{n}$ is often used in most applications for the ACF and PACF. Figure D1 is a correlogram showing the sample autocovariance function, the autocorrelation function (ACF) and the partial autocorrelation function (PACF) plot using the actuating errors from the primary control loop of the ExxonMobil cascade data.

Notice the cycling effect in the autocovariance and autocorrelation function plots (the first two graphs respectively). The plot in Figure D1 can be interpreted using Table D1, to mean that autocorrelation effect in the third plot significantly vanishes after lag 7. As a result the data was differenced such that get a new error data where err(k) = e(k+7)-e(k). Where "e" is the actuating error prior to differencing and "err" is the actuating error after differencing. The new actuating error from the reference period was analyzed using the health monitor. The results are shown in Figure D2. Notice that the sampling ratio reduced from 34 in Figure 4.28 to 29 in Figure D2.

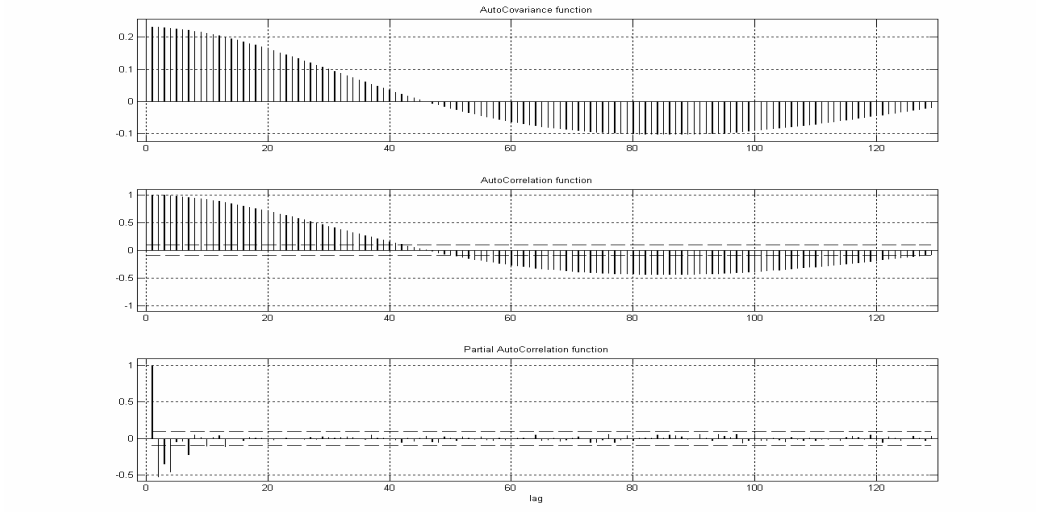


Figure D1 AFC, PACF of the Primary Control Loop data from ExxonMobil Data

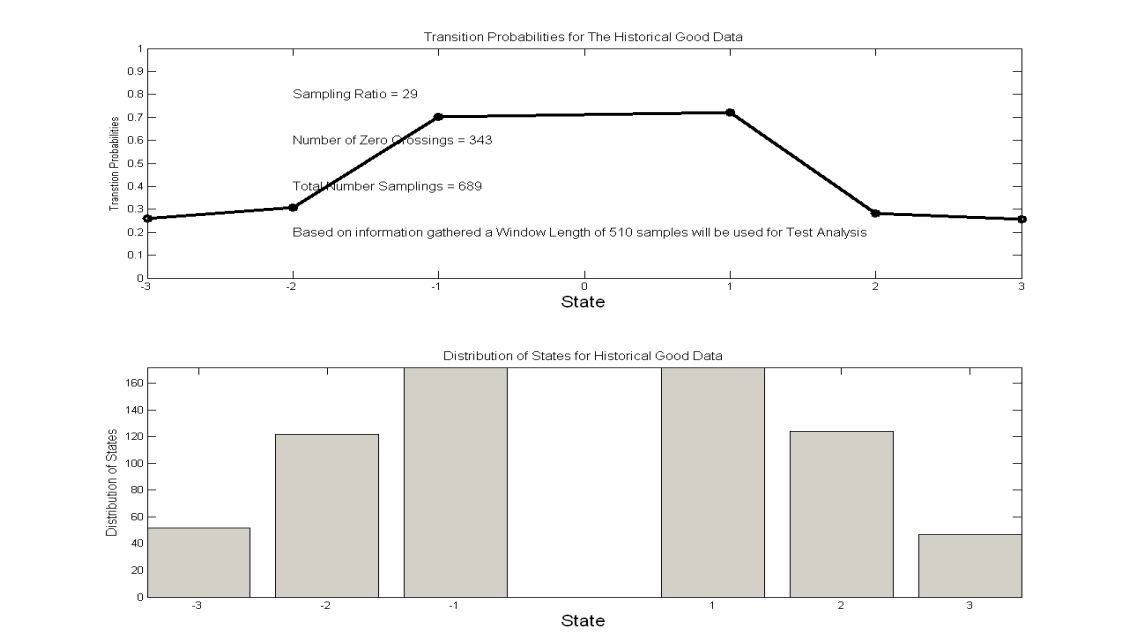


Figure D2 **Distribution of States and Transition Probabilities from Reference Good Data after Differencing the Data** (Window Length = 510 Samples; Sampling Ratio = 29; Difference = 7; α_T = 1%, β = 1%, λ = 0.9)

Also the Window length decreased from 833 in Figure 4.28 to 510 in Figure D2. Moreover the number of states required was determined to be 6 (±3 states), which was less that required in Figure 4.28. Figure D3 shows the performance output. Notice how the monitor flagged the period when the controller was known to be experiencing difficulties. Comparing Figures D3 and 4.29, it is noticed that there was no flagging in the case of Figure 4.29. It is likely that the strong correlation present in the original data might have had an impact on the analysis resulting in a large sampling ratio and window length which may in turn be responsible for the missed observations.

The analysis was repeated for the secondary loop by plotting the correlogram and studying the ACF and PACF shown in Figure D4. It was noticed that The PACF cut-off after lag 12. Hence, the data was differenced by 12 samples apart and analysis of the reference data is shown in Figure D5. The algorithm estimated a sampling ratio of 3 and a window length of 457 samples in comparison with Figure 4.30 where the monitor estimated, for the original data, a sampling ratio of 11 and a window length of 513 samples. The performance output for the secondary loop is shown in Figure D6. Notice that in addition to flagging within the same region, which was flagged in Figure 4.31, other areas were also flagged. It turns out that, there were control problems in these regions as well that was probably not revealed in Figure 4.31 due perhaps, to the long window length and large sampling ratio.

In summary, the presence of autocorrelation in a data can have a significant impact on the sampling ratio and window length. If it is identified that a strong autocorrelation exist in the data, it may be a good idea to study the correlogram and perhaps reduce or remove the extent of the correlation before carrying out performance analysis.

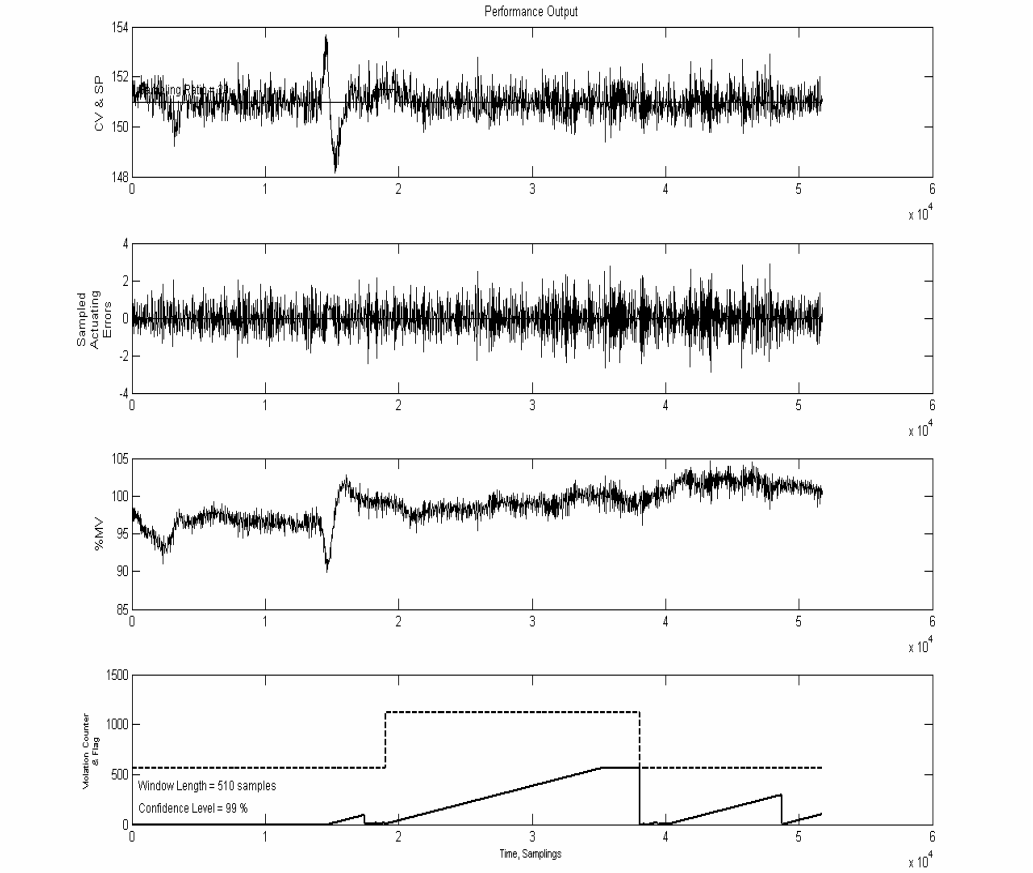


Figure D3 **Control Loop Performance Output** (Sampling Period = 5s, Sampling Ratio = 1; Window length = 510 Samples, Startup Period = 0 Samples, Grace Period 560 Samples, Violation Counter Trigger = Length of Grace Period +1, Overall Level of Significance α_T) = 1%)

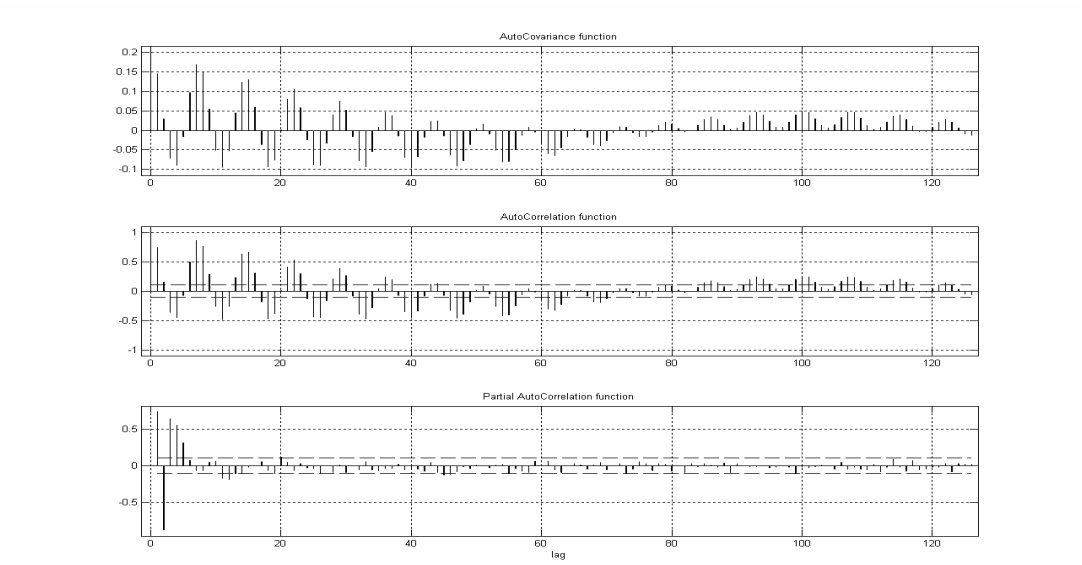


Figure D4 AFC, PACF of the Secondary Control Loop data from ExxonMobil Data

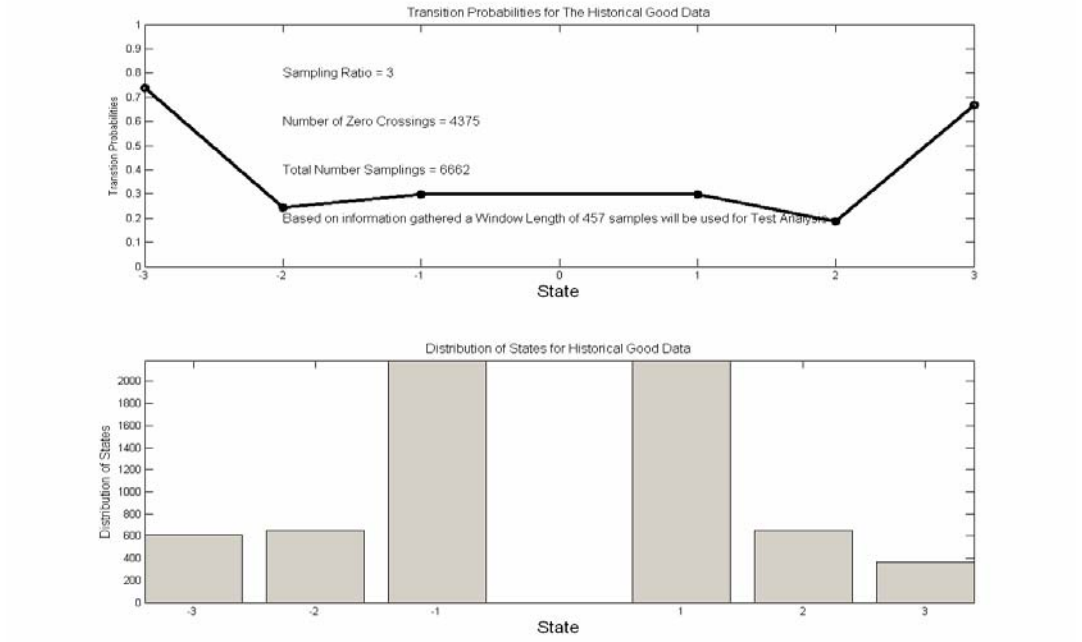


Figure D5 Distribution of States and Transition Probabilities from Reference Good Data after Differencing the Data (Window Length = 457; Samples; Sampling Ratio = 3; Difference = 7; α_T = 1%, β = 1%, λ = 0.9)

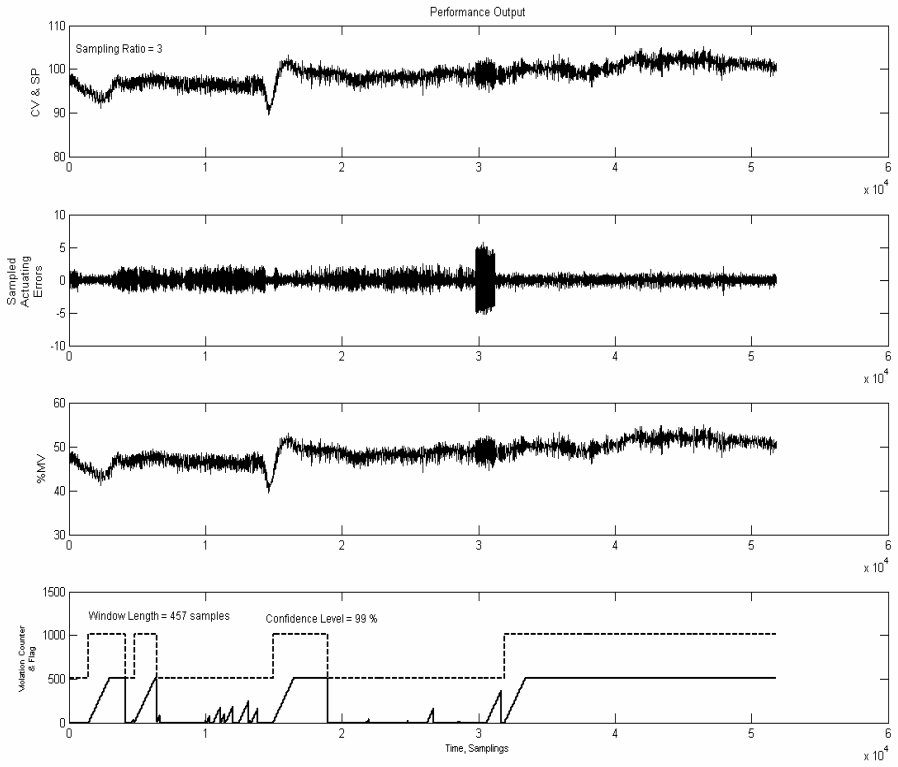


Figure D6 **Control Loop Performance Output** (Sampling Period = 5s, Sampling Ratio = 1; Window length = 457 Samples, Startup Period = 0 Samples, Grace Period 507 Samples, Violation Counter Trigger = Length of Grace Period +1, Overall Level of Significance (α_T)) = 1%)

APPENDIX E

Model Predictive Control (MPC) Details

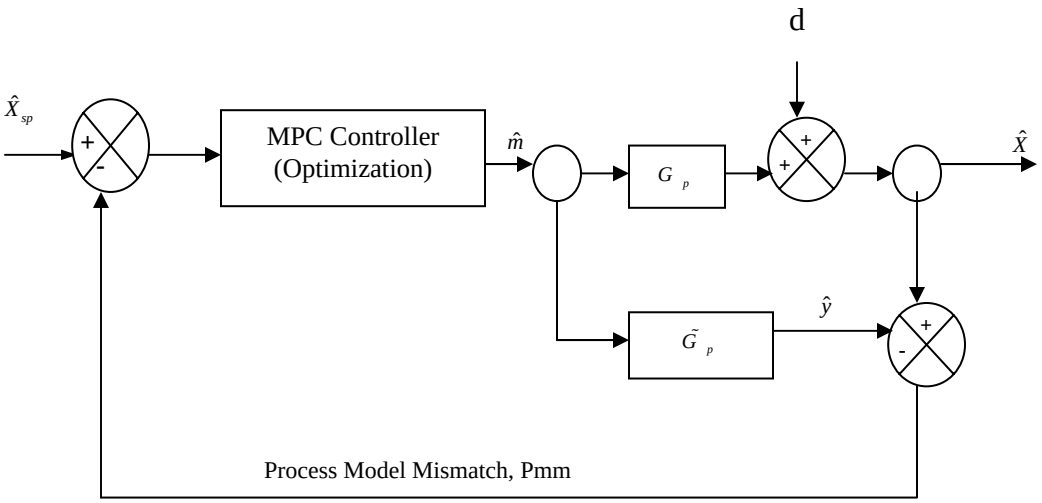


Figure E1 **Schematic Diagram of a Process Controlled with MPC Technique** (e = Actuating Error = Process Model Mismatch (Residuals), X = Controlled Variable, m = Manipulated Variable, $\tilde{G}_p$ = Step Response Model, G_p = Process, $\hat{X}_{sp}$ = Desired Trajectory (Setpoint), d = Disturbance)

Process Model: $$G_p = \frac{(-3s+1)e^{-3s}}{(2s+1)(5s+1)} \qquad \text{(E1)}$$

The parameters were used for the simulation:

Move suppression factor, f = 10

Prediction horizon "PH" = 40 samples

Control horizon "CH" = ¼*PH

Number of controlled variables =1

Number of input variables = 1

Algorithm:

At the start of the algorithm, obtain a prediction horizon and control horizon. Obtain the step response coefficients over the entire prediction horizon and determine the dynamic matrix A_{DMC} and the dynamic controller matrix K_{DMC} (discussed elsewhere; Lee *et. al*, 1998; Brosilow *et al*, 2002; Ogunaike *et al*, 1994; Marlin 2002; Erickson *et al*, 1999), where

K_{DMC} = (Transpose(A_{DMC})*A_{DMC} + f^2)*(Transpose(A_{DMC}))

Since the model does not account for disturbances and load changes. To ensure better prediction, the prediction vector is corrected with the mismatch between the process and the model (i.e. process model mismatch, (Pmm)). In this work, the process is assumed to be represented by a simulator. The simulator is represented by the Equation (E1), and the output from the simulator (y_{sim}), plus random noise is used to represent the true process measurement (X). Also, obtain a vector of ones "Id" equal in length to the prediction horizon to be used for scalar to vector transformations. Figure E2 is the step response output due to a unit step change in the controller output signal from which the response coefficients are obtained.

1. Obtain the current process measurement "X" and initialize the prediction vector y^p to that value.
2. Estimate the disturbance or process model mismatch (i.e. Pmm = X*Id - y^p).
3. Correct the predicted vector y^p with the Pmm (i.e. $y^p = y^p$ + Pmm).

4. Determine the error (Err) between the corrected value of y^p and the desired trajectory "X_{sp}" (i.e. Err = X_{sp}*Id - y^p).

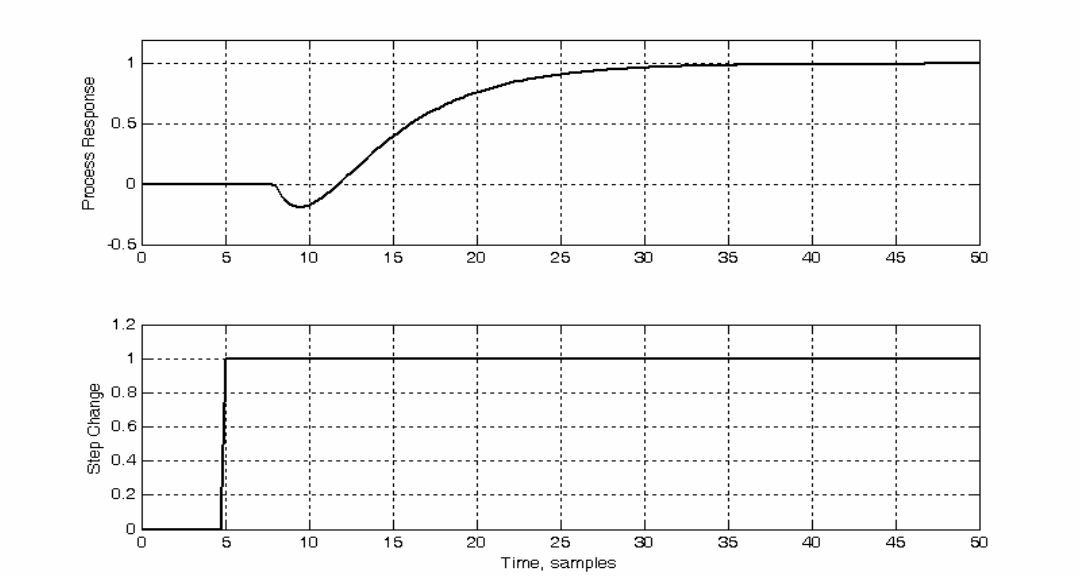


Figure E2 Step Response of Second-Order plus Time Delay (SOPTD) Process with Inverse Response

5. Estimate the control action "ΔU" as K_{DMC}*Err and implement only the first element as U = U + ΔU(1,1)
6. Allow the process to run and obtain the process measurement "X" as X = y_{sim} + noise. (Where y_{sim}, is the simulator output which is shown below)
7. Update the prediction vector y^p to compensate for the control move just implemented (i.e. $y^p = y^p + A_{DMC}$*ΔU)
8. The first element in y^p is not needed any more so shift all elements in y^p up by one and replace the last element with the penultimate element.
9. At the next sampling go to step 2

Process Simulator

The process, which is represented by $G_p = \frac{(\lambda s+1)e^{-\theta s}}{(t_{p1}s+1)(t_{p2}s+1)}$ is first transformed into a differential equation and then solved using the fourth-order Runge-Kutta (RK4) method after decoupling the second order differential equation (obtained by inverting the model from the laplace domain to time domain), to two first-order differential equations that are solved for the process output, given the input signal from the MPC controller, and after applying the effect of the transport or time delay.

$$G_p = \frac{y(s)}{U(s)} = \frac{(\lambda s+1)e^{-\theta s}}{(t_{p1}s+1)(t_{p2}s+1)} \tag{E2}$$

Considering the part of Equation (E2) that does not include the delay and expanding gives:

$$(t_{p1}s+1)(t_{p2}s+1)\,y(s) = (\lambda s+1)U(s) \tag{E3}$$

$$\left(t_{p1}t_{p2}s^2 + (t_{p1}+t_{p2})s+1\right)y(s) = (\lambda s+1)U(s) \tag{E4}$$

Inverting Equation (E4) results in the second order differential equation:

$$t_{p1}t_{p2}\frac{d^2y(t)}{dt^2} + (t_{p1}+t_{p2})\frac{dy(t)}{dt} + y(t) = \lambda\frac{du}{dt} + u(t) \tag{E5}$$

Let $R = t_{p1}t_{p2}$ and $B = (t_{p1}+t_{p2})$, $y(0) = 0$ and $\frac{dy(0)}{dt} = 0$

$$\frac{d^2y(t)}{dt^2} + \frac{Bdy(t)}{Rdt} + \frac{1}{R}y(t) = \frac{\lambda du}{Rdt} + \frac{1}{R}u(t) \tag{E6}$$

$$\frac{d^2y(t)}{dt^2} = -\frac{Bdy(t)}{Rdt} - \frac{1}{R}y(t) + \frac{\lambda du}{Rdt} + \frac{1}{R}u(t) \tag{E7}$$

Let $z(t) = \frac{dy(t)}{dt}$, then $\frac{dz(t)}{dt} = \frac{d^2y(t)}{dt^2}$, $z(0) = 0$

$$\frac{dz}{dt} = -\frac{B}{R} z(t) - \frac{1}{R} y(t) + \frac{\lambda\left(u(t) - u(t-1)\right)}{R\Delta t} + \frac{1}{R} u(t) \tag{E8}$$

Applying the time delay component, θ gives

$$\frac{dz}{dt} = -\frac{B}{R} z(t) - \frac{1}{R} y(t) + \frac{\lambda\left(u(t-\theta) - u(t-\theta-1)\right)}{R\Delta t} + \frac{1}{R} u(t-\theta) \tag{E9}$$

$$\frac{dy}{dt} = z \tag{E10}$$

Equations (E9) and (E10) are then solved using the Rk4 formulas for a second-order differential equation decoupled into two first order differential equations below:

Let $f_1(t, y, z) = \frac{dz}{dt}$, $\quad f_2(t, y, z) = z$, then

$$\begin{aligned} k_{1z} &= f_1(t,y,z) \\ k_{1y} &= f_2(t,y,z) \end{aligned} \tag{E11}$$

$$\begin{aligned} k_{2z} &= f_1(t+h/2, y+k_{1y}/2, z+k_{1z}/2) \\ k_{2y} &= f_2(t+h/2, y+k_{1y}/2, z+k_{1z}/2) \end{aligned} \tag{E12}$$

$$\begin{aligned} k_{3z} &= f_1(t+h/2, y+k_{2y}/2, z+k_{2z}/2) \\ k_{3y} &= f_2(t+h/2, y+k_{2y}/2, z+k_{2z}/2) \end{aligned} \tag{E13}$$

$$\begin{aligned} k_{4z} &= f_1(t+h, y+k_{3y}, z+k_{3z}) \\ k_{4y} &= f_2(t+h, y+k_{3y}, z+k_{3z}) \end{aligned} \tag{E14}$$

$$\begin{aligned} z &= z + h*(k_{1z} + 2*k_{2z} + 2*k_{3z} + k_{4z})/6 \\ y &= y + h*(k_{1y} + 2*k_{2y} + 2*k_{3y} + k_{4y})/6 \end{aligned} \tag{E15}$$

Where h = time step = sampling time interval,

t = time and

z = an intermediate variable

y = Solution to Equation (E7) (and $y_{sim} = y$)

The measured process output “X” in step 6 is sum of y plus random noise

(X = y + noise).

APPENDIX F

Comparison between Exact Binomial Analysis and the Normal Approximation to the Binomial Distribution

This appendix is an attempt to clarify the reason why it is necessary to use the exact binomial distribution and not the normal approximation to the binomial distribution that was discussed in Chapter 3.

Using the normal approximation to the Binomial relationship discussed in Chapter 3, the data discussed in chapter 4, Section 4.2.1 was analyzed and the results are discussed below. The output of the reference data analysis is shown in Figure F1. The algorithm estimated a sampling ratio 1 and a window length of 380 samples. In comparison when the exact binomial relation was used in Section 4.2.1 the sampling ratio was 1, but the algorithm estimated a window length of 526 samples (i.e. 146 samples more). This seemed unexpected at first site since a short window length that also reduces the associated Type-I and Type-II errors is always desirable. However, when the test data was analyzed as shown in Figure F2, it was observed that between sampling 40000 to 46000 when the controller was made sluggish, the monitor flagged on and off. This could be the result of numerous Type-II errors being made due to a short window length.

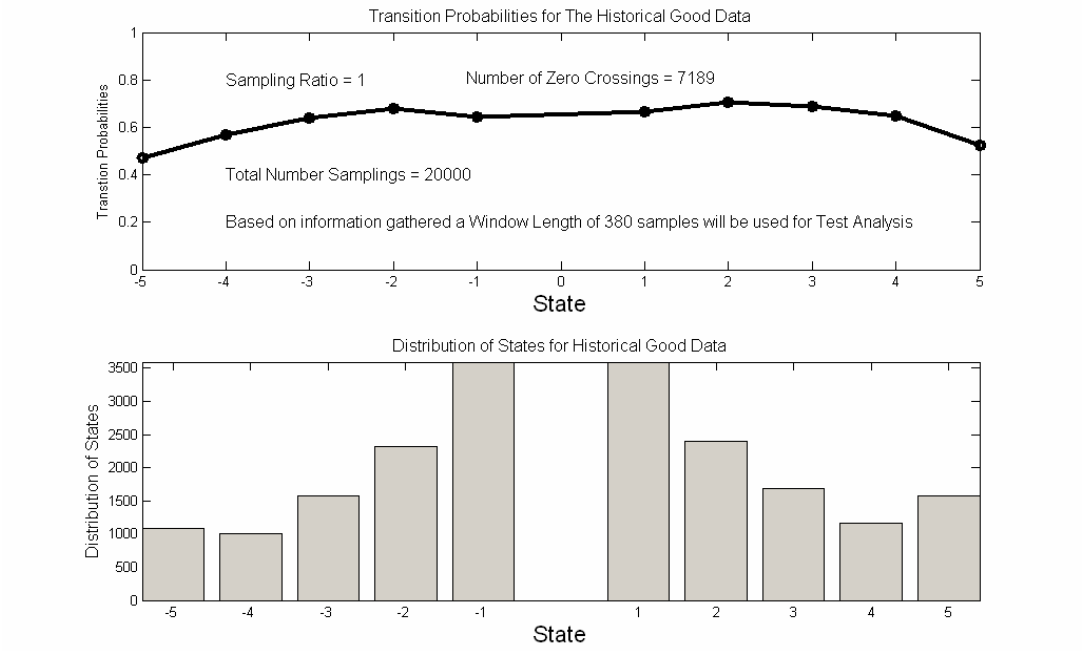


Figure F1 Distribution of States and Transition Probabilities from Reference Good Data Using Normal Approximation to the Binomial Distribution Relation (Window length = 380 samples; Sampling Ratio = 1)

Again when the pressure drop data discussed in section 4.3.2 was analyzed using the normal approximation to the binomial, the monitor estimated a sampling ratio of 2 (same as before in section 4.3.2), but a window length of 432 samples (368 samples less than previous) as shown in Figure F3. When the test data was analyzed (Figure F4), the monitor flagged throughout the entire during of the testing. However, it is known that there were instances when control performance was good and that performance degradation in the loop was gradual. The performance output in Figure F4 reveals the possibility of excessive Type-I errors when the approximate methods is used.

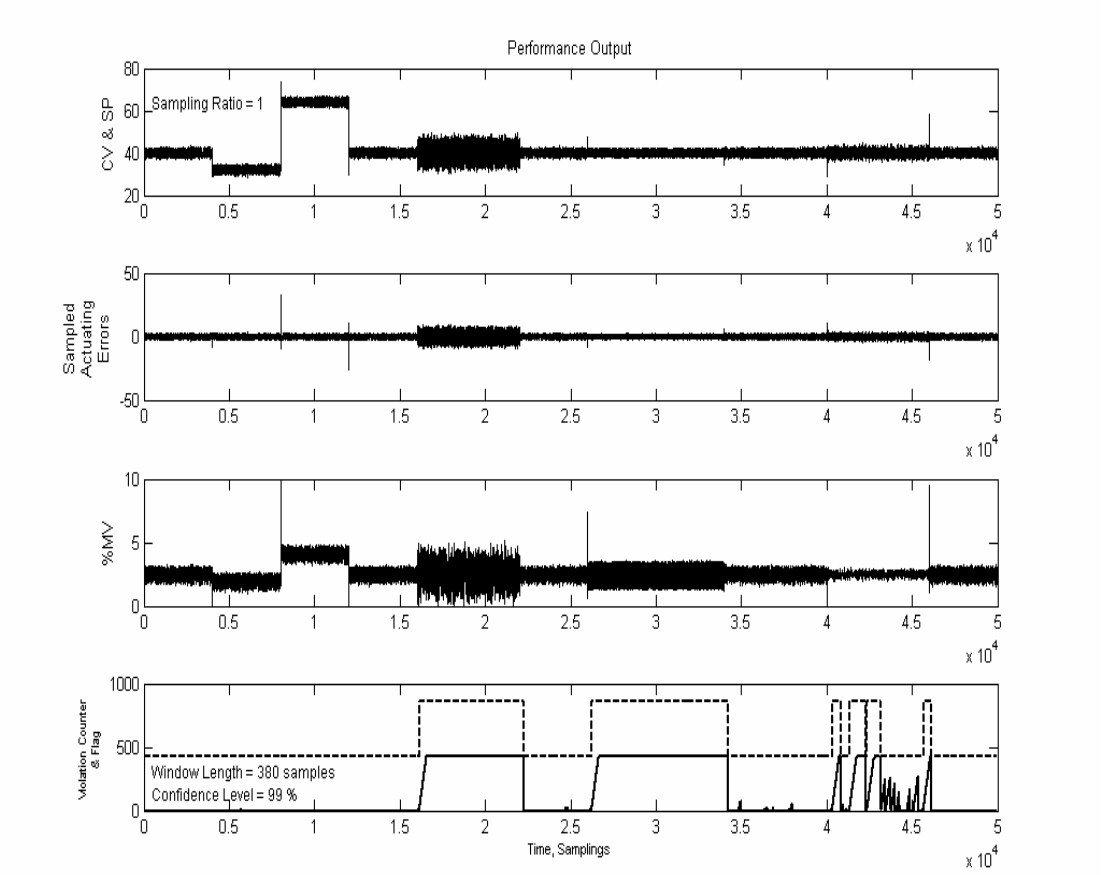


Figure F2 **Control Loop Performance Output** (Sampling Period = 0.25s, Sampling Ratio = 1; Window length = 380 Samples, Startup Period = 0 Samples, Grace Period 430 Samples, Violation Counter Trigger = Length of Grace Period +1, Overall Level of Significance (α_T) = 1%)

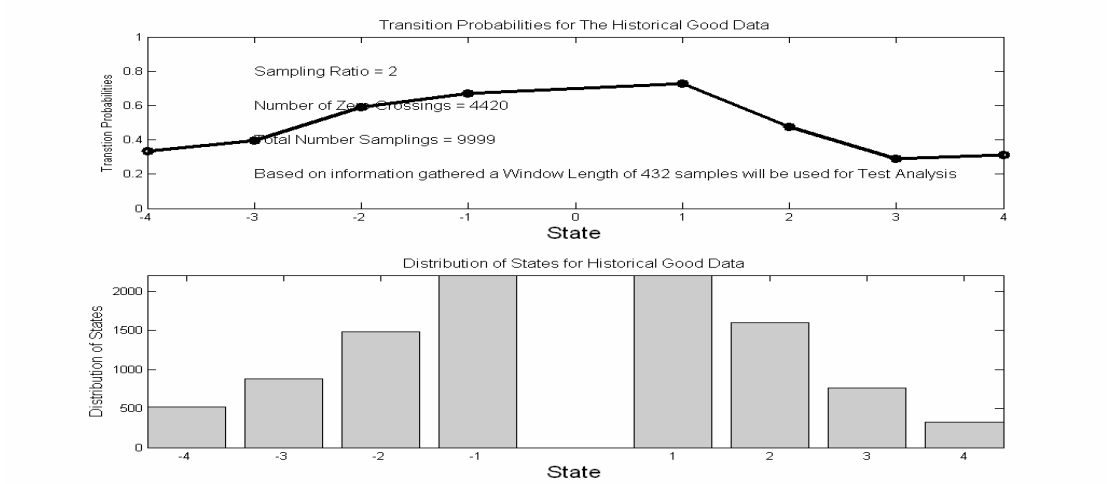


Figure F3. Distribution of States and Transition Probabilities from Reference Good Data Using Normal Approximation to the Binomial Distribution Relation (Window length = 432 Samples; Sampling Ratio = 1)

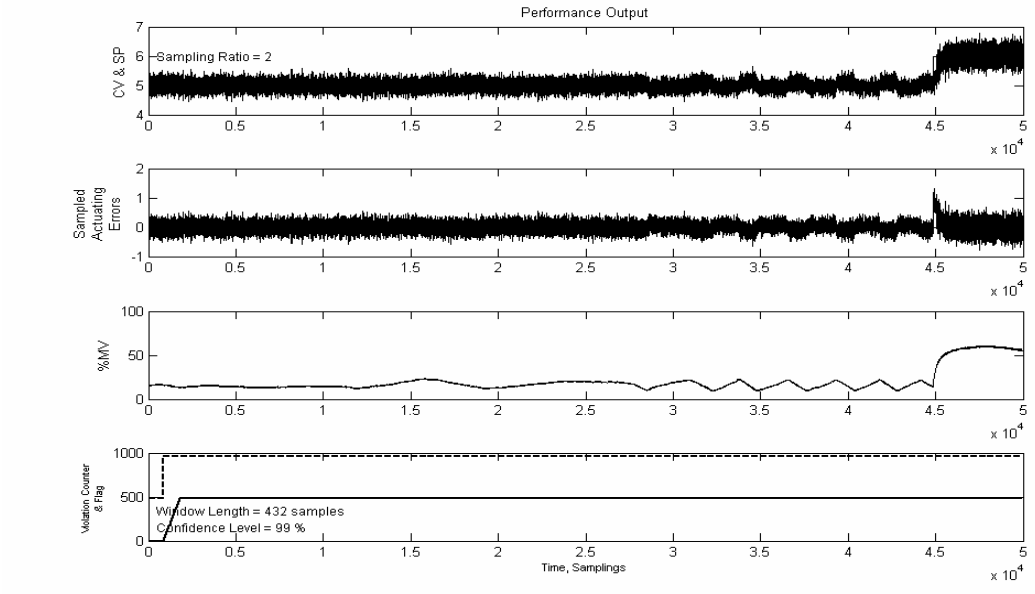


Figure F4 Control Loop Performance Output (Sampling Period = 0.25s, Sampling Ratio = 1; Window length = 432 Samples, Startup Period = 0 Samples, Grace Period 482 Samples, Violation Counter Trigger = Length of Grace Period +1, Overall Level of Significance (α_T) = 1)

APPENDIX G

Why Ignoring Amplitude Does not Limit the General Performance of the Health Monitor

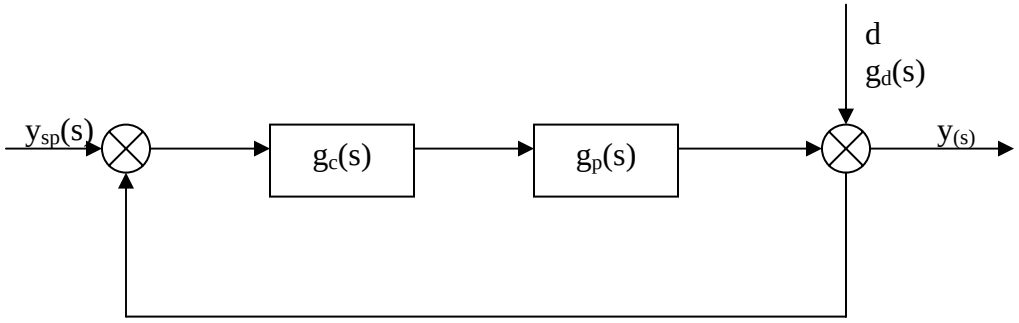


Figure G1 Schematic Diagram of First-Order Process with a PI Controller

Consider the closed loop response of the simplified system above. Let $g_p(s) = \frac{K_p}{\tau_p s + 1}$ and for a PI controller, $g_c(s) = K\left(1 + \frac{1}{\tau_I s}\right)$, the closed loop response for the system in Figure 1 above is given by

$$y(s) = \frac{g_c g_p}{1 + g_c g_p} y_{sp} + \frac{g_d}{1 + g_c g_p} d \qquad \text{(H1)}$$

Assuming servo control, $y(s) = \frac{g_c g_p}{1 + g_c g_p} y_{sp}$ (H2)

Substituting g_p and g_c,

$$\frac{y(s)}{y_{sp}(s)} = \frac{\frac{K_p}{(\tau_p s + 1)} K\left(1 + \frac{1}{\tau_I s}\right)}{1 + \frac{K_p}{(\tau_p s + 1)} K\left(1 + \frac{1}{\tau_I s}\right)} \qquad \text{(H3)}$$

$$\frac{y(s)}{y_{sp}(s)} = \frac{\dfrac{K_p K_c}{(\tau_p s+1)}\left(\dfrac{\tau_I s+1}{\tau_I s}\right)}{1+\dfrac{K_p K_c}{(\tau_p s+1)}\left(\dfrac{\tau_I s+1}{\tau_I s}\right)} \tag{H4}$$

Simplifying,

$$\frac{y(s)}{y_{sp}(s)} = \frac{\dfrac{K_p K_c(\tau_I s+1)}{\cancel{(\tau_p s+1)\tau_I s}}}{\dfrac{(\tau_p s+1)(\tau_I s)+K_p K_c(\tau_I s+1)}{\cancel{(\tau_p s+1)\tau_I s}}} \tag{H5}$$

$$\frac{y(s)}{y_{sp}(s)} = \frac{K_p K_c(\tau_I s+1)}{(\tau_p s+1)(\tau_I s)+K_p K_c(\tau_I s+1)} \tag{H6}$$

$$\frac{y(s)}{y_{sp}(s)} = \frac{K_p K_c(\tau_I s+1)}{(\tau_p \tau_I s^2+\tau_I s+K_p K_c \tau_I s+K_p K_c)} \tag{H7}$$

$$\frac{y(s)}{y_{sp}(s)} = \frac{c(\tau_I s+1)}{(as^2+\tau bs+c)} \tag{H8}$$

Where

$a = \tau_p \tau_I$

$b = \tau_I(1+K_p K_c)$

$c = K_p K_c$

Then, $y(s)(as^2+bs+c) = c(\tau_I s+1)y_{sp}$ (H9)

Consider a case where there is no change in setpoint, thus

$y(s)(as^2+bs+c) = 0$, (H10)

Inverting this Equation gives

$$a\frac{d^2y}{dt^2}+b\frac{dy}{dt}+cy=0 \qquad \text{(H11)}$$

Let $y=e^{\lambda t}$ be a characteristic equation,

Then, $y'=\lambda e^{\lambda t}$; $y''=\lambda^2 e^{\lambda t}$ substituting in Equation H11above gives

$$a\lambda^2 e^{\lambda t}+b\lambda e^{\lambda t}+ce^{\lambda t}=0 \qquad \text{(H12)}$$

$\left(a\lambda^2+b\lambda+c\right)e^{ct}=0$, but $e^{\lambda t}\neq 0$ therefore

$$\left(a\lambda^2+b\lambda+c\right)=0 \qquad \text{(H13)}$$

The roots of Equation (H13) are given by

$$\lambda_1,\lambda_2=\frac{-b\pm\sqrt{b^2-4ac}}{2a} \qquad \text{(H14)}$$

The system is oscillatory when the roots of this Equation are complex: i.e. $4ac > b^2$

Let $\omega=\frac{\sqrt{b^2-4ac}}{2a}$, (H15)

Then, $\lambda_1,\lambda_2=\frac{-b}{2}\pm i\omega$, Where $i=\sqrt{-1}$

$$\lambda_1=\frac{-b}{2}-i\omega,\lambda_2=\frac{-b}{2}+i\omega$$

$y_1=e^{\lambda_1 t}$; $y_2=e^{\lambda_2 t}$

$y=k_1y_1+k_2y_2$

$y_1=e^{\lambda_1 t}$; $y_2=e^{\lambda_2 t}$

$y=k_1e^{\lambda_1 t}+k_2e^{\lambda_2 t}$

$y=k_1e^{(-.5b-i\omega)t}+k_2e^{(-0.5b+i\omega)t}$

$$y=k_1\left(e^{-0.5bt}e^{-i\omega t}\right)+k_2\left(e^{-0.5bt}e^{0.5i\omega t}\right) \qquad \text{(H16)}$$

$$y = e^{-0.5bt}\left(k_1 e^{-0.5i\omega t} + k_2 e^{0.5i\omega t}\right) \tag{H17}$$

Equation (H17) can be further simplified as

$$y = e^{-0.5bt}\left(\left(k_1 + k_2\right)\cos\omega t + i\left(k_1 - k_2\right)\sin\omega t\right) \tag{H18}$$

Where ω is the natural frequency of oscillation. From Equation (H15),

$\omega = \dfrac{\sqrt{b^2 - 4ac}}{2}$, back substituting all the variables

$$\omega = \frac{\sqrt{\left(\tau_I\left(1 + KpKc\right)\right)^2 - 4\tau_p\tau_I K_p K_c}}{2\tau_p\tau_I}$$

$$\omega = \sqrt{\frac{\left(\tau_I\left(1 + KpKc\right)\right)^2}{4\tau_p^2\tau_I^2} - \frac{4\tau_p\tau_I K_p K_c}{4\tau_p^2\tau_I^2}}$$

$$\omega = \sqrt{\left(\frac{1 + KpKc}{2\tau_p}\right)^2 - \frac{K_p K_c}{\tau_p\tau_I}} \tag{H19}$$

This analysis reveals that, the natural frequency of oscillation ω is such that it depends on the process parameters, i.e. $\omega = f\left(K_c, K_p, \tau_I, \tau_p\right)$. Moreover, the natural period of oscillation T is related to the natural frequency as $\omega = 2\pi/T$, hence, $T = f\left(K_c, K_p, \tau_I, \tau_p\right)$. While the analysis in this example considered a PI controller and a first-order process, other systems considered also show that the natural period of oscillation is a function of process parameters such as the process gain, process time constant and others.

Therefore, if an event such as the process gain, process time constant, controller gain and controller time constant, changes the amplitude of a signal, it will affect the natural frequency of oscillation of the signal and hence the natural period of oscillation.

This will lead to differences in the period of oscillation of the data sampled in the reference stage and the period of oscillation of the data sampled during testing or performance monitoring and the health monitor will detect such changes. It must however be stated that if an incident changes the amplitude of oscillation of a signal such that the period of oscillation is not affected, then perhaps the monitor may be unable to detect such changes.

APPENDIX H

Distinguishing Between State Space Modeling in Time Series and State Modeling in a Markov chain

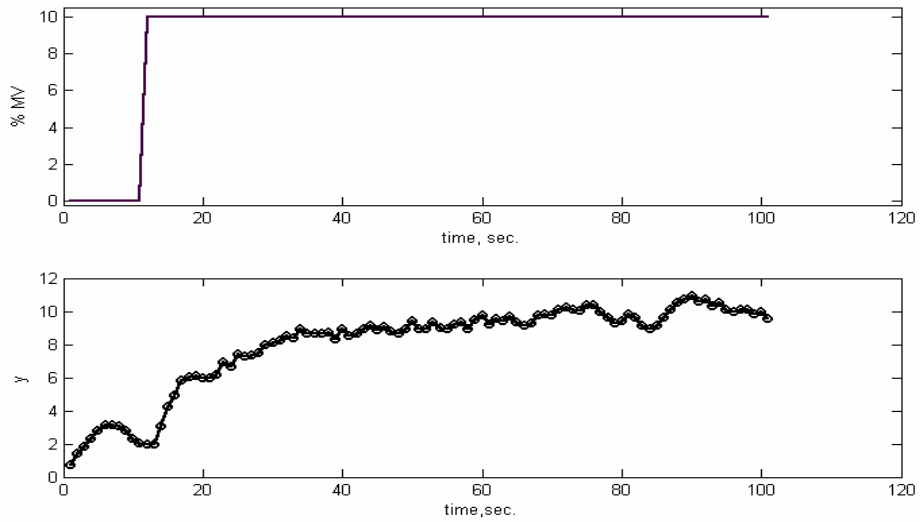


Figure H1 **Time Series of Controller Output Signal and Process response (**y = Process Response, %MV = Change in controller output signal (u))

Considering Figure H1, the process response values can be modeled using the Autoregressive Moving Average (ARMA) representation in Equation (H1) as

$$y_i = \underbrace{a_1 y_{i-1} + a_2 y_{i-2}}_{2^{nd}\ order\ AR} + \underbrace{b_0 u_i + b_1 u_{i-1}}_{2^{nd}\ order\ MA} + \underbrace{n_i}_{Noise} \qquad \text{(H1)}$$

Where $a's$ and $b's$ are coefficients and are determined by least square modeling. Without excitation (i.e. a change in u), the values of *b* cannot be estimated. If noise dominates process output *y*, and there is no real change in *y*, the values of *a*, cannot be

determined either. If a process is oscillating naturally, then Equation H1 will be expressed in values of *a*.

If all one knows is that a *y* value equals 1 say, then one cannot know whether it is on the up path (and the next value is likely 2) or on the down path (and the next y value is likely -1). However, if the history is known (i.e. the past *y* values), then one can confidently use the model in Equation H1 to predict the next *y* (with uncertainty of noise).

By Contrast, using run length as a state value as shown Figure H2, a state of +1 historically moves to a state of +2 with a 100% probability, a state of +3 moves to +4 with 66.67% probability. A state of +6 persists in that state with 50% probability.

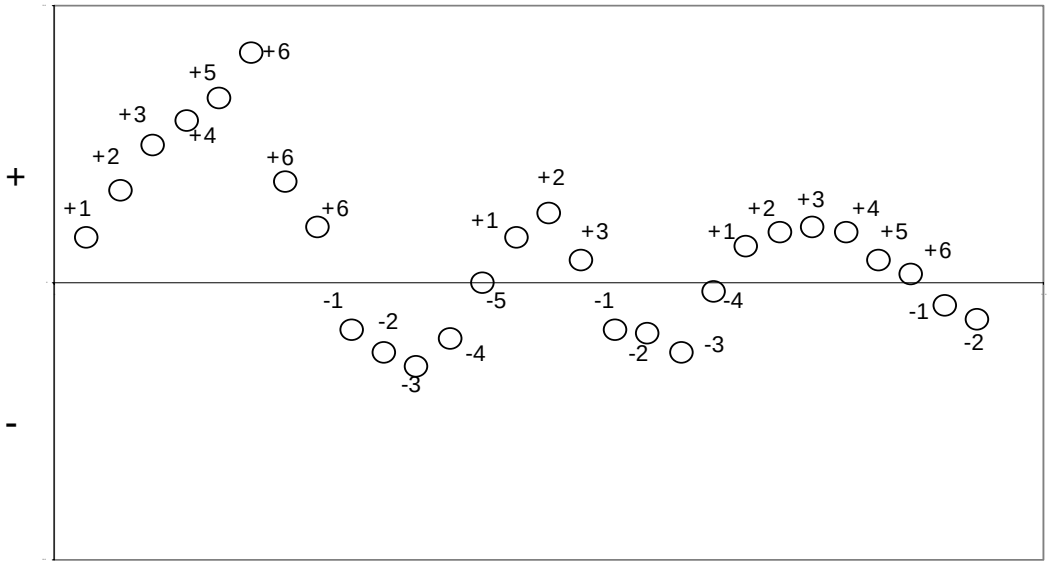


Figure H2 Modeling Run Length as States in a Markov chain

The state value +1, +2, … , +6 contains the history. A token cannot get to a state +6 without getting to state +1 then +2 … then +5. Therefore, in Markov chain state

modeling, the past value is inconsequential. Moreover, in this work, one is not looking for a relationship between y and u, so no excitation is required.

APPENDIX I

Glossary of Some Terminologies Used in this Work

Data: Actuating error or process-model mismatch.

Actuating error: The consecutive deviations of the desired variable from a target value (e.g. Setpoint – Controlled Variable). The units depend on the process and measurement.

Run length: The number of contiguous past data of like sign between consecutive zero crossings. A run length is dimensionless.

Zero Crossing: An event characterizing the switching of sign of the actuating errors from + to –, or – to +. In the special case where the error is equal to zero, it does not signify a zero crossing and the same sign as the prior state is maintained.

Transition: An event involving the change in state from k to j, say.

State: In addition to the definition of a state as given prior, a state is identified by the characteristics of the sign of the data (i.e. actuating error). A data characterized by a positive sign denotes a positive state. While, one characterized by a negative sign denotes a negative state. If the number of states were limited to a maximum value of +3 or –3 for instance, then a run length of 5 negative actuating errors would be in state of –3. A state has no units.

Transition Probability: The probability of moving from state k to state j at the next observation. The elements of the Transition probability must always satisfy the condition that $0 \leq P_{kj} \leq 1$. It has no units.

Count: The number of times a state has been occupied (cumulative state value). It has units of number of samples.

Window: Reference period of the past N samplings, which provide data for statistical comparison at each sampling). It has units of number of samples.

Sampling Ratio: The ratio of the number of actuating error samples from the controller to the number sampled by the health monitor for analysis. The units are Number of Controller Samples/Health Monitor Sample.

VITA

SAMUEL O. OWUSU

Candidate for the degree of Doctor of Philosophy Chemical Engineering

Dissertation: **A CONTROL LOOP PERFORMANCE MONITOR**

Major: Chemical Engineering

Biography:

Education: Bachelor of Science (Honors) Chemical**,** June 1994, University of Science and Technology, Kumasi, Ghana. Received a Master of Science, Chemical Engineering degree December 2001 from North Carolina Agricultural and Technical State University, Greensboro, North Carolina. Completed the requirement for the Doctor of Philosophy Degree at Oklahoma State University in May 2006.

Professional Experience: Worked for the Ghana National Petroleum Corporation from October 1995 till October 1998 as an Assistant Operations Officer moving on to the Market Research Department after one year as an Assistant Market Research Officer. Employed as a graduate teaching and Research Assistant by Department of Chemical Engineering, North Carolina A & T State University, Greensboro, North Carolina from 1998 to 2001. Consulted for College of Engineering Academic Enrichment Program as a supplementary Teaching Assistant.
Employed as a graduate research assistant by Oklahoma State University (OSU) School of Chemical Engineering in advanced process control from 2002 to present. Assisted in teaching transport phenomena and visual basic for application (VBA). Consulted for the College of Engineering Center for Academic Excellence as a supplementary teaching assistant in Physics, Computer Programming. Coordinated and organized control related seminar for the OSU College of Engineering Faculty.

Name: Samuel Odei Owusu Date of Degree: May, 2006

Institution: Oklahoma State University Location: Stillwater, Oklahoma

A CONTROL LOOP PERFORMANCE MONITOR

Pages in Study: 206 Candidate for the Degree of Doctor of Philosophy

Major Field: Chemical Engineering

Scope and Method of Study: Control Loop Monitoring Using Markov Chains and Binomial Statistics

Findings and Conclusions:

Good control practice is essential for industrial safety and profitability. Real time monitoring of industrial control processes continues to receive both industrial and academic attention due to the impact of control related faults on corporate bottom line.

Controllers are tuned for optimum performance but once tuned, process conditions change with time and what was once a good controller becomes a bad one unable to control the process effectively. It will be nice to automatically monitor controller performance so that operators can take immediate action without the tedium control loop monitoring. A number of control loop monitoring products have been developed for controller assessment but most of these programs operate offline, are cumbersome to understand or are themselves fraught with inherent shortcomings making their usage inefficient.

This work proposes a simple, efficient, and practicable method to automatically flag poor controller performance in real time. It uses only the run length of the actuating errors. Run length is defined as a state in a Markov chain, and transitions between states (which are binomially distributed) are then modeled using binomial statistics. Transition probabilities from operation are then compared with the control limits (estimated using binomial statistics) established from a user-defined period of good control.

Initial test results indicate that the program is effective and adaptable to numerous control configurations.

ADVISER'S APPROVAL: Dr. R. R. Rhinehart