CHAPTER 2

LITERATURE SURVEY AND PROPOSED STUDY

2.1 INTRODUCTION

Sequential Probability Ratio Test (SPRT) is an ongoing statistical analysis repeatedly conducted as data is collected. It is used in anomaly detection and decision making for electronics, structures and process controls. The data are repeatedly reassessed and a decision is made either to:

(1) Reject the null hypothesis and stop collecting data

(2) Fail to reject the null hypothesis and stop collecting data

(3) Continue collecting data until a decision regarding the null hypothesis can be reached.

The SPRT sets threshold boundaries, which take the form of parallel lines, one of which represents the expected outcome and the other a significantly different outcome. When the value of the calculated test statistic falls outside of these threshold boundaries, a conclusion can be drawn and data collection stops.

![Sequential Probability Ratio Test](Fig: 2.1.1 Sequential Probability Ratio Test)
2.2 PARAMETER ESTIMATION METHODS

Parameter estimation is of primary importance in software reliability estimation. Two most popular estimation techniques are Maximum Likelihood Estimation (MLE) and Least Squares Estimation (LSE). The MLE technique estimates parameters by solving a set of simultaneous equations. It is the most widely used estimation technique. In many cases, the maximum likelihood estimators are consistent and asymptotically normally distributed as the sample size increases (Zhao and Xie, 1996). In this thesis, the parameters are estimated by MLE technique for Time domain data.

Depending on the format in which test data are available, there are two common types of failure data: time-domain (i.e. ungrouped) data and interval-domain (i.e. grouped) data (Pham, 2006). The time-domain approach involves recording the individual times at which failure occurred. The interval-domain approach is characterized by counting the number of failures occurring during a fixed period (e.g., hour, week, day). These data are usually used by practitioners when analyzing, assessing and predicting reliability applications. Some software reliability models can handle both types of data.

**Time Domain Data**

Assuming that the data are given for the occurrence times of the failures or the times of successive failures, \( s_j \) for \( j = 1, 2, ..., n \). Given that the data provide \( n \) successive times of observed failures \( s_j \) for \( 0 \leq s_1 \leq s_2 \leq \cdots \leq s_n \), we can convert these data into the time between failures \( t_i \) where \( t_i = s_j - s_{i-1} \) for \( i = 1, 2, ..., n \). Given the recorded data on the time of failures, the Log Likelihood Function (LLF) takes on the following form (Pham, 2006):

The log likelihood function for Time domain data is given as

\[
LLF = \sum_{i=1}^{n} \log [\lambda (t_i)] - m(t_n) \quad (2.2.1)
\]
**Type 2: Interval Domain Data**

Assuming that the data are given for the cumulative number of detected errors \( y_i \) in a given time-interval \((0,t_i)\) where \( i = 1, 2, \ldots, n \) and \( 0 < t_1 < t_2 < \ldots < t_n \), then the LLF takes on the following form (Pham, 2006):

\[
LLF = \sum_{i=1}^{n} (y_i - m(t_i)) \log \left[ m(t_i) - m(t_{i-1}) \right] - m(t_n)
\]  

(2.2.2)

### 2.2.1 MLE

The principle of maximum likelihood estimation (MLE), originally developed by R.A. Fisher in the 1920s, states that the desired probability distribution is the one that makes the observed data “most likely,” which means that one must seek the value of the parameter vector that maximizes the likelihood function. The resulting parameter vector, which is sought by searching the multi-dimensional parameter space, is called the MLE estimate. The method of MLE is one of the most useful techniques for deriving point estimators. A MLE method is versatile and applies to many models and to different types of data. Although the methodology for maximum likelihood estimation is simple, the implementation is mathematically intense. For computational convenience, the MLE estimate is obtained by maximizing the log-likelihood function.

If we conduct an experiment and obtain \( N \) independent observations, \( T = t_1, t_2, \ldots, t_N \). The probability density function of \( k \) unknown parameters \( \theta_1, \theta_2, \ldots, \theta_k \) is given as \( f(t; \theta_1, \theta_2, \ldots, \theta_k) \). Assuming that the random variables are independent, then the likelihood function, \( L(T; \theta_1, \theta_2, \ldots, \theta_k) \), is the product of the probability density function evaluated at each sample point:

\[
L(t_1, t_2, \ldots, t_N | \theta_1, \theta_2, \ldots, \theta_k) = L = \prod_{i=1}^{N} f(t_i; \theta_1, \theta_2, \ldots, \theta_k)
\]
The maximum likelihood estimator $\hat{\theta}$ is found by maximizing $L(T; \theta_1, \theta_2, \ldots, \theta_k)$ with respect to $\theta$. The symbol (^) is used here to distinguish Maximum Likelihood estimators from the parameters being used.

Likelihood function by using $\lambda(t)$ is expressed as, $L = e^{-m(t)} \prod_{i=1}^{n} \lambda(t_i)$.

Taking the natural logarithm of the above equation, we can obtain $lnL$.

The logarithmic likelihood function is given by, $\log L = \log \left( e^{-m(t)} \prod_{i=1}^{n} \lambda(t_i) \right)$

The ML estimates (Cohen, 1965) of the unknown parameters $\theta_1, \theta_2, \ldots, \theta_k$ are obtained by maximizing and differentiating $lnL$ with respect to each of the unknown parameters. Solving the equations simultaneously and equating to zero.

$$\frac{\partial (\log L)}{\partial \theta_j} = 0; \quad j = 1, 2, \ldots, k.$$ 

2.2.2 MMLE

In order to overcome the numerical iterative way of solving the log likelihood equations and to get analytical estimators rather than iterative, some approximations to the estimating equations can be adopted from Kantam and Dharmarao (1994), Kantam and Sriram (2001). Several authors like Mehrotra and Nanda (1974), Cohen and Whitten (1980), Tiku et al. (1986), Tiku and Suresh (1992) obtained modified maximum likelihood (MML) estimates by making linear approximations to certain functions in estimating equations of parameters of normal, log-normal, logistic, exponential and Rayleigh distributions.

2.3 ORDER STATISTICS

To improve and understand the logic behind process control methods, it is necessary to give some thought to the behavior of sampling. If the length of a single failure interval is measured, it is clear that occasionally a length will be found which is towards one end of the tails of the process’s normal distribution.
This occurrence may lead to the wrong conclusion that the process requires adjustment. If a sample of four or five is taken, it is extremely unlikely that all four or five failure interval lengths will lie towards one extreme end of the distribution. If we take the average or length of four or five failure intervals, we shall have a much more reliable indicator of the state of the process. Any change in the process mean, unless it is extremely large, will be difficult to detect from individual results alone. A large number of individual readings are necessary before such a change was confirmed.

The distribution of sample means reveals the change much quicker than individuals. Therefore, on a chart for sample means, plotted against time, the change in level would be revealed almost immediately. For this reason sample means rather than individual values are used to control the centering of processes.

A subgroup or a sample is a small set of observations on a process parameter or its output, taken together in time. The size and the frequency of sampling are the two major problems in choosing a subgroup. The smaller the subgroup, there is less opportunity for variation within it. The larger the sample size the narrower the distribution of the means and they become more sensitive to detect change (Oakland, 2008).

It is understood that, in any type of process control, careful selection of subgroups is very important. The software failure data is in the form of <failure number, failure time>. By grouping a fixed number of data into one, the noise values may compensate each other for that period and thus the noise inherent in the failure data is reduced to great extent (Malaiya et al., 1990).

2.4 SPRT PROCEDURE

Wald's procedure is particularly relevant if the data is collected sequentially. Sequential Analysis is different from Classical Hypothesis Testing were the number of cases tested or collected is fixed at the beginning of the experiment. In Classical Hypothesis Testing the data collection is executed without analysis and consideration of the data. After all data is collected the analysis is done and conclusions are drawn. However, in Sequential Analysis every case is analyzed directly after being collected, the data collected upto that moment is then compared with certain threshold values, incorporating the new
information obtained from the freshly collected case. This approach allows one to draw conclusions during the data collection, and a final conclusion can possibly be reached at a much earlier stage as is the case in Classical Hypothesis Testing. The advantages of Sequential Analysis are easy to see. As data collection can be terminated after fewer cases and decisions taken earlier, the savings in terms of human life and misery, and financial savings, might be considerable.

In the analysis of software failure data we often deal with either Time Between Failures or failure count in a given time interval. If it is further assumed that the average number of recorded failures in a given time interval is directly proportional to the length of the interval and the random number of failure occurrences in the interval is explained by a Poisson process then we know that the probability equation of the stochastic process representing the failure occurrences is given by a Homogeneous Poisson Process with the expression

$$P[N(t) = n] = \frac{e^{-\lambda t} (\lambda t)^n}{n!}$$

(2.4.1)

Stieber (1997) observes that if classical testing strategies are used, the application of software reliability growth models may be difficult and reliability predictions can be misleading. However, he observes that statistical methods can be successfully applied to the failure data. He demonstrated his observation by applying the well-known sequential probability ratio test (SPRT) of Wald (1947) for a software failure data to detect unreliable software components and compare the reliability of different software versions. In this paper we consider popular SRGM HLSRGM and adopt the principle of Stieber (1997) in detecting unreliable software components in order to accept or reject the developed software. The theory proposed by Stieber (1997) is presented in Section 2.4.1 for a ready reference. Extension of this theory to the SRGM – HLSRGM is presented in Section 2.4.2. Application of the decision rule to detect unreliable software with respect to the proposed SRGM is given in Chapter 3, Chapter 4, Chapter 5 and Chapter 6, where the parameters are estimated using MLE, MMLE and Order Statistics approaches.

2.4.1 Wald’s Sequential Test for a Poisson Process

The sequential probability ratio test (SPRT) was developed by A.Wald at Columbia University in 1943. Due to its usefulness in development work on
military and naval equipment it was classified as ‘Restricted’ by the Espionage Act (Wald, 1947). A big advantage of sequential tests is that they require fewer observations (time) on the average than fixed sample size tests. SPRTs are widely used for statistical quality control in manufacturing processes. An SPRT for homogeneous Poisson processes is described below.

Let \( \{N(t), t \geq 0\} \) be a homogeneous Poisson process with rate \( \lambda \). In our case, \( N(t) \) = number of failures up to time \( t \) and \( \lambda \) is the failure rate (failures per unit time). Suppose that we put a system on test (for example a software system, where testing is done according to a usage profile and no faults are corrected) and that we want to estimate its failure rate \( \lambda \). We can not expect to estimate \( \lambda \) precisely. But we want to reject the system with a high probability if our data suggest that the failure rate is larger than \( \lambda_1 \) and accept it with a high probability, if it’s smaller than \( \lambda_0 \). As always with statistical tests, there is some risk to get the wrong answers. So we have to specify two (small) numbers \( \alpha \) and \( \beta \), where \( \alpha \) is the probability of falsely rejecting the system. That is rejecting the system even if \( \lambda \leq \lambda_0 \). This is the "producer’s" risk. \( \beta \) is the probability of falsely accepting the system. That is accepting the system even if \( \lambda \geq \lambda_1 \). This is the “consumer’s” risk. With specified choices of \( \lambda_0 \) and \( \lambda_1 \) such that \( 0 < \lambda_0 < \lambda_1 \), the probability of finding \( N(t) \) failures in the time span \( (0, t) \) with \( \lambda_1, \lambda_0 \) as the failure rates are respectively given by

\[
Q_1 = \frac{e^{-\lambda_1 t} [\lambda_1 t]^{N(t)}}{N(t)!} \tag{2.4.1.1}
\]

\[
Q_0 = \frac{e^{-\lambda_0 t} [\lambda_0 t]^{N(t)}}{N(t)!} \tag{2.4.1.2}
\]

The ratio \( \frac{Q_1}{Q_0} \) at any time \( t \) is considered as a measure of deciding the truth towards \( \lambda_0 \) or \( \lambda_1 \), given a sequence of time instants say \( t_1 < t_2 < t_3 < \ldots < t_k \) and the corresponding realizations \( N(t_1), N(t_2), \ldots, N(t_k) \) of \( N(t) \). Simplification of \( \frac{Q_1}{Q_0} \) gives

\[
\frac{Q_1}{Q_0} = \exp(\lambda_0 - \lambda_1) + \left(\frac{\lambda_1}{\lambda_0}\right)^{N(t)}
\]

The decision rule of SPRT is to decide in favor of \( \lambda_1 \), in favor of \( \lambda_0 \) or to continue by observing the number of failures at a later time than \( t \) according as

37
\( \frac{Q_1}{Q_0} \) is greater than or equal to a constant say \( A \), less than or equal to a constant say \( B \) or in between the constants \( A \) and \( B \). That is, we decide the given software product as unreliable, reliable or continue the test process with one more observation in failure data, according as

\[
\frac{Q_1}{Q_0} \geq A \quad (2.4.1.3)
\]

\[
\frac{Q_1}{Q_0} \leq B \quad (2.4.1.4)
\]

\[
B < \frac{Q_1}{Q_0} < A \quad (2.4.1.5)
\]

The approximate values of the constants \( A \) and \( B \) are taken as \( A \approx \frac{1 - \beta}{\alpha} \), \( B \approx \frac{\beta}{1 - \alpha} \)

Where ‘\( \alpha \)’ and ‘\( \beta \)’ are the risk probabilities as defined earlier. A simplified version of the above decision processes is to reject the system as unreliable if \( N(t) \) falls for the first time above the line \( N_U(t) = a t + b_2 \) \hfill (2.4.1.6)

To accept the system to be reliable if \( N(t) \) falls for the first time below the line \( N_L(t) = a t - b_1 \) \hfill (2.4.1.7)

To continue the test with one more observation on \( (t, N(t)) \) as the random graph of \( [t, N(t)] \) is between the two linear boundaries given by equations (2.4.1.6) and (2.4.1.7) where

\[
a = \frac{\lambda_1 - \lambda_0}{\log \left( \frac{\lambda_1}{\lambda_0} \right)} \hfill (2.4.1.8)
\]

\[
b_1 = -\frac{\log \left( \frac{1 - \alpha}{\beta} \right)}{\log \left( \frac{\lambda_1}{\lambda_0} \right)} \hfill (2.4.1.9)
\]

\[
b_2 = -\frac{\log \left( \frac{1 - \beta}{\alpha} \right)}{\log \left( \frac{\lambda_1}{\lambda_0} \right)} \hfill (2.4.1.10)
\]
The parameters $\alpha, \beta, \lambda_0$ and $\lambda_1$ can be chosen in several ways. One way suggested by Stieber (1997) is $\lambda_0 = \frac{\lambda \log(q)}{q-1}$, $\lambda_1 = q \cdot \frac{\lambda \log(q)}{q-1}$, where $q = \frac{\lambda_1}{\lambda_0}$.

If $\lambda_0$ and $\lambda_1$ are chosen in this way, the slope of $N_U(t)$ and $N_L(t)$ equals $\lambda$.

The other two ways of choosing $\lambda_0$ and $\lambda_1$ are from past projects and from part of the data to compare the reliability of different functional areas.

2.4.2 Sequential Test for SRGMS

In Section II, for the Poisson process we know that the expected value of $N(t) = \lambda t$ called the average number of failures experienced in time 't'. This is also called the mean value function of the Poisson process. On the other hand if we consider a Poisson process with a general function $m(t)$ as its mean value function the probability equation of such a process is

$$P[N(t) = Y] = \frac{[m(t)]^y}{y!} \cdot e^{-m(t)}$$

Depending on the forms of $m(t)$ we get various Poisson processes called NHPP.

We may write

$$Q_1 = \frac{e^{-m_1(t)}[m_1(t)]^{N(t)}}{N(t)!}$$

$$Q_0 = \frac{e^{-m_0(t)}[m_0(t)]^{N(t)}}{N(t)!}$$

Where, $m_1(t)$, $m_0(t)$ are values of the mean value function at specified sets of its parameters indicating reliable software and unreliable software respectively. Let $P_0, P_1$ be values of the NHPP at two specifications of $b$ say $b_0, b_1$ where $(b_0 < b_1)$ respectively. It can be shown that for our models $m(t)$ at $b_1$ is greater than that at $b_0$. Symbolically $m_0(t) < m_1(t)$. Then the SPRT procedure is as follows:

Accept the system to be reliable if $\frac{Q_1}{Q_0} \leq B$

i.e., $\frac{e^{-m_1(t)}[m_1(t)]^{N(t)}}{e^{-m_0(t)}[m_0(t)]^{N(t)}} \leq B$
\[
\log \left( \frac{\beta}{1-\alpha} \right) + m_i(t) - m_o(t) \leq \frac{\log \left( \frac{1}{\beta} \right) + m_i(t) - m_o(t)}{\log m_i(t) - \log m_o(t)}
\]

i.e., \( N(t) \leq \frac{\log \left( \frac{1}{\beta} \right) + m_i(t) - m_o(t)}{\log m_i(t) - \log m_o(t)} \)

(2.4.2.1)

Decide the system to be unreliable and reject if \( \frac{\bar{Q}}{Q_0} \geq A \)

\[
\log \left( \frac{1}{\beta} \right) + m_i(t) - m_o(t) \geq \frac{\log \left( \frac{1}{\beta} \right) + m_i(t) - m_o(t)}{\log m_i(t) - \log m_o(t)}
\]

i.e., \( N(t) \geq \frac{\log \left( \frac{1}{\beta} \right) + m_i(t) - m_o(t)}{\log m_i(t) - \log m_o(t)} \)

(2.4.2.2)

Continue the test procedure as long as

\[
\log \left( \frac{1}{\beta} \right) + m_i(t) - m_o(t) < \frac{\log \left( \frac{1}{\beta} \right) + m_i(t) - m_o(t)}{\log m_i(t) - \log m_o(t)}
\]

(2.4.2.3)

Substituting the appropriate expressions of the respective mean value function \(-m(t)\) of HLSRGGM, we get the respective decision rules and are given in followings lines

Acceptance region:

\[
N(t) \leq \log \left( \frac{1}{\beta} \right) + \frac{2a \left( e^{-b_f} - e^{-b_f(t)} \right)}{1 + e^{-b_f} + e^{-b_f(t)} + e^{-b_f(t) - b_f}}
\]

(2.4.2.4)

Rejection region:

\[
N(t) \geq \log \left( \frac{1}{\beta} \right) + \frac{2a \left( e^{-b_f} - e^{-b_f(t)} \right)}{1 + e^{-b_f} + e^{-b_f(t)} + e^{-b_f(t) - b_f}}
\]

(2.4.2.5)

Continuation region:

\[
N(t) < \log \left( \frac{1}{\beta} \right) + \frac{2a \left( e^{-b_f} - e^{-b_f(t)} \right)}{1 + e^{-b_f} + e^{-b_f(t)} + e^{-b_f(t) - b_f}}
\]

(2.4.2.6)

It may be noted that in the above model the decision rules are exclusively based on the strength of the sequential procedure \((\alpha, \beta)\) and the values of the
respective mean value functions namely, \( m_0(t), m_1(t) \). If the mean value function is linear in ‘t’ passing through origin, that is, \( m(t) = \lambda t \) the decision rules become decision lines as described by Stieber (1997). In that sense equations (2.4.2.1), (2.4.2.2), (2.4.2.3) can be regarded as generalizations to the decision procedure of Stieber (1997).

2.5 RELEVANT LITERATURE

This section presents a detailed survey of the literature that lead to carry out this research work grouped under the topics such as, Software reliability and their growth models, parameter estimation methods, order statistics and SPRT. This survey helped me to trace out some research problems for further investigation that formed my research work. The brief contents of such research articles are presented in chronological order till date as far as possible.

Generally, software reliability growth models (SRGMs) are composed of both Analytical and Data-driven models. Analytical SRGMs have three major sub-categories: Non-Homogeneous Poisson Process (NHPP) models, Markov models, and Bayesian models. They are constructed by analyzing the dynamics of the software failure process, and their applications are developed by fitting them against software failure data. To provide mathematical tractability, traditional SRGMs only model fault-detection process with the assumption of perfect and immediate fault correction.

SPRT require substantially fewer observations than fixed sample size test. In sequential testing it is not only the observations \( X \), that are random variables, but also the number of observations, \( K \). Inspired by the Neyman-Pearson lemma (1933), which provides a method of constructing a most powerful statistical test for deciding between two simple hypotheses, Wald (1947) proposed the SPRT. It is designed to decide between two simple hypotheses. Sequential analysis is a type of statistical inference where the sample size is not set in advance. Rather, data are drawn and evaluated sequentially, until there is enough evidence to reach a conclusion within a desired level of assurance. Sequential procedures require, on average, substantially fewer observations than their equally statistically powerful
The contrast is best exemplified by Wald’s famous Sequential Probability Ratio Test (SPRT). Compared to the (fixed-sample-size) Neyman-Pearson test, SPRT’s sample size may be as much as 50% smaller.

The sequential probability ratio test (SPRT) has the remarkable property that among all sequential tests procedures it minimizes the expected number of samples for a given level of certainty and regardless of which hypothesis is true (Wald and Wolfowitz, 1948).

The SPRT was initially developed for situations in which there is a random sample of a variable with a discrete or continuous distribution with one parameter variable and two simple hypotheses on the value of that parameter. The sequential approach (Wald, 1947; Siegmund, 1985) is often used in many applications and quality control, because of two reasons. 1) For some observation schemes it is natural to follow the sequential approach to construct a probability model. 2) Sequential statistical procedures have some optimal properties, e.g. SPRT minimizes the expected sample size (Ghosh, 1970).

Schick and Wolverton (1973) describe the most commonly used software reliability growth models as divided into two groups of time domain and data domain.

Based on the notion of the distribution function of random variable Forman and Singpurwalla (1977) developed a probabilistic model describing the software failure phenomenon to suggest estimates of the parameters in the model and termination procedure for debugging the software.

For non-iid case, Lai (1981) has shown us a large sample result for the SPRT, when the observations satisfy a “slowly changing sequence” condition. It is an asymptotical result and provides only little information for how to make an efficient mastery test.

In Reckase (1983) and Spray (1993), they have done some numerical studies of the performance of the SPRT under tailored test set-up for both the mastery and the multiple category criterion-referenced tests.
Balakrishnan (1985) has introduced half logistic distribution by folding the well known logistic distribution at its median. His main contribution in the paper is tabulation of means, variances and covariances of order statistics in samples drawn from half logistic distribution. The modes of all order statistics, percentiles of the extreme order statistics are also given. Okumoto (1985) proposed a statistical method that can be used to monitor, control and predict the quality of a software system being tested. The method consists of three steps: estimation of the failure, intensity based on groups of failures, fitting the logarithmic poison model to the estimated failure intensity data, and constructing confidence limits for the failure intensity process. Amrit L. Goel (1985) presented an overview of the key modelling approaches provided a critical analysis of the underlying assumptions, and assess the limitations and applicability of software reliability models during the software development life cycle.

Balakrishnan and puthenura (1986) obtained the coefficients to get the best linear unbiased estimates of location and scale parameters in half logistic distribution.

Yasshi Masuda et al. (1989) introduced an objective function which incorporates average cost, time dependent value of the software system and the cumulative running cost. Ehrlich et al. (1990) used the software reliability data collected during the testing of a system to measure the software quality in terms of experienced software failures.

Frederique M. Vallee et al. (1991) presented how NHPP approach can be applied to the industrial world. Approximate ML estimation of location and scale parameters in half logistic model is considered by Balakrishnan and Wong (1991). Lyu and Nikora (1991a) proposed linear combinations of software reliability models for the purpose of automating the procedures of software reliability analysis. Lyu and Nikora (1991b) suggested the concept of equally weighted linear combination model resulting from linear combination of three popular software reliability models.

Zhao M. et al. (1992) studied a simple software reliability model, the log-power NHPP. Balakrishnan and Chan (1992) considered a scaled half logistic
distribution and developed theory of linear estimation for its scale parameter in small and large samples.

Hossain A. Syed et al. (1993) presented a necessary and sufficient condition for the likelihood estimates to be finite, positive and unique and suggested a modification to Goel and Okumoto model. Kantam and Dharmarao (1993) suggested a modification to estimate the scale parameter of half logistic distribution in ML method of estimation to get simpler and more efficient estimator.

Rosaiah et al. (1993a and 1993b) considered the modified ML (MML) estimation in gamma distribution with and without prior relations between its parameters. Other similar works include Kantam and Srinivasa Rao (1993) in Rayleigh distribution from left censored samples, Kantam and Srinivasa Rao (2002) extended MML estimation to log-logistic distribution.


Stieber (1997) insisted application of statistical methods to failure data, presented an application which allows the detection of unreliable software components and the comparison of the reliability of different software versions.

Reynolds and Stoumbos (1998) and Stoumbos and Reynolds (2001) utilized SPRT to generate a statistical process control model to monitor the process changes.

Gross et al. used SPRT to monitor anomalies in computer servers (Gross and Lu, 2002; Whisnant, Gross and Lingurovska, 2005) and equipment in nuclear plants (Gross and Humenik, 1991).
Carina Anderson (2003) focuses on exploring the state of practice of the verification and validation process and presented methods for achieving efficient fault detection during the software development.

Keiller and Mazzuchi (2005) suggested some methods of comparing software reliability models using smoothing techniques. Torabi and Behboodian (2005) redefined some concepts about fuzzy hypotheses testing, and then applied give the sequential probability ratio test for fuzzy hypotheses testing with fuzzy observations.

Several authors have proposed SPRT for fuzzy case. Neyman Pearson lemma for fuzzy hypothesis testing was given by Taheri and Behboodian(2006). Talukdar and Baruah(2007) have fuzzified SPRT. Sheta (2007) explored the use of Particle Swarm Optimization (PSO) algorithm to estimate SRGM parameters. Expectation Maximizing (EM) algorithm is a general technique for MLE. Li et al. (2007) employed the algorithm to estimate the parameters of the software reliability model.

Torabi and Mirhosseini(2009) introduced a SPRT for fuzzy hypothesis testing. They developed fuzzy SPRT for testing the mean of Normal distribution with known variance and the parameter of Bernoulli distribution. Akbari(2011) proposed a new approach for SPRT of fuzzy hypothesis under density probability function.

David spiegelhalter et al. (2003) investigated the use of the risk-adjusted sequential probability ratio test in monitoring the cumulative occurrence of adverse clinical outcomes.

Krishna Mohan et al.,(2011) worked on estimating the parameters using maximum likelihood estimation for Rayleigh SRGM for both time domain and interval domain data. They made use of the parameters to know, whether the software under consideration is within control or out of control by analyzing the failure data using statistical process control and sequential probability ratio test.

Liu (2011) proposed a function based nonlinear least squares estimation method which extends the potential fitting functions of traditional least square
estimation in estimating parameters of Jelinski-morando reliability model. Kulldorff et al. (2011) proposed a maximized sequential probability ratio test (MaxSPRT) based on a composite alternative hypothesis, which works well across a range of relative risks. They illustrated the use of this method on vaccine safety surveillance and compare it with the classical SPRT.

Specifically, with the evolution of SRGMs, imperfect debugging has been incorporated into the modeling framework from many aspects. Satyaprasad et al. (2011) also assumed the testing as imperfect, and worked on exponential imperfect debugging model to the software quality.

Krishna mohan et al. (2012) made a performance comparison of exponential and Rayleigh SRGMs. In wireless sensor networks, there are many nodes which can be captured and compromise the sensor nodes and take secret key from the nodes then make many replicas of them, degrades the network communication. To avoid this node compromised attack, sequential probability ratio testing (SPRT) is used well in mobile sensor networks. Nidharshini and Janani (2012) detected the compromised node in mobile sensor networks.

Cheng, S and Pecht (2012) presented a systematic method to select model parameters for the sequential probability ratio test by using a cross-validation technique. This method can improve the accuracy of the sequential probability ratio test by reducing the false and missed alarm probabilities caused by improper model parameters.

One of the important fields in statistics is testing hypothesis of correlation coefficient. The extension of the idea of testing correlation to fuzzy hypothesis is of great interesting. Sevil Bacanli and Duygu Icen (2013) examined the use of fuzzy hypothesis testing approach for the Sequential Probability Ratio Test (SPRT) of correlation coefficient.

The use of maximum likelihood estimation, modified maximum likelihood estimation in assessing the quality of the software using the exponential SRGM is the work of Srinivasa rao(2013). Use of two step approach in estimating the parameters for Half logistic (Shaheen, 2013) and exponential (Murali mohan,
SRGMs is the work carried out in assessing the quality of the software product.

2.6 PROPOSED STUDY

Accordingly, the various research publications related to Half logistic distribution with reference to problems of statistical inference applied to quality control and reliability are carefully scanned for possible further investigations. After thoroughly scanning the literature presented above, we are motivated to study the following research problems related to assessing software reliability using Sequential Probability Ratio Test as these seem to be unattempted / unavailable in published form.

(i) We propose a method of applying SPRT on the inter failure cumulative data between observations of failure. The said method can be easily and fruitfully applied to monitor the software failure process for Half Logistic Distribution based Non-Homogeneous Poisson Process (NHPP). Maximum Likelihood Estimation (MLE) is used for parameter estimation.

(ii) We also propose a method of applying SPRT on the cumulative failure data which involves evaluation of parameters for Half Logistic Distribution using Modified Maximum Likelihood Estimation (MMLE).

(iii) We also propose a method based on order statistics of cumulative quantity between observations of time domain failure data using mean value function of HLD based on NHPP, if waiting for fixed number of failures is acceptable.

(iv) We also propose a method of applying SPC on the Interval domain cumulative failure data. The said method can be easily and fruitfully applied to monitor the software failure process for Half Logistic Distribution based Non-Homogeneous Poisson Process (NHPP). Maximum Likelihood Estimation (MLE) is used for parameter estimation.
The research findings out of our proposed study are presented with detailed discussion in the following chapters. Reprints of some of our results published in standard journals are appended towards the end of the thesis, with relevant reference in the introduction of the respective chapters.