CHAPTER 2

2.1 Literature Survey and Proposed Study

This section presents a detailed survey of the research literature that lead to carry out this research work grouped under the topics such as, software reliability and their growth models, NHPP, SPC and SPRT. This survey helped me to trace out some research problems for further investigation that formed my research work.

Page (1954) developed CUSUM charts using Wald’s sequential testing theory. 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.

Trivedi and Shooman (1975) suggested a many state Markov model for the purpose of estimation and prediction of software performance parameters. The availability and reliability of the software system are also presented for the purpose of predictions.

Schick and Wolverton (1978) described the most commonly used software reliability growth models as divided into two groups of time domain and data domain. Littlewood (1975, 1979) proposed software reliability models with Markov and modular program structures. Cheung (1980) developed user oriented software reliability model. Musa (1980) worked out the need for potential use of software reliability measurement and made a comparison of software and hardware reliabilities.

Goel and Okumoto (1979) considered the probabilistic nature of software failure phenomenon based on an NHPP. They have analyzed the failure process to develop a suitable mean value function which in turn is used to get software performance measures like distribution of the cumulative number of software errors, software reliability, and maximum likelihood estimation of the parameters of the model and joint confidence region of the parameters. Boehm et al. (1975) described the experience with automated aids to the design of large scale reliable software.
Shooman model (1976) considered reliability of modular software introducing the path based approach by using the frequencies with which different paths are run. Schick and Wolverton (1978) described the most commonly used software reliability growth models as divided into two groups of time domain and data domain.

Musa (1980) presented the need for potential use of software reliability measurement and made a comparison of software and hardware reliabilities. Lai (1981) has shown a large sample result for the SPRT, when the observations satisfy a “slowly changing sequence” condition. Ramamurthy and Bastani (1982) reviewed the status and perspectives of software reliability as on 1982. 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.

Lyu and Nikora (1991) proposed linear combination of software reliability models for the purpose of automating the procedures of software reliability analysis. Singpurwalla (1991) developed an optimal stopping solution for a single stage software system. The decision is based on a utility function. Two utility functions were suggested, one based on cost and the other being the realized reliability of the software. Obviously a single test case was used. An NHPP based SRGM influenced by error removal phenomenon is suggested by Kapur and Garg (1992) along with its predictive validity. Malaiya et al. (1992) proposed the choice of an appropriate software reliability model on the basis of a predictability measure.

Eric Martinus Mathiesen (1994) studied and presented that by combining accurate real time data collection, SPC methods and a reliable simulation program, system engineers will be better able to realize real savings in monetary terms, increased operational availability and decreased mission time. Macgregor J.F et. al (1995) presented advantages of SPC for continuous and batch multivariate processes. Ronald P Anjard (1995) presented a sampled method, a fool-proof, for selection of the appropriate SPC chart. Henry R Neave and Donald J Wheeler (1996) examined the difference between Shewhart’s charts and those of the academic version of the control charts.
Lynn and Yang (1995) proposed software reliability model with Bayesian computation. Nara et al. (1996) report the suitability and stability of the NHPP and trend curve SRGMs to actual software development projects. Wood (1996) has considered a panel of eight SRGMs in order to estimate software reliability on the basis of coded data on four releases of developed software. Aroui and Soler (1996) proposed a Bayesian approach for software reliability prediction in a nonparametric scenario.

Iannino et al. (1984) gave a descriptive narration of various criteria for the comparison of software reliability models based on predictive validity, quality of assumptions, capability, applicability, and simplicity. Matsumoto et al. (1988) discuss the evaluation procedure of a SRGM using data from a single program testing process applied to exponential, hyper exponential and S-shaped models, ranking the S-shaped model as superior with respect to estimation.


Demmy and Petrini (1989) presented the major elements of SPC system, the steps required to apply SPC to software development activities and summarized the major advantages and disadvantages of SPC approach to software development. Fault density and failure intensity of an SRGM are used as two metrics to monitor software development capabilities and to measure customer satisfaction with the developed product in the research investigation of Huensch et al. (1990).

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. Malaiya et al. (1990) tried to smooth the noise by data grouping in preprocessing the failure data. They also tried some other methods like, windowing and data dependent grouping. The time to repair, mean logistic delay time variates at the system level are modeled by Crow (1990).
Singpurwalla (1991) and Dallal and Mallows (1998), both proposed a Bayesian decision theoretic approach. Dallal and Mallows provides an exact but a complicated solution, thereby an asymptotic solution. Singpurwalla addressed a two stage problem where the solution is complicated by pre-posterior analysis. Ehrlich et al. (1991) analyzed individual fault failure sequence resulting from delayed fault detection and software correction during test execution to conform the applicability of a Poisson process.

Tohma et al. (1991) investigated six ways of the estimation of parameters in a hypergeometric distribution to get an estimate of number of initial faults in a program at the beginning of its testing/debugging, along with their relative accuracies. Sofer and Miller (1991) used a nonparametric method of estimating the software failure rate in completely monotone models which can be compared with parametric approaches. Vallee and Ragot (1991) demonstrated the application of NHPP approach to the industrial world generating accurate predictions with specific applications in space research.

Stott (1991) discusses the use of Shewart control charts for monitoring the defect density at each stage of software development. Lantzy (1992) examined some of the principles of statistical process control that have been successfully applied to a variety of manufacturing processes and offers a set of transformations on these principles that permit their application to software process. Khoshgoftaer et al. (1992) suggested an SRGM with the help of curve fitting techniques, predicting the number of faults in a system through fitting nonlinear regression models to the number of faults in a program module.

Zhao and Xie (1992) proposed a time dependent delay function, by arguing that detected faults become harder to correct with test in effect. Downs and Scott (1992) quantified and compared the performance of software reliability models with respect to a measure that helps to prefer one model to the other. Kenny (1993) proposed an SRGM useful for operational use and it can estimate the residual number of defects, anticipated arrival time of customer reported failure taking into the concept of —power function of time‖ associated with Weibull distribution. Necessary and sufficient conditions for the existence and finiteness of the MLEs of the parameters of SRGMs are derived by Hossain and Dahiya (1993).
Yamada et al. (1993) suggested an SRGM taking into consideration, the notion of software reliability during testing phase. Hossain and Dahiya (1993) derived necessary and sufficient conditions for the existence and finiteness of the MLEs of the parameters of SRGMs and applied to various standard models. Crow (1993) suggested confidence interval procedures for power law based on NHPP SRGM. Luke et al. (1993) suggested how SPC can be applied to improve the maintenance of CMS-2 software developed for a naval military fleet. Kapur et al. (1994) incorporated the concept of imperfect debugging with testing effort in a software reliability model that can be used to plan the amount of testing effort required to achieve a predetermined target in terms of errors removed in a given span of time.

Zeephongsekul et al. (1994) have introduced an SRGM with imperfect debugging and gave a method of MLE to estimate its parameters. Nara et al. (1996) report the suitability and stability of the NHPP and trend curve SRGMs to actual software development projects. Quality control limit curves are also suggested for the effect of application of the models. Wood (1996) has considered a panel of eight SRGMs in order to predict software reliability on the basis of coded data on four releases of developed software using the principle of least squares. Of the eight models the most popular exponential model - Goel and Okumoto (1979) was the choice for prediction purposes.

Aroui and Soler (1996) studied a Software reliability prediction in a nonparametric scenario through Bayesian approach. Haworth (1996) investigates the use of regression control charts for software maintenance in a medium sized manufacturing firm. Shima et al. (1997) examined the failure intensity distribution of an SRGM from an empirical failure data and show that the procedure predicts number of detected faults more accurately than the conventional ones. Xie and Goh (1997) proposed that the exact probability control limits, which are calculated from the Poisson distribution and the binomial distribution, can be applied to monitor attribute quality characteristics. Safety critical analysis applied in reliability modelling in order to increase the trust in the reliability of such products is the contribution of Schneidewind (1997) by integrating software risk analysis, safety criteria, reliability prediction and stopping rule for testing.
Gokhale et al. (1998) suggested software reliability analysis with fault detection and debugging. The study is based on the estimates of residual number of faults in the software. Pointing out the inadequacy of some SRGMs in describing the failure process of a software failure data, Gokhale and Trivedi (1998) propose the mean value function of an NHPP as given by a log logistic model that can indicate the increasing as well as decreasing nature of the failure occurrence. Hong et al. (1999) proposed a statistical method to control the software defect detection process together with further defect prevention analysis.

Pham and Pham (2001) studied the predictive performance of a software reliability model inserting pseudo failures based on Bayes approach. Chang (2001) proposed a NHPP model to analyze the change-point software reliability. Keiller and Mazzuchi (2002) studied the performance of a set of SRGMs using smoothing techniques including Laplace trend test. Huang and Kuo (2003) described a unified scheme of estimation in an SRGM using weighted arithmetic mean, weighted harmonic mean and weighted geometric mean. Their unified approach was verified to cover many of the well known NHPP models under different sets of conditions. Shyur (2003) developed a software reliability growth model that incorporates with both imperfect debugging and change-point problem. Spiegelhalter et al. (2003) investigated the use of the risk-adjusted sequential probability ratio test in monitoring the cumulative occurrence of adverse clinical outcomes.

Groen et al. (2004) proposed reliability data collection and analysis system employing Bayesian techniques to estimate reliability measures. Gokhale and Mullen (2004) have considered a Poisson process with log normal distribution as the mean value function and presented goodness of fit tests for software failure data. Crow (2005) gave a detailed note about the methods improving the effectiveness of the reliability tasks. R. Satya Prasad (2007) studied some problems of software reliability prediction and analysis when the random phenomenon in a software failure data with a half logistic distribution as means value functions. R. Satya Prasad et al. (2011) described sequential probability ratio test (SPRT) procedure to detect reliable/unreliable software components based on the theory proposed by Stieber (1997).
On the assumption of improvement in reliability along with time irrespective of bugs are corrected or not, a simple software reliability model is proposed by Jalote and Murphy (2004). A unified mathematical modelling for explaining the imperfect debugging in a software fault process by dividing the original failure process of the model into two different NHPPs is introduced by Lo and Huang (2004). The concept of infinite time to next failure is the general consideration of many SRGMs indicating that the software product is too good. A relaxation of this criterion is the summary of the study by Grottke and Trivedi (2005) to give a method for mending time to failure distributions.

When to terminate the software testing process and release the software is an important issue with software development. Such an optimal release time analysis framework has been broadly accepted and many research works have been conducted by extending it through different perspectives (Dohi et al., 1999; Teng and Pham, 2004; Huang and Lin, 2005). With different software reliability models combined with different release criteria, many papers such as Ross (1985), Dalal and Mallows (1988), Littlewood and Wright (1997), Kimura et al. (1999), Xie and Hong (1999), Zhang and Pham (2000), Dai et al. (2003), Jain and Priya (2005) dealt with this topic.

Cheong Wee Tat (2005) focused on the study of SPC techniques for high yield processes, and included some topics on high reliability systems. It deals with the statistical aspects of establishing SPC in high reliability systems, which provide insight and promising opportunity for future research on high reliability systems.

Sargut.K.U, Demirors.O (2006) performed a case study of the application of SPC techniques using existing measurement data in an emergent software organization. They also analyzed defect density, rework percentage and inspection performance inertial. And provided a practical insight as the usability of SPC for the selected metrics in the specific process and described their observations on the difficulties and the benefits of applying SPC to an emergent organization.

Mustsumi Komuro (2006) described experiences of applying SPC techniques to software development processes. They had given a number of real examples to apply SPC in Hitachi software. They also described characteristics of software
development processes, their influence on SPC, and lessons learned when applying SPC to software process. In particular, they discussed the importance of self-directed and proactive improvements. Liu Jiyiing (2006) made study to disadvantages of Shewhart attributes chart and time between events charts. The methods are presented for improving the performance of control charts which is more effective and economical.

Xie and Yang (2003), Bhaskar and Kumar (2006) have studied imperfect correction comprehensively. The concept of multiple change points and imperfect debugging are integrated in a single model to study the software reliability by Huang and Lin (2005).

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. Lyu et al. (2007) employed the algorithm to estimate the parameters of the software reliability model. Koutras et al. (2007) presented the basic principles and recent advances in the area of statistical process control charting with the aid of run rules. Pham (2007) proposed a software reliability model connecting the imperfect debugging and learning phenomenon by a common parameter between the two functions called the imperfect-debugging fault-detection dependent parameter model.

Watts S.Humphrey (2008) described that the first step in Statistical Process Management is to redefine the quality management strategy. To achieve high levels of software quality, it is necessary to switch from looking for defects to managing the process. Drumm Paul M (2009) presented a 5 step method for identifying process issues, and applying specifically to application performance issues. And also shown how to collect data related to the application performance issues. Furthermore shown how to use data with SPC in the form of control charts such as X bar/R bar to determine and identify trends and issues with application process performance. Also provided the guidelines resulting how an application development team gang their application from an SPC point of view and how to put the findings back into their development process to improve their application.
Hairulliza Mohammad Judi (2009) presented the implementation of quality control in three Malaysian companies and identified the factors that influence the selection of quality control techniques in these companies. Also it was identified that the initial role of SPC is to prevent product or process deterioration rather than identifying the product or process deterioration. Samul E. Buttrey (2009) presented a free add-in for excel that draws the most common sorts of control charts.

Jacob and Sreejith (2008) presented a control chart procedure for process reliability monitoring on Time between failure data to detect any change in intensity parameter, considering a single, ‘r’ failures through $\lambda$ chart and $\lambda_r$ chart.

Boffoli et al. (2008) formalized and kept a set of guidelines together in a disciplined process for guiding practitioners in correctly using SPC during process monitoring. Cao and Zhu (2010) proposed a forecasting method of software failure using fractals. 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 SPRT 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 compared it with the classical SPRT.

Monalessa Perini Barcellos, Ana Regina Rocha, Ricardo de Almeida Falbo (2010) presented an instrument for evaluating the suitability of measurement repositories in order to support software organizations implementing SPC. It was also highlighted that software organizations had increased their interest on Software Process Improvement (SPI). It was also mentioned that in high maturity levels, SPI involved implementing SPC which requires measures and data that are suitable for this context. Alain Abran, Alain april (2011) presented that SPC is an effective way to identify stability and interpreted through control charts.

After thoroughly scanning the literature presented in this Section, I am motivated to study the following research problems related to software reliability.
(i) Propose a Burr Type XII based software reliability growth model with Interval domain data and study the reliability assessment with Maximum likelihood estimation.

(ii) A Process control mechanism to assess the quality of a developed software based on the cumulative observations of Interval domain failure data using mean value function of Burr Type XII model.

(iii) An attempt to adopt the SPRT methodology of Stieber (1997) to our proposed Burr Type XII model with Interval domain data for reliable software using 11 distinct software failure data sets.

Our attempts and findings in these directions are presented in the chapters that follow with an aim of proposing analytical techniques for quality software.