CHAPTER -3

Research Methodology

Today, science and technology require high performance hardware and high quality software in order to make improvements and achieve breakthroughs. Software Reliability is an useful measure in planning and controlling the resources during the development process so that high quality software can be developed. Software Reliability was defined as the probability of software not causing failure of a system for a limited duration under specified conditions. Even though the definition looks very simple it constitutes a wide range of research activities with different sub activities. The sub activities are grouped in to number of fields, they are technological assessment of software reliability, quality concern of activity, management activity of project and selection of suitable software.

There are many probabilistic and statistical approaches to modelling software reliability. Software reliability estimates are used for various purposes: during development, to make the release decision; and after the software has been taken into use, as part of system reliability estimation, as a basis of maintenance recommendations, and further improvement, or a basis of the recommendation to discontinue the use of the software. The hardware reliability will continue to change even after the product is delivered however the software reliability continue to change as well as to improved throughout the process development and only before the product is delivered for application. Testing is a common assessment of hardware and software reliability techniques. The results of testing process are used in software growth models to formulate the defect and failure data. They are shown as software reliability measure grouped such as software reliability modelling and software testing.

Non Homogenous Poisson Process (NHPP) is a general class of well developed stochastic process model in reliability engineering. These models are also termed as fault counting models and can be either finite failure or infinite failure models, depending on how they are specified. In these models, the number of failures experienced so far follows the NHPP distribution. The NHPP model class is a close relative of the homogenous Poisson model, the difference is that here the expected...
number of failures is allowed to vary with time. Hence, they are useful for both calendar time data as well as for the execution time data. In this thesis, NHPP type of software reliability models and methods for estimating software reliability are used. The NHPP based SRGMs are proved to be quite successful in practical software reliability engineering. Many of the SRGMs assume that each time a failure occurs, the fault that caused it can be immediately removed and new faults are not introduced. It is usually called perfect debugging. The mean value function $m(t)$ is the characteristic of the NHPP model. Its generalization, the model under consideration in the thesis is the Burr type XII model. There are different approaches to modelling software reliability such as, Reliability growth models, Coverage-based models and Component-based models. In this thesis reliability growth models are focused. There is considerable statistical literature on modelling the reliability growth process of finding and fixing defects in a software product.

In this thesis the failure processes that have occurred in small computer programs were studied and the behaviour of the failures was investigated. The concept of well developed and widely applied to software reliability model to small computer programs. All the different data sets are shown in each Chapter. Statistical analysis was performed to all the date set to investigate the behaviour of the variable. In the Reliability analysis the concept of Burr type XII model were applied on data collected and estimating the parameter of the model was performed using maximum likelihood.

Generally, the SRGMs are classified into two groups. The first group contains models, which use machine execution time (i.e., CPU time) or calendar time as a unit of fault detection/removal period. Such models are called Continuous time models. Hence this type of models are also called failure count models. The second group contains models which use the number of test cases as a unit of fault detection period. Such models are called discrete time models, since the unit of software fault detection period is countable. A large number of models have been developed in the first group while there are fewer in the second group. In this thesis, we explore a broad class of NHPP models based on Continuous distributions. In case of interval domain data, it is consider k predetermined time intervals, denoted by $[t_{i-1}, t_i)$
for \( i = (1, \ldots, k) \). The failure data consists of the number of failures per time interval, denoted by \( y_i \) for \( i = (1, \ldots, k) \). The total number of failures is denoted by \( n_k \).

In this thesis the researcher considered two methodologies SPC and SPRT to achieve the reliability of software which has been described in respective Chapters. The Software reliability modelling uses statistical models for the previous historical data of similar projects are used for modelling parameters such as fault density; defect density and defect detection rate of the software that are being used. Some well established software reliability models are Musa execution time model, Goel – Okumoto NHPP model, Putnam model, Jelinski-Moranda model and little wood-verall model.

SPC is a methodology that aims to provide process control in statistical terms. It is to determine the study of how best one can describe and analyze the data and then draw conclusion based on available data. It is used to identify and eliminate errors in software development process and also to improve software reliability. The concepts of SPC are used to monitor the performance of a software process over time in order to verify that the process remains in the state of statistical control. It helps in finding assignable causes, long term improvements in the software process. Software quality and reliability can be achieved by eliminating the causes or improving the software process or its operating procedures.

Statistical Process Control (SPC) is about using control charts to manage software development efforts, in order to effect software process improvement. The practitioner of SPC tracks the variability of the process to be controlled. The early detection of software failures will improve the software reliability. The selection of proper SPC charts is essential to effective statistical process control implementation and use. The SPC chart selection is based on data, situation and need. Many factors influence the process, resulting in variability. The causes of process variability can be broadly classified into two categories, viz., assignable causes and chance causes. Critical business application requires reliable software, but developing reliable software is one of the most difficult problems facing the software industry. Therefore consistence and capability of the software is determined by using statistical and process control methods. This thesis reviews various research work.
performed about using the SPRT and SPC to measure and analyze the software reliability and the research concentrates on statistical techniques and the usage of SPC in Software Reliability.

In Sequential Probability Ratio Test (SPRT) the classical hypothesis testing, the data collection is executed without analysis and consideration of the data. After all the data is collected and analyzed, conclusions are drawn using sequential analysis. This is a method of statistical inference whose features is that number of observation required by the procedure is not determined in advance of the experiment. The decision to terminate the experiment depends, at each stage, on the results of the observations which are previously made. A merit of sequential method is applied to test the statistical hypothesis, that can be constructed which require, on the average of substantially smaller number of observation that equally reliable to test the procedure which is based on a predetermined number of observations.

The reliability prediction techniques are useful in the level of the software to be developed at early stages of development life cycle. A major problem of software reliability prediction model is that they fail to predict the reliability accurately. Reason for the same being its limitations to particular organization and particular product. Hence customization of particular model is not possible.