CHAPTER 3

PROBLEM STATEMENT AND RESEARCH METHODOLOGY

3.1 INTRODUCTION

The vital management activity to be carried out in software development is accurate estimation. The manifesto from the literature survey reveals that several effort models exist. This is research primarily focuses on the major root cause issues behind the estimation. To produce this intangible software the life cycle itself is very complex and different from other product models available in the markets. If forecasting was accurate then everything will be convenient and smooth. The popular empirical estimation models like SLIM, COCOMO, SEER - SEM influenced by Putnam’s framework, Price-S, SLIM and other models require accurate inputs for estimation of effort. In other words, the accuracy of the output of these models is depends upon the accuracy in the input parameters. It is very difficult to obtain such accurate information at the early development stage of a software project.

The managers face with uncertainty and unambiguous behavior at different time intervals in requirements of customers, technologies, familiarity level of the software developer, external environment, and government policies. Hence it is very hard to control. Analyzing the dominant effort manipulators eliminates all these major issues and its impact on estimation of effort. The subsequent sections talk about the contents of the software
Engineering dataset, the procedures to clean it by identifying the missing values, and a of imputation methods and how it is employed using Graphical and Genetic approach.

3.2 RESEARCH OBJECTIVES

The objectives of this research are defined in such a way to achieve the goal of the research and they are stated as follows:

- To analyze the effort multipliers and propose a scientific method to segregate the effort multipliers with reference to their impact in software development activities.
- To derive a transformation model of effort estimation scheme using graphical approach.
- To formulate an imputation model from the transformation model for effort estimation using Genetic approach.
- To validate the proposed hybrid imputation model for effort estimation using standard data set in terms of standard performance factors.
- The motivation of this research has attempted to show a trendy model for improving COCOMO through the use of Genetic Algorithms. The project development parameters are optimized to schedule and cost parameters.

Several problems were observed in this research to develop a hybrid imputation method in comparison with the current models, and it is recommended that these problems be investigated in potential research activities.
3.3 EXPERIMENTAL METHODOLOGY

The focus of this research is to estimate the effort needed in the development of a project along with interrelated factors which concerns effort and productivity. The software development effort estimation accuracy reported in many models are biased and this makes high inaccuracy introduction. This research does empirical assessment on COCOMO – II and improvement is made in this model.

A new hybrid method was proposed for improving the accuracy by combining graphical and genetic approach. Basically software projects to be developed are complex. This hybrid method addresses large interrelated effort factors which dominates the development of a process. The effort attributes are number of screen use in the software, volatile system requirements and the reusable software components. The major additional issues addressed are lack of measurement and the continuous change in software development process.

In the first stage isolation of effort multiplier is performed. Hence there is no significant relationship exists between effort multipliers using graphical method segregation of effort multipliers is done using their impact as two group’s mainly optimistic and pessimistic group. Multipliers in pessimistic group are directly proportional to the overall effort and those in optimistic group are inversely proportional. Optimistic group includes ACAP, PCAP, PCON, AEXP, PEXP, LTEX, TOOL, SITE and SCED. Pessimistic Group includes RELY, DATA, CPLX, RUSE, DOCU, TIME, STOR and PVOL.

The simplest method for treatment of missing value is to discard the observation with the missing value. However, with large dimension data sets (common in data mining) a significant portion of the observations may
have missing values. This discarding-data approach can lead to biased estimation and can cause larger standard errors due to reduced sample size. According to Gelman and Hill the discarding-data approach can be divided into three categories: complete analysis, available-case analysis and non-response weighting. Complete analysis refers to excluding any missing values of either input or output data. This method can cause bias to the analysis when the missing units differ systematically from the completely observed cases.

The focus of previous studies on missing data is to use imputation to better estimate the parameters of the statistical model instead of filling missing fields (Little and Rubin 2002). The literature on missing data in software Engineering industry applications is sparse. The straightforward arbitrary imputation method replaces the missing value with an arbitrarily selected value from another observation in the same variable. These imputation methods may be considered to be accurate if the proportion of missing values is small (Yarandi 2002). Using genetic approach we can perform operations in a hierarchical way. It does accurate estimates in an iterative and recursive fashion. It also permits computations on different type of variables are performed. It also supports reusability of subprograms and intermediate values are done efficiently. GA’s does optimization with high rating of reliability.

GA is most widely used for solving complex effort estimation techniques. GA is popularly known for their contribution towards robustness and they are probabilistic techniques that start from initial population of randomly generated potential solutions to such complex problems. This crisis is solved using repetitive application of Genetic operators such as Selection, crossover and mutation. The evolution process proceeds through stages of generations by allowing selected members of the current population; these are
chosen on the basis of fitness criteria, to combine through a crossover operator to produce offspring thus forming a new population. The evolution process is repeated until certain criteria are met.

GA-based scheduling algorithms aim to evolve near optimal effort after certain number of generations. The initial step is necessarily to employ GA to encode any possible solution to the optimization problem as a set of binary strings referred as chromosomes. Each chromosome represents one solution to the effort problem. This set of chromosomes is called as population. Deriving initial population is the second step in GA. Chromosomes is randomly selected from the list and included from which the actual evolution starts. This initial population is called as first generation.

Finally to evaluate the each chromosomes quality, chromosomes are assigned with a fitness value. The chromosome with a smaller fitness value represents a superior solution. The purpose of the GA search is to come across a chromosome that has the most favorable (smallest) fitness value. Based on the relative quality of each chromosome it is either eliminated or duplicated and this stage is called as selection process. This is further followed by crossover. Chromosomes are selected in pairs with some probability and chromosome components are replaced to make a valid chromosome this may exist in the current population or may not exist. Mutation is followed by crossover where each string is chosen with some probability where new population is created with the current population and evaluated. If the terminating criteria are not reached further it goes through the Genetic stages of selection mechanism, crossover mechanism, mutation mechanism and evaluation.
This is done using the Genetic procedure as follows (Goldberg 1989; Michalewicz 1996)

- Generate arbitrarily a population of solutions.
- Compute the fitness for each of the population elements.
- Create offspring’s by three genetic operators (Reproduction, crossover and mutation).
- Evaluate the new solution and calculate the fitness of each solution.
- If optimum solution is achieved, stop and return otherwise repeat the three genetic operators.

This research solves the problem of missing data and does enhanced estimation of the effort required for software using the hybrid experimental framework that is shown below in Figure 3.1. With this model the software practitioners understand the impact of missing and how it could be treated efficiently and accurately with the help of Genetic Algorithms.
Figure 3.1 Experimental Framework
Basically GA for any optimization problem is executed through the steps like encoding initial population, using fitness function perform evaluation, selection, crossover, mutation and termination condition. The researchers earlier performed various GA based effort estimation models, but this hybrid model is dominant it comprises of Graphical method for segregating the effort multipliers and Genetic Algorithms for imputing the blurred information and it assures that best solution is provided at the stipulated time.

The key part of this research study is the advancement and validation of artifacts for sensible Software Project effort management. Quality is very important for many organizations as one of the parameters of success. The significant outcomes and the research findings for the described hybrid framework can be used in improving the effort estimation accuracy of software projects. Nevertheless, these results and research findings are as very important for SPM.

3.4 SUMMARY

Software development effort can be either activity based or holistic based. The major driver in either of these two models is the size of the project. In these kind of problems if previous data can support the estimation process it could be accurate but mainly in holistic approach even in the absence of baseline data this model works well with imputational mechanisms. COCOMO is a holistic model and since extension of this model is taken in this approach this also lies in the same category.

Initial phase deals with isolation of effort multiplier. The significance in the relationship exists between effort multipliers using graphical method segregation of effort multipliers is done using their impact
as two group’s mainly optimistic and pessimistic group. Multipliers in pessimistic group are directly proportional to the overall effort and those in optimistic group are inversely proportional.

Using genetic approach the operations are done in a hierarchical way. It does accurate estimation in an iterative and recursive fashion. This also allows computations on different type of variables are performed. It also supports reusability of subprograms and intermediate values are done efficiently. GA’s does optimization with high rating of reliability. The next chapter of the thesis deals with model formulation using graphical method. The outcome of this Graphical Model is fed as input to the Genetic Algorithm which does optimization in an efficient manner.