CHAPTER 1
INTRODUCTION

1.1 PREAMBLE

This thesis discuss, in general terms, the application of soft computing algorithms (EMO) for designing a framework. Objectives of the thesis can be summarized as follows:

- Identification of Software Usability aspects
- Identification of software complexity metrics
- Establishing a trade-off between software complexity and its deliverability (usability aspects) by Evolutionary Multi-Objective Optimization (EMO).
- Development of a Framework by using Fuzzy Rule based System (FRBS) for quantification of software complexity and usability.

The work described herein not only concentrates on the software complexity part but it also contributes in the establishment of usability aspects of software. The present work proposes a framework to conquer the existing problems of establishing the trade-off relationship among software complexity and software deliverability. The motive of this thesis is to establish the trade-off using soft computing algorithms which is unique.

1.2 NEED FOR THE PRESENT WORK

Ever since the software development came into existence, there has been a rift between the software developers and users. Software developers in order to ease their development cycle, try to embed multiple features in a single module making the usability of the module tougher.

It is the usability feature which plays a major role in having the product more sellable, but at the same time it should cater to all high level needs of the consumer. Although, there are number of companies developing certain guidelines about software development process, the major focus is on user-centred application development, software is evaluated with various tools like cognitive tools and complexity Matrices to find out the degree of acceptability amongst the users.

The deliverability aspect of the software in most of the cases is contextual i.e. Analytical or Empirical Methods. The Analytical method depends upon potential interaction with the system and finding out the flaw in the system. Secondly, Empirical evaluation method which is based on the actual usage data.

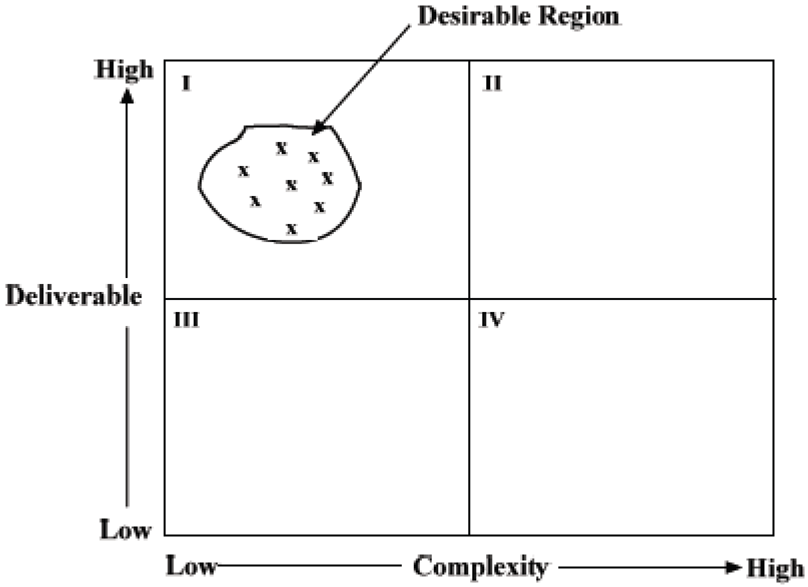

Figure 1.1: Deliverability v/s complexity

The main problem arises to maintain a balance between complexity and deliverability, as both the quantities are very much inter-related, it is very difficult to raise the deliverability without increasing the complexity of the system. The basic common criteria for deliverability are:

(i) Ease of use
(ii) Task Support
(iii) Navigation
(iv) Help
(v) Scalability with disturbing the ease of use.

There are various models and schemes developed for checking basic software quality improvement in terms of Flow of Data, Mean Time to Repair (MTTR) in addition to the Mean Time between Failure (MTBF), but there exists a huge gap in terms of usability of the software. Usability of the software refers to the ease of use in driving the desired result. The ISO/DIS defined the term usability as "Degree to which a software package can be utilized by a specific user to attain specific objectives with maximum efficiency, satisfaction as well as effectiveness in a precise usage circumstance". Relation between complexity and usability has an inverse relationship (Figure 1.2). As complexity raises the usability aspect of deliverable in terms of "Ease of Use" goes down.

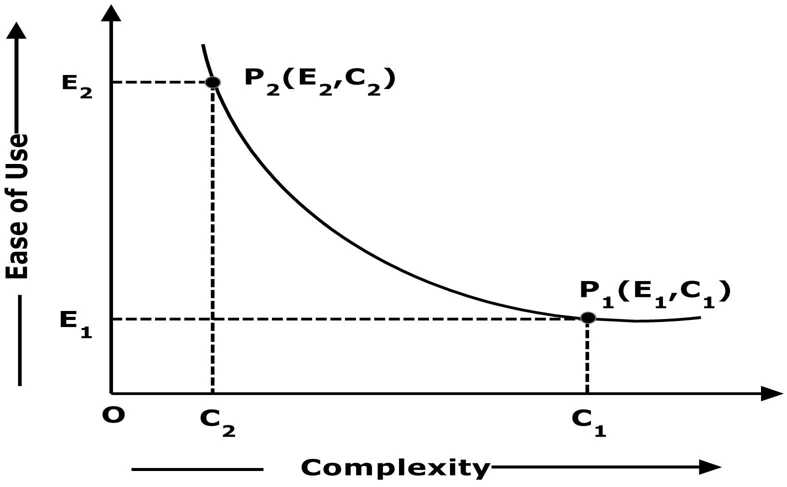

Figure 1.2: Plot between complexity and Ease of Use

As functionality of the software increases the deliverability value to the client also increases but at the same time, complexity also increases. There are chances that the user may not use some of the required functionality. In order to proceed with the problem, authors have used the concept of Expert System in calculating the elasticity between the two variables complexity and usability.

In general, complexity disturbs the ecological aspect of the messages in module. A complex information module can be represented in three dimensional formats as suggested in Albers [4, 6]: Knowledge Level, Detail Level and Cognitive Abilities. In order to increase all the three levels there will be a compromise with the Usability Aspect form HCI preview.

The deliverable aspects nowadays focuses on Human Centered application where customer's involvement plays a major role in design phase, but the customer always tells the requirement in the form of stories which looks much simpler during requirement gathering stage but when implemented on real scale. The complexity of the software increases considerably.

1.3 OUTCOMES OF HIGH SOFTWARE COMPLEXITY

High complexity in the software leads to various negative results. Following are the outcomes of high complexity in the software:

- Lack of Adoption
- More end-user training
- More software Technical support
- Less likability of Software
- Low user’s performance
- Lack of customer satisfaction

It is broadly known that enterprise resource planning (ERP) software systems put up with very complex user interfaces. The software complexity of these user interfaces negatively affects the usability aspect of these software systems.

Present study has revealed that a need exists to advance the overall usability of the ERP software systems. The Specific approach as well as the criterion for evaluating the usability aspect of ERP software products have not been developed or broadly published. This work proposes a set of heuristics that can be used to measure the usability of ERP systems and similar kinds of software systems.

This work gives the description about the complexity analysis; a quantitative approach to the software usability engineering which has been effectively used in a number of real-world software projects. The complexity analysis depends on finding and quantifying impediments that get in the way of easily learning and using software. The impediments – such as confusing user interfaces, long sequences of manual steps and cryptic error messages – are quantified by the measures named as “complexity metrics”.

The complexity metrics gives the easily-understood comparisons of usability between the steps in a task, overall tasks, releases, and the products. They are developed through thorough, exhaustive rating scales related with the following six aspects of software usability: navigational guidance, context shifts, error feedback input parameters, new concepts and system feedback. Even though the complexity analysis is a lighter-weight usability evaluation method as compared to the usability testing, the empirical results show that the complexity metrics are powerfully related to the usability testing time-on task measures.

The association between the complexity of software and its usability is contradictory in nature and hence can be represented by the following equation:

Maximize (deliverability, usability)

Or

Minimize (un-deliverability, complexity)

Equation 1

1.4 SCOPE OF THE WORK

To facilitate the trade-off among the software complexity and deliverability, the concept of Evolutionary Fuzzy Rule Generation using Messy Genetic Algorithm has been used. The multi-objective evolutionary algorithm is well known technique in finding out optimum solution in case of multiple goals. The problem which is single objective optimization in nature could have a single optimal solution while multi-objective generates multiple solutions produces the vectors representing the value of trade-off.

The work presents the issue of establishing the trade-off between the software complexity and its deliverability aspect. Based on the management of trade-off [8]-[10], the Popularity index of the software is determined.

1.5 OBJECTIVES

There are two objectives of the present work which are as follows:

1. Establishing a trade-off between software complexity and its deliverability (usability aspects) by Evolutionary Multi-Objective Optimization.
2. Development of a Framework by using Fuzzy Rule based System (FRBS) for the quantification of software complexity and its usability.

1.6 RESEARCH METHODOLOGY

The steps followed are mentioned below:

1. The first step covers the establishment of software complexity metrics to evaluate the complexity of any software or its particular application.

2. The second step involves the identification of software usability aspects to evaluate the deliverability of that particular software.

3. The third step involves the establishment of trade-off between the software complexity and deliverability using evolutionary multi-objective optimization (EMO).

4. The fourth step involves the development of framework using Fuzzy Rule Based System (FRBS) for the quantification of software complexity and its usability.

5. The last step deals with bringing the thesis to a Conclusion by justifying the work which is done till date.

1.7 ORGANISATION OF THE THESIS

Chapter 2 gives a brief introduction to evolutionary multi-objective optimization. It also highlights the various characteristics and components soft computing algorithms. Lastly, it discusses about software complexity metrics, its evaluation criteria and six aspects of software usability.

Chapter 3 presents literature review of the soft computing algorithms. It highlights the work by various researchers and companies in the form of their white papers to discuss the basic ideology of software deliverability criteria's and usability index over the last few years.

Chapter 4 involves the completion of the first objective that is the establishment of trade-off between software complexity and deliverability with the help of the concept of EMO.

Chapter 5 involves the development of framework using Fuzzy Rule Based System (FRBS) for the quantification of software complexity and usability.

Chapter 6 presents the verification and validation of the developed framework using the concept of Hypothesis Testing and the mathematical model in support of the framework using rough set theory.

Chapter 7 draws the previous chapters to a conclusion and indicates how the deliverability of the software significantly drops with the increase in the complexity along with the development of framework for potential future developments.