Chapter - 3

Cognitive Complexity
Metric Suite Framework for Aspect Oriented Programming
CHAPTER - 3

COGNITIVE COMPLEXITY METRIC SUITE FRAMEWORK FOR ASPECT ORIENTED PROGRAMMING

3.1 Introduction

This chapter presents theoretical work that builds a suite of Cognitive Complexity Metrics (CCM) for Aspect-Oriented design. The previous chapter has insisted upon the need for cognitive complexity metrics. Though the literature review has provided the various existing AOP metrics and their pros and cons, still there is a lack of Cognitive Complexity Metrics Suite (CCMS) which considers the various AOP features. It also insists upon the need for proposing a CCMS and an automated tool for computing the metrics in CCMS. This chapter provides the overall CCMS framework and various validation methods needed to validate the proposed metric suite.

3.2 CCMS Framework

The proposed CCMS framework consists of four phases namely Metric phase, Validation phase, Tool phase and Comparative Study phase. Figure 3.1 shows the block diagram of the proposed CCMS framework.

3.2.1 Metric Phase

The proposed metrics suite, CCMS consists of four metrics, namely, Cognitive Weighted Method per Class (CWMC), Cognitive Weighted Coupling on Advice Execution (CWCAE), Cognitive Weighted Pointcut per Aspect (CWPA), and Cognitive Weighted Coupling on Attribute Reference (CWCoAR). They are used to measure the cognitive complexity arising due to Advice, Joinpoint, Pointcut, and Attribute Reference. These AOP features play a vital role in determining the quality of the design in AOP paradigm.
Figure 3.1 CCMS Framework
In the Metric phase, Cognitive Weighted Methods per Class (CWMC), which considers the cognitive complexity of the different types of advice, such as before, after, and around. Cognitive Weighted Coupling on Advice Execution (CWCAE), which considers the cognitive complexity of the different types of joint points. Cognitive Weighted Pointcut per Aspect (CWPA), which adding Cognitive Weight of the Pointcut Designator (CWPD) and cognitive weight of the joinpoint signature used in an aspect. Coupling on Attribute Reference (CoAR) is calculated by adding the weight of the reference variable. But, there is no cognitive complexity metric for CoAR and does not consider attribute allusion either statically, by inheritance, or dynamically. The metric can measure the attribute factor complexity based on cognitive perspective and consider the different types of attribute reference.

3.2.2 Validation Phase

The practical success of any proposed metric depends on the establishment of its validation, understandability by its users and the tight link between the metric and the attribute that is intended to measure. The establishment of understandability and the tight link between the links highly depend on the validation. Therefore, the proposed metric must be evaluated theoretical and experimental for its validation.

Theoretical Validation

Fenton et al. [Fen, 97] defined some properties used for the data collection process and are described as follows:

- Accuracy: the higher the difference between the actual data and measured data, the lower is the accuracy and vice-versa.
Replicability: means that the analysis can be done at different times by different people by using the same setting.

Correctness: According to the metrics definition data is collected.

Precision: Data is expressed by some decimal places. Less decimal places show a lower accuracy.

Consistency: It counts the differences in the metric values when collected using different tools by different people.

**Experimental Validation**

Experimental validation is the process of demonstrating the applicability of Metric(s) or Measure(s) using small application programs. Normally, most of the metrics are demonstrated through the implementation of the small application program(s). However, when a researcher is applying only this method for their validation, it may not be sufficient even for a complete demonstration of the metric(s).

The other common practice to validate software metrics/measures is through case studies. These case studies are sometimes small projects reported in the literature or on the web [Bas, 96] [Gib, 89] or some large programs. Naturally, this is the way to implement the metric on complex software products.

**3.2.3 Tool Phase**

There are many metric tools available to automatically compute the traditional AOP metrics. But, there is no tool to compute various cognitive complexity metrics of AOP design. Hence, the Cognitive Complexity Metrics Analysis Tool (CCMAT) has been developed to collect and analyze various cognitive complexity metrics.
The CCMAT is used to carry out a comparison between AOP metrics and AOP Cognitive Complexity Metrics. AOP metrics are used since it has been widely accepted by the researchers as standard metrics for Aspect-Oriented system.

It is also used to collect CWMC, CWCAE, CWPA, and CWCoAR metrics and various existing metrics for AOP programs for comparative study. The CCMAT is used to analyse the quality of the AspectJ projects.

### 3.2.4 Empirical Validation Phase

Empirical studies [Bas, 96] are used to investigate software development and practices for understanding, to evaluate, and to develop in proper contexts. It allows the analyst to test out the theories with the support of empirical observations. It includes formal experiments, case studies and surveys observed in industry, laboratory or classroom [Kit, 95]. All these different types of empirical validation techniques can also be applied to the validation of software complexity measures.

In software engineering, several researchers have performed empirical validation with students in a classroom/laboratory environment [Wan, 93] [Wan, 03] [Bas, 96] [Zaz, 08]. The examples of experiments in the classroom include controlled experiments (with graduate students).

The empirical validation is to perform an analysis of the results and the comparative study, which have been done in previous stages. The results of the analysis and comparative study help the developer to convince industry to apply the new metric program for advanced empirical validation.
3.3 Conclusion

The academicians/developers of the new metrics try to prove their claim by evaluating their proposal through different means. These different evaluating standards are theoretical validation (for example, evaluation through measurement theory), experimentation in the class room, case studies, different examples from the web, etc. However, they are essential but not complete. It is found that most of the newly proposed metrics are not empirically validated and hence the chances for the success of these measures are not promising. In general, the empirical methods suggest proposing a model, developing statistical/qualitative methods, applying to case studies, measure and analyzing, validating the model and repeating the procedure [Bas, 96]. All these forms of empirical validations are recommended for any empirical study in software engineering. Henceforth, the proposed CCMS framework is used to measure the complexity of AOP software concerning the cognitive complexity. In the following chapter, CWMC metric is to be defined and validated.