Chapter - 7

Cognitive Weighted Coupling on Attribute Reference
CHAPTER - 7

COGNITIVE WEIGHTED COUPLING ON ATTRIBUTE REFERENCE

7.1 Introduction

Aspect Oriented Programming (AOP) is a programming paradigm that aims to increase modularity by allowing the separation of cross-cutting concerns [Rei, 05]. AOP includes programming methods and tools that support the modularization of concerns at the level of the source code, while “Aspect-Oriented Software Development” refers to a whole engineering discipline. All AOP implementations have some crosscutting expressions that encapsulate each concern in one place.

A software metric is a standard of measure of a degree to which a software system or process possesses some property [Chi, 94]. The goal is obtaining objective, reproducible and quantifiable measurements, which may have numerous valuable applications in the schedule and budget planning, cost estimation, quality assurance testing, software debugging, software performance optimization, and optimal personnel task assignments.

Several software industries have moved to AOP paradigm in order to increase their capability through reusability function offered by AOP. The use of AOP has increased the complexity [Ivi, 13]. So, there is a need for introducing new complexity measures. The complexity reflects the cognitive load in programming and hence cognitive complexity plays a vital role in measuring the complexity. A new metric namely Cognitive Weighted Coupling on Attribute Reference (CWCoAR) is proposed for an AOP system which is an extension of the CoAR proposed by Kumar and
Grover [Kum, 09]. CWCoAR includes the cognitive complexity due to Attribute Reference and so, it is an enhanced indicator of the complexity of AOP systems.

7.2 Motivation

Ananthi Sheshasaayee and Roby Jose [Ana, 15] studied about Aspect Oriented Coupling and Cohesion Measures for Aspect-Oriented systems. This study is planned to frame an idea about the coupling, cohesion measures and framework all along with tool support for the coupling measures.

Mandeep Kaur and Rupinder Kaur [Man, 15] analysed Improving the Design of Cohesion and Coupling Metrics for Aspect-Oriented Software Development. This study focuses on developing metrics for better calculation of coupling and cohesion values. OOPs also have some limitations like in the system decomposition, there are some functionalities cannot be assigned to the single module. There is one paradigm that enhances software design and promotes reusability called Aspect Oriented Paradigm (AOP).

Classes are the building blocks of AOP. Class is an encapsulation of attributes and methods. The attributes are used for storing and manipulating the data in the program. Attributes are one of the major factors which will affect the complexity of the class and it is clear that the use of different data type of attributes increase the complexity of the programs. There is no specific measure exists to calculate the complexity due to attributes. Hence, a new metric CWCoAR has been proposed for AOP system with attribute reference.
7.3 Formulation of the CWCoAR Metric

7.3.1 Existing Metric

Kumar and Grover [Kum, 09] extended the work of Ceccato and Tonnella [Cec, 04], Zhao (2004), and others by introducing a new coupling metrics framework for AOP. Their research specified 17 types of connections that exist in an AOP system between attributes, operations, or components. From these 17 connection types, the authors developed six new metrics, namely, Coupling on Attribute Type (CoAT), Coupling on Parameter Type (CoPT), Coupling on Attribute Reference (CoAR), Coupling on Object Invocation (CoOI), Coupling on Inheritance (CoI), and Coupling on High-level Association (CoHA). The authors acknowledged that these metrics were simply a framework, and lacked empirical evaluation. No other studies appear to either verify or utilize these metrics.

Coupling on Attribute Reference (CoAR) [Kul, 06] defines the number of operations from one component that references an attribute of another component. But, there is no cognitive complexity metric for CoAR and does not consider attribute allusion by statically, by inheritance, or dynamically. This Attribute complexity metric was based on cognitive perspective and considering the different types of attribute allusion.

7.3.2 Proposed Metric

The Cognitive Weighted Coupling on Attribute Reference (CWCoAR) complexity metric augments the cognitive complexity based on the different types of attribute allusion to the components. The attribute allusion can range from statically, by legacy, or with dynamism. The scope of this article is AspectJ only, and in this, the range of attribute allusion is implemented statically, by inheritance, or dynamically. Based on these three types of attribute allusion, the new CWCoAR can be defined as,
Cognitive Weighted Coupling on Attribute Reference

\[
\text{CWCoAR} = ((\text{CWS}_\text{SRV} \times \text{SRV}) + (\text{CW}_\text{IRV} \times \text{IRV}) + (\text{CW}_\text{DRV} \times \text{DRV})) \quad \ldots (7.1)
\]

\text{CWS}_\text{SRV} = \text{Cognitive Weight of Static Reference Variable}

\text{SRV} = \text{Number of Static Reference Variable in Aspect}

\text{CW}_\text{IRV} = \text{Cognitive Weight of Inheritance Reference Variable}

\text{IRV} = \text{Number of Inheritance Reference Variable in Aspect}

\text{CW}_\text{DRV} = \text{Cognitive Weight of Dynamic Reference Variable}

\text{DRV} = \text{Number of Dynamic Reference Variable in Aspect}

7.4 Calibration of CW for CoAR

Cognitive weights for different reference variable are calibrated in this section. In order to find the Cognitive Weight for Static Reference Variable (\text{CWS}_\text{SRV}), Cognitive Weight for Inheritance Reference Variable (\text{CW}_\text{IRV}), and Cognitive Weight for Dynamic Reference Variable (\text{CW}_\text{DRV}), a Comprehension test was conducted for three different groups of students to find out the time taken to understand the complexity of different types of reference variables. These groups of students had sufficient exposure to Java and Aspect. In the region of 60 students, who have scored 65% and above marks in Semester examination, were selected in group. One undergraduate group and two postgraduate groups are called for the comprehension test and supplied 9 different programs namely, P1 to P9, three for each type of reference variable.

The time taken by each student to understand the program and the time was recorded after the completion of each program. This process is repeated for each group of students. To be accurate, these program comprehension tests were conducted, and the comprehension timings were registered. The average time taken to
comprehend each program from P1 to P9 by each group was calculated, to get 27 different Comprehension Mean Time (CMT). Since 3 different groups of students have done the comprehension test for the same program, their values are averaged to obtain the 9 different values. These values are tabulated in Table 7.1, under the column CMT. The Figure 7.1 shows the calibration of cognitive weights.

<table>
<thead>
<tr>
<th>Program #</th>
<th>SRV (Min)</th>
<th>IRV (Min)</th>
<th>DRV (Min)</th>
</tr>
</thead>
<tbody>
<tr>
<td>P1</td>
<td>22</td>
<td>14</td>
<td>15</td>
</tr>
<tr>
<td>P2</td>
<td>20</td>
<td>16</td>
<td>15</td>
</tr>
<tr>
<td>P3</td>
<td>25</td>
<td>18</td>
<td>19</td>
</tr>
<tr>
<td>P4</td>
<td>23</td>
<td>15</td>
<td>16</td>
</tr>
<tr>
<td>P5</td>
<td>27</td>
<td>19</td>
<td>17</td>
</tr>
<tr>
<td>P6</td>
<td>21</td>
<td>17</td>
<td>18</td>
</tr>
<tr>
<td>P7</td>
<td>22</td>
<td>20</td>
<td>19</td>
</tr>
<tr>
<td>P8</td>
<td>26</td>
<td>21</td>
<td>20</td>
</tr>
<tr>
<td>P9</td>
<td>23</td>
<td>17</td>
<td>19</td>
</tr>
<tr>
<td><strong>Average Comprehension Time</strong></td>
<td><strong>209</strong></td>
<td><strong>157</strong></td>
<td><strong>158</strong></td>
</tr>
<tr>
<td><strong>Cognitive Weight ( Rounded)</strong></td>
<td><strong>2</strong></td>
<td><strong>1.6</strong></td>
<td><strong>1.6</strong></td>
</tr>
</tbody>
</table>

Figure 7.1 Calibration of Cognitive Weights
The following section explains the calculation of CWCoAR with a case study.

### 7.5 Illustration

The proposed CWCoAR metric given by Eq 7.1 is evaluated using both Java and AspectJ programs, given in Appendix C. From the program, the existing and proposed metric value is calculated.

\[
\text{CoAR} = \text{Static Reference Variable} + \text{Inheritance Reference Variable} + \text{Dynamic Reference Variable}
\]

\[
\text{CoAR} = 1 + 1 + 1 = 3.
\]

Similarly, applying to the newly proposed complexity metric CWCoAR, the complexity value can be calculated. This complexity metric includes both the structural complexity as well as the cognitive complexity of the program. Hence, in the calculation of CWCoAR, each Reference Variable is multiplied by the corresponding cognitive weight.

\[
\begin{align*}
\text{CWCoAR} &= (\text{CW}_{\text{SRV}} \times \text{SRV}) + (\text{CW}_{\text{IRV}} \times \text{IRV}) + (\text{CW}_{\text{DRV}} \times \text{DRV}) \\
&= (2 \times 1) + (1.6 \times 1) + (1.6 \times 1) \\
&= 2 + 1.6 + 1.6 \\
\text{CWCoAR} &= 5.2
\end{align*}
\]

### 7.6 Comparative Study

In this section, the comparative study is done to validate the CWCoAR complexity metric, as it is done in other cases of newly proposed complexity metric. To do the comparative study, a comprehension test was conducted to a group of students who are doing their master’s degree. There were forty students in the group who participated in the test. The students were given five different programs, P1 to P5, in Java for the comprehension test. The time taken to complete the test in minutes
is captured in online, to maintain the accuracy. The average time taken to comprehend each program by all students is calculated and placed in Table 7.2 under the column head Comprehension Mean Time (CMT). The CoAR and CWCoAR values are calculated manually for each of the five programs as demonstrated in the case study section. The values are also tabulated in Table 7.2 under the column CoAR and CWCoAR.

Table 7.2 Complexity Metric Values and CMT Values

<table>
<thead>
<tr>
<th>Program #</th>
<th>CoAR</th>
<th>CWCoAR</th>
<th>CMT (Min)</th>
</tr>
</thead>
<tbody>
<tr>
<td>P1</td>
<td>3</td>
<td>5.2</td>
<td>20</td>
</tr>
<tr>
<td>P2</td>
<td>4</td>
<td>6.2</td>
<td>23</td>
</tr>
<tr>
<td>P3</td>
<td>2</td>
<td>3.2</td>
<td>18</td>
</tr>
<tr>
<td>P4</td>
<td>6</td>
<td>6.6</td>
<td>26</td>
</tr>
<tr>
<td>P5</td>
<td>5</td>
<td>8</td>
<td>27</td>
</tr>
</tbody>
</table>

Based on the values in Table 7.2, Pearson Correlation test was conducted between the CoAR and CMT. The correlation value \( r (\text{CoAR, CMT}) \) is 0.948504. Again the Pearson Correlation with CWCoAR and CMT was calculated, and the
value \( r \) (CWCoAR, CMT) is 0.950962. Both the correlations were found to be positive, implying that both CoAR and CWCoAR correlates well with CMT values captured in the empirical test conducted. This shows that the CMT values are truthful and meaningful. The bigger correlation value for CWCoAR than the CoAR implies that CWCoAR is a better indicator of the complexity of the classes with various scopes of methods. This fact is further clarified clearly in the correlation chart given in Figure 7.2.

Thus the proposed CWCoAR complexity metric, as it includes the cognitive complexity, is proved to be more robust and more realistic complexity metric than CoAR complexity metric which considers only the architectural complexity.

7.7 Theoretical Validation

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

a. **Accuracy** - The higher the difference between the actual data and measured data and the lower is the accuracy and vice-versa. The difference between CoAR and CWCoAR is lower, so the accuracy is higher.

b. **Replicability** - Means that the analysis can be done at different times by different people using the same setting. Data are taken from rural and urban PG students at a different time.

c. **Correctness** - According to the metrics definition data was collected. The value of CWCoAR is collected and calculated through the WPA metric.

d. **Precision** - Data is expressed by a number of decimal places. Less number of decimal places shows a lower accuracy. The decimal place of the data is high. So it shows a higher accuracy.
e. **Consistency** - It counts the differences in the metric values when collected using different tools by different people. Accordingly, we found the difference between existing metric - CoAR and proposed metric – CWCoAR by giving different programs by different students.

### 7.8 Conclusion

A CWCoAR metric for measuring the aspect level complexity has been formulated. The Attribute Reference complexity of the aspect includes the static attribute reference and dynamic attribute reference. CWCoAR has proved that complexity of the class is based on the cognitive weights of the various types of Attribute Reference. The metric is evaluated through statistical analysis, case study and a comparative study, and proved to be a better indicator of the aspect level coupling complexity. The next chapter deals with a new Integrated Development Environment (IDE) to integrate and to deploy the proposed framework.