International Journal of Scientific Research in Computer Science, Engineering and Information Technology
 | Volume 2 | Issue 5 | ISSN : 2456-3307

# Extended Metric of Cognitive Weighted Coupling on Method Call

**K. R. Martin[*1], Dr. E. Kirubakaran[2], Dr. E. George Dharma Prakash Raj[3]**
[*1]Department of Computer Science, Tiruchirappalli, India
[2]AGM (SSTPS), BHEL, Tiruchirapalli, TamilNadu, India
[3]Asst Professor in School of Computer Science, Bharathidasan University, Tiruchirappalli, TamilNadu, India

**ABSTRACT**

Aspect Oriented Software Development is a innovative and emergent paradigm, which supports the separation of concerns that are speckled over the system. Software metrics are used in measuring desirable software and software development characteristics of Aspect Oriented Software Development (AOSD). The maintenance cost can be precise if software metrics is applied during the development phase (Chidamber, 1994). As Aspect Oriented Software Development is an evolving paradigm a study on maintainability and its accompanying metrics need to be exactly accomplished. This paper presents a new cognitive complexity metric namely cognitive weighted coupling between objects for measuring coupling in Aspect Oriented Software.

**Keywords :** Software Metrics, Aspect Oriented Software Development (AOSD), Control Coupling, Global Data Coupling, Internal Data Coupling, Data Coupling, Lexical Content Coupling, Cognitive Weighed Coupling Between Objects (CWCBO).

## I. INTRODUCTION

Software engineering is a challenging and composite task. Software metrics are one way to determine quality within a system, indicating to problem extents that can be criticised prior to software release. Metrics challenge to measure a certain trait of a software system. These traits can range from traditional measurements such as the number of lines of code to the relationships shaped between components in a system. There are frequent approaches to appraisal complexity of software but none of them have been accepted as a true measure of complexity of a class (Kiczales, 1997).

Aspect Oriented Programming (AOP) extends the traditional object-oriented programming (OOP) model to progress code salvage across different object hierarchies. AOP can be used with object oriented programming. AspectJ is an implementation of aspect-oriented programming for Java. AspectJ adds to Java just one new concept, a join point and that adds name for an existing Java concept. Maintainability is observed a software quality that plays a vibrant role in software eminence level. If the software's eminence level is higher, the less effort/cost the software maintenance cycle entails.

Aspect J has no CWCBO metric to extent the different type of coupling proposed by several researchers. So, there is a necessity for cognitive weighted coupling metric for the class and aspect level coupling measurement. Hence our core objective is to express a cognitive weighted coupling metric to measure the coupling at the various levels.

## II. LITERATURE REVIEW

Numerous metrics have been suggested for OO systems by researchers. A metric suite proposed by Chidamber and Kemerer (C&K) is one of the best known suites of OO metrics. The six metrics proposed by CK are Weighted Method per Class (WMC), Depth of Inheritance Tree (DIT), Response For a Class (RFC), Number Of Children (NOC), Lack of Cohesion of Methods (LCOM) and Coupling Between Objects (CBO) (Chidamber, 1994).

Ceccato and Tonella (2004) familiarized many aspect oriented metrics which encompassed aspect oriented coupling metrics as well. The metrics that the study used was the extension of the metrics suite from objects oriented metrics. The work also collected the value for the metrics from software using the developed tool.

A.Aloysius and G.Arockia Sahaya Sheela (2015) studied about aspect oriented metrics. This paper expresses the development and implementation of various metrics for AOP design paradigm and outlines the future directions.

Kulesza et al (2006) grants a quantitative study that consider the positive and negative effects of AOP on maintenance deeds of a web information system. The study also reflected the positive and negative effects of AOP on coupling measures when compared to the object oriented version of the same system.

Ananthi Sheshasaayee and Roby Jose (2015) considered about Aspect Oriented Coupling and Cohesion Measures for aspect oriented systems. This revision is intended to frame an indication about the coupling, cohesion measures and framework all along with tool provision for the coupling measures.

Mandeep Kaur and Rupinder Kaur (2015) investigated Improving the Design of Cohesion and Coupling Metrics for Aspect Oriented Software Development. This learning emphases on developing metrics for better calculation of coupling and cohesion ideals.

A. Aloysius and L. Arockiam (2012) proposed cognitive complexity metric namely cognitive weighted coupling between objects for measuring coupling in object- oriented systems. In this metric, five types of coupling that may exist between classes: control coupling, global data coupling, internal data coupling, data coupling and lexical content coupling are reflect in calculating CWCBO.

## III. COGNITIVE WEIGHTED COUPLING BETWEEN OBJECTS (CWCBO) IN OOP

CK (1996) define Coupling Between Objects (CBO) for a class to be the count of the number of other classes to which it is directly coupled. This number represents an object's fan-out to external objects. The metric's basis is in the fact that if an object is coupled to another it uses another's methods or instance variables.

Coupling Between Objects (CBO) for a class is a count of the number of other classes to which it is coupled. This definition is flexible in three ways.

- ✓ Which direction a class is coupled to another
- ✓ How a class is actually coupled to another
- ✓ The value to give a coupling relationship to distinguish its strength from another coupling.

Edward Berard (2001) has proposed several types of coupling which are defined as follows:

Table I Various Types of Coupling

| **Coupling Types** | **Definition** |
|---|---|
| Control Coupling | Passing control flags between modules so that one module controls the sequencing of the processing steps in another module. |
| Global Data Coupling | Two or more modules share the same global data structures. |
| Internal Data Coupling | One module directly modifies local data of another module. |
| Data Coupling | Output from one module is the input to another Using parameter lists to pass items between routines |
| Lexical Content Coupling | Some or all of the contents of one module are included in the contents of another. |

In Object Oriented Programming (OOP), CWCBO metric for classes includes several types of coupling. The several types coupling are Control Coupling (CC), Global Data Coupling (GDC), Internal Data Coupling (IDC), Data Coupling (DC) and Lexical Content Coupling (LCC) (Joseph, 2001). This metric is proposed by Aloysius and Arockiam (2012).

## IV. METRIC ANALYSIS

### A. Existing Metric

Numerous metrics have been proposed for AOP systems by researchers. One of the metric proposed by Ceccato et.al (2004) is CBO. It's an equivalent of the CBO metric from CK Metrics suite (1994). Coupling

Between Objects (CBO) for a class or aspect is a count of the number of other classes to which it is coupled. CBO was fragmented into two (CMC and CFA) to distinguish coupling on operations from coupling on attributes.

### B. Novel Metric for AOP

Coupling Between Objects (CBO) for a class is a count of the number of other classes to which it is coupled. But it does not consider several types of coupling for class. So the novel metric for AOP, CWCBO metric is consider several types of coupling for class and calculated by adding the coupling complexity of classes ($CWCBO_C$) and aspects ($CWCBO_A$). CWCBO can be calculated using the following equation.

$$\mathbf{CWCBO = CWCBO_C + CWCBO_A}$$

Where,
CWCBO(S) is the CWCBO for a version of AO software.
$CWCBO_C$ is Coupling Between Object for Classes.
$CWCBO_A$ is Coupling Between Object for Aspects.

## V. EMPIRICAL METRIC DATA COLLECTION & EVALUATION CRITERIA

This segment deliberates the CWCBO metric, empirical data, collection statistics, analysis and its implication.

### A. Calibration

In this section, an experiment is conducted to allocate cognitive weight to the various type of couplings discussed in section III. A comprehension test has been conducted for a group of students to find out the time taken to understand complexity of object oriented program with respect to different types of coupling. The collection of students designated had plenty exposure in analysing the Aspect-oriented programs, as they had endured courses in AspectJ language. 30 students taken from Rural, 30 students taken from Urban were nominated to participate in the comprehension test.

The time taken by students to comprehend the programs was recorded after the completion of each program. The time taken for comprehension of all these programs was noted and the mean time to comprehend was calculated. Five different programs have been administered in each case, totally twenty five different mean timings were recorded. Average time was calculated for each program from the individual time taken by students which shows in Table II.

Table II. Categorized Average Comprehension Time

| **Programs** | **Average Comprehension Time (In Minutes)** | | | | |
|---|---|---|---|---|---|
| | **LCC** | **DC** | **1DC** | **GDC** | **CC** |
| P1 | 33.4 | 29 | 23.7 | 17 | 9,78 |
| P2 | 34 | 28 | 24 | 17 | 10 |
| P3 | 33.5 | 28 | 23.4 | 16.1 | 11 |
| P4 | 33.3 | 27 | 24 | 17 | 9.9 |
| P5 | 33.48 | 27.1 | 23 | 16 | 10 |
| Average | 33.453 | 27.83 | 23.41 | 16.53 | 10.16 |

The average comprehension time, for programs are enumerated in table 6.2. These programs are based on Aspect Oriented Programming. The mean time is also calculated for each type of the programs and is tabulated.

### B. Statistical Analyses

For each coupling, mean was selected as a measure of central tendency. Table III illustrate statistical computation of different types of coupling.

Table III. Mean values for different types of coupling

| **Prog rams** | **Mean Comprehension Time(In Hours)** | | | | |
|---|---|---|---|---|---|
| | **LCC** | **DC** | **IDC** | **GDC** | **CC** |
| P1 | 0.56 | 0.5 | 0.4 | 0.28 | 0.16 |
| P2 | 0.6 | 0.47 | 0.4 | 0.3 | 0.2 |
| P3 | 0.56 | 0.47 | 0.39 | 0.27 | 0.2 |
| P4 | 0.56 | 0.5 | 0.4 | 0.3 | 0.2 |
| P5 | 0.558 | 0.45 | 0.4 | 0.3 | 0.2 |

| | | | | | |
|---|---|---|---|---|---|
| Mean | 0.5575 | 0.464 | 0.39 | 0.276 | 0.169 |
| ST DEV | 0.0674 | 0.014 | 0.005 | 0.006 | 0.005 |

A standard derivation close to 0 indicates that the data points tend to be very close to the mean of the set.

**CWCBO**

Coupling Between Objects (CBO) for a class is a count of the number of other classes to which it is coupled. But it does not consider several types of coupling for class & aspect. So the new metric for AOP, CWCBO metric is consider various types of coupling for class and aspect and calculated by adding the coupling complexity of classes ($CWCBO_C$) and aspects ($CWCBO_A$). CWCBO can be calculated using the following equation.

**CWCBO = $\mathbf{CWCBO_C}$ + $\mathbf{CWCBO_A}$** --------------> (1)

**$\mathbf{CWCBO_C}$ :**
$CWCBO_C$ = (CC*WFCC) + (GDC*WFGDC) + (IDC*WFIDC) + (DC*WFDC) + (LCC*WFLCC)

Where
CC is the total number of modules that contains Control Coupling.
GDC is the count of Global Data Coupling.
IDC is the count of Internal Data Coupling.
DC is the count of Data Coupling.
LCC is count of Lexical Content Coupling.
WFCC is the Weighting Factor of CC.
WFGDC is the Weighting Factor of DC.
WFIDC is the Weighting Factor of IDC.
WFDC is the Weighting Factor of DC.
WFLCC is the Weighting Factor of LCC.

**$\mathbf{CWCBO_A}$ :**
$CWCBO_A$ = CWCMC + CWCFA

CMC counts Number of modules or interfaces declaring methods that are possibly called by a given module. This metric does not considered the various return types. The proposed metric called CWCMC (Reilly, 2005), counts Number of modules or interfaces declaring different return type methods that are possibly called by a given module is multiplied by number of parameter.

CFA counts Number of modules or interfaces proclaiming fields that are accessed by a given module. This metric does not considered the various data types. The proposed metric called CWCFA (Martin, 2017), which considers the cognitive complexity of the different data types of character, integer, float, long and double.

The Weighting Factor of each type of coupling is calibrated in Table IV using the method discussed in the Empirical Metric Data Collection. The weight value is calculated based on the mean value of different types of coupling. To normalize the mean value to get appropriate weight value. The following table explained the rounded values of each type of coupling that is called weighting factor of each type of coupling.

Table IV. Weight Value of Each Type of Coupling

| Coupling | Weight Value |
|---|---|
| LCC | 0.6 |
| DC | 0.5 |
| IDC | 0.4 |
| GDC | 0.3 |
| CC | 0.2 |

Table V. Coupling Complexity metric value for the sample program

| Program# | Existing Metric Value (CBO) | Proposed Metric Value (CWCBO) |
|---|---|---|
| 1 | 5 | 2.7 |

## VI. DATA COLLECTION PROPERTIES

Fenton et al. (1997) defined some properties which were used for the data collection process and are described as follows:

- ***Accuracy*** The higher the difference between the actual data and measured data and the lower is the accuracy and vice-versa. The difference between CWCBO and CBO is lower so the accuracy is higher.
- ***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 different time.

- ***Correctness*** According to the metrics definition data was collected. The value of CWCBO is collected and calculated through the CBO metric.
- ***Precision*** Data is expressed by number of decimal places. Less decimal place shows a lower accuracy. The decimal place of the data is high (i.e. 0.466).so it shows a higher accuracy.
- ***Consistency*** It counts the differences with the metric values when collected using different tools by different people. Accordingly we found the difference between existing metric - CBO and proposed metric – CWCBO by giving different programs by different students.

The above properties are very mutual to be in every situation such as network and cloud. Cloud is now a essential prototype for outsourcing diverse computer needs of organisations. So, there is a necessity to propose metrics based on cloud with substantial these properties.

## VII. COMPARATIVE STUDY

A comparative study has been made with most widely accepted the metric proposed by Ceccato et.al (2004) is CBO. Coupling Between Objects (CBO) for a class or aspect is a count of the number of other classes to which it is coupled. CBO was split into two (CMC and CFA) to distinguish coupling on operations from coupling on attributes. The current CWCBO metric is one step ahead of existing CFA metric, because it includes the complexity that arises due to the various type of coupling. Another advantage of CWCBO metric is that, it takes cognitive weights into consideration and data collection satisfies the fenton et.al (1997) properties.

**Table VI.** Complexity Metric Values and Mean Comprehension Time

| Program# | Existing Metric Value (CBO) | Proposed Metric Value (CWCBO) | Mean Comprehension Time |
|---|---|---|---|
| 1 | 3 | 2.3 | 13.5 |
| 2 | 2 | 1.7 | 14.6 |
| 3 | 3 | 3.5 | 15.7 |
| 4 | 5 | 2.7 | 21.5 |
| 5 | 4 | 5.4 | 33.67 |

In order to compare the proposed metric a comprehension test was conducted to rural and urban degree students. There were sixty students who participated in the test; the students were given five different programs in AspectJ for the comprehension test. The test was to find out the output of the given programs. The time taken to complete the test in minutes is recorded. The average time taken by all the students is calculated. In the following Table VI, a comparison has been made with CBO, CWCBO and the comprehension test result.

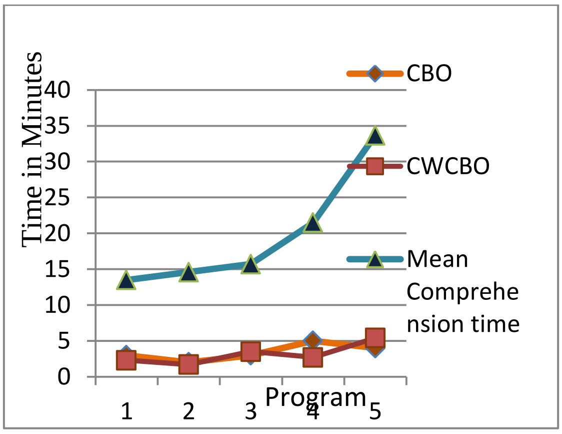


**Figure 1.** Complexity Metric Values Vs Mean Comprehension Time

CWCBO metric is consider various types of coupling for class and aspect and calculated by adding the coupling complexity of classes ($CWCBO_C$) and aspects ($CWCBO_A$). This is better indicator than the existing CBO. The weight of each coupling type is calculated by using cognitive weights and weighting factor of coupling type similar to which is suggested by Wang et al (2014). It is found that the resulting value of CWCBO is larger than the CBO. This is because, in CBO, the weight of each coupling is assumed to be one. However, including cognitive weights for calculation of the CWCBO is more realistic because it considers different types of coupling. The results are shown in the Table V. A correlation analysis was performed between CBO Vs Comprehension Time with r = 0.590448 and CWCBO Vs Comprehension time with r = 0.875912. CWCBO has more positively correlated than CBO.

## VIII. CONCLUSION AND FUTURE SCOPE

A CWCBO metric for measuring the class & aspect level complexity has been formulated. CWCBO includes the cognitive complexity due to different types of coupling. CWCBO has proven that, complexity of the aspect getting affected, which is based on the cognitive weights of the various types of coupling. The assigned cognitive weight of the various types of coupling is validated using the comprehension test. The metric is evaluated through an experimental and proved to be a better indicator of the aspect level complexity. The metrics are persistently used in every state. In future, more metrics can be smeared in cloud environment also.

## IX. REFERENCES


[1]. Ananthi Sheshasaayee and Roby Jose (2015) 'A Descriptive Study about Aspect Oriented Coupling and Cohesion Measures', International Journal of Innovative Science, Engineering & Technology, Vol. 2 Issue 8, August 2015.

[2]. A.Aloysius, G.Arockia Sahaya Sheela (2015) 'Aspect Oriented Programming Metrics-A Survey', IJETCCT, Vol.1, pp.125-130, 2015.

[3]. Aloysius.A,L.Arockiam (2012) 'Coupling Complexity Metric: A Cognitive Approach', MECS, 2012, vol 9, pp.29-35.

[4]. Bartsch, M., Harrison, R (2006) 'An Evaluation of Coupling Measures for AspectJ', LATE Workshop AOSD.

[5]. Ceccato M, and Tonella P (2004) 'Measuring the Effects of Software Aspectization', WARE, 2004.

[6]. Chidamber S.R., Kemerer, C.F. (1994) 'A metrics suit for object oriented design', IEEE, Trans. Software Engineering, vol 20, pp 476-498.

[7]. Eclipse, Inc. (2009) available at https://eclipse.org/aspectj/doc/released/progguide/startingaspectj.html.

[8]. Joseph D.Gradecki, Nicholas Lesiecki (2003) 'Mastering AspectJ – Aspect-Oriented Programming in Java'.

[9]. Kumar, Avadhesh, Rajesh, and P. S. Grover (2009) 'Generalized coupling measure for aspect-oriented systems' ACM SIGSOFT, Software Engineering Notes 34, pp. 1-6.

[10]. Mandeep Kaur (2015) 'Improving the Design of Cohesion and Coupling Metrics for Aspect Oriented Software Development', International Journal of Computer Science and Mobile Computing, Vol.4 Issue.5, May- 2015, pp. 99-106.

[11]. Meneely, A., Smith, B., & Williams, L. (2012) 'Validating software metrics: A spectrum of philosophies' ACM Transactions on Software Engineering and Methodology (TOSEM), 21(4), 24.

[12]. N.Fenton, S.P.fleeger (1997) 'Software Metrics: A Rigorous and Practical Approach' PWS Publishing Company.

[13]. Kiczales, G., Lamping, J., Mendhekar, A., Maeda, C., Lopes, C., Loingtier, J. M., & Irwin, J. (1997) 'Aspect-oriented programming', Springer Berlin Heidelberg, pp. 220-242.

[14]. Kitchenham, B., Pfleeger, S. L., & Fenton, N. (1995) 'Towards a framework for software measurement validation' Software Engineering, IEEE Transactions on, 21(12), pp. 929-944.

[15]. Kulesza, Uira (2006) 'Quantifying the effects of aspect-oriented programming: A maintenance study' Software Maintenance, ICSM'06. 22nd IEEE International Conference on. IEEE.

[16]. K.R.Martin, (2017) 'Coupling on Field Access Metric: A Cognitive Approach', International Journal of Advanced Research in Computer Science and Software Engineering (IJARCSSE), ISSN: 2277 128X, Volume 7, Issue 1, January 2017, pp. 208-212, 2017.

[17]. O'Reilly (2015) 'AspectJ Cookbook'.

[18]. P.K. Singh, O. P. Sangwan, A. Pratap, A. P. Singh (2014) 'An Analysis on Software Testability and Security in Context of Object and Aspect Oriented Software Development', International Journal of Security and Cybercrime, Romania, Vol. 3, Issue 1, pp. 17-28, 2014.

[19]. RamnivasLaddad (2003) 'AspectJ in Action: Practical Aspect-Oriented Programming', Manning Publications co.

[20]. The AspectJ Team (2002) The AspectJ Programming Guide. 2003.

[21]. Zhao, Jianjun, and Baowen Xu (2004) 'Measuring aspect cohesion' Fundamental Approaches to Software Engineering, Springer Berlin Heidelberg, pp. 54-68