Volume 7, Issue 1, January 2017 ISSN: 2277 128X

International Journal of Advanced Research in Computer Science and Software Engineering

Research Paper

Available online at: www.ijarcsse.com

Coupling on Field Access Metric: A Cognitive Approach

[1]K. R. Martin, [2]Dr. E. Kirubakaran, [3]Dr. E. George Dharma Prakash Raj

[1] Department of Computer, Science, St.Joseph's College, Tiruchirappalli, TN, India

[2] AGM (SSTPS), BHEL, Tiruchirapalli, TN, India

[3] Asst Professor in School of CS, Engineering and Applications, Bharathi-dasan University, Tiruchirappalli, TN, India

DOI: 10.23956/ijarcsse/V7I1/0102

Abstract: ***Aspect-Oriented Software Development(AOSD) is a programming paradigm that overcomes the limitations of Object- Orientation (Programming) providing more suitable abstractions for modularizing crosscutting concerns that cannot be decomposed from the rest of the software artifacts. Software metric is a measure of some property of a piece of software or its specifications. Coupling is one of the software metric for Aspect Oriented Software. Coupling measures have important applications in software development and maintenance. They are used to help developers, testers and maintainer's because regarding software complexity and software quality attributes. Coupling or dependency is the degree to which each program module relies on each one of the other modules. This paper presents a new cognitive complexity metric namely Cognitive Weighted Coupling on Field Access for measuring coupling in Aspect Oriented Software.***

Keywords: AOP, AOSD, WMC, LOCM, CBO

I. INTRODUCTION

The aspect-oriented programming (AOP) is a new paradigm for improving the system's features such as modularity, readability and maintainability. Aspect-oriented software development (AOSD) is a new technique to support separation of concerns in software development. AspectJ is an implementation of aspect-oriented programming for Java. It adds to Java only a few new constructs: pointcuts, advice, inter-type declarations and aspects.Coupling is an internal software attribute that can be used to indicate the degree of interdependence among the components of a software system. Coupling is thought to be a desirable goal in software construction, leading to better values for external attributes such as maintainability, reusability, and reliability. Thus, in AO systems, the coupling is mainly about the degree of interdependence among aspects and/or classes.In AspectJ has no **CWCFA** metric to measure the different data types proposed by various researchers. So, there is a need for Cognitive Weighted, Coupling on Field Access (CFA)for the Aspect level attributes measurement. Hence our main goal is to define a CWCFA metric to measure the Complexity of different types of data.

II. LITERATURE REVIEW

Several metrics have been proposed for OO systems by researchers. A metric suite proposed by Chindamber 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 (LOCM) and Coupling Between Objects (CBO) [].

Ceccato and Tonella[7] introduced many aspect oriented metrics which included 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.ArockiaSahayaSheela[10] studied about aspect oriented metrics. This paper addresses the development and implementation of various metrics for AOP design paradigm and outlines the future directions.

Kulesza et al[5] presents a quantitative study that assess the positive and negative effects of AOP on maintenance activities of a web information system. The study also considered the positive and negative influences of AOP on coupling measures when compared to the object oriented version of the same system.

Ananthi Sheshasaayee and Roby Jose[2] 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 [1] 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.

III. METHODOLOGY

A. Existing Metric - Coupling on Field Access (CFA)

Number of modules or interfaces declaring fields that are accessed by a given module. Similarly to CMC, CFA measures the dependences of a given module on other modules, but in terms of accessed fields, instead of methods. In

OO systems this metric is usually close to zero, but in AOP, aspects might access class fields to perform their function, so observing the new value in aspectized software may be important to assess the coupling of an aspect with other classes/aspects.

B. Proposed Metric - Cognitive Weighted Coupling on Field Access (CWCFA)

Several metrics have been proposed for AOP systems by researchers. One of the metric proposed by Ceccato et.al [3] is CFA. CFA counts Number of modules or interfaces declaring fields that are accessed by a given module. This metric does not considered the various data types. The proposed metric called CWCFA, which considers the cognitive complexity of the different data types of character, integer, float, long and double.

CWCFA = IN*WFIN + CH*WFCH + FL*WFFL + LO*WFLO + DO*WFDO

IV. EMPIRICAL METRIC DATA COLLECTION & EVALUATION CRITERIA

This section discusses the CWCFA metric, empirical data, collection statistics, analysis and its implication.

A. CFA Metric

For empirical analysis, CFA metric is selected for AO software. This metric used to find the complexity of various data types using Cognitive Approach.

B. Calibration

In this section, an experiment is conducted to assign cognitive weight to the various data types for attributes. A comprehension test has been conducted for a group of students to find out the time taken to understand complexity of aspect oriented program with respect to different data types. The group of students selected had sufficient exposure in analysing the Aspect and object oriented programs, as they had undergone courses in AspectJ language. 30 students taken from Rural, 30 students taken from Urban were selected 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 4.1.

Table 1 Categorized Average Comprehension time

Program	**Average Comprehension Time (In Minutes)**				
	character	**Integer**	**Float**	**Long**	**Double**
P1	9.78	17	24	29	33
P2	10.4	16.6	24	28	34
P3	11	16	23	28	34
P4	9.9	17	24	28	33.3
P5	10	16	23	27	33.5

The average comprehension time, for programs are listed in Table 1. These programs are based on Aspect Oriented Programming. The mean time is also calculated for each category of the programs and is tabulated.

V. STATISTICAL ANALYSES

For each data types, mean was selected as a measure of central tendency. Table 2 illustrate statistical computation of different data types.

Table 2 Mean values of different data types

Program	**Average Comprehension Time (In Hours)**				
	character	**Integer**	**Float**	**Long**	**Double**
P1	0.16	0.3	0.4	0.5	0.6
P2	0.17	0.28	0.4	0.5	0.6
P3	0.2	0.3	0.4	0.47	0.6
P4	0.2	0.3	0.4	0.5	0.56
P5	0.2	0.3	0.4	0.5	0.56
Average	0.169	0.275	0.389	0.46	0.55
STDEV	0.314	0.335	0.299	0.574	0.1067

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

VI. CWCFA

Several metrics have been proposed for AOP systems by researchers. One of the metric proposed by Ceccato et.al [3] is CFA. CFA counts Number of modules or interfaces declaring fields that are accessed by a given module. This metric does not considered the various data types. The proposed metric called CWCFA, which considers the cognitive complexity of the different data types of character, integer, float, long and double.

CWCFA = IN*WFIN + CH*WFCH + FL*WFFL + LO*WFLO + DO*WFDO -----> (1)

Here,

CWCFA is over all Cognitive Complexity of Coupling on Field Access.
WFIN is Weighting Factor of Integer Data Type.
WFCH is Weighting Factor of Character Data Type.
WFFL is Weighting Factor of Float Data Type.
WFDO is Weighting Factor of Double Data Type.
WFLO is Weighting Factor of Long Data Type.

The Weighting Factor of each data type is calibrated in Table 3 using the method discussed in the Empirical Metric Data Collection. The weight value is calculated based on the mean value of different data type. To normalize the mean value to get appropriate weight value. The following table explained the rounded values of each data type that is called weighting factor of each data type.

Table 3 Weight Value of Each Data Type

Data type	Weight Value
WFCH	0.2
WFIN	0.3
WFFL	0.4
WFLO	0.5
WFDO	0.6

VII. COMPARATIVE STUDY

A comparative study has been made with most widely accepted the metric proposed by Ceccato et.al [3] is CFA. CFA is Number of modules or interfaces declaring fields that are accessed by a given module. The current CWCFA metric is one step ahead of existing CFA metric, because it includes the complexity that arises due to the various data types. Another advantage of CWCFA metric is that, it takes cognitive weights into consideration and data collection satisfies the fentonet.al [7] properties. 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 5, a comparison has been made with CFA, CWCFA and the comprehension test result.

Table 5 Complexity Metric Values and Mean Comprehension Time

Program#	Existing Metric Value (CFA)	Proposed Metric Value (CWCFA)	Mean Comprehension Time
1	1	0.6	13.5
2	3	1.7	14.6
3	2	0.9	15.7
4	4	2.1	21.5
5	5	3	33.67

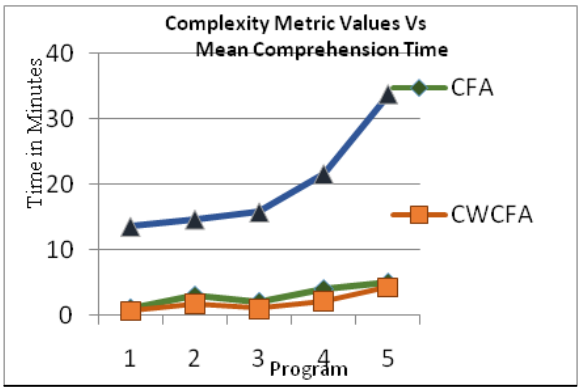

Figure 1 Complexity Metric Values Vs Mean Comprehension Time

CWCFA counts number of modules or interfaces declaring different types of fields multiplied by corresponding weighting factor that are accessed by a given module This is better indicator than the existing CFA. The weight of each data type is calculated by using cognitive weights and weighting factor of data type similar to which is suggested by Wang et al [14]. It is found that the resulting value of CWCFA is larger than the CFA. This is because, in CFA, the weight of each field is assumed to be one. However, including cognitive weights for calculation of the CWCFA is more realistic because it considers different data types. The results are shown in the Table 5. A correlation analysis was performed between CFA Vs Comprehension Time with r = 0.873828 and CWCFA Vs Comprehension time with r = 0.90139. CWCFA has more positively correlated than CFA.

VIII. CONCLUSION

A CWCFA metric for measuring the class & aspect level complexity has been formulated. CWCFA includes the cognitive complexity due to different data types of fields. CWCFA has proven that, complexity of the aspect getting affected, which is based on the cognitive weights of the various data types of fields. The assigned cognitive weight of the various data types of field 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.

REFERENCES

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

[2] The AspectJ Team. The AspectJ Programming Guide. 2003.

[3] Ceccato M, and TonellaP,"Measuring the Effects of Software Aspectization", WARE, 2004.

[4] Kumar, Avadhesh, Rajesh, and P. S. Grover. "Generalized coupling measure for aspect-oriented systems." ACM SIGSOFT Software Engineering Notes 34.3 (2009): 1-6.

[5] Bartsch, M., Harrison, R, "An Evaluation of Coupling Measures for AspectJ", LATE Workshop AOSD, 2006.

[6] N.Fenton, S.P.fleeger, "Software Metrics: A Rigorous and Practical Approach". PWS Publishing Company, 1997.

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

[8] Zhao, Jianjun, and Baowen Xu. "Measuring aspect cohesion." Fundamental Approaches to Software Engineering. Springer Berlin Heidelberg, 2004. 54-68.

[9] AnanthiSheshasaayee and Roby Jose,"A Descriptive Study about Aspect Oriented Coupling and Cohesion Measures", International Journal of Innovative Science, Engineering & Technology, Vol. 2 Issue 8, August 2015.

[10] E. Kirubakaran, K.R.Martin, A Review on Coupling Metrics in Aspect Oriented System, International Journal of Control Theory and Applications (IJCTA), ISSN: 0974-5572, 9(27), 2016, pp. 93-97, 2016.(Scopus Indexed)

[11] G. ArockiaSahaya Sheela., A.Aloysius, Aspect Oriented Metrics – A survey (IJETCCT), International Journal of Emerging Trends in Computing and Communication Technology, (Vol.2): ISSN: 2348 4454.

[12] G.ArockiaSahaya Sheela, A.Aloysius, Design and Analysis of Aspect Oriented Metric CWCAE using Cognitive Approach, International Journal of Engineering Research & Technology (IJERT), ISSN: 2278 – 0181, Vol. 5 Issue 04, April, 2016.(Scopus Indexed)

[13] Mandeep Kaur, "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, pg. 99-106

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

[15] 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.

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

[17] P.K. Singh, O. P. Sangwan, A. Pratap, A. P. Singh, 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.

[18] A.Aloysius, G.ArockiaSahaya Sheela, "A Review on Aspect Oriented Programming Metrics" 2015.

[19] Aloysius.A,"Coupling Complexity Metric: A Cognitive Approach", MECS, 2012, vol 9, pp 29-35.

[20] The AspectJ Team, "The AspectJ Programming Guide",2001. Joseph D.Gradecki, Nicholas Lesiecki, "Mastering AspectJ – Aspect-Oriented Programming in Java", 2003.

[21] RamnivasLaddad, "AspectJ in Action: Practical Aspect-Oriented Programming", Manning Publications co., 2003.

[22] O'Reilly, "AspectJ Cookbook", 2005.

[23] https://eclipse.org/aspectj/doc/released/progguide/startingaspectj.html

ABOUT AUTHOR

K R Martin is working as Assistant Professor in Department of Computer Science, St. Joseph's College (Autonomous), Tiruchirappalli, Tamil Nadu, India. He has obtained the Master of Computer Applications degree in 1999, Master of Philosophy degree in 2007 and Master of Business Administration degree in 2011 from Bharathidasan University, Trichy. He has 18 years of experience in teaching Computer Science. His research areas are Software Engineering and Software Metrics. He has published many research articles in the National/International Conferences and Journals. He is currently pursuing Doctor of Philosophy Program and his current area of research is the Cognitive Complexity of Aspect-oriented Software Metrics.

Dr.E.Kirubakaran obtained his B.E. (Hons.) degree in Mechanical Engineering from Regional Engineering College, Trichy in the year 1978 and joined in Bharat Heavy Electricals Ltd. Trichy in 1979. He obtained M.E. in Computer Science in 1984 and Ph.D. in Computer Science in 1999 from Regional Engineering College. In 2005, he got his M.B.A. degree from IGNOU. For 37 years. He has been working in designing, Developing and maintenance of Software Systems at BHEL, Trichy. He has been a visiting faculty to a number of institutions in and around Trichy. He has also guided a number Research Scholars for Ph.D. He has served as Syndicate member of Bharathidasan University, Academic Council Member in Anna University and Member of Board of Studies in many colleges/ Universities. He had also held the post of Secretary, Vice Chairman and Chairman of Computer Society of India, Trichy.

Dr.E.George Dharma Prakash Raj completed his Masters Degree in Computer Science and Masters of Philosophy in Computer Science in the years 1990 and 1998. He has also completed his Doctorate in Computer Science in the year 2008. He has more than twenty five years of Academic experience and nineteen years of Research experience in the field of Computer Science. Currently he is working as a Faculty in the School of Computer Science and Engineering at Bharathidasan University, Trichy, India. He has published several papers in International Journals and Conferences related to Computer Science and has been an Editorial Board Member, Reviewer and International Programme Committee Member in many International Journals and Conferences. He has convened many National and International Conferences related to Computer Science.

