Assessment of Object Oriented Metrics for Software Reliability

Santanu Kr. Misra
Associate Professor
Computer Science and Engineering Department
Sikkim Manipal Institute of Technology
Majitar, Rangpo, East Sikkim

Bijoyeta Roy
Assistant Professor
Computer Science and Engineering Department
Sikkim Manipal Institute of Technology
Majitar, Rangpo, East Sikkim

***Abstract:-* Software Reliability is defined as the probability of failure-free software operation for a specified period of time in a specified environment. In this paper an approach for early estimation of the reliability of software products based on their design models using Object-Oriented Metrics is discussed. A trained metrics namely CK metrics is used to predict the reliability of individual modules. The final product reliability is obtained from these predicted values. This approach can help to reduce the cost of designing certain modules and testing effort to meet product reliability goals. It helps not only in the minimization of cost but also in the development of high quality software system.**

***Keywords:* Software Reliability, Object oriented metrics, CK metrics.**

I. INTRODUCTION

Object-oriented metrics are used to evaluate and predict the quality of software. It is required for the estimation of the threshold value for reliability. Object oriented metrics are used to maintain the process and the product in various stages of development of the software. These metrics supplies the information that is needed to control resources and processes used to produce the software. Object oriented metrics serves as indicators to represent a metric data that provides a view into an ongoing software system development project. Reliability of a software product denotes trustworthiness or dependability. It can be said that a software product having a large number of defects would be unreliable and the reliability of the system would improve if number of defects in it were reduced. However there is no simple relation between reliability of the system and number of latent defects in the system. Removing defects from parts of the system, which are rarely used, would make little difference to the observed reliability. In order to express reliability of a software product quantitatively we need object oriented metrics. The object oriented metrics contributes for measuring principle structures for the development of software product and, if they are improperly structured and designed it may negatively affect the design and code quality attributes for which the software product cannot perform the required tasks under given conditions for a given period of time. So in this paper Chidamber-Kemerer (CK) metrics suiteis consideredfor estimating reliability of the software product.

II. LITERATURE SURVEY

In the past literature on software reliabilityChidamber and Kemmerer proposed a set of six metrics in 1994 to identify certain design and code in Object Oriented software. In this paper the concepts about different object oriented metrics and how they play important role in maintaining the reliability of software to perform required tasks under given conditions over a specified period of time is discussed[2]. Rosenberg, Linda and Lawrence Hyatt's et.al.in 1995discussed how concepts and structures used in object oriented metrics affect the quality of the software. From this paper an idea is gathered to evaluate the criteria of various metrics for estimating reliability of the software [3].Seyyed Mohsen Jamali's et.al.in 2006 described certain characteristics of object oriented metrics such as localization, encapsulation, information hiding, inheritance, object abstraction which provides dependable guidelines for development of the object oriented software [9].In 2011 ArtiChhikara. R.S. Chhilla discussed several object oriented metrics and effect of those metrics upon the quality of the software product. From this paper an idea is gathered about the working principle and the formulas of the metrics to implement those metrics and to evaluate the complexity of those metrics [10].In June-2013 Johnny Antony's et.al.in their paper discussed how the object oriented metrics are most beneficial and reliable for the estimation of threshold value for reliability of the software product. This paper gavethe idea to extract the metric parameters from the source code and those metric values are evaluated to provide the threshold for the software reliability. The result provided a standard against which the software reliability can be evaluated and necessary corrective actions can be implemented [11].

III. CHIDAMBER-KEMERER(CK) METRICS SUITE

This metric suite provides information about the positions of the software development project and checks whether developers are following object oriented principles in their design. They claim that using several of their metrics collectively assists managers and designers to make better design decision. CK metrics have shown a significant amount of interest and is considered as the most well known suite of measurements for Object Oriented software. Chidamber and Kemmerer proposed six metrics.

(a) Weighted Methods per Class (WMC):

It is defined as the sum of the complexities of all methods of a class. WMC is intended to measure the complexity of a class. Therefore, it is an indicator of the effort needed to develop and maintain a class. Classes with large number of methods need more time and effort to develop and maintain the class because it will depend on subclasses inheriting all the methods.

(b) Coupling between Objects (CBO):

According to this metric "Coupling between Object Classes" (CBO) for a class is a measure of the number of other classes to which it is coupled, i.e. a class is coupled with another if the methods of one class use the methods or instance variables of the other class. If Coupling between Objects (CBO) increases then the reusability of a class will decrease. Thus, the CBO values for each class should be kept as low as possible.

(c) Response set of a Class (RFC):

It is number of methods in the set of all methods that can be called with respect to a message sent as a request to the object of a class. It includes all methods available within the class hierarchy.

(d) Depth of Inheritance Tree (DIT):

It is defined as the maximum length from the node to the root of the tree. Depth depends upon the number of length of maximum ancestral classes from the class node to the root of the inheritance tree.

(e) Number of Children (NOC):

It is actually the number of subclasses which inherits the methods from the parent class. NOC measures the number of immediate subclasses subordinate to a class in a class hierarchy and the number of subclasses belonging to that class.

(f) Lack of Cohesion (LCOM):

Lack of Cohesion of Methods (LCOM) defines the lack of the cohesion of class. Cohesion actually measures how closely the local methods are related to the instance variables of that class. It is the measure of the relative strength of the module.

IV. PROPOSED METHODOLOGY

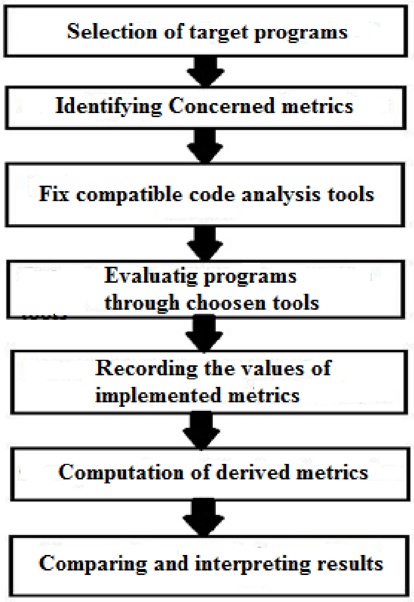

Selection of target programs
Identifying Concerned metrics
Fix compatible code analysis tools
Evaluatig programs through choosen tools
Recording the values of implemented metrics
Computation of derived metrics
Comparing and interpreting results

Figure 1: Steps of methods followed for computation of metrics.

Here literature review of various papers is done to analyze the target program in order to extract some metric parameters from the source code with the help of some metrics to achieve the threshold value for gaining the reliability of the software product. Reliability is obtained by comparing the values of the implemented metrics with the value of the derived metrics. After that it is decided about the compatible code analysis tools which can help to analyze the source code of the program to extract the metric value from it and evaluate the programs through those selected code analysis tools. Then the values of the metrics are recorded to compare with the value of the derived metrics. The derived metrics are then computed to compare and interpret the result which provides the threshold value for the reliability of the software.

CK metric suite is selected for estimating the threshold for Reliability of the software. The threshold value of the CK metrics based on the experience and the principle that if the metric values are too low may represent poor utilization of advantages of Object-Oriented technology and too high values may represent too much complexity. CK metrics values are assigned weighted values. Threshold for the reliability (R) is calculated using relationship established between Reliability and CK metrics. They are like:-

Reliability α 1/WMC
Reliability α 1/RFC
Reliability α 1/DIT
Reliability α 1/LCOM
Reliability α 1/CBO

It can be said that, if R-min (Reliability minimum) < R-value < R-max (Reliability Maximum), then p i.e. program is reliable.The first step towards the experiment is to propose threshold values for all CK metrics. The threshold value considered for the three metrices are given in the tablebelow:

TABLE 1: Threshold values of the metrics

	WMC	DIT	NOC
Threshold	6-30	1-6	1-3

a) Assigning weighted values to the metrics

There have some rules to assigning weighted values to the metrics. They are,

Rule -1

If Value of Metric lies between the lower limit and (Mean of lower limit and upper limit) of the threshold, then the Weightage given to Metric is 1. Mathematically: If (Lower Value of Threshold ≤ Value of Metric ≤ Mean of Threshold), then Weightage (Metric) = 1.

Rule -2

If Value of Metric lies between the (mean of lower limit and upper limit) and upper limit of the threshold, then Weightage given to Metric is 2. Mathematically: If (Mean of Threshold ≤ Value of Metric ≤ Upper Limit of Threshold), then Weightage (Metric) =2.

Rule -3

If Value of Metric lies outside the Threshold, then the Weightage given to Metric is 7.

Rule -4

In the case of NOC, $(\log(\text{upper threshold}))^2$ is consideredfor R(Max)and (log(lower threshold))2 is considered for R(Min).If any of the CK metric value is outside the thresholds, then this metric is neglected.

b)Calculating threshold of Reliability using Rule 1 to Rule 4

The threshold value of reliability can be calculated as below,

$R\text{-value}=k*(1/ (wt (WMC) +wt (DIT)) + (\log (wt (NOC)))^2$ …… (1)

R-value is the reliability value.

The lower limit value of reliability i.e. R (Min) can be calculated as below,

$R(Min)=k*(1/(wt(WMC)+wt(DIT))+(\log(L\text{-}Lt(NOC)))^2$ ……(2)

The upper limit value of reliability i.e. R(Max) can be calculated as below,

$R(Max)=k*(1/(wt(WMC)+wt(DIT))+(\log(U\text{-}Lt(NOC)))^2$. ……. (3)

It is considered that, k is always 1.

The Threshold for reliability of software based on the relationship of Reliability and CK Metrics lies between R(Min)<R-value<R(Max).

Suppose, 5 programs P1, P2, P3, P4 and P5 having their own WMC, DIT and NOC. Now, the reliability for P1 is calculated. Here, in table 1it is seen that the program P1 having WMC=13, DIT=3 and NOC=3. So, for calculating the reliability the rules for providing the threshold value to all the metrics are used. It is assumed that the threshold for WMC is 6 to 30. Then, the WMC of P1 i.e. 13 is greater than 6 i.e. the lower limit of the threshold and less then the mean of the threshold i.e. (6+30)\2=18. So, the weightage of WMC i.e. wt (WMC) = 1. Again, for DIT the threshold 1 to 6 is considered and program P1 having DIT=3. As 3 is greater than 1 i.e. the lower limit of the threshold and less than mean of the threshold i.e. (1+6)\2=3.5. So, the Weightage of DIT i.e. wt (DIT) is 1. Now, for NOC the threshold 1 to 3 is taken and program P1 is having the NOC=3. As 3 is greater than 1 i.e. the lower limit of the threshold and greater than the mean of the threshold i.e. (1+3)\2=2 so it will follow 2nd rule. Now, as 3 is greater than 2 i.e. the mean of the threshold and less than 6 i.e. the upper limit of the threshold. Then, NOC got the Weightage i.e. wt (NOC)=2.So,the reliability of the P1(R value)is calculated by taking K=1.

$Rvalue=k*(1/(wt(WMC)+wt(DIT))+(\log(wt(NOC)))^2$

Then, R-value of $P1=1*(1/(1+1))+(\log(2)))^2=0.9804$

Now, R (Min) and R(Max) of P1 is calculated as

$R(Min)=k*(1/(wt(WMC)+wt(DIT))+(\log(L\text{-}Lt(NOC)^2$

Considering K=1,
$R(Min)=1*(1/(1+1))+(\log(1)))^2 =0.5$

Similarly R(Max) is calculated as
$R(Max)=k*(1/(wt(WMC)+wt(DIT))+(\log(U\text{-}Lt(NOC)))^2$

Considering K=1,
$R (Max) =1*(1/(2+2))+(\log(3)))^2=1.4629$

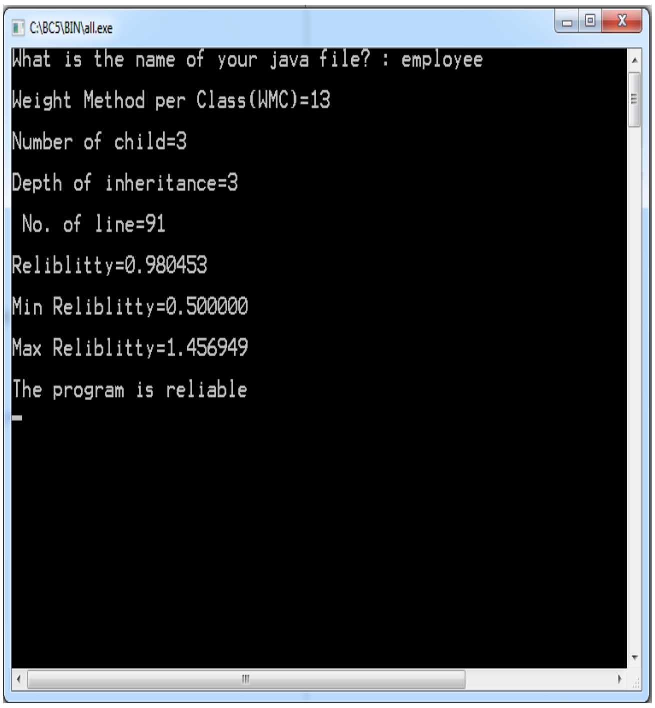
Here, for R (Min) and R (Max) the minimum weightage and maximum weightage of WMC and DIT that is lower limit and upper limit are taken respectively [9].

So, as 0.9804 i.e. R-value of P1 is greater than the 0.5 i.e. R (Min) and less than 1.4629 i.e. R(Max), it can be said that P1 is reliable. Similarly it can be checked that another programs i.e. P2, P3, P4 and P5 are reliable or not.

TABLE 2: Reliability table

Metrics	P1	P2	P3	P4	P5
WMC	13	12	7	28	18
DIT	3	2	2	5	3
NOC	3	3	1	2	1
Reliability	0.980	0.427	0.500	0.202	0.940

RESULTS AND DISCUSSIONS



```
What is the name of your java file? : employee
Weight Method per Class(WMC)=13
Number of child=3
Depth of inheritance=3
 No. of line=91
Reliblitty=0.980453
Min Reliblitty=0.500000
Max Reliblitty=1.456949
The program is reliable
```

Figure 2: Obtaining Reliability after implementation of all the Metrics

Here after implementing four metrics Lines of Code (LOC), Depth of Inheritance (DIT), Number of Children (NOC) and Weighted Methods per Class (WMC) by taking a target input java program named employee.java, the values of the different metrices obtained are as follows:

Weight Methods per Class (WMC) = 13

Number of Children (NOC) = 3

Depth of Inheritance (DIT) = 3 and

Total Lines of Code (LOC) =91.

And, by using the rules to calculate the Reliability in the C program, the Reliability of the targeted JAVA program “employee” is found to be 0.980453 and also the maximum and minimum reliability is found out. The maximum reliability achieved is 1.456949 and the minimum reliability achieved is 0.500000. By checking whether the reliability value of the target JAVA program falls within the minimum and maximum reliability valuesi.e 0.980453 lies between 0.500000 and 1.456949.Therefore it is concluded that the target JAVA program is reliable.

SUMMARY AND CONCLUSION

In this paper Lines of Code (LOC), Depth of Inheritance (DIT), Number of Children (NOC), Weighted Methods per Class (WMC), Response set For Class (RFC) is implmented for maintaining the reliability of the software system. This study made an assessment of the relationship between CK metrics and the reliability of the object oriented software system. Here CK metrics suite is taken into considerationfor estimating the threshold of reliability. This study suggests that by keeping the value of NOC, DIT, LOC, WMC, RFC within the threshold, high reliability of the system can be attained. Thus it can be concluded that CK metric parameters are useful indicators for predicting the reliability and quality of the system.

REFERENCES

1. M. Lorenz and J. Kidd, “Object-Oriented Software Metrics”, Prentice Hall, 1994.
2. S. R. Chidamber and C. F. Kemerer, “A Metrics Suite for Object-Oriented Design,” IEEE Transactions on Software Engineering, Vol 7 No. 05 June 1994.
3. Rosenberg, Linda and Lawrence Hyatt, “Software Quality Metrics for Object-Oriented System Environments,” SATC, NASA Technical Report SATC-TR, 1995.
4. Victor R. Basili, Lionel Briand, Walcelio L. Melo, “A Validation of Object Oriented Design Metrics as Quality Indicators”, Department of Computer Science, University of Maryland, USA, April 1995.
5. Fernando Brito Abreu, Rita Esteves,“Design Of Eiffel Programs-Quantitative Evaluation Using The Mood Metrics”, University Of California, USA, July 1996.
6. Jubair J. Al-Ja'afer and KhairEddin M. Sabri,”Chidamber-Kemerer (ck) and Lorenz & J. Kidd”, IEEE transactions on Software Engineering, Vol 5, May 1998.
7. Daniel Rodriguez, Rachel Harrison: “An Overview of Object Oriented Design Metrics”, The University Of Reading Computer Science (RUCS) /TR/ A March 2001.
8. Bansiya J. and C. G. Davis: “A Hierarchical Model for Object-Oriented Design Quality Assessment”, IEEE Transactions on Software Engineering, 28, (1), (2002), pp. 4–17.
9. Seyyed Mohsen Jamali: “Object Oriented Metrics”, Department Of Computer Engineering, Sharif University Of Technology, January 2006
10. ArtiChhikara, R. S. Chhilla and Sujata, “Prediction of the Quality of Software product Using Object Oriented Metrics”, International Journal of Computer Engineering and Software Technology Vol.2, Issue 1, June 2011; pp. 1-6.
11. Johny Antony P,”Predicting Reliability of Software Using Thresholds of CK Metrics”, International Journal of Engineering Research & Technology (IJERT) ISSN: 2278-0181 Vol. 2 Issue 6, June – 2013.