

OBJECT ORIENTED MAINTAINABILITY AND TESTABILITY MEASUREMENT USING ANALYTIC HIERARCHY PROCESS

ABSTRACT

Applying AHP to measure object-oriented design quality has been verified. The AHP is utilized in two stages of measurement process. First, in defining relative quality based on metric values that has been converted to AHP's pairwise comparison scheme; second, in expert judgment verification by answering pairwise preference questionnaire. First stage measurement is conducted based on two sets of object-oriented metrics, i.e. MOOD and MOOSE. The results of quality rank over a number of OO design samples are proven consistent by applying MOOD and MOOSE as the criteria. Moreover, experts' judgment on relative quality of selected samples supports the robustness of AHP in measuring maintainability and testability of OO programs.

Keywords: object oriented, testability, AHP.

1. Introduction

Object oriented (OO) system possesses specific properties that matches good OO characteristics. Several research groups have introduced metrics to measure whether good OO characteristics are embedded in a software system. Among many OO metrics available, there are two popular sets, i.e. Metrics for Object-Oriented Design (MOOD) and Metrics for Object-Oriented Software Engineering (MOOSE). The metrics deliver a set of quantitative values, each of them represents specific property of OO embedded in the software being measured. Considering those metrics as the multi criteria, [1] proposed the use of AHP to derive a single quantitative value that represents overall degree of quality. Applying the mechanism to a number of OO programs would result in rank of quality based on their OO characteristics. This method has become a useful aid tool for marking student works in "Foundation of Programming" class. This is referred to as **stage one** for further reference in this paper.

In **stage two**, the validity of quality rank produced by the above-mentioned tool would be verified by comparing it to expert judgment on maintainability and testability features of the programs. Inspired by the work of Khanna in [2], expert's perception on software maintainability and testability is derived from pairwise preference according to AHP

scheme. The rank derived from the experts is compared to the rank delivered in stage one.

This paper reports the result of verifying stage one's rank by conducting survey by means of questionnaire to experts in stage two. The results would greatly ensure that OO Metrics dan AHP form a robust mechanism for quality measurement.

2. Literature Review

The emerging object-oriented paradigm has led to a number of metrics to measure the possessiveness of OO properties. Two popular OO metrics are MOOD and MOOSE. MOOD are used to represent quality of an integrated system [3] consists of six metrics: Method Hiding Factor (MHF), Attribute Hiding Factor (AHF), Method Inheritance Factor (MIF), Attribute Inheritance Factor (AIF), Polymorphism Factor (POF), Coupling Factor (COF). MOOSE are used to measure quality of each class as the component of OO system [4] consists of Weighted Method per Class (WMC), Depth of Inheritance Tree (DIT), Number of Children (NOC), Coupling between Object Classes (CBO), Response set For a Class (RFC) and Lack of Cohesion in Methods (LCOM). Christariny developed an OO Metrics Calculator based on predefined formulas that are provided for each of the metrics [5].

The above-mentioned metric sets have been evaluated in [6] to define ranks of quality over a number of students' works from a "Foundation of Programming" class in Faculty of Computer Science at Universitas Indonesia. Metrics were used to evaluate students' works in three phases, each of which is named after the assignment complexity that was gradually increasing, covering Encapsulation (E), Inheritance (EI), and Polymorphism (EIP). Figure 1 depicts the three phases of the assignment during one semester of lectures. For every phase, weights of criteria were set to suit specific feature being evaluated. Predefined weights of the criteria is shown in Table 1.

Metrics values collected in each phase of assignment were converted to pairwise preference scale of 1 up to 9 according to AHP scheme. The results of applying AHP were consistently shown for the best 20 students' work.

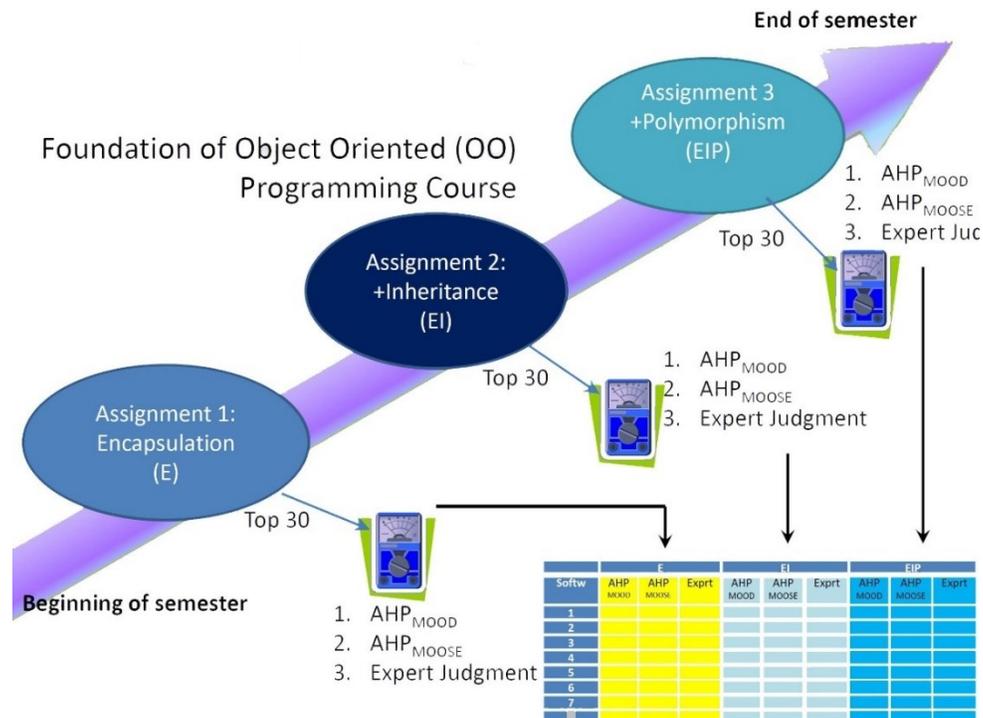


Figure 1. Experimental Scenario

3. Hypotheses/Objectives

A good design of object-oriented software is the one who has the most degree of possessiveness on OO characteristics. Better design which is on top of quality rank must have more comprehensive properties including higher maintainability and easier testability compared to those in lower quality ranks. Blind review by experts on the samples of students' work would expectedly endorse the rank produced by the experiments in [6].

4. Research Design/Methodology

From previous research at [6], the best 20 of students' works were consistently ranked by applying AHP. The result is reevaluated to verify their relative quality by inviting experts to answer a questionnaire assessing maintainability and testability features of the program samples. In each question, the experts must define their relative preference over a pair of design samples. To obtain a contrast comparison, we took only 5 out of 20 best samples in rank 1st, 5th, 10th, 15th and 20th.

Questions that must be answered by experts are referred to work by Khanna in [2]. Some of the questions included are: **In which program design is easier to locate part of code for extending the feature of XXX? Compare each of the following pairs and mark your preference to represent the intensity of preference in the scale of 1 to 9.** The whole process of applying AHP scheme uses the Super Decision tool available in [7].

Expert judgment reported in [6] is referred to manual marking by the lecturers, whereas expert judgment in this paper referred to as AHP's pairwise comparison stage two of the assessment process. All subsets of samples were randomly assessed and blindly reviewed by the experts in answering the questionnaire.

5. Data/Model Analysis

Three phases of assignments were conducted in stage one, each of which applied the criteria weight accordingly as shown in Table 1.

Table 1. Weight of criteria in every phase of assignment

Sample set	E	EI	EIP
Encapsulation	0.700	0.618	0.594
- MHF	0.500	0.500	0.500
- AHF	0.500	0.500	0.500
Inheritance	0.100	0.238	0.263
- MIF	0.500	0.500	0.500
- AIF	0.500	0.500	0.500
Polymorphism	0.100	0.044	0.081
- POF	1.000	1.000	1.000
Coupling	0.100	0.100	0.062
- COF	1.000	1.000	1.000
Inconsistency ratio	0.00	0.21	0.03

Hierarchical criteria applied in stage one is shown in Figure 1, for the measurement based on MOOD as well as MOOSE.

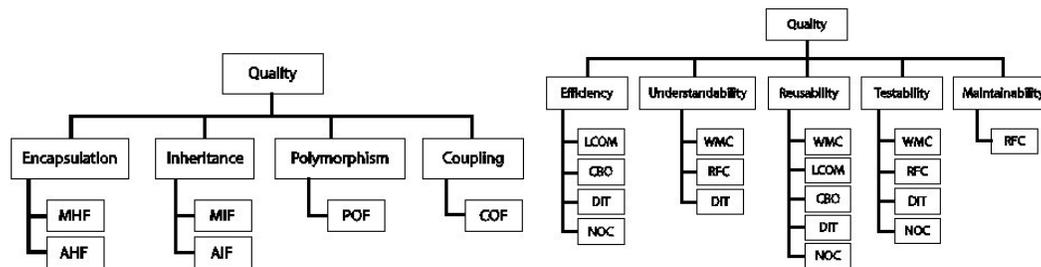


Figure 1. Hierarchical criteria of a) MOOD and b) MOOSE

6. Limitations

The measurement tools employed in stage one is designed for executable Java codes only. Verification was only applied to program samples taken from the best 20 of students' works. For the rest, this measurement mechanism delivers uncertain results due to poor possessiveness of OO characteristics in the program. Significant difference of metric values is required to make contrast comparison between each pair of the criteria.

7. Conclusions

This paper reports utilization of AHP in two stages of OO quality measurement. The result of stage one shows a consistent relative quality over a number of Java programs, measured by MOOD as well as MOOSE metrics. In addition, applying AHP in stage two reconfirm that relative quality among the subset samples are held consistently with the one delivered by stage one. The overall result demonstrates the role of AHP combined with OO metrics to form a robust mechanism in OO quality measurement.

8. Key References

- [1]P. Mursanto (2013) Accuracy Improvement of Object Oriented Software Design Measurement using Analytic Network Process, *Int'l Symposium on Analytic Hierarchy Process (ISAHP)*, Kuala Lumpur, 1 July.
- [2]P. Khanna (2014). Testability of Object-Oriented Systems: An AHP-Based Approach for Prioritization of Metrics. *Proceeding of Int'l Conference on Contemporary Computing and Informatics (IC3I), SJCE, Mysore India, 27-29 Nov, 273–281.*
- [3]Abreu F.B., Esteves R., and Goulao M. (1996) “The design of eiffel programs: Quantitative evaluation using the mood metrics,” Proc. of TOOLS, (USA).
- [4]S. Chidamber and C. Kemerer (1994) “A metrics suite for object oriented design,” *IEEE Transactions on Software Engineering*, 20(6):476-493.
- [5]Christariny (2004) Metrics Calculator for Object Oriented System, *Final Project*, Faculty of Computer Science, Universitas Indonesia.
- [6]P. Mursanto (2017) AHP for Quality Measurement of Object Oriented Software Design, *11th Int'l collaboration Symposium on Information, Production, and Systems (ISIPS)*, Kitakyushu, 14-16 Nov.
- [7]C. D. Foundation (2018) “Super decisions.” available at Jan 25th, 2018 on <https://www.superdecisions.com>.