Response to ‘More comments on: A cohesion measure for object-oriented classes’‡

Heung-Seok Chae1,∗,†, Yong-Rae Kwon2 and Doo-Hwan Bae2

1R&D Center, Tong Yang Systems Corporation, 24, Ogum-dong, Songpa-gu, Seoul 138-130, Korea
2Division of Computer Science, Department of Electrical Engineering and Computer Science, Korea Advanced Institute of Science and Technology (KAIST), 373-1, Kusong-dong, Yusung-gu, Teajon 305-701, Korea

SUMMARY

The authors insist that monotonicity is a necessary property of a good cohesion metric and the violation of the monotonicity property limits the application of CBMC. They also state that the augmented CBMC can also be used as a guideline for quality evaluation and restructuring of poorly designed classes. This paper raises the question about the necessity of monotonicity by analyzing the reason that causes CBMC to violate the monotonicity property. In addition, we give a detailed description of the restructuring procedure based on CBMC. Copyright© 2003 John Wiley & Sons, Ltd.

KEY WORDS: class; object-orientation; cohesion; interaction patterns; restructuring

1. INTRODUCTION

CBMC was proposed to overcome the limitation of the existing cohesion measures by incorporating the characteristics of object-oriented classes [1–3]. In the first comments [4], the authors insisted that CBMC does not satisfy the monotonicity property and proposed the augmented CBMC by adopting a cut set instead of glue methods. In the response paper [5], we posed a question on the necessity of the monotonicity and emphasized the role of glue methods with respect to the evaluation and restructuring of poorly designed classes.

In the second comment, the authors described the meaning and the importance of the monotonicity property. They also exemplified the application of the augmented CBMC to quality evaluation and restructuring of classes. Their second comment can be summarized as follows.

(1) The monotonicity is a necessary property of a good cohesion metric. The CBMC does not fulfill monotonicity. This actually means that CBMC is ill-defined, which limits its application in the evaluation of the design quality of classes.

∗Correspondence to: Heung-Seok Chae, R&D Center, Tong Yang Systems Corporation, 24, Ogum-dong, Songpa-gu, Seoul 138-130, Korea.
†E-mail: hschae@salmosa.kaist.ac.kr
‡The more extensive paper can be found at http://salmosa.kaist.ac.kr
function Restructure(\(C : \text{class}\)) returns \([C', C_1, \ldots, C_n : \text{class}\]) begin
create \(C'\) such that \(C'\) is equivalent to \(C\)
\(M(C') = M(C)\)
\(V(C') = V(C)\)
\(S = \{G^r_i | G^r_i \text{ is the reference graph at the child nodes of } T_s(G^r(C))\}\)
for each \(G^r_i\) in \(S\) begin
if (CBMC\((G^r_i)\) is sufficiently high and can be a model of an entity) then begin
create new class \(C_i\) for \(G^r_i\) such that
\(V(C_i) = V(G^r_i)\)
\(M(C_i) = M(G^r_i) \cup \{m | m \in M_s(C) \text{ and } I(m) \subseteq V(C_i)\} \cup \{m_o(v) | v \in V(C_i)\}\)
update \(C'\) such that
\(V(C') = [V(C') - V(C_i)] \cup \{o_i | T(o_i) = C_i\}\)
for each \(m_i\) in \(M(C_i)\)
replace the implementation of the corresponding \(m_i\) in \(M(C')\) with \(o_i.m_i\)
for each \(m_i\) in \(M(C')\)
replace each occurrence \(\{v | v \in V(C_i)\}\) in \(m_i\) with \(o_i.m_a(v)\)
end
end
end

Definitions:
- \(M(C)\) or \(M(G^r(C))\): the set of methods in class \(C\) or reference graph of class \(C\)
- \(V(C)\) or \(V(G^r(C))\): the set of instance variables in class \(C\) or reference graph of class \(C\)
- \(T_s(G^r):\) the structure tree for reference graph \(G^r\)
- \(M_s(C)\): the set of special methods in class \(C\)
- \(I(m)\): the set of instance variables that are directly or indirectly referenced by method \(m\)
- \(m_o(v)\): accessor method to instance variable \(v\)
- \(T(o_i)\): the class of object \(o_i\)

Figure 1. The restructuring procedure based on CBMC.

(2) The augmented CBMC could be used as a guideline for quality evaluation and for restructuring of poorly designed classes.

In this paper, we clarify the argument on monotonicity by analyzing the fundamental reason for the violation of monotonicity. In addition, CBMC was originally designed to evaluate the design quality of classes and to restructure the poorly designed classes. We emphasize the better application of CBMC to the evaluation of the design quality and the restructuring of classes by describing the restructuring procedure.

2. RESTRUCTURING BASED ON CBMC

Originally, CBMC was designed to help to identify poorly designed classes and restructure them [1]. The class with low \(F_c\) and high \(F_s\) can be restructured by creating a new class for the subreference graph of high cohesion. Figure 1 shows the restructuring procedure that is thoroughly defined based on CBMC.
For example, Figures 2 and 3 show the newly defined classes CTires and CEngine when the restructuring procedure is applied to class Car that is used as an example in the second comment paper. Figure 4 shows the reference graph of class Car3, a class produced after restructuring class Car. Two dotted ellipse tires and engine denote the objects of new classes CTires and CEngine, respectively. In the reference graph, the rectangle with rounded edge denotes special method.

In summary, the fundamental concepts behind CBMC directly address the restructuring procedure.

- The restructuring procedure is applied to classes with a weak connectivity factor and a strong structure factor.
- The subreference graph with strong CBMC can be a candidate for a new class.
- The members of the newly defined class correspond to the subreference graph.

The augmented CBMC can be partly used for restructuring poorly designed classes. However, the restructuring based on the augmented CBMC can miss some candidate classes that are captured
by the original CBMC. Figure 5 shows the structure tree $T'_s(Car)$ for class $Car$ by cut-set-based decomposition. The subreference graphs $G'_1$ show the distinction between the original CBMC and the augmented CBMC. In the structure tree based on the original CBMC, a new class $CTires$ is created. However, the corresponding subreference graph $G'_1$ is not a candidate for restructuring.

The analysis of the structure tree $T'_s(Car)$ reveals that method $Drive()$ in the $G'_1$ makes it difficult to identify a new class from $G'_1$. The method $Drive()$ actually binds together the two candidate classes, $CTires$ and $CEngine$, in class $Car3$. In the case of the structure tree based on the original CBMC, the glue method $Drive()$ is removed from the reference graph and thus two constituent classes are easily identified. In contrast, in the cut-set-based decomposition approach glue methods still remain in one of the two subreference graphs and, thus, the remaining glue methods prevent the containing subreference graph from being a candidate constituent class. Moreover, the structure tree produced by cut-set-based decomposition always leads to only two subreference graphs. This point also can make it difficult to identify several constituent classes.

In addition, we believe that the fundamental rationale for the recursive application of CBMC to the subreference graphs is the exact equivalence of the subreference graph to the candidate constituent class. However, in the augmented CBMC, the subreference graphs may not match the candidate constituent class. Therefore, we still are not sure that there is an admissible rationale in the recursive application of cut-set-based decomposition to subreference graphs.

Figure 5. $T'_s(Car)$. The structure tree for class $Car$ by cut-set-based decomposition.
Figure 6. The structure trees for classes $C_1$, $C_2$ and $C_3$.

3. RATIONALES ON CBMC

Monotonicity means that the addition of a relation into a class should not decrease the cohesion of the class [6]. We will describe the monotonicity property of CBMC using three classes $C_1$, $C_2$ and $C_3$ introduced in the first comment [4]. Figure 6 shows the structure trees for classes $C_1$, $C_2$ and $C_3$.

CBMCs for these classes show two phenomena that are inconsistent from the viewpoint of monotonicity and/or intuition.

- **P1.** $\text{CBMC}(C_2) < \text{CBMC}(C_1)$. The phenomenon $P1$ can be generalized as follows. Assume class $C$ has subreference graphs $G^1_r$, $G^2_r$ and $G^k_r (k \geq 3)$ and each of them has maximum cohesion. When there exists a relation between two subreference graphs $G^i_r$ and $G^j_r$ where
  
  \[ |M(G^i_r)| \times |V(G^j_r)| + |M(G^j_r)| \times |V(G^i_r)| > 2,\]

  two subreference graphs $G^i_r$ and $G^j_r$ are combined into another subreference graph $G^k_r$. Since the combined reference graph $G^k_r$ is no longer a Most Cohesive Component (MCC) [3], the overall cohesion of class $C$ decreases.

- **P2.** $\text{CBMC}(C_3) = \text{CBMC}(C_1)$. The phenomenon $P2$ can be generalized as follows. When relations are added into the reference graph $G^k_r$, the CBMC will increase continuously. When the number of relations between $G^i_r$ and $G^j_r$ reaches
  
  \[ |M(G^i_r)| \times |V(G^j_r)| + |M(G^j_r)| \times |V(G^i_r)|,\]

  $G^k_r$ becomes an MCC, which results in the overall cohesion being equal to that of the original class $C$. 

These two phenomena are due to the computational approach to the structure factor. The structure factor is defined as the average cohesion of all the subreference graphs without regard to the number of subreference graphs. Therefore, the CBMC of class $C_3$ with two subreference graphs is equal to that of class $C_1$ with three subreference graphs (phenomenon $P2$).

The larger number of subreference graphs may seem to indicate less cohesion. However, we believe that the number of subreference graphs has no relation to the well-designedness because the large objects of many attributes and behaviors captured by a class inherently require the corresponding number of instance variables and methods. Therefore, we believe that the structure factor does not depend on the number of subreference graphs. It is analogous to the rationale that the number of special methods has no influence on the computation of the connectivity factor. In addition, classes $C_1$ and $C_3$ are restructured so that the CBMCs of the classes are equal. This is another reason why we believe that the number of subreference graphs is not important in the evaluation of cohesion.

In contrast, the CBMC depends on the cohesion of the subreference graphs; that is, the subreference graph of low CBMC reduces the CBMC of the class. According to this rationale, class $C_1$ should be more cohesive than class $C_2$ because the subreference graphs of class $C_1$ are more cohesive than those of class $C_2$ (phenomenon $P1$). In the structure tree by glue methods decomposition, a less cohesive reference graph such as $G_1^2(C_2)$ is recursively decomposed into another subreference graph until each subreference graph reaches the maximum cohesion. Therefore, the CBMC is inversely proportional to the depth of the structure tree.

4. CONCLUSION

As the authors mentioned in the second comment, the augmented CBMC can be used as a guideline for evaluating and restructuring poorly designed classes. However, the restructuring method based on the augmented CBMC may not be so precise and accurate as our restructuring method because CBMC is originally designed with the quality evaluation and restructuring of classes in mind.

We admit that our CBMC does not satisfy the monotonicity property. However, we believe that CBMC can be useful for evaluating the design quality of classes and the restructuring of poorly designed classes in object-oriented systems.

REFERENCES
