

참조 그래프에 의한 클래스 응집도 척도 분석

김성애**, 최완규*, 이성주**

**조선대학교 전자계산학과

*광주대학교 컴퓨터전자통신공학부

e-mail: sakim@cafe.chosun.ac.kr

Analysis of Class Choeshion Measure by Reference Graph

Sung-Ae Kim**, Wan-Kyoo Choi*, Sung-Joo Lee**

**Dept. of Computer Science, Cho-Sun University

*Div. of Computer Electronic&Communication Eng. ,Kwang-Ju
University

요약

본 연구에서는 클래스의 응집도를 측정하는 기존 척도들을 클래스의 참조 그래프에 근거하여 분석하였다. 기존의 척도들이 참조 그래프의 관점에서 어떠한 문제점들이 있는가를 분석하고, 기존 척도들의 분석에 근거하여 클래스 응집도를 측정하기 위해 응집도 척도가 고려해야 할 사항들을 제시하였다.

1. 서론

객체지향 패러다임에서 소프트웨어 품질, 프로그램 생산성, 프로그램 복잡도와 같은 다양한 속성들을 측정하기 위한 척도들이 제안되었다. 또한 더욱 신뢰성 있고 유지보수 가능한 소프트웨어를 생산하기 위해 연구자들은 응집도라고 불리는 또 다른 속성을 측정해왔다[1]. 응집도는 모듈 구성 요소들의 연관성이라고 하는 소프트웨어 속성으로 응집도가 높을수록 소프트웨어는 이해하고 유지보수하기가 쉽다[2].

ADT에서의 응집도를 측정하기 위한 여러 연구들[1, 3, 4, 5, 6, 7, 8, 9, 10, 11]이 제안 되었다. 이런 연구들은 클래스의 응집도는 클래스 내의 데이터 요소들간의 상호작용의 함수 또는 클래스에 의해 데이터 요소들이 사용되어지는 방법이라고 가정하고, 데이터 요소들 간의 상호작용을 중재하는 메소드들의 연결성의 정도에 의해 응집도를 측정한다. 그러나 기존의 객체지향 패러다임을 위한 응집도 척도들은 응집도에 대한 우리의 직관 또는 이해와 일치하지 않는 경우가 발생한다.

따라서, 본 연구에서는 응집적인 클래스에 대한

정의들과 참조 그래프의 관점에서 기존 척도들의 문제점을 분석하고, 참조 그래프에 근거한, 즉 클래스 내의 데이터 요소들 간의 관계에 근거한 응집도 척도들이 고려해야 할 사항들을 제시한다.

2. 응집적인 클래스

Embley[12]는 추상데이터 타입(abstract data types: ADT)에 대한 참조그래프에 근거하여 ADT에 대한 5가지의 응집도 범주를 정의하였다. 즉, 분리가능 응집도(separable cohesion), 다면적 응집도(multifaceted cohesion), 비대표적 응집도(non-delegation cohesion), 은닉 응집도(canceled cohesion), 이상적 응집도(model cohesion).

ADT의 도메인 집합을 D, 오퍼레이션(즉, 메소드) 집합을 P라 할 때, ADT에 대한 참조 그래프는 방향성 그래프 $G=(V, E)$ 이다. 여기서, 정점들의 집합 $V=DUP$ 이고, 간선들의 집합 E는 오퍼레이션과 도메인간에 참조 관계를 나타낸다.

Embley에 의하면, 분리가능 응집도를 갖는 클래스보다 다면적 응집도를 갖는 클래스가 응집도가 높고, 이상적 응집도를 갖는 클래스가 가장 바람직한 클래스이다.

이상적 응집도를 갖는 클래스는 단 하나의 도메인만을 정의하고 클래스의 모든 메소드들이 그 도메인을 참조하도록 정의된 경우이다[12]. 클래스의 멤버들이 그들간에 최대 연결을 가질 때, 클래스는 가장 응집적이다. 즉, 클래스 내의 모든 메소드가 클래스 내의 모든 인스턴스 변수와 연결될 때 클래스는 가장 응집적이다[14]. 응집적인 클래스의 모든 메소드들은 대부분의 인스턴스 변수들을 사용한다[1].

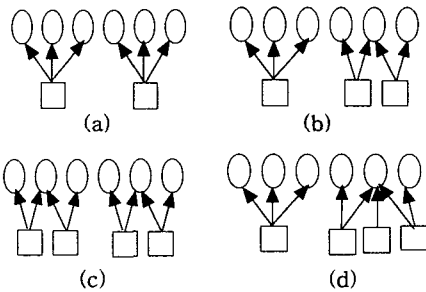
클래스의 응집도 정의 및 등급의 결정이 주관적이고 클래스의 의미론적 분석에 기초하므로, 위의 정의들은 특정 상황에서 주어진 응집도 등급의 근사치를 측정하는 구문론적 기반 척도를 유도하는 지침서로 사용되어야 한다[13].

따라서 본 연구에서는 위의 정의들을 응집도 척도의 분석을 위한 지침서로 이용한다.

3. 응집도 척도들의 분석

• LCOM[3]

LCOM은 Embely의 정의와 그림 1의 메소드의 추가에 대해서 모순된 결과를 보여준다.



(그림 1) 참조 그래프들

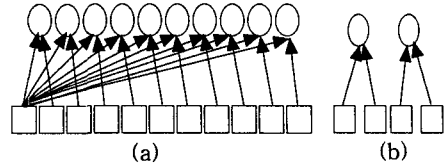
따라서 LCOM은 두 클래스의 구조적 응집성을 구분할 수 없다, 즉 메소드들이 인스턴스 변수들을 공유하는 방법을 구분할 수 없다[5].

• Hitz and Montazeri[4, 5]의 Connectivity Metric

Hitz 등은 연결 매트릭(connectivity metric)이 LCOM의 단점을 극복했다고 주장한다[5]. 그러나 연결 매트릭이 LCOM과 결합하여 사용되므로, LCOM의 최대값(bound)를 결정하는 문제를 해결하지 못했다[1].

• Chen과 Lu[6]의 metric

Chen과 Lu의 매트릭은 파라미터 집합들 간의 관계성의 정도를 구분하지 못하는 문제가 있다[1].

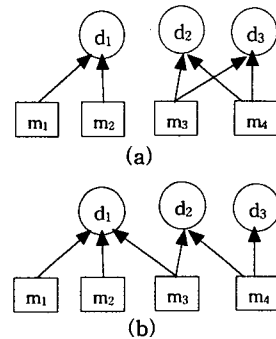


(그림 2) 참조 그래프들

직관적으로 그림 2(b)가 그림 2(a) 보다 최대 연결에 가까우므로 그림 2(b)가 그림 2(a)보다 더 응집적이어야 하지만 Chen과 Lu의 매트릭에 의하면 그림 2(a)가 그림 2(b)보다 더 응집적이다.

• Baxter, Chu와 Patel[7]의 Composite Cohesion measure

그림 3은 Baxter 등의 합성 응집도의 문제점을 보여준다. 직관적으로 그림 3(b)가 그림 3(a)보다 더 응집적이어야 하지만 Baxter의 매트릭에 의하면 그림 3(a)가 그림 3(b)보다 더 응집적이다.

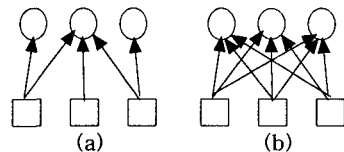


(그림 3) 참조 그래프

이것은 좋은 프로그래밍에 대한 우리의 이해와 응집도와 이 척도간에는 모순이 존재한다[1].

• Bieman과 Kang[8]의 매트릭들

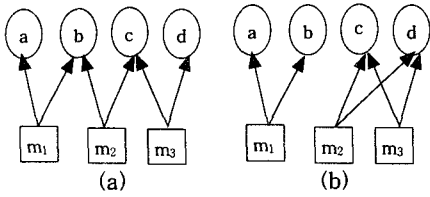
많은 메소드들을 갖는 클래스들은 항상 낮은 응집도를 갖는다. 이것은 이 클래스들의 응집성을 항상 정확히 반영하는 것이 아니다[1]. 그림 4는 또다른 문제점을 보여주는데 그림 4(b)가 그림 4(a)보다 더 응집적이어야 하지만, TCC와 LCC로는 두 가지의 경우를 구분할 수 없다. 그림 4(a)와 4(b)에 대해 $TCC=LCC=1$ 이다.



(그림 4) 참조 그래프들

• Briand와 Morasca[10,11]의 RCI와 Park등[9]의 CC

그림 5는 RCI와 CC의 문제점을 보여준다. 두 개의 클래스는 상이한 연결패턴을 갖게 됨으로 두 클래스의 응집도는 달라야 하며 그림 5(a)가 그림 5(b)보다 더 응집적이어야 하지만, 두 클래스에 대한 RCI와 CC값은 동일하다.



(그림 5) 참조 그래프들

RCI와 CC가 클래스 멤버들간의 연결패턴이 아니라, 연결 개수만을 고려하므로 하므로 이러한 문제가 발생한다[14].

• Otto와 Mehra[1]의 매트릭들

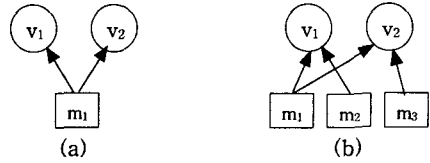
C1과 C2의 SDC=WDC=DA=0이 되므로 이것은 응집도에 대한 우리의 직관과 일치하지 않는다.

v1	v2	Class C1
2	2	<pre>class C1 { int v1, v2; public: m1() { v1 = 0; v2 = 1; }; }</pre>

(그림 6) 클래스 C1의 매트릭 데이터 슬라이스

v1	v2	Class C2
2	2	<pre>class C2 { int v1, v2; public: m1() { v1 = 0; v2 = 1; }; int m2() {return v1;} int m3() {return v2;} };</pre>
1	1	

(그림 7) 클래스 C2의 매트릭 데이터 슬라이스



(그림 8) 그림 6과 7에 대한 참조 그래프

그림 8(a)와 8(b)는 구분되어야 하지만, SDC, WDC, DA는 이것을 구분하지 못한다.

4. 클래스 응집도 척도를 위한 제안

클래스의 응집도를 측정하기 위한 연구들은 클래스의 응집도는 클래스 내의 데이터 요소들간의 상호작용의 함수 또는 클래스에 의해 데이터 요소들이 사용되어지는 방법이라고 가정하고, 주로 데이터 요소들 간의 상호작용을 증대하는 메소드들의 연결성의 정도에 의해 응집도를 측정하였다.

이것이 응집도에 대한 우리의 이해와 측정값간의 모순을 발생시키는 주요 이유가 된다.

따라서 메소드와 도메인의 참조관계에 근거한 클래스 응집도 척도는 메소드들간의 참조 관계뿐만 아니라 또 다른 요인들을 고려하여야 한다.

첫째, 도메인의 개수이다.

둘째, 메소드의 개수이다.

셋째, 메소드와 도메인간의 참조 관계의 개수이다.

넷째, 메소드와 도메인간의 간접 참조 관계이다.

다섯째, 도메인 참조 그래프에서 서로소인 부분 그래프이다. 도메인 참조 그래프에서 서로소인 부분 그래프를 갖는 클래스는 분리가능 응집도를 갖는다. 이러한 클래스는 더 높은 응집도를 갖도록 분리가능한 부분들로 분할할 수 있다. 그러므로, 도메인의 개수와 메소드의 개수가 비슷하고, 유사한 형태의 참조관계가 존재하는 상황에서 서로 소인 부분 그래프를 갖는 클래스는 하나의 연결 그래프로 구성된 클래스보다 낮은 응집도를 가져야한다.

5. 결론

본 연구에서는 기존의 척도들의 분석에 근거하여, 클래스의 응집도를 측정하기 위해서 응집도 척도가 고려해야 할 사항 즉, 도메인의 개수, 메소드의 개수, 메소드와 도메인간의 참조 관계의 개수, 메소드와 도메인간의 간접 참조 관계, 도메인 참조 그래프에서 서로소인 부분 그래프 등을 제시하였다. 이러한 요인들을 고려한 응집도 척도는 기존 척도들의 문제점들을 해결하고 응집도에 관한 우리의 직관과 더욱 일치하는 측정을 할 수 있을 것이다.

참고문헌

- [1] Bindu Mehra, "A Critique of Cohesion Measures in the Object-Oriented Paradigm", *Masters Thesis*, Department of Computer Science, Michigan Technological university, 1997.
- [2] L. M. Otto et al., "Developing Measures of Class Cohesion for Object-Oriented Software.", *7th Annual Oregon Workshop on Software Metrics*, 1995.
- [3] Shyam R. Chidamber, Chris F. Kemerer, "Towards a Metrics Suite for Object-Oriented Design," *In Proc. OOPSLA' 91*, ACM, pp.197-211, 1991.
- [4] Martin Hitz, Behzad Montazeri, "Measuring coupling and cohesion in object-oriented systems", *Proc. of the International Symposium of Applied Corporate Computing (ISACC' 95)*, Oct. 1995.
- [5] Martin Hitz, Behzad Montazeri, Chidamber and Kemerer's Metrics Suite: A Measurement Theory Perspective", *IEEE Transaction On Software Engineering*, vol. 20, no. 6, pp.267-271, 1996.
- [6] J.Y.Chen, J.F.lu, "A new metric for object-oriented design", *Information and Software Technology*, pp.232-240, 1993.
- [7] Rich Baxter, William Chu, Sukesh Patel, "A measure for composite model cohesion", *14th ICSE*, pp.38-48, 1992.
- [8] James M. Bieman, Byung-Kyoo Kang, "Cohesion and reuse in an object-oriented paradigm", *Proc. ACM Symposium on Software Reusability (SSR-95)*, pp.259-262, 1995.
- [9] Sunghee Park et al., "Metrics Measuring Cohesion and Coupling in Object-Oriented Programs", *Journal of Korean Information Science Society*, vol.25, no.12, pp.1779-1787, 1998.
- [10] Lionel C. Briand, S. Morasca, V.R. Basili, "Defining and Validating High-Level Design Metrics", *Technical Report CS-TR-3301-1*, University of Maryland, Dept. of Computer Science, College Park, Md., 1994
- [11] Lionel C. Briand, S. Morasca, V.R. Basili, "Defining and Validating High-Level Design Metrics", *IEEE Transactions on Software Engineering*, vol. 25, no.5, pp.722-743, 1999.
- [12] Embley, D.W.and Woodfield, S.N., "Accessing the quality of abstract data types written in Ada.", *Proc. of Pheonix Conf. on Computers & Comm.*, pp.205-213, 1987.
- [13] Lionel C. Briand, John W. Daly, and Juergen Wuest, "A Unified Framework for Cohesion Measurement", *Proceedings of the 4th International Software Metrics Symposium (METRICS '97)*, 1997
- [14] HeungSeok Chae, YongRae Kwon, "A Cohesion Measure for classes in Object-Oriented Systems", *Proceedings of the 5th International Symposium on Software Metrics*, 1998