객체지향 패러다임에서 응집도 측정

최완규
광주대학교 컴퓨터전자통신공학부
wkchoi@kwangju.ac.kr

Measuring Cohesion in the Objec-oriented Paradigm

Wan-Gyu Choi
Div. of Computer, Electronics&Communication, Kwangju Univ.

요 약

클래스의 응집도를 측정하기 위한 여러 연구들이 제안되었지만, 이런 연구들은 데이터 상호작용에 의해 응집도를 측정하므로 메소들간에 데이터 상호작용이 없지만 객체의 또 다른 속성들을인 데이터들이 함께 속하는 경우를 고려하지 못하고 있다. 따라서 본 연구에서는 데이터 상호작용이 없는 경우를 고려하고, 또한 클래스 내의 멤버들과 멤버들간의 연결을 모두 고려하여 응집도를 측정할 수 있는 새로운 응집도 척도를인 강 클래스 응집도(Strong Class Cohesion: SCC)와 약 클래스 응집도(Weak Class Cohesion: WCC)를 제안하였다. 또한 기존 척도들과의 비교평가를 통해서 WCC와 SCC가 향상된 측정을 제시함을 보여주었다.

1. 서 론

응집도는 모듈 구성 요소들의 연관성이라고 하는 소프트웨어의 속성이다. 소프트웨어의 응집도가 높을수록 소프트웨어는 이해하고 유지보수하기가 쉽다[1].

클래스에서의 응집도를 측정하기 위한 여러 연구들[2, 3, 4, 5, 6, 7, 8, 9, 10]이 제안되었다. 이런 연구들은 데이터 상호작용에 의해 응집도를 측정하는 것에 근거하고 있다. 그러나, 그들은 메소들간에 데이터 상호작용이 없지만 객체의 쪼 다른 속성들을인 데이터들이 함께 속하는 경우를 고려하지 못하고 있다[2]. 또한 기존의 척도들은 메소드들간의 연결성이나 인스턴스 변수와 메소드의 참조관계에 의해서만 응집도를 측정한다[10].

따라서 클래스의 응집도를 측정하기 위해서는 클래스의 멤버들인 인스턴스 변수와 메소드, 클래스 내의 멤버들간의 연결을 이루는 메소드와 인스턴스변수간의 참조관계, 메소드간의 호출을 모두 고려해야한다. 또한 데이터 상호작용이 없는 경우를 고려하여 클래스의 응집도 측정이 이루어져야한다.

따라서 본 연구에서는 데이터 상호작용이 없는 경우를 고려하고, 또한 클래스 내의 멤버들과 멤버들간의 연결을 모두 고려하여 응집도를 측정할 수 있는 새로운 응집도 척도를인 강 클래스 응집도(Strong Class Cohesion: SCC)와 약 클래스 응집도(Weak Class Cohesion: WCC)를 제안한다.

응집도 측정이 의미 있는 측정을 제공하기 위해서는 척도가 엄밀하게 정의되어야하고, 측정을 위한 속성을 정확히 반영해야하고, 이런 속성들을 포착하는 모델에 근거해야한다[10]. 본 연구에서는 클래스 내의 멤버와 멤버들간의 연결을 식별하기 위해 참조 관계 그래프를 이용한다.

2. 클래스 응집도 척도

클래스의 응집도를 측정하기 위한 척도들은 데이터 상호작용이 없는 경우를 반영해야하고, 또한 데이터 연결을 고려하기 위해서 다음과 같은 요인들을 고려해야 한다[11].

첫째는 인스턴스 변수의 개수이다. 둘째는 메소드의 개수이다. 셋째, 메소드와 인스턴스변수간의 직접 참조 관계의 개수이다. 넷째, 메소드와 인스턴스변수간의 간접 참조 관계이다. 다섯째, 참조 그래프에서 서로소인 부분 그래프이다.

본 연구에서는 새로운 응집도 척도를 정의하기 위해서 참조관계 그래프를 이용한다.

$D=\{d_1, d_2, \cdots, d_n\}$가 클래스 C 내에서 정의된 데이터 요소들인 인스턴스 변수들의 집합이고, $M=\{m_1, m_2, \cdots, m_k\}$이 클래스 C 내에서 정의된 메소드들의 집합이라 하자. $m_i(m_i \in M)$가 $d_i(d_i \in D)$를 m내에서 사용하면 m_i가 d_i를 직접 참조한다고 하고, $m_i \rightarrow d_i$으로 나타낸다. $m_i(m_i \in M)$가 $m_j(m_j \in M)$를 직접 호출(call)하고 $m_j(m \in M)$가 $d_i(d \in D)$를 m_j내에서 사용하고, m_i가 d_i를 직접 참조하지 않으면, m_i가 d_i를 간접 참조한다고 하고, $m_i \dashrightarrow d_i$으로 나타낸다. 클래스 C의 참조 관계 그래프 G(V, E)는 방향성을 가지는 그래프로서, $V=(D \cup M)$이고, $E = \{(d, m) \mid m \rightarrow d \text{ or } m \dashrightarrow d,\ d \in D,\ m \in M\}$이다.

본 연구에서는 위에서 정의한 5가지 요인을 고려하여 응집도를 측정할 수 있는 응집도 척도들을 다음과 같이 정의한다. I_m는 메소드 m(m∈M)에 의해 직·간접적으로 참조되는 인스턴스 변수들의 집합이라고 하자.

그러면, $I = \bigcup_{i=1}^{|M|} I_{m_i}$, if $I_{m_j} \in M$

$M^* = \{m \mid |I_m| \neq 1,\ m \in M\}$,

$I^* = \bigcup_{i=1}^{|M^*|} I_{m_i}$, if $m_i \in M^*$

$D^* = (D - I) \cup I^*$

E^*={(d, m) | m→d or m⇢d, d∈D^*, m∈M^*}이다.

이때, 응집도 척도들은 다음과 같이 정의된다.

정의 1: *SCC와 WCC*
참조 관계 그래프 G로 표현된 임의의 클래스 C에 대해서, SCC와 WCC는 다음과 같다.

$$If\ |D|=0\ or\ |M|=0,\quad SCC = WCC = 0$$

$$otherwise\quad SCC = 1 - \frac{NI \times NM - NDR}{NI \times NM \times \sqrt{NDS}} \qquad 식(1)$$

$$WCC = 1 - \frac{NI \times NM - (NDR + NIR)}{NI \times NM \times \sqrt{NDS}}$$

여기서
NI: Cardinality of D^*
NM: Cardinality of M^*
NDR: Number of Direct References in E^*
NIR: Number of indirect References in E^*
NDS: Number of Disjoint Sets in G

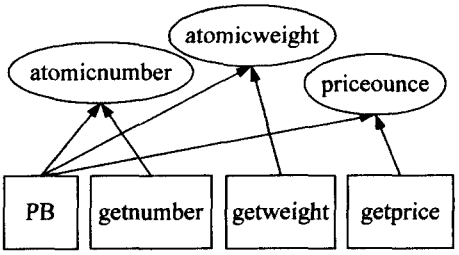
3. 기존척도와의 비교

대부분의 기존의 척도들은 데이터 상호작용(data interaction)에 근거하고 있다. 따라서 데이터 상호작용이 없는 클래스에 대해서 기존의 척도들은 항상 낮은 응집도를 갖는 것으로 측정한다.
예를 들어서 그림 1의 PB 클래스[1]의 예를 보자

```
class PB{
private:
 unsigned atomicnumber;
 float atomicweight, priceperounce;
public:
 PB() {atomicnumber = 82;
   atomicweight=207.2; priceperounce=0.01;}
 unsigned getnumber(){
   return atomicnumber;}
 float getweight() {
   return atomicweight;]
 float getprice() {
   return priceperounce;}
};
```

(그림 1) PB 클래스

그림 2는 PB 클래스에 관한 참조 관계 그래프를 보여준다.



(그림 2) PB 클래스에 대한 참조 관계 그래프

PB 클래스는 같이 속하는 데이터와 메소드를 갖는 분명히 응집적인 클래스이다[1]. 그러나, 표 1에서와 같이 기존의 척도들은 PB 클래스가 낮은 응집도를 갖는 클래스로 측정한다.

<표 1> PB 클래스에 대한 기존 척도들의 측정값들

metrics	use constructor	no use constructor
LCOM[3]	0	3
Hit et al.[4]	0.3	0
Chen&Lu[5]	25	100
Baxter et.al[6]	0.539	0.333
TCC[7]	0.5	0
LCC[7]	1.0	0
SDC[2]	0	0
WDC[2]	0	0
A[2]	0	0
CC[8]	16	0
RCI[9]	0.5	0.33
CCM[10]	0.5	0.20

기존의 대부분의 척도들이 PB 클래스가 매우 낮은 응집도를 갖는 클래스라고 하는 것은 그들이 데이터 상호작용이 없는 메소드들을 클래스 응집도 측정에 포함하기 때문이다. 이에 비해서 본 연구에서 제안한 SCC와 WCC는 데이터 상호작용이 없는 메소드들을 클래스 응집도 측정에 포함하지 않기 때문에 PB 클래스에 대해서 매우 응집적인 클래스로 측정한다.
다음은 PB 클래스에대한 SCC와 WCC의 측정의 예를 보여준다.

D = {atomicnumber, atomicweight, priceperounce}
M = {PB, getnumber, getweight, getprice}
I_{PB}={atomicnumber, atomicweight, priceperounce}
$I_{getnumber}$ = {atomicnumber}
$I_{getweight}$ = {atomicweight}
$I_{getprice}$ ={priceperounce}
I = {atomicnumber, atomicweight, priceperounce}
M^* = {PB}
I^* = {atomicnumber, atomicweight, priceperounce}
D^* = {atomicnumber, atomicweight, priceperounce}

metrics	use constructor	no use constructor
SCC	1	1
WCC	1	1

<표 2> PB 클래스에 대한 SCC와 WCC의 측정값

표 2에서 SCC와 WCC는 PB 클래스가 매우 응집적인 클래스임을 나타낸다.

4. 결 론

본 연구에서는 클래스의 멤버들인 인스턴스 변수와 메소드, 클래스 내의 멤버들간의 연결을 이루는 메소드와 인스턴스변수간의 참조 관계, 메소드간의 호출을 모두 고려해고, 또한 데이터 상호작용이 없는 경우를 고려하여 클래스의 응집도 측정할 수 있는 척도들인 SCC와 WCC를 제안하였다.

또한, 기존 척도들과의 비교를 통해서 SCC와 WCC가 데이터 상호작용이 없는 메소드를 측정에서 제외시킴으로써, 기존 척도들에 비해 나은 측정을 제공함을 보여주었다.

참 고 문 헌

[1] L. M. Otto et al., "Developing Measures of Class Cohesion for Object-Oriented Software.", 7th Annual Oregon Workshop on Software Metrics, 1995.

[2] Bindu Mehra, "A Critique of Cohesion Measures in the Object-Oriented Paradigm", Masters Thesis, Department of Computer Science, Michigan Technological university, 1997.

[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, Chidamber and Kemerer's Metrics Suite: A Measurement Theory Perspective", IEEE Transaction On Software Engineering, vol. 20, no. 6, pp.267-271, 1996.

[5] J.Y.Chen, J.F.lu, "A new metric for object-oriented design", Information and Software Technology, pp.232-240, 1993.

[6] Rich Baxter, William Chu, Sukesh Patel, "A measure for composite model cohesion", 14th ICSE, pp.38-48, 1992.

[7] 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.

[8] 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.

[9] 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.

[10] 김성애, 최완규, 이성주, "객체지향 패러다임에서 저해요인에 기반한 응집도 척도", 정보처리논문지, vol.7, no.11, 2000.

[11] 최완규, 김성애, 이성주, "객체지향 패러다임을 위한 응집도 척도들에 관한 연구", 한국정보처리학회 소프트웨어공학연구회지, vol.3, no.3, 2000.