

러프집합과 퍼지집합에 기반한 기능중심 컴포넌트의 재사용도 측정

Measuring Reusability of the Function-Oriented Component Based on Rough and Fuzzy Sets

박병권 · 김혜경 · 최완규 · 이성주

Byung-Gweun Bak, Hye-Kyoung Kim, Wan-Kyoo Chio and Sung-Joo Lee

조선대학교 전자계산학과

요 약

사용자가 최소의 이해와 수정 노력으로 적합한 컴포넌트를 선택할 수 있는 방안이 요구된다. 따라서 본 논문에서는 컴포넌트의 재사용도 측정을 위한 혼합적 척도를 제안한다. 현업에서의 연구와 경험을 통해서 증명된 객관성 있는 척도들을 측정인자로 설정한다. 러프집합을 이용하여 각 측정인자들이 컴포넌트 재사용에 미치는 영향의 정도를 평가하고, 각 측정인자들의 상대적 중요도를 구한다. Sugeno의 퍼지적분을 이용하여 측정인자들의 중요도와 측정값들을 종합함으로써, 컴포넌트들의 재사용도를 측정한다. 마지막으로 제안된 척도가 ordinal scale과 ratio scale에 따름을 보여준다.

ABSTRACT

A method that users can select necessary components and evaluate similar components by the minimum effort is needed. Therefore, in this paper, we propose a hybrid metric for measuring the component reusability. We use metrics proven by many researches and experiments in industry studies as measuring factors. We estimate how much factors affect the component reusability by using rough sets and calculate the relative significance of factors. We evaluate the component reusability by integrating the calculated significances and the measured values of factors using Sugeno's fuzzy integral. Lastly, we show that the proposed metric obeys the ordinal scale and the ratio scale.

1. 서 론

소프트웨어 재사용에서, 재사용 정보들은 코드 단위(code fragment), 논리적 구조(logical information), 기능적 구조(functional architecture), 영역 지식(external knowledge), 소프트웨어 사용과 기술 이전에 관한 지식(environment-level knowledge)등이다[12]. 특히 소스 코드는 프로그래머의 일차적인 산물이며, 소스 코드는 소프트웨어 재사용의 방법 중 가장 오래된 방법이고, 기 개발된 프로그램에서 재사용 가능한 코드 단위를 추출하거나 프로그램 개발 시 재사용을 고려한 코드를 개발하여 컴포넌트 라이브러리에 저장한 후, 이를 재사용하려는 많은 연구들이 진행되어 왔다[12, 19].

컴포넌트 재사용은 검색 단계, 이해 단계, 적용 단계로 이루어진다[12]. 목적과 일치되는 컴포넌트를 찾는 검색 단계에서 적합한 컴포넌트가 발견되면 그것을 그대로 사용하고, 기능적으로 유사한 컴포넌트이면 일부 수정한 후 사용하고, 유사한 컴포넌트가 없을 때

는 새로운 컴포넌트를 개발한다. 이해 단계에서는 검색단계에서 찾은 컴포넌트들 중 목적에 가장 근접한 컴포넌트를 선정하여 이해하고 목적에 맞게 수정한다. 적용 단계에서는 수정된 컴포넌트를 합성하여 새로 개발할 소프트웨어를 구현한다.

현실적으로 검색 단계에서 발견된 컴포넌트들은 목적과 일치하는 컴포넌트가 아닌 세부적인 구현 내용만 다르고 기능적으로 유사한 컴포넌트들일 수 있다. 이런 경우에 컴포넌트들의 단순한 나열은 기존 컴포넌트에 대한 이해와 적용을 어렵게 함으로, 사용자는 기존 컴포넌트의 재사용을 단념할 수 있으며, 특히 유사 컴포넌트들의 종류가 많을수록 더욱 그렇다[3]. 그러므로 사용자가 최소의 이해와 수정 노력으로 적합한 컴포넌트를 선택할 수 있는 방안이 요구된다.

따라서 본 논문에서는 컴포넌트의 재사용도 측정을 위한 혼합적 척도를 제안한다. 먼저 현업에서의 연구와 경험을 통해서 증명된 객관성 있는 척도들을 측정인자로 설정하였다. 러프집합을 이용하여 각 측정인자

본 논문은 1998년도 조선대학교 학술연구비에 의해 연구되었음.

들이 컴포넌트 재사용에 미치는 영향의 정도를 평가하여 각 측정인자들의 중요도를 구한다. 다음으로 중요도를 가중치로 하고, Sugeno의 퍼지적분을 이용하여 각 측정 인자들의 가중치와 측정값들을 종합함으로써, 컴포넌트들의 재사용도를 측정한다.

본 논문의 구성은 2장에서 러프집합과 퍼지 척도에 대하여 간략하게 기술하고, 3장에서 컴포넌트 재사용도 측정을 위한 혼합적 척도를 제시하고, 통계적 방법을 통한 척도의 검증 방법을 설명하며, 4장에서 재사용도 측정을 보여주고, 척도의 타당성을 검증하며, 5장에서 결론 및 향후 연구과제에 대해서 논의한다.

2. 러프 집합 과 퍼지 척도

2.1 러프집합

1982년 Pawlak에 의해 제안된 러프 집합 이론은 애매한 범주들에 대한 새로운 수학적 접근 방법이다[10]. 러프 집합은 자동 분류, 패턴 인식, 학습 알고리즘 등 다양한 분야에 적용될 수 있다[11, 20].

그림 1의 근사공간 A는 순서쌍 $A=(U, R)$ 이다. 여기서 U는 전체집합이고 R은 식별불가능(동치)관계이다. $(x_1, x_2) \in R$ 이면, x_1 과 x_2 는 A에서 식별불가능이다. 관계 R의 동치류들은 원자집합이라 한다. A에서 원자집합들의 합집합은 A에서 정의가능이라 한다.

그림 1에서,

$$U/R = \{E_1, E_2, \dots, E_{15}, E_{16}\}$$

$$\text{원자집합들} : E_1, E_2, \dots, E_{15}, E_{16}$$

X의 상한근사:

$$\begin{aligned} R^*(X) &= \cup \{E_i \in U/R : E_i \cap X \neq \emptyset\} \\ &= \cup \{E_1, E_2, \dots, E_{11}, E_{12}\} \end{aligned}$$

X의 하한근사:

$$\begin{aligned} R_*(X) &= \cup \{E_i \in U/R : E_i \subseteq X\} \\ &= \cup \{E_6, E_7\} \end{aligned}$$

2.2 퍼지척도

퍼지집합 이론은 현실의 불확실한 상황을 그대로 표현해주는 방법으로서, 1965년 Zadeh에 의해 제안되었다. 퍼지집합은 퍼지 경계를 가지는 집합으로, 집합

내의 소속 등급을 클래스에 있는 모든 객체에 연관시킨다. 퍼지함수는 퍼지집합 내의 퍼지집합의 값을 $[0,1]$ 구간에서 사상시킨다[21].

일반적인 퍼지 집합 $A = \{(a, 0.5), (b, 0.4), (c, 1)\}$ 가 있을 때, 이 집합은 전체 집합 X 내의 원소 a, b, c가 집합 A에 소속될 가능성이 각각 0.5, 0.4, 1임을 나타낸다. 따라서 퍼지집합 A의 소속함수 μ_A 는 전체 집합 X에 퍼지값을 부여한다, 즉

$$\mu_A : X \rightarrow [0, 1]$$

이렇게 하여 얻은 원소 x의 퍼지 값 $\mu_A(x)$ 는 x가 A에 소속될 정도를 나타내며, 이때 함수 μ_A 는 집합이고, 1개씩 정의된다.

전체 집합 X 내의 여러 개의 부분집합 A_1, A_2, \dots 등을 생각해 보자. 이 집합들은 멱집합 $P(X)$ 의 부분집합이 된다. 이때 어느 원소 a가 있다고 하면, 원소 a가 집합 A_1, A_2, \dots 등에 속할 가능성은 어떻게 될까? 이 가능성 역시 주관적인 판단에 의해서 결정될 수 있고, 이 가능성도 애매한 상태를 나타내기 위해서 퍼지값으로 표현될 수 있다. 이와 같이 어느 한 원소 a가 집합 A_1, A_2, \dots 등에 소속될 가능성을 나타낼 때 사용되는 척도를 퍼지척도(fuzzy measure)라 한다. 즉 퍼지 척도 g는 다음과 같이 $P(X)$ 의 원소에 $[0, 1]$ 의 값을 부여한다.

$$g : P(X) \rightarrow [0, 1]$$

퍼지집합은 집합 A가 있을 때, 이 집합에 소속되는 원소들을 정의한다. 이때 집합 A의 경계(boundary)가 모호하게 정의되기 때문에 집합 A에 정의되는 소속함수 μ_A 의 값이 퍼지값을 가지게 된다. 즉, 이때는 집합 A에 주어진 불확실성을 표현한다.

퍼지척도는 임의의 원소 a가 있을 때, 이 원소가 어느 집합에 소속될 수 있는가 하는 믿음의 정도(degree of evidence)를 나타낸다. 이때도 a가 임의의 집합에 속할 가능성이 불확실하므로 퍼지척도 g가 퍼지 값을 가진다. 퍼지척도 g는 원소마다 정의된다.

특히 퍼지척도를 이용한 Sugeno의 퍼지 적분은 특정 대상이 여러 항목에 대해서 평가가 되고 각 평가 항목의 중요도가 차이가 있을 때 이들 평가 값들을 종합하는데 유용하게 이용될 수 있다[21].

3. 컴포넌트 재사용도

3.1 재사용도 측정 인자들

재사용을 위한 양질의 컴포넌트(component)는 정확성, 효율성, 유지보수성, 신뢰성 등이 있어야 한다. 재

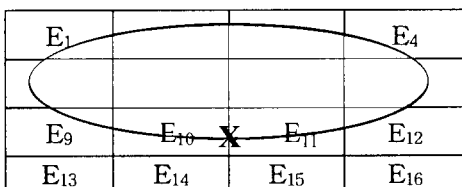


그림 1. 근사공간 A

사용을 위한 컴포넌트의 평가기준은 소규모, 간편한 구조, 집약된 문서화, 동일 언어 등이다. 그러나 컴포넌트의 구축 방법인 기능 중심(function-oriented)과 객체 지향(object-oriented)에 따라 품질 평가요인이 다를 수 있다[3, 19].

기능 중심으로 구성된 모듈의 품질 측정은 규모(size)와 난이도(difficulty)의 매트릭스를 수집한 정적 분석기에 의해 쉽게 측정될 수 있으며, 기존의 소프트웨어 모듈의 품질 평가 방법들인 McCabe의 Cyclomatic Number, Halstead의 Software Science, Lines of Code 등을 도입해도 가능하다[10, 19].

따라서, 본 연구에서는 현업에서의 연구와 실험을 통해서 품질 분류 기준들이 제시된 대표적인 규모와 논리 구조 척도들인 LOC(lines of code), McCabe의 Cyclomatic Number, Halstead의 Volume, Difficulty, Effort를 측정 인자로 설정하였다.

여러 연구들은 실험적인 연구들 통해서 LOC, McCabe의 Cyclomatic Number와 Halstead의 Software Science에 대한 모듈 분류 기준들을 표 1, 2, 3, 4와 같이 제시하였다[3, 5, 7, 8, 15].

실험적인 연구들은 코드의 라인수(LOC)는 50-100

표 1. 소프트웨어 척도의 상한 값[15]

Measure Limits	Values
Cyclomatic Number	≤ 15
Normalized Cyclomatic Number	≤ 0.4
Halstead's Effort	≤ 300000

표 2. 척도들의 상한 값[7]

Measure	Safe	Flag	Alarm
LOC	<50	50~100	>100
Effort	<50000	50000~100000	>105000
Cyclomatic Number	<10	11~20	>20

표 3. 척도들의 수용가능 범위[8]

Measure	Low	Medium	High
ELOC	100	500	>1000
Effort	10,000	100,000	>100,000
Difficulty	<10	<25	>50
Cyclomatic Number	10	50	>100

표 4. 척도들의 수용가능 범위[3]

Measure	Minimum	Maximum
Halstead's Volume	2000	10000
Cyclomatic Complexity	5	15
Regularity	0.7	1.30
Reuse Frequency	10	50

표 5. 컴포넌트 분류 기준

Domain \ Measure	0	1	2
LOC	(0, 30]	(30, 50]	(50, 100]
Cyclomatic Number	(0, 10]	(10, 20]	(20, 50]
Volume	(0, 1000]	(1000, 4000]	(4000, 10000]
Difficulty	(0, 10]	(10, 25]	(25, 100]
Effort	(0, 10000]	(1000, 100000]	(10000, 300000]

범위, 복잡도(Cyclomatic Number)는 10 이하, 난이도(Difficulty)는 10 이하, 프로그램 노력도(Effort)는 1,000 이하가 적합하고, 볼륨(Volume)이 크면 컴포넌트의 이해와 적용을 어렵게 하므로, 볼륨은 평균적으로 200-1,000의 범위가 적합하다고 제시하였다[8].

이러한 분류 기준들은 연구와 경험을 통해서 증명되어 왔지만, 다소 임의적이며, 실제로 각 현장에서의 요구들은 변할 수 있다[8].

본 논문에서는 기존의 연구에서 제시된 기준들을 종합하여 컴포넌트 분류 기준을 표 5와 같이 설정하고, 통계적 실험을 통해 분류 기준의 유용성을 검증한다.

기능 중심으로 구성된 모듈의 품질 측정에서, 각 품질 평가 척도들에 대한 통합된 하나의 척도가 제시되려면, 각 방법들이 척도에 미치는 영향도 및 각 방법들을 결합시킬 방법이 고려되어야 한다. 이를 위해서 소속등급(membership grade)과 관련되어 특정 지을 수 있는 퍼지집합을 이용할 수 있다[19].

따라서, 본 논문에서는 각 측정 인자들의 측정값을 [0, 1] 구간으로 정규화하기 위한 퍼지 소속함수를 식 (1)과 같은 S자형 함수로 정의한다. 각 측정 인자들을 정규화하기 위한 x_0 와 x_1 의 값들은 표 5에서 제시된 값들 중 최소값과 최대값이다. 예를 들어서 LOC에 대해서 $x_0=0$ 이고 $x_1=100$ 이다.

$$\mu(x) = \begin{cases} 1 & , x \leq x_0 \\ 1 - 2 \left(\frac{x - x_0}{x_1 - x_0} \right)^2 & , x_0 < x \leq \frac{x_0 + x_1}{2} \\ 2 \left(\frac{x_1 - x}{x_1 - x_0} \right)^2 & , \frac{x_0 + x_1}{2} < x \leq x_1 \\ 0 & , x > x_1 \end{cases} \quad (1)$$

3.2 척도의 중요도

컴포넌트의 재사용도 평가에 적용될 수 있는 모든 품질 특성(측정 인자)들은 컴포넌트 재사용도 평가에 동일한 영향을 미치지 않는다. 따라서 객관적으로 적절한 가중치를 부여할 수 있는 방법이 사용되어야 한다[1]. 측정 인자들을 적용하여 결과를 얻기 위해서

각 인자들의 중요도를 변화시켜 적용해야 할 필요가 있다. 측정인자의 중요도는 지식을 기초로 하여 미리 전제되어 가중치에 의해 표현되지만, 경험적 지식이 없는 상황에서 가중치의 설정은 매우 어렵다.

그러나 러프집합의 지식 표현시스템은 각 속성(측정인자)들이 갖는 강도(strength)와 분류 능력(classification power)의 차이의 정도를 쉽게 산출한다.

러프집합에서는 속성의 중요도를 발견하기 위해서 주어진 지식 표현 시스템에서 속성을 삭제하고, 삭제된 속성이 없이 분류들이 어떻게 변하는가를 조사한다. 특정 속성의 제거가 분류를 상당히 변화시킨다면, 그 속성은 지식표현 시스템에서 중요도가 높은 속성이다[11].

러프 집합에서 속성의 중요도는 다음과 같이 구한다.

지식 표현 시스템에서 조건 속성(C)의 모든 원소 범주들이 의사결정 속성(D)의 원소 범주들로 정의될 수 있으면 지식 D가 지식 C의 모든 특성을 제공하므로 지식 C는 지식 베이스에서 필요 없다. 이것은 지식간의 종속성으로 표현되며 식은 다음과 같다.

$$k = \gamma_C(D) = \frac{|POS_C(D)|}{|U|} \quad (2)$$

$POS_C(D)$: D의 C-positive region
 $|POS_C(D)|$: $POS_C(D)$ 의 원소의 개수
 $|U|$: 전체 집합의 원소의 개수

지식표현 시스템의 조건 속성들 중 특정 속성들은 의사결정에 있어서 다른 속성들 보다 더 중요할 수 있다. 이것은 속성의 중요도로 표현되며, C의 부분 집합 C'의 중요도는 식(3)과 같다[11, 20].

$$S_C = \gamma(D) - \gamma_{C'}(D) \quad (3)$$

$\gamma(D)$: C의 D에 대한 종속성
 $\gamma_{C'}(D)$: C-C'의 D에 대한 종속성

전체 속성의 중요도에서 특정 속성의 중요도가 점유하는 비율로 표현되는 속성의 상대적 중요도는 식(4)와 같다.

$$RS_i = \frac{S_i}{\sum_{j=1}^n S_j}, (i=0, 1, \dots, n) \quad (4)$$

RS_i : 속성 i의 상대적 중요도
 S_i, S_j : 속성 i, j의 중요도

3.3 재사용도 측정

본 논문에서는 각 척도들이 재사용도에 미치는 영향의 정도를 결합시키기 위해서 Sugeno의 퍼지적분을 이용한다. Sugeno의 퍼지적분은 어떤 대상을 여러 항목에 대해서 평가되고 각 평가 항목의 중요도에 차이가 있을 때, 이 들 각 항목에 대한 평가 값을 퍼지척도를 사용하여 종합하는 방법으로 사용될 수 있으며 [21], 주관적인 판단이 개입되는 평가 문제에서도 유용하게 이용될 수 있다.

집합 X가 유한 집합이고, P(X)는 X의 멱집합이고, 함수 $h(x)$ 는 평가항목 $x(x \in X)$ 에 대한 평가치이고, $E \in P(X)$ 에 대해 정의되는 퍼지척도 $g(E)$ 는 전체적인 평가에 대한 평가항목 E의 평가치에 기여하는 정도라 하자.

이 때 $x_i \in X (i=1, 2, \dots, n)$ 에 대해서 $h(x_i) \leq h(x_{i+1})$ 이고, $E_i = \{x_k \mid k=i, i+1, \dots, n\}$ 이면, 평가함수 h의 상대적 중요도 함수 g에 대한 Sugeno의 퍼지적분은 다음과 같다.

$$\int_X h(x) \quad g(\cdot) = \text{Max}_{i=1}^n \text{Min}[h(x_i), g(E_i)] \quad (5)$$

3.4 척도의 검증

Bowles는 “척도가 측정한다고 주장한 것을 측정한다는 것을 보여줄 수 있을 때 척도는 검증(validation)된다”라고 하였다[5]. Zuse는 “척도의 검증의 문제는 경험적 관계이고 특히 경험적 관계의 의미이다”라고 전제하고, 소프트웨어 척도의 검증을 위해서 척도가 어떤 형태의 스케일(scale)을 전제하는가를 결정하는 방법을 보여준다[5]. Norman[9]은 통계적 분석을 통해서 척도가 전제하고 있는 스케일을 가짐을 보여줄 수 있다고 하였다.

우리는 척도의 스케일 속성을 결정하기 위해서 통계적 방법을 도입하였다. 먼저 분류기준의 적합성을 검증하고, 다음으로 제안된 척도가 측정된 재사용도와 관련된 우리의 직관과 일치하는 ordinal scale을 전제하고 있음을 보여준다. 마지막으로 제안된 척도가 ratio scale을 전제함을 보여준다

F가 재사용의 용이한 정도를 나타내는 척도이고, 측정인자들의 집합을 $M = \{LOC, Cyclomatic\ number, Volume, Difficulty, Effort\}$ 이라 한다.

3.4.1 분류기준의 적합성

척도 M의 분류 기준 값(M_c)들이 다음 식과 같이 벡터 F의 원소들을 구분할 수 있으면, 분류기준 값들은 적합하다고 할 수 있다.

$$M_i > M_c \Leftrightarrow F_i > F_c \quad \text{and} \\ M_i < M_c \Leftrightarrow F_i < F_c \quad (6)$$

식 (6)은 Mc를 Fc의 간접적인 척도로 사용할 수 있을 만큼 Mc가 충분한 구분력(discriminative power)을 가지고 있는 지를 평가한다. 적절한 통계적 방법(카이제곱 분할표)을 사용하여 식 (6)이 성립하는 정도를 보여줄 수 있다[9]. 즉,

$$\chi^2 > \beta_\alpha \quad (7)$$

β_α : α 에 의해 명시된 임계치

3.4.2 재사용도 척도와 ordinal scale

척도 F가 ordinal scale을 전제한다면, 우리는 컴포넌트 A가 컴포넌트 B보다 재사용하기에 더 용이하다고 주장할 수 있다. 척도 F가 ordinal scale을 갖기 위해서는 인자들의 집합 M과 F의 순위 상관관계 계수(rank correlation coefficient) r이 명시된 임계치(specified threshold)을 초과해야 한다[9]. 즉,

$$r > \beta_c \quad (8)$$

β_c : c에 의해 명시된 임계치

순위 상관관계 계수 r은 분산 분석에서의 결정계수 R²에 의해 산출할 수 있다[9]. 즉 $r = \sqrt{R^2}$. 식 (8)은 M의 간접적인 척도로서 F를 사용할 수 있을 만큼 M의 순위(rank)와 F의 순위가 충분히 일치하는 가를 보여준다. 즉,

$$\begin{aligned} \text{순위}[M_1 < M_2 \dots < M_n] &\Leftrightarrow \\ \text{순위}[F_1 < F_2 \dots < F_n] &\quad (9) \end{aligned}$$

순위 관계가 표본 샘플들에 대해서 설명되고, β_c 가 0.7 이상이면 M이 F와 충분히 일치한다고 할 수 있다[9].

3.4.3 재사용도 척도와 ratio scale

척도 F가 ordinal scale을 전제한다면, 우리는 컴포넌트 A가 컴포넌트 B보다 재사용하기에 몇 퍼센트 더 용이하다고 주장할 수 있다. 척도 F가 ratio scale을 갖기 위해서는 M의 함수 f(M)이 β_p 의 정확성으로 F의 값을 예측할 수 있어야 한다. 또는

$$\left| \frac{F_a - F_e}{F_a} \right| < \beta_p \quad (10)$$

F_a : 실제 측정값,
 F_e : 추정값

식 (10)은 M의 간접적인 척도로서 F를 사용할 수 있을 만큼 M의 값과 F의 값 사이에 충분한 상관관계가 있는가를 보여준다. 즉,

$$M_{j+1} > M_i \Leftrightarrow F_{j+1} > F_i$$

$$\begin{aligned} M_{j+1} = M_i &\Leftrightarrow F_{j+1} = F_i \\ M_{j+1} < M_i &\Leftrightarrow F_{j+1} < F_i \end{aligned} \quad (11)$$

척도 M이 ratio scale을 전제하는가는 중회귀분석(multiple linear regression analysis)을 통해서 보여줄 수 있다[9, 16]. 즉

$$\begin{aligned} \text{Test for Predictability} &> \beta_p \\ \beta_p &: p\text{에 의해 명시된 임계치} \end{aligned} \quad (12)$$

4. 재사용도 측정 및 척도의 검증

실험을 위해서 수집된 18,281개의 C 언어로 쓰여진 컴포넌트들 선택하여 Win95 환경에서 C++로 직접 작성한 코드 분석기 및 러프집합 엔진과 통계 분석 프로그램을 이용하였다.

본 실험에서 F가 재사용의 용이한 정도를 나타내는 척도이고, 측정인자들의 집합을 $M = \{LOC, Cyclomatic\ number, Volume, Difficulty, Effort\}$ 이라 한다.

4.1 재사용도 측정

측정 인자들의 집합 M의 상대적 중요도가 (0.23958, 0.24479, 0.18229, 0.19792, 0.13542)일 때, 컴포넌트 A의 측정값이 (56, 8, 926.85, 33.25, 30817.97)이면, 식 (1)을 적용한 A의 소속 등급은 (0.38, 0.94, 0.98, 0.77, 0.97)이다. 예를 들어서, LOC 값 56의 소속 등급은 $2 \left(\frac{56-0}{100-56} \right)^2 = 0.3872$ 이다.

이때, Sugeno의 퍼지적분을 적용하여 A의 재사용도는 다음과 같이 구할 수 있다.

$$\begin{aligned} &h_i(LOC) \wedge g\{LOC, Dif, Cyc, Eff, Vol\} \\ &\vee h_i(Dif) \wedge g\{Dif, Cyc, Eff, Vol\} \\ &\vee h_i(Cyc) \wedge g\{Cyc, Eff, Vol\} \end{aligned}$$

표 6. 측정값

컴포넌트	LOC	Cyc	Vol	Dif	Eff
1	56	8	926.85	33.25	30817.97
2	28	2	589.37	12.54	7390.71
3	46	4	828.17	32.88	27237.72
4	35	7	585.14	50.75	29696.34
5	39	3	887.12	18.08	16046.51
.
.
.
18,277	27	2	215.49	10.63	2292.06
18,278	22	2	449.78	2.82	1269.55
18,279	7	2	74	9	666.07
18,280	9	2	130.79	10	1307.98
18,281	21	3	261.34	18.20	4756.43

표 7. 재사용도 측정을 위한 지식표현 시스템

U	LOC	Cyc	Vol	Dif	Eff	Reuse
1	2	0	0	2	1	1
2	0	0	0	1	0	1
3	1	0	0	2	1	1
4	1	0	0	1	1	1
5	0	0	0	0	0	1
6	1	0	0	1	0	1
7	1	1	0	1	1	1
8	2	0	1	2	2	1
9	2	0	1	2	1	1
10	2	1	1	2	1	1
11	0	0	0	2	0	1
12	2	0	1	1	1	1
13	2	0	0	1	1	1
14	1	0	0	0	0	1
15	2	1	0	2	1	1
16	0	0	0	2	1	1
17	0	0	0	1	1	1
18	1	1	0	2	1	1
19	0	1	0	1	0	1
20	1	0	1	2	1	1
21	1	0	1	1	1	1
22	2	1	0	1	1	1
23	2	1	1	2	2	1
24	2	2	1	2	1	1
25	2	2	1	1	1	1
26	2	0	0	0	0	1
27	1	1	1	2	1	1
28	1	1	1	1	1	1
29	2	0	1	0	0	1
30	1	1	0	0	0	1
31	0	0	1	1	1	1
32	2	1	1	1	1	1
33	0	0	1	2	1	1
34	2	0	2	1	1	1
35	1	1	1	2	2	1
36	2	1	0	1	0	1
37	2	2	1	2	2	1
38	0	1	0	0	0	1
39	2	2	0	0	0	1
40	1	0	1	2	2	1
41	0	1	0	2	1	1
42	2	0	0	1	0	1
43	1	1	0	1	0	1
44	1	0	0	2	0	1
45	2	1	0	0	0	1
46	2	0	2	2	2	1
47	0	1	0	1	1	1
48	2	2	1	0	0	1
49	1	2	0	2	1	1
50	1	2	1	2	1	1
51	2	1	2	2	2	1
52	2	2	0	2	1	1
53	1	0	1	0	0	1

* LOC=Lines of Code, Cyc=Cyclomatic number, Vol=Volume, Dif=Difficulty, Eff=Effort

$$\begin{aligned} & \vee h_A(Eff) \wedge g(\{Eff, Vol\}) \\ & \vee h_A(Vol) \wedge g(\{Vol\}) \\ & = (0.38 \wedge 1) \vee (0.77 \wedge 0.76422) \vee (0.94 \wedge 0.56250) \vee \\ & (0.97 \wedge 0.31771) \vee (0.98 \wedge 0.18229) \\ & = 0.76 \end{aligned}$$

4.2 재사용도 척도의 검증

표 6과 같은 측정값들을 갖는 실험을 위해 수집된 18,281개의 C 언어로 쓰여진 컴포넌트들을 표 5의 분류 기준에 따라 분류하고, 중복된 규칙들을 제거하여 표 7과 같이 지식표현 시스템을 구성하였다. 표 7에서 C={LOC, Cyc, Vol, Dif, Eff}이고, D={reuse}이다.

표 7의 규칙 1의 (2, 0, 0, 2, 1)는 50 < LOC ≤ 100, Cyc ≤ 10, Vol ≤ 1000, 25 < Dif ≤ 100, 10000 < Dif ≤ 100000 을 의미하고, Reuse=1은 현재의 컴포넌트가 재사용 컴포넌트임을 의미한다.

식(2)~(4)를 적용하여 측정된 각 척도들의 상대적 중요도는 표 8과 같다

주어진 컴포넌트들에 식 (1)을 적용한 퍼지 소속등급과 식 (5)를 적용한 재사용도의 분포는 표 9와 그림 2와 같다. 표 5에서 제시된 분류 기준에 대한 퍼지 소속등급과 재사용도 측정값은 표 10과 같다.

4.2.1 분류 기준의 검증

식 (6)이 성립되는 정도를 검증하기 위해서 각 측정 인자들에 대한 분할표(contingency 표)을 구성하였

표 8. 각 척도들의 상대적 중요도

척도	상대적 중요도
Lines of code	0.23958
Cyclomatic number	0.24479
Volume	0.18229
Difficulty	0.19792
Effort	0.13542

표 9. 각 컴포넌트들의 소속등급과 재사용도 측정값

컴포넌트	LOC	Cyc	Vol	Dif	Eff	재사용도
1	0.38	0.94	0.98	0.77	0.97	0.76
2	0.84	0.99	0.99	0.96	0.99	0.84
3	0.57	0.98	0.98	0.78	0.98	0.76
4	0.75	0.96	0.99	0.48	0.98	0.75
5	0.69	0.99	0.98	0.93	0.99	0.76
.
.
.
18,277	0.85	0.99	0.99	0.97	0.99	0.85
18,278	0.90	0.99	0.99	0.99	0.99	0.90
18,279	0.99	0.99	0.99	0.98	0.99	0.98
18,280	0.98	0.99	0.99	0.98	0.99	0.98
18,281	0.91	0.99	0.99	0.93	0.99	0.91

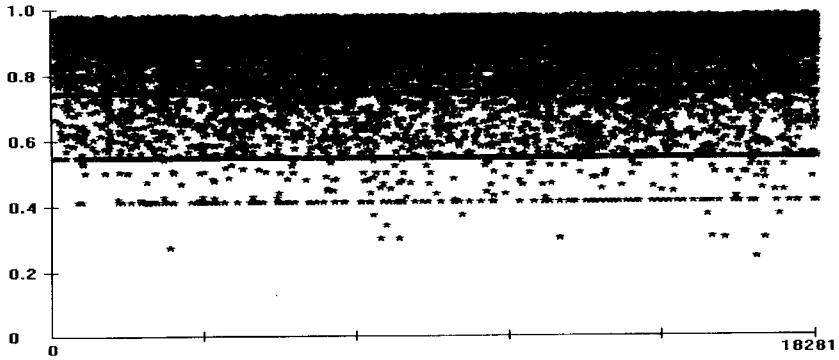


그림 2. 재사용도 측정값의 분포

표 10. 분류기준에 대한 재사용도 측정값

분류기준	LOC	Cyc	Vol	Dif	Eff	재사용도
0	0.82	0.92	0.98	0.98	0.99	0.82
1	0.50	0.68	0.68	0.87	0.77	0.68
2	0	0	0	0	0	0

표 11. LOC의 분할표

재사용도	LOC			합계
	≤30	30~50	≤100	
높다	11292	0	0	11292
보통	272	3514	1225	5011
낮다	0	291	1687	1978
합계	11564	3805	2912	18281

표 12. 측정인자들의 chi-square

척도	χ^2
Lines of code	22550.88
Cyclomatic number	6547.54
Volume	9843.99
Difficulty	11126.34
Effort	16267.63

표 13. 컴포넌트들의 순위 표

컴포넌트	LOC	Cyc	Vol	Dif	Eff	재사용도
1	2222	2909	3686	3283	3443	2860
2	7062	15674	6332	11898	8589	7574
3	3332	7818	4323	3362	3828	4824
4	5241	4150	6378	940	3548	2749
5	4547	9893	3924	8388	5618	4806
.
.
.
18,277	7564	13704	12844	13222	13084	7741
18,278	9376	13780	8125	18079	14850	9643
18,279	17356	13572	17063	14455	16330	16583
18,280	16260	13666	15297	13656	14768	16288
18,281	9785	9904	11730	8330	10389	10220

다. 예를 들어서 LOC에 대한 분할표는 표 11과 같다.

표 12에서 보여주듯이, 각 측정인자들에 대한 χ^2 의 값은 임계치($\beta_{0.01}=13.28$) 보다 매우 높게 나타난다. 결국 표 5에서 주어진 분류 기준들은 재사용도의 구분력에 있어서 유효하다고 할 수 있고, 우리는 재사용도가 높음($F \geq 0.82$), 보통($0.68 \leq F < 0.82$), 낮음($F < 0.68$)을 구분하기 위해서 분류기준들을 사용할 수 있다.

그림 2의 분포도와 표 12의 값들에서 알 수 있듯이 각 측정인자들의 값이 클수록 재사용도가 낮은 것으로 나타났다. 즉, 재사용도가 낮은 컴포넌트들은 기준 값보다 측정값이 컸으며, 또한 각 측정 인자들의 분포와 컴포넌트들의 재사용도의 분포가 대응되었다.

4.2.2 재사용도 척도와 ordinal scale

식 (9)가 성립되는 정도를 검증하기 위해서 표 13과 같이 표 9의 각 컴포넌트들의 순위(rank)를 부여하였다.

표 13에 대한 분석 결과 유의수준 0.01에서 M과 F의 순위 관계는 유의하고, 순위 결정계수는 $r = 0.9999$ 로서 F와 M사이에는 99.99% 정도의 순위 관계가 성립하는 것으로 나타났다. 즉, 척도 F는 ordinal scale을 전제한다고 할 수 있다. 그러므로 우리는 F를 컴포넌트들이 상대적 재사용도에서 얼마나 다른가를 평가하기 위해서 사용할 수 있다.

예를 들어서, 표 9에서 모듈 2가 모듈 1보다 재사용도가 높다고 말할 수 있다.

4.2.3 재사용도 척도와 ratio scale

식 (11)이 성립되는 정도를 검증하기 위해서 표 9에 대한 중회귀분석을 수행하였다. 분석결과 선형회귀 모형은 유의수준 0.01에서 유의하고, $f(M)$ 의 예측 가능성 결정계수가 $R^2 = 0.9999$ 로서 $\beta_{0.01} (= 0.99)$ 보다 큰 것으로 나타났다. 즉, $f(M)$ 이 99.99% 이상 정확하게 F의 값을 정확히 예측할 수 있으므로 척도 F는 ratio scale에 따른다고 할 수 있다. F가 ratio scale에 따르

면, 산술평균과 비율(퍼센트)을 계산하고 비교할 수 있다[5]. 우리는 F를 컴포넌트들 간의 상대적 재사용도를 비교하기 위해서 사용할 수 있다.

4.3 평가

컴포넌트의 재사용성을 측정하려는 연구들에는 퍼지집합과 사용자의 경험을 바탕으로 해당 컴포넌트를 재사용할 때 요구되는 노력의 양을 측정하여 후보 컴포넌트들의 등급을 결정하는 Diaz[13]의 연구, 각각의 항목에 대한 상대적인 중요도를 사용자가 직접 명시하도록 허용하여, 사용자의 입력 값에 의하여 후보 컴포넌트들의 등급을 결정하는 Burton[2]의 연구, 엔트로피에 기반을 둔 학습 알고리즘을 이용한 권기현[18]의 연구 등이 있다.

이런 연구들은 메트릭의 삽입과 삭제가 어렵고, 정량적인 비교 평가를 위해서 정성적인 메트릭스를 정량화해야 하고, 컴포넌트의 추가 삭제가 어렵고, 항목들의 중요도에 대한 객관적인 제시가 없으며, 비슷한 유형의 컴포넌트들이 존재하는 상황에서 사용자가 최종적으로 재사용 여부를 판정해야 하며[18], 가정된 전문가의 지식에 근거하는 문제점이 지적되었다.

본 연구에서 제안한 재사용도 측정은 재사용 컴포넌트들에서 추출된 정보에 근거하여 재사용도를 측정하므로 메트릭의 삽입과 삭제가 용이하다. 메트릭의 삽입이 발생한 경우, 메트릭의 분류 기준이 주어지면 전체 지식표현 시스템과 재사용 평가를 위한 퍼지 소속함수가 자동으로 생성되며, 추가된 메트릭을 포함한 재사용도가 측정될 수 있다. 둘째, 정량화된 측정값들이 각 측정 속성들의 분포에 대응하였다. 그림 2와 표 6의 값들에서 알 수 있듯이 각 측정 요소들의 값이 클수록 재사용도가 낮은 것으로 나타남은 측정의 정성적 형태(경험적 관계)를 정량적인 형태(경험적 관계의 의미)로 변환하는데 문제가 없음을 알 수 있다. 셋째, 컴포넌트의 추가 삭제가 용이하다. 즉 재사용 컴포넌트들에서 추출된 데이터에 근거하므로 주어진 컴포넌트를 포함한 측정은 제안된 측정에서는 자연스런 현상이다. 넷째, 본 연구에서는 러프집합을 이용하여 재사용 결정에 미치는 각 메트릭들의 중요도를 산출하였다. 다섯째, 산출된 측정값들이 측정이론에서 다루어지는 ordinal과 ratio 스케일에 따름으로 사용자는 객관적인 평가 자료를 통해서 각 컴포넌트의 재사용도를 비교 평가할 수 있다. 여섯째, 제안한 측정 방법은 어떤 가정된 지식에 기반하는 것이 아니라 현업에서 실험과 연구를 통해 타당함이 인정된 컴포넌트들에 관한 지식에 근거하여 재사용도를 측정함으로써 판단지식의 정확성에 관한 문제를 어느 정도 완화한다.

5. 결 론

본 논문에서는 현업에서의 연구와 경험을 통해서 증명된 객관성 있는 척도들을 측정인자로 설정하였다. 러프집합을 이용하여 각 측정인자들이 컴포넌트 재사용에 미치는 영향의 정도를 평가하여 각 측정인자들의 중요도를 구한다. 다음으로 중요도를 가중치로 하고, Sugeno의 퍼지적분을 이용하여 측정값들을 종합함으로써, 컴포넌트들의 재사용도를 측정할 수 있는 방법을 제안하였다.

우리는 제시된 분류기준이 타당함을 보여주었고, 제안된 척도가 ordinal scale과 ratio scale을 따름을 보여주었다. 결국 제안된 척도를 이용하여 각 컴포넌트들의 재사용도를 비교할 수 있음을 보여주었다. 즉, 사용자는 제안된 척도를 이용하여 유사한 컴포넌트들 중에서 재사용하기에 가장 쉬운 컴포넌트를 선택할 수 있음을 보여주었다.

향후 연구 과제로는 척도를 C 언어 이외의 다른 언어로 작성된 컴포넌트들에 적용할 수 있는 방안에 대한 연구가 요구된다. 또한 더 많은 측정 인자들을 도입하는 방안과 객체 지향 언어에서의 척도의 적용에 대한 연구가 요구된다.

참고문헌

- [1] A. Balog, R. Trifu and A. Raduta, "Quality Evaluation of Public Administration Software Products: A Practical Study", *The 5th European Conference on Software Quality*, pp. 276-285, 1996.
- [2] B.A. Burton, et. al, "The Reusable Software Library", *Software Reuse: Emerging Technology*, IEEE CS Press, pp. 129-137, 1990.
- [3] Caldiera, G. and V.R. Basili, "Identifying and Qualifying Reusable Software Components", *IEEE Computer*, Vol. 24, No.2, pp. 61-70, Feb., 1991.
- [4] Choi W.K., et.al, "The Analysis of Significance of the Reusability Decision Metrics Using Rough Set", *Proceedings of the Third Asian Fuzzy Systems Symposium*, pp. 302-307, 1998.
- [5] Horst Zuse, "Software Complexity-Measures and Methods", pp. 39-66, 560-563, Walter de Gruyter, Berlin · New York, 1991.
- [6] Lesiie J. Waguespack and Sunil Baldani, "Software Complexity Assessment : An Introduction and Annotated Bibliography", *ACM SIGSOFT*, Software Eng. Notes, Vol. 12, No. 4, pp. 52, Oct., 1987.
- [7] Lewis John, Henry Salie, "A Methodology for Integrating Maintainability Using Software Metrics", *Proceedings : Conference on Software Maintenance*, Miami, Florida, IEEE, pp. 32-39, Oct., 1989.
- [8] Lowell J. Arthur, "Measuring Programmer Productivity and Software Quality", pp. 138-142, John

- Wiley&Sons, New York, 1985.
- [9] Norman F. Schneidewind, "Methodology For Validating Software Metrics", *IEEE Transactions On Software Engineering*, Vol. 18, No. 5, pp. 410-422, May, 1992.
- [10] Pawlak Z., "Rough sets", *International Journal of Computer and Information Sciences*, 11, pp. 341-356, 1982.
- [11] Pawlak Z., *Rough Sets-Theoretical Aspects of Reasoning about Data*, Kluwer Academic Publishers, Dordrecht, Boston, London, 1991.
- [12] Peter Freeman, "TUTORIAL : Software reusability", pp. 2-8, pp. 10-23, pp. 50-56, pp. 106-116, pp. 151-154, pp. 155-178, IEEE Computer Society Press, Washington D.C., 1987.
- [13] R.Prieto-Diaz, P.Preeman, "A Software Classification Scheme Reliability", *Tutorial : Software Reusability*, ed. P. Freeman, IEEE Press, pp. 106-116, 1987.
- [14] Rogers S.Pressman, "Software engineering : A Practitioner's approach", pp. 452-458, McGraw-Hill Book Company, NewYork 1987.
- [15] Szentes J., Gras j., "Some Practical Views of Software -Complexity metrics and a Universal Measurement Tool", *First Australian Software Engineering Conference*, Canberra, pp. 14-16, May, 1986.
- [16] Taghi M. Khoshgoftaar, "Predictive Modeling Techniques of Software Quality from Software Measures", *IEEE Transactions On Software Engineering*, Vol.18, No. 11, pp. 979-987, Nov., 1992.
- [17] Warren Harrison et. al, "Applying Software Complexity Metricc to Program Maintenance", *Computer Journal*, Vol. 15, pp. 65-79, Sep., 1982.
- [18] 권기현, 이경환, "학습알고리즘을 이용한 재사용 의 사결정 지원", *한국정보과학회지*, 제 21권, 제12호, pp. 2235-2242, 1994.
- [19] 이경환, 「소프트웨어 재이용을 위한 연구」, 연구보고서, 과학기술처, BSN20840, 1990.
- [20] 이성주 정환목, 최완규, "러프 집합과 응용", 조선대학교출판국, 1998.
- [21] 이형광, 오길록, "퍼지이론 및 응용", pp. 1.1-1.40, 9.1-9.36, 홍릉과학출판사, 1991.



박 병 권 (Byung-Gweun Bak)

1988년 : 조선대학교 전자계산학과 졸업 (학사)
 1990년 : 조선대학교 전자계산학과 졸업 (석사)
 1991 ~ 1994년 : 광주은행 전산업무부
 1998년 : 조선대학교 대학원 전자계산학과 박사과정 수료
 1995~현재 : 서강정보대학 전자계산학과 조교수

관심분야 : 소프트웨어 공학, 객체지향시스템, 퍼지집합, 러프집합



김 혜 경 (Hye-Kyoung Kim)

1993년 : 조선대학교 전산통계학과 졸업 (학사)
 1998년 : 조선대학교 대학원 전자계산학과 박사과정
 1999~현재 : 조선대학교 대학원 전자계산학과 박사과정
 관심분야 : 소프트웨어 공학, 객체지향 시스템, 퍼지집합, 러프집합



최 완 규 (Wan-Kyoo Chio)

1988년 : 서울대학교 종교학과 (학사)
 1992~1993년 : (주)공성통신 전산실
 1993~1995 : 한양시스템 전산실
 1997년 : 조선대학교 전자계산학과 (석사)
 1997~현재 : 조선대학교 전자계산학과 박사과정

관심분야 : 소프트웨어 공학, 프로그래밍 언어, 객체지향 시스템, 러프집합



이 성 주 (Sung-Joo Lee)

1970년 : 한남대학교 물리학과 (학사)
 1992년 : 광운대학교 전자계산학과 (이학석사)
 1998년 2월 : 대구 효성카톨릭대학교 (이학박사)
 1988~1990년 : 조선대학교 전자계산소 부소장
 1995~1997년 : 조선대학교 정보과학대학장

1981~현재 : 조선대학교 전자계산학과 교수
 관심분야 : 소프트웨어 공학, 프로그래밍 언어, 객체지향 시스템, 러프집합