

## UML순차도의 조종구조를 리용한 소프트웨어프로젝트크기결정방법

승남철, 한석민

프로젝트크기를 결정하는 대표적인 방법[1]으로 LOC, FP, COCOMO 등이 있으며 최근에는 UML도식들을 리용하여 프로젝트크기를 결정하기 위한 연구가 진행되고있다. 이 방법들에서는 UML설계의 유스케스도(Use Case Diagram), 클래스도(Class Diagram), 순차도(Sequence Diagram) 등을 리용하여 고준위립도에서 프로젝트크기를 결정하였다.

립도(granularity)는 대면부가 제공하는 조작의 개수와 하나의 조작호출로 전송되는 자료의 량이다. 고준위립도에서 프로젝트크기평가는 체계의 내부상태를 검은통으로 보고 프로젝트의 거시적인 크기를 빨리 평가할수 있으나 흰통검사를 진행하는데 드는 노력이 정확히 평가되지 않는다.[2]

UML순차도를 리용하여 프로젝트크기를 평가하는 COSMIC결정방법에서는 UML순차도에서 통보문들의 개수를 리용하여 프로젝트크기를 결정하였다. 이 방법은 순차도의 매 연산자들에 대한 고려가 없으므로 저준위립도로 실현되는 프로젝트들의 크기평가를 정확하게 할수 없다.

론문에서는 프로젝트크기를 정량화하기 위한 한가지 방법으로서 저준위립도로 실현되는 프로젝트크기결정방법을 제안하였다.

### 1. 프로젝트크기평가방법들과 UML순차도의 조작적의미론

프로젝트크기를 결정하는 한가지 방법인 FP는 소프트웨어요구명세서로부터 기능들의 개수를 구하고 그것들의 난도에 따라 무게를 곱하고 다시 일정한 상수를 곱하는 방법으로 계산된다.

프로그램작성언어에 의존하지 않는 FP에 의한 프로젝트크기결정방법을 현실에 적용하기 쉽도록 하기 위한 COSMIC결정방법은 사용자기능요구들을 나타내는 UML순차도의 크기를 측정하여 프로젝트크기를 결정한다.

UML순차도에서 리용되는 대표적인 호상작용연산자들의 조작적의미론은 표 1과 같다.

표 1. UML순차도에서 리용되는 대표적인 호상작용연산자들의 조작적의미론

호상작용연산자	기능설명	조작적의미론
alt	론리조건에 기초하여 서로 다른 통보문을 전달하는 호상작용토막들로 분할한다.	if A then X else Y (A는 조건식, X, Y는 통보문)
opt	조건식이 참이면 통보문을 전달한다.	if A then X (A는 조건식, X는 통보문)
par	통보문들이 병행으로 전송된다는것을 의미한다.	(X; Y) or (Y; X)
loop	호상작용제약으로 서술된 수만큼 통보문이 반복전송된다.	while A then X (A는 조건식, X는 통보문)

## 2. UML순차도의 조종구조를 고려한 프로젝트크기결정방법

### 1) UML순차도통보문들의 형태에 따르는 무게결정

COSMIC결정방법에서는 프로젝트크기결정에 영향을 주는 통보문들의 형태를 결정하였다.(그림 1)

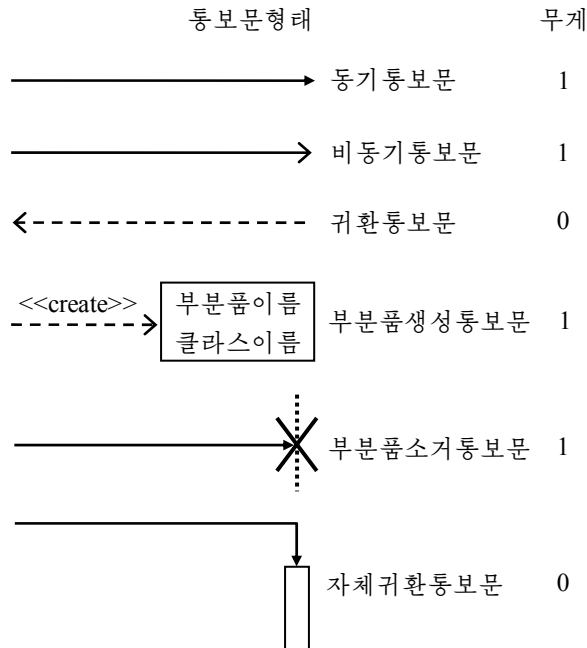


그림 1. COSMIC에서 정의한 UML순차도통보문형태에 따르는 무게

이 방법에서는 전체 프로젝트크기 CFP를 모든 통보문들의 무게를 합한것으로 결정하였다.

$$CFP = \sum_{m_i \in MSG} FS(m_i)$$

여기서  $m_i$ 는  $i$ 번째 통보문이고  $FS(m_i)$ 는 그 통보문에 할당된 무게이다.

### 2) alt, opt, loop연산자들의 무게결정방법

COSMIC결정방법은 UML순차도의 조종구조에 대한 고려가 없으므로 조종구조연산자들의 무게를 평가하기 위하여 alt, opt, loop연산자들에 적당한 무게를 할당하고 프로젝트크기결정에 리용한다.

매 연산자의 무게결수  $g$ 를 그 조작적의미론으로부터 다음과 같이 설정한다.

$$g(x) = \begin{cases} 1, & x = \text{opt} \\ 2, & x = \text{alt} \\ n, & x = \text{loop} \\ 0, & \text{기타} \end{cases}$$

식에서 보여준것처럼 opt연산자에는 무게 1을, alt연산자에는 무게 2를, loop연산자에는

순환의 수  $n$ 을, 기타 연산자들에 0을 무게로 준다.

UML순차도의 alt, opt, loop연산자들의 구조는 그림 2와 같다.

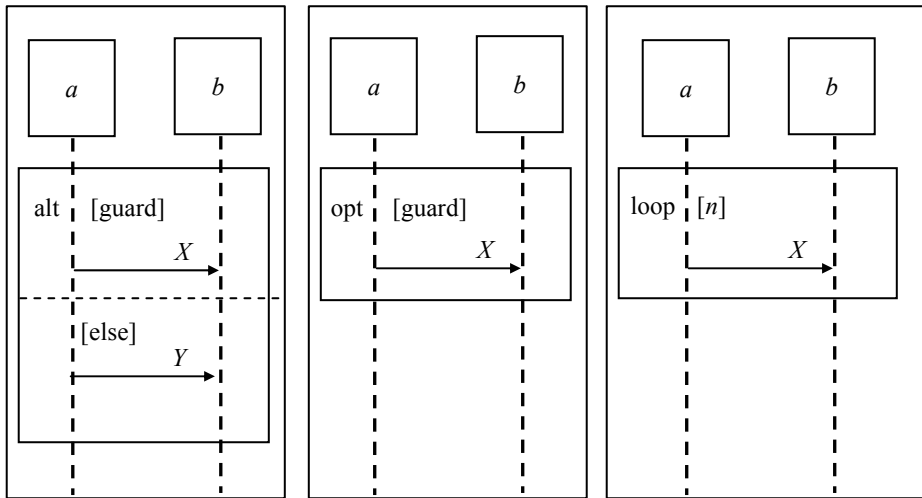


그림 2. UML순차도의 alt, opt, loop연산자들의 구조

연산자들이 중첩되어 놓이는 경우에는 시험경로가 조건문이나 순환의 수들에 대한 적으로 표시되므로 조종구조의 크기는 그안에 놓이는 연산자들의 무게의 적으로 표현할 수도 있으나 일반적으로 독립경로시험을 진행하므로 연산자들의 중첩구조에서의 무게를 매 연산자들의 무게합으로 정의한다.

연산자들의 무게합 OPFP는 다음과 같이 표시된다.

$$OPFP = \sum_{i=1}^m g_i(x)$$

여기서  $m$ 은 순차도에 들어있는 연산자의 개수이고  $g_i(x)$ 는  $i$ 번째 연산자의 무게이다.

이로부터 프로젝트크기를 통보문들의 무게합과 조종구조크기의 합으로 평가한다.

$$ACSPF = CSP + CSFP$$

이러한 평가방법은 통보문들의 무게합을 그대로 리용하면서도 소프트웨어의 조종구조를 반영한것으로 하여 프로그램작성자와 소프트웨어검사자들의 노력을 정확히 평가할수 있는 또 하나의 기준으로 된다.

### 3. 저준위립도에서 UML순차도에 의한 일정계획작성프로젝트크기결정

제안한 방법을 리용하여 업무체계개발에서 흔히 보게 되는 일정계획작성프로젝트크기를 평가해보자.

일정계획작성체계에 대한 UML순차도는 그림 3과 같다.

MVC모형에 기초한 프로그램작성으로 보고 부분품을 View, Controller, Model로 설정하였다.

COSMIC결정방법과 논문에서 제기한 ACSFP방법으로 평가한 결과는 다음과 같다.

공정지표개수가  $m=10$ 이고 매 공정지표당 공정내용의 개수가  $n=10$ 인 경우 프로젝트

크기를 결정하면 다음과 같다.

순차도에 alt연산자 3개, loop연산자 2개, opt연산자 2개가 있다.

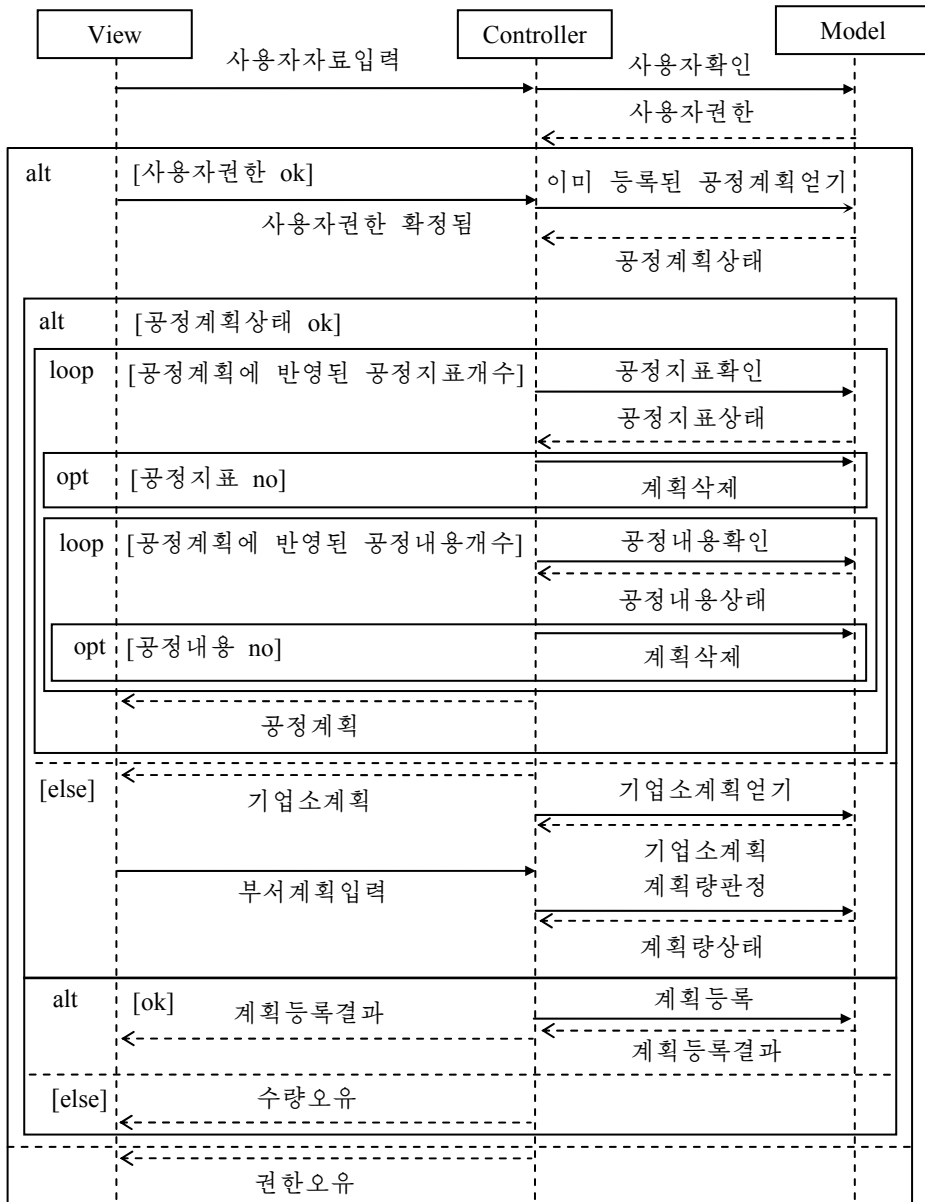


그림 3. 일정계획작성체계에 대한 UML순차도

표 2에 일정계획작성프로젝트크기를 보여주었다.

표 2. 일정계획작성프로젝트크기

측정방법	프로젝트크기
COSMIC	9
ACSF	$9+2 \times 3+10+10+1 \times 2 = 37$

표 2에서 보여준것처럼 COSMIC결정방법은 통보문의 개수를 프로젝트크기로 평가하였으므로 프로젝트의 기능점수와 프로그램코드행수를 결정할수 있다. 그러나 프로젝트가 저준위립도로 실현되는 경우 UML순차도의 조종구조를 고려하여 측정한 ACSFP가 프로젝트크기를 상세하게 결정할수 있으며 특히 이것은 소프트웨어에 대한 검사에서 시험경우를 생성하는 노력을 COSMIC결정방법보다 더 정확히 평가할수 있다.

## 맺 는 말

UML순차도를 리용하여 프로젝트크기를 평가하는 COSMIC결정방법의 제한성을 극복하기 위하여 순차도의 연산자들의 조작적의미론을 분석한데 기초하여 연산자들의 무게결정방법을 제안함으로써 프로젝트크기를 더 정확히 평가할수 있는 한가지 방법을 제안하고 그것을 일정계획작성을 위한 프로젝트에 적용하여 평가하였다.

## 참 고 문 헌

- [1] Nabil Messaoudi et al.; Procedia Computer Science, 83, 658, 2016.
- [2] Asma Sellami et al.; Information and Software Technology, 59, 222, 2015.

주체107(2018)년 8월 5일 원고접수

### **A Method for Determinating Size of Software Project Using Control Structure of the UML Sequence Diagram**

*Sung Nam Chol, Han Sok Min*

In this paper we analyzed the semantics of opt, alt and loop operators in UML sequence diagram and decided on its structural size, and suggested a method for determinating size of project with the weight of the UML sequence diagram's messages and operators.

Key words: UML, sequence diagram, model checking