

# Gompertz 소프트웨어 비용 추정 모델

이 상 운<sup>†</sup>

요 약

본 논문은 소프트웨어 비용추정 모델의 적합성을 평가하고, 가장 적합한 모델을 제시하였다. 먼저, 해당 모델의 함수를 변수변환시켜 선형식으로 만든다. 다음으로 실제 개발 소프트웨어의 비용 데이터가 모델의 선형식에 얼마나 적합한지로 모델의 성능을 평가한다. 모델 성능평가에는 절대오차 대신 상대오차 개념인 MMRE를 적용하였다. 기존의 소프트웨어 비용추정 모델은 Weibull, Gamma와 Rayleigh 함수를 따르고 있다. 본 논문에서는 성장곡선의 일종인 Gompertz 곡선 모델을 제안하였다. 추가로 다른 성장곡선들도 적합성을 검증하였다. 모델 성능평가 결과 Gompertz 성장곡선이 소프트웨어 비용추정 모델로 가장 적합한 성능을 보였다.

키워드 : 소프트웨어 비용추정 모델, 와이불 함수, 감마 함수, 레일레이 함수, 고펜츠 함수, 변수변환

## A Gompertz Model for Software Cost Estimation

Sang-Un Lee<sup>†</sup>

ABSTRACT

This paper evaluates software cost estimation models, and presents the most suitable model. First, we transformed a relevant model into variables to make in linear. Second, we evaluated model's performance considering how much suitable the cost data of the actual development software was. In the stage of model performance evaluation criteria, we used MMRE which is the relative error concept rather than the absolute error. Existing software cost estimation model follows Weibull, Gamma, and Rayleigh function. In this paper, Gompertz function model is suggested which is a kind of growth curve. Additionally, we verify the compatibility of other different growth curves. As a result of evaluation of model's performance, Gompertz function was considered to be the most suitable for the cost estimation model.

Key Words : Software Cost Estimation Model, Weibull Function, Gamma Function, Rayleigh Function, Gompertz Function, Variable Transformation

### 1. 서 론

소프트웨어 개발 관리자에게 개발될 소프트웨어 비용(또는 노력)을 사전에 보다 정확히 추정하는 능력과 더불어 추정된 총 노력을 개발의 각 시점별로 적절히 분배 관리하여 프로젝트를 성공적으로 완수할 수 있는 능력이 가장 크게 요구되고 있다. 왜냐하면, 적절한 개발비용 확보와 관리는 프로젝트의 이익을 창출할 수 있기 때문이다. 이러한 능력을 구비하는데 필요한 것이 소프트웨어 비용추정 모델이다.

소프트웨어 비용 추정 모델은 비용 모델(Cost Model)과 제약 모델(Constraint Model)로 분류된다[1]. 일반적으로 비용모델은 소프트웨어에 소요되는 총 비용을 추정하는 과정에 적용되고, 제약모델은 추정된 총 비용을 개발 각 단계별

로 인력을 적절히 분배하는데 적용되고 있다. 그러나 제약 모델을 이용해 소프트웨어 개발에 소요되는 총 비용을 추정할 수도 있다. 비용모델은 주 비용 인자인 소프트웨어 규모(Size)와 2차적인 조절 요인 또는 비용에 영향을 미치는 요인들로부터 노력(Effort)을 직접 추정하는 회귀모델이다. 전형적인 비용추정 회귀모델은  $E = a + bS^c$ 의 형태를 취하고 있으며, 대표적인 모델로는 COCOMO[2]와 COCOMO II[3]가 있다. 여기서  $E$ 는 소프트웨어 개발 노력(또는 비용),  $S$ 는 소프트웨어 규모이며,  $a, b, c$ 는 상수이다. 제약모델은 시간에 따른 노력, 기간(Duration) 또는 인력(Manpower)간의 관계를 표현하는 모델로 생명주기 모델(Life Cycle Model) 또는 모수 비용 분석 모델(Parametric Cost Analysis Model)이라고도 부른다. 대표적인 제약모델로는 Putnam의 Rayleigh 모델[4], Pillai와 Sukumaran Nair의 Gamma 모델[5]과 Knafll과 Gonzales의 Weibull 모델[6]이 있다.

과거 30년 동안 소프트웨어 비용 추정 분야에 대한 많은

<sup>†</sup> 정 회 원 : 강릉대학교 컴퓨터정보공학부 조교수  
논문접수 : 2007년 8월 8일, 심사완료 : 2008년 1월 15일

연구가 진행되고 있지만 지금까지도 개발비용을 정확히 추정하고 인력을 적절히 분배할 수 있는 모델이 없는 실정이다[7]. 이의 실 예로 Standish CHOAS 보고서 2001[8]에 따르면 28%의 프로젝트만이 사전에 추정된 일정과 비용으로 개발이 완료된 반면에 49%의 프로젝트는 계획 대비 45%의 비용이 초과되었기 때문이다.

본 논문은 비용모델로 총 노력(비용)이 적절히 추정되었다고 가정하고, 추정된 총 노력에 기반하여 개발 시점별로 인력을 적절히 분배할 수 있는 제약모델을 제안하여 프로젝트의 성공률을 보다 향상시키고자 한다. 먼저, 기존에 제시된 제약모델의 성능을 비교하고, 보다 적합한 모델을 제시한다. 일반적으로 적용하고 있는 MMRE (Mean Magnitude of Relative Error)[9]로 모델의 성능을 평가한다. 제안된 모델을 적용함으로써 프로젝트 관리자는 보다 효율적이고 현실에 적합한 인력분배를 통해 프로젝트를 성공률을 향상시킬 수 있을 것이다.

2장에서는 비용추정 제약모델과 모수를 추정하는 방법을 고찰하여 본다. 3장에서는 새로운 제약모델을 제시하고 모수를 추정하는 방법도 기술한다. 4장에서는 실제 수행된 프로젝트 데이터를 대상으로 제약모델들의 성능을 검증한다. 마지막으로 5장에서는 결론 및 향후 연구 과제를 기술한다.

## 2. 제약 비용추정 모델과 연구배경

### 2.1 제약 비용추정 모델

Putnam[4]은 소프트웨어 생명주기 전체에 소요되는 총 노력과 설계, 코딩, 시험 및 유지보수 등 각 단계별로 투입되는 노력도 Rayleigh 분포를 따름을 관찰하였다. Rayleigh 분포는 임의의 시간  $t$ 까지 프로젝트에 투입된 누적 노력  $F(t)$  (CDF, Cumulative Distribution Function)는 식 (1)로 표현된다. 또한, 임의의 시간  $t$ 에서 프로젝트에 투입되는 노력  $f(t) = dF(t)/dt$  (pdf, Probability Density Function)는 식 (2)로 유도된다. 여기서  $E$  (Life Cycle Effort)는 Rayleigh 곡선 아래에 있는 영역으로 년 인원으로 표시되며, 소프트웨어 생명주기 전반에 걸쳐 투입된 총 노력이다.  $a = 1/2t_d^2$ 이며,  $t_d$ 는  $f(t)$ 가 최대가 되는 시점으로 경험적으로 볼 때, 개발 종료 시점이다.

$$F(t) = E(1 - e^{-at^2}) \tag{1}$$

$$f(t) = 2Eate^{-at^2} \tag{2}$$

식 (1)과 (2)의 모수  $a$ 를 구하기 위해서는 Rayleigh 분포의 pdf인 식 (2)를 변수변환을 통해 식 (3)과 같이 시간  $t$ 에 대해 선형 식으로 유도하여 추정한다. 즉, 주어진 데이터에 대해  $\ln[\frac{f(t)}{t}]$  vs.  $t^2$  관계가 선형이 된다. 식 (3)은  $\ln[f(t)] = a + bt^2 + \ln(t)$ 로 다시 쓸 수 있다.

$$\ln[\frac{f(t)}{t}] = \ln(\frac{E}{t_d^2}) + [-\frac{1}{t_d^2}]t^2 = a + bt^2 \tag{3}$$

Pillai와 Sukumaran Nair[5]가 제안한 Gamma 분포 함수의 CDF는 식 (4)로, 이를 미분한 pdf는 식 (5)로 표현된다.

$$F(t) = \frac{\alpha \cdot \Gamma(\beta \cdot t; \gamma)}{\Gamma(\gamma)} \tag{4}$$

$$f(t) = \alpha\beta(\beta t)^{\gamma-1} \cdot \frac{e^{-\beta t}}{\Gamma(\gamma)} \tag{5}$$

식 (5)를  $\ln[f(t)] = \ln[\frac{\alpha\beta}{\Gamma(\gamma)}] + (-\beta)t + (\gamma-1) \cdot \ln(t)$  변수변환을 거쳐 간략화하면 식 (6)과 같이 표현된다. 이 식에  $c=2$ 를 대입하여 식 (7)과 같이 시간  $t$ 에 대해 선형함수로 변환시킨 후 모수를 추정할 수 있다.

$$\ln[f(t)] = a + bt + c \cdot \ln(t) \tag{6}$$

$$\ln[\frac{f(t)}{t^2}] = a + bt \tag{7}$$

Pillai와 Sukumaran Nair[5]은 식 (7)로 모수를 추정한 후, Gamma 함수인  $f(t) = \frac{E}{\Gamma(\alpha)\beta^\alpha} t^{\alpha-1} e^{-\frac{t}{\beta}}$ 에  $\alpha = 3, \beta = \frac{t_d}{2}$ 를 대입하여 식 (8)의 Gamma 분포 pdf를 제안하였다.

$$f(t) = \frac{8E}{\Gamma(3)t_d^3} (t^2 e^{-\frac{2}{t_d}t}) = \frac{4E}{t_d^3} t^2 e^{-\frac{2}{t_d}t} \tag{8}$$

Knafl과 Gonzales[6]는 식 (9)와 식 (10)의 Weibull 분포 함수를 제안하였다. 여기서,  $E$ 는 분포의 규모 (Scale)를 결정하며,  $a$ 은 분포의 모양 (Shape)을 결정한다.

$$F(t) = E(1 - e^{-at^a}) \tag{9}$$

$$f(t) = E\alpha t^{\alpha-1} e^{-at^\alpha} \tag{10}$$

식 (10)에 대해  $\ln[f(t)] = \ln(E\alpha) + (-\alpha)t^\alpha + (\alpha-1) \cdot \ln(t)$  변수변환을 수행하고, 단순화 시키면 식 (11)이 된다. Knafl과 Gonzales[6]는 일반화된 Weibull 분포함수로  $\ln[f(t)] = a + bt^d + f \cdot \ln(t)$ 을 제안하고,  $e = 3, f = 1$ 을 적용하여 식 (12)로 모수를 추정하였다.

$$\ln[f(t)] = a + bt^d + (d-1) \cdot \ln(t) \tag{11}$$

$$\ln[\frac{f(t)}{t}] = a + bt^3 \tag{12}$$

식 (11)의 Weibull 분포에서  $d=2$ 이면 식 (3)의 Rayleigh 분포가 되며,  $d=3$ 이면 식 (6)의 Gamma 분포가 된다. 따라서 기존의 제약모델들은 모두 Weibull 분포함수이며, pdf로부터 변수변환을 통해 시간  $t$ 에 대해 선형함수

를 만들어 모수를 추정하는 형태를 따르고 있다.

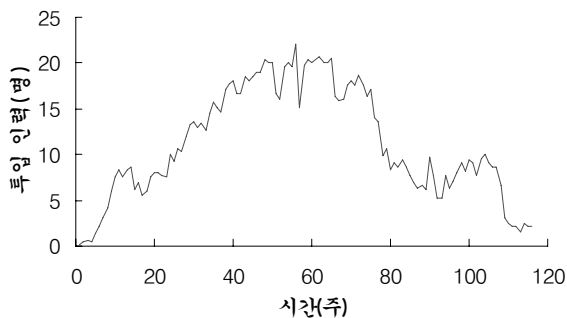
2.2 기존 제약모델의 성능 및 연구배경

모델의 성능 평가는 Warburton[10]가 제시한 (그림 1)의 실제 수행된 프로젝트를 대상으로 하였다. 왜냐하면 Pillai와 Nair[5]가 Gamma 모델을 제시하여 Putnam[4]의 Rayleigh 분포 모델과 비교한 논문과 Knafli과 Gonzales[6]가 Weibull 분포 모델을 제시하여 Rayleigh와 Gamma 모델과 비교한 논문에서도 본 데이터를 활용하였다. 즉, 기존의 모든 모델들을 제안하기 위해 적용한 데이터이기 때문에 본 논문에서도 이 데이터를 활용해 모델의 성능을 평가하고자 한다.

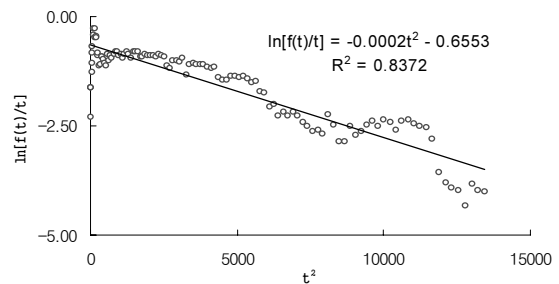
이 데이터는 소나와 화력제어계통 (Sonar and Fire Control System)의 훈련/시뮬레이터 프로젝트로 요구사항 명세화, 설계, 코딩, 시험, 시스템 통합과 납품을 포함한 개발 생명주기 전 과정에 투입된 인력분포를 나타내고 있다. 0.2명에서 최대 22명까지 총 116주 동안 주 단위로 다르게 투입하여 총 1300.8명 (Man/Year)으로 개발되었다. 즉,  $E$ 는 1300.8명임을 알 수 있다.

기존에 제안된 Rayleigh, Gamma와 Weibull 모델의 성능은 (그림 1)의 주당 투입 인력  $f(t)$ 가 시간  $t$ 에 대해 식 (3)의  $\ln[\frac{f(t)}{t}]$  vs.  $t^2$ , 식 (7)의  $\ln[\frac{f(t)}{t^2}]$  vs.  $t$ 와 식 (12)의  $\ln[\frac{f(t)}{t}]$  vs.  $t^3$  관계가 얼마나 선형이 되느냐에 따라 결정된다. 즉, 단위시간에 투입된 인력의 양  $f(t)$ 과 시간  $t$ 와의 관계이다. (그림 1)의 데이터에 대해 이들 관계를 표현하여 (그림 2)에 제시하였다.

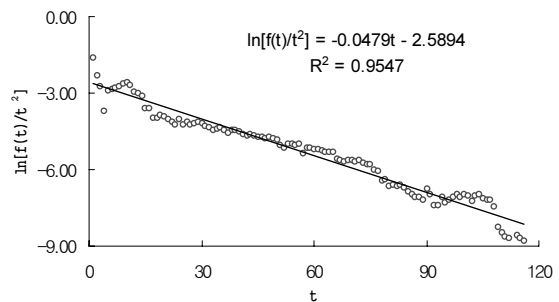
모델의 성능 평가 기준으로 MMRE (Mean Magnitude of Relative Error)를 적용한다. 일반적으로 측정값의 차가 크거나 양의 종류가 다를 경우에는 오차의 한계에 의한 정확도 판정법에는 문제가 있어 측정값과 오차의 비율을 생각한 것이 상대오차(Relative Error, RE)이다. 우리가 일상 생활에서 사용하고 있는 “몇 %의 오차를 감안한다”고 말하는 것은 상대오차를 의미한다. 상대오차가 적을수록 측정값은 정밀하다고 볼 수 있으며,  $RE = \frac{(\text{실측치} - \text{추정치})}{\text{실측치}} \times 100(\%)$ 로 구해진다.  $n$ 개의 데이터가 있을 경우 제안된 모델의 성능을 평가하기 위해서는  $MMRE$  (Magnitude of



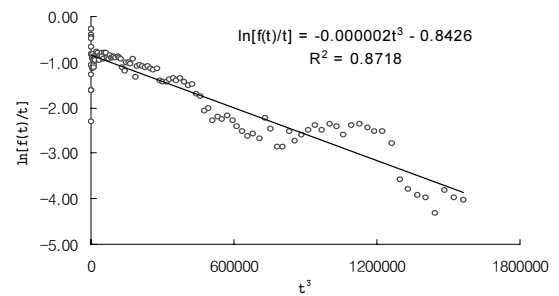
(그림 1) 실제 프로젝트의 단위시간 당 투입인력



(a) Rayleigh 모델



(b) Gamma 모델



(c) Weibull 모델

(그림 2) 기존 모델의 모수추정 회귀분석

the RE) =  $|RE|$ 를 계산하고 다시  $MMRE$  (Mean MRE) =  $\frac{1}{n} \sum MRE_i, i = 1, 2, \dots, n$ 를 계산한다.  $MMRE$ 가 적을수록 제안된 모델이 실제 데이터를 잘 표현한다고 볼 수 있다.

(그림 2)의 관계로부터 모델의 성능을 평가한 결과는 <표 1>과 같다. 결정계수와 MMRE 측면에서 볼 때 모델의 성능은 Rayleigh 함수 < Weibull 함수 < Gamma 함수 순이다. 따라서 Pillai와 Nair[5]의 Gamma 함수가 Putnam[4]의 Rayleigh와 Knafli과 Gonzales[6]의 Weibull 함수와 비교시 가장 좋은 결과를 얻었다.

본 논문에서는 기존의 제약모델과는 별도로 변수변환을 통해 보다 선형적인 관계를 가지는 함수가 추가적으로 존재할 수 있다고 생각되어 이러한 함수를 찾아 적용하고자 하였다.

먼저, pdf의 변수변환을 거친 선형관계인 식 (3)의  $\ln[\frac{f(t)}{t}]$  vs.  $t^2$ , 식 (7)의  $\ln[\frac{f(t)}{t^2}]$  vs.  $t$ 와 식 (12)의

〈표 1〉 모델 성능 평가

모델	관계	모델 성능	
		결정계수( $R^2$ )	MMRE
Rayleigh 함수	$\ln[\frac{f(t)}{t}]$ vs. $t^2$	0.8392	21.29%
Gamma 함수	$\ln[\frac{f(t)}{t^2}]$ vs. $t$	0.9547	5.90%
Weibull 함수	$\ln[\frac{f(t)}{t}]$ vs. $t^3$	0.8718	20.27%

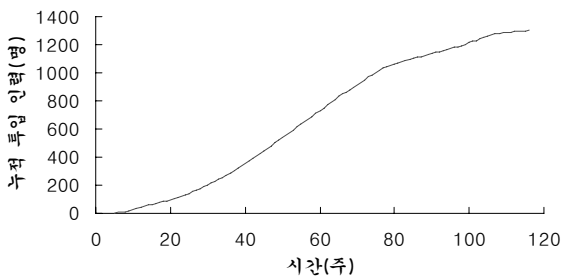
$\ln[\frac{f(t)}{t}]$  vs.  $t^3$ 에서  $t^{0.67}$ ,  $t^{1.33}$ ,  $t^{2.66}$  등  $t$  값을 변화시켜보았으나 기존의 모델을 보다 좋은 결과를 얻지 못하였다. 다음으로, CDF를 변수변환시켜 선형관계를 유도할 수 있는 함수를 찾고자 하였다. 3장에서는 CDF로부터 변수변환을 거쳐 직접 모수를 추정할 수 있는 모델이 소프트웨어 비용추정에 적합한지 검증한다.

### 3. Gompertz 제약 비용추정 모델

(그림 1) 데이터의 CDF인 누적 투입 인력  $F(t)$ 를 그리면 (그림 3)과 같이 S자 형태의 성장곡선을 나타낸다. 따라서 성장곡선 형태를 표현하는 함수를 이용해 소프트웨어 비용을 추정하면 가장 적합한 모델이 될 수 있을 것이다. 따라서 본 논문에서 제안하는 제약 비용추정 모델은 이러한 가정에 기반을 두고 있다.

현존하는 실세계의 많은 성장 현상 (동물, 인간의 성장과정 또는 인간의 학습 능력 등)은 초기에는 점진적으로 증가하다가 중간 구간에서 보다 빠르게 증가하고 한계점으로 도달하면서 성장 속도가 늦어지고 일정한 시간 구간 후에는 최대 값에서 수평이 되는 S자 패턴을 보이고 있다. 성장 곡선을 일명 S자 (S-shaped) 곡선, 시그모이드 (Sigmoidal) 곡선 또는 학습 곡선 (Learning Curve)이라 불리운다 [11,12]. 성장곡선을 표현하기 위해 로지스틱, 일반화된 로지스틱, Pearl, Gompertz 등 다양한 수학적 함수가 제안되었다. 그러나 대부분의 함수가 모수 추정이 어려운 단점을 갖고 있다. 반면에 Gompertz 함수는 CDF를 간단한 변수변환을 거쳐 선형함수로 만들 수 있기 때문에 본 논문에서는 이 모델을 적용한다.

Gompertz 함수는 영국의 공증인이며 수학자인 Benjamin Gompertz[13]에 의해 제안되었으며 식 (13)으로 표현된다.



(그림 3) 실제 프로젝트의 누적 투입인력

이 곡선은  $-\infty$  시간에서는 초기 값 0을,  $+\infty$  시간에서는  $a$  값을 가지며, 곡선이 굴절되는 지점은  $t=\ln(b)/c$  시점으로 곡선이 비대칭이 되는 특징을 갖고 있다.

$$F(t) = ae^{-be^{-ct}} \tag{13}$$

식 (13)은 식 (14)와 같이 두 번의 자연로그 변수변환을 거치면 선형 방정식이 얻어지며, 상용 소프트웨어 도구를 이용하여 선형회귀분석을 수행하면 모수들의 값을 구할 수 있다.

$$\ln[-\ln(\frac{F(t)}{a})] = \ln(b) - ct \tag{14}$$

이 함수의 모수를 추정하는데 있어 한 가지 어려운 점은 모수  $a$ 의 적절한 값을 결정하는 점이다. 소프트웨어의 총 개발비용을 나타내는  $a$ 는 모델에서 가장 중요한 모수이므로 이론적인 근거 없이 시행착오적 방법으로 추정할 수는 없다. 따라서 통계학분야에서 모수추정에 일반적으로 적용되고 있는 최소 자승법이나 최우 추정법을 활용할 수 있을 것이다. 본 논문에서는 이의 대안으로  $\ln[-\ln(\frac{F(t)}{a})]$ 와  $t$  사이의 상관계수가 최대가 되도록 데이터를 가급적 직선 형태로 만드는 방법을 적용하였다. 이 방법이 식 (14)를 이용하는 가장 합리적인 추정법이라 할 수 있다.

추가적으로 변수변환을 통해 모수를 간단하게 추정할 수 있는 식 (15)~(17)의 함수들도 참고로 적합성을 검증하여 본다.

$$F(t) = ab^t$$

$$\log[F(t)] = \log(a) + \log(b) \cdot t \tag{15}$$

$$F(t) = ae^{bt}$$

$$\ln[F(t)] = \ln(a) + b \cdot \ln(t) \tag{16}$$

$$F(t) = at^b$$

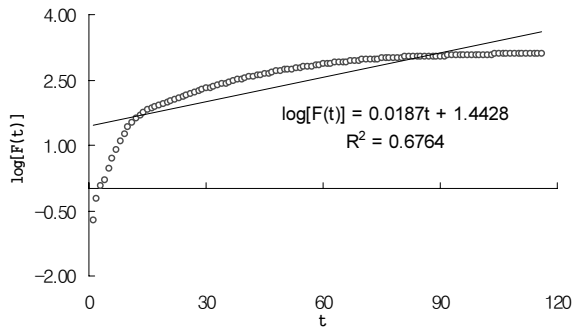
$$\log[F(t)] = \log(a) + b \cdot \log(t) \tag{17}$$

### 4. 적용 및 평가

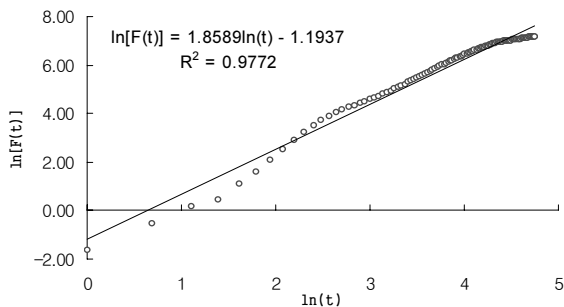
(그림 3) 데이터를 식 (14)의  $\ln[-\ln(\frac{F(t)}{a})]$  vs.  $t$ , 식 (15)의  $\log[F(t)]$  vs.  $t$ , 식 (16)의  $\ln[F(t)]$  vs.  $\ln(t)$ 와 식 (17)의  $\log[F(t)]$  vs.  $\log(t)$  관계를 그리면 (그림 4)와 같다.

여기서, Gompertz 함수의 모수  $a$ 는  $\ln[-\ln(\frac{F(t)}{a})]$ 와 시간  $t$  사이의 상관계수가 최대가 되는 1,394를 적용하였다.

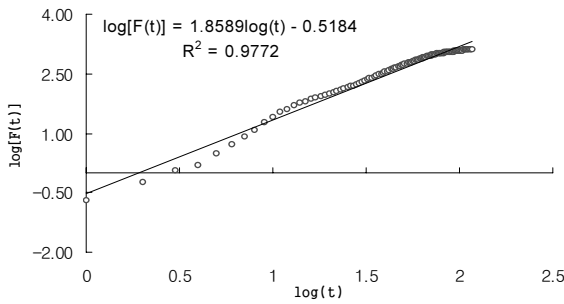
(그림 4)로 얻어진 데이터의 결정계수와 MMRE는 <표 2>와 같다. <표 2>를 기준에 제안된 제약모델 성능인 <표 1>과 MMRE를 비교하면 Rayleigh < Weibull <  $ab^t$  < Gamma <  $ae^{bt}$  <  $at^b$  < Gompertz 함수 순으로 성능이 좋을 수 있다.



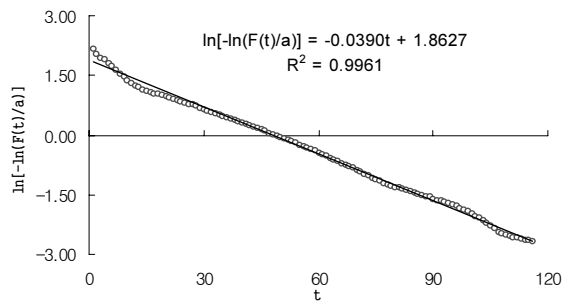
(a)  $F(t) = ab^t$  모델



(b)  $F(t) = ae^{bt}$  모델



(c)  $F(t) = at^b$  모델



(d) Gompertz 함수 모델

(그림 4) 제안된 모델의 모수추정 회귀분석

대표적인 4개 모델에 대해 성능 비교는 <표 3>에 제시하였다. 제안된 Gompertz 함수 모델이 기존의 제약모델들 보다 적게는 52%, 많게는 87%의 성능 향상을 보였다. 따라서 Gompertz 함수 모델이 소프트웨어 비용 추정 모델로 가장 적합하다고 판단된다.

<표 2> 모델 성능 평가

모델	회귀분석 관계	모델 성능	
		결정계수( $R^2$ )	MMRE
$F(t) = ab^t$	$\log[F(t)]$ vs. $t$	0.6764	9.16%
$F(t) = ae^{bt}$	$\ln[F(t)]$ vs. $\ln(t)$	0.9772	4.59%
$F(t) = at^b$	$\log[F(t)]$ vs. $\log(t)$	0.9772	3.94%
Gompertz 함수 $F(t) = ae^{-bc^{-t}}$	$\ln[-\ln(\frac{F(t)}{a})]$ vs. $t$	0.9961	2.83%

<표 3> 모델 성능 비교

모델	MMRE	성능 향상			
		Rayleigh 함수 대비	Gamma 함수 대비	Weibull 함수 대비	Gompertz 함수 대비
Rayleigh 함수	21.29%	-	-260.8%	-5.0%	-652.3%
Gamma 함수	5.90%	+72.3%	-	+70.9%	-108.5%
Weibull 함수	20.27%	+4.8%	-243.6%	-	-616.3%
Gompertz 함수	2.83%	+86.7%	+52.0%	+86.0%	-

### 5. 결론 및 향후 연구과제

본 논문에서는 소프트웨어 개발비용을 추정하는 분야를 연구하였다. 연구 대상은 개발에 소요되는 총 비용은 적절히 추정되었다고 가정하고, 총 비용을 개발 각 시점별로 적절히 분배하여 프로젝트를 성공적으로 완료할 수 있는 제약모델에 초점을 두었다.

기존에 제안된 제약모델들은 pdf로부터 변수변환을 거쳐 선형관계로부터 모수를 추정하는 방식을 따르고 있다. 반면에 제안된 모델은 CDF로부터 변수변환을 통해 모수를 추정하는 방식으로 70~80년대에 제안되었으며, 실제 프로젝트에 투입되는 단위시간당 인력의 양과 누적 인력의 양을 적절히 표현하지 못하는 단점을 갖고 있다. 실제 데이터를 대상으로 MMRE를 분석한 결과 제안된 Gompertz 함수 모델은 기존의 제약모델인 Rayleigh, Gamma와 Weibull 함수 모델들 보다 52%에서 87%의 성능향상을 나타내었다. 제안된 모델을 적용하여 보다 실제적으로 인력분배 계획을 작성하면 소프트웨어 개발 성공률을 높일 수 있을 것이다.

개발될 소프트웨어 뿐 아니라 개발조직의 특성에 따라서 프로젝트의 인력 분배 프로파일은 다양성을 보일 수 있다. 그러나 실제 수행 프로젝트의 인력 프로파일 데이터 획득의 어려움으로 다양한 인력 분배 프로파일에 대해 본 제안된 모델의 적합성은 검증하여 일반화된 모델로 제시하지는 못하였다, 따라서, 추후 이 분야에 대한 연구를 수행할 것이다.

### 참고 문헌

[1] N. E. Fenton and S. L. Pfleeger, "Software Metrics: A Rigorous and Practical Approach," International Thomson Computer Press, 1997.  
 [2] W. Boehm, "Software Engineering Economics," IEEE Trans. on Software Eng., Vol.10, No.1, pp.7-19, 1984.  
 [3] B. W. Boehm et al, "Software Cost Estimation with COCOMO II," Prantice-Hall, 2000.

[4] L. H. Putnam, "A General Empirical Solution to the Macro Software Sizing and Estimating Problem," IEEE Trans. on Software Eng., Vol.SE-4, No.4, pp.345-361, 1978.

[5] K. Pillai and V. S. Sukumaran Nair, "A Model for Software Development Effort and Cost Estimation," IEEE Trans. on Software Eng., Vol.23, No.8, pp.485-497, 1997.

[6] G. J. Knafl and C. Gonzales, "An Evaluation of Software Cost Models," <http://facweb.cs.depaul.edu/research/techreports/TR98-005.doc>, DePaul University, 1998.

[7] L. C. Briand, K. E. Eman, D. Surmann, I. Wiczorek, and K. D. Maxwell, "An Assessment and Comparison of Common Software Cost Estimation Modeling Techniques," International Software Engineering Research Network Technical Report, ISERN-98-27, 1998.

[8] Standish Group, "Extreme CHAOS," The Standish Group International, Inc., 2001.

[9] T. Foss, E. Stensrud, B. Kitchenham, and I. Myrtveit, "A Simulation Study of the Model Evaluation Criterion MMRE," IEEE Trans. on Software Eng., Vol.29, No.11, pp.985-995, 2003.

[10] R. D. H. Warburton, "Managing and Predicting the Costs of Real-Time Software," IEEE Trans. on Software Eng., Vol.SE-9, No.5, pp.562-569, 1983.

[11] C. Henry, "The Growth Curve," <http://www.anzpug.org/jsp/index.jsp>, PRIMAVERA Users Groups, Technology and Operations Management, California Polytechnic and State University, 2004.

[12] H. A. Kanter and T. J. Muscarello, "Learning (Experience) Curve Theory: A Tool for the Systems Development and Software Professional," Object-Z Systems Inc., 2001.

[13] B. Gompertz, "On The Nature of The Function Expressive of The Law of Human Mortality, and on a New Mode of Determining the Value of Life Contingencies," Phil. Trans. Roy. Soc. London. Vol. 123, pp.513-585, 1832.



### 이 상 운

e-mail : [sulee@gangnung.ac.kr](mailto:sulee@gangnung.ac.kr)

1983년~1987년 한국항공대학교 항공 전자공학과(학사)

1995년~1997년 경상대학교 컴퓨터과 학과(석사)

1998년~2001년 경상대학교 컴퓨터과학과 (박사)

1992년~2002년 국방품질관리소 항공전자장비 소프트웨어 품질보증 담당

2003년 강원도립대학 컴퓨터응용과 전임강사

2004년~2007년 2월 국립 원주대학 여성교양과 조교수

2007.3~현재 강릉대학교 컴퓨터정보공학부

멀티미디어정보공학 전공 조교수

관심분야: 소프트웨어 프로젝트 관리, 소프트웨어 개발 방법론, 소프트웨어 척도 (소프트웨어 규모, 개발노력, 개발 기간, 팀 규모), 분석과 설계 방법론, 소프트웨어 시험 및 품질보증, 소프트웨어 신뢰성, 신경망, 뉴로-퍼지, 알고리즘