論文
5-1-2

소프트웨어 개발주기 인력분포 모델

(A Manpower Distribution Model for Software Development Cycle)

박석규(Seok-Gyu Park)[1] 박중양(Joong-Yang Park)[2] 박재홍(JaeHeung Park)[3]

요 약

성공적인 프로젝트 계획은 활용 가능한 일정과 더불어 프로젝트를 개발하는데 요구되는 인력을 얼마나 정확히 추정하느냐에 달려있다. 현존하는 소프트웨어 인력 추정 모델들은 생명주기 전반에 걸쳐 투입되는 총 인력과 단위시간당 소요되는 인력 분포를 제공한다. Putnam의 Rayleigh 모델이나 Phillai et al.의 Gamma 모델 모두 소프트웨어 개발이 종료되는 시점에서 투입되는 인력이 절정에 도달한다는 가정하에 모델을 제시하였다. 이는 개발완료시점까지 40%의 인력이 투입되며, 나머지 60%는 유지보수 단계에 투입됨을 의미한다. 그러나 Warburton은 실제 프로젝트는 대부분의 인력이 개발에 투입되고 설계가 종료된 시점에서 투입인력 규모가 절정에 도달하며, 유지보수에는 단지 적은 규모의 인력만이 소요됨을 관찰하였다. 따라서, 제안된 기존 모델들은 실제 프로젝트에 적용하는데 문제가 발생한다. 이러한 문제점을 해결하기 위해 본 논문은 투입인력 절정 시점 모수를 고려하지 않는 시그모이드 모델을 제시하였다. 제시된 모델을 실제 데이터에 적용하여 향상된 성능을 보였다. 따라서, 제안된 시그모이드 모델은 소프트웨어 개발주기 동안 투입되는 인력 분포를 추정하는데 있어서 Rayleigh나 Gamma 모델의 대안으로 적용이 가능하다.

ABSTRACT

Successful project planning relies on a good estimation of the manpower required to complete a project in addition to the schedule options that may be available. Existing software manpower estimation models present the total manpower and instantaneous manpower distribution for the software life cycle. Putnam's Rayleigh and Phillai et al.'s Gamma models present a model with assumption that the manpower is needed at the system delivery. This means that 40 percent of total manpower is applied at the software development, and the other 60% is applied during maintenance phase. However Warburton observes the manpower is needed during development phase with the peak at the completion of the software design phase. So, the existing models were not appropriate to be applied to practical projects. This paper suggests the Sigmoid model which does not consider the point of manpower peak to fix the problem above. The suggested model showed some improvement when practical data was applied. Therefore, the Sigmoid model can be used as alternative of Rayleigh and Gamma model to estimate distribution of manpower during software development phas

1) 정회원 : 경상대학교 대학원 컴퓨터과학과 박사과정
2) 정회원 : 경상대학교 통계정보학과 교수
3) 정회원 : 경상대학교 컴퓨터과학과 교수

논문접수 : 2003. 12. 23.
심사완료 : 2004. 1. 9.

1. 서 론

소프트웨어 프로젝트 계획단계에서 개발에 소요되는 총 인력 뿐만 아니라 시간에 따른 개발인력 투입 규모를 추정하는 것이 프로젝트를 관리할 때 발생되는 다양한 의사결정, 소요 예산 및 개발인원 할당과 계약체결 여부에 신뢰할 만한 정보를 제공한다.소프트웨어 개발 인력 분포에 대한 대표적인 연구로는 Putnam[1]의 SLIM (Software LIfecycle Management) 모델인 Rayleigh 모델과 Phillai et al.[2]의 Gamma 모델이 있다. 그러나 이들 모델들은 투입되는 인력 규모가 절정에 도달하는 시점은 총 투입 인력의 약 40% 해당하는 개발 종료시점으로 가정하였으며, 나머지 60%는 유지보수 단계에 소요됨을 고려하여 모델을 제시하였다. 이에 대해 Warburton[3]은 실제 프로젝트에 투입된 인력 분포를 관찰한 결과 대부분이 개발에 투입되고 약간의 인력이 유지보수 단계에 투입됨을 제시하였다. 또한, 투입인력의 규모가 절정에 도달하는 시점은 설계 종료시점이었다. 이와 같은 실제 시스템 개발에 소요되는 인력을 추정하기 위해서는 Putnam[1]의 Rayleigh 모델이나 Phillai et al.[2]의 Gamma 모델을 적용하면 큰 편차를 보일 수 있다. 또한, 사업 초기에 추정한 개발기간과 인력 규모 한도 내에서 개발을 종료하는 경우가 거의 없으므로, 투입 인력이 절정에 도달하는 개발 종료시점을 결정하기가 어려운 것이 현실이다. 이의 예로, 대형 소프트웨어 프로젝트의 1%만이 계획된 기간과 예상 비용한도 내에서 고객을 만족시키며 완료되었는데 반해, 대부분의 프로젝트들은 1년 이상의 일정이 지연되고 초기 예상 비용의 2배 정도가 초과되었다.[4] 이와 같은 이유로 인해, 프로젝트 관리 측면에서 소프트웨어 개발 및 유지보수 비용을 줄이고자 체계적인 연구가 수행되고 있으며, 소프트웨어 비용산정 및 프로젝트의 일정관리 모델을 개발하는 계기가 되었다.

따라서, 본 논문은 실제 개발된 프로젝트들의 정보를 근거로 개발 종료시점에서 인력이 최대로 소요된다는 Putnam의 가정을 고려하지 않고 개발 과정에서 인력이 최대로 소요되며, 유지보수 단계부터 폐기까지는 개발단계와 별도로 구별되어 소프트웨어 프로젝트가 진행된다고 가정한다. 이 가정에 근거하여 개발단계만을 고려할 경우, 인력이 절정에 도달하는 시점을 예측할 수 없으므로, 이 모수를 사용하지 않는 소프트웨어 프로젝트 소요 인력 분포를 추정하는 모델을 제시한다. 이 모델은 소프트웨어의 유지보수에서 폐기단계 까지를 고려하지 않고 단지 개발단계만을 고려하였으므로 SDCM (Software Development Cycle Management) 모델이라 칭한다.

2장에서는 기존의 개발인력 추정과 관련된 모델들과 문제점을 살펴본다. 3장에서는 기존 제안된 모델들의 문제점을 보완할 수 있는 시그모이드 모델을 제시한다. 이어서, 4장에서는 제안된 모델을 실제 데이터를 이용하여 적용해 보고, 기존 제안된 모델들과 성능을 평가하여 모델의 적합성을 제시한다.

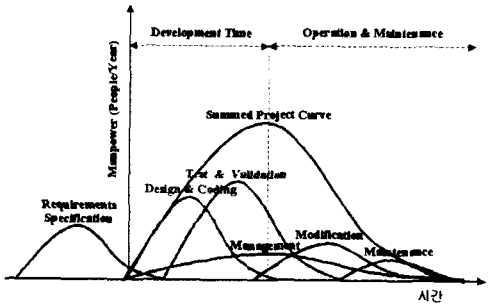
2. 기존 연구 및 문제점

2.1 기존 연구

소프트웨어 생명주기 (Life Cycle) 전반에 걸쳐 투입되는 인력에 관한 연구로 Putnam[1], Boehm[5], Nordan[6,7], Warburton[3]과 Pillai et al[2] 등이 있다.

Norden[6]은 IBM에서 개발된 다양한 하드웨어 개발과정에 투입된 노력은 근사적으로 Rayleigh 분포를 따른다는 사실을 관찰하였다. 이후 Putnam[1]에 의해 이 관찰 결과가 소프트웨어 프로젝트에 적용되었다. 150개의 소프트웨어 프로젝트가 Norden[7]과 Putnam[1]에 의해 연구되었으며, Nordon[7]은 50개의 Computer System Command 프로젝트에 적용한 결과 Rayleigh 분포를 잘 따름을 검증하였다. 1976년과 1977년에 개발된 다른 시스템들로부터 획득된 150개의 데이터에 대해 적용한 결과 모든 시스템이 이 패턴을 따르지는 않지만 다수의 프로젝트가 시간에 따른 동일한 기본적인 인력 패턴인 상승, 절정과 점차적으로 지수적 감소 현상을 보였다. 반면에 어떤 인력 패턴은 거의 직사각형 형태를 따르고 있었다. 따라서, 프로젝트 수준에서 소프트웨어 생명주기 전반 (계획, 요구명세 분석, 설계, 코딩, 시험과 유지보수 중 계획과 요구사항 분석단계 제외)에 걸쳐 소요되는 총 인력뿐만 아니라 생명주기의 각 단계 (즉, 설계, 코딩, 시험 및 유지보수)에 대한 인력도 [그림 1]과 같이 Rayleigh 분포를 따름을 관찰하였다. 임의의 시간 t 시점까지 프로젝트에 투입된 누적 인력 y를 식 (1)로, 임의의 시간 t 시점에서 프로젝트에 투입

되는 인력 $\dot{y}$는 식 (1)을 미분한 식 (2)로 표현된다.



[그림 1] 소프트웨어 개발인력 분포
[Fig. 1] Manpower Distribution of Software Development

$$y = E(1 - e^{-at^2}) \text{ MY} \quad (1)$$

$$\dot{y} = 2Eate^{-at^2} \text{ MY/YR} \quad (2)$$

여기서 E는 Rayleigh 곡선 아래에 있는 영역으로 년 인원으로 표시되며, 소프트웨어 생명주기 전반에 걸쳐 투입된 총 인력이다. $a = \frac{1}{2t_d^2}$ 이며, t_d는 $\dot{y}$이 최대가 되는 시점으로 경험적으로 볼 때, 시스템이 운영되는 시점에 근접한다. 따라서, $t_{\dot{y}_{max}} = t_d$를 시스템의 개발기간이라 하며, 소프트웨어 생명주기 전반에 걸쳐 총 소요되는 개발노력의 약 40%에 도달하는 시점이다. 식 (2)에서 $a = \frac{1}{2t_d^2}$를 치환하면, 식 (3)이 된다.

$$\dot{y} = \frac{E}{t_d^2} te^{-\frac{t^2}{2t_d^2}} \quad (3)$$

Putnam[1]의 Rayleigh 인력 분포는 총 예산의 약 40%에서 절정을 보이며 이 시점은 첫 번째 시스템이 납품되는 시점이다. 이것은 총 프로젝트 경비의 60%를 이루고 있는 장시간의 변경과 성능개선 활동이 수행되는 유지보수 단계가 뒤따름을 의미한다. Warburton[3]은 Rayleigh 분포 모델에서 투입인력의 절정을 결정하는 시점 (즉, 개발 완료 시점)은 불충분한 가정이라고 제기하였다. 왜냐하면, Warburton[3]은 데이터 처리 프로그램의 생명주기는 이러한 형태로 장시간의 변경과 성능개선이 수행되는 것이 사실이라는 것을 경험하였다. 예로, 은행의 대차대조표 프로그램은 최종 형식을 아직 사용하기에는 불충분하지만 첫 번째 프로그램이 운영됨을 발견하였다. 은행 대차대조표 형식은 월 단위로 변경되어 6개월에 걸쳐 보다 복잡한 형태로 발전되었으며, 몇 개월 후 확정되었다. 또한, 은행 조건이 변경되어 여러 번의 프로그램 변경이 진행되었다. 이 경우는 소프트웨어가 엄격한 요구사항 명세서와 일치하지 않으면 납품이 되지 않는 실시간 처리 전술적 내장 프로그램 (Tactical Embedded Program)과 대조된다. 전술적 내장 프로그램은 프로젝트 개발이 종료된 시점이 납품 시점이 된다. 즉, 납품될 때까지 모든 개발 작업이 종료되고 단지 약간의 유지보수 요원만이 남아있게 된다. 그러나 내장 소프트웨어의 시험단계에서 투입되는 인력은 증가, 절정과 점차 길게 감소하는 분포를 따름을 관찰하였다. 따라서, 실시간 소프트웨어 개발과정에 적합한 모델링을 위해 Putnam[1]의 Rayleigh 모델을 변경하는 것이 필요함을 제기하였다.

Boehm[5]은 소프트웨어 개발 초기에 느린 개발인력 형성 부분과 마지막 단계에서의 점차 길게 감소하는 Rayleigh 분포 특성이 대부분의 소프트웨어 프로젝트 개발인력 분포에 일치하지 않음을 지적하였다. 일반적으로 소프트웨어 프로젝트는 개발 초기에 하드웨어 프로젝트보다 빠른 형성 비율 (Buildup Rate)을 가지며, 이는 Rayleigh 분포와 편향된 결과를 나타낸다.

Rayleigh 곡선을 비선형 회귀분석으로 주어진 데이터로 적합 (Fitting) 시키는 것은 어려운 과정이므로 Putnam[1]은 식 (2)에 대해 자연로그 변환을 통해 식 (4)의 선형으로 변환시켜 모수를 추정하였다.

$$\ln\left(\frac{\dot{y}}{t}\right) = \ln\left(\frac{K}{t_d^2}\right) + \left(-\frac{1}{2t_d^2}\right)t^2 \quad (4)$$

$\ln\left(\frac{\dot{y}}{t}\right)$를 y, t^2을 x로 치환하면 식 (4)는 식 (5)의 단순선형형태가 된다.

$$y = mx + c \quad (5)$$

여기서, $m = -\frac{1}{2t_d^2}$ 은 기울기, $c = \ln\left(\frac{K}{t_d^2}\right)$는 절

편이다. $m=-\frac{1}{2t_d^2}$ 로부터 $t_d=\sqrt{-\frac{1}{2m}}$ 을 얻는다. 그러므로 적합된 직선의 기울기 m이 0 이거나 양수이면 t_d는 실질적인 값을 가지지 못하며, 선형회귀 추정은 기울기 m의 값이 0가 아닌 음수인 경우에만 의미 있는 값을 얻을 수 있다. 따라서, 프로젝트에 투입되는 인력이 절정에 도달할 때까지 증가하는 현상을 보이므로 선형회귀모델의 기울기 m이 양수 값을 가진다. 이에 대해 Warburton[3]은 Putnam[1]의 Rayleigh 모델은 인력이 절정에 도달한 이후에야 모수 t_d와 K 값을 추정할 수 있음을 제기하였다. 이는 약 40%의 인력이 투입되어 개발이 종료되고 유지보수 단계에 도달하였을 때 비로소 투입인력을 추정할 수 있으며, 개발단계에서는 불가능함을 의미한다.

이의 연구 결과를 토대로, Pillai et al.[2]은 Putnam[1]의 개발노력 분포가 Rayleigh 분포를 따르는데 따른 문제점을 해결하기 위해 $\ln(\frac{\dot{y}}{t})$ 대신 $\ln(\frac{\dot{y}}{t^2})=\gamma+\mu t$ 변환을 통해

$$\dot{y}=e^{\gamma}t^2e^{-\mu t}=\frac{1}{\Gamma(\alpha)\beta^{\alpha}}t^{\alpha-1}e^{-\frac{t}{\beta}},\ \alpha=3,\ \beta=\frac{t_d}{2}$$

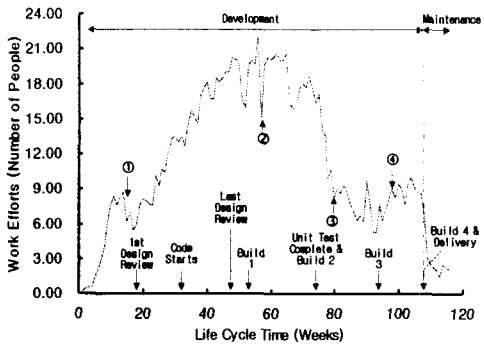
로 치환하여 식 (6)의 Gamma 모델을 제시하였다.

$$\dot{y}=\frac{8E}{\Gamma(3)t_d^3}(t^2e^{-\frac{2}{t_d}t})=\frac{4E}{t_d^3}t^2e^{-\frac{2}{t_d}t} \quad (6)$$

개발단계에서 모수 추정이 불가한 Putnam[1]의 Rayleigh 모델의 단점을 보완한 Gamma 모델조차도 t_d (즉, 개발 완료 시점에서 투입인력이 절정에 도달한다.)를 모수로 사용하고 있음을 알 수 있다.

2.2 문제점

Phillai et al.[2]이 Gamma 모델을 제안하면서 적용한 데이터는 Warburton[3]이 제시한 [그림 2] 데이터이다.



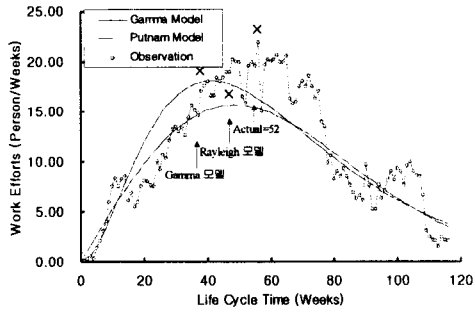
[그림 2] 소프트웨어 생명주기 투입인력 사례
[Fig. 2] Example of Manpower Distribution for Software Development

[그림 2] 데이터는 소나와 화력제어계통 (Sonar and Fire Control System)의 훈련/시뮬레이터 프로젝트로 요구사항 명세화, 설계, 코딩, 시험, 시스템 통합과 납품을 포함한 개발 전 과정에 투입된 인력분포이다. 이 시스템은 116주의 기간동안 1300.8명(Man/Year)이 투입되었으며, 7일 이상의 MTTF (Mean Time To Failure) 품질수준이 달성되었을 때 개발이 종료되고 고객에게 납품 (Delivery) 되었다. Putnam[1]이 제시한 [그림 1]은 첫 번째 납품이 된 시점에서 인력이 절정을 이루고 있으나 [그림 2]의 실제 데이터에서는 소프트웨어 설계 종료시점에서 발생하였으며, 여전히 Rayleigh 분포는 따르고 있음을 알 수 있다. 프로젝트는 검토를 포함한 Top-Down 설계, Walkthrough를 포함한 구조적 코딩, 단위시험, 구축과 엄격한 형상관리가 이루어졌으며, 설계의 모든 부분이 동시에 개발되지 않고 중요한 모듈과 주요 구성품이 먼저 개발되었다. 코딩은 공식 설계검토 (Formal Design Review)가 승인된 후에 시작되었다. 어떤 모듈들은 다른 모듈들이 코딩되는 동안 설계가 이루어졌으며, 18에서 20명의 엔지니어가 개발단계를 중첩하여 업무를 수행하였고 엄격한 형상관리와 공식 검토 절차가 필수적으로 수행되었다.

[그림 2]에서 인력 변동 이유를 살펴보면, ① 시점은 고객의 자금문제 발생, ② 시점은 겨울의 휴가와 눈보라로 인한 문제, ③ 시점은 자금 고갈에 대한 심리학적 함정, ④ 시점은 이전에 구축된 프로그램의 오

류 수정으로 인해 수반되는 재 구축 문제로 인해 발생하였다. 기존에 제안된 모델들의 문제점을 살펴보자.

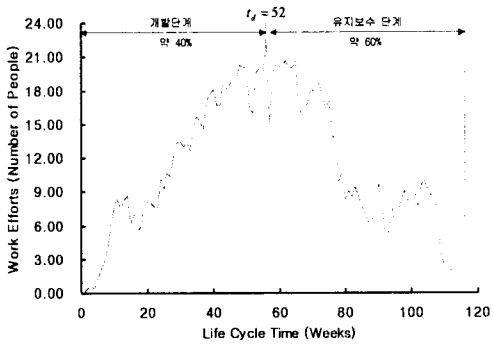
첫 번째로, t_d가 개발 종료시점 ($t_d = 106$ Weeks)이라고 가정하지 않고 개발중 노력이 최대로 소요되는 시점 ($t_d = 52$ Weeks)이라고 가정하였을 경우, 실제 데이터에 Putnam[1]의 Rayleigh 모델과 Phillai et al.[2]의 Gamma 모델을 적용하여 보자. Phillai et al.[2]은 t_d 모수에 대해 Putnam[1]의 Rayleigh 모델은 $t_d = 48.4379$, Gamma 모델은 $t_d = 41.228$로 추정하였다. 이의 결과를 반영하여 2개 모델을 실제 데이터에 적용하면 [그림 3]의 결과를 얻는다.[2]



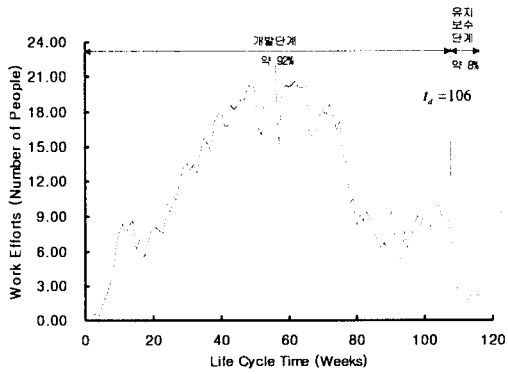
[그림 3] 실제의 투입인력 절정시점 ($t_d = 52$ Weeks)을 t_d로 가정시 Rayleigh와 Gamma 모델 적용 결과
[Fig. 3] Applied Results of Rayleigh and Gamma Model in accordance with t_d is assumed to Actual Manpower Peak

[그림 3]에서 Putnam[1]의 Rayleigh 모델의 단점을 보완한 Phillai et al.[2]의 Gamma 모델이 이론적으로는 Putnam[1]의 Rayleigh 모델보다 타당하나 실제로 인력이 절정을 이루는 시점이 보다 좌측으로 이동되어 나쁜 결과를 얻음을 알 수 있다. 따라서, 투입인력 절정 시점을 Putnam[1]의 Rayleigh 모델에 비해 보다 우측으로 이동시키는 작업이 필요하다. 또한, 2개 모델 모두 실제의 t_d 시점을 적절히 표현하지 못하는 단점을 갖고 있다.

두 번째로, [그림 1]의 Putnam[1]이 제시한 개발 완료시점 (t_d)에서 투입인력의 규모가 절정 (Peak)에 도달한다는 가정에 기반할 때, [그림 2]의 실제 데이터에 적용하면 [그림 4]의 개발과 유지보수 기간이 [그림 5]로 변경되어야만 한다.



[그림 4] 실제 투입인력 절정시점 ($t_d = 52$ Weeks 기준)과 오적 용된 t_d
[Fig. 4] Actual Manpower Peak (*Based on* $t_d = 52$ Weeks) and Misapplied t_d

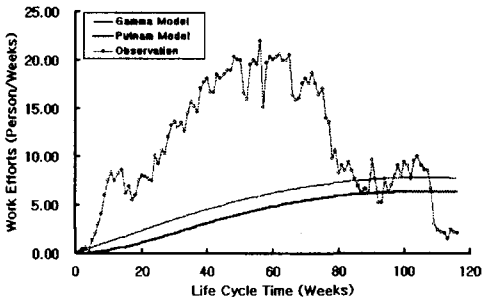


[그림 5] 실제의 개발과 유지보수 기간 ($t_d = 106$ Weeks 기준)
[Fig. 5] Actual Development and Maintenance Period (*Based on* $t_d = 106$ Weeks)

[그림 5]의 결과를 반영하여 Putnam[1]의 Rayleigh 모델과 Phillai et al.[2]의 Gamma 모델의 모수를 추정하여 적합시킨 결과는 [그림 6]과 같다. 그림에서 2

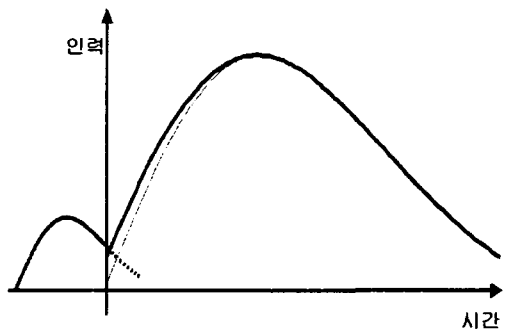
개 모델 모두 실제 인력 분포를 전혀 표현하지 못하는 결과는 나타내었다. 또한 실제 개발중에 인력이 절정을 이루는 시점을 정확히 추정하기 어려운 점이 있다.

끝으로, Putnam[1]의 Rayleigh 모델과 Phillai et al.[2]의 Gamma 모델은 계획과 요구사항 분석 관련 인력을 고려하지 않았다. 이를 고려하면 [그림 1]은 설계 시작시점에서 인력의 절편을 가진 [그림 7]과 같이 변경되어야 한다.



[그림 6] 투입인력 절정시점 (개발 완료 시점, t_d= 106 Weeks)고려시 Rayleigh와 Gamma 모델 적용 결과

[Fig. 6] Applied Results of Rayleigh and Gamma Model Considered Manpower Peak Time (Development Complete Time, t_d= 106 Weeks)



[그림 7] 요구사항 분석 인력을 고려한 설계 시작시점의 투입 인력

(점선 : Putnam's Rayleigh 모델, 실선 : 실제 투입되는 인력)

[Fig. 7]Manpower of Initial Design Time Take Into Account Requirement Analysis Manpower

따라서, t_d를 고려하지 않으면서, 개발 초기에 투입 인력 규모가 0에 가깝지 않고 절편을 가지는 모델 개발이 필요하다.

3. 시그모이드 소프트웨어 개발인력 분포 모델

3.1 모델 제안

생명주기의 초기에 개발에 소요되는 인력 추정시 개발 종료 시점인 t_d값은 알 수 있지만 실제로 개발 중 최대의 인력이 투입되는 시점은 알지 못한다. 따라서, 제안 모델은 다음의 가정에 근거하여 모델을 개발하고자 한다.

가정 1. 투입인력이 절정에 도달하는 시점은 개발 완료 시점이 아니며, 개발 과정 중에 발생한다.

가정 2. 설계 시작 시점부터 개발인력을 추정하는 모델이 아니라 계획과 요구사항 분석 단계도 포함한 소프트웨어 생명주기 전체에 대한 인력분포를 표현한다.

가정 3. 계획과 요구사항 분석단계 까지 고려함으로써, 시작 시점에서 투입인력이 0에 가까운 값이 아니고 어느 정도의 절편 값을 갖는다.

위에 제시된 가정들을 모두 만족시킬 수 있는 함수로 식(7)의 시그모이드 함수 (Sigmoid Function)[8]를 적용할 수 있다. 시그모이드 함수는 그리스 문자인 Sigma 글자와 같은 형태를 갖고 있어 불리워진 함수로 신경망에서 뉴론의 입력과 출력간의 비선형성 정도를 표현하는데 사용되고 있다.[9]

$$y=\frac{1}{1+e^{-\alpha t}},\quad \dot{y}=\frac{e^{-\alpha t}}{(1+e^{-\alpha t})^2} \tag{7}$$

식 (7)의 형태를 취하는 식 (8)의 모델을 적용하여 시간에 따른 투입인력을 추정한다. 여기서 a와 b는 상수이다. 본 제안 모델을 "시그모이드 모델"이라 칭한다.

$$\dot{y}=\frac{Ke^{-\alpha}}{(1+be^{-\alpha})^2},\quad \alpha=\frac{t-a}{b},\quad a>0,\ b>0 \tag{8}$$

3.2 모수 추정

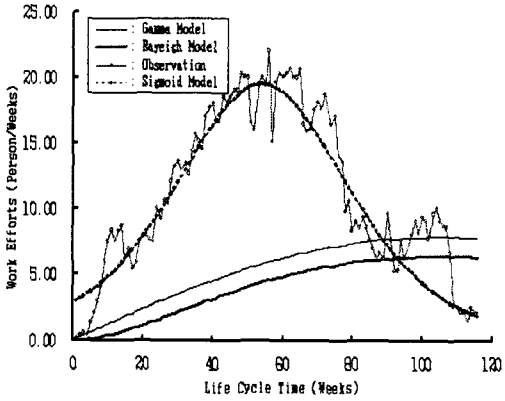
시스템 개발이 t_n 시점까지 진행되었다고 가정하자. 각 개발시간에서의 투입인력 $\dot{y}_i$, $y_i = \sum_{i=1}^{n} \dot{y}_i$, $i = 1, 2, \cdots, n$ 데이터를 얻을 수 있다. 현재 시점에서 투입된 인력 추정과 $t_{n+d}.(d \geq 1)$ 시점에서 투입되어야 하는 인력을 예측하기 위해서는 먼저 모델에 있는 모수를 추정해야만 한다. 모수를 추정하는 방법들 중에서 최우추정법 (Maximum Likelihood Estimation Method, MLE)을 많이 사용하고 있으며 이의 대안으로 최소자승법 (Least Squares Method)도 많이 적용하고 있다.[10] 최소자승 추정치는 식 (9)를 최소화시킴으로써 얻어진다.[11] 여기서, $\overline{y_i}$는 y_i의 평균값이다.

$$\sum_{i=1}^{n} [y_i - \overline{y_i}]^2 \qquad (9)$$

4. 모델 평가

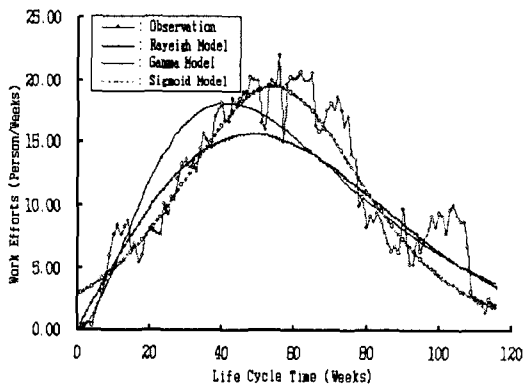
4.1 실제 데이터에 적용

[그림 2] 데이터에 대해, 식 (8)의 시그모이드 모델의 모수 추정 결과 $\hat{a} = 7.014$, $\hat{b} = 16.711$을 얻었으며, 이를 실제 데이터에 적용하여 투입인력을 적합시킨 결과는 [그림 8]과 [그림 9]에 제시되어 있다. [그림 8]에서 프로젝트의 실제 개발이 완료된 시점을 기준으로 했을 경우, Rayleigh나 Gamma 모델은 부적합한 결과를 보인 반면, 제안된 시그모이드 모델은 인력 분포를 거의 정확히 표현할 수 있음을 알 수 있다. 또한, [그림 9]에서 투입인력 절정 시점을 비교시 Rayleigh와 Gamma 모델은 인력이 절정에 도달하는 시점을 부정확하게 추정한 반면, 제안된 시그모이드 함수는 거의 정확히 추정하였다.



[그림 8] 실제 개발 완료시점 ($t_d = 106$ Weeks) 고려시 인력 추정

[Fig. 8] Manpower Estimation Based on Actual Development Complete Time ($t_d = 106$ Weeks)



[그림 9] 투입인력 절정 시점 ($t_d = 52$ Weeks) 고려시 인력 추정

[Fig. 9] Manpower Estimation Based on Manpower Peak Time ($t_d = 106$ Weeks)

4.2 모델 평가

다른 여러 가지 모델들을 비교하는데 있어, 어떤 의미 있는 척도로서 모델의 추정 정확도를 평가하는 것이 필요하다. 회귀분석의 경우 회귀직선에 의해 종속변수가 설명되는 정도를 결정계수(Coefficient of determination, R^2)라 한다.[12] 종속변수의 값은 독립변수 에 의해 결정되는 부분과 미지의 오차의 합으로 나타나며, 총 변동 중에서 회귀직선에 의해 설명되는 비율이 결정계수이다. 따라서,

$R^2(0 \le R^2 \le 1)$이 0에 가까우면 추정된 회귀직선은 쓸모가 없으며, 값이 클수록 쓸모 있는 회귀직선이 된다. *MMRE*를 살펴보자. 상대오차 (Relative Error, *RE*)는 $RE = \frac{(\text{실측치} - \text{추정치})}{\text{실측치}} \times 100(\%)$로 구해지며, *MRE* (Magnitude of the RE)는 $|RE|$이며, n개의 데이터에 대한 *MMRE* (Mean *MRE*)는 $\frac{1}{n}\sum_i MRE_i,\ i = 1, 2, \cdots, n$로 계산된다. *MMRE*가 작은 값이면 좋은 모델임을 알 수 있다.[13,14]

따라서, 제안된 모델을 평가하는 기준으로 R^2와 *MMRE*를 고려하였다. 이들 평가기준에 의해 제안된 모델과 기존 모델의 성능은 <표 1>과 <표 2>에 제시하였다.

모델	R^2 (순위)	
	$t_d = 106$인 경우	$t_d = 52$인 경우
Putnam의 Rayleigh 모델	0.0133(3)	0.8344(2)
Phillai et al.의 Gamma 모델	0.0245(2)	0.7360(3)
시그모이드 모델	0.8733(1)	0.8733(1)

<표 1> 모델 성능 비교 (R^2)

<Table 1> Compare of Model Performance (R^2)

<표 1>에서 개발 완료시점 ($t_d = 106$)에 투입인력이 최대가 된다는 가정을 고려한 경우와 실제 개발인력이 최대로 소요되는 $t_d = 52$를 고려한 경우 모두 제안된 시그모이드 모델이 가장 좋은 결과를 얻었으며, 실제 데이터의 인력 변동의 87.33%를 설명하는 능력을 갖고 있다.

모델	*MMRE* (순위)	
	$t_d = 106$인 경우	$t_d = 52$인 경우
Putnam의 Rayleigh 모델	66.69%(2)	33.94%(2)
Phillai et al.의 Gamma 모델	72.87%(3)	50.62%(3)
시그모이드 모델	47.37%(1)	47.37%(3)

<표 2> 모델 성능 비교 (*MMRE*)

<Table 2> Compare of Model Performance (*MMRE*)

<표 2>에서 개발 완료시점 ($t_d = 106$)에 투입인력이 최대가 된다는 가정을 고려한 경우 제안된 모델의 성능이 가장 좋은 결과를 보였으나, 실제 개발인력이 최대로 소요되는 $t_d = 52$를 고려한 경우는 가장 나쁜 결과를 보였다. 이는 개발 초기의 $1 \le t \le 7$ 기간에서 제안된 모델이 실제 투입 인력을 과다 추정한 결과로 인해 발생하였으며, 이 기간을 제외한 나머지 기간에서는 <표 3>과 같이 *MMRE*가 가장 좋은 결과를 보였다.

모델	*MMRE* (순위)	
	$t_d = 106$인 경우	$t_d = 52$인 경우
Putnam의 Rayleigh 모델	67.93%(2)	30.06%(3)
Phillai et al.의 Gamma 모델	71.56%(3)	26.11%(2)
시그모이드 모델	17.25%(1)	17.25%(1)

<표 3> 모델 성능 비교 ($1 \le t \le 7$을 제외한 기간에서의 *MMRE*)

<Table 3> Compare of Model Performance (*MMRE* excepted for $1 \le t \le 7$ Interval)

따라서, 제안된 시그모이드 모델이 소프트웨어 생명주기 전반에 걸쳐 소요되는 인력을 추정하는 모델로서 적합하며, 또한, 기존의 Rayleigh 모델이나 Gamma 모델의 대안으로 적용할 수 있음을 알 수 있다.

5. 결론 및 향후 연구과제

본 논문은 소프트웨어 생명주기 전체에 걸쳐 투입되는 총 인력의 규모와 단위시간에 투입되는 인력을 추정하는 분야에 관한 연구를 수행하였다.

소프트웨어 생명주기 투입인력 분포에 대한 대표적인 연구로는 Putnam[1]의 Rayleigh 분포 모델과 Phillai et al.[2]의 Gamma 분포 모델이 있다. 그러나 이 모델들은 투입 인력이 절정에 도달하는 시점을 고려하여 모델을 제시하였다. 제시된 모델에서 투입되는 인력이 절정에 도달하는 시점은 개발이 종료되고 고객에게 소프트웨어를 납품하는 시점을 의미한다. 이 시점까지 투입된 인력은 총 투입 인력의 약 40% 정도임을 밝혔다. 따라서, 나머지 60%는 변경과 성능 향상을 수행하는 유지보수 단계에 투입됨을 의미한

다. 이는 1970년대에 체계적인 개발방식을 채택하지 않은 시점에서 개발된 프로젝트들을 대상으로 Putnam[1]이 관찰한 결과로 통계적 이론에 기반한 모델이 아니다. 체계적인 개발기법을 사용한 프로젝트들이 실제 투입인력이 최대가 되는 시점은 프로젝트를 납품하는 시점이 되지 않는다. 실제로 수행된 프로젝트를 살펴보면, 대부분의 인력이 개발단계에 투입되고 단지 약간의 인력이 유지보수 단계에 투입됨을 Warburton[3]은 관찰하였다. 따라서, 투입인력이 절정에 도달하는 시점은 납품시점이 아니라 소프트웨어 개발 기간중에 발생한다. 이와같은 이유로 인해, 투입인력이 절정에 도달하는 납품시점에 대한 모수를 사용하는 기존의 Rayleigh와 Gamma 모델은 실제 프로젝트의 투입인력을 추정하는데 문제점이 발생한다. 또한, 소프트웨어 개발 종료시점을 추정하는 것은 어려운 문제이며, 거의 대부분의 프로젝트에서 정확한 추정을 할 수 없는 것이 현실이다.

따라서, 본 논문은 소프트웨어 투입인력을 추정하는 모델을 제시함에 있어, 기존 모델들의 단점을 보완하기 위해 투입인력이 절정에 도달하는 시점을 모수로 사용하지 않는 시그모이드 모델을 제시하였다. 제시된 모델을 실제 프로젝트에 적용한 결과 기존의 Rayleigh나 Gamma 모델들 보다 적합함을 보였으며, 투입인력이 절정에 도달하는 시점도 거의 정확히 추정함을 보였다.

본 논문은 하나의 실제 프로젝트를 대상으로 모델의 적합성을 보였다. 그러나 실제로는 다양한 투입인력 패턴을 갖는 것이 현실적이므로 다양한 프로젝트 데이터에 적용하여 보다 일반화된 모델 적합성을 보이는 것이 필요하다. 따라서, 추후 이 분야에 대한 연구를 수행할 것이다.

참고 문헌

[1] 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, 1978.

[2] 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.

[3] 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.

[4] K. H. Möller and D. J. Paulish, "Software Metrics - A Practitioner s Guide to Improved Product Development", Chapman & Hall Co., New York, 1993.

[5] B. W. Boehm, "Software Engineering Economics," Prentice Hall, 1981.

[6] P. V. Norden, "Curve Fitting for a Model of Applied Research and Development Scheduling," IBM J. Research and Development, Vol. 3, No. 2, pp. 232-248, 1958.

[7] P. V. Norden, "Project Life Cycle Modeling : Background and Application of the Life Cycle Curves," U. S. Army Computer System Command, 1977.

[8] M. T. Hargan, H. B. Demuth, and M. Beale, "Neural Network Design," PWS Publishing Company, 1996.

[9] http://users.computerweekly.net/robmorton/projects/neural/sogmoid.html

[10] 박중양, 유창열, 이부권, "초기하분포 소프트웨어 신뢰성 성장 모델에서의 모수 추정과 예측 방법," 정보처리학회 논문지, 제 5권, 제9호, pp. 2345-2352, 1998.

[11] Y. Thoma, K. Tokunaga, S. Nagase and Y. Murata, "Structural Approach to the Estimation of the Number of Residual Software Faults Based on the Hyper-Geometric Distribution," IEEE Trans. on Software Eng., Vol. 15, No. 3, pp. 345-355, 1989.

[12] 김우철 et al., "현대통계학," 영지출판사, 1994.

[13] S. D. Conte, H. E. Dunsmore, and V. Y. Shen, "Software Engineering Metrics and Models," Menlo Park, CA: Benjamin Cummings, 1986.

[14] J. Verner and G. Tate, "A Software Size Model," IEEE Trans. on Software Eng., Vol. 18, No. 4, pp265-278, 1992.

박 석 규

1988년 4월 ~ 2001년 2월
진주산업대학교 전산실장
1990년 3월 ~ 1992년 8월
경남대학교 컴퓨터공학과 석사
2001년 3월 ~ 현재 강원도립대학 컴퓨터응용과 조교수
관심분야 : 소프트웨어공학, 시스템 분석 및 설계, 멀티미디어

박 중 양

1982년 연세대학교 상경대학 응용통계학과(학사)
1984년 한국과학기술원 산업공학과 응용통계전공(석사)
1994년 한국과학기술원 산업공학과 응용통계전공(박사)
1984년-1989년 경상대학교 전산통계학과 교수
1989년-현재 경상대학교 통계정보학과 교수, 컴퓨터정보통신연구소 연구원
관심분야 : 소프트웨어 신뢰성, 신경망, 선형통계 모형, 실험 계획법 등

박 재 홍

1978년 충북대학교 수학교육과(학사)
1980년 중앙대학교 대학원 전산학과(석사)
1988년 중앙대학교 대학원 전산학과(박사)
1983년-현재 경상대학교 컴퓨터과학과 교수, 컴퓨터정보통신연구소 연구원
관심분야 : 소프트웨어 신뢰성, 시험도구 자동화, 시스템 분석 및 설계, 신