

재작업과 불확실성을 고려한 일정 시뮬레이션 방법론

A Method for Schedule Simulation Considering Rework and Uncertainty

김찬목[†] · 박영원*

Chan-Mook Kim · Young-Won Park

Abstract The majority of development projects fail to meet their target schedule and cost, with the overrun typically between 40 and 200 percent. These overruns happen because the planners underestimate the work or do not consider the need to rework at project planning. Representative schedule planning/management techniques such as Gantt Chart, PERT/CPM etc. that are used in domestic project planning are unable to reflect rework. This paper proposes a method to consider rework to provide more realistic estimates at schedule planning. Additionally, to prevent the underestimation of the work this paper proposes a simulation method that calculates a probabilistic estimated schedule and the associated variance based on the random variable modeling of individual task completion dates.

Keywords : Schedule Planning, Schedule Simulation, PERT/CPM, DSM(Design Structure Matrix)

요 지 대부분의 개발 프로젝트는 목표 비용과 일정을 40~200% 초과한다. 이러한 초과 현상은 계획시 업무를 과소평가하거나 재작업을 고려하지 않음으로써 발생한다. 현재 국내에서 사용되는 대표적인 일정 계획/관리 기법(Gantt Chart, PERT/CPM 등)들은 재작업을 반영할 수 없다. 이 논문은 일정 계획시 보다 현실적인 일정 계획 값을 예측하기 위해서 재작업을 고려하는 방법을 고안하고, 또한 과업의 과소평가를 방지하기 위해 수행시간의 불확실성을 확률변수로 나타내어 다루기 위해서 시뮬레이션을 수행하여 최종 완료 날짜의 예상치와 분산도를 계산하는 방법을 제안한다.

주 요 어 : 일정 계획, 일정 시뮬레이션, PERT/CPM, 몬테카를로 시뮬레이션, 설계 구조 매트릭스

1. 서 론

시스템이 거대해지고 복잡해짐에 따라 하나의 프로젝트에 보다 많은 과업들이 참여하게 되고 이에 따라 과학적으로 계획하고 관리해야할 필요성이 증가되었다. 이러한 요구에 의해 발생한 계획 및 관리 기법들이 Gantt, PERT, CPM 등이다. 그러나 이러한 관리 기법들을 이용하여 프로젝트를 계획함에도 불구하고, 대부분의 프로젝트가 목표 비용 및 일정을 40~200% 초과하게 되었다. 그 이유는 이러한 계획 및 관리 기법들이 업무를 과소평가하거나 재작업을 반영하지 못하기 때문이다. 반복 작업은 처음 수행하거나 위험부담이 있는 작업에서 상세한 업무 공정 순위를 파악하기 위해, 또는 품질을 향상시키기 위

해 피할 수 없는 프로젝트의 특성 중 하나이다. 특히나 철도시스템과 같은 거대 시스템의 설계 및 개발 프로젝트의 경우에는 필수적이다. 설계 및 개발 프로젝트의 각 업무는 마치 “닭과 계란”의 문제처럼 서로 유기적으로 연결되어 있어서, 어느 작업이 먼저 수행되어야 된다고 단언할 수 없다는 것이다. 이러한 관계로 인해서 재작업이 필연적으로 발생한다.

2. 고전적 일정 계획/관리 기법

2.1 간트 차트(Gantt Chart)

간트 차트는 Henry Gantt가 고안한 가장 유명한 프로젝트 관리 도구이다. 통계적으로 전 세계의 80%가 사용하고 있다고 한다.

간트 차트는 과업 수행기간과 프로젝트 일정을 그래픽적인 방법으로 보여준다. 그러나 간트 차트는 업무들 간의 관계를 명확히 보여주지 못한다는 단점을 가지고 있다.

[†] 책임저자 : 정회원, 아주대학교 시스템공학과 박사과정

E-mail : kcmtg@nate.com

TEL : (031)330-7662 FAX : (031)330-7118

* 아주대학교 시스템공학과 교수

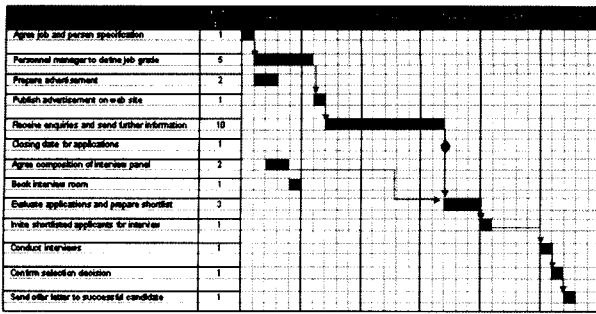


Fig. 1. Gantt Chart

2.2 CPM(Critical Path Method)

CPM은 원가절감에 주 목표를 두고 구현되는 프로젝트 관리 계획 및 관리 기법으로서 1956년부터 1958년까지 미국 듀폰사의 건설계획 추진에 적용되면서 시작되었다.

CPM의 주 특징은 그래픽을 이용하여 프로젝트를 네트워크로 표현한다는 것이다. 이 네트워크를 이용하여 가능한 경로들과 전체 업무 수행기간(duration) 및 주경로를 찾아내고 업무들 간의 의존성을 찾아낸다.

CPM은 프로젝트 내의 모든 업무들을 명확하게 정의할 수 있다는 가정 하에서 시작한다. 그러나 프로젝트 계획 단계에서 프로젝트를 구성하는 개별 업무들의 수행시간을 명확하게 예측하기란 사실상 어렵다. 또한 CPM은 반복 작업을 표현할 수 없다. 예를 들어 A작업 다음에 B작업을 수행하고 다시 A작업으로 돌아가서 B작업을 수행하는 반복 작업(iteration)을 원천적으로 표현할 수 없다. Fig. 2는 CPM의 표현 방법을 보여준다[1]. 주경로를 찾는 방법 등에 대한 상세한 내용은 여러 문헌들에서 다루고 있다[2].

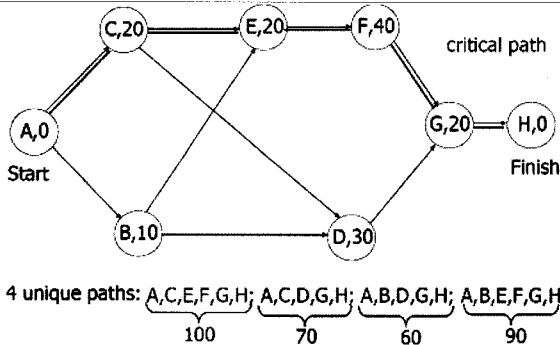


Fig. 2. CPM(Critical Path Method)

2.3 PERT(Project Evaluation and Review Technique)

CPM과 PERT의 차이는 업무 수행시간을 어떻게 예측하는가에 있다. CPM은 이전 프로젝트의 경험을 토대로 하여 업무 수행시간을 정확하게 예측할 수 있다는 가정에서 출발한다. 그러므로 이전 프로젝트의 경험 업무 수행시간을

채택해도 커다란 오차가 발생하지 않는 건설 프로젝트 (construction project)에 적합하다. 이에 비해 PERT는 업무 수행시간을 확률로 예측한다. 각각의 프로젝트는 고유의 특성을 가지므로 업무 수행시간을 예측하는데 있어서 불확실성이 존재하므로 확률론을 적용하여 예측한다는 것이다. PERT는 각각의 업무에 대해 베타 분포에 기반하여 다음과 같은 3가지 업무 수행시간을 결정한다[1].

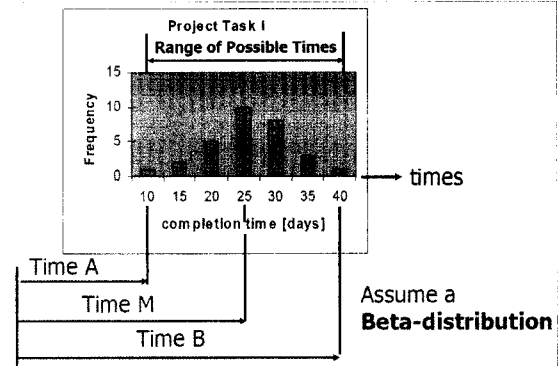


Fig. 3. Time estimation of PERT(Beta distribution)

① A(optimistic time estimate):

과업을 완료하는데 가장 짧은 시간, 모든 일이 올바르게 진행될 경우

② M(most likely task duration):

평범한 작업 조건하에서의 과업 수행시간, 지난 경험에 기반한 가장 빈번히 발생한 작업 수행시간

③ B(pessimistic time estimate):

특별히 극한 환경 하에서 과업을 수행할 경우 소요되는 시간, 예상 못한 지연을 포함하여 가장 소요시간을 예측하기 어려움, 이 시간을 1%도 초과해서는 안됨

각 업무에 대한 시간소요치를 결정한 후에 다음과 같은 공식을 이용하여 기대 시간(TE)과 각 업무의 분산(TV)을 구한다[1].

$$TE = \frac{A + 4M + B}{6} \tag{1}$$

$$TV(\sigma_i^2) = \left(\frac{B - A}{6} \right)^2 \tag{2}$$

프로젝트를 구성하는 각각의 업무에 대한 확률분포가 결정되면 전체 프로젝트를 수행하는데 걸리는 시간을 구할 수 있다. 프로젝트의 총 수행시간에 대한 분포는 정규분포 (중심극한정리에 따라 독립된 표본들의 수가 많으면 분포가 정규분포에 접근하게 됨)를 따른다. 아래 공식을 이용하여 표준편차 점수(sigma score) Z를 구하면 목표일(T) 내에

완료할 수 있는 확률을 구할 수 있다.

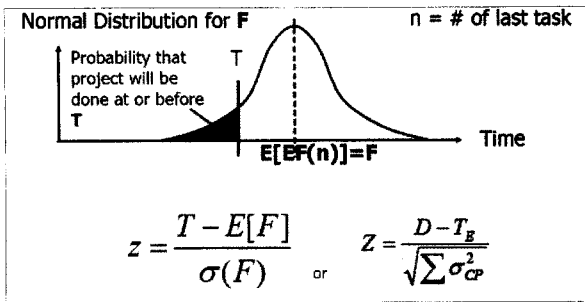


Fig. 4. Z Calculation of PERT

그러나 PERT가 업무 수행시간 예측의 불확실성을 다루면서 CPM보다는 개발 프로젝트의 일정을 보다 타당하게 예측할 수는 있지만 역시 재작업을 고려하지 못한다는 한계를 가진다.

3. 제작업을 고려한 일정 계획법

3.1 일정 계획시 재작업을 고려해야 하는 이유

일반적으로 개발 프로젝트는 다음과 같은 특성을 가지고 있다.

- 후반부 또는 뒤늦게 부가적인 시간이나 자원이 요구됨
- 일정이나 예산의 초과를 가짐
- 90% 완료 시점에서 분발하는 영향이 있음
- 제품이 발매된 후에 결점이 발견됨
- 프로젝트 성격이 unique함으로, 프로젝트 간의 성과를 분석 검토하여 교훈을 얻기가 어려움

이러한 개발 프로젝트의 특성으로 인하여 대부분의 개발 프로젝트는 아래 Fig. 5와 같이 목표 비용과 일정을 40~200% 초과한다[3,4].

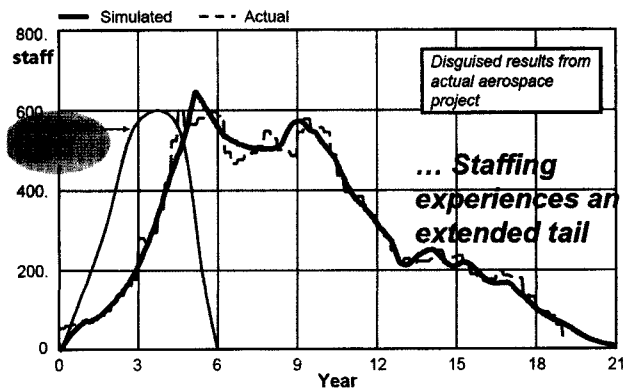


Fig. 5. Project plan vs actuality

이러한 차이는 불확실성 속에 작업을 과소평가하거나 재작업을 고려하지 못함으로써 발생한다. 아래 그림과 같이 프로젝트 계획에 대한 전통적인 관점과 비교하여 살펴보면 그 이유를 확연히 알 수 있다. Fig. 6은 전통적인 프로젝트 계획법을 묘사한 것으로서, Fig. 5에서 가는 실선으로 표현된 일반적인 프로젝트 계획과 동일하다. Fig. 7은 재작업을 고려한 프로젝트 계획법을 묘사한 것으로서, Fig. 5에서 굵은 실선으로 표현된 실제 프로젝트 수행 결과와 동일하다.

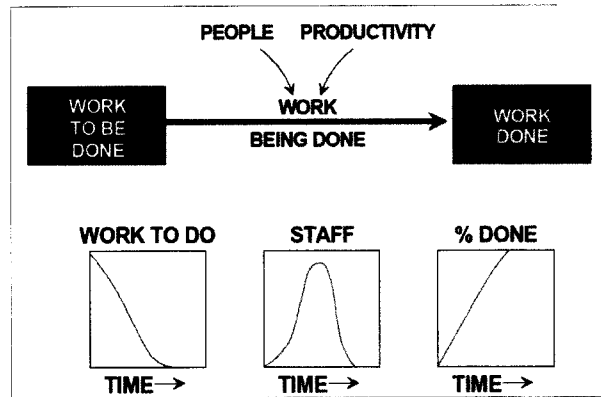


Fig. 6. Traditional project plan

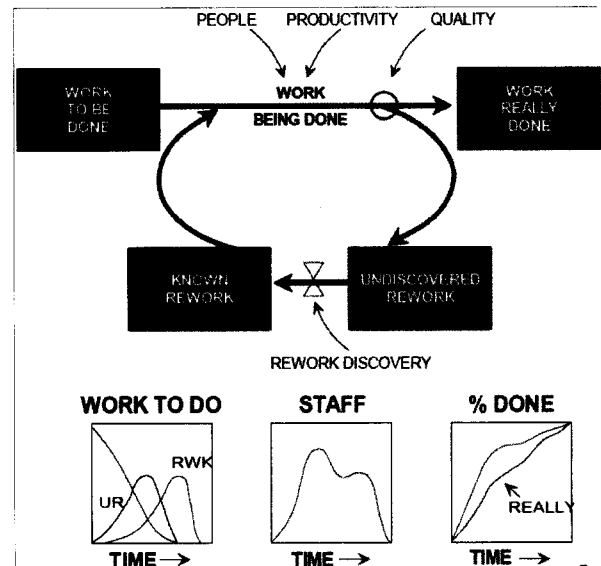


Fig. 7. Project plan considering iteration(rework)

Fig. 7에서 보듯이 반복 작업은 2가지 경우가 있다[5]. 계획된 반복 작업은 품질을 높이기 위해 의도적으로 반복 작업을 수행하는 경우로써, 어느 작업 간에 반복 작업이 생기는지 알 수 있다. 이와 반대로 계획되지 않은 반복 작업은 업무 수행 중 예러나 예측하지 못한 문제로 인해 발생되거나, 업무 결과물의 품질이 원하는 수준을 만족하지 못할 때 발생한다. 그러므로 계획되지 않은 반복 작업은 어디서

일어날지 예측할 수 없다.

이러한 반복 작업(또는 재작업)으로 인해서 초기에 예측한 계획과 실제 프로젝트 결과는 커다란 오차를 보이게 된다. Fig. 5에 재작업의 영향을 표시하면 Fig. 8과 같다.

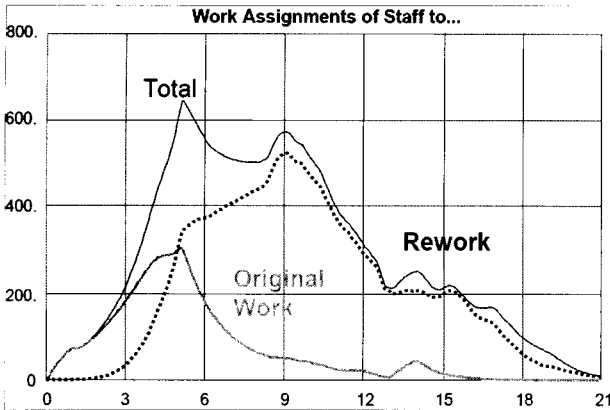


Fig. 8. Iteration impact on project

결국 실제 프로젝트 결과가 초기 계획의 목표 일정 및 비용을 초과하지 않는 보다 정확한 계획을 수립하기 위해서는 재작업을 필히 고려해야 함을 알 수 있다.

3.2 일정 프로세스에서 재작업 표현 방법

앞에서 언급하였듯이 고전적인 일정 계획 방법론(CPM, PERT 등)은 피드백 루프에 의한 반복 작업을 표현할 수 없다.

이러한 문제점을 해결할 수 있는 방법론이 설계구조 매트릭스(DSM: Design Structure Matrix)이다. Task based

DSM은 특히 프로젝트의 업무 프로세스를 표현하기에 적당하다. Task based DSM의 일반적인 표현 방법은 아래 Fig. 9와 같다[6].

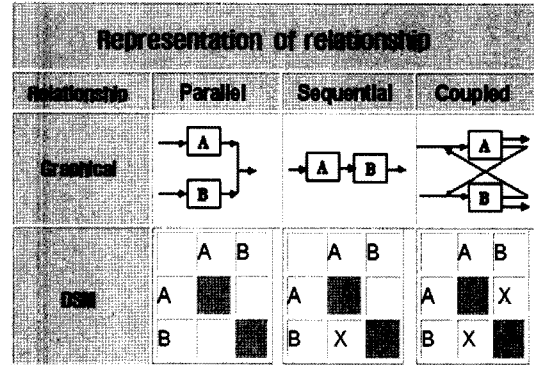


Fig. 9. Representation of relationship between elements

이러한 표현법을 이용하여 병렬적인 업무 관계 및 연속적인 업무 관계, 상호 연결된 업무 관계를 표현할 수 있다. 특히 고전적 일정 계획법에서 표현하지 못하던 피드백 루프에 의한 재작업도 표현할 수 있다. 아래 그림은 고전적 방법론에서 표현하지 못하는 피드백을 DSM에서 어떻게 표현할 수 있는지를 보여준다. Fig. 10의 CPM에서 업무 G에서 업무 C로의 피드백에 의한 반복 작업(붉은 색 선)은 표현 불가능하지만 Fig. 10에서 보듯이 DSM에서는 간단히 표현할 수 있다.

3.3 재작업을 고려한 일정 시뮬레이션 프로그램 작성

재작업이 없는 경우의 프로젝트 총 수행시간을 확률적으

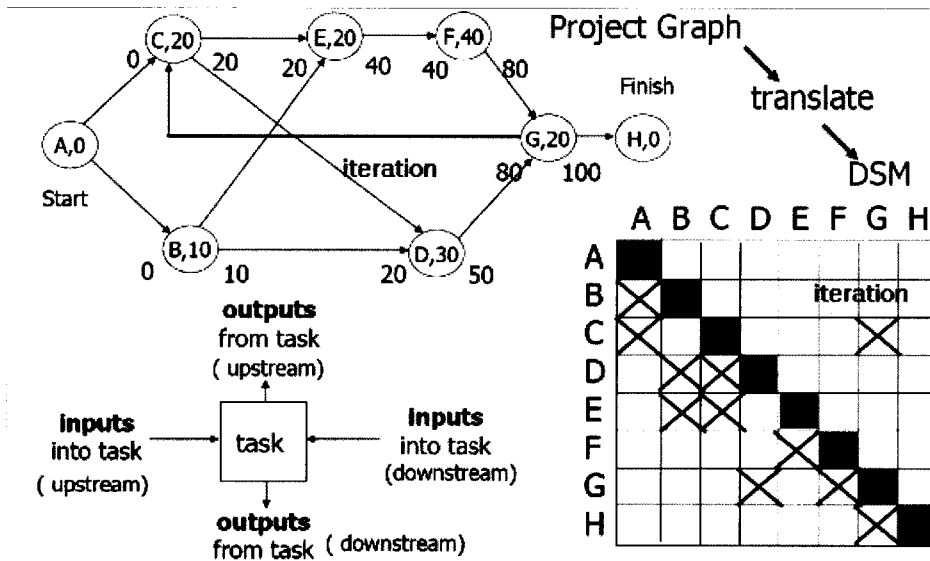


Fig. 10. Representation of iteration on DSM

로 계산하는 방법은 위에서 설명한 바와 같이 간단하다. 그러나 피드백을 통한 재작업이 있을 경우의 확률계산은 간단하지 않다. 이와 같이 재작업이 있을 경우의 프로젝트 총 수행시간을 확률적으로 계산하기 위해서는 몬테카를로 시뮬레이션(Monte Carlo Simulation)을 이용한다. 그러나 아직까지 DSM으로 표현된 일정 프로세스의 경로를 탐색하고 이에 따라 몬테카를로 시뮬레이션을 수행할 수 있는 프로그램이 현재 개발되어 있지 않다. 이에 본 논문은 재작업을 포함하는 일정 프로세스를 DSM으로 표현하고, 일정 시뮬레이션을 수행하여 프로젝트의 총 수행시간을 확률적으로 계산하는 프로그램을 작성하였다. 개발 도구(tool)는 광범위하게 사용되는 엑셀(Excel)에서 구동 가능하도록 VBA를 사용하여 프로그래밍하였다.

3.3.1 프로젝트 업무 및 업무 수행시간 확률분포 정의

일정 시뮬레이션을 수행하기 위해서 먼저 프로젝트를 구성하는 각 업무를 정의하고 각 업무에 대한 수행시간 확률분포를 정의한다.

본 논문에서 사용되는 예는 자기부상열차 시스템 개발 사업이다. 자기부상열차 시스템 개발 사업은 여러 하부 과제들로 구성되어 있다. 그 중 “자기부상열차 차량 개발 및 제작” 과제(과제 1)와 “자기부상열차 부상/추진 시스템 성능 개선” 과제(과제 2)의 업무들을 예로 선택하였다[7].

Table 1. Project task

	업 무	과 제
1	차량 요구사항 분석	1
2	차량 기본사양 개발	1
3	부상시스템 요구사항 분석	2
4	차량 일반도 작성	1
5	차량 조감도 작성	1
6	부상시스템 기본설계	2
7	이중화 마그네트 드라이브 설계	2
8	열차 성능해석	1
9	부상시스템 상세설계	2
10	이중화 마그네트드라이브 시제품제작	2
11	Single Magnet 부상시험 장치 설계	2
12	부상시스템 개발용 MMI 시스템 설계	2
13	Single Magnet 부상시험	2
14	Single Magnet 부상 동특성 해석	2
15	차상 신호 시스템 개념/기본 설계	1
16	영업 운행에 따른 소요 편성수 검토	1

프로젝트를 구성하는 업무가 정의되면 각 업무에 대한 수행시간 확률분포를 정의한다. 본 논문에서는 삼각분포(Triangular distribution)로 정의하였다. 삼각분포는 확률변수(random variable) X가 존재할 수 있는 분포함수인 확률밀도함수(Probability density function)의 형태가 삼각형일 것이라는 가정에서 출발한다.

삼각분포의 모양은 아래 Fig. 11과 같다. 이때 삼각분포의 확률밀도함수 P(x)는 삼각형의 면적이 1이라는 것과 직선의 방정식을 이용하여 다음과 같이 구할 수 있다.

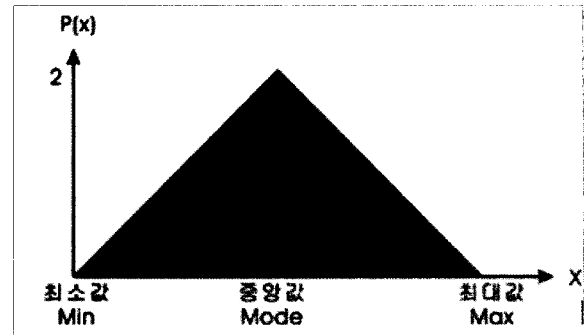


Fig. 11. Triangular distribution

- $\min \leq x < \text{mode}$ 인 경우

$$P(x) = \frac{(2-0) \times (x-\min)}{(\text{mode}-\min)} \times \frac{1}{(\text{max}-\min)} \quad (1)$$

$$= \frac{2}{(\text{max}-\min)} \times \frac{(x-\min)}{(\text{mode}-\min)}$$

- $x = \text{mode}$ 인 경우

$$P(x) = \frac{2}{(\text{max}-\min)} \quad (2)$$

- $\text{mode} \leq x < \text{max}$ 인 경우

$$P(x) = \frac{2}{(\text{max}-\min)} \times \frac{(\text{max}-x)}{(\text{max}-\text{mode})} \quad (3)$$

아래 Fig. 12는 각 업무에 대한 수행시간을 입력할 수 있는 템플릿을 엑셀로 작성한 모습이다. 각 업무의 확률분포를 정의한 다음 “완료” 버튼을 클릭하면 자동으로 다음 단계로 넘어간다.

3.3.2 프로젝트 업무 간 관계 정의

각 업무에 대한 수행시간을 정의한 후 각 업무 간의 관계를 정의한다.

(iteration을 포함한 일정을 계산하기 위한)

일정 토폴로

이래와 같이 되는 번호의 순서대로 입력하고 입력 단?

16

완료

조기 화:

완료

조기 화:

- ※ 각각의 업무는 상각분포로 입력됩니다.
- Min date(Optimistic time estimate): 업무를 가장 빨리 끝내는 데 걸리는 시간
 - Mode date(Most likely task duration): 경험에 기반하여 가장 빈번히 발생하는 업무 수행 시간
 - Max date(Pessimistic time estimate): 예상치 못한 지연 등으로 인한 업무를 가장 늦게 완료하는데 걸리는 시간

Number	Activity Name	Min date	Mode date	Max date
1	자랑 요구사항 분석	50	60	70
2	자랑 기본사항 개발	80	90	100
3	부상시스템 요구사항 분석	25	30	40
4	자랑 일반도 작성	110	130	140
5	자랑 조감도 작성	160	180	200
6	부상시스템 기본설계	40	45	50
7	이중화 마그네트 드라이브 설계	25	30	35
8	결자 성능해석	140	150	160
9	부상시스템 상세설계	40	45	50
10	이중화 마그네트 드라이브 1차 시제품 제작	80	90	100
11	single magnet 부상시험 장치 설계/제작	120	135	150
12	부상시스템 개발용 MMI 시스템 설계/제작	55	60	70
13	single magnet 부상시험	65	75	80
14	single magnet 부상 동특성 해석	60	75	85
15	자랑 신호 시스템 개발/7 불 설계	200	240	260
16	영업 운영에 따른 소요 편성수 검토	55	60	70

Fig. 12. Distribution definition for Task

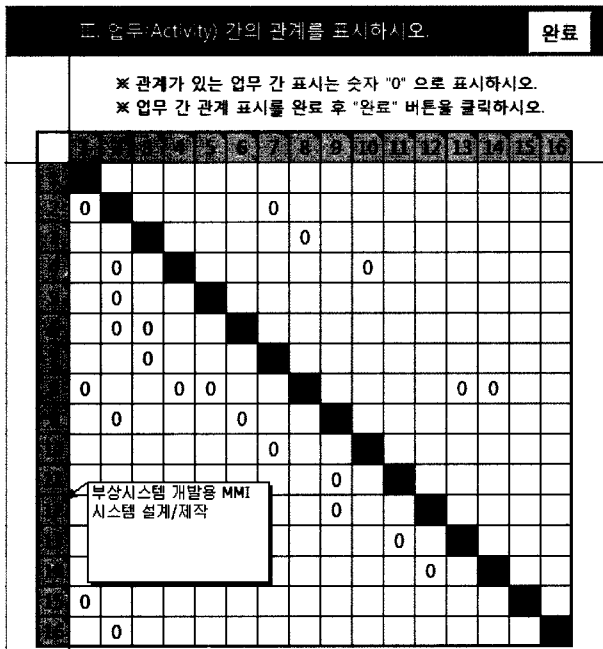


Fig. 13. Iteration matrix

Fig. 13은 16개의 업무 간의 관계를 DSM으로 표현한 것이다. 그림에서 대각 행렬의 위쪽 부분에 있는 5개 셀은 피드백을 나타낸다. 이러한 피드백으로 인해 반복 작업이 발생하며 이러한 반복 작업을 계획된 반복 작업이라 한다. 관계를 정의한 후 "완료" 버튼을 클릭하여 다음 단계를 진행한다.

3.3.3 반복 작업 발생 확률 정의

위 업무 간의 관계 정의에서 5개의 반복 작업 구간이 발생함을 알 수 있었다. 이 단계에서는 5개의 반복 작업이 발생할 확률을 정의한다. 예를 들어 Fig. 13에서 보듯이 Task 7에서 Task 2 사이에 반복 작업이 발생한다. 이때 Task 7의 결과물에 의해 Task 2가 재작업될 확률이 얼마인지를 백분율로 표시한다. 반복 작업 확률은 재작업의 횟수가 증가할수록 줄어든다. 본 프로그램에서는 최대 3회로 정의하였다. 아래 Fig. 14는 재작업 발생 확률을 정의한 모습이다.

IV. 시적업(iteration) 발생 확률을 설정하십시오. 완료

iteration 발생확률은 0 ~ 100 사이 값이며, 횟수가 더해질수록 줄어듭니다.
 ※ 입력 완료 후 완료 버튼을 클릭하십시오.

iteration 발생 경로	iteration 최초 발생확률	iteration 2회 발생
"Act.7" 에서 "Act.2" 으로부터 iteration 발생	20	5
"Act.8" 에서 "Act.3" 으로부터 iteration 발생	10	2
"Act.10" 에서 "Act.4" 으로부터 iteration 발생	20	5
"Act.13" 에서 "Act.8" 으로부터 iteration 발생	15	3
"Act.14" 에서 "Act.8" 으로부터 iteration 발생	20	10

Fig. 14. Probability of iteration occurrence

3.3.4 반복 작업의 영향(impact) 정의

반복 작업 발생 확률을 정의한 다음에 반복 작업의 영향을 정의한다. 반복 작업 영향이란 예를 들어 Task 7에서 반복 작업이 발생하여 Task 2가 재작업될 때를 생각해 보자. Task 2는 Task 4에 어떤 결과물을 전달하므로 반복 작업에 의해 Task 2가 재작업되어 Task 2의 결과물이 변하게 되면 Task 2의 결과물을 받아서 업무를 수행하는 Task 4도 재작업되어야 할 것이다. 이때 Task 4는 본래 작업을 100% 다시 수행하지 않아도 될 것이다. 예를 들어 Task 4는 재작업될 경우 이전 수행 결과물을 20% 재사용할 수 있다면, Task 4는 본래 업무의 70%만 재작업하면 될 것이다. 여기서 70%가 Task 2 재작업시 Task 4의 재작업(또는 반복 작업)의 영향이라고 정의한다. 이때 주의할 사항은 다수의 재작업 영향이 발생할 경우이다. 예를 들어 Task 7에서 Task 2로 반복 작업이 발생할 경우 Task 6을 살펴보자. Task 6의 행(row)을 살펴보면 Task 2와 Task 3으로부터 2개의 결과물을 받으며 이로 인해 2개의 재작업 영향을 가진다. 이 경우에 Task 6의 재작업 영향은 두 값 중 가장 큰 값을 선택해야 한다. 이 알고리즘을 코딩한 것이 Fig. 15이며, 각 재작업 영향을 정의한 것이 Fig. 16이다.

```

(일반)
Function rework_fun(re_rowws, re_columns) As Single
    Dim ro As Integer
    Dim col As Integer
    Dim rework_sel(1 To 50, 1 To 50) As Single

    ro = re_rowws
    col = re_columns

    act_select_dur(si_num, ro) = WorksheetFunction.RoundUp(a

    For i = (ro + 1) To col
        For j = i - 1 To 1 Step -1
            rework_sel(i, j) = rework(i, j)
        Next j
        act_select_dur(si_num, i) = WorksheetFunction.RoundU

        For j = i - 1 To 1 Step -1
            rework_sel(i, j) = 0
        Next j
    Next i

    rework_fun = ro
End Function
    
```

Fig. 15. Selection algorithm of iteration impact

V. 제작업(Iteration)의 영향(Impact)을 설정하십시오. 완료

* 제작업 영향이란:
 - Act.1이 제작업될 경우 Act.2의 종착업 중 몇 개의 작업이 다시 수행되어야
 * 제작업 영향은 0 ~ 100 사이 값입니다.

작업 간 관계	제작업 %
Act.1, 제작업시 Act.2, 의 ○○% 제작업	100
Act.1, 제작업시 Act.8, 의 ○○% 제작업	90
Act.1, 제작업시 Act.15, 의 ○○% 제작업	80
Act.2, 제작업시 Act.4, 의 ○○% 제작업	70
Act.2, 제작업시 Act.5, 의 ○○% 제작업	90
Act.2, 제작업시 Act.6, 의 ○○% 제작업	90
Act.2, 제작업시 Act.9, 의 ○○% 제작업	90
Act.2, 제작업시 Act.16, 의 ○○% 제작업	80
Act.3, 제작업시 Act.6, 의 ○○% 제작업	90

Fig. 16. Iteration impact

3.3.5 일정 시뮬레이션(Monte Carlo Simulation)

일정 시뮬레이션의 주요 알고리즘은 크게 2가지이다. 첫째는 난수 발생 모듈을 랜덤함수(Rnd())를 이용하여 각 업무의 수행시간 분포에서 확률에 따라 하나의 수행시간을 선택하는 알고리즘이다. 두 번째는 경로를 탐색하여 시작 시간을 정의하는 알고리즘이다. 예를 들어 A, B, C 3개의 업무(Task)를 생각해 보자. 초기에 모든 업무의 시작 시간은 0으로 설정한다. 그 다음 업무 A → B → C 순서로 시작 시간을 재설정한다. 이때 업무 C의 시작 시간은 업무 B의 시작시간에 B의 수행시간을 더한 값과 업무 C에 설정된 시작시간을 비교하여 큰 값을 업무 C의 시작시간으로 설정한다. 이를 수식으로 쓰면 다음과 같다.

$$\text{Task[C].start} = \text{Max}\{\text{Task[B].start} + \text{duration}, \text{Task[c].start}\} \quad (6)$$

Fig. 17은 시작시간을 설정하는 알고리즘을 이해하기 쉽게 표현한 것이다.

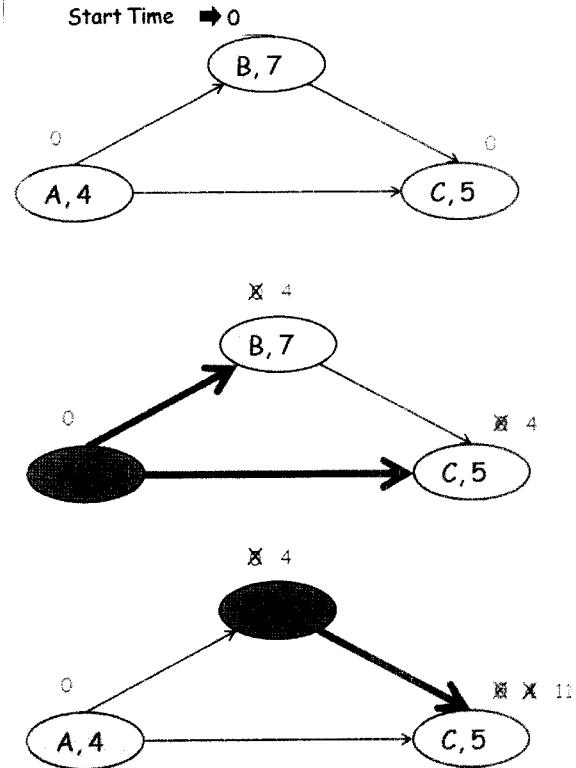


Fig. 17. Algorithm of start time setting

반복 작업이 있을 경우는 이보다 복잡해진다. 경로를 따라 가면서 시작시간을 재설정함과 동시에 그 업무가 몇 번째로 수행되는지를 판단해야 한다. 예를 들어 업무 A가 제작업에 의해 2번째 수행된다면, 업무 A의 1번째가 아니라 2번째 시작시간만 다시 설정하게 된다. 아래 Fig. 18은 반복 작업이 있을 경우를 고려한 몬테카를로 시뮬레이션 알고리즘을 설명한 것이다.

- Step 1: 초기화(모든 업무의 시작시간, 종료시간을 초기화)
- Step 2: 입력된 Min, Mode, Max에 따라 삼각분포 정의
- 반복 구문 시작(10000번 수행)---
- Step 3: 난수를 발생하여 각 업무 분포에 따라 각 업무 수행시간을 선택
- Step 4: 정의된 DSM 관계표시에 따라 총 업무 수행시간 계산(행 기준)
 - Step 4.1 선택된 업무가 몇 번째 수행되는지 조사
 - Step 4.2 선택된 업무의 시작시간에 선택된 수행시간(duration)을 더하여 종료시간을 계산
 - Step 4.3 난수를 발생하여 정의된 Iteration 발생확률과 비교하여 Iteration이 발생여부 조사
 - Step 4.4 선택된 업무로부터 정보를 받는 모든 업무의 시작시간은 저장된 시작시간과 선택된 업무의 종료시간과 비교하여 큰 값을 시작시간으로 저장
- 반복 구문 종료-----
- Step 5: 각 업무의 종료시간 중 최대로 큰 값을 total_finish_time에 저장
- Step 6: 10000개의 total_finish_time에 대한 도수 분포표를 작성
- Step 7: 히스토그램 작성

Fig. 18. Algorithm of Monte Carlo Simulation

4. 결론

앞서 설명한 바와 같이 대부분의 프로젝트는 계획시의 일정 및 비용을 40~200% 초과하게 되며, 이는 프로젝트 계획시 업무 일정을 과소평가하거나 모든 프로젝트 특성 상 많은 경우 필요한 재작업을 고려하지 않음으로써 발생한다.

이에 따라 본 페이퍼에서는 프로젝트 일정 계획시 현실과 유사한 결과값을 채택하기 위한 방법들을 연구하였다.

연구 결과를 요약하면 다음과 같다.

- ① 기존의 일정 계획법 분석
- ② 일정 프로세스에서 재작업 표현 방법 연구
- ③ 재작업과 불확실성을 고려한 일정 시뮬레이션 (Monte Carlo Simulation) 프로그램 작성
 - 일정 프로세스 탐색 프로그램 작성
 - 확률 계산 프로그램 작성
 - 결과 그래프 출력 프로그램 작성

본 논문에서는 프로젝트 계획 중 일정 계획 부분만을 다루었다. 자원(비용, 팀, 조직)의 활용성을 고려한 계획방법에 대한 연구는 향후 추진 예정이다.

참고문헌

1. Olivier de Weck(2003), "System & Project Management", MIT lecture material
2. PMBOK 지침서 2004 개정 프로젝트 팀(2004), "프로젝트 관리 지식체계 지침서", pp. 147, Project Management Institute
3. Peter Morris and George Hough(1987), "The anatomy of major projects: A study of the reality of project management", John Wiley and Sons
4. Edward B.Roberts(1992), "Management of Technology: Global Benchmarking", Massachusetts Institute of Technology, December 10
5. Yonas Nebiyeloul-Kifle(2005), "Application of the design structure matrix to integrated product development process", MIT, pp.13
6. Tyson R. Browning(2001). "Applying the design structure matrix to system decomposition and integration problems: a review and new directions", IEEE Transactions on Engineering Management, Vol. 48, No. 48, August, pp.299-304
7. 도시형 자기부상열차 실용화 사업단(2007). "인터페이스 조정회의".

접수일(2008년 8월 12일), 수정일(2008년 9월 10일),
게재확정일(2009년 2월 13일)