

자원 제약을 고려한 프로젝트 일정문제

Resource constrained project scheduling problem

- 재작업 가능한 활동이 있는 경우 -
- with reworkable activities -

2019년 10월 02일

CONTENTS

- 제1장 서론
- 제2장 이론적 배경(문헌연구)
- 제3장 문제의 모형
- 제4장 알고리즘
- 제5장 모의실험 결과
- 제6장 결론
- 제7장 참고문헌

서론

01



1. 문제의 배경

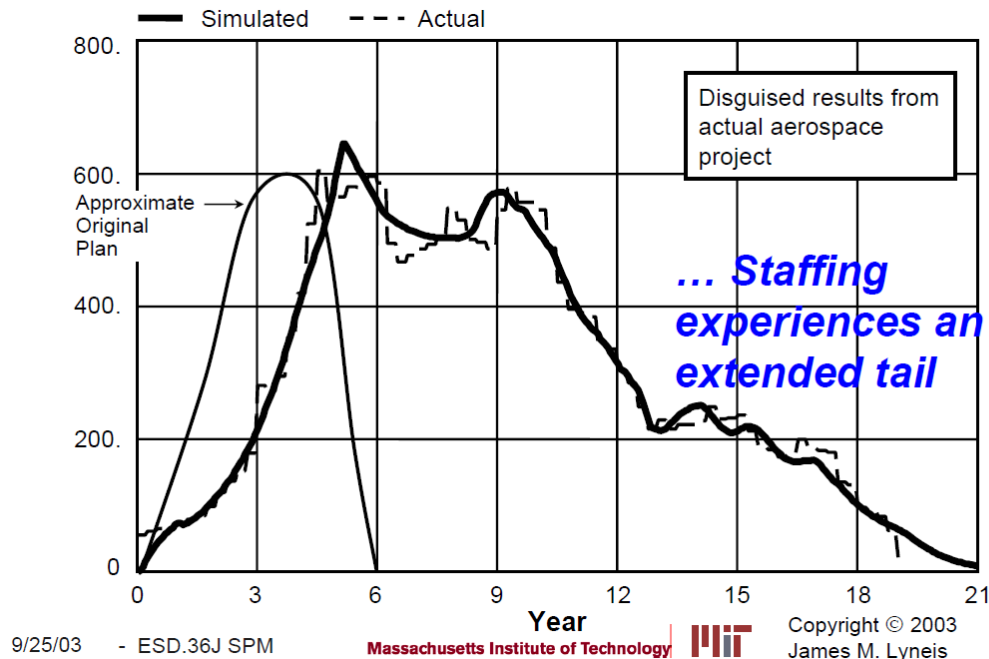
1.1 연구 배경 및 목적

- 현대 사회는 과학 기술의 성장 및 다양한 분야의 발전으로 프로젝트의 필요성이 증대되었고, 프로젝트의 체계적인 관리방법 또한 발전되어 왔다. 프로젝트 일정문제(Project Scheduling Problem) 영역에서도 프로젝트가 가지고 있는 불확실성으로 인해 발생하는 오차를 줄이기 위해 다각도로 연구가 진행되어 왔다.
- 프로젝트 일정계획의 정확성을 높이는데 필요한 요소를 규명하고 일정문제에 적용하는 방안을 연구함으로써 실행대비 계획 수립시의 일정산정에 대한 오차를 줄이고자 노력했음을 알 수 있다.
- 일반적으로 프로젝트는 초기에 수립된 일정계획과 달리 애기치 못한 변수의 등장으로 인해 활동의 재작업이 필수적이라고 할 만큼 일어나는 것이 현실이다. 대개의 경우, 재작업은 프로젝트가 잘못 관리되거나 프로젝트 관리의 부재로 발생할 수 있으며 프로젝트 작업 감시 단계 및 통제 단계 시점에서 발생할 수 있다(PMBOK 2017).



1. 문제의 배경

1.1 연구 배경 및 목적



- 실제로 진행된 개발 프로젝트의 대부분은 초기에 산정된 목표 비용과 일정을 적게는 40%에서 많게는 200%를 초과하기도 하는데, 이는 초기 일정산정에서 업무를 과소평가하거나 재작업을 고려하지 않아서 생긴 문제로 볼 수 있다.



1. 문제의 배경

1.2 문제 요약

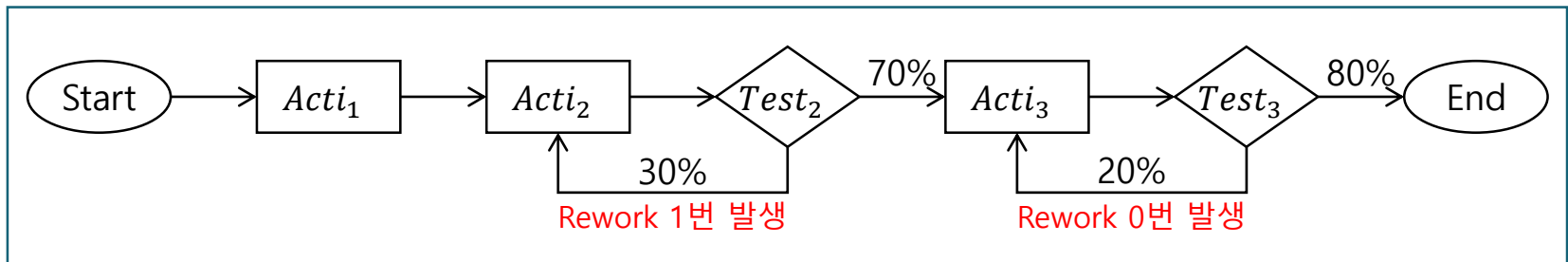
- 따라서 본 논문에서는 일정문제의 산정에 활동의 비용과 기간에 확률적으로 적용되는 요소를 찾아 적용하고, 이를 적용하는 모델을 개발하고자 한다.
- 본 논문의 차별성 : 기존의 **RCPSP(자원제약을 고려한 프로젝트 일정문제)**연구에서는, **재작업이라는 불확정요소에 대해 확률적으로 적용**한 연구는 찾을 수 없었으며, 이를 본 논문에서 다루고자 한다.



1. 문제의 배경

예시) 3개의 활동이 있고, 활동2, 3에 시험이 있는 일정

- Due Date : 12 day, Penalty Cost : 4/day



- ① T=0 에서의 산정 : Test의 기간은 기대 값 $E(x) = Acti_x \times (\text{불합격확률}_x)$ 으로 산정. $Test_2 = 0.6$, $Test_3 = 0.8$
 Crashing으로 $Acti_3$ 재작업 기간의 5 → 4 (Total Cost 변동 ≤ 0 경우 적용)

	$Acti_1$	$Acti_2$	$Test_2$	$Acti_3$	$Test_3$	
Duration	2.6	4	0.6	4	0.8	Complete Time : 13 -> 12
Cost	4.2	4		3		Penalty Cost : 4 -> 0
Crashing Cost / day	3	4		3		Crashing Cost : 0 -> 4
Normal Duration	4	5		5		
Crash Duration	2	2		4		Total Cost : 12.2 -> 11.2



1. 문제의 배경

예시) 3개의 활동이 있고, 활동2, 3에 시험이 있는 일정(계속)

② T=6.6 에서의 산정 : $Acti_2$ 까지 완료, $Test_2$ 불합격에 따라 $Acti_2$ 재작업 및 기간 4 추가

	$Acti_1$	$Acti_2$	$Test_2$	$Acti_3$	$Test_3$	
Duration	2.6	4 + 4	0.6	4	0.8	Complete Time : 12 -> 16
Cost	4.2	4		3		Penalty Cost : 0 -> 16
Crashing Cost / day	3	4		3		Crashing Cost : 0
Normal Duration	4	5		5		
Crash Duration	2	2		4		Total Cost : 11.2 -> 31.2

③ T=6.6 에서의 *Crashing* 적용 : $Acti_2$ 재작업 기간의 4 → 2 (*Total Cost* 변동 ≤ 0 경우 적용)

	$Acti_1$	$Acti_2$	$Test_2$	$Acti_3$	$Test_3$	
Duration	2.6	4 + 2	0.6	4	0.8	Complete Time : 16 -> 14
Cost	4.2	4		3		Penalty Cost : 16 -> 8
Crashing Cost / day	3	4		3		Crashing Cost : 0 -> 8
Normal Duration	4	5		5		
Crash Duration	2	2		4		Total Cost : 31.2



1. 문제의 배경

예시) 3개의 활동이 있고, 활동2, 3에 시험이 있는 일정(계속)

④ T=8.6 에서의 산정 : $Acti_2$ 까지 재 완료, $Test_2$ 합격에 따라 $Test_2$ 기대값 $0.6 \rightarrow 0$

	$Acti_1$	$Acti_2$	$Test_2$	$Acti_3$	$Test_3$	
Duration	2.6	4 + 2	0	4	0.8	Complete Time : 14 -> 13.4
Cost	4.2	4		3		Penalty Cost : 16 -> 5.6
Crashing Cost / day	3	4		3		Crashing Cost : 8
Normal Duration	4	5		5		
Crash Duration	2	2		4		Total Cost : 31.2 -> 28.8

⑤ T=12.6 에서의 산정 : $Acti_3$ 까지 완료, $Test_3$ 합격에 따라 $Test_3$ 기대값 $0.8 \rightarrow 0$, Complete.

	$Acti_1$	$Acti_2$	$Test_2$	$Acti_3$	$Test_3$	
Duration	2.6	4 + 2	0	4	0	Complete Time : 13.4 -> 12.6
Cost	4.2	4		3		Penalty Cost : 5.6 -> 2.4
Crashing Cost / day	3	4		3		Crashing Cost : 8
Normal Duration	4	5		5		
Crash Duration	2	2		4		Total Cost : 28.8 -> 25.6

이론적 배경

02



2. 이론적 배경

2.1 프로젝트 일정문제에 재작업이 있는 활동의 필요성

- 대부분의 개발 프로젝트는 목표 비용과 일정을 40~200% 초과한다, 이러한 초과현상은 계획 시 업무를 과소평가하거나 재작업을 고려하지 않음으로써 발생하며, PERT와 CPM등의 고전적인 일정 기법으로는 재작업을 계획에 반영할 수 없기 때문이다.(김찬묵·박영원, 2009).

2.2 프로젝트 일정문제 선행연구

- 2.2.1 PERT(Program Evaluation & Review Technique) / CPM(Critical Path Method) 일정 계획 :
1958년 미국 해군에 의해 폴라리스 미사일(Polaris Missile) 프로젝트의 일정 계획 및 통제를 위해 개발된 PERT와, 1959년 Kelly & Walker(1959)에 의해 개발된 CPM은 그 수행기간이 각각 확정적인지 확률적인인지로 구분할 수 있다(Tara, 2011).



2. 이론적 배경

저자	주요 내용
Lu & AbouRizk (2000)	이산 사건 모델링 접근법과 단순화된 중요 활동 식별 방법을 통합한 PERT 시뮬레이션 모델에 대해 연구 활동 분할을 통한 다중 자원 평준화 문제의 비용의 최소화에 대한 해법을 하이브리드 메타 휴리스틱 방식을 제시함.
Mota et al. (2007)	전력 시스템의 정전 사고 이후 복원 계획 수립에 PERT/CPM 그래프 이론을 기반으로 한 방법론을 제안
Herroelen & Leus(2001)	주공정일정(CC, Critical Chain-scheduling) 및 여유 시간관리(BM, Buffer Management)로 알려진 프로젝트 관리에 제약이론을 적용
Bie et al.(2012)	활동 간 의존도가 프로젝트 기간 성과에 미치는 영향을 분석함으로써, 활동 간 종속성을 가정하여 버퍼 크기를 결정하는 방법을 제시

- 2.2.2 자원 제약을 고려한 프로젝트 일정문제(RCPSP, Resource Constrained Project Scheduling Problem):
자원 제약을 고려한 프로젝트 일정문제는 활동의 선행관계, 제약된 자원, 마감기한, 패널티 비용의 제약조건하에 활동의 우선순위와 모드의 선택에 관한 문제이다.



2. 이론적 배경

- 2.2.3. 정수 선형 계획법(Integer linear Programming):
정수 선형 계획법은 변수를 정수 값으로 제한하여 최적화 문제를 다루는 기법을 말한다.
- 2.2.4. 자원 제약을 고려한 일정문제의 휴리스틱 기법:
자원 제약을 고려한 프로젝트 일정문제(RCPSP)는 Blazewicz(1978)가 매우 큰 문제(NP-Hard Class)임을 증명한 이후, 다양한 방법의 휴리스틱 기법이 연구되어 왔다. 특히, Patterson et al.(1990)은 역추적(Backtracking) 알고리즘을 이용하여 프로젝트의 기간을 최소화시키고, 프로젝트 순현재가치(NPV, Net Present Value)를 최대화시키는 유용한 절차를 제시하였다. 순현재가치 최대화 문제를 가진 자원 제약을 고려한 일정 문제는 Doersch & Patterson(1977)에 의해 소개되었고, Yang et al.(1993)에 의해서 후속 연구가 진행되었다(이현기, 2017).



2. 이론적 배경

2.3 최신동향

연구자	연구내용	주요분야
Patterson (1973)	한정된 자원하의 프로젝트 일정문제를 위한 7가지 휴리스틱	Heuristic
Patterson et al.(1990)	자원 제약을 고려한 프로젝트 일정문제를 위한 역추적 알고리즘	backtracking algorithm
안태호 (1998)	자원 제약을 고려하여 기간 단축이 가능한 복수의 양식을 지닌 단일 프로젝트 일정 문제: 자원 가용량이 시간에 따라 변하는 경우	Heuristic
Lova et al.(2006)	다중 경로 휴리스틱 기법	Multi-Mode Heuristic
Van Peteghem & Vanhoucke (2010)	다중 모드 자원 제약하의 프로젝트 일정문제에 대한 유전 알고리즘	Genetic Algorithm
Kapur et al. (2012)	소프트웨어 출시 일정 계획 수립을 위한 2차원 다중 릴리스 소프트웨어 안정성 모델링 및 최적화 문제를 해결하기 위한 휴리스틱 알고리즘	Meta Heuristic
김갑식 (2016)	자원 제약을 고려한 프로젝트 일정계획에서 선행 활동의 품질영향계수가 후행활동에 미치는 문제를 해결하기 위한 최적화 및 휴리스틱 알고리즘	Exact Solution Neighbourhood Heuristic
Etgar et al. (2017)	버전계획 및 릴리스 문제에 대한 순 현재가치를 최대화할 수 있는 제품 출시 일정계획을 위한 다양한 최적화 기법과 휴리스틱 기법	Exact Solution Meta Heuristic
이현기 (2017)	자원의 렌트를 이용한 자원의 추가투입으로 인한 자원 가용량의 변동을 통한 비용최소화 휴리스틱 알고리즘	Heuristic

문제의 모형

03



3. 수학적 모델

3.1 문제의 개요

- 1) 활동의 재작업이 있는 프로젝트에서 활동 i 을 라하고 그 실행 방법인 모드 m 를, 활동 i 가 재작업을 포함하여 프로젝트에 실행되는 횟수를 r , 활동 i 가 최종 반복된 횟수를 $R(i)$ 라 하고 그 값은 알려져 있다고 하자.
- 2) $R(i)$ 와 r 의 관계는 $r = 1, \dots, R(i), \dots, \infty$ 로 표현할 수 있다.
- 3) 또한 활동 i 의 r 번째 반복이 수행되는지 여부를 $c_{i,r}$ 이라 하고 그 값이 $r \leq R(i)$ 인 경우 1이고 아니면 0이라 하자, 또한 활동 i 가 r 번째 반복에서 모드 m 으로 수행되는지 여부를 $x_{i,r,m}$ 라 하고 그 값이 0 또는 1의 값을 가진다면, 활동 i 의 r 번째 반복에서의 기간은 $\sum_m dur_{i,m} \times x_{i,r,m} \times c_{i,r}$ 로 나타낼 수 있으며, 그 비용은 $\sum_m Acti_cost_{i,m} \times x_{i,r,m} \times c_{i,r}$ 이다.
- 4) 이는 반복되는 활동 i 에 대한 활동기간의 총 합을 $\sum_r \sum_m dur_{i,m} \times x_{i,r,m} \times c_{i,r}$ 로, 활동 비용의 합을 $\sum_r \sum_m Acti_cost_{i,m} \times x_{i,r,m} \times c_{i,r}$ 로 수식화할 수 있다. 따라서 이 프로젝트 총 비용은 $\sum_i \sum_r \sum_m Acti_cost_{i,m} \times x_{i,r,m} \times c_{i,r} + (Penalty_cost)$ 로 표현할 수 있다.



3. 수학적 모델

3.1 문제의 개요

- 5) 일반적으로 RCPSP의 목적함수는 $\sum_i \sum_m Acti_cost'_{i,m} \times x_{i,m} + (Penalty_cost')$ 이며,
- 6) 이로부터 4와 5의 함수가 같다면, 다음과 같은 식을 만들 수 있다.

$$\sum_i \sum_r \sum_m Acti_cost_{i,m} \times x_{i,r,m} \times c_{i,r} + (Penalty_cost)$$

$$\sum_i \sum_m Acti_cost'_{i,m} \times x_{i,m} + (Penalty_cost')$$

$$\sum_i \sum_m Acti_cost'_{i,m} \times x_{i,m} \times E^1 = \sum_i \sum_r \sum_m Acti_cost_{i,m} \times x_{i,r,m} \times c_{i,r}$$

- 7) 그러나 일반적인 프로젝트의 특성상 정수 선형계획법을 사용하는 것이 타당하며, RCPSP의 경우 자원 제약이라는 특성에 의하여 모든 문제는 위의 수식만으로 풀 수는 없다.

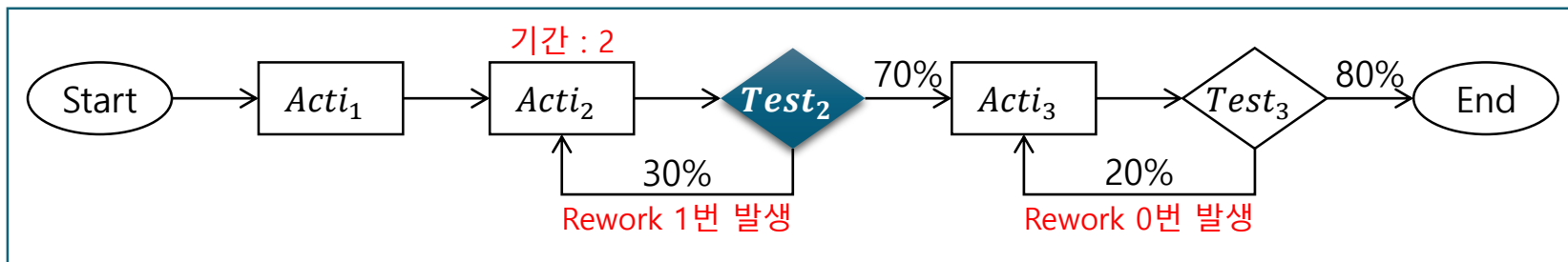


3. 수학적 모델

3.2 조건부활동의 기간과 비용에 대한 확률적 기댓값

- Sub Problem의 $TestCost$ 에 대한 기대 값 수식화

1. $Test_2$ 의 기대값 산출 : $Acti_2$ 의 기간이 2이며, $Test_2$ 불합격 시 $Acti_2$ 로만 갈 경우



$Test_2$ 가 Pass할 기대 값은 아래와 같다.

$$\underbrace{0 \cdot 70\%}_{\text{첫 번째 통과할 확률}} + \underbrace{2 \cdot 30\% \cdot 70\%}_{\text{두 번째 통과할 확률}} + \underbrace{2 \cdot 30\% \cdot 2 \cdot 30\% \cdot 70\%}_{\text{세 번째 통과할 확률}} + \dots + \underbrace{(2 \cdot 30\%)^{\infty} \cdot 70\%}_{\text{마지막 통과할 확률}}$$



3. 수학적 모델

3.2 조건부활동의 기간과 비용에 대한 확률적 기댓값

일반화 하면 (재 작업 Act_i 의 기간을 D_x 라 하고 $Test$ 활동의 기대 값 $E(x)$ 는 아래와 같다.)

$$E(x) = 0 \cdot p + 2 \cdot 1 \cdot (1-p)p + 2 \cdot 2 \cdot (1-p)^2p + \dots + 2 \cdot \infty \cdot (1-p)^\infty p$$

$$= 0 \cdot p + D_x \cdot p \sum_{n=1}^{\infty} n(1-p)^n$$

$$\text{since } \sum_{n=1}^{\infty} nx^n = \frac{x}{(x-1)^2} \text{ Sum of power series}$$

$$= D_x \cdot p \cdot \frac{(1-p)}{(1-p-1)^2}$$

$$= D_x \cdot p \cdot \frac{(1-p)}{p^2}$$

$$= D_x \cdot \frac{(1-p)}{p}$$



3. 수학적 모델

3.2 조건부활동의 기간과 비용에 대한 확률적 기댓값

$$(\text{총 비용}) = \left\{ \sum_{i=1}^N \left\{ (\text{조건부활동 } i \text{의 비용}) \times \left(1 + \frac{(1-p_i)}{p_i} \right) \right\} \right\} + (\text{총 패널티 비용})$$

그러나, 위의 은 각각의 활동 i 의 재작업 확률 p_i 로 인하여 실제 시뮬레이션에서의 총 비용과 다를 수밖에 없으며, 이는 실제 시뮬레이션의 총 비용에 대하여 다음과 같이 정리할 수 있다.

(모든 활동의 재작업이 없는 총 비용) \leq (실제 시뮬레이션 총 비용)

(실제 시뮬레이션 총 비용)의 이론적 평균 = 수식화된 (총 비용)



3. 수학적 모델

3.3 문제의 가정

- 1) 활동의 수 : 프로젝트 전체 활동의 수가 N 개일 경우, 산정에 사용되는 실제 활동의 수는 N 개이며, 재작업 발생에 대한 부분은 활동의 기간에 산정 정책을 적용하여 계산한다. 또한 시작과 종료에 해당하는 두 개의 가상 활동(Dummy Activity)을 각각 0, $N+1$ 이라 하고 기간, 비용, 자원이 0이다.
- 2) 활동의 선·후 관계 : 활동 간에는 선·후 관계가 존재하며, 선행 활동이 모두 종료한 이후 후행은 시작할 수 있다. 활동 간의 선행 관계 집합을 R 라 할 때, $(i, j) \in R$ 이면, 활동 i 는 활동 j 가 종료되기 전에는 시작할 수 없음을 의미한다.
- 3) 활동의 비선매성 : 일단 활동이 시작되면 연속적으로 완료시까지 수행되어야 한다(Non-preemptive)
- 4) 모드(Mode) : 활동이 수행되는 방법을 모드로 할 때 각 활동마다 하나 이상의 모드가 존재한다.



3. 수학적 모델

3.3 문제의 가정

- 5) 소요자원 : 활동은 소요자원을 소비하며, 소비된 자원은 재사용이 불가하다.
- 6) 모드 내 활동의 기간단축 : 모드 내 활동의 기간 단축은 불가하다.
- 7) 프로젝트 마감기한(Due Date) : 프로젝트는 마감기한이 존재하며 프로젝트의 완료시점이 마감기한보다 지연되면, 지연시점부터 단위기간마다 지체보상금(Penalty Cost)이 발생한다.
- 8) 가용자원 : 프로젝트에서 활동이 사용할 수 있는 가용자원은 사전에 알려져 있다.
- 9) 기타추가비용 : 활동비용과 지체보상금을 제외한 비용의 추가는 발생하지 않는다.



3. 수학적 모델

3.4 문제의 수식화

$$\text{Min} \sum_i \sum_m \text{Acti_Cost}_{i,m} \times x_{i,m} + \text{PenaltyCost} \times \max\{S_{N+1} - \text{DueDate}, 0\}$$

Subject to

$$x_{i,m} \in \{0,1\} \quad \forall i,m$$

$$\sum_{m \in M(i)} x_{i,m} = 1 \quad x_{i,m} \in \{0,1\} \quad \forall i$$

$$e_i = \begin{cases} 1 & \text{if } p_i = 100\% \\ E & \text{otherwise} \end{cases}$$

$$\text{Acti_cost}_{i,m} = \sum_{m \in M(i)} NC_{i,m} \times x_{i,m} \times e_i \quad i = 0, 1, 2, \dots, N, N+1$$

$$s_i + d_{i,m} \leq s_j \quad \forall (i,j) \in P$$

$$s_0 = 0$$

$$\sum_{i \in S_t} \sum_{m \in M(i)} R_{i,m,k} \times x_{i,m} \leq b_{t,k} \quad \forall t,k$$

$$d_{i,m} = \sum_{m \in M(i)} \text{dur}_{i,m} \times x_{i,m} \times e_i \quad \forall i,m$$

알고리즘

04



4. 알고리즘

4.1 Solution Space

- 본 문제의 해의 영역은 목표함수의 해의 영역과 목표함수의 를 추정하기 위해 시뮬레이션 되는 실제 프로젝트의 해의 영역 두 가지로 구분된다

4.1.1 목표함수의 해의 영역 :

목표함수의 해의 영역은 일반적인 RCPSP 해의 영역과 동일하다.

$$\left\{ \begin{array}{cc} Mode & Dur \\ m_1 & d_1 \\ m_2 & d_2 \\ m_3 & d_3 \\ \dots & \dots \\ m_N & d_N \end{array} \right\} \quad \left\{ \begin{array}{c} Start \ Time \\ s_1 \\ s_2 \\ s_3 \\ \dots \\ s_N \end{array} \right\}$$

[그림 4-1] 활동의 배정 관련 해 영역

여기서 기간은 $d_i = dur_i \times E$ 가 된다.



4. 알고리즘

4.1 Solution Space

4.1.2 시뮬레이션 해의 영역 :

$$\sum_i \sum_r \sum_m Acti_cost_{i,m} \times x_{i,r,m} \times c_{i,r} + (Penalty_cost)$$

에서

$$\left\{ \begin{array}{cccccc} c_{1,1} & c_{1,2} & \cdots & c_{1,R(i)} & \cdots & c_{1,\infty} \\ c_{2,1} & c_{2,2} & \cdots & c_{2,R(i)} & \cdots & c_{2,\infty} \\ c_{3,1} & c_{3,2} & \cdots & c_{3,R(i)} & \cdots & c_{3,\infty} \\ \vdots & \vdots & \vdots & \vdots & & \vdots \\ c_{N,1} & c_{N,2} & \cdots & c_{N,R(i)} & \cdots & c_{N,\infty} \end{array} \right\}$$

[그림 4-2] 활동 관련 부분 해

$$\left\{ \begin{array}{cc} Mode & Dur \\ m_1 & d_1 \\ m_2 & d_2 \\ m_3 & d_3 \\ \cdots & \cdots \\ m_N & d_N \end{array} \right\} \quad \left\{ \begin{array}{c} Start \ Time \\ s_1 \\ s_2 \\ s_3 \\ \cdots \\ s_N \end{array} \right\}$$

[그림 4-3] 활동의 배정 관련 해 영역

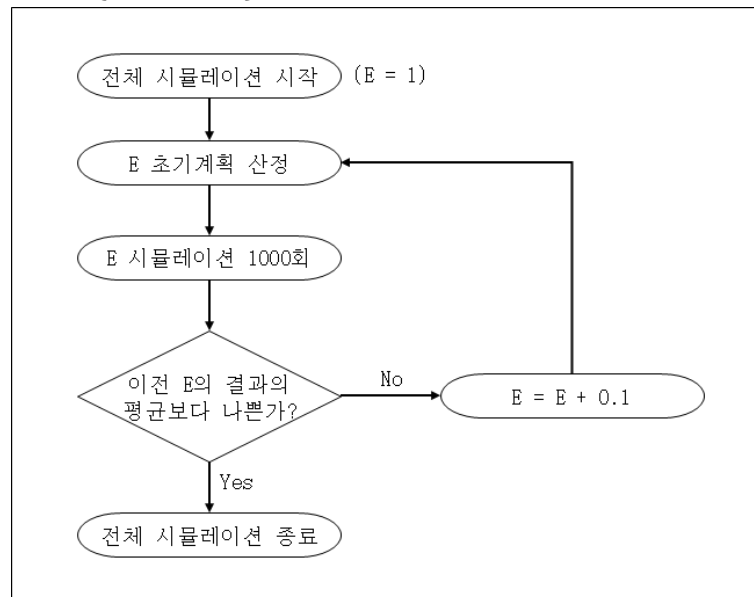


4. 알고리즘

4.2 휴리스틱 기법을 이용한 시뮬레이션

4.2.1 시뮬레이션 휴리스틱 기법 약속

- 1) 이웃해 탐색 기법을 주된 방법으로 한 휴리스틱 기법으로 $E=1$ 인 초기계획의 비용 최소화 일정을 탐색한다.
- 2) 초기계획을 기준으로 조건부활동이 종료될 때 통과확률에 따라 반복이 일어나는 시뮬레이션을 1000회 반복하여 결과의 비용의 평균을 구한다.
- 3) E 에 0.1씩 증가시키며 시뮬레이션 결과의 평균이 최소화되는 가중치 산정정책을 찾는다. 단계 0. {초기화}





4. 알고리즘

4.2 휴리스틱 기법을 이용한 시뮬레이션

4.2.2 E 초기계획의 비용 최소화 휴리스틱 기법

Step 0. 초기화. 임시 목적함수 A에 또는 매우 큰 값을 저장한다. 활동의 수, 활동별 모드의 수, 활동비용, 선행관계 등 기초 데이터를 설정한다. 조건부 활동에 한하여 활동기간을 로 설정한다. Goto Step 1.

Step 1. 최초 실행 가능 해를 작성하고, 목적함수 값을 계산하여 임시 목적함수 B에 저장한다. Goto Step 2.

Step 2. 현 실행 가능 해를 다양한 기법을 사용하여 개선하고, 목적함수 값이 B보다 좋으면 B에 저장한다. Goto Step 3.

Step 3. 제한된 연산 시간이 초과되었으면 Goto Step 4. 아니면, Goto Step 2.

Step 4. B값이 A값보다 좋으면 A값에 B값을 넣는다. Goto Step 5.

Step 5. 제한된 연산 횟수가 초과되었으면 Goto Step 6. 아니면, Goto Step 1.

Step 6. 최종 목적함수를 A로 하여 종료한다.



4. 알고리즘

4.2 휴리스틱 기법을 이용한 시뮬레이션

4.2.3 E 시뮬레이션 기법 약속

Step 0. 초기화(current time=0)초기계획에 해당되는 우선순위, 모드, 잔여자원 등 모든 데이터를 기초 데이터로 설정한다.Goto Step 1.

Step 1. 다음 산정시점 탐색(Search Next Time)이전 current time보다 큰 조건부활동의 최소 종료시점을 찾고 current time을 종료시점으로 업데이트 한다. 이때 조건부활동의 종료시점은 $s_i + d_{i,m}$ 가 아닌 $s_i + dur_{i,m}$ 을 사용한다. Goto Step 2.찾을 조건부확률이 없으면 Goto Step 4.

Step 2. 조건부활동의 재작업 여부 확인current time에 종료되는 조건부활동이 재작업 통과확률과 랜덤 함수 값과의 비교를 통하여 재작업인지 확인하고, 재작업인 경우 시작시점이 current time인 조건부활동을 추가한다.Goto Step 3.

Step 3. 잔여 활동의 최적화 일정산정현재 활동 중 시작시점이 current time을 포함한 이후인 활동들만을 대상으로 최적화된 일정을 산정하여 배치한다.Goto Step 1.

Step 4. 현재 산정된 일정으로 프로젝트 종료까지 진행한 후 종료한다.

모의실험 결과

05



5. 모의실험 결과

5.1 초기계획 모의실험 조건 설정

5.1.1 데이터 생성 기본구성

기호	구분	데이터셋
N	활동의 수	10
M	모드의 수	2
K	자원 종류의 수	2
Due Date	수행완료일	50
Penalty	단위 당 패널티	5
Num_of_Sim	당 시뮬레이션 횟수	1000

- (1) 모드별 수행기간 및 수행비용 생성 : 5에서 10사이의 값을 임의로 생성
- (2) 자원의 종류와 활동/모드별 자원 요구량 : 5과 10사이의 값을 임의로 생성
- (3) 프로젝트 종료 시점(Project Completion Time) : 모든 활동이 종료된 시점으로 생성
- (4) 프로젝트 네트워크 : 프로젝트 네트워크는 노드와 노드 관계를 0.6의 확률로 연결하였으며, 선행 관계 맞도록 필요한 연결은 추가 및 필요 없는 연결은 모두 삭제하였다.



5. 모의실험 결과

5.1 모의실험 결과

5.2.1 초기계획과 잔여계획 산정의 모의실험 결과

데이터셋 3개 별로 각 10개씩, 총 30문제에 대하여 연산하는데 소요된 CPU 연산시간을 기록하였다

[표 5-2] 계획산정 테스트용 데이터셋 기본구성

데이터셋	활동의 수	모드의 수	자원의 수
1	10	2	2
2	15	2	2
3	20	2	2

[표 5-3] 초기계획 산정 소요시간 (단위 초)

문제번호	Set 1	Set 2	Set 3
1	0	0	0
2	0	0	0
3	0	0	0
4	0	0	0
5	0	0	0
6	0	0	0
7	0	0	0
8	0	0	0
9	0	0	0
10	0	0	0
평균	0	0	0
최대	0	0	0
최소	0	0	0

데이터셋 1에서 활동의 해 공간은 $10!=3,628,800$, 데이터셋 3에서 활동의 해공간은 $20!=2.4329E+18$ 로 $6.70443E+11$ 배의 차이를 보이나 휴리스틱 기법으로 인하여 모두 1초미만의 소요시간을 가진다.

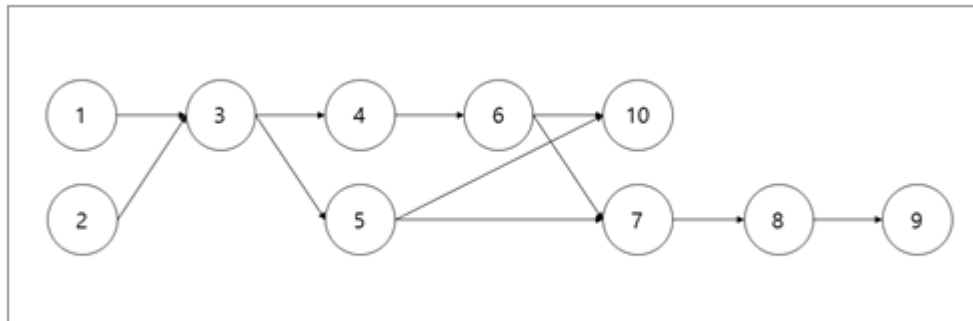


5. 모의실험 결과

5.1 모의실험 결과

5.2.2 가장 좋은 가중치 산정정책 를 추정하는 모의실험 결과

본 연구에서는 가장 좋은 가중치 산정정책 E값을 추정하기 위하여, 를 1부터 2.9까지 0.1씩 증가시키며 실험을 하였다. 각 값에서 초기계획 산정을 포함한 시뮬레이션을 각 1000회 실시하고 그 결과의 평균과 분산을 구하여 가장 좋은 값을 찾는다.



[그림 5-1] 데이터셋 1의 프로젝트 네트워크

[그림 5-1]은 데이터셋 1의 네트워크이며, 이 네트워크에서 모든 활동의 수 $N=10$, 모드의 수 $M=2$, 자원종류의 수 $K=2$, 마감기한은 70, 패널티는 5이며, 조건부 활동은 3, 5, 7 세 개로 재작업 없이 통과할 확률은 각각 0.8, 0.7, 0.8 이다.



5. 모의실험 결과

5.1 모의실험 결과

[표 5-5] 데이터셋 1의 모의실험 결과

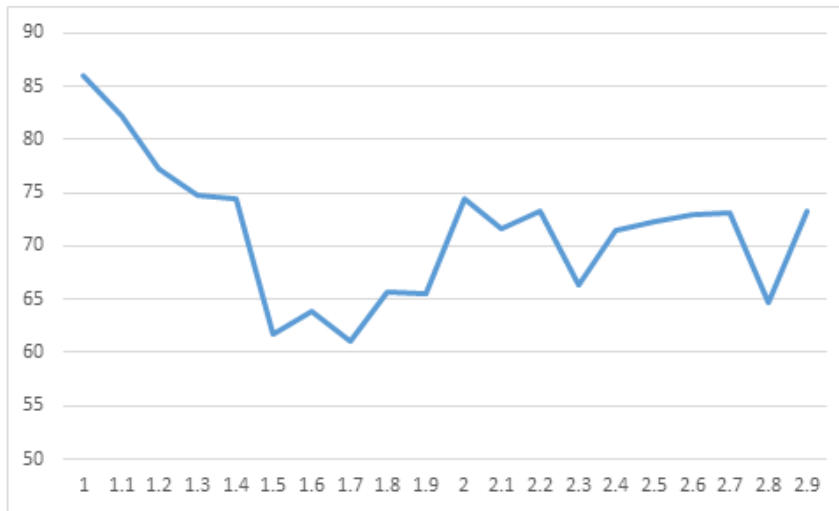
E	첫 번째 실험		두 번째 실험	
	총 비용 평균	총 비용 분산	총 비용 평균	총 비용 분산
1.0	85.98	2,577.61	71.26	1,567.68
1.1	82.16	2,306.07	80.99	2,096.40
1.2	77.24	1,757.76	68.87	1,034.43
1.3	74.75	1,282.91	74.48	1,209.42
1.4	74.48	1,532.50	64.83	1,619.81
1.5	61.80	710.76	74.01	1,124.42
1.6	63.89	921.59	72.57	941.59
1.7	61.04	470.52	63.45	559.57
1.8	65.68	586.22	74.35	1,129.92
1.9	65.52	540.83	73.61	1,142.52
2.0	74.36	1,173.76	72.27	937.67
2.1	71.69	862.93	72.05	967.37
2.2	73.30	1,132.42	73.03	1,060.80
2.3	66.34	532.51	66.21	568.62
2.4	71.48	968.09	72.33	1,048.76
2.5	72.29	1,024.19	72.01	1,060.37
2.6	72.92	1,003.15	72.71	861.12
2.7	73.14	1,110.25	73.38	1,131.67
2.8	64.77	404.25	73.29	905.40
2.9	73.30	1,197.24	66.30	546.86
평균	71.31	1,104.78	71.60	1,075.72
MAX	85.98	2,577.61	80.99	2,096.40
MIN	61.04	404.25	63.45	546.86

[표 5-5]의 결과로 데이터셋 1의 경우 두 실험 모두 E=1.7일 때 가장 낮은 값을 가진다.

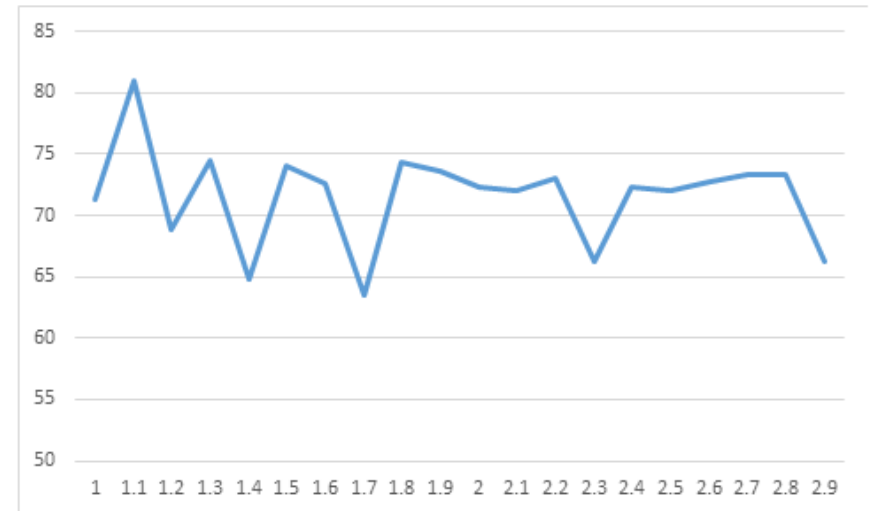


5. 모의실험 결과

5.1 모의실험 결과



[그림 5-2] Set 1에서 E 의 변화에 따른 첫 번째 실험의 총 비용의 평균



[그림 5-3] Set 1에서 E 의 변화에 따른 두 번째 실험의 총 비용의 평균

결론

06

6. 결론

- 본 논문에서는 '자원 제약을 고려한 프로젝트 일정문제'에 활동의 재작업 통과 확률을 적용하여 활동의 우선순위와 모드의 선택의 최선해를 구하는 휴리스틱 알고리즘을 제안하였다.
- 이는 기존의 프로젝트 관리의 영역에서 불확정적인 요소를 원가관리나 품질 관리, 리스크관리를 통해 풀어내었던 부분을 일정관리의 일정계획의 영역에서 적용할 수 있음을 의미하며, 모의실험을 통하여 그 가능성을 입증하였다.

6.1 연구의 결과

본 연구에서는 활동의 기간과 비용에 산정정책 값을 적용하여, 기존의 일정문제가 재작업이라는 불확정성을 고려하지 못했을 때의 시뮬레이션 평균 총비용과 적정한 값이 적용되었을 때의 시뮬레이션 평균 총비용을 모의실험을 통하여 비교하여, 적정한 값을 적용하는 것이 프로젝트 총 비용을 최소화 할 수 있다는 점을 입증하였다.

[표 6-1] E 에 따른 시뮬레이션 결과 총 비용 평균값

데이터셋	$E = 1$ 일 때 평균	적정한 E 의 값	적정한 E 에서 평균
1	86, 71.3	1.7	61, 63.5
2	35.8, 37.8	1.2 또는 1.9	33.8, 34
3	40, 35.9	1.4 또는 1.7	33.1, 35.41



6. 결론

6.2 연구의 시사점

- 본 연구에서는 '자원 제약을 고려한 프로젝트 일정문제'에 불확정적인 요소를 확률적으로 적용하는 방법을 제시하였으며, 프로젝트 일정문제에 정책적으로 적용할 수 있는 의사결정변수인 E값에 대한 시뮬레이션 결과를 통하여 추정할 수 있음을 보여줌으로써, 일정문제에서 보다 정확한 목적함수를 구하는 방법을 제시하였다.
- 또한, 의사결정변수의 영향을 재작업 통과확률이 100%가 아닌 활동에 한하여 0-1변수를 통하여 적용함으로써, 특정요소에만 적용되는 정책변수를 수식화 하였다. 이는 조건을 통해 특정한 산정요소에만 확률적 영향을 미치는 경우 수식화를 통하여 적용할 수 있음을 보여준다.



6. 결론

6.3 연구의 한계 및 향후 연구과제

- 본 논문에서는 불확정성이 확률로 적용되지 않는 경우보다, 확률로 적용하는 것이 더 나은 결과를 보여줄 수 있다는 점과 그 수식화에 초점을 맞추었다. 그로 인해, 모의실험을 통하여 제시된 결과 값과 초기계획의 예상 값의 오차를 줄이는 부분이 부족하게 연구되었다.
- 또한, 문제의 일반화된 증명을 위하여, 일반적이지 않은 특정 시뮬레이션(활동의 많은 반복이 일어난 경우)이 전체 시뮬레이션 평균에 영향을 주는 경우를 배제하지 않았다.
- 향후 본 연구를 확장하기 위해서는 불확정성을 일정에 좀 더 세밀하게 적용할 수 있는 연구가 필요하며, 다른 한편으로 본 논문과 같이 확률이 적용된 시뮬레이션에서 표본의 평균과 너무 동떨어진 값이 있는 경우 배제하여 연구하는 것과 같이 방법론적인 진화가 필요하다. 또한 재작업과 같이 계획과 실제와의 오차를 만드는 많은 불확정적인 요소들을 적용할 수 있는 아이디어와 연구, 그리고 최종적으로 결과에 가까운 일정계획을 만들 수 있는 연구가 필요하다.

참고 문헌

07



7. 참고 문헌

[국내 문헌]

강금식, 정우석(2008), 경영과학, 서울: 범영사.

권오운(2018), 기술지도사[생산계획편], 서울: (주)ATPM컨설팅

김갑식(2016), 자원 제약을 고려한 프로젝트 일정계획에서 활동의 품질과 기간과 및 비용 간 트레이드오프문제: 선택 가능한 모드가 복수인 경우, 송실대학교 대학원 박사학위 논문.

김찬목, 박영원, "재작업과 불확실성을 고려한 일정 시뮬레이션 방법론", 한국철도학회 논문집, 12(1), 135-143

남재덕(2008), 프로젝트 포트폴리오 평가 문제: 불확실성 하의 최초 구성 및 주기적 개장을 중심으로, 송실대학교 대학원 박사학위 논문

류미애(2017), 자원제약을 고려한 다단계 프로젝트 일정 문제, 송실대학교 대학원 논문

민택기(2005), 프로젝트 포트폴리오 평가문제: 시뮬레이션을 이용한 다속성 평가를 중심으로, 송실대학교 대학원 박사학위 논문.

안태호(1998), "자원 제약을 고려하여 기간 단축이 가능한 복수의 양식을 지닌 단일 프로젝트 일정 문제: 자원 가용량이 시간에 따라 변하는 경우의 휴리스틱 해법," 한국 OA학회지, 3(4), 154-163.

이상법(2000), 현대 생산·운영관리 제2판, 서울: 도서출판 명경사.

이현기(2017), 자원 제약을 고려한 프로젝트 일정문제: 기간 단축이 가능한 복수의 모드를 갖고 자원의 가용량을 변경할 수 있는 경우, 송실대학교 대학원 박사학위 논문.



7. 참고 문헌

[국외 문헌]

- Ahn, T., & Erenguc, S. S. (1995), "The resource constrained project scheduling problem with multiple crashable modes: An exact solution method," Working Paper Series in Department of Decision and Information Sciences, University of Florida, 95-101.
- Ahn, T., & Erenguc, S. S. (1998), "The resource constrained project scheduling problem with multiple crashable modes: a heuristic procedure," European Journal of Operational Research, 107(2), 250-259.
- Bie, L., Cui, N., & Zhang, X. (2012), "Buffer Sizing Approach With Dependence Assumption Between Activities In Critical Chain Scheduling," International Journal of Production Research, 50(24), 7343-7356.
- Blazewicz, J. (1978), "Complexity of computer scheduling algorithms under resource constraints," In Proc. First Meeting AFCET-SMF on Applied Mathematics, 16(6), 169-178.
- Bowman, E. H. (1959), "The schedule-sequencing problem," Operations Research, 7(5), 621-624.
- Davis, E. W. (1973) "Project scheduling under resource constraints historical review and categorization of produres," AIIE Transactions, 5(4), 297-313
- Davis, E. W., & Heidorn, G. E. (1971), "An Algorithm For Optimal Project Scheduling Under Multiple Resource Constraints," Management Science, 17(12), B-803.



7. 참고 문헌

[국외 문헌]

- Debels, D., & Vanhoucke, M. (2007), "A decomposition-based genetic algorithm for the resource-constrained project-scheduling problem," *Operations Research*, 55(3), 457-469.
- Doersch, R. H., & Patterson, J. H. (1977), "Scheduling a project to maximize its present value: a zero-one programming approach," *Management Science*, 23(8), 882-889.
- Etgar, R., Gelbard, R., & Cohen, Y. (2017), "Optimizing Version Release Dates Of Research And Development Long-Term Processes," *European Journal of Operational Research*, 259(2), 642-653.
- Herroelen, W., & Demeulemeester, E. (1992), "Recent Advances In Branch-And-Bound Procedures For Resource-Constrained Project Scheduling Problems," *Katholieke Universiteit Leuven, Departement Toegepaste Economische Wetenschappen*, 1-31.
- Herroelen, W., & Leus, R. (2001), "On The Merits And Pitfalls Of Critical Chain Scheduling," *Journal of Operations Management*, 19(5), 559-577.
- Johnson, T. J. R. (1967), *An algorithm for the resource constrained project scheduling problem*, unpublished Ph.D. Dissertation, Massachusetts Institute of Technology, Cambridge.
- Kapur, P. K., Pham, H., Aggarwal, A. G., & Kaur, G. (2012), "Two Dimensional Multi-Release Software Reliability Modeling And Optimal Release Planning," *IEEE Transactions on Reliability*, 61(3), 758-768.



7. 참고 문헌

[국외 문헌]

- Kelley, J. E. (1961), "Critical-Path Planning And Scheduling: Mathematical Basis," *Operations Research*, 9(3), 296-320.
- Kelley, J. E., & Walker, M. R. (1959), "Critical-path Planning And Acheduling", In *Papers Presented At The December 1-3, Eastern Joint IRE-AIEE-ACM Computer Conference*, 160-173.
- Lova, A. L., Tormos, M. P., & Barber, F. (2006), "Multi-Mode Resource Constrained Project Scheduling: Scheduling Schemes, Priority Rules And Mode Selection Rules," *Inteligencia Artificial, Revista Iberoamericana de Inteligencia Artificial*, 10(30), 69-86.
- Lu, M., & AbouRizk, S. M. (2000), "Simplified CPM/PERT Simulation Model," *Journal of Construction Engineering and Management*, 126(3), 219-226.
- Mota, A. A., Mota, L. T. M., & Morelato, A. (2007), "Visualization Of Power System Restoration Plans Using CPM/PERT Graphs," *IEEE Transactions on Power Systems*, 22(3), 1322-1329.
- Nudtasomboon, N., & Randhawa, S. U. (1997), "Resource-Constrained Project Scheduling With Renewable And Non-Renewable Resources And Time-Resource Tradeoffs," *Computers and Industrial Engineering*, 32(1), 227-242.
- Patterson, J. H. (1973), "Alternate methods of project scheduling with limited resources," *Naval Research Logistics Quarterly*, 20(4), 764-784.



7. 참고 문헌

[국외 문헌]

- Patterson, J. H., & Huber, W. H. (1974), "A horizon-varying, zero-one approach to project scheduling," *Management Science*, 20(6), 990-998.
- Patterson, J. H., & Roth, G. (1976), "Scheduling a project under multiple resource constraints: a zero-one approach," *AIIE Transactions*, 8(4), 449-456.
- Patterson, J. H., Slowinski, R., Talbot, F. B., & Weglarz, J. (1990), "Computational experience with a backtracking algorithm for solving a general class of precedence and resource constrained scheduling problems," *European Journal of Operational Research*, 49(1), 68-79.
- PMBOK (2017), *A Guide To The Project Management Body Of Knowledge—sixth Edition*, Pennsylvania: Project Management Institute, Inc.
- Ponz-Tienda, J. L., Yepes, V., Pellicer, E., & Moreno-Flores, J. (2013), "The resource leveling problem with multiple resources using an adaptive genetic algorithm," *Automation in Construction*, 29(1), 161-172.
- Scharge, L. (1970), "Solving resource-constrained network problems by implicit enumeration—non-preemptive case," *Operation Research*, 18(2), 263-278.
- Słowiński, R., Soniewicki, B., & Węglarz, J. (1994), "DSS for multi-objective project scheduling," *European Journal of Operational Research*, 79(2), 220-229.
- Stinson, J. P., Davis, E. W., & Khumawala, B. M. (1978), "Multiple resource-constrained scheduling using branch and bound," *AIIE Transactions*, 10(3), 252-259.



7. 참고 문헌

[국외 문헌]

- Tara, H. A. (2011), Operation Research: An Introduction-9th edition, Chicago: Prentice Hall.
- Van Peteghem, V., & Vanhoucke, M. (2010), "A Genetic Algorithm For The Preemptive And Non-Preemptive Multi-Mode Resource-Constrained Project Scheduling Problem," European Journal of Operational Research, 201(2), 409-418.
- Wagner, H. (1959), "An integer programming model for machine scheduling," Naval Research Logistics Quarterly, 6(1), 131-140.
- Yang, K. K., Talbot, F. B., & Patterson, J. H. (1993), "Scheduling a project to maximize its net present value: an integer programming approach," European Journal of Operational Research, 64(2), 188-198.



경청해 주셔서 감사 합니다.



Q&A