



## 저작자표시-비영리-변경금지 2.0 대한민국

이용자는 아래의 조건을 따르는 경우에 한하여 자유롭게

- 이 저작물을 복제, 배포, 전송, 전시, 공연 및 방송할 수 있습니다.

다음과 같은 조건을 따라야 합니다:



저작자표시. 귀하는 원저작자를 표시하여야 합니다.



비영리. 귀하는 이 저작물을 영리 목적으로 이용할 수 없습니다.



변경금지. 귀하는 이 저작물을 개작, 변형 또는 가공할 수 없습니다.

- 귀하는, 이 저작물의 재이용이나 배포의 경우, 이 저작물에 적용된 이용허락조건을 명확하게 나타내어야 합니다.
- 저작권자로부터 별도의 허가를 받으면 이러한 조건들은 적용되지 않습니다.

저작권법에 따른 이용자의 권리는 위의 내용에 의하여 영향을 받지 않습니다.

이것은 [이용허락규약\(Legal Code\)](#)을 이해하기 쉽게 요약한 것입니다.

[Disclaimer](#)

박사 학위 논문

## 자원 제약을 고려한 프로젝트 일정 문제

- 재작업 가능한 활동이 있는 경우 -

## Resource Constrained Project

## Scheduling Problem

- with Reworkable Activities -

2019년 12월

승실대학교 대학원

경영학과

백 인 섭



박사 학위 논문

## 자원 제약을 고려한 프로젝트 일정 문제

- 재작업 가능한 활동이 있는 경우 -

## Resource Constrained Project Scheduling Problem

- with Reworkable Activities -

2019년 12월

숭실대학교 대학원

경영학과

백 인 섭

박사학위 논문

## 자원 제약을 고려한 프로젝트 일정 문제

- 재작업 가능한 활동이 있는 경우 -

지도교수 안 태 호

이 논문을 박사학위 논문으로 제출함

2019년 12월

승실대학교 대학원

경영학과

백 인 섭

백 인 섭 의 박사 학 위 논 문 을 인 준 함

심 사 위 원 장

유 한 주



심 사 위 원

최 정 인



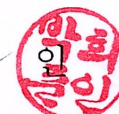
심 사 위 원

박 종 우



심 사 위 원

백 리 정



심 사 위 원

한 태 호



2019년 12월

숭실대학교 대학원

## 목 차

국문초록 .....	v
영문초록 .....	vii
제 1 장 서론 .....	1
1.1 연구 배경 및 목적 .....	1
1.2 연구의 방법 .....	3
1.3 연구의 구성 .....	4
제 2 장 이론적 배경 .....	5
2.1 프로젝트 일정문제 선행연구 .....	5
2.2 RCPSP관련 선행연구 .....	9
2.3 재작업관련 선행연구 .....	16
제 3 장 문제의 모형 .....	19
3.1 문제의 개요 .....	19
3.2 문제의 가정 .....	19
3.3 문제의 수식화 .....	22
제 4 장 알고리즘 .....	26
4.1 시점 $t$ 에서의 프로젝트 일정 문제 .....	28
4.2 해법 .....	33

제 5 장 모의실험 결과 .....	40
5.1 모의실험 조건 설정 .....	41
5.2 모의실험 결과 .....	42
5.2 모의실험 결과 요약 .....	60
 제 6 장 결론 .....	 61
6.1 연구의 결과 .....	61
6.2 연구의 시사점 .....	64
6.3 연구의 한계 및 향후 연구과제 .....	65
 참고문헌 .....	 68



## 표 목 차

[표 5-1] 모의실험 데이터셋 기본구성 .....	41
[표 5-2] 데이터셋 1의 재작업 가능 활동과 재작업 확률 .....	43
[표 5-3] 데이터셋 1의 기간 및 비용 .....	44
[표 5-4] 데이터셋 1의 필요자원 .....	45
[표 5-5] 데이터셋 1의 모의실험 결과 .....	46
[표 5-6] 데이터셋 2의 재작업 가능 활동과 재작업 확률 .....	49
[표 5-7] 데이터셋 2의 기간 및 비용 .....	50
[표 5-8] 데이터셋 2의 필요자원 .....	51
[표 5-9] 데이터셋 2의 모의실험 결과 .....	52
[표 5-10] 데이터셋 3의 재작업 가능 활동과 재작업 확률 .....	55
[표 5-11] 데이터셋 3의 기간 및 비용 .....	56
[표 5-12] 데이터셋 3의 필요자원 .....	57
[표 5-13] 데이터셋 3의 모의실험 결과 .....	58
[표 6-1] $M$ 의 변화에 따른 총 비용의 변화 .....	63
[표 6-2] $M=0$ 일 때와 최저 평균 $M$ 일 때의 비율 .....	63

## 그 립 목 차

[그림 5-1] 데이터셋 1의 프로젝트 네트워크 .....	43
[그림 5-2] 데이터셋 1의 총 비용 그래프 .....	46
[그림 5-3] 데이터셋 1의 활동비용 합 그래프 .....	47
[그림 5-4] 데이터셋 1의 지체보상금 그래프 .....	47
[그림 5-5] 데이터셋 2의 프로젝트 네트워크 .....	49
[그림 5-6] 데이터셋 2의 총 비용 그래프 .....	52
[그림 5-7] 데이터셋 1의 활동비용 합 그래프 .....	53
[그림 5-8] 데이터셋 2의 지체보상금 그래프 .....	53
[그림 5-9] 데이터셋 3의 프로젝트 네트워크 .....	55
[그림 5-10] 데이터셋 3의 총 비용 그래프 .....	58
[그림 5-11] 데이터셋 3의 활동비용 합 그래프 .....	59
[그림 5-12] 데이터셋 3의 지체보상금 그래프 .....	59

## 국문초록

# 자원 제약을 고려한 프로젝트 일정 문제

- 재작업 가능한 활동이 있는 경우 -

백인섭

경영학과

승실대학교 대학원

PERT/CPM(Program Evaluation & Review Technique/Critical Path Method)과 같은 프로젝트 일정문제(Project Scheduling Problem)는 프로젝트에서 일정을 계획하고 기간을 추정할 때 사용할 수 있는 과학적 의사결정 모델이다. 특히, 자원 제약을 고려한 프로젝트의 일정 문제(Resource Constrained Project Scheduling Problem, RCPSP)는 자원 제약을 반영한 모델이며, 이 모델을 사용할 경우, 최소비용에 최적화된 작업(활동)의 시작시기와 작업(활동)의 수행방법을 선택할 수 있다.

그러나 RCPSP는 확정적 모형(Deterministic Model)이기 때문에 일반기업의 프로세스에서 흔히 볼 수 있는 재작업과 같은 불확실성을 가진 확률적 요소를 직접 적용하기 어렵다. 따라서 RCPSP에 재작업을 적용할 수 있는 수학적 모델과 알고리즘의 개발이 필요하다.

본 연구에서는 RCPSP에 재작업을 추가한 수학적 모형을 만들고, 적용할 수 있는 알고리즘을 제시하고자 하며, 모의실험을 통해 모델이 유효함

을 입증하고자 하였다. 모의실험결과에 따르면, 재작업이 적용되지 않은 RCPSP 알고리즘으로 작성된 일정보다 재작업이 적용된 RCPSP 알고리즘으로 작성된 일정이 동일한 실험조건하에서 더 좋은 결과를 보였다.

본 연구의 특징은 RCPSP라는 확정적 모델에 확률적 요소를 적용하였다는 점이다. 이는 RCPSP를 사용하여 일정을 작성할 때 확률적인 요소도 적용할 수 있도록 모델을 확장하였다는 점에 그 의의가 있다.

## ABSTRACT

# Resource Constrained Project Scheduling Problem

– with Reworkable Activities –

In-soub Paek

Department of Business Administration

Graduate School Soongsil University

Project scheduling problems such as PERT/CPM(Program Evaluation & Review Technique/Critical Path Method) are scientific decision making models that can be utilized when scheduling a project. Especially, RCPSP(Resource Constrained Project Scheduling Problems) are models that reflect resource constraints, and when used, it is possible to know the optimal activity starting point and the optimal mode to choose from that allows minimum cost.

However, for RCPSP being a deterministic method, it is hard to incorporate stochastic elements such as rework which can easily be found in most corporate processes. Thus, there is the needs for a RCPSP mathematical model and algorithm that rework can be

incorporated.

This study planes to come up with a RCPSP mathematical model that incorporates rework, suggest an algorithm that the model can be applied with, confirm the validity of the model with simulations. Based on the simulation results, schedules drawn with the algorithm of this study showed better results than the schedules drawn by RCPSP algorithms without the element of rework under same conditions.

This study has a distinction that it incorporated a stochastic element in a deterministic model. Its significance is that it expanded the model, allowing the application of stochastic elements when scheduling with RCPSP models.

# 제 1 장 서 론

## 1.1 연구의 배경 및 목적

일반적으로 프로젝트는 상대적으로 짧은 기간과 자원의 통제가 불충분하기 때문에 공식적이고 자세한 계획(Planning)이 필요하다. 프로젝트 환경에서 계획은 예측된 환경 내에서 미리 정해진 행동 과정을 확립하는 것으로 기술할 수 있다. 계획의 중요한 구성요소 중 하나인 일정(Scheduling)은 활동이 언제 시작 또는 완료되는지를 보여주는 것이며, 일정의 작성에서 일정기술(Scheduling techniques)은 경쟁이 치열한 산업의 특징인 프로젝트의 복잡성과 대량의 데이터 및 엄격한 마감기한을 대처하는데 도움이 된다(Kerzner, 2017).

프로젝트 일정 문제의 대표적인 기법들로는 PERT(Program Evaluation & Review Technique), CPM(Critical Path Method), TCTP(Time-cost trade-off problem), RCPSP(Resource Contained Project Scheduling Problem), GERT(Graphical Evaluation & Review Technique) 등을 꼽을 수 있으며, CPM, TCTP, RCPSP 모델은 일반적으로 불확실성이 없는 확정적(Deterministic)인 경우를 다루며 PERT, GERT 모델은 불확실성이 있는 확률적(Stochastic)인 경우를 다룬다.

PERT는 활동의 수행기간에 불확실성이 있는 경우 활동에 대한 정확한 지식을 바탕으로 각 활동의 수행기간을 세 가지(낙관적, 비관적, 가장가능성 높은)의 수행기간으로 추정한 후 하나의 기댓값으로 계산하는 방법이며(Malcolm, 1959), CPM은 모든 경로에서 여유기간이 없는 주경로(Critical Path)를 찾는 방법으로 주경로에 있는 활동의 수행기간은 각 활동의 기간이 이산적인 경우 프로젝트의 총 기간에 직접적으로 관여된다(Kelley & Walker, 1959). GERT는 확률적 네트워크를 분석하기 위한 기

술로 네트워크의 논리를 몬테카를로 시뮬레이션을 이용하여 분석하는 방법이다(Pritsker, 1966).

일반적인 기업에서 프로젝트 일정을 계획하는 경우 가장 현실적인 대안은 RCPSP일 것이다. RCPSP 중 많이 연구되고 있는 복수의 모드를 지닌 문제(Resource Constrained Project Scheduling Problem with Multi-Modes, RCPSPMM)의 경우 작업(활동)의 시작시간과 그 실행방법(모드)의 선택을 의사결정 변수로 가지고 있으며, 자원이 한정된 경우를 다루고 있기 때문에 매우 현실적이다. Sprecher(2002)는 선행나무(Precedence Tree) 개념을 사용하여 작업(활동)의 시작시간과 실행방법(모드)의 선택을 해로 선정하였으며, 모든 해를 탐색하는 전체열거법(Total Enumeration)을 통해 최적의 해를 구하는 RCPSPMM을 소개하였다.

일반적으로 RCPSP는 확정적인 경우를 다룬다. 그러나 기업의 프로세스에는 확정적인 요소만 있는 것은 아니다. 불확정적 요소도 존재하고 있으며, 불확정적 요소 중 하나인 재작업은 일반적인 기업의 프로젝트에서도 흔히 관찰된다. 예를 들어 Reichelt & Lyneis(1999)에 따르면 항공우주프로젝트에서 초기 설계도면이 완료된 후 대부분의 작업이 재작업 되었으며, 일부 도면은 8번 개정 되었지만 일반적으로 3에서 5회 재 발행된다고 하였으며, 프로젝트에서 예산과 일정이 초과되는 원인 중 하나라고 하였다.

RCPSP의 영역에서도 불확정적 요소를 적용하기 위해 RCPSP의 확률적 확장모델인 SRCPSP(Stochastic Resource Constrained Project Scheduling Problem)에 대한 연구가 진행되고 있으나, SRCPSP는 일반적으로 기준일정(Baseline Schedule)은 생성되지 않으며, 총 기간(Makespan)의 최소화를 목적으로 한다(Rostami et al., 2018).

본 논문에서는 RCPSP에 재작업이라는 불확정적 요소를 추가하는 방법



을 찾고, 재작업이 추가된 RCPSP의 목적함수값을 최소화 하는 방법을 연구하고자 한다.

## 1.2 연구의 방법

일반적인 RCPSP는 활동의 기간이나 비용이 확정적인 경우를 대상으로 한다. 그러나 현실에서 프로젝트를 수행할 경우 재작업이란 흔히 관찰되는 요소이다. 따라서 본 연구에서는 RCPSP 모형에 재작업을 추가하는 모형을 개발하고자 한다.

현실에서 재작업이란 여러 가지를 의미할 것이다. 예를 들어 작업 A가 최초 1회에 기간이 100일 경우, 작업이 끝난 뒤 검수결과에 따라 재작업이 발생하였다고 하자. 이때 작업 A의 일부 수정이 필요해서 재작업으로 10일이 추가될 수도 있고, 작업 A의 전체 수정이 필요해서 재작업으로 처음부터 다시 하는 경우도 있을 것이다. 일부 수정과 같이 지연기간이 기존기간의 일부에 해당하는 경우는 PERT등에서 미리 계획하거나 분석할 수는 있으나 전체수정이 한번에서 여러 번까지 반복해서 발생하는 경우는 다루기 어려운 것이 현실이다. 따라서 일반적인 논문에서는 활동의 시행횟수가 1인경우만 다루지만 본 논문에서는 활동의 수는 동일하지만 시행횟수가 자연수(1, 2, 3, ...)로 이루어진 사례를 중점적으로 다루고자 한다.

본 논문에서 재작업이란 활동기간의 단축이나 지연을 의미하지 않으며, 활동의 종료 시 검수를 통해 발생하는 프로세스의 반복으로 정의한다. 또한 본 논문의 모의실험에서는 활동의 수행횟수가 확률적인 경우를 다룬다.

이에 본 논문에서는 RCPSP에 재작업을 가상으로 적용하기 위해 (1) 최소비용 RCPSP에 재작업이 적용된 문제를 수리적으로 모델링하고, (2) 알고리즘을 제시하며, (3) 알고리즘을 코드로 전환한 후 (4) 모의 데이터를 생성하고 모의실험을 수행한다. 모의실험의 결과를 통해 제시된 기법이 실

제로 작동함과 유효함을 입증한다.

### 1.3 연구의 구성

본 연구는 총 6장으로 구성되어 있다, 제1장에서는 연구배경 및 목적과 연구의 방법, 연구의 구성을 서술하였고, 제2장은 본 연구에 관련된 이론적 배경으로 본 연구에 대한 선행연구를 살펴보았으며, 제3장에서는 본 연구의 수학적 모형으로 제시하였다. 제4장에서는 사용된 수학적 모형의 최적해를 구하기 위해 사용된 알고리즘을 약술하였으며, 이어 제5장에서는 모의실험 결과를 기술하였다. 마지막으로, 제6장에서는 실험 결과를 중심으로 연구의 결과를 요약하고, 그 한계 및 향후 연구방향을 제시하였다.

## 제 2 장 이론적 배경

본 연구의 주제는 프로젝트 일정문제의 하나인 RCPSP(Resource Contained Project Scheduling Problem)에 활동의 재작업을 적용한 모형을 만들고 그 유효성을 입증하는 것이다. 따라서 프로젝트 일정문제와 RCPSP, 불확실성 및 재작업관련 문헌연구를 하였다.

### 2.1 프로젝트 일정문제 선행연구

프로젝트에 대한 여러 정의중 하나는 Project Management Institute [PMI](2017)에서는 ‘고유한 제품, 서비스 또는 결과물을 산출하기 위해 한 시적으로 투입하는 노력’이라고 정의하고 있다. 이와 같이 프로젝트는 고유한 제품, 서비스 또는 결과물의 산출이라는 특성을 가지고 있기 때문에 태생적으로 불확실성을 산출할 수밖에 없다. 따라서 효과적이고 효율적으로 프로젝트를 실행할 수 있도록 관리하려면 ‘한시적인 기간’에 대한 프로젝트 일정의 계획과 관리가 매우 중요할 것이다. 즉, 프로젝트의 성공적 완료를 위해서는 명확한 업무 범위 설정과 제한된 자원의 활용, 기간 내 완수를 위한 일정 계획 수립, 프로젝트에 내포되어 있는 불확실성을 줄이기 위한 신속한 의사결정이 핵심 요인이라고 할 수 있다.

프로젝트에서 일정이란 각 활동의 시작시간과 완료일을 정하는 것이 기본일 것이다. 이와 같은 요소를 계획하는 것을 일정계획이라 할 때, 일정 계획을 개발하는 방법으로서 PMI(2017)에서는 일정 네트워크 분석, 주공정법(CPM, Critical Path Method), 자원최적화, 일정단축, 분석기법 등을 제시하고 있다.

### 2.1.1 PERT/CPM(Program Evaluation & Review Technique/Critical Path Method)

1958년 미국 해군에 의해 폴라리스 미사일(Polaris Missile) 프로젝트의 일정 계획 및 통제를 위해 개발된 PERT와, 1959년 Kelley & Walker(1959)에 의해 개발된 CPM은 그 수행기간이 각각 확정적인지 확실적인인지로 구분할 수 있다(Tara, 2011).

CPM은 최소 완료시간과 주경로 활동(Critical Activity)을 구하는 것을 중점으로 제시된 네트워크 기반의 일정기법으로, 주경로 활동이란 해당 활동의 지연이 전체 완료시간에 직접적으로 지연되는 활동들을 말한다. 반대로 주경로 활동이 아닌 활동(Noncritical Activity)은 여유시간(Slack)내의 지연은 전체 완료시간에 영향을 미치지 않는 활동들을 말한다(Kelley & Walker, 1959). PMI(2017)에서 주공정법은 ‘프로젝트의 최소 기간을 산정하고 일정모델의 논리 네트워크 경로에서 일정의 유연한 조정이 허용되는 기간을 결정하는 데 사용한다.’라고 설명한다.

이러한 주공정법은 1957년 미국 래밍톤-랜드(Remington-rand)사의 켈리(J. E. Kelley)와 듀퐁(Du Pont)사의 워커(M. R. Walker)에 의해 개발되었다. 주공정법은 프로젝트 각 활동 시간을 확정하고, 시간과 비용을 고려하는 것으로, 비슷한 시기에 개발된 PERT(Program Evaluation and Review Technique)와 유사하다.

Debels & Vanhoucke(2007)에 따르면 주공정법은 상호활동을 진행하는 모든 프로젝트에 사용 가능하며, 건축, 소프트웨어 개발, 연구 프로젝트, 제품 개발 등 모든 형태의 프로젝트에 사용되고 있다고 하였다.

PERT는 일정 계획 시 활동의 수행기간을 확률적으로 추정하여 사용한다. 그 방법으로 낙관치(Optimistic Time), 최빈치(Most likely Time), 비관치(Pessimistic Time)를 추정한 후 1:4:1의 비율로 활동의 평균 수행기

간을 산정하는 확률적 모형으로, 비용을 고려하지 않으며, 각 활동기간의 기댓값과 분산을 구하고 일반적으로 중심극한의 정리를 적용하여 사용한다(이상법, 2000).

Lu & AbouRizk(2000)는 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 제시하였다.

Mota et al.(2007)는 전력 시스템의 정전 사고 이후 복원 계획 수립에 PERT/CPM 그래프 이론을 기반으로 한 방법론을 제안하였다.

Herroelen & Leus(2001)는 주공정일정(CC, Critical Chain-scheduling) 및 여유 시간관리(BM, Buffer Management)로 알려진 프로젝트 관리에 제약이론을 적용하였다.

Bie et al.(2012)는 활동 간 의존도가 프로젝트 기간 성과에 미치는 영향을 분석함으로써, 활동 간 종속성을 가정하여 버퍼 크기를 결정하는 방법을 제시하였다.

### 2.1.2 자원 최적화 기법(Resource Optimizing Technique)

자원최적화는 작업에 필요한 자원의 점유 또는 사용을 자원의 가용량이 하로 조정하기 위해 사용되며, PMI(2017)에서는 자원의 수요와 공급에 따라 일정 모델을 조정하는데 사용할 수 있는 자원최적화기법의 예로 자원 평준화(Resource Levelling)와 자원평활화를 제시하고 있다. 자원 평준화는 자원의 가용량과 작업의 자원 요구량 사이에 균형을 유지하면서, 자원 제약에 근거하여 시작시점과 종료시점을 조정하는 기법으로, 동일한 기간에 한 종류의 자원을 두개 이상의 활동에서 사용할 경우 또는 자원의 사용량을 일정한 수준으로 유지해야 하는 경우에 사용하며 이로 인해 주공정

이 변경될 수 있다. 자원평활화는 프로젝트에 대한 자원 요구사항이 미리 정해진 자원한도를 초과하지 않도록 일정모델의 활동을 조정하는 기법이며 자원평준화와는 반대로 일부 자원이 최적화 되지 않을 수 있으나, 주공정을 변경하지 않는다.

이 분야의 관련 연구로는 분지(Branch) 프라이스 앤 컷(Price-and-Cut) 알고리즘을 활용한 Coughlan et al.(2010), 메타휴리스틱 방식을 활용한 Alsayegh & Hariga(2012)와, 유전자 알고리즘을 활용한 Li & Demeulemeester(2016)이 있다(이현기, 2017).

### 2.1.3 일정단축 기법(Schedule Compression)

일정단축기법은 일정에 관련된 목표를 충족하기 위해 일정의 기간을 단축하거나 시작시간을 앞당기기 위해 사용한다. PMI(2017)에서는 일정 단축기법으로 공정압축법(Crashing)과 공정중첩 단축법(Fast Tracking)을 예로 들고 있다. 공정압축법은 자원을 보충하여 최소한의 추가 비용으로 일정을 단축하기 위해 사용하는 기법이며, 반드시 실행 가능한 대안을 제시하지 않거나 리스크 및 원가의 상승을 초래할 수도 있다. 공정중첩 단축법은 일반적으로 순차적으로 수행되는 활동 또는 단계를 특정구간에서 동시에 수행하는 방식으로 재작업과 리스크의 증가를 초래할 수 있으며, 주공정에서 기간을 단축하기 위해 활동을 중첩할 수 경우에만 효과적인 기법이다(PMI, 2017).

### 2.1.4 시뮬레이션 기법(Simulation)

프로젝트 일정수립에서 시뮬레이션기법은 불확실성 원인을 사용하여 전체 프로젝트에 대한 가능한 일정 결과를 계산할 수 있으며 가장 일반적인 시뮬레이션 기법은 몬테카를로 분석이다. 또한 프로젝트의 목표 달

성에 미칠 잠재적 영향을 평가하기 위해 불확실성 유발 근원의 종합적 영향을 모델링한다(PMI, 2017).

Jerry(1984)에 따르면 시뮬레이션은 모든 산업과 생활에 제한 없이 적용가능하다고 하였으며, 실제 상황이나 시스템에서 시간의 흐름과 유사한 환경을 만들어 이를 모방한 것을 말한다. 그러므로 시뮬레이션은 실제 상황과 얼마나 유사하게 모델 구현했는지가 실험 성공의 관건이라 하였다.

시뮬레이션을 이용하면 생산 프로젝트에서는 생산 진행 시 발생 할 수 있는 재작업, 고장 등에 대한 확률모델을 적용하여 반복 실험을 진행할 수 있다(Zhang et al., 2007). 이러한 시뮬레이션 기반의 최적화 방법은 피드백 루프를 사용하는데 이때 시뮬레이션을 통해 매개변수를 받아서 사용하는 방법으로 적용할 수 있다(Werner et al., 2006). 이런 시뮬레이션 기반의 최적화 방법은 피드백 루프를 통해 이루어지는데 최적화를 할 때 시뮬레이션을 통해 얻은 매개변수를 사용하는 방법으로 적용된다(Carson & Maria, 1997; Werner et al., 2006).

## 2.2 RCPSP관련 선행연구

### 2.2.1 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. 이 문제의 경우 실제의 프로젝트 일정계획에 적용할 수 있는 프로젝트 일정문제 중 가장 현실적인 제약조건들로 이루어져 있다. 이 문제의 해결을 위해 이전 연구에서는 한계열거법, 분지한계법, 선형계획법, 정수계획법, 2차 방정식, 0-1 계획법, 암묵적열거법 등을 사용하였

다.

한계열거법은 Davis & Heidorn(1971)가 최적해법을 제안함으로 시작되었다(민택기, 2005). 장점으로 작업의 소요기간에 따라 자원 요구가 변화되며 작업의 연속성에 관한 다양한 가정들이 허용되며, 단점으로 문제의 크기가 증가함에 따라 급격히 해 공간이 증가한다는 점이다(남재덕, 2008). 분지한계법(Branch and Bound)은 Johnson(1967)과 Schrage(1970)에 의해 연구되었으며, 선형계획법은 Stinson et al.(1978)이 노드 선택 휴리스틱을 이용하여 수식화 하였다. 이들은 분지한계트리(Branch and Bound Tree) 상에서 열등 해를 효율적으로 제거하는 우월 규칙(Dominance Rule)을 수립하여 효율성을 높였다. 정수계획법은 Wagner(1959)가 기계 일정 계획 문제를 연구에서 수식화 하였다.

0-1계획법은 주어진 제약조건에 대해 0 또는 1로 만드는 변수를 곱하는 것으로 해당 제약을 조건부로 적용함으로 주어진 문제에 대한 수식화를 표현할 수 있으며, 효율적인 수식화는 문제를 표현하기 위해 최소화된 제약 또는 적은 수의 0-1변수를 가진다. 이는 제약과 0-1변수로 변수의 정의가 측정되는 만큼 그 결과가 문제의 구조와 연관되며, 그 만큼 중요하다(Patterson & Roth, 1976). 0-1계획법은 Bowman(1959)이 활동의 수행에 이용하였고, Patterson & Huber(1974)는 정수 선형 프로그램(Linear Program)의 수식화에 이용하였으며, Patterson & Roth(1976)는 순서 문제(Sequencing Problem)의 수식화에 이용하였다.

다중 자원 제약하의 프로젝트 일정문제(MRCPSP, Multi-modes Resource Constrained Project Scheduling Problem)의 경우 각 활동이 자원 및 기간 제약조건과 함께 모드 집합 중 하나에서 수행될 수 있음을 의미한다(Slowinski et al., 1994).



### 2.2.2 NP-hard Class

JSP(Job-shop Scheduling Problem)는 가장 잘 알려진 조합 최적화 문제 중 하나이며, 일정계획 문제를 해결하는 것으로 제품별 진행해야 하는 기계 또는 수행해야 할 활동이 결정되어있는 경우이다(안태호, 1998). 이러한 JSP는 NP-hard Class로 분류된다(Garey et al., 1976; 안태호, 1998). 즉 전통적인 JSP를 일반화하면 모든 작업을 모든 기계에서 진행할 수 있는 구조의 문제이며 이는 RSPSP와 같은 일정문제 해결을 위해 활용된다(Moslehi & Mahnam, 2011).

본 연구의 문제영역은 전형적인 복수의 모드를 가진 RCPSP이며, 일반적으로 RCPSPMM의 경우 해 영역의 크기가 커지면 최적해를 산출하는데 소요되는 연산 시간 역시 급증하게 되어 NP-Hard Class의 문제가 된다(Blazewicz, 1978). Davis(1973)는 정수 선형 계획법을 적용한 연구에서 이 런 분야의 문제를 직접적으로 다루는 것은 연산 시간의 증가로 현실적이지 않다고 했다. 이러한 이유로 NP-Hard Class의 경우 휴리스틱 기법을 사용하는 것이 현실적이다(안태호, 1998).

### 2.2.3. 정수 선형 계획법(Integer linear Programming)

정수 선형 계획법은 변수를 정수 값으로 제한하여 최적화 문제를 다루는 기법을 말한다. 자원 제약을 고려한 일정문제에서 Nudtasomboon & Randhawa(1997)은 정수 선형 프로그래밍 해법에 관하여 연구하였으며, 이들은 활동이 시작하면 중지되지 않는 경우(Preemption of Activity) 자원의 재생여부에 따른 문제와 시간과 자원의 트레이드오프 문제 들을 다루었으며, 이를 통해 일정문제에 대한 요소들을 통합적으로 고려한 기법을 다루었다.

Patterson & Roth(1976)는 다중 자원 제약하의 프로젝트 일정문제

(MRCPSP, Multiple Resource Constraint Project Scheduling Problem)를 해결하기 위해 암묵적인 열거(Implicit Enumeration)의 (0-1프로그래밍) 알고리즘을 제시하였다. 즉 0-1 공식의 구조를 특별히 활용하여 주어진 다중 자원 문제를 해결하는 데 필요한 계산 시간을 크게 줄였다. 아울러 다중 자원 제약하의 프로젝트 일정계획 문제는 기타 관련 시퀀싱 문제를 포함하고 있음을 고려하여, 제한된 계산 결과와 함께 시퀀싱 문제에 대한 접근 방식의 확장절차를 제시하였다(류미애, 2017).

Herroelen & Demeulemeester(1992)는 자원 제약하의 프로젝트 일정문제가 활동의 시작 우선순위 제약 조건 및 필요한 재생 가능 자원에 대한 지속적인 가용성 제약 조건에 따라, 기간을 최소화하기 위한 프로젝트의 일정 수립과 관련 있다고 설명하였다. 즉 자원 제약하의 프로젝트 일정문제를 우선순위 다이어그램 유형의 선·후행관계 제약조건, 활동 준비 시간 및 마감 시간, 가변적인 자원 가용성의 경우까지 확장하는 ‘일반화된 자원 제약하의 프로젝트 일정계획(GRCPSP, General Resource Constrained Project Scheduling)’을 해결하기 위한 절차를 제시하였다.

또한 자원 제약하의 프로젝트 일정문제의 비 선점 조건을 완화하는 ‘선매적 자원 제약하의 일정문제 (PRCPSP, Preemptive Resource Constrained Project Scheduling Problem)’의 경우, 추가 활동비용 없이 활동을 중단하고 나중에 재개할 수 있는 절차를 제시하였다(류미애, 2017).

#### **2.2.4 자원 제약을 고려한 시간-비용 트레이드오프 문제 (RCTCTP, Resource Constrained Time-Cost Trade-off)**

프로젝트 일정문제에서 시간과 비용간의 트레이드오프와 RCPSP의 통합으로 형성되었으며, Weglarz(1979)와 Talbot(1982)은 이를 수식화하였다. 이 문제의 특징은 활동이 자원을 소모하며, 재생 가능 자원, 재생 불가

능한 자원, 이중 제약 자원을 모두 고려한 점이다. 김갑식(2016)은 프로젝트에서 활동의 품질이 기간과 비용에 큰 영향을 미치는 경우가 흔히 존재하며, 선행 활동이 품질이 후행활동의 재작업, 기간 추가, 비용 추가 등에 영향을 주는 연구를 수행하였다.

RCTCTP와 관련된 주요 연구로는 Talbot(1982), Patterson et al.(1989, 1990), Speranza & Vercellis(1993), Sprecher(1994), Ahn & Erenguc(1995), Nudtasomboon & Randhawa(1997), Sprecher & Drexler(1998), Hartmann(2001), 그리고 Demeulemeester et al.(2000)의 연구가 있다. 그들의 연구에서 분지 한계(Branch-and-Bound) 절차가 개발되었고, De Reyck et al.(1998) 등은 지역 탐색(Local Search)과 타부 탐색(Tabu Search) 휴리스틱 절차를 개발하였다(김갑식, 2016).

#### **2.2.5. 자원 제약을 고려한 일정문제의 휴리스틱 기법**

RCPSP는 제조, 생산계획, 프로젝트 관리 및 기타 다양한 분야에서 광범위한 응용 프로그램을 갖는 일반적인 스케줄링 일반적인 문제이다. RCPSP는 1960년대부터 연구되어 왔으며 NP-Hard 문제이다. 또한 그 해결방법은 주로 휴리스틱이다(Pellerin, Perrier & Berthaut, 2019).

Patterson(1973)은 휴리스틱을 7가지 방식으로 실험하여 프로젝트 지연, 자원의 유휴 기간, 휴리스틱 처리 시간 등에 근거한 휴리스틱 성능을 각각 측정하였다. 실험 결과로 시간과 관련된 휴리스틱 방식은 자원을 비효율적으로 사용하는 대신, 프로젝트의 일정 지연이 최소화되는 일정을 도출하였다. 또한 자원과 관련된 휴리스틱 방식은 프로젝트 소요기간이 지연되는 반면, 자원을 효율적으로 계획한다는 것을 제시하였다. 이때 최소 여유기법(Minimum Slack Method)은 총 프로젝트 지연을 기준으로 할 때, 가장 효율적인 것으로 결론지었다(이현기, 2017).

Davis(1973)는 자원 제약을 고려한 일정 문제를 중 대표적인 것으로 조합 문제(Combinational Problem)을 보고, 자원 제약을 고려한 일정 문제와 조립라인 균등화(ALB, Assembly Line Balancing) 간의 트레이드오프 문제를 연구하였다. 그리고 조립라인 균등화가 반복적 운전 및 동일한 생산물의 다량 생산과 관련된 반면, 프로젝트 일정 계획 문제는 항상 한시적이며, 최초 생산되는 단일 품에 좌우된다고 하였다. 프로젝트 일정계획 문제 중 자원의 형태는 첫 번째로, 모든 활동에 공통적으로 사용되는 유일한 자원의 형태와 두 번째로는 기간마다 한 개 이상의 자원의 종류를 사용하지만, 각 자원 단위(Job Shop 문제로 가정)에 제약 조건과 이용 가능성을 가진 각 활동이 사용가능한 유일한 형태, 세 번째로는 활동 당 한 개 이상의 자원 형태(다중 자원)로 구분하였다(남재덕, 2008).

Patterson et al.(1990)은 자원 제약을 고려한 프로젝트 일정문제를 위한 역추적(backtracking) 알고리즘을 제시하였다.

안태호(1998)는 원 제약을 고려하여 기간 단축이 가능한 복수의 양식을 지닌 단일 프로젝트 일정 문제: 자원 가용량이 시간에 따라 변하는 경우에 대하여 제시하였다.

Lova et al.(2006)는 다중 경로 휴리스틱 기법에 대하여 제시하였다.

Debels & Vanhoucke(2007)는 각 활동의 수행기간이 불확실할 때, 자원이 제한된 경우의 유전자 알고리즘을 활용한 방안을 제시 하였다. 이 경우 비용의 최소화 문제로 네 가지 개선 절차를 포함한 시뮬레이션 방법을 제시하였다(이현기, 2017).

Van Peteghem & Vanhoucke(2010)는 다중 모드 자원 제약하의 프로젝트 일정문제에 대한 유전 알고리즘을 제시하였다.

Kapur et al.(2012)는 소프트웨어 출시 일정 계획 수립을 위한 2차원 다중 릴리스 소프트웨어 안정성 모델링 및 최적화 문제를 해결하기 위한 휴

리스트틱 알고리즘을 제시하였다.

Alsayegh & Hariga(2012)는 활동 분할을 통한 다중 자원 평준화 문제의 비용의 최소화에 대한 해법을 하이브리드 메타 휴리스틱 방식을 제시하였다.

Ponz-Tienda et al.(2013)의 연구에서는 유전자 알고리즘을 활용하여 ‘프로젝트 스케줄링 문제 라이브러리(PSPLIB, Project Scheduling Problem LIBrary)’를 테스트하여 다양한 알고리즘 간의 탐색 시간을 비교하였다.

Naber & Kolisch(2014)는 활동의 자원 사용이 변동적인 FRCPSP 문제를 메타 휴리스틱 기반의 알고리즘으로 연구하였다.

Kellenbrink & Helber(2015)는 공항의 항공기 이·착륙 스케줄링 사례를 통하여 기존의 RCPSP 문제에 변경 가능한 프로젝트 구조를 추가한 유전자 알고리즘을 제시하였다.

Li & Demeulemeester(2016)는 자원 평준화 문제(RLP)를 확률적으로 분석하여 자원 가용량과 활동 소요량의 편차를 최소화하는 유전자 알고리즘을 제시하였다.

Etgar et al.(2017)은 버전계획 및 릴리스 문제에 대한 순 현재가치를 최대화할 수 있는 제품 출시 일정계획을 위한 다양한 최적화 기법과 휴리스틱 기법을 제안하였다.

이현기(2017)는 자원의 렌트를 이용한 자원의 추가투입으로 인한 자원 가용량의 변동을 통한 비용최소화 휴리스틱 알고리즘을 제안하였다.

Geiger(2017)는 MM-RCPSP 문제의 해결을 위한 지역 검색 기법 제시하였으며, 반복 지역 및 가변 이웃 탐색 기법을 기반으로 로컬 탐색을 위한 병렬 방식을 구현하였다.

Li & Lim(2018)은 소매업체를 위한 효율적인 재고 및 물류 시스템에서 매일 각 상점에서 각 재고 보유 단위(SKU)에 대한 수요를 예측하기 위해

Greedy Aggregation-Decomposition 휴리스틱 기법을 제안하였다.

Chand, Singh & Ray(2019)의 연구에서는 RCPSP의 각 의사결정단계에서 우선순위 규칙 및 후속 활동에 대한 정당성을 사용하는 롤아웃-정의(RJ, Rollout-Justification) 절차에 진화된 유전자 알고리즘을 사용하였다.

Chua, Xua & Lib(2019)는 여러 개의 겹치는 모드의 자원이 제한된 프로젝트 스케줄링 문제에서 롤아웃 정책을 기반으로 근사 동적 프로그래밍(ADP, Approximate Dynamic Programming) 알고리즘 개발에 관하여 연구하였다.

## 2.3 재작업 관련 선행연구

### 2.3.1 불확실성하의 스케줄링

일반적인 RCPSP의 경우 대부분 재작업과 같은 불확정적 요소가 거의 존재하지 않는 결정론적인 환경을 고려하고 있다. 그러나 실제의 프로젝트를 수행할 경우에는 다양한 종류의 리스크를 가지고 있으며, 이는 계획된 일정대로 진행하는데 저해되는 요소이다(Li & Ierapetritou, 2008; Chaari et al., 2014; 안정일, 2018;이현주, 2019). 이러한 불확실성을 가진 환경에서의 과학적 의사결정에 대한 연구는 산업 발전에 맞춰 지속적으로 진행되어 왔다. 최근 컴퓨터 기술의 발전과 실험 시스템의 기능 향상 등으로 다양한 관점에서 연구가 진행되고 있으며 급속히 발전하고 있다(Kochenderfer, 2015).

Chakraborty et al.(2017)는 활동기간이 불확실한 RCPSP에 대해 연구하였으며, 활동기간에 대한 확률분포함수를 랜덤 변수로 표시하는 문제를 해결하기 위해 의 불확실성을 결정적 제약조건으로 통합하기위한 6가지 다른 휴리스틱을 개발하였다. SRCPS(Stochastic Resource Constrained Project Scheduling Problem)는 RCPSP의 확률적 확장모델이며, SRCPS에

서 활동 기간은 알려진 확률분포 또는 활동의 수행기간에 따른 임의의 변수로 모델링 된다. 일반적으로 기준일정은 생성되지 않으며, 총 기간(Makespan)의 최소화를 목적으로 한다(Rostami et al., 2018). 이를 해결하기 위해 Stork(2001)은 엄밀한 분기한정 알고리즘을 제안하였으며, Creemers(2015)는 엄밀한 마코프체인 알고리즘을 제안하였다. Chen et al.(2018)은 17가지 자원기반 우선순위 규칙 휴리스틱의 성능을 조사하였으며, Ballestin & Leus(2009)는 GRASP(Greedy Randomized Adaptive Search Procedure)의 적응된 버전을 제안하였으며, Fang et al.(2015)은 진화적 메타휴리스틱인 EDA(Estimation of Distribution Algorithm) 추정을 제안하였다.

Wang & Ning(2018)은 프로젝트 일정을 위한 세 가지 불확실한 선택 제약 프로그래밍 모델을 제시하였으며, Davari & Demeulemeester(2019)는 선택이 제한된 효율적인 분기한정법 알고리즘을 개발하였다. Creemers, Leus & Lambrecht(2010)는 최적의 스케줄링 정책을 결정하기 위해 연속 시간 마코프체인을 기반으로 한 SDP(Stochastic Dynamic Programming)를 사용하였다.

Raghavan, Yoon, & Srihari(2018)는 전자제품 제조서비스(EMS, Electronics Manufacturing Services) 분야에서 확률적인 재작업 및 재처리시간을 가진 일정문제를 해결하기 위해 변형된 유전자 알고리즘을 개발하였다.

### 2.3.2 재작업 관련 선행연구

제품개발 분야에서 재작업은 여러 가지 다른 정의가 있다. Costa & Sobek(2003)은 동일한 범위와 추상적 수준에서 활동(디자인)의 반복이라 하였으며, Wynn & Eckert(2017)는 입력 또는 가정이 변경되었기 때문에

유사한 방식으로 작업을 다시 하는 것으로 정의하였으며, Mitchell & Nault(2007)는 상향과 하향에서 이전에 이루어진 작업으로 인한 설계변경이라 정의하였으며, Arundacahawat et al.(2009)은 불필요한 설계를 되풀이하는 노력으로 정의하였다. Kennedy et al.(2014)는 결함으로 인해 해당 프로젝트의 최종일이 변경될 때 발생하는 작업으로 정의하였으며, Taylor & Ford(2006)는 변화로 인해 다시 해야 할 일로 나타내었다. Smith & Eppinger(1997)는 재작업이 처음에는 불완전한 가정으로 시도되었기 때문에 필수적인 반복으로 정의하였다.

건축 프로젝트 분야에서 Love(2002)는 처음에 부정확하게 시행된 활동 또는 불필요한 일을 다시 하는 것을 재작업으로 정의하였으며, Hwang et al.(2009)에 따르면 부적합, 품질 편차, 품질 실패 및 결함과 같은 용어는 재작업과 동의어로 간주되었다(Abdul-Rahman, 1995; Barber et al., 2000; Burati et al., 1992; Hegazy et al., 2011; Josephson & Hammarlund, 1999; Josephson et al., 2002).



## 제 3 장 문제의 모형

### 3.1 문제의 개요

본 연구는 자원제약을 고려한 프로젝트 일정문제(RCPSP, Resource Contained Project Scheduling Problem)를 다룬다. 기존 RCPSP 모형은 재작업을 고려하지 않아 각 활동의 수행횟수는 1회이다.

본 연구의 모형은 재작업이 발생할 수도 있는 활동들을 포함한다. 재작업의 형태는 재작업의 범위나 횟수 측면에서 다양할 수 있다. 활동 A에 재작업이 발생한다고 하자. 활동 A의 시작부터 끝까지 다시 작업이 이루어져야 하는 경우도 있고, 전체가 아닌 부분만 재작업이 이루어져야 하는 경우도 있다. 본 모형에서는 활동 전체에 대해 재작업이 이루어지고, 재작업의 횟수에 제한이 없으며, 재작업이 발생할 확률이 사전에 알려져 있는 경우를 다룬다.

### 3.2 문제의 가정

본 문제의 가정은 다음과 같다.

- (1) 프로젝트의 활동 : 프로젝트의 활동의 수는  $N$ 개이며, 편의상 시작과 종료에 해당하는 두 개의 가상 활동(Dummy Activity)을 네트워크에 포함한다. 가상활동이므로, 이 두 활동의 실행에 소요되는 기간, 비용, 필요자원은 모두 0이다. 모든 활동이 종료되어야 프로젝트는 종료된다.
- (2) 활동의 선·후 관계 : 활동 간에는 선·후 관계가 확정적으로 존재하며, 선행 활동이 모두 종료한 이후에 후행 활동은 시작할 수 있다. 활동 간의 선·후 관계에 대한 집합을  $H$ 라 하자.  $(i,j) \in H$ 이면

- 활동  $j$ 는 활동  $i$ 가 종료되기 전에는 시작할 수 없음을 의미한다.
- (3) 활동의 비선매성 : 활동은 시작되면 종료 시점까지 연속적으로 수행되어야 하며 종료이전에 중단할 수 없다(Non-preemptive).
  - (4) 모드(Mode) : 활동이 수행되는 방법을 모드라 한다. 모드는 각 활동마다 하나이상 존재한다. 활동이 어떤 모드로 수행되는가에 따라 활동의 완료에 필요한 기간, 비용이 달라질 수 있다. 각 활동의 모드별 수행에 필요한 기간과 비용은 사전에 알려져 있다. 예를 들어, 활동 A를 수행하는 대안이 두 개 존재한다고 하자. 첫 대안(모드)으로 활동 A를 수행하면 기간 20이 소요되고 비용은 50이며, 두 번째 대안(모드)을 택하면 기간은 10이지만 비용은 80이라 하자. 이처럼 두 개 이상의 대안(모드)이 존재하는 경우 활동의 수행기간과 비용은 어떤 대안(모드)을 선택하느냐에 따라 결정된다.
  - (5) 자원 : 프로젝트를 수행하는데 시설, 장비 또는 인력 등의 자원이 필요하다. 각 활동의 수행에 필요한 자원의 종류와 수량은 모드에 따라 다르다. 활동 A가 모드 1로 수행될 때 자원 1은 3, 자원 2는 5만큼 필요하다고 하자. 이 경우 활동 A가 시작되는 시점부터 종료되는 시점까지 자원 1은 3 단위, 자원 2는 5단위가 활동 A에 의해 점유됨을 의미한다. 활동 A가 종료되면, 해당 자원의 점유는 종료된다.
  - (6) 프로젝트의 가용자원 수량 : 프로젝트 전 기간에 걸쳐 사용가능한 자원의 양은 알려져 있다. 어떤 시점에서든 활동들에 의해 요구되는 자원의 양은 가용자원의 양을 초과할 수 없다.
  - (7) 재작업 가능 활동 : 재작업이 가능한 활동의 집합을  $RA$ , 재작업이 발생하지 않는 활동의 집합을  $\overline{RA}$ 라 하자.  $RA$ 에 속하는 활동

들의 재작업 여부는 활동의 수행이 종료되었을 때 검수결과에 따라 결정된다. 재작업이 결정되면, 해당 활동의 이전 작업은 모두 무시된다. 예를 들어, 활동 B가 모드 1(기간 15, 비용 20)로 수행되었고, 검수결과가 재작업이라 하자. 이 경우, 활동 B는 모드 1로 작업을 다시 수행해야 하며, 이때 소요되는 기간 15와 비용 20은 이전 작업에 추가로 발생하는 기간과 비용이다. 활동 B가 원래 작업 1회에 재작업 1회를 수행하여 합격 결과를 받았다면, 총 기간은  $15 \times 2 = 30$ , 비용은  $20 \times 2$ 가 된다.

- (8) 재작업의 연속성 : 재작업 판정시 재작업은 즉시 이루어져야 한다. 예를 들어, 활동 C의 모드 1의 경우 기간이 30이라 하자. 활동 C가 원래 작업 1회에 재작업 2회를 수행하여 합격 결과를 받았다면, 재작업이 즉시 이루어지므로 활동 C의 수행기간은  $30 \times 3 = 90$ 이 된다.
- (9) 재작업 확률 : RA에 속한 활동 별로 재작업 확률은 사전에 알려져 있다.
- (10) 프로젝트 마감기한(Due Date) : 프로젝트는 마감기한이 존재하며, 프로젝트의 종료시점이 마감기한보다 지연되면, 지연시점부터 매 시점마다 지체보상금(Penalty Cost)이 발생한다. 프로젝트의 마감기한과 지체보상금은 사전에 알려져 있다.

### 3.3 문제의 수식화

본 문제의 수식은 다음과 같다.

$$\begin{aligned}
 \text{Minimize} \quad & \sum_{i \in RA} \sum_{m \in M(i)} Acti\_Cost_{i,m} \times x_{i,m} \\
 & + \sum_{i \in RA} \left( \sum_{m \in M(i)} Acti\_Cost_{i,m} \times x_{i,m} \right) \times R_i \\
 & + Pen\_Cost \times \max \{s_{N+1} - Due\_Date, 0\}
 \end{aligned} \tag{1}$$

Subject to

$$x_{i,m} \in \{0, 1\} \quad \forall i, m \tag{2}$$

$$\sum_{m \in M(i)} x_{i,m} = 1 \quad \forall i \tag{3}$$

$$d_i = \begin{cases} \sum_{m \in M(i)} dur_{i,m} \times x_{i,m} \times R_i & \text{if } i \in RA \\ \sum_{m \in M(i)} dur_{i,m} \times x_{i,m} & \text{otherwise} \end{cases} \quad \forall i \tag{4}$$

$$s_i + d_i \leq s_j \quad \forall (i, j) \in H \tag{5}$$

$$s_0 = 0 \tag{6}$$

$$\sum_{i \in S_t} \sum_{m \in M(i)} R_{i,m,k} \times x_{i,m} \leq b_k \quad \forall t, k \tag{7}$$

where,

$i$  프로젝트에서 활동을 나타내는 인덱스이다.  
 $i = 0, 1, 2, 3, \dots, N, N+1$ 이며, 여기서 활동 0과  $N+1$ 은 각각 프로젝트의 시작과 종료를 의미하는 가상의 활동이다.

$RA$  재작업 가능한 활동의 집합이다.

$\overline{RA}$  재작업 불가능한 활동의 집합이다.

$m$	활동의 모드를 나타내는 인덱스이다.
$M(i)$	활동 $i$ 에서 선택 가능한 모드의 집합이다.
$x_{i,m}$	활동 $i$ 가 모드 $m$ 으로 수행되는 여부를 나타내는 0-1 의사결정변수이다. 활동 $i$ 가 모드 $m$ 으로 수행되면 1, 아니면 0의 값을 갖는다.
$Acti\_cost_{i,m}$	활동 $i$ 가 모드 $m$ 으로 수행될 때 활동 $i$ 의 수행비용이다.
$dur_{i,m}$	활동 $i$ 가 모드 $m$ 으로 수행될 때 활동 $i$ 의 수행기간이다.
$R_i$	활동 $i$ 가 프로젝트에서 수행된 횟수를 나타내는 확률변수이다. 수행횟수 $R_i$ 는 자연수에 속한다.
$d_i$	활동 $i$ 의 수행기간이다.
$s_i$	활동 $i$ 의 시작시점을 나타내는 의사결정변수이다.
$H$	활동들의 선·후행 관계를 나타내는 집합이다. ( $i,j$ ) $\in H$ 이면, 활동 $i$ 가 종료되기 전에는 활동 $j$ 가 시작될 수 없다.
$S_t$	시점 $t$ 에 진행 중인 활동의 집합이다. 개념적인 서술이며 시점 $t$ 에 시작된 활동은 포함되나, 종료된 활동은 제외된다.
$R_{i,m,k}$	활동 $i$ 가 모드 $m$ 으로 진행될 때, (기간별) 자원 $k$ 의 소요량이다.
$k$	자원의 종류를 나타내는 인덱스이다.
$b_k$	자원 $k$ 의 매 시점별 가용량이다. 가용량은 프로젝트 전 기간에 걸쳐 동일하다.

$Due\_Date$	전체 프로젝트의 마감기한이다.
$Pen\_cost$	프로젝트의 마감기한을 초과하는 경우 매 시점별로 발생하는 지체보상금이다.

본 연구의 목적함수인 식[1]은 본 문제가 프로젝트 비용의 최소화임을 보여준다. 본 모형에서 프로젝트 비용은 크게 세 부분으로 나누어진다. 첫 번째는 재작업이 없는 활동들의 비용 합계로서, 각 활동이 어떤 모드로 진행되느냐에 따라 결정된다. 두 번째는 재작업이 가능한 활동들의 비용 합계로서, 각 활동이 어떤 모드로 수행되고 또한 수행횟수가 몇 번인가에 따라 결정된다. 마지막은 지체보상금으로서, 프로젝트 종료일이 마감일(Due\_Date)을 초과하는 경우 지연기간에 따라 결정된다.

제약식 [2]에서  $x_{i,m}$ 은 활동  $i$ 가 모드  $m$ 으로 수행되는지를 나타내는 0-1 변수이다. 제약식 [3]은 각 활동별로 단 하나의 모드만이 선택돼야 함을 나타낸다. 제약식[4]는 활동  $i$ 의 총 수행기간  $d_i$ 를 나타낸다.

활동  $i$ 가 재작업이 없는 활동이면( $i \in \overline{RA}$ ),  $d_i$ 는 어떤 모드를 사용하느냐에 따라 결정된다. 즉,  $x_{i,m}$ 에 의해 결정된다. 활동  $i$ 가 재작업이 가능한 활동이면( $i \in RA$ ),  $d_i$ 는 사용 모드와 반복횟수에 따라 결정된다. 즉, 의사결정변수인  $x_{i,m}$ 과 확률변수인  $R_i$ 에 의해 결정된다.

제약식 [5]는 선·후행 관계에 대하여 나타내고 있다. 선행활동  $i$ 의 종료시점( $s_i + d_i$ )이후에 후행 활동  $j$ 가 시작할 수 있음을 의미한다. 여기서 집합  $H$ 는 활동들의 선·후행 관계를 나타내는 집합이며,  $(i,j) \in H$ 이면, 활동  $i$ 가 활동  $j$ 의 선행임을 나타낸다. 제약식 [6]은 활동  $i$ 가 0인 가상활동의 시작시점  $s_0$ 가 0이라는 것을 뜻하며, 이는 프로젝트의 시작 시점을 0으로 설정한다는 의미이다. 식 [7]은 자원관련 제약식이다.  $S_i$ 는 개념적 서술이

며 시점  $t$ 에서 진행 중인 활동들의 집합을 나타낸다.  $\sum_{i \in S_t} \sum_{m \in M(i)} R_{i,m,k} \times x_{i,m}$   
 은 시점  $t$ 에 진행 중인 활동들이 필요로 하는 자원  $k$ 의 요구량의 합이다.  
 즉 매 시점별, 자원 종류별 활동들에 의한 요구량의 합이 자원의 가용량  $b_k$   
 를 초과할 수 없음을 나타낸다.

## 제 4 장 알고리즘

본 연구는 RCPSP(Resource Contained Project Scheduling Problem)의 복수의 모드(RCPSPMM, RCPSP with Multiple Modes)를 다룬다. 일반적인 RCPSPMM은 활동의 수행횟수가 1로 되어 있으나, 본 연구는 일부 활동의 수행횟수가 확률적인 RCPSPMM을 다루고 있다.

일반적으로 RCPSPMM의 경우 전형적인 NP-Hard Class의 문제이다. 이 문제의 경우 해영역의 크기가 커지면 모든 해 영역을 탐색하는 소요시간이 급증하게 된다(Ahn & Erenguc, 1998). 따라서 활동의 수가 작은 경우에는 모든 해 영역을 탐색하는 최적화 기법을 사용할 수 있으나, 활동의 수가 큰 경우에는 모든 해 영역을 탐색하는 데 소요되는 시간이 급격히 증가하게 되어 현실적인 시간 내에 최적해를 도출하는 것이 불가능하게 된다. 그러므로 활동의 수가 큰 경우에는 비교적 짧은 시간 내에 우수한 해를 찾을 수 있는 휴리스틱 기법을 사용하는 것이 현실적이다.

본 연구에서 활동의 수행횟수는 어느 특정 시점에 결정되는 것이 아니다. 재작업이 가능한 활동들은 완료될 때마다 검수를 받는다. 검수 판정이 합격이면 해당 활동은 완전히 완료되며, 불합격이면 해당 활동은 재작업에 들어간다. 검수결과에 따라 아직 완료하지 못한 활동의 집합이 달라지므로, 검수결과가 나오는 시점에서 아직 완료하지 못한 활동들을 대상으로 프로젝트 일정을 다시 작성할 필요가 발생한다. 그러므로 본 연구의 모형은 한 번에 해를 구할 수 있는 문제가 아니라 순차적으로 해를 구하는 다단계 의사결정문제이다. 재작업 여부가 확정되는 시점마다 일정을 재작성하는 방법을 정리하면 아래와 같다.



(1) (최초의 일정)

$t = 0$ 으로 한다. 모든 활동들을 대상으로 최초의 일정을 작성한다.

(2) (다음 번 일정 재작성 시점 탐색)

시점  $t$ 까지 아직 완료하지 않았으며 재작업이 가능한 활동들 중 완료 시점이 가장 빠른 시점을 택한다.

- 이 시점을  $t$ 라 하자. 시점  $t$ 에 종료하는 활동들의 재작업 여부를 판정한다.(판정에 소요되는 시간은 무시할 정도라 가정한다.) 해당 활동의 재작업 여부는 사전에 주어진 재작업 확률에 따라 결정된다.

재작업이 가능한 활동들이 모두 시점  $t$ 까지 완전히 완료(합격)되었으면, 종료한다.

(3) (일정 재작성)

시점  $t$ 까지 완료하지 않은 활동들을 대상으로 일정을 재작성한다.  
(2)를 반복한다.

본 모형을 재작업 가능한 활동의 완료시점별 다단계 의사결정문제로 제시하였으므로, 이제 남은 이슈는 시점별 일정 (재)작성 문제이다. 본 연구에서는 확률변수인 활동의 수행횟수를 상수로 변환하여 일정을 작성하는 방법을 제안한다. 예를 들어, 활동 A는 재작업 확률이 30%이고 두 모드가 있으며, 모드 1의 경우 기간과 비용은 10과 20, 모드 2의 경우 기간과 비용은 5와 30이라 하자. 활동 A에 모드 1을 배정한다고 하자. 한 번에

합격 판정받으면 기간과 비용은 10과 20이 되고, 두 번 만에 합격 판정 받으면 기간과 비용은 20과 40이 되며, 세 번 만에 합격 판정 받으면 기간과 비용은 30과 60이 된다. 일정을 작성하는 시점에서 활동 A의 기간과 비용을 어떻게 추정하느냐에 따라 문제의 해와 목적함수 값이 달라진다. 예를 들어, 재작업 횟수를 (재작업 확률)의 2배로 산정한다면 기간과 비용은 1회 수행 기간과 비용의  $(1+0.3 \times 2)$ 가 되므로 추정 기간과 비용은 16과 32가 되고, 재작업 횟수를 (재작업 확률)의 4배로 산정한다면 기간과 비용은 1회 수행 기간과 비용의  $(1+0.3 \times 4)$ 가 되므로 추정 기간과 비용은 22와 44가 된다. 어떻게 추정치를 설정하느냐에 따라 시점별로 최적의 일정이 달라지고, 목적함수값 또한 달라진다. 문제는 프로젝트 파라미터, 즉 활동들의 모드 구성, 활동 간의 선행관계, 프로젝트 마감일과 지체보상금에 따라 적절한 추정방식이 다르다는 점이다. 그러므로 본 연구에서는 해당 프로젝트에 적합한 활동별 기간/비용의 추정방식을 시뮬레이션을 통해 탐색하는 절차를 제안한다.

#### 4.1 시점 $t$ 에서의 프로젝트 일정 문제

시점  $t$ 에서의 확률변수  $R_i$ 를 포함한 문제와  $R_i$ 를 상수로 변환시킨 문제의 수학적 모형을 소개한다. 두 문제를 구분하기 위해 확률변수  $R_i$ 를 포함한 문제는 확률 모형으로,  $R_i$ 를 상수로 변환시킨 문제는 추정 모형이라 부른다.

##### 4.1.1 시점 $t$ 에서의 확률 모형, $Problem^t$

시점  $t$ 에서의 확률 모형( $Problem^t$ )은 아래와 같이 수식화될 수 있다.

*Problem*<sup>t</sup>

$$\begin{aligned}
\text{Minimize} \quad & \sum_{i \in \overline{RA} \cap (A \cup C)} \sum_{m \in M(i)} \text{Acti\_Cost}_{i,m} \times x_{i,m} \\
& + \sum_{i \in RA \cap (A \cup B \cup C)} \left( \sum_{m \in M(i)} \text{Acti\_Cost}_{i,m} \times x_{i,m} \right) \times R_i^t \\
& + \text{Pen\_Cost} \times \max \{s_{N+1} - \text{Due\_Date}, 0\} \quad [1]
\end{aligned}$$

Subject to

$$x_{i,m} \in \{0,1\} \quad \forall i \in (A \cup B \cup C), m \quad [2]$$

$$x_{i,m} = x_{i,m}^{PREV} \quad \forall i \in (A \cup B), m \quad [3]$$

$$\sum_{m \in M(i)} x_{i,m} = 1 \quad \forall i \in (A \cup B \cup C) \quad [4]$$

$$d_i = \begin{cases} \sum_{m \in M(i)} \text{dur}_{i,m} \times x_{i,m} \times R_i^t & \text{if } i \in RA \cap (A \cup B \cup C) \\ \sum_{m \in M(i)} \text{dur}_{i,m} \times x_{i,m} & \text{if } i \in \overline{RA} \cap (A \cup C) \end{cases} \quad \forall i \quad [5]$$

$$s_i + d_i \leq s_j \quad \forall (i,j) \in H, \forall i, j \in (A \cup B \cup C) \quad [6]$$

$$s_i = s_i^{PREV} \quad \forall i \in A \quad [7]$$

$$s_i = t \quad \forall i \in B \quad [8]$$

$$s_i \geq t \quad \forall i \in C \quad [9]$$

$$\sum_{i \in S_\tau} \sum_{m \in M(i)} R_{i,m,k} \times x_{i,m} \leq b_k \quad \forall \tau \geq t, k \quad [10]$$

where,

$t$                       일정이 작성되는 기준 시점이다. 최초의 시점은 0  
                              이며, 재작업이 가능한 활동의 수행이 완료될 때  
                              마다  $t$ 값은 업데이트된다.

$PREV$                        $t > 0$ 인 경우, 현재의  $t$  직전  $t$ 이다.

$x_{i,m}^{PREV}$	$t > 0$ 인 경우에 한한다. $Problem^{PREV}$ 에서 $x_{i,m}$ 의 값이다.
$s_i = s_i^{PREV}$	$t > 0$ 인 경우에 한한다. $Problem^{PREV}$ 에서 $s_i$ 의 값이다.
$A$	최초의 일정, $Problem^0$ 에서는 {}이다. ( $t > 0$ 인 경우) $Problem^{PREV}$ 에서 시점 $t$ 이전에 시작되었으나 $t$ 까지 종료되지 않은 활동들의 집합이다. $A = \{i \mid s_i^{PREV} < t, s_i^{PREV} + d_i^{PREV} > t\}$ 여기에 속한 활동들의 시작시기와 모드는 $Problem^{PREV}$ 에서 $Problem^t$ 로 승계된다.
$B$	시점 $t$ 에 재작업 판정받은 활동의 집합이다. $B = \{i \in RA \mid s_i^{PREV} + d_i^{PREV} = t, \text{재작업 판정}\}$ $t = 0$ 이면 {}이다. $t > 0$ 인 경우, $Problem^t$ 에서 시작시점은 $t$ 가 되며, 모드는 $Problem^{PREV}$ 에서 $Problem^t$ 로 승계된다.
$C$	시점 $t$ 이후에 시작하는 활동의 집합이다. $t = 0$ 이면 모든 활동이 이 집합에 속한다. $t > 0$ 인 경우, $C = \{i \mid s_i^{PREV} \geq t\}$
$R_i^t$	활동 $i$ 가 현 시점 $t$ 에 수행될 횟수를 나타내는 확률변수이다. 수행횟수 $R_i^t$ 는 자연수에 속한다.

식 [1]은 시점  $t$ 에서의 확률모형( $Problem^t$ )이 프로젝트의 비용 최소화 문제임을 보여준다. 프로젝트 비용은 세부분으로 나누어진다. 첫 번째는 재

작업이 없는 활동들( $i \in \overline{RA}$ ) 중 시점  $t$ 에서 완료되지 않은 활동( $A \cup C$ )의 비용 합계이며, 두 번째는 재작업이 가능한 활동( $i \in RA$ ) 중 시점  $t$ 에서 재작업 되거나 시점  $t$ 에서 완료되지 않은 활동들( $A \cup B \cup C$ )의 비용 합계이며, 마지막은 지체보상금이다.

제약식 [3]은 집합 A와 B에 속한 모든 활동의 모드  $m$ 에 대한 결정이  $Problem^{PREV}$ 에서 승계되었음을 뜻한다. 제약식 [7]은 집합 A의 활동들의 시작시점이  $Problem^{PREV}$ 에서 승계되었음을 뜻한다. 제약식 [8]은 집합 B의 활동들의 시작시점이  $t$ 임을 뜻한다. 제약식 [9]는 집합 C의 활동들의 시작시점이  $t$ 이후임을 뜻한다.  $S_\tau$ 는 개념적인 서술이며 시점  $\tau (\tau \geq t)$ 에서 진행 중인 활동들의 집합을 나타낸다.

#### 4.1.2 시점 $t$ 에서의 추정 모형, $Problem\_Est^t$

활동  $j$ 의 시작시점( $s_j$ )를 결정하기 위해서는 활동  $j$ 의 선행활동들의 종료시점( $s_i + d_i$ )이 확정되어야 한다. 그런데  $Problem^t$ 에서 재작업이 가능한 활동의 경우,  $d_i$ 는 확률변수  $R_i$ 에 따라 그 값이 결정된다. 추정 모형은  $R_i$ 를 제거하고 그 위치에  $R_i$ 를 추정한 상수를 사용한다. 이로 인해 확률적 모형이 확정적 모형으로 변환되어, 각 활동의 시작시점과 모드 결정이 가능하게 된다.

$Problem\_Est^t$

$$\begin{aligned}
 &Minimize \sum_{i \in \overline{RA} \cap (A \cup C)} \sum_{m \in M(i)} Acti\_Cost_{i,m} \times x_{i,m} \\
 &+ \sum_{i \in RA \cap (A \cup B \cup C)} \left( \sum_{m \in M(i)} Acti\_Cost_{i,m} \times x_{i,m} \right) \times Est\_R_i^t \\
 &+ Pen\_Cost \times \max \{s_{N+1} - Due\_Date, 0\}
 \end{aligned} \tag{1}$$

식 [1]은 시점  $t$ 에서의 추정 모형( $Problem\_Est^t$ )가 시점  $t$ 에서의 확률모형( $Problem^t$ )에서  $R_i^t$ 를 삭제하고 그 위치에 상수  $Est\_R_i^t$ 를 사용한 것임을 나타낸다. 즉  $Problem\_Est^t$ 는 확률변수 대신 상수를 사용함으로써 확률변수가 없는 확정적 모형으로 변환되었다.

#### 4.1.3 $R_i$ 를 추정하는 방법

확률변수  $R_i$ 을 상수로 추정하는 방법 중 하나는  $R_i$ 의 기댓값(Expected Value)이다.

활동  $i$ 의 재작업 확률을  $p_i (0 \leq p_i < 1)$ 라 하자. 활동  $i$ 의 재작업 횟수는 수렴하며, 멱급수의 합(Sum of Power Series)의 공식을 통하여 아래와 같이 상수로 정리된다.

$$\begin{aligned}
 & \text{활동 } i \text{의 재작업 횟수의 기댓값} \\
 &= (\text{재작업 1회}) \cdot p_i^1 \cdot (1-p_i) + (\text{재작업 2회}) \cdot p_i^2 \cdot (1-p_i) + \dots \\
 & \quad + (\text{재작업 } n \text{회}) \cdot p_i^n \cdot (1-p_i) + \dots + (\text{재작업 } \infty \text{회}) \cdot p_i^\infty \cdot (1-p_i) \\
 &= \sum_{n=1}^{\infty} n \cdot p_i^n \cdot (1-p_i) = (1-p_i) \sum_{n=1}^{\infty} n p_i^n \\
 &= (1-p_i) \times \frac{p_i}{(1-p_i)^2} \quad (\text{Since Sum of Power Series}) \\
 &= \frac{p_i}{(1-p_i)}
 \end{aligned}$$

재작업은 일단 활동 수행이 완료된 이후에 발생하므로, 활동의 수행 횟

수  $R_i$ 의 기댓값은 아래와 같다.

$$E(R_i) = 1 + \text{재작업의 기댓값} = 1 + \frac{p_i}{1-p_i}$$

$R_i$ 의 추정치로  $R_i$ 의 기댓값을 사용하는 것이 모든 프로젝트 상황에서 최선이라는 보장은 없다. 그러므로 본 연구에서는 재작업의 기댓값에 일정 배수(Multiplier)를 곱하여 다양한 값의  $R_i$ 의 추정치를 생성하고자 한다.

$$Est\_R_i^t = 1 + \frac{p_i}{1-p_i} \times M_i$$

$M_i = 1$ 이면  $Est\_R_i$ 는  $R_i$ 의 기댓값이 되고,  $M_i = 0.5$ 이면  $Est\_R_i$ 는 재작업 횟수를 기댓값 대비 50%만 반영한 추정치가 되며,  $M_i = 2$ 이면  $Est\_R_i$ 는 재작업 횟수를 기댓값 대비 200% 반영한 추정치가 된다.

## 4.2 해법

### 4.2.1 $Problem\_Est^t$ 의 휴리스틱 기법 약술

$Problem\_Est^t$ 는 전형적인 RCPSPPM이다. 본 논문에서 제안하는 휴리스틱 기법은 이웃탐색 기법을 기반으로 한 다수 경로(Multi-Pass) 알고리즘으로 아래와 같이 표현된다.

#### Step 1. [해의 생성]

실행 가능한 해를 생성한다. Goto Step 2.

#### Step 2. [해의 부분 변경을 통한 개선]

다수의 이웃 탐색 기법을 반복적으로 사용하여 해를 개선한다.  
더 이상 해가 개선되지 않으면 Step 3로 간다.

### Step 3. [진행 여부 판단]

미리 설정한 연산 시간을 초과하였거나, 미리 설정한 새로운  
실행 가능한 해 생성 제한 횟수를 초과하였으면 종료한다.  
아니면 새로운 실행 가능한 해를 생성하기 위해 Step 1로 간  
다.

본 휴리스틱 기법의 알고리즘은 Step 1에서 실행 가능한 해를 생성하는  
것으로 시작된다. 해의 생성이란 각 활동 별 시작시기와 모드를 모두 결  
정한다는 것이며, 실행 가능한 해란 배정된 시작시기와 모드가 문제의 제  
약 조건을 모두 충족한다는 의미이다.

이렇게 생성된 해는 Step 2에서 이웃 탐색 기법을 사용하여 개선을 시도  
한다. Step 1에서 시작시기와 모드가 모두 결정된 것에 비해, Step 2에서  
적용하는 이웃 탐색 기법은 해의 전면적인 생성이 아니라 해의 일정 부분  
만을 특정하여 그 부분만을 변경해 본다. 변경할 때 목적함수값이 개선되  
면 그 부분이 변경된 새로운 해를 생성하고, 개선되지 않으면 해의 변경을  
취소한다. 해의 특정 부분을 지속적으로 변경함으로써, 해의 개선을 계속  
시도해 본다. 이러한 이웃 탐색 기법을 통한 해의 개선이 더 이상 가능하  
지 않게 되면, Step 1로 가서 해의 부분적인 변경이 아닌, 전체 활동의 시  
작시기와 모드가 새로이 결정된 새로운 해를 생성하고 다시 Step 2로 가  
서 이웃 탐색을 통한 해의 부분 개선을 시도한다.

이런 종류의 알고리즘은 연산 시간이나 특정 Step을 방문하는 횟수를  
사용하여 종료시킬 수 있다. 본 연구에서는 Step 3를 방문한 횟수에 제한  
두는 방법을 사용하였다. 즉, Step 3를 방문할 때 누적 방문 횟수가 미리



설정한 값을 초과하면, 지금까지 발견된 해 중 가장 좋은 해를 최선의 해 (Best Solution)로 삼고, 알고리즘을 종료한다.

#### 4.2.1.1 실행 가능한 해의 생성 - Step 1.

$Problem\_Est^t$ 에서 해(Solution)이란 활동별 시작시기  $s_i$ 와 실행 모드  $m$ 으로 구성되어 있다. 수학적으로는 0-1 변수인  $x_{i,m}$ 들 중 1의 값을 갖는  $m$ 이 실행 모드가 된다.

$Problem\_Est^t$ 에서 활동  $i$ 가  $A$ (시점  $t$ 에 이미 진행 중인 활동들) 또는  $B$ (재작업 판정받아 시점  $t$ 에 기존의  $m$ 으로 재작업 시작되어야 하는 활동들)에 속한다면, 활동  $i$ 의  $s_i$ 와  $m$ 은 이미 고정되어 있다. 아직  $s$ 와  $m$ 이 미배정된 활동들은  $C$ 에 속한 활동들이므로, Step 1에서는  $C$ 에 속한 활동들의  $s$ 와  $m$ 을 순차적으로 결정하여 일정을 완성한다. 그런데  $s$ 와  $m$ 은  $Problem\_Est^t$ 에 존재하는 각종 제약조건들을 준수하면 결정되므로, Step 1에서 완성되는 일정 또는 해는 실행 가능한 해(Feasible Solution)이다.

##### (1) 활동별 모드 결정

$C$ 에 속한 활동들의 모드는 난수를 사용하여 임의로 선택한다. 활동별 모드가 결정되면, 그 활동의 기간도 결정된다.

##### (2) 활동별 시작시기 결정

$A$ 와  $B$ 에 속한 활동들은 시작시기가 이미 결정되어 있으므로, 여기서는  $C$ 에 속한 활동들의 시작시기를 결정한다.

###### (2-1) 초기화

$C$ 에 속한 모든 활동들의  $s_i$ 가 미정인 상태로 시작한다.

###### (2-2) $s_i$ 를 배정할 활동 선택

선행활동이 없거나 모든 선행활동의  $s_i$ 가 확정된 활동들 중 아직  $s_i$ 가 배정되지 않은 활동을 선택한다.

- 미배정된 활동이 없으면, 모든 활동들의  $s_i$ 가 배정되었으므로 종료한다.

- 이 조건을 충족하는 활동이 여러 개 있으면, 그 중 하나를 임의로 선택한다. 선택한 활동을  $i$ 라 하자.

#### (2-3) $s_i$ 배정

다음 조건을 충족하는  $\tau$  중 가장 작은 값을 그 활동의 시작 시기  $s_i$ 로 한다.

(a)  $\tau$ 는 활동  $i$ 의 선행 활동들의 종료시기보다 빠를 수 없다.

(b) 활동  $i$ 가 모드  $m$ 으로 시점  $\tau$ 에 시작하여도 시점  $\tau$ 에 자원 제약 조건을 충족한다. ( $Problem^t$  제약식 [10])

### (3) 목적함수값 산정

(1)과 (2)에서  $s_i$ 와  $m$ (선택된 모드의  $x_{i,m} = 1$ , 선택되지 않은 모드의  $x_{i,m} = 0$ )이 결정되었으므로, 목적함수에  $s_i$ 와  $x_{i,m}$ 을 대입하여 목적함수값을 구한다.

#### 4.2.1.2 이웃 탐색기법을 통한 해의 개선 - Step 2.

$Problem\_Est^t$ 에서는 이웃 탐색 기법을 통해 해의 일정 부분만을 변경하여 그 해와 목적함수 값을 개선한다.

Step 2에서는 Step 1에서 생성된 실행 가능한 해의 일정 부분만을 특정하여 변경한다. 부분해의 변경 후 목적함수 값이 개선되었다면 새로운 해로 채택하고 개선되지 않았다면 부분해의 변경을 취소한다.

이 Step에서 다루지는 기법들은 (1) 모드를 변경하여 해를 개선시키는 기법과 (2) 활동의 배정을 변경하여 해를 개선시키는 기법으로 분류한다.

**(1) 모드를 변경하여 해를 개선하는 기법**

$C$ 에 속한 활동들 중 하나를 임의로 선택하여, 모드 배정을 달리하여 본다. 선택한 활동을  $i$ 라 하고, 기존의 시작시기를  $\bar{s}_i$ 라 하자. 일정에서 활동  $i$ 의 모드가 변경되면,  $d_i$ 이 달라질 수 있고 또한 기간별 필요한 자원의 양이 달라질 수도 있다. 이로 인해 활동  $i$ 가 기존의  $\bar{s}_i$ 에 시작할 수 없게 될 수도 있으며, 활동  $i$ 의 시작시점과 기간 그리고 자원 요구량이 달라짐에 따라 시점  $\bar{s}_i$  이후에 시작하도록 배정된 활동의 시작시점이 달라질 수도 있다.

(1-1)  $C$ 에 속한 활동들 중 하나를 임의로 선택한다. 그 활동을  $i$ 라 하자. 활동  $i$ 의 모드를 변경해 본다. 현재의 시작시점을  $\bar{s}_i$ 라 하고, 변경된 모드를  $\bar{m}$ 이라 하자.

(1-2)  $C$ 에 속한 활동들 중 시작시점이  $\bar{s}_i$  이후인 활동들의 시작시점 배정을 취소한다.

(1-3) 활동  $i$ 의 시작시기를 배정한다.

(1-4) 시작시기가 미배정인 활동들의 시작시기를 배정한다. 시작시기 배정은 Step 1의 [활동별 시작시기 결정] 방식에 준한다.

이러한 변경으로 해의 목적함수값이 개선되면 해를 변경하고, 개선되지 않으면 기존의 해로 복귀한다.

(2) 활동의 시작시기 배정에서 우선순위를 변경하여 해를 개선하는 기법

Step 1의 [활동별 시작시기 결정]에서 시작시기가 배정될 수 있는 활동이 복수인 경우가 있다. 프로젝트 선행관계만 고려하면 그러한 활동들은 동일한 시점에 시작될 수 있으나, 자원 제약으로 인해 동 시점에 시작하는 것이 불가능한 경우도 있다. 이 경우 어느 활동을 먼저 선택하여 시작시기를 배정하느냐에 따라 일정이 달라진다.

이 기법은 활동들의 모드는 그대로 유지하고 활동들의 시작시기를 변경하는 기법으로, Step 1의 [활동별 시작시기 결정] 부분과 동일하다. Step 1의 [활동별 시작시기 결정]에  $s_i$ 가 배정될 수 있는 활동이 복수인 경우, 배정할 활동을 임의로 선택하게 되어 있다. 어떤 활동이 선택되느냐에 따라 일정이 달라질 수 있다.

변경된 일정의 목적함수값이 개선되면 해를 변경하고, 그렇지 않으면 기존의 해를 유지한다.

#### 4.2.1.3 개선 기법 진행 여부 판단 - Step 3.

Step 3는 개선 기법의 진행 여부를 결정하는 단계이다. 본 연구에서 제안하는 휴리스틱 기법은 Multi-Pass 알고리즘이므로 Pass 방문 횟수에 제한을 둘 수도 있고 CPU Run-time에 제한을 둘 수도 있다.

#### 4.2.2 Problem의 시뮬레이션

Problem은 확률적 의사결정 모형으로서, 그 목적함수값은 의사결정( $s_i$ 와  $m$ )과 확률변수  $R_i$ 들이 어떤 값들을 갖게 되느냐에 따라 달라진다. 그런데  $R_i$ 들의 값은 어떤 시점에서 동시에 값이 결정되는 것이 아니라, 재작업

검수 결과가 합격이 될 때 최종적으로 확정된다. 즉, 프로젝트 진행에 따라 단계별로 값이 결정된다는 특징이 있다.

본 연구에서는 *Problem*의 목적함수값을 구하기 위해 다음과 같은 절차를 제안한다.

- (1) 확정되지 않은  $R_i$ 는 모두 추정치로 변환하여 일정을 작성한다.  
(시작시기와 모드 결정)
- (2) 가상으로 프로젝트 시점을 증가시켜 가장 빠른 재작업 검수 시점을 찾는다. 난수를 사용하여 재작업 검수 결과를 발생시키고, 잔여 일정을 재작성한다.
- (3) 재작업 가능한 활동들 중 아직 합격처리 되지 못한 활동이 있으면 (2)를 반복한다. 재작업 가능한 활동들이 모두 합격 처리되면 모든  $R_i$ 값들이 확정되며, 이때 일정은 최종 일정이 되며 *Problem*의 목적함수값이 결정된다.

그런데 (3)에서 결정되는 목적함수값은 검수 시 어떤 난수가 발생하였느냐에 따라 달라질 수 있다. 본 연구는 *Problem*의 해를 찾는 과정을 충분히 많이 반복하여 *Problem*의 목적함수값에 분포를 구하고자 한다.

## 제 5 장 모의실험 결과

본 모의실험의 목적은 제시된 모형이 실제로 작동함을 증명하고, 본 알고리즘의 유효성을 입증하기 위함이다.

본 논문에서 제안한 알고리즘은 확률적 모형에서 확률적인 부분을 확정된 값으로 추정하여 확정적 모형으로 변환시키고, 변환된 모형을 다단계로 풀어나가는 알고리즘이다. 그러므로 문제의 해는 (1) 불확실성(알고리즘 측면에서는 발생 난수)과 (2) 추정방식에 따라 결정된다.

따라서 모의실험에서는 (1)의 해결방안으로, 불확실성을 분포적으로 파악하기 위해 충분히 많은 수의 모의실험을 수행하고자 한다. (2)에 대한 해결방안으로 다양한 추정방식을 도입해 보고자 한다. (1)과 (2)의 결합을 통해, 분포적으로 우월한 추정방식을 파악하는데 본 모의실험의 의의가 있다.

본 모의실험에서 분포적으로 우월한 추정방식의 기준은 하나의 추정방식( $M$ 값)으로 실험된 결과(프로젝트 총 비용)들의 분포에 대한 평균이 다른 추정방식으로 실험된 결과들의 평균보다 더 작은 경우를 의미한다. 현실에서는 주어진 비즈니스 환경에 따라 분포에 대한 전체의 평균보다 더 좋은 기준이 있을 수도 있으나 본 모의실험에서는 분포에 대한 평균을 사용한다.

본 모의실험에서 알고리즘은 Microsoft Office Professional Plus 2016 Excel VBA(Visual Basic for Applications) 언어로 코드를 작성하였으며, 연산의 수행은 Intel Core i5-3570K CPU, 3.4 GHz 환경에서 진행하였다.

본 모의실험의 결과는 세 부분으로 나누어져 있다. 첫 번째는 모의실험에 대한 조건 설정이며, 두 번째는 *Problem*의 시뮬레이션의 결과로  $M$ 값(일정 작성 시 수행횟수의 기댓값을 계산하기 위한 계수)의 변화에 따른

총 비용과 활동비용의 합, 그리고 지체보상금의 시뮬레이션 결과의 평균값을 보여준다. 세 번째는 전체 시뮬레이션 결과에 대한 요약이다.

## 5.1 모의실험 조건 설정

### 5.1.1 데이터 생성 기본구성

실험 조건은 동일한 환경과 조건에서 생성된 데이터를 실험하였으며 *Problem*의 시뮬레이션을 실험하기 위한 데이터셋의 기본구성은 [표 5-1]과 같다.

[표 5-1] 모의실험 데이터셋 기본구성

기호	구분	Set 1	Set 2	Set 3
N	활동의 수	20	20	20
M	모드의 수	5	5	5
K	자원 종류의 수	2	2	2
Due Date	수행완료일	350	350	350
Penalty	단위 당 패널티	50	50	50
Num_of_Sim	$M$ 값별 시뮬레이션 횟수	1000	1000	1000

본 모의실험에서는 [표 5-1]과 같이 세 가지 프로젝트를 진행한다. 프로젝트들은 활동의 수(N)가 각각 20이고 각 활동에 대한 선택 가능한 모드의 수(M)는 5로 설정하였으며, 각 모드 내에서 선택 가능한 자원 종류(K)는 2개로 설정한다. 그 외 모드별 수행기간, 수행비용, 필요자원 등의 계수는 프로젝트의 데이터를 생성할 때 난수 발생을 통하여 임의로 생성하였다.

## 5.2 모의실험 결과

*Problem*의 시뮬레이션에서 재작업 기댓값에 대한 배수( $M$ )별 진행은 시점  $t$ 에서의 일정작성과 프로젝트 종료까지  $t$ 의 변화에 따른 실행으로 나누어진다. 먼저  $t=0$ 일 때 초기일정을 작성한 후 가장 가까운 재작업 가능활동의 검수시점  $t$ 까지 일정을 진행한다. 그리고 재작업의 검수 후 일정의 재작성과 새로운 시점  $t$ 까지의 진행이 프로젝트 종료 시까지 반복되며, 프로젝트 종료 시  $M$ 별 시뮬레이션 1회에 대한 목적함수 값이 발생된다.

데이터셋 별  $M$ 의 값은 0부터 4까지 0.4의 간격으로 총 11개의 값을 가지며, 각 해당되는  $M$ 의 값별 시뮬레이션 횟수는 각 1,000회이다. 하나의  $M$ 의 값이 완료되면 총 비용과 활동비용의 합, 지체보상금의 평균을 구한다. *Problem*에서 총 비용은 모든 활동의 수행비용의 합에 지체보상금을 더한 것과 같다.

데이터셋 당 시뮬레이션의 횟수는  $M$ 의 개수 11개에 시뮬레이션 1,000회를 곱하여 11,000회가 진행되며, 각 데이터셋에 대해 그 결과는 다음과 같은 순서로 나타낸다.

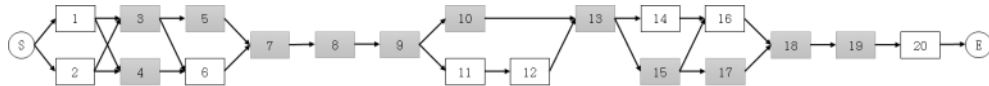
- (1) 프로젝트 네트워크 : 해당 데이터셋의 네트워크이다.
- (2) 재작업 가능 활동과 재작업 확률 : 해당 네트워크의 재작업 가능 활동과 재작업 확률로 계수이다.
- (3) 활동의 모드별 수행기간과 수행비용 : 해당 활동의 모드가 선택되는 경우의 수행기간과 수행비용으로 계수이다.
- (4) 활동의 모드별 필요자원 : 해당 활동의 모드가 선택되는 경우의 필요자원이다.
- (5) 시뮬레이션 결과 : *Problem* 시뮬레이션의 결과로 각  $M$ 값별 총 비용, 활동비용의 합, 지체보상금을 평균과 그래프로 나타낸다.
- (6) 시뮬레이션 결과 요약 : (5)에 대한 요약이다.



### 5.2.1 데이터셋 1의 모의실험 결과

#### (1) 프로젝트 네트워크 - 데이터셋 1

데이터셋 1의 프로젝트 네트워크는 활동의 수를 20으로 한 후, 각 활동 간의 선행관계를 0.7의 확률로 임의로 생성하였다. 임의로 생성된 선행관계에서 정상적인 선행관계의 작성에 필요한 연결은 추가하였으며 필요 없는 연결은 모두 삭제하였으며, 그 결과는 [그림 5-1]과 같다.



[그림 5-1] 데이터셋 1의 프로젝트 네트워크

#### (2) 재작업 가능 활동과 재작업 확률 - 데이터셋 1

데이터셋 1의 재작업 가능 활동의 수는 (1)의 프로젝트 네트워크에서 60%를 임의로 선택 하였으며, 재작업 확률은 10%에서 30%사이의 값을 임의로 생성하였다. 재작업 가능 활동은 [그림 5-1]의 네트워크에서 짙은 색 배경으로 표시되었으며, 그 결과는 [표 5-2]와 같다.

[표 5-2] 데이터셋 1의 재작업 가능 활동과 재작업 확률

재작업 가능 활동	재작업 확률	재작업 가능 활동	재작업 확률
3	0.22	10	0.30
4	0.12	13	0.19
5	0.14	15	0.10
7	0.24	17	0.19
8	0.20	18	0.16
9	0.14	18	0.16

(3) 활동의 모드별 수행기간과 수행비용 - 데이터셋 1

데이터셋 1의 활동의 모드별 수행기간은 10에서 82사이의 값을 임의로 생성하였으며, 각 수행별 비용은 50에서 576사이의 값을 임의로 생성하였다. 그 결과는 [표 5-3]과 같다.

[표 5-3] 데이터셋 1의 기간 및 비용

활동	수행기간					수행비용				
	모드1	모드2	모드3	모드4	모드5	모드1	모드2	모드3	모드4	모드5
1	13	14	17	26	50	405	203	127	101	83
2	15	17	23	37	69	339	176	109	81	72
3	13	14	17	27	49	326	170	113	91	85
4	12	14	19	27	49	261	132	83	65	53
5	14	17	20	31	56	386	202	132	106	88
6	14	15	18	28	52	410	222	138	101	85
7	13	14	19	27	53	349	177	111	84	75
8	14	15	19	29	55	214	118	77	61	58
9	15	16	20	30	56	232	123	84	66	61
10	12	13	15	22	43	288	148	97	71	66
11	12	13	15	24	44	375	203	134	102	89
12	12	14	17	27	50	240	128	81	59	53
13	13	14	16	25	46	296	158	105	81	76
14	11	13	17	26	48	373	196	122	97	88
15	12	13	16	25	47	396	214	145	106	90
16	13	14	18	27	49	298	156	100	80	66
17	14	16	19	28	53	425	231	141	113	99
18	13	15	19	29	57	323	178	119	97	90
19	12	14	18	29	55	419	212	130	103	84
20	13	15	20	29	57	340	187	127	106	95

(4) 활동의 모드별 필요자원 - 데이터셋 1

데이터셋 1의 활동의 모드별 필요자원의 종류는 2가지이며 그 요구량은 5에서 10사이의 값을 임의로 생성하였다. 그 결과는 [표 5-4]와 같다.

[표 5-4] 데이터셋 1의 필요자원

활동	모드1		모드2		모드3		모드4		모드5	
	자원1	자원2	자원1	자원2	자원1	자원2	자원1	자원2	자원1	자원2
1	9	9	6	6	5	6	7	6	10	9
2	8	9	6	6	5	9	9	9	7	7
3	8	9	8	6	6	7	8	7	8	7
4	8	5	8	6	10	10	10	9	7	9
5	10	6	6	8	5	7	8	6	6	6
6	7	8	7	6	9	7	7	7	9	7
7	9	8	7	10	8	8	6	8	9	7
8	10	6	8	9	6	7	5	6	10	9
9	8	8	6	8	9	9	5	8	7	5
10	7	8	7	7	10	10	8	8	8	5
11	6	6	5	6	7	8	8	9	5	7
12	5	7	8	8	8	7	10	9	8	7
13	9	6	5	6	9	10	7	8	9	5
14	9	7	6	8	10	10	8	5	8	10
15	5	10	5	6	8	9	5	5	8	7
16	7	8	9	7	7	5	7	8	9	8
17	8	9	7	9	9	8	9	6	9	5
18	6	10	9	6	6	6	10	6	6	9
19	8	9	9	7	9	9	9	6	7	8
20	7	6	8	8	6	5	8	9	8	7

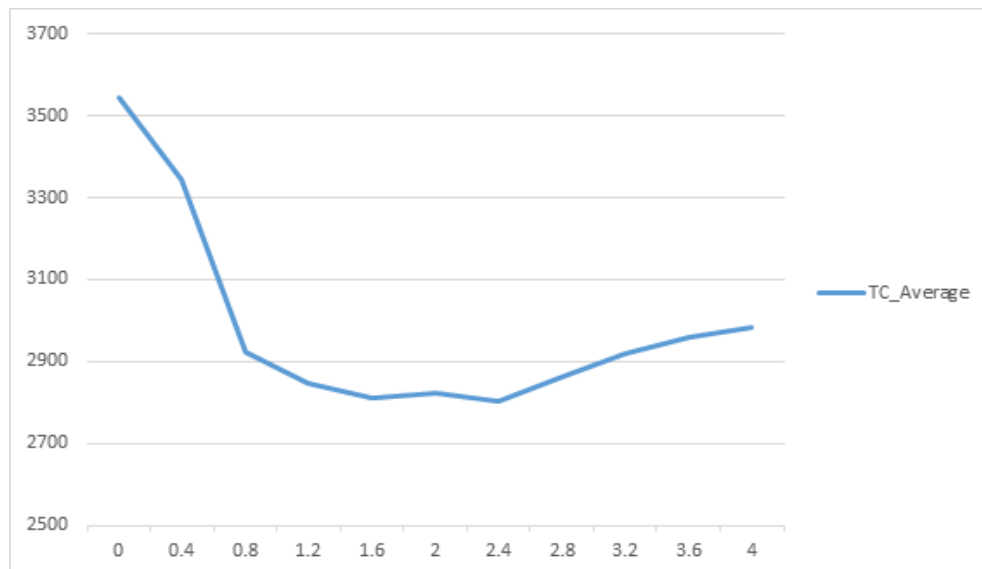
(5) 시뮬레이션 결과 - 데이터셋 1

데이터셋 1에 대한 시뮬레이션은 하나의  $M$ 값에 대해 1,000회를 실행하였으며, [표 5-5]는 각각의  $M$ 값에 대한 시뮬레이션의 평균으로 작성된다.

[표 5-5] 데이터셋 1의 모의실험 결과

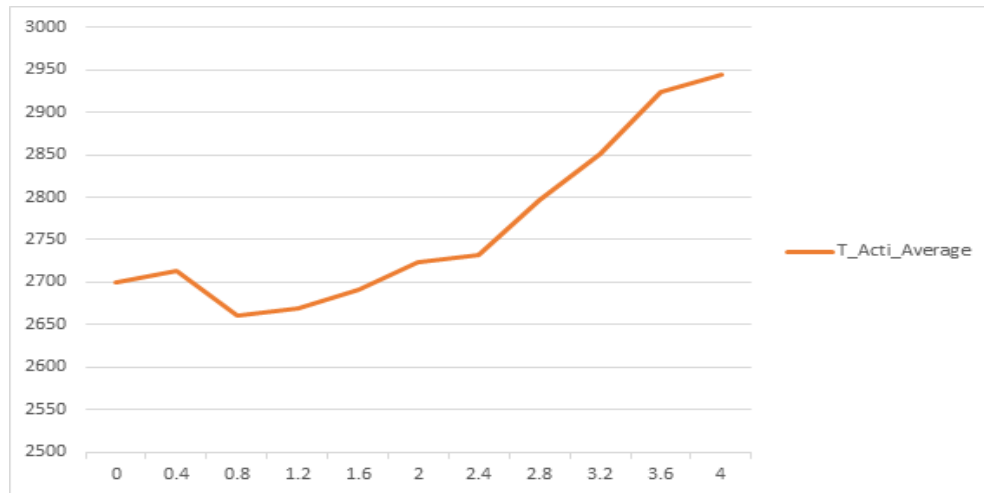
$M$	0	0.4	0.8	1.2	1.6	2	2.4	2.8	3.2	3.6	4
총 비용	3544	3343	2923	2846	2809	2824	2803	2861	2921	2960	2981
활동비용의 합	2700	2713	2661	2669	2691	2723	2732	2797	2852	2924	2944
지체보상금	844	630	261	177	118	101	71	65	69	36	37

[그림 5-2]는 [표 5-5]에서  $M$ 의 변화에 따른 총 비용의 변화를 나타내는 그래프이다.



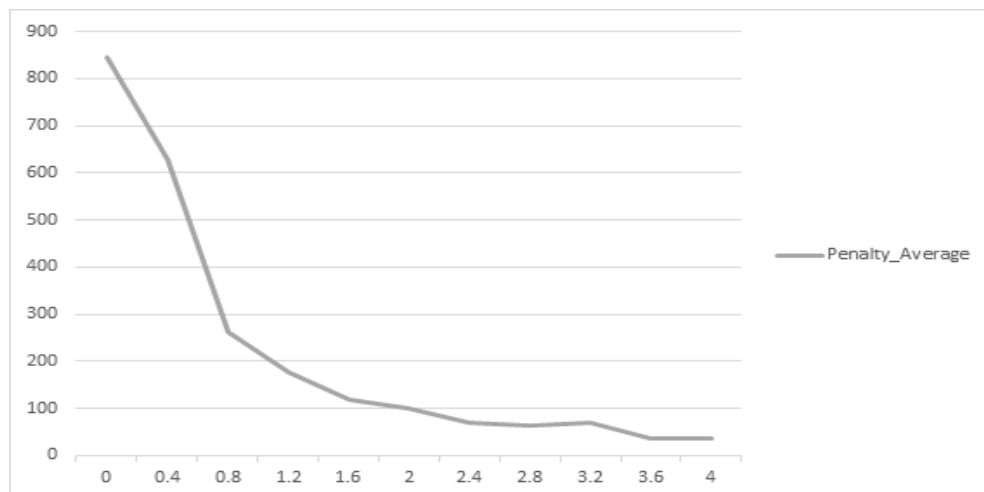
[그림 5-2] 데이터셋 1의 총 비용 그래프

[그림 5-3]는 [표 5-5]에서  $M$ 의 변화에 따른 활동비용의 합의 변화를 나타내는 그래프이다.



[그림 5-3] 데이터셋 1의 활동비용 합 그래프

[그림 5-4]는 [표 5-5]에서  $M$ 의 변화에 따른 지체보상금의 변화를 나타내는 그래프이다.



[그림 5-4] 데이터셋 1의 지체보상금 그래프

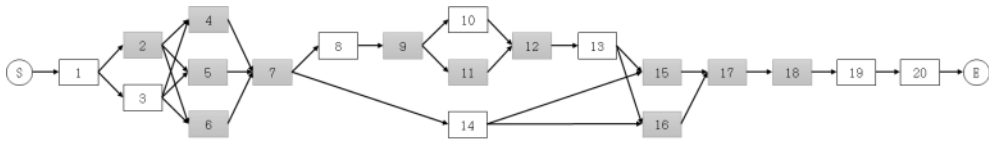
(6) 시뮬레이션 결과 요약 - 데이터셋 1

- 시뮬레이션의 평균에서 프로젝트의 총 비용은  $M=2.4$ 일 때 2,803으로 최솟값을 가진다. 또한  $M$ 이 0부터 1.2까지 총 비용은 점차 낮아지며, 1.6에서 2.4까지 비슷한 값을 유지하다가 2.4이후 점차 커지는 경향을 보인다.
- 시뮬레이션의 평균에서 프로젝트 활동 수행비용의 합은  $M$ 이 0.8일 때 2661로 최솟값을 가진다. 또한  $M$ 이 0.8에 최솟값을 가지고, 0.8이후 점차 커지는 경향을 보인다.
- 시뮬레이션의 평균에서 지체보상금은  $M$ 이 0부터 3.6까지 낮아지며, 3.6이후 크게 유의미한 변동은 없는 것으로 판단된다.

## 5.2.2 데이터셋 2의 모의실험 결과

### (1) 프로젝트 네트워크 - 데이터셋 2

데이터셋 2의 프로젝트 네트워크는 활동의 수를 20으로 한 후, 각 활동 간의 선행관계를 0.7의 확률로 임의로 생성하였다. 임의로 생성된 선행관계에서 정상적인 선행관계의 작성에 필요한 연결은 추가하였으며 필요 없는 연결은 모두 삭제하였으며, 그 결과는 [그림 5-5]과 같다.



[그림 5-5] 데이터셋 2의 프로젝트 네트워크

### (2) 재작업 가능 활동과 재작업 확률 - 데이터셋 2

데이터셋 2의 재작업 가능 활동의 수는 (1)의 프로젝트 네트워크에서 60%를 임의로 선택 하였으며, 재작업 확률은 10%에서 30%사이의 값을 임의로 생성하였다. 재작업 가능 활동은 [그림 5-5]의 네트워크에서 짙은 색 배경으로 표시되었으며, 그 결과는 [표 5-6]와 같다.

[표 5-6] 데이터셋 2의 재작업 가능 활동과 재작업 확률

재작업 가능 활동	재작업 확률	재작업 가능 활동	재작업 확률
2	0.14	11	0.14
4	0.13	12	0.23
5	0.22	15	0.19
6	0.22	16	0.16
7	0.27	17	0.27
9	0.12	18	0.16

(3) 활동의 모드별 수행기간과 수행비용 - 데이터셋 2

데이터셋 2의 활동의 모드별 수행기간은 10에서 82사이의 값을 임의로 생성하였으며, 각 수행별 비용은 50에서 576사이의 값을 임의로 생성하였다. 그 결과는 [표 5-7]과 같다.

[표 5-7] 데이터셋 2의 기간 및 비용

활동	기간					비용				
	모드1	모드2	모드3	모드4	모드5	모드1	모드2	모드3	모드4	모드5
1	14	16	20	30	57	201	102	69	56	52
2	11	13	16	23	43	397	217	132	100	90
3	13	14	18	26	50	390	200	133	105	86
4	13	13	16	25	45	259	134	92	71	62
5	15	17	22	35	65	465	235	146	106	97
6	15	17	21	32	57	361	184	116	89	73
7	11	11	13	20	39	369	189	126	91	86
8	14	15	18	29	57	316	162	101	75	70
9	14	15	19	29	53	301	159	98	77	62
10	12	12	15	21	38	418	215	131	97	80
11	13	14	18	28	55	353	185	123	94	78
12	12	14	19	28	51	329	180	117	91	73
13	12	14	17	25	47	327	176	108	90	82
14	11	12	15	23	43	386	204	125	104	88
15	15	16	20	30	55	237	120	78	62	53
16	11	12	14	21	41	294	149	92	72	58
17	13	13	15	22	42	228	122	78	62	55
18	13	14	17	27	51	225	113	74	56	53
19	11	12	14	20	36	355	187	116	94	77
20	13	16	21	32	63	431	216	136	106	98



(4) 활동의 모드별 필요자원 - 데이터셋 2

데이터셋 2의 활동의 모드별 필요자원의 종류는 2가지이며 그 요구량은 5에서 10사이의 값을 임의로 생성하였다. 그 결과는 [표 5-8]와 같다.

[표 5-8] 데이터셋 2의 필요자원

활동	모드1		모드2		모드3		모드4		모드5	
	자원1	자원2	자원1	자원2	자원1	자원2	자원1	자원2	자원1	자원2
1	10	7	9	9	8	7	8	8	8	5
2	9	5	6	7	7	7	9	5	9	6
3	5	9	5	6	8	6	9	6	6	8
4	8	8	6	10	7	7	5	9	7	7
5	10	6	8	5	9	7	8	7	6	9
6	6	6	7	7	9	6	7	9	8	9
7	6	5	8	8	8	7	8	6	10	7
8	9	6	7	5	9	9	6	7	8	5
9	8	10	5	9	5	10	10	8	8	6
10	6	10	8	9	6	6	6	5	7	10
11	6	9	9	5	9	9	6	6	8	5
12	8	7	10	6	6	8	6	7	5	5
13	10	9	7	7	7	9	7	9	7	7
14	6	6	5	9	9	7	7	10	8	7
15	9	7	6	10	8	9	8	7	10	7
16	7	6	9	9	6	10	8	9	9	7
17	8	9	6	9	7	8	5	5	9	10
18	5	8	10	6	7	7	10	6	9	8
19	6	5	10	7	8	5	5	10	6	9
20	7	6	6	9	9	6	7	9	9	7

(5) 시뮬레이션 결과 - 데이터셋 2

데이터셋 2에 대한 시뮬레이션은 하나의  $M$ 값에 대해 1,000회를 실행하였으며, [표 5-9]는 각각의  $M$ 값에 대한 시뮬레이션의 평균으로 작성된다.

[표 5-9] 데이터셋 2의 모의실험 결과

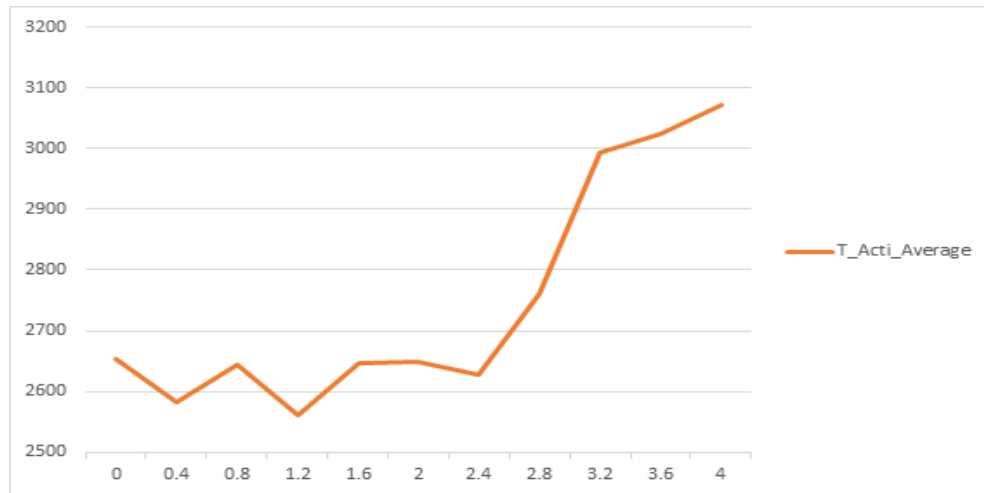
$M$	0	0.4	0.8	1.2	1.6	2	2.4	2.8	3.2	3.6	4
총 비용	3109	2885	2893	2702	2751	2765	2716	2825	3020	3053	3093
활동비용의 합	2654	2582	2644	2561	2646	2650	2629	2760	2994	3024	3072
지체보상금	455	303	249	141	105	115	88	65	25	29	21

[그림 5-6]는 [표 5-9]에서  $M$ 의 변화에 따른 총 비용의 변화를 나타내는 그래프이다.



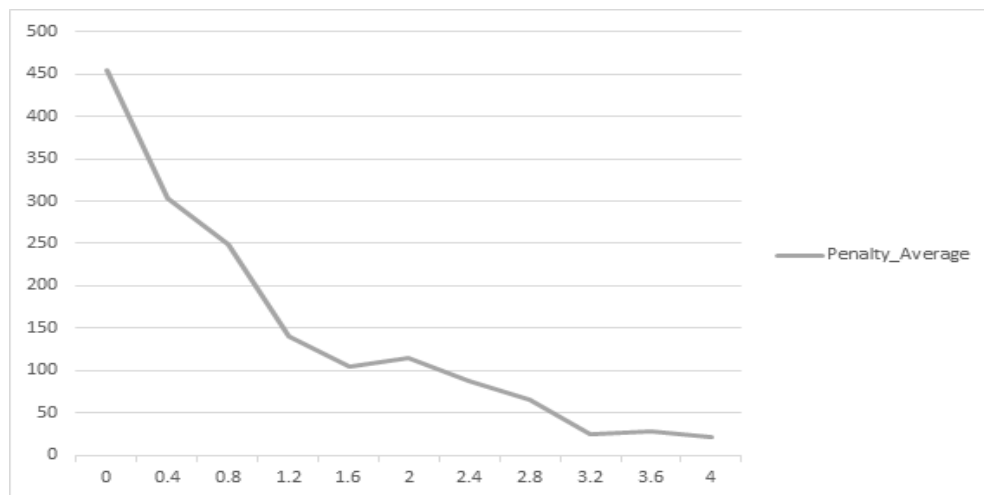
[그림 5-6] 데이터셋 2의 총 비용 그래프

[그림 5-3]는 [표 5-9]에서  $M$ 의 변화에 따른 활동비용의 합의 변화를 나타내는 그래프이다.



[그림 5-7] 데이터셋 1의 활동비용 합 그래프

[그림 5-8]는 [표 5-9]에서  $M$ 의 변화에 따른 지체보상금의 변화를 나타내는 그래프이다.



[그림 5-8] 데이터셋 2의 지체보상금 그래프

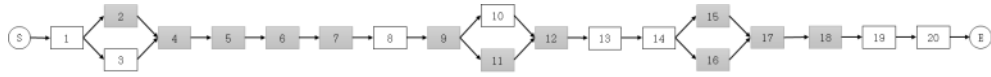
(6) 시뮬레이션 결과 요약 - 데이터셋 2

- 시뮬레이션의 평균에서 프로젝트의 총 비용은  $M=1.2$ 일 때 2,702으로 최솟값을 가진다. 또한  $M$ 이 0부터 1.2까지 총 비용은 점차 낮아지며, 2.4이후 점차 커지는 경향을 보인다.
- 시뮬레이션의 평균에서 프로젝트 활동 수행비용의 합은  $M$ 이 1.2일 때 2,561로 최솟값을 가진다. 또한  $M$ 이 0.0부터 2.4까지 불규칙한 값을 가지며, 2.4이후 점차 커지는 경향을 보인다.
- 시뮬레이션의 평균에서 지체보상금은  $M$ 이 0부터 3.2까지 지속적으로 낮아지며, 3.2이후 크게 유의미한 변동은 없는 것으로 판단된다.

### 5.2.3 데이터셋 3의 모의실험 결과

#### (1) 프로젝트 네트워크 - 데이터셋 3

데이터셋 3의 프로젝트 네트워크는 활동의 수를 20으로 한 후, 각 활동 간의 선행관계를 0.7의 확률로 임의로 생성하였다. 임의로 생성된 선행관계에서 정상적인 선행관계의 작성에 필요한 연결은 추가하였으며 필요 없는 연결은 모두 삭제하였으며, 그 결과는 [그림 5-9]과 같다.



[그림 5-9] 데이터셋 3의 프로젝트 네트워크

#### (2) 재작업 가능 활동과 재작업 확률 - 데이터셋 3

데이터셋 3의 재작업 가능 활동의 수는 (1)의 프로젝트 네트워크에서 60%를 임의로 선택 하였으며, 재작업 확률은 10%에서 30%사이의 값을 임의로 생성하였다. 재작업 가능 활동은 [그림 5-9]의 네트워크에서 짙은 색 배경으로 표시되었으며, 그 결과는 [표 5-10]와 같다.

[표 5-10] 데이터셋 3의 재작업 가능 활동과 재작업 확률

재작업 가능 활동	재작업 확률	재작업 가능 활동	재작업 확률
2	14	11	14
4	13	12	23
5	22	15	19
6	22	16	16
7	27	17	27
9	12	18	16

(3) 활동의 모드별 수행기간과 수행비용 - 데이터셋 3

데이터셋 3의 활동의 모드별 수행기간은 10에서 82사이의 값을 임의로 생성하였으며, 각 수행별 비용은 50에서 576사이의 값을 임의로 생성하였다. 그 결과는 [표 5-11]과 같다.

[표 5-11] 데이터셋 3의 기간 및 비용

활동	기간					비용				
	모드1	모드2	모드3	모드4	모드5	모드1	모드2	모드3	모드4	모드5
1	14	16	20	30	57	201	102	69	56	52
2	11	13	16	23	43	397	217	132	100	90
3	13	14	18	26	50	390	200	133	105	86
4	13	13	16	25	45	259	134	92	71	62
5	15	17	22	35	65	465	235	146	106	97
6	15	17	21	32	57	361	184	116	89	73
7	11	11	13	20	39	369	189	126	91	86
8	14	15	18	29	57	316	162	101	75	70
9	14	15	19	29	53	301	159	98	77	62
10	12	12	15	21	38	418	215	131	97	80
11	13	14	18	28	55	353	185	123	94	78
12	12	14	19	28	51	329	180	117	91	73
13	12	14	17	25	47	327	176	108	90	82
14	11	12	15	23	43	386	204	125	104	88
15	15	16	20	30	55	237	120	78	62	53
16	11	12	14	21	41	294	149	92	72	58
17	13	13	15	22	42	228	122	78	62	55
18	13	14	17	27	51	225	113	74	56	53
19	11	12	14	20	36	355	187	116	94	77
20	13	16	21	32	63	431	216	136	106	98

(4) 활동의 모드별 필요자원 - 데이터셋 3

데이터셋 3의 활동의 모드별 필요자원의 종류는 2가지이며 그 요구량은 5에서 10사이의 값을 임의로 생성하였다. 그 결과는 [표 5-12]와 같다.

[표 5-12] 데이터셋 3의 필요자원

활동	모드1		모드2		모드3		모드4		모드5	
	자원1	자원2	자원1	자원2	자원1	자원2	자원1	자원2	자원1	자원2
1	10	7	9	9	8	7	8	8	8	5
2	9	5	6	7	7	7	9	5	9	6
3	5	9	5	6	8	6	9	6	6	8
4	8	8	6	10	7	7	5	9	7	7
5	10	6	8	5	9	7	8	7	6	9
6	6	6	7	7	9	6	7	9	8	9
7	6	5	8	8	8	7	8	6	10	7
8	9	6	7	5	9	9	6	7	8	5
9	8	10	5	9	5	10	10	8	8	6
10	6	10	8	9	6	6	6	5	7	10
11	6	9	9	5	9	9	6	6	8	5
12	8	7	10	6	6	8	6	7	5	5
13	10	9	7	7	7	9	7	9	7	7
14	6	6	5	9	9	7	7	10	8	7
15	9	7	6	10	8	9	8	7	10	7
16	7	6	9	9	6	10	8	9	9	7
17	8	9	6	9	7	8	5	5	9	10
18	5	8	10	6	7	7	10	6	9	8
19	6	5	10	7	8	5	5	10	6	9
20	7	6	6	9	9	6	7	9	9	7

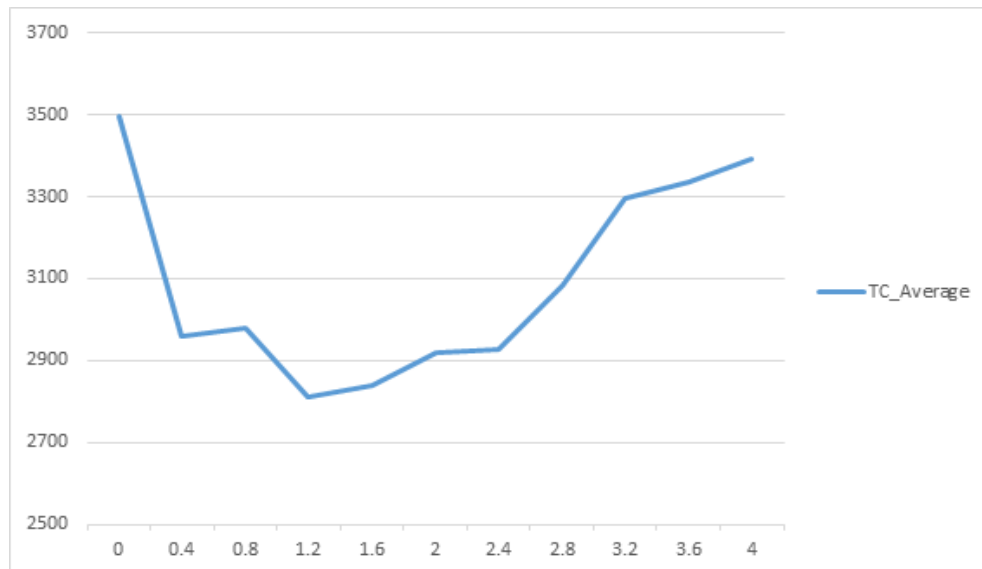
(5) 시뮬레이션 결과 - 데이터셋 3

데이터셋 1에 대한 시뮬레이션은 하나의  $M$ 값에 대해 1,000회를 실행하였으며, [표 5-13]는 각각의  $M$ 값에 대한 시뮬레이션의 평균으로 작성된다.

[표 5-13] 데이터셋 3의 모의실험 결과

$M$	0	0.4	0.8	1.2	1.6	2	2.4	2.8	3.2	3.6	4
총 비용	3497	2958	2979	2810	2839	2917	2928	3084	3297	3337	3392
활동비용의 합	2908	2724	2785	2712	2800	2879	2906	3064	3280	3327	3386
지체보상금	589	234	194	97	39	38	22	20	17	10	7

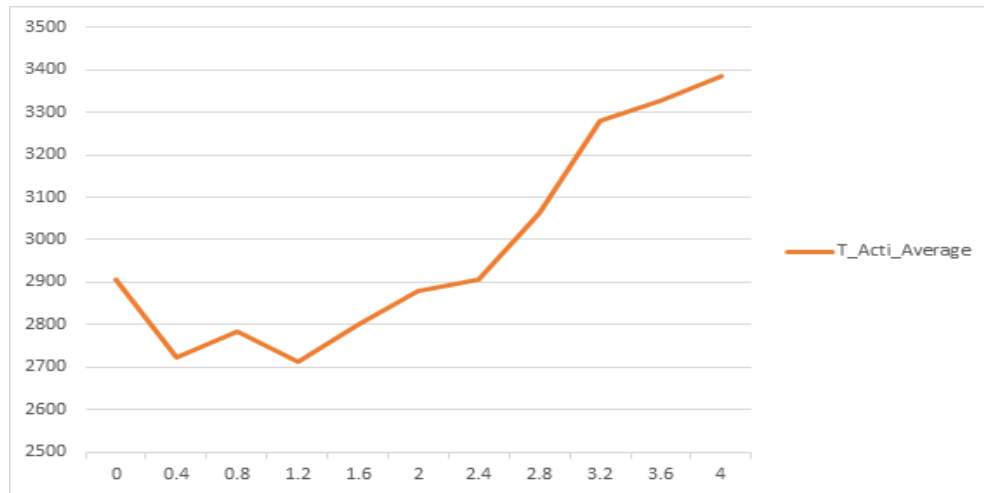
[그림 5-10]는 [표 5-13]에서  $M$ 의 변화에 따른 총 비용의 변화를 나타내는 그래프이다.



[그림 5-10] 데이터셋 3의 총 비용 그래프

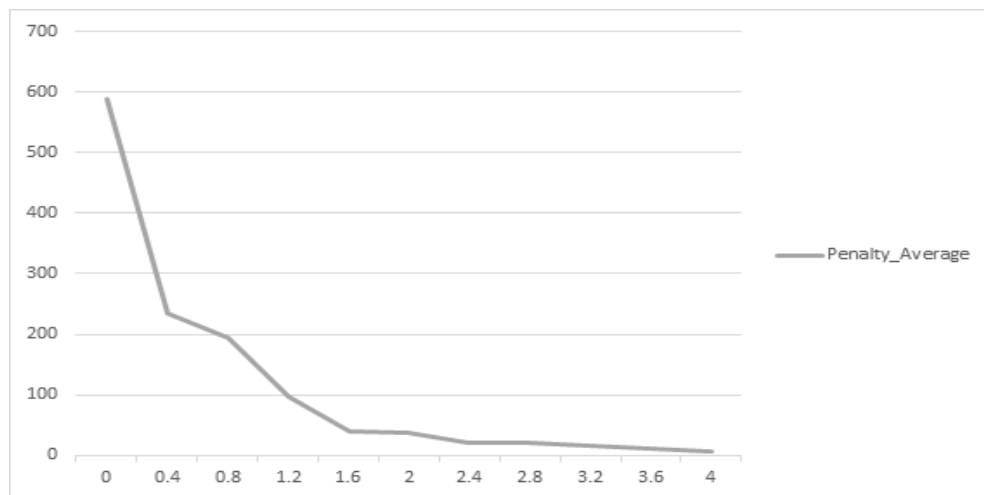


[그림 5-11]는 [표 5-13]에서  $M$ 의 변화에 따른 활동비용의 합의 변화를 나타내는 그래프이다.



[그림 5-11] 데이터셋 3의 활동비용 합 그래프

[그림 5-12]는 [표 5-13]에서  $M$ 의 변화에 따른 지체보상금의 변화를 나타내는 그래프이다.



[그림 5-12] 데이터셋 3의 지체보상금 그래프

(6) 시뮬레이션 결과 요약 - 데이터셋 3

- 시뮬레이션의 평균에서 프로젝트의 총 비용은  $M=1.2$ 일 때 2,810으로 최소값을 가진다. 또한  $M$ 이 0부터 1.2까지 총 비용은 점차 낮아지며, 1.2 이후 점차 커지는 경향을 보인다.
- 시뮬레이션의 평균에서 프로젝트 활동 수행비용의 합은  $M$ 이 1.2일 때 2,819로 최소값을 가진다. 또한  $M$ 이 1.2이후 점차 커지는 경향을 보인다.
- 시뮬레이션의 평균에서 지체보상금은  $M$ 이 0부터 1.6까지 급격히 낮아지며, 1.6이후 크게 유의미한 변동은 없는 것으로 판단된다.

### 5.3 모의실험 결과 요약

전체 프로젝트의 모의실험의 결과에 따르면, 프로젝트의 최소 평균을 도출하는  $M$ 값은 프로젝트마다 다르다. 하지만 모든 프로젝트에서  $M$ 값이 0인 경우 평균이 크고,  $M$ 이 증가하면 평균이 감소하다가 다시 증가하는 전형적인 Convex 형태가 관찰된다.

본 논문에 제시된 알고리즘은 해당 프로젝트에 적합한 추정방식을 구하고, 그러한 추정방식을 적용했을 때의 평균 비용을 구할 수 있다. 이때의 평균비용이 해당 프로젝트의 최소 평균비용이 된다.

## 제 6 장 결 론

본 논문은 프로젝트에서 발생할 수 있는 확률적 요소 중 하나인 재작업을 확정적 모델인 RCPSP(Resource Contained Project Scheduling Problem)에 추가하는 방법을 제안하였으며, 모의실험을 통하여 제시된 알고리즘이 실제로 작동하고 유효하다는 것을 입증하였다.

재작업을 RCPSP에 추가하는 방법은 다음과 같은 순서로 제시하였다. 첫째, RCPSP는 확정적 요소를 대상으로 하는 일정문제 중 하나로 불확실성을 지닌 재작업을 직접 적용할 수 없으므로 본 논문에서는 재작업을 재작업 확률이 있는 활동( $i$ )의 수행횟수( $R_i$ )가 추가되는 것으로 정의하여 수학적 모형( $Problem$ )을 제안하였다. 둘째,  $Problem$ 에 사용되는 활동의 수행횟수는 재작업 가능 활동에 대한 검수를 통과할 때 확정되는 확률적 요소이므로 사전에 추정할 필요가 있어 이를 활동별 수행횟수의 기댓값으로 추정하는 방법을 제안하였다. 셋째,  $Problem$ 에 수행횟수의 추정치가 적용된 추정모형( $Problem\_Est$ )과 그 목적함수값을 최소화하는 알고리즘을 제시하였다. 넷째, 다양한 활동횟수의 추정치( $Est\_R_i$ )를  $Problem$ 의 시뮬레이션에 적용하여 목적함수값의 분포를 구할 수 있는 절차를 제시하였다.

### 6.1 연구의 결과

일반적인 RCPSP는 전형적인 NP-Hard 문제로 해영역의 크기가 커지면 모든 해 영역을 탐색하는 것이 현실적으로 불가능하다. 따라서 휴리스틱기법을 이용하여 알고리즘을 작성하였으며, 재작업을 적용하기 위해 활동의 수행횟수의 기댓값으로부터 추정하였다. 본 연구에서 제안된 추정모형( $Problem\_Est$ )에 대한 알고리즘은 이웃탐색기법을 기반으로 한 휴리스틱 기법을 사용하였다. 제안된 확률적 모형( $Problem$ )의 모의실험에

서 기초 데이터는 활동 20, 모드 5, 자원종류 2는 고정되었으며, *Problem*의 시뮬레이션 각 문제마다 목적함수값의 분포를 구하기 위한 활동횟수의 추정치( $Est\_R_i$ )는 11가지, 각  $Est\_R_i$ 별로 1,000회의 목적함수값을 구하였으며, 목적함수값을 1회 구할 때마다 추정모형(*Problem\_Est*)에 대한 알고리즘은 초기일정작성에 100회에 재작업가능활동의 검수 마다 추가로 1회씩 일정을 작성한다.

수행횟수는 재작업이 발생했을 때 그 횟수의 정보가 갱신되며, 프로젝트 종료 시까지 단계적으로 확정된다. 따라서 일정의 작성 시점에는 알려지지 않은 값이다. 따라서 수행횟수를 재작업횟수의 기댓값으로부터 추정하였다. 수행횟수는 재작업 횟수에 1을 더한 값으로 재작업확률에 먹급수의 합의 공식을 사용하면 재작업 횟수에 대한 기댓값이 계산 가능하며 이로부터 수행횟수의 기댓값도 추정이 가능하다. 또한 기댓값으로 계산된 수행횟수의 추정치가 모든 프로젝트상황에 최선이라는 보장이 없기 때문에 다양한 추정치를 재작업이 적용된 시뮬레이션을 통해 적절한 추정치를 찾아보았다.

시뮬레이션의 일정작성 알고리즘에 사용한  $M$ 은 활동별 재작업 횟수에 대한 기댓값에 곱해지는 계수이며, 모의실험에서는 재작업 횟수에 대한 기댓값에  $M$ 을 곱하여 사용한다. 모의실험에서는 수행횟수의 추정치를 계산할 때  $M$ 을 0부터 4까지 0.4의 간격으로 적용하였다. 재작업을 적용한 시뮬레이션을 실험하기 위하여 총 3개의 데이터셋을 만들었으며,  $M$ 을 적용하여 작성한 일정에 따라 시뮬레이션을 진행하였다.

재작업이 적용된 시뮬레이션은 재작업이라는 확률변수로 인하여 목적함수값이 특정되지 않는다. 따라서 목적함수값의 평균을 구하기 위하여 해당  $M$ 에 대한 시뮬레이션을 1000회 하였다. 따라서 시뮬레이션은 11가지의  $M$ 과 3개의 데이터셋을 곱한 33,000번이 실행되었으며, 그 결과는

[표 6-1]과 같다.

[표 6-1]  $M$ 의 변화에 따른 총 비용의 변화

$M$	0	0.4	0.8	1.2	1.6	2	2.4	2.8	3.2	3.6	4
데이터셋 1	3544	3343	2923	2846	2809	2824	2803	2861	2921	2960	2981
데이터셋 2	3109	2885	2893	2702	2751	2765	2716	2825	3020	3053	3093
데이터셋 3	3497	2958	2979	2810	2839	2917	2928	3084	3297	3337	3392

수행횟수 추정치 공식에 따르면 [표 6-1]에서  $M$ 이 0인 경우는 수행횟수의 기댓값이 1(재작업 횟수에 대한 기댓값 0)로 적용된 것과 같으며, 실험에 적용된 3개의 데이터셋 중 데이터셋 1은  $M$ 이 2.4, 데이터셋 2와 3은  $M$ 이 1.2에서 가장 좋은 결과가 나왔다.

[표 6-2]  $M=0$ 일 때와 최저 평균  $M$ 일 때의 비율

데이터셋	① $M=0$ 일 때	② 최저 평균일 때	비율(②/①)
1	3544	2803	79%
2	3109	2702	87%
3	3497	2810	80%

[표 6-2]의 결과에 따르면 데이터셋 1의 경우 본 논문에서 제시된 모형에  $M$ 을 2.4로 적용하여 일정을 작성한 경우 재작업이 고려되지 않은 RCPSP를 이용하여 일정을 작성한 것 보다. 총 비용에서 21%의 절감을 가져온다. 데이터셋 2와 3의 경우도 각각 13%, 20% 절감됨을 알 수 있다. 따라서 본 논문에서 제시하는 모델이 모의실험 데이터에서 유효함을

보여준다.

## 6.2 연구의 시사점

본 연구에서는 확률적 모델에서 확률적인 부분을 확정적으로 추정하여 확정적 모델로 만든다. 이는 본 연구에 제시된 조건(재작업의 발생확률이 알려져 있고, 재작업 발생 시 추가되는 기간이 정해져 있는 경우)과 동일한 형태의 불확정적 요소가 있는 경우, 확정적으로 추정하는 방식에 대한 방법을 제시하였다는 점에 의의가 있다. 본 연구와 같은 문제에서 결과 값은 확률적인 요소로 인한 불확실성과 확정적으로 추정하는 방식에 따라 결정된다. 그런데 확률적인 요소는 확률이 알려진 경우 수학적 기댓값으로 계산이 가능하다. 따라서 체계적이고 효과적으로 추정하는 방식을 제안하는 것은 필요하다.

또한 본 연구에서는 시점  $t$ 에서의 일정을 작성하기 위한 추정모형 (*Problem\_Est*)을 제시하고 있다. 이는 프로젝트의 일정작성에 RCPSP가 매우 현실적인 대안임을 고려하면 실제 프로젝트가 진행 중일 때 일정을 작성하거나 분석할 때에도 도움이 될 수 있다.

예를 들어, 어떤 IT 소프트웨어 개발 프로젝트에서 일정을 분석할 때 고려해야하는 사항이 RCPSP의 제약조건(제한된 인력자원과 마감기한 등)과 동일하고, 설계 또는 디자인 등 해당 활동의 평균적인 재작업확률이 알려져 있을 경우, 해당 프로젝트가 진행 중일 때도 추정모형(*Problem\_Est*)을 적용하여 분석하거나 일정을 재작성할 수도 있을 것이다. 물론 프로젝트가 진행되기 전에도 기존의 RCPSP를 사용하여 일정을 작성하거나 분석하는 경우와 같이 재작업 가능 활동이 포함된 프로젝트의 일정의 작성과 분석에도 사용할 수 있다.

이와 같이 재작업 가능 활동이 포함된 프로젝트는 본 논문에서 제안된

알고리즘을 사용하여 프로젝트의 계획단계이전에도, 진행 중에도 재작업 확률에 대한 데이터가 있거나 그 확률을 추정할 수 있다면, 해당 프로젝트에 대한 이익을 판단하여 프로젝트의 수주여부 또는 진행여부에 대한 의사결정에 도움이 될 것이다.

그리고 RCPSP에 재작업을 추가하여 일정을 작성하거나 분석한다는 것은 프로젝트의 실행단계에서 발생할 것으로 예정하고 있는 위험(Risk)에 대한 비용을 계획단계에서 일정에 대한 추가기간으로 대체할 수 있다는 것이다. 이는 PMI(2017)의 PMBOK Guide에서 설명하고 있는 품질비용(COQ)의 내부 실패 비용 중 일정 추가에 대한 추가비용을 일정관리의 영역에서 일정을 작성할 때 분담할 수 있다는 뜻이기도 하다.

예를 들어 프로젝트 관리자가 확정적 모델만을 사용하여 일정을 작성하였다고 하자. 이때 불확실성이 있는 확률적인 요소들은 우발적으로 발생하는 위험비용으로 대처해야 하는 것이 일반적이다. 재작업의 경우도 일정기간의 산정에 임의적으로 여유기간을 추가하는 방법 외에는 일반적으로 원가에서 발생할 수 있는 예비비로 관리한다(PMI, 2017). 본 연구에서 제안된 알고리즘을 해당 프로젝트의 일정계획 또는 분석에 이용할 수 있다면, 재작업 기간을 고려하여 활동에 추가하는 기간을 체계적으로 산정할 수 있을 것이다. 따라서 본 연구에서 제안된 알고리즘은 재작업이 있는 프로젝트에서 그 확률이 알려져 있을 경우 일정을 작성하거나 분석하는 것에 있어서 보다 체계적인 대안이 될 것이다.

### 6.3 연구의 한계 및 향후 연구과제

본 연구의 모의실험 데이터에서의 재작업은 현실의 프로젝트에서 재작업과 다를 것이다. 첫째로 현실의 모든 재작업이 본 논문에서 정의된 재작업처럼 원래의 작업과 동일한 시간과 비용이 추가되지 않을 것이다. 어떤

재작업은 활동의 전체를 재작업하지 않고 수정으로 인해 추가되는 기간이 30%일 것이고, 어떤 재작업은 자원을 구할 수 없어서 재작업이 불가능할 수도 있다. 그러므로 본 논문에서 제안된 알고리즘은 이 부분에 대해 한계를 가지고 있다.

두 번째로 모든 활동의 재작업확률이 재작업이 발생할 때마다 똑같은 재작업 확률을 가지지 않을 것이다. 모의실험에서는 작업의 반복횟수와 상관없이 활동의 재작업확률은 동일하게 적용되었다. 그러나 반복횟수에 따라 재작업확률이 다르게 적용될 수도 있다. 예를 들어 알려진 확률로 오류를 일으키는 기계가 그로인해 재작업을 하는 경우라면 동일한 재작업확률을 적용할 수 있을 것이다. 그러나 주어진 요구사항에 따른 상품의 설계나 디자인의 경우는 재작업횟수가 증가할 때 마다 성공확률이 높아질 것이다. 따라서 이 부분에 대한 한계를 가지고 있다.

세 번째로 본 연구의 모의실험에서는 네트워크의 주경로(Critical Path)의 활동과 주경로가 아닌 활동에 대한 확률적 기댓값에 대한 추정치 변환에 적용되는  $M$ 값이 모든 재작업 가능 활동에 동일하게 적용되었다. 그러나 여유(Slack)기간이 많은 경로에서 활동이 재작업 되는 것과 주경로의 활동이 재작업 되는 것은 프로젝트 총비용의 증가에 관여되는 정도가 다를 것이다. 주경로의 활동이 재작업 되는 것은 프로젝트 종료일정에 직접적으로 연관되며, 여유가 있는 경로의 활동이 재작업 되는 것은 프로젝트 종료일정에 적게 연관될 것이다. 따라서 향후 이에 대한 연구가 필요할 것이다.

네 번째로 본 모의실험에서는 일반적인 RCPSP에서도 다루고 있는 일정 단축(Crashing)과 같은 기법을 적용하지 않았다. 즉, 본 연구의 해의 영역은 활동의 시작시간과 모드의 선택에 대한 결정이며, 이는 본 모형과 같은 해의 영역을 가진 RCPSP의 경우 즉시 적용할 수 있다는 장점과 함께 본



모형과 다른 해의 영역을 가진 RCPSP는 추가적인 개발이 필요하다는 한계를 가지고 있다.

마지막으로 본 연구의 모의실험에서 사용된 데이터는 가상의 자료로 현실과 차이가 있을 수 있다. 따라서 실제의 데이터를 수집하여 적용하고 분석해서 나오는 현실의 결과와는 다를 수 있는 한계를 가지고 있다.

본 연구는 앞에서 서술한바와 같은 한계를 가지고 있으며, 향후 본 연구에서 다루지 못한 부분에 대해 추가적인 연구와 지속적으로 필요하다.

## 참 고 문 헌

### [국내 문헌]

- 이상범. (2000). *현대 생산·운영관리* 제2판. 도서출판 명경사.
- 김갑식. (2016). *자원 제약을 고려한 프로젝트 일정계획에서 활동의 품질과 기간과 및 비용 간 트레이드오프문제: 선택 가능한 모드가 복수인 경우*. 숭실대학교 대학원 박사학위논문.
- 남재덕. (2008). *프로젝트 포트폴리오 평가 문제: 불확실성 하의 최초 구성 및 주기적 개장을 중심으로*. 숭실대학교 대학원 박사학위논문.
- 류미애. (2017). *자원제약을 고려한 다단계 프로젝트 일정 문제*. 숭실대학교 대학원 박사학위논문.
- 민택기. (2005). *프로젝트 포트폴리오 평가문제: 시뮬레이션을 이용한 다속성 평가를 중심으로*. 숭실대학교 대학원 박사학위논문.
- 안정일. (2018). *반도체 측정공정을 위한 재측정 디스패칭 룰: 페널티기법을 이용한 불확실성하의 스케줄링을 중심으로*. 숭실대학교 대학원 박사학위논문.
- 안태호. (1998). 자원 제약을 고려하여 기간 단축이 가능한 복수의 양식을 지닌 단일 프로젝트 일정 문제: 자원 가용량이 시간에 따라 변하는 경우의 휴리스틱 해법. *한국OA 학회지*, 3(4), pp. 154-163.
- 이현기. (2017). *자원 제약을 고려한 프로젝트 일정문제: 기간 단축이 가능한 복수의 모드를 갖고 자원의 가용량을 변경할 수 있는 경우*. 숭실대학교 대학원 박사학위논문.
- 이현주. (2019). *패션의류 유통산업의 매장 간 수평이동 배분 알고리즘에 관한 연구*. 숭실대학교 대학원 박사학위논문.

[국외 문헌]

- Project Management Institute (PMI). (2017). *A Guide To The Project Management Body Of Knowledge* - sixth ed. Pennsylvania: Project Management Institute, Inc.
- Kerzner, H. (2017). *PROJECT MANAGEMENT: A Systems Approach to Planning, Scheduling, and Controlling* - 12th ed. New Jersey: John Wiley and Sons Inc.
- Tara, H. A. (2011). *Operation Research: An Introduction* - 9th ed. Chicago: Prentice Hall.
- Abdul-Rahman, H. (1995). The cost of non-conformance during a highway project: a case study. *Constr. Manag. Econ*, 13(1), pp. 23-32.
- 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, pp. 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), pp. 250-259.
- Alsayegh, H., & Hariga, M. (2012). Hybrid meta-heuristic methods for the multi-resource leveling problem with activity splitting. *Automation in Construction*, 27(1), pp. 89-98.

- Arundachawat, P., Roy, R., Al-Ashaab, A., & Shehab, E. (2009). Design rework prediction in concurrent design environment: current trends and future research directions. in *Proceedings of the 19th CIRP Design Conference - Competitive Design*. Cranfield University Press
- Ballestin, F., & Leus, R., (2009). Resource-constrained project scheduling for timely project completion with stochastic activity durations. *Production and Operations Management*, 18(4), pp. 459-474.
- Barber, P., Graves, A., Hall, M., Sheath, D., & Tomkins, C. (2000). Quality failure costs in civil engineering projects. *International Journal of Quality & Reliability Management*, 17(4/5), pp. 479 - 492.
- 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), pp. 7343-7356.
- Blazewicz, J. (1978). Complexity of computer scheduling algorithms under resource constraints. In *Proc. First Meeting AFCET-SMF on Applied Mathematics*, 16(6), pp. 169-178.
- Bowman, E. H. (1959). The schedule-sequencing problem. *Operations Research*, 7(5), pp. 621-624.
- Burati, J.L., Farrington, J.J., & Ledbetter, W.B. (1992). Causes of quality deviations in design and construction. *J. Constr. Eng.*

- Manag*, 118(1), pp. 34 - 49.
- Carson, Y., & Maria, A. (1997). Simulation optimization: methods and applications. In *Proceedings of the 29th conference on Winter simulation*, pp. 118-126. IEEE Computer Society.
- Chaari, T., Chaabane, S., Aissani, N., & Trentesaux, D. (2014). Scheduling under uncertainty: Survey and research directions. In *Advanced Logistics and Transport (ICALT). 2014 International Conference on*, pp. 229-234. IEEE.
- Chand, S., Singh H., & Ray T. (2019). Evolving heuristics for the resource constrained project scheduling problem with dynamic resource disruptions. *Swarm and Evolutionary Computation*, 44, pp. 897 - 912.
- Chand, S., Singh, H. & Ray, T. (2019). Evolving rollout-justification based heuristics for resource constrained project scheduling problems. *Swarm and Evolutionary Computation*, 50, pp. 100556.
- Chen, Z., Demeulemeester, E., Sijun, B., & Yuntao, G., (2018). Efficient priority rules for the stochastic resource-constrained project scheduling problem. *European Journal of Operational Research*, 270(3), pp. 957-967.
- Chua Z., Xua Z., & Lib H. (2019). New heuristics for the RCPSP with multiple overlapping modes. *Computers & Industrial Engineering*, 131, pp. 146 - 156.
- Costa, R., & Sobek, D. (2003). Iteration in engineering design: inherent and unavoidable or product of choices made?. in *ASME*

- 2003 *International design engineering technical conferences and Computers and information in engineering conference*, pp. 669–674. American Society of Mechanical Engineers.
- Coughlan, E. T., Lübbecke, M. E., & Schulz, J. (2010). A branch-and-price algorithm for multi-mode resource leveling. *In: Festa P. (eds) Experimental Algorithms. SEA 2010. Lecture Notes in Computer Science, 6049*, pp. 226–238. Springer, Berlin, Heidelberg.
- Creemers, S. (2015). Minimizing the expected makespan of a project with stochastic activity durations under resource constraints. *Journal of Scheduling, 18*(3), pp. 263 – 273.
- Creemers, S., Leus, R., & Lambrecht, M. (2010). Scheduling Markovian PERT networks to maximize the net present value. *Operations Research Letters, 38* (1), pp. 51 – 56.
- Davari, M., & Demeulemeester, E. (2019). A novel branch-and-bound algorithm for the chance-constrained resource-constrained project scheduling problem. *International Journal of Production Research, 57*(4), pp. 1265–1282
- Davis, E. W. (1973). Project scheduling under resource constraints historical review and categorization of procedures. *AIIE Transactions, 5*(4), pp. 297–313.
- Davis, E. W., & Heidorn, G. E. (1971). An Algorithm For Optimal Project Scheduling Under Multiple Resource Constraints. *Management Science, 17*(12), pp. B-803.

- Debels, D., & Vanhoucke, M. (2007). A decomposition-based genetic algorithm for the resource-constrained project-scheduling problem. *Operations Research*, 55(3), pp. 457-469.
- Demeulemeester, E., De Reyck, B., & Herroelen, W. (2000). The discrete time/resource trade-off problem in project networks: a branch-and-bound approach. *IIE Transactions*, 32(11), pp. 1059-1069.
- De Reyck, B., Demeulemeester, E., & Herroelen, W. (1998). Local search methods for the discrete time/resource trade-off problem in project networks. *Naval Research Logistics Quarterly*, 45(1), pp. 553-578.
- 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), pp. 642-653.
- Fang, C., Kolisch, R., Wang, L., & Chundi, M. (2015). An estimation of distribution algorithm and new computational results for the stochastic resource-constrained project scheduling problem. *Flexible Services and Manufacturing Journal*, 27(4), pp. 585 - 605.
- Garey, M. R., Johnson, D. S., & Sethi, R. (1976). The complexity of flowshop and jobshop scheduling. *Mathematics of operations research*, 1(2), pp. 117-129.
- Geiger, M. J. (2017). A multi-threaded local search algorithm and computer implementation for the multi-mode, resource-constrained

- multi-project scheduling problem, *European Journal of Operational Research*, 256(3), pp. 729-741.
- Pritsker, A. A. B. (1966). *GERT Graphical evaluation and review technique*. Memorandum RM-4973-NASA, Rand Corporation.
- Hartmann, S. (2001). Project scheduling with multiple modes: a genetic algorithm. *Annals of Operations Research*, 102(Issue1-4), pp. 111-135.
- Hegazy, T., Said, M., & Kassab, M. (2011). Incorporating rework into construction schedule analysis. *Autom. Constr.*, 20(8), pp. 1051 - 1059.
- 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*, pp. 1-31.
- Herroelen, W., & Leus, R. (2001). On The Merits And Pitfalls Of Critical Chain Scheduling. *Journal of Operations Management*, 19(5), pp. 559-577.
- Hwang, B.-G., Thomas, S.R., Haas, C.T., & Caldas, C.H. (2009). Measuring the impact of rework on construction cost performance. *J. Constr. Eng. Manag.*, 135(3), pp. 187 - 198.
- Jerry, B. (1984). *Discrete-event system simulation*. Pearson Education India.
- Johnson, T. J. R. (1967). *An algorithm for the resource constrained project scheduling problem* (Doctoral dissertation, Massachusetts



Institute of Technology). Cambridge.

Josephson, P. E., & Hammarlund, Y. (1999). The causes and costs of defects in construction: a study of seven building projects. *Autom. Constr.*, 8(6), pp. 681 - 687.

Josephson, P. E., Larsson, B., & Li, H. (2002). Illustrative benchmarking rework and rework costs in Swedish construction industry. *Journal of Management in Engineering*, 18(2), pp. 76 - 83.

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), pp. 758-768.

Kellenbrink, C., & Helber, S. (2015). Scheduling Resource-Constrained Projects with a Flexible Project Structure, *European Journal of Operational Research*, 246(2), pp. 379-391.

Kelley, J. E., & Walker, M. R. (1959). Critical-path Planning And Acheduling. In *Papers presented at the December 1-3, 1959, eastern joint IRE-AIEE-ACM computer conference*, pp. 160-173.

Kennedy, B. M., Sobek, D. K., & Kennedy, M. N. (2014). Reducing Rework by Applying Set Based Practices Early in the Systems Engineering Process. *Systems Engineering*, 17(3), pp. 278-296.

Kochenderfer, M. J. (2015). *Decision making under uncertainty: theory and application*. MIT press.

Li, C., & Lim, A. (2018). A greedy aggregation - decomposition

- method for intermittent demand forecasting in fashion retailing, *European Journal of Operational Research*, 269(3), pp. 860–869.
- Li, H., & Demeulemeester, E. (2016). A genetic algorithm for the robust resource leveling problem. *Journal of Scheduling*, 19(1), pp. 43–60.
- Li, Z., & Ierapetritou, M. (2008). Process scheduling under uncertainty: Review and challenges. *Computers & Chemical Engineering*, 32(4), pp. 715–727.
- 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), pp. 69–86.
- Love, P. E. D., (2002). Influence of project type and procurement method on rework costs in building construction projects. *Journal of Construction Engineering and Management*, 128(1), pp. 18 - 29.
- Lu, M., & AbouRizk, S. M. (2000). Simplified CPM/PERT Simulation Model. *Journal of Construction Engineering and Management*, 126(3), pp. 219–226.
- Malcolm, D. G., J. H. Roseboom, C. E. Clark & W. Frazar. (1959). Application of a Technique for Research and Development Program Evaluation. *Operations Research*, 7(5), pp. 646–669.
- Mitchell, V. L., & Nault, B. R. (2007). Cooperative planning, uncertainty, and managerial control in concurrent design. *Management Science*, 53(3), pp. 375–389.

- Moslehi, G., & Mahnam, M. (2011). A Pareto approach to multi-objective flexible job-shop scheduling problem using particle swarm optimization and local search. *International Journal of Production Economics*, 129(1), pp. 14-22.
- 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), pp. 1322-1329.
- Naber, A., & Kolisch, R. (2014). MIP Models for resource-constrained project scheduling with flexible resource profiles, *European Journal of Operational Research*, 239(2), pp. 335-348.
- 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), pp. 227-242.
- Patterson, J. H. (1973). Alternate methods of project scheduling with limited resources. *Naval Research Logistics Quarterly*, 20(4), pp. 764-784.
- Patterson, J. H., & Huber, W. H. (1974). A horizon-varying, zero-one approach to project scheduling. *Management Science*, 20(6), pp. 990-998.
- Patterson, J. H., & Roth, G. (1976). Scheduling a project under multiple resource constraints: a zero-one approach. *AIIE Transactions*, 8(4), pp. 449-456.
- Patterson, J. H., Slowinski, R., Talbot, F. B., & Weglarz, J. (1989).

- An algorithm for a general class of precedence and resource constrained scheduling problems. *Advances in Project Scheduling*, 187(1), pp. 3-28.
- 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), pp. 68-79.
- Pellerin R., Perrier N., & Berthaut F. (2019). A survey of hybrid metaheuristics for the resource-constrained project scheduling problem. *European Journal of Operational Research*, 280, pp. 395 - 416
- 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), pp. 161-172.
- Raghavan, V. A., Yoon, S. W., & Srihari, K. (2018). A modified Genetic Algorithm approach to minimize total weighted tardiness with stochastic rework and reprocessing times. *Computers & Industrial Engineering*, 123, pp. 42 - 53
- Reichelt, K. Lyneis, J. (1999). The Dynamics of Project Performance: Benchmarking the Drivers of Cost and Schedule Overrun. *European Management Journal*, 17(2), pp. 135-150
- Rostami, S., Creemers, S., & Leus, R. (2018). New strategies for

- stochastic resource-constrained project scheduling. *Journal of Scheduling*, 21(3), pp. 349-365
- Scharge, L. (1970). Solving resource-constrained network problems by implicit enumeration - non-preemptive case. *Operation Research*, 18(2), pp. 263-278.
- Speranza, M. G., & Vercellis, C. (1993). Hierarchical models for multi-project planning and scheduling. *European Journal of Operational Research*, 64(1), pp. 312-325.
- Sprecher, A. (1994). *Resource-constrained project scheduling: Exact methods for the multi-mode case*. Lecture Notes in Economics and Mathematical Systems, Vol. 409, Berlin: Springer-Verlag.
- Sprecher, A. (2002). Network Decomposition Techniques for Resource Constrained Project Scheduling. *Journal of the Operational Research Society*, 53(4), pp. 405-414.
- Sprecher, A., & Drexl, A. (1998). Multi-mode resource-constrained project scheduling by a simple, general and powerful sequencing algorithm. *European Journal of Operational Research*, 107(1), pp. 431-450.
- Slowinski, R., Soniewicki, B., & Węglarz, J. (1994). DSS for multi-objective project scheduling. *European Journal of Operational Research*, 79(2), pp. 220-229.
- Smith, R. P., & Eppinger, S. D. (1997). Identifying controlling features of engineering design iteration. *Management science*, 43(3), pp. 276-293.

- Stinson, J. P., Davis, E. W., & Khumawala, B. M. (1978). Multiple resource - constrained scheduling using branch and bound. *AIIE Transactions*, 10(3), pp. 252-259.
- Stork, F. (2001). *Stochastic Resource-Constrained Project Scheduling*. Ph.D. Thesis. Technische Universität Berlin.
- Talbot, F. B. (1982). Resource-constrained project scheduling problem with time-resource trade-offs: the non-preemptive case. *Management Science*, 28(1), pp. 1197-1210.
- Taylor, T., & Ford, D. N. (2006). Tipping point failure and robustness in single development projects. *System Dynamics Review: The Journal of the System Dynamics Society*, 22(1), pp. 51-71.
- 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), pp. 409-418.
- Wagner, H. (1959). An integer programming model for machine scheduling. *Naval Research Logistics Quarterly*, 6(1), pp. 131-140.
- Wang, X., & Ning, Y. (2018). Uncertain chance-constrained programming model for project scheduling problem. *Journal of the operational research society*, 69(3), pp. 384-391.
- Weglarz, J. (1979). Project scheduling with discrete and continuous resources. *IEEE Transactions on Systems*, 9(10), pp. 644-650.
- Werner, S., Horn, S., Weigert, G., & Jähnig, T. (2006). Simulation based scheduling system in a semiconductor backend facility. In

*Proceedings of the 38th conference on Winter simulation*, pp. 1741-1748.

Wynn, D. C., & Eckert, C. M. (2017). Perspectives on iteration in design and development. *Research in Engineering Design*, 28(2), pp. 153-184.

Zhang, H., Jiang, Z., & Guo, C. (2007). Simulation based real-time scheduling method for dispatching and rework control of semiconductor manufacturing system. In *2007 IEEE International Conference on Systems, Man and Cybernetics*, pp. 2901-2905.