



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](#)

공학석사학위논문

인력 공백 상황에서 실질 지연 분석을 활용한  
프로젝트 일정 재조정 방법론

Project Rescheduling Methodology for Workforce  
Absence Situation using Net Delay Analysis

2023 년 8 월

서울대학교 대학원  
산업공학과

조 윤 형

인력 공백 상황에서 실질 지연 분석을 활용한  
프로젝트 일정 재조정 방법론

Project Rescheduling Methodology for Workforce Absence  
Situation using Net Delay Analysis

지도교수 홍 유 석

이 논문을 공학석사 학위논문으로 제출함

2023 년 8 월

서울대학교 대학원

산업공학과

조 윤 형

조윤형의 공학석사 학위논문을 인준함

2023 년 8 월

위 원 장 \_\_\_\_\_ 이 덕 주 \_\_\_\_\_ (인)

부위원장 \_\_\_\_\_ 홍 유 석 \_\_\_\_\_ (인)

위 원 \_\_\_\_\_ 이 성 주 \_\_\_\_\_ (인)

## 초록

제품 개발 과정은 더 혁신적인 제품을 빠르게 개발하고 출시하기 위해 서로 다른 전문성을 보유한 팀들이 협업하여 다양한 의사결정을 내리는 일련의 활동들로 구성된 프로젝트라고 볼 수 있다. 복잡하게 얽혀 있는 활동들이 다양한 기술을 보유한 팀들에 의해 제품 개발에 대한 의사결정 결과를 주고 받는 제품 개발 프로젝트는 다중 기술 자원 프로젝트 스케줄링 문제(Multi-skilled resource-constrained project scheduling problem)로 볼 수 있고, 이 경우 프로젝트의 기간을 최소로 설정하는 것이 제품 개발의 궁극적인 목표라고 할 수 있다. 하지만 직원의 잦은 이직, COVID-19 등의 질병으로 인한 직원의 병가, 나아가 기계 장비 정비 등 다양한 사유로 인해 필요한 순간에 필요한 자원이 부재한다면 계획한대로 프로젝트를 정상적으로 수행할 수 없게 되고, 정해진 기간을 넘어 프로젝트가 지연되는 결과를 초래할 수 있다. 본 연구는 프로젝트에 일어난 변경 사항에 맞춰 프로젝트 일정을 수정하는 reactive scheduling의 방식에 기반해서 인력 공백 이후 자원 재배치를 통해 프로젝트가 지연되는 것을 최대한 막고자 하고, 새로운 인력 없이 프로젝트에 남아 있는 인원들을 활용하는 방법을 개발한다. 본 연구에서 제안하는 실질 지연 기반의 프로젝트 재스케줄링 방법론은 대체 자원을 투입했을 때 발생하는 활동의 지연이 프로젝트 전체의 지연에 기여하는 정도를 실질 지연으로 정의하고, 이를 가장 작게 만드는 자원을 선정하여 인력 공백이 발생한 활동에 임시로 배치한다. 특정 활동에 대해 자원이 가지는 적합도라는 기준을 적용하여 실질 지연을 계산하고, 실질 지연을 최소화할

수 있는 전략을 통해 가장 적합한 대체 자원을 선정한다. 나아가 PSPLIB을 기반으로 한 가상 예제를 통해 다른 자원 할당 전략에 비해 본 연구의 방법론이 지연 최소화에 더 효과적임을 보인다.

**주요어:** 프로젝트 스케줄링, 인력 공백, project rescheduling, 실질 지연, 산업공학

**학번:** 2021-21043

# 목차

초록	ii
목차	iv
표 목차	vi
그림 목차	vii
제 1 장 서론	1
제 2 장 선행 연구	5
2.1 다중 기술 자원 프로젝트 스케줄링 .....	5
2.2 스케줄 변경 관리 연구 .....	6
2.3 프로젝트 일정 재조정 문제 .....	9
2.4 연구 의의 .....	11
제 3 장 프로젝트 재스케줄링 방법론	14
3.1 문제 상황 .....	14
3.2 실질 지연 기반 재스케줄링 휴리스틱 .....	18
3.2.1 연관 활동 .....	20
3.2.2 연관 활동의 실질 지연 도출 .....	21

3.2.3	실질 지연을 활용한 대체 자원 선정 및 일정 업데이트.....	27
<b>제 4 장</b>	<b>실험 및 분석</b>	<b>30</b>
4.1	실험 데이터 .....	30
4.2	실험 방법 .....	34
4.2	실험 결과 분석.....	37
<b>제 5 장</b>	<b>결론 및 추후 연구 방향</b>	<b>41</b>
5.1	결론 .....	41
5.2	한계점 및 추후 연구 방향 .....	42
	<b>참고문헌</b>	<b>45</b>
	<b>부록</b>	<b>50</b>
	<b>Abstract</b>	<b>55</b>

## 표 목차

표 3.1	예시 프로젝트 활동 정보	15
표 3.2	예시 프로젝트 인적 자원 정보	16
표 3.3	예시 프로젝트의 일정 계획	17
표 4.1	프로젝트 1 활동 정보	32
표 4.2	프로젝트 1 인적 자원 정보	33
표 4.3	비교 검증을 위해 활용한 대체 자원 선정 규칙	36
표 4.4	대체 자원 선정 규칙에 따른 프로젝트 지연 결과 비교	38
표 A.1	프로젝트 2 활동 정보	51
표 A.2	프로젝트 2 인적 자원 정보	52
표 A.3	프로젝트 1 일정 계획	53
표 A.4	프로젝트 2 일정 계획	54



## 그림 목차

그림 2.1	강건한 스케줄링을 위한 방법론 .....	8
그림 3.1	예시 프로젝트 네트워크 .....	15
그림 3.2	실질 자연 기반의 재스케줄링 휴리스틱 흐름도 .....	19
그림 3.3	인력 공백 활동 및 유휴 자원 파악 .....	22
그림 3.4	대체 자원 후보들의 다음 활동 및 연관 활동 분석 .....	25
그림 3.5	실질 자연 기반의 재스케줄링 휴리스틱 세부 흐름도 .....	28

# 제 1 장 서론

경쟁이 점점 치열해지고 있는 현대 사회에서 혁신적인 제품이 더욱 빠르게 개발되기를 바라는 소비자들의 요구가 강해지고 있고(Eppinger 등, 1994) 기업에서는 이에 대응하여 여러 가지 방법을 동원하여 제품 개발 속도를 향상시키고 제품 출시 시점을 앞당기고자 노력하고 있다. 특히 하드웨어 제품에 비해 상대적으로 짧은 주기로 개발이 이루어지는 소프트웨어 시스템의 경우에는 급변하는 고객의 수요와 기술 요구사항 등을 만족시키는 동시에 시스템을 빠르게 개발하기 위해서 애자일 소프트웨어 개발 방법론(agile software development)을 적용한다(Abrahamsson 등, 2017).

제품 개발 과정을 하나의 프로젝트로 본다면 기업의 경쟁력을 제고하기 위해 프로젝트의 기간을 단축시키는 것이 중요하다. 성공적인 제품의 개발을 위해 다양한 기능의 팀들이 제품 개발 프로젝트에 참여하고, 여러 엔지니어들의 긴밀한 협업 아래 제품의 개발부터 생산, 그리고 출시 이후 소비자가 사용하는 일련의 과정 속에서 발생하는 다양한 이슈들을 효과적으로 다룰 수 있다(Nevins와 Whitney, 1989; Eppinger 등, 1994). 서로 다른 전문성을 보유한 팀들로 프로젝트가 구성되는 만큼 굉장히 많은 활동들을 거쳐 제품 개발이 이루어지게 되고(Browning과 Ramasesh, 2007) 제품 디자인 정보(design information)에 대한 의사결정 결과를 프로젝트 내 활동들끼리 서로 주고 받으면서 연관 관계가 형성되며(Chua와 Hossain, 2011) 제품의 크기가 커지고 복잡해질수록 활동 간 연관 관계도 더 복잡해지고 의사결정이 어려워진다(Ulrich 등, 2008).

다양한 기술, 전문성을 가진 자원들이 참여하는 프로젝트의 스케줄을 정하는 다중 기술 자원 프로젝트 스케줄링 문제(Multi-skilled resource-constrained project scheduling problem, MSRCPSP)는 자원 제약을 고려한 프로젝트 스케줄링 문제(Resource-constrained project scheduling problem, RCPSP)를 확장시킨 것으로 최근 많은 주목을 받으며 활발히 연구가 진행되어 왔다(Afshar-Nadjafi, 2021). MSRCPSP 문제는 제품 개발을 비롯하여 생산 스케줄링, R&D, 건축공학 등 여러 분야에서 활용되고 있고 프로젝트의 기간을 최대한 줄이기 위한 목적으로 각 활동의 일정, 그리고 각 활동에 할당되어야 하는 자원을 결정하는 다수의 연구가 진행되었다.

하지만 자원이 안정적으로 확보되지 않는다면 필요한 순간에 필요한 자원이 부재하여 계획한대로 프로젝트를 정상적으로 수행할 수 없게 된다. 직원 이직, 기계 장비 정비 등의 이유로 프로젝트 진행 중에 자원 확보가 불확실해질 수 있고, 최근 COVID-19로 인한 팬데믹 선언 이후 갑작스러운 직원의 부재, 휴가는 굉장히 빈번하게 발생할 수 있는 현상이 되었다. 단순히 프로젝트 구성원 중 일부가 사라지는 것으로 인한 불확실성을 넘어 인원 변동 없이 모든 구성원이 그대로 존재하더라도 작업에서 요구하는 기술의 부재, 미숙함으로 인해 불확실성이 발생할 수 있다. 특히 소프트웨어 산업에서는 새로운 기술들이 매우 빠른 속도로 개발되고 있고, 변화하는 고객의 요구에 따라 새로운 기술들을 반영하여 작업을 해야 하지만 해당 기술을 완벽하게 탑재한 인력이 존재하는 경우는 드물다(Otero 등, 2009).

자원 확보의 어려움으로 작업 수행에 차질이 생기게 되면 다른 작업과의 소통 과정에서 미완성의 정보가 전달되거나 필요한 정보를 제공하는데 오래 걸릴 수 있고, 결과적으로 프로젝트 마감에 지연될 수 있다(Chua와 Hossain, 2011). 빠른 개발 속도가 중요한 제품 개발 프로젝트에서 프로젝트 기간이 길어지는 것은 치명적이고

이에 따라 자원 확보의 불확실성이 프로젝트 기간에 미치는 영향에 대한 분석, 그리고 프로젝트 관리자로 하여금 이에 더 빠르고 효과적으로 대응할 수 있는 프로젝트 스케줄링 방법을 고안해야 할 필요성이 부각된다. 자원 확보 불확실성을 비롯하여 예기치 못한 변경에 대처할 수 있는 방법으로 불확실성을 포함하여 강건하게 프로젝트 일정을 계획하는 proactive scheduling, 그리고 일어난 변경 사항에 맞춰서 프로젝트 일정을 수정하는 reactive scheduling이 있다(Herroelen과 Leus, 2005).

하지만 불확실성에 대비한다고 해도 언제든지 예상하지 못한 일이 일어나 프로젝트가 실패할 위험으로 이어질 수 있기 때문에 proactive scheduling과 함께 reactive scheduling이 같이 이루어져야 하고, 본 연구에서는 후자의 방식에 기반해서 인적 자원 재배치 전략을 개발하고자 한다. 연구에서 제안하는 방법론은 갑자기 발생한 자원 공백에 대응하는 전략으로 동적 재스케줄링에 해당한다. 한편, 프로젝트의 활동에 요구되는 기술을 정확히 보유하고 있는 자원을 구하는 것은 매우 어렵다(Otero 등, 2009). 또한 공백이 발생한 인력과 동일한 기술을 보유한 인력을 새롭게 채용하여 필요한 기술을 습득하도록 교육하는 것이 비용 및 시간 제약으로 인해 불가능할 수 있다. 이를 고려해 본 연구는 기존 인력을 활용하여 프로젝트 지연을 최소화하고자 한다. 자원 확보의 불확실성이 프로젝트 기간에 미치는 영향을 분석함으로써 프로젝트 관리자로 하여금 인력 공백에 더 빠르고 효과적으로 대응할 수 있는 프로젝트 재스케줄링 휴리스틱을 제시하고자 한다.

본 논문은 총 5개 장으로 구성된다. 제 2장은 다중 기술 자원 프로젝트 스케줄링(MSRCPSP), 스케줄 변경 관리 및 프로젝트 일정 재조정(rescheduling)에 관한 기존 문헌을 검토한다. 제 3장은 실질 지연 기반 재스케줄링 휴리스틱에 대해 설명한다. 적합도와 실질 지연에 대한 정의와 함께 각각을 계산할 수 있는 방법을

다루고, 앞선 2개의 지표가 어떻게 활용되어 재스케줄링 전략이 결정되는지를 제시한다. 4장에서는 가상 예제를 통해 본 연구에서 제안한 방법론을 토대로 재스케줄링 전략에 따른 프로젝트 지연 시간을 구하고, 기존의 대체 자원 선정 규칙을 적용했을 때 나오는 결과와 비교한다. 이 과정에서 본 연구의 전략이 가지는 효과성을 입증하고, 마지막 5장은 결론 및 추후 연구 방향에 대해 정리하며 논문을 마무리한다.

## 제 2 장 선행연구

### 2.1 다중 기술 자원 프로젝트 스케줄링

한정된 시간과 자원을 바탕으로 프로젝트 기간의 최소화를 목적으로 하여 프로젝트를 구성하는 작업들의 일정과 해당 작업에 필요한 자원들을 할당하는 문제는 자원 제약 프로젝트 스케줄링 문제 (Resource-constrained project scheduling problem, RCPSP)로 정의되고, 프로젝트 수행을 위해 다양한 기술을 보유한 자원들을 고려하는 다중 기술 자원 프로젝트 스케줄링 문제 (MSRCPSP)는 지난 20년 동안 집중적으로 연구가 이루어져 왔다(Afshar-Nadjafi, 2021). 소프트웨어 개발 분야에서 개발자들이 프로그래밍, 데이터 분석, 디버깅 등의 기술을 보유한 채로 개발 작업을 수행하는 것에서 비롯된 MSRCPSP 문제(Bellenguez-Morineau와 Néron, 2007) 하에서는 하나 이상의 기술을 수행할 수 있는 자원들 중 특정 작업에서 요구하는 기술들을 충족하는 자원들이 할당되고, 자원 제약에 더불어 기술 제약까지 프로젝트 스케줄링에 포함되기 때문에 복잡한 MSRCPSP 문제를 더 효과적으로 풀 수 있는 다양한 방법론들이 개발되어 왔다.

Bellenguez-Morineau와 Néron(2007)은 특정 기술의 조합을 요구하는 작업과 다중 기술을 보유한 자원으로 구성된 프로젝트의 기간을 최소화하기 위해 branch-and-bound 알고리즘을 활용하였고, 3개의 서로 다른 전략으로 노드를 구축하고 탐색 트리 가지치기를 위한 2가지 하한선을 제안하였다. 더 많은 기술을 보유한 자원을 활용하기 위해서는 그만큼 더 많은 비용이 발생하기 때문에 Li와 Womer(2009)는 프로젝트

비용 최소화를 목적으로 MILP(mixed integer linear programming) 모델과 CP(constraint programming) 모델을 결합한 Hybrid Benders Decomposition(HBD) 알고리즘을 개발하여 MSRCPSPP 문제를 해결하였다. Correia와 Saldanha-da-Gama(2014)는 비선형 목적함수를 포함하는 수학적 모델을 개발하여 자원 활용을 위해 투입되는 고정 비용과 함께 프로젝트 종료 시점에 따라 결정되는 가변 비용이 MSRCPSPP에 미치는 효과를 분석하였다.

자원마다 보유한 기술의 수준이 다를 수 있고, 기술 수준에 따라 작업 수행 시간이 상이할 수 있기 때문에 기술에 따른 자원의 효율성을 고려하여 프로젝트 기간을 줄일 수 있는 방법들에 대한 연구가 이루어져 왔다(Bellenguez-Morineau와 Néron, 2005; Heimerl과 Kolisch, 2010). Bellenguez-Morineau와 Néron(2005)은 자원이 가지는 기술의 계층적 수준을 다루었고, 비슷한 방식으로 Heimerl과 Kolishch(2010)는 자원의 이질적 효율성으로 표현하여 서로 다른 기술과 효율성을 가진 자원들을 할당하고 프로젝트 일정을 계획하는 문제를 풀기 위해 MILP 모델을 활용하였다. 인건비 최소화를 목적으로 한 앞선 연구와 달리 Firat과 Hurkens(2012)는 매일마다 수행되는 작업의 수를 최대화하는 것을 목적으로 하여 연구를 수행하였다. 더불어 Snauwaert와 Vanhoucke(2022)는 기술의 계층적 수준이 효율성과 비용, 품질의 관점에서 MSRCPSPP에 어떻게 영향을 주는지 분석하였다.

## 2.2 스케줄 변경 관리 연구

앞서 언급한 연구들로부터 제안된 모델과 알고리즘을 토대로 다중 기술을 보유한 자원이 참여하는 프로젝트의 일정을 계획할 수 있지만, 프로젝트를 수행하는 도중 여러 가지 원인에 의해 변경 사항이 발생할 수 있다. 대표적으로, 직원의 휴가 및 기계

고장으로 인한 정비와 같은 이유로 작업을 수행하기 위한 자원을 제때 활용할 수 없는 문제가 발생할 수 있지만(Buddhakulsomsiri와 Kim, 2007), 대부분의 연구들은 프로젝트 기간 동안 활용 가능한 자원이 고정되어 있다는 가정 하에 프로젝트 스케줄링을 위한 알고리즘이나 휴리스틱을 개발하였다(Afshar-Nadafi, 2021). 자원 확보의 불확실성 외에도 새로운 작업의 추가, 또는 기존 작업의 삭제 등으로 인한 작업 변동, 작업 시간의 변동, 작업에 필요한 원자재의 수급 지연(Herroelen과 Leus, 2005), 소프트웨어 개발의 경우 요구사항 변경, 긴급한 버그 수정, 개발 작업 오류(Xiao 등, 2010) 등의 요인들이 초기에 계획된 프로젝트 스케줄의 변경을 야기할 수 있고, 예정된 마감 기한을 넘겨서 프로젝트가 지연될 경우에는 이러한 지연으로 인해 불필요하게 높은 비용이 발생하게 된다(Herroelen과 Leus, 2005).

기존의 결정적 스케줄링 모델(static deterministic scheduling model)이 가지는 타당성에 대해 회의적인 시각이 대두되었고(Goldratt, 1997), 프로젝트 수행 중 마주할 수 있는 예기치 못한 변경에 대비하기 위해 불확실성을 반영한 MSRCPSP 문제와 robust scheduling, dynamic scheduling, proactive scheduling(Goldratt, 1997, Herroelen과 Leus, 2005), 그리고 reactive scheduling에 대한 연구(Herroelen 등, 1999; Herroelen과 Leus, 2005)들이 이루어졌다. Lambrechts 등(2008)은 자원 확보 불확실성을 고려한 MSRCPSP 문제에 대해 프로젝트의 불안정성(instability cost)를 최소화하는 강건한 스케줄을 찾는 것을 목적으로 하였다. 불안정성이란, 작업의 계획된 시작 시간과 실제 작업 시작 시간 간의 절대 편차(absolute deviation)의 가중합을 말하고, 작업에 시간 버퍼나 자원 버퍼를 추가하여 프로젝트 진행 중 발생하는 예기치 못한 변경을 완충하는 방식으로 강건한 스케줄을 찾고자 하였다.

강건한 스케줄을 구축하는 방법은 그림 2.1과 같이 정리할 수 있고, Van de Vonder 등(2006)은 초기 프로젝트 스케줄의 강건성을 최대화하는 것을 목적으로 시간



버퍼를 할당하는 방법에 대해 연구하였으며, 자원 버퍼를 각 작업에 할당하는 방법에 대해 연구한 Lambrechts 등(2011)과 함께 Alfawzan과 Haouari(2005), Abassi 등(2006)은 추가 시간 없이 작업의 float time을 최대로 활용하면서 프로젝트 기간을 최소화하는 동시에 강건성을 최대화할 수 있는 방법을 연구하였다.

기대할 수 있는 불확실성을 반영하고 proactive scheduling을 활용하여 수립한 프로젝트 일정에 변경이 발생한 경우 프로젝트 전반, 또는 변경된 작업 일정 일부에 대해 스케줄링을 다시 수행해야 하며, 초기 계획에서 벗어난 프로젝트 일정을 수정하는 과정을 reactive scheduling으로 정의한다(Deblaere 등, 2011). Reactive scheduling은 변경으로 인해 영향을 받는 작업들을 지연된 시간만큼 뒤로 미루는 right shift rule(Sadeh 등, 1993), 그리고 새롭게 일정을 계획하는 full rescheduling(Artigues와 Roubellat, 2000)의 방식으로 수행할 수 있고, 다음 3절에서 reactive scheduling, 또는 rescheduling을 위한 다양한 방법에 대한 연구들에 대해 소개하고 살펴보고자 한다.

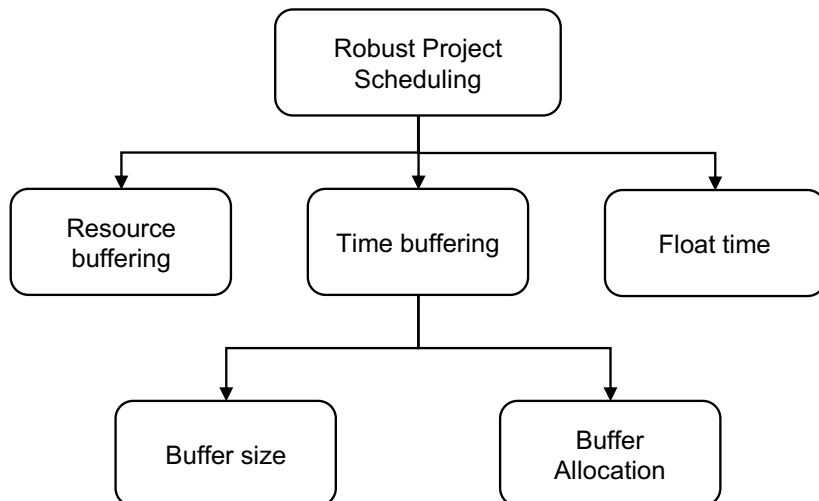


그림 2.1: 강건한 스케줄링을 위한 방법론 (Lambrechts 등, 2011)

## 2.3 프로젝트 일정 재조정 문제

Vieira 등(2003)이 제조 시스템 하에서의 일정 재조정(rescheduling)을 위한 프레임워크를 제시하였고, 표 2.1에서 묘사되듯이 리스케줄링 환경(rescheduling environments), 리스케줄링 전략(rescheduling strategies), 리스케줄링 정책(rescheduling policies), 그리고 리스케줄링 방법(rescheduling methods)이 프레임워크에 포함된다.

리스케줄링 환경이란 스케줄링의 대상이 되는 작업들의 집합을 의미하고, 작업의 수가 유한한 경우에는 정적 환경(static environment), 그 반대의 경우에는 동적 환경(dynamic environment)이 된다. 초기 일정이 생성되는지 여부에 따라 리스케줄링 전략이 결정되고, predictive-reactive rescheduling 전략 하에서 언제, 그리고 어떻게 재조정을 수행할 것인지에 따라 리스케줄링 정책이 결정된다. 마지막으로, 리스케줄링 방법을 통해 불확실성을 반영하여 프로젝트 스케줄을 생성하거나 변경 발생 이후 기존의 스케줄을 수정하거나 최적의 상태로 갱신한다.

프로젝트 일정 재조정 문제는 Vieira 등(2003)의 연구를 비롯해서 생산 계획을 다루는 영역에서 많은 연구가 이루어졌고 이에 따라 생산활동을 위한 기계의 리스케줄링 방법에 대해 많은 논의가 이루어졌다(Vieira 등, 2003; Aytug 등, 2005; Katragjini 등, 2012). 앞선 리스케줄링 프레임워크를 건축 프로젝트에 적용시킨 Liu와 Shih(2009)는 생산성 변동으로 인해 작업 시간이 예정보다 길어지는 경우에 프로젝트 전체의 지연을 최소화하도록 리스케줄링 방법 중 complete regeneration과 partial rescheduling 두 가지 방식을 활용하였고, CP(constraint programming) 모델을 기반으로 한 리스케줄링 방식을 제안하였다. Vonder 등(2007)은 작업 시간이 변동하는 RCPSP 문제에 대해 full rescheduling을 위한 네 가지 휴리스틱 기법을 제안하였고, 작업이 수행될 수 있는 방식이 여러 가지가 존재하는 multi-mode RCPSP(MMRCPPSP)

문제에 대해 자원 확보와 작업 시간 두 가지 측면에서 변동이 발생하는 경우 일정을 수정하기 위해 hybrid mixed integer programming 및 constraint propagation과 tabu search 휴리스틱을 적용시키는 연구도 진행된 바 있다(Zhu 등, 2005; Deblaere 등, 2011). 대부분의 연구들이 리스케줄링 방법 중 full rescheduling/complete regeneration에 집중하여 이루어졌지만, 프로젝트 내의 모든 작업들에 대해 스케줄링을 다시 수행해야 하는 단점을 보완하기 위해 변경에 의해 영향을 받게 되는 일부의 작업들만 수정하여 리스케줄링의 복잡성을 낮추는 partial rescheduling 방법에 대한 연구 또한 진행되어 왔다(ElMekkawy와 ElMaraghy, 2003).

다중 기술을 보유한 자원이 참여하는 프로젝트에 대해 자원 확보에 변동이 발생한 이후 프로젝트 리스케줄링을 하는 방법에 초점을 맞춘 연구들도 존재한다. Ahmadpour와 Ghezavati(2019)는 자원의 부재 여부가 반영된 일정표가 주어진 프로젝트에 대해 다중 기술을 보유한 자원 확보의 불확실성을 고려하여 fuzzy scheduling 모델을 구축하였다. 한편 특정 자원의 공백이 발생했어도 해당 자원이 보유한 기술을 동일하게 가진 다른 자원이 대체 자원으로서 투입되어 작업이 차질 없이 수행될 수 있도록 할 수 있고, 이러한 다중 기술 자원 간 동적 기술 전환(dynamic skill assignment)를 통해 자원 확보 불확실성으로 인한 방해 효과를 완화할 수 있다(Azizi와 Liang, 2013). 이에 Wang 등(2022)은 앞서 나온 동적 기술 전환 방식을 활용하였고, 동일한 기술일 경우 다중 기술 자원끼리 수행해야 하는 작업을 교환할 수 있는 상황에서 자원 확보 불확실성을 내포한 MSRCPSM 문제를 유전자 알고리즘(genetic algorithm)과 자원 할당을 위한 우선순위 전략(priority rule)을 결합한 GA-PR 스케줄링 알고리즘을 통해 해결하고자 하였다.

## 2.4 연구 의의

최근 COVID-19로 인한 팬데믹이 선언된 이후 적지 않은 직원들이 일시적으로 작업을 수행할 수 없는 상태가 되는 것이 빈번해졌고 직원 이직, 기계 장비 정비 등의 이유를 포함하여 자원 확보의 불확실성은 프로젝트 일정을 계획하기 위해 고려해야 하는 중요한 요소가 되었다. 자원 확보의 불확실성이 프로젝트 소요 시간에 미치는 영향을 분석하며 이러한 불확실성에 더 빠르고 효과적으로 대응할 수 있는 프로젝트 스케줄링 방법을 고안해야 하는 프로젝트 관리자의 역할이 더욱 강조되었다. 시간 및 자원 버퍼를 추가하여 불확실성에도 강건한 프로젝트 일정을 계획하는 proactive scheduling 기반의 방법들이 많이 연구되었지만 그럼에도 프로젝트를 수행하면서 일어날 수 있는 모든 가능성을 전부 고려하는 것은 불가능하기 때문에 예측 결과를 벗어난 변화에 대해서는 reactive scheduling 기반의 방법을 통해 일정 계획을 수정해야 한다(Van de Vonder 등, 2007).

자원 이탈로 인해 프로젝트를 수행할 인원이 부족해진 상황에서 프로젝트가 계획된 기간을 넘어 지연되는 것을 막기 위해 자원이 보유한 기술을 바탕으로 이탈 인력을 완벽하게 대체할 수 있는 자원을 탐색하고 선정하는 방법에 대한 연구들이 이루어졌다. 3절에서도 소개하였듯이, 동적 기술 전환 방식을 활용하여 프로젝트를 나간 사람이 보유한 기술 중 일부를 후보 자원이 보유하게 된다면 해당 자원이 인력 공백의 자리에 투입될 수 있고, 프로젝트의 지연을 최소화하는 방향으로 프로젝트 내 자원을 재배치하는 방법이 연구되었다(Azizi와 Liang, 2013). Wang 등(2022)은 외부에서 필요한 자원을 활용하는 것이 가능할 때 외부 인원으로 인한 비용과 함께 프로젝트 지연을 최소화할 수 있는 자원 재배치 방법론을 개발하며 앞선 연구를 발전시켰다.

앞선 연구들은 활동에서 필요로 하는 기술끼리 서로 독립적이고 각 기술마다 요구하는 자원의 수가 결정되는 상황을 전제한다. 활동을 수행하기 위해 필요한 특정 기술을 보유한 자원이라면 누구든 활동에 참여할 수 있고 같은 기술을 보유한 자원끼리 서로의 업무를 대신해줄 수 있다. 그러나 활동에서 필요로 하는 여러 종류의 기술 중 하나 또는 일부만을 보유한 자원이 해당 활동을 완벽하게 작업하는 것은 불가능에 가깝고 서로 다른 기술을 가지고 있는 자원들끼리 작업을 위해 소통하는 과정에서 불필요한 비용이 발생할 수 있다. 다시 말해, 본 연구에서는 활동이 요구하는 기술 집합과 동일하거나 이를 포함하는 기술 집합을 가진 자원들이 활동에 참여하도록 프로젝트 스케줄이 계획되는 것을 가정한다.

이 뿐만 아니라 Wang 등(2022)의 연구에서는 인력 공백이 발생하는 시점을 사전에 알고 있고 이러한 상황을 처음부터 반영하여 계획된 프로젝트 기간에 맞추어 자원을 활동에 할당하고 자원이 활동에 투입되는 시점을 결정하는 방법론을 개발하였다. 프로젝트의 일정을 계획하는 단계에서 인력 공백이 일어날 시간을 정확히 알 수 없고 프로젝트 스케줄링에 반영할 수 없기 때문에 인력 공백이 일어난 시간 이후로 자원 계획을 수정하여 스케줄 변화에 대응할 수 있는 방안을 마련해야 한다. 따라서 이러한 한계점을 보완하기 본 연구에서는 비용을 발생시키는 외부 인력의 충원이라는 옵션은 배제한 채로 인력 공백의 시점 및 해당 시점 이후 인원 부족을 겪는 활동들의 시작 시간마다 프로젝트 내부에서 대체 자원을 선정하여 대상 활동에 임시로 투입하며, 활동이 끝나는 시간에 맞춰 프로젝트 일정을 업데이트한다. 활동의 기술 집합과 자원의 기술 집합이 일치하는 정도를 기준으로 적절한 대체 자원이 선정되고, 완전히 일치하는 자원이 다른 활동을 작업 중인 상태이거나 여유 시간이 부족한 경우 완벽한 대체 자원 대신 그보다 미숙한 사람을 대체 자원으로 활용한다.

Brooks(1995)의 연구에 따라 미숙한 능력의 인력은 학습 곡선과 추가적인 소통에 소요되는 오버헤드(overhead)로 인해 차후에 지연을 발생시키게 된다는 문제점이 존재한다. 하지만 미숙하더라도 다음에 계획된 활동까지 여유 시간이 충분하다면 투입 이후 생산성의 차이로 일어나는 작업 지연을 상쇄시키며 프로젝트가 지연되는 것을 방지할 수 있다.

따라서 본 연구는 모든 자원에 대해 인력 공백 이후 자원-활동 할당 계획을 전부 다시 세우는 것 대신 인력 공백으로 인해 직접적인 영향을 받는 일부 활동에 대해 자원 재배치 고려 시점에서 대기 중인 인적 자원이 대상 활동에 얼마나 적합한 인력인지, 그리고 특정 자원이 대체 자원으로 선정되었을 때 프로젝트 전체의 완료 시간에 미치는 영향을 정량적으로 계산하여, 궁극적으로 프로젝트의 지연 시간을 최소화하는 방향으로 적절한 대체 자원을 탐색하는 프로젝트 재스케줄링 전략을 수립한다.

## 제 3 장 프로젝트 재스케줄링 방법론

본 연구는 여러 기술을 가진 인적 자원들이 참여하는 프로젝트에서 자원의 공백이 발생했을 때 프로젝트의 지연을 최소화할 수 있는 프로젝트 재스케줄링(project rescheduling) 방법론을 개발한다. 프로젝트가 진행 중인 임의의 시점에서 인적 자원 한 명이 휴가, 이직 등의 이유로 프로젝트에서 이탈하였으나, 이를 대체할 수 있는 새로운 인력이 외부에서 공급되지 않는 상황을 전제로 한다. 즉, 기존 자원에서 한 명이 줄어들고 남은 자원만으로 프로젝트를 끝까지 완수해야 하는 문제가 생겼다. 인적 자원이 부족해져 기존에 계획했던 기한보다 프로젝트가 지연될 수 있고 그에 따라 비용이 늘어나기 때문에(Lambrechts 등, 2008), 자원 부족에 따른 프로젝트 지연을 최소화할 수 있도록 기존 스케줄 하에서 이탈한 자원을 임시로 대체할 수 있는 자원을 내부에서 선정하고 인력 공백이 발생한 시점 이후의 활동들의 시작 시간을 업데이트한다.

### 3.1 문제 상황

본 연구는 여러 개의 활동으로 구성된 프로젝트가 진행되고 있으며, 프로젝트의 완료 시점까지 새로운 인력이 외부에서 투입되지 않는 상황을 고려한다. 프로젝트의 활동마다 요구되는 기술이 존재하고, 인적 자원마다 보유하고 있는 기술 수준이 조금씩 다르다. 그림 3.1은 총 7개의 활동들로 구성된 예시 프로젝트에 대해 네트워크 형식으로 표현한 그림이다. 그림 3.1에서 관찰할 수 있듯이, 활동 간에는 선행 관계가

표 3.1: 예시 프로젝트 활동 정보

활동 ( $i$ )	기술 요구 사항 ( $S_i$ )	필요 자원 수 ( $n_i$ )	소요 시간 ( $p_i$ )	후행 활동
1	{0, 0, 0, 0, 0}	0	0	2, 3
2	{0, 0, 0, 0, 1}	4	6	8
3	{1, 1, 0, 0, 0}	3	3	4, 5, 6
4	{0, 0, 1, 0, 0}	5	5	7
5	{0, 1, 1, 0, 0}	6	8	8
6	{0, 0, 0, 1, 1}	2	6	7
7	{0, 0, 1, 0, 0}	5	5	9
8	{1, 1, 0, 0, 0}	3	4	9
9	{0, 0, 0, 0, 0}	0	0	-

존재하며, 선행 활동이 완료된 이후에 후행 활동을 시작한다. 각 활동  $i$ 마다 요구되는 기술이 다르다. 프로젝트에서 요구하는 기술  $k$ 가 총  $K$ 개일 때, 활동  $i$ 에서 필요로 하는 기술 요구 사항(skill requirement)은 집합  $S_i = \{s_{ik} \mid k = 1, 2, \dots, K\}$ 로 나타낼 수 있다. 활동  $i$ 가 기술  $k$ 를 요구한다면  $s_{ik}$ 는 1의 값을, 그렇지 않다면 0의 값을 가진다. 표 3.1은 각 활동 별로 요구되는 기술 요구 사항과 기술 요구 사항을 만족하는 필요 자원의 수( $n_i$ ), 소요 시간( $p_i$ ), 그리고 선행 관계 상 후행 활동들을 나타낸다. 소요

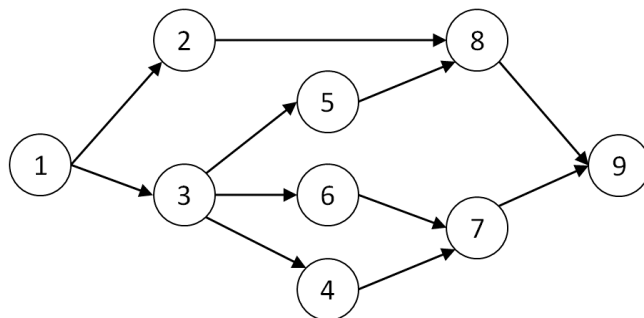


그림 3.1: 예시 프로젝트 네트워크



표 3.2: 예시 프로젝트 인적 자원 정보

자원 ( $j$ )	기술 역량 ( $C_j$ )	자원 ( $j$ )	기술 역량 ( $C_j$ )
1	{1, 1, 1, 0, 0}	7	{0, 1, 1, 1, 0.5}
2	{1, 1, 1, 0.5, 0}	8	{0.5, 1, 1, 1, 0.5}
3	{1, 1, 1, 0, 0.5}	9	{0, 0, 1, 1, 1}
4	{1, 1, 1, 0.5, 0.5}	10	{0.5, 0, 1, 1, 1}
5	{0, 1, 1, 1, 0}	11	{0, 0.5, 1, 1, 1}
6	{0.5, 1, 1, 1, 0}	12	{0.5, 0.5, 1, 1, 1}

시간은 활동에 배정된 자원들이 기술 요구 사항을 충분히 만족하는 경우 활동에 걸리는 시간이다. 활동 1과 9는 가상 활동(dummy activity)으로 프로젝트의 시작과 종료 활동을 의미하며 작업하는데 필요하는 인적 자원의 수가 0이고 소요 시간 또한 0이다.

네트워크 상의 모든 활동이 종료되어야 프로젝트가 종료되고, 프로젝트에 참여하는 총  $J$ 명의 인적 자원이 보유하고 있는 기술 역량(skill capability)은 집합  $C_j = \{c_{jk} \mid k = 1, 2, \dots, K\}$ 로 표현한다. 그림 3.1의 예시 프로젝트에는 총 12명의 자원이 참여하며, 각 자원이 프로젝트에서 요구하는 5가지 기술에 대하여 보유하고 있는 기술 역량은 표 3.2와 같다.  $c_{jk}$ 는 각 자원이 보유하고 있는 기술의 수준을 나타내는 값으로, 자원  $j$ 가 보유하고 있지 않은 기술에 대해서는 0의 값을 가지지만 특정 기술  $k$ 를 보유한 경우에는 자원  $j$ 의 기술 수준에 따라  $c_{jk}$ 의 값이 결정된다. 기술  $k$ 에 대한 전문적인 지식을 가지고 있으며 이에 따라 기술 수준이 충분히 높을 때  $c_{jk}$ 는 1의 값을 가지고, 기술  $k$ 에 대한 지식이 상대적으로 부족하여 기술 수준이 높지 않을 때에는 0.5의 값을 가진다.

표 3.3: 예시 프로젝트의 일정 계획

활동 ( $i$ )	참여 자원 ( $R_i$ )	시작 시간 ( $t_i^s$ )	종료 시간 ( $t_i^f$ )	후행 활동
1	-	0	0	2, 3
2	9, 10, 11, 12	0	6	8
3	1, 2, 3	0	3	4, 5, 6
4	8, 9, 10, 11, 12	6	11	7
5	2, 3, 4, 5, 6, 7	3	11	8
6	9, 10	11	17	7
7	1, 2, 3, 4, 5	17	22	9
8	1, 2, 3	11	15	9
9	-	22	0	-

프로젝트 일정 계획은 Correia 등(2012)에서 다루어진 다중 기술 자원 프로젝트 스케줄링 기법을 바탕으로 계획된다. 기존에 계획된 프로젝트 상에서 활동 별 참여 자원들은 해당 활동에 필요한 기술 수준이 충분히 높다. 즉, 프로젝트에 인력 공백이 발생하기 전의 최초 일정 계획은 프로젝트 활동에 필요한 인적 자원의 기술 역량과 전체 인적 자원의 수가 최적화된 상태이다. 프로젝트 일정 계획은 표 3.3과 같이 활동 별 참여 자원( $R_i$ ), 시작 시간( $t_i^s$ ) 및 종료 시간( $t_i^f$ )로 나타낼 수 있다. 예를 들어, 활동 3과 8은 자원 1, 2, 그리고 3이 수행한다. 프로젝트는  $t = 0$ 일 때 활동 2와 3부터 시작해서,  $t = 22$ 에 활동 7을 끝으로 종료한다.

이와 같이 프로젝트 일정이 계획되어 있을 때, 임의의 시점  $t$ 에 한 명의 자원( $j$ )이 나가 프로젝트의 종료 시점까지 돌아오지 않는 경우 자원  $j$ 가 작업하고 있던 활동  $i$  뿐만 아니라 프로젝트 일정 계획 상 자원  $j$ 가 배정되어 있는 후속 활동에도 자원 제약이 발생하게 된다.

표 3.3과 같이 최적화되어 있던 프로젝트 계획에 추가적인 제약이 발생한 경우에 대해 본 연구에서는 공백이 발생한 활동에 임시로 자원을 투입하는 재스케줄링 방법론을 개발하였다. 즉, 기존 일정 계획에서 활동 별로 배정되어 있던 참여 자원은 수정하지 않은 상황에서 인력 공백이 발생한 활동들에 대해 대체 자원을 선정하고, 선행 활동이 끝나는 시간에 따라 후행하는 활동들의 시작 시간을 수정함으로써 전체 프로젝트 지연을 최소화하고자 한다.

### 3.2 실질 지연 기반 재스케줄링 휴리스틱

실질 지연 기반 재스케줄링 방법론은 그림 3.2와 같이 시점  $t$ 에 자원  $j$ 가 프로젝트에서 나간 경우, 인력에 공백이 생긴 활동들을 대상으로 유휴 자원들을 파악하여 그 중 하나를 대체 자원으로 선정하고 후속 활동들의 일정 계획을 업데이트하는 방법이다. 자원  $j$ 의 역량과 일치하지 않는 역량의 유휴 자원이 대체 자원으로 선정된다면 인력 공백이 일어난 활동  $i$ 를 작업하는데 소요되는 시간이 예정보다 길어지며 활동  $i$ 의 종료 시간이 계획된  $t_i^f$ 보다 지연될 수 있다. 이 때, 유휴 자원의 투입 이후 인력 공백 활동  $i$ 의 종료 시간에 의해 영향을 받는 (i) 연관 활동들을 파악하고, (ii) 각 연관 활동에 발생하는 실질 지연을 계산하여 이 중 최대 실질 지연의 크기가 가장 작은 유휴 자원을 대체 자원으로 선정하는 휴리스틱을 적용한다. 다시 말해, 활동  $i$ 의 종료 시간에 따라 최악의 경우 시작 시간이 지연되는 정도를 최소화할 수 있는 대체 자원을 선정하게 된다. 인력 공백 시점  $t$ 는 인력 공백이 발생한 시간을 포함하여 프로젝트를 나간 자원( $j$ )이 배정되어 있던 후속 활동들의 시작 시간( $t_{i \in A(j)}^s$ )을 포함한다. 후속 활동들의 시작 시간은 일정이 재조정되면서 업데이트됨에 주의한다. 인력 공백 활동 집합  $A(j)$ 는 시점  $t$ 에 자원  $j$ 가

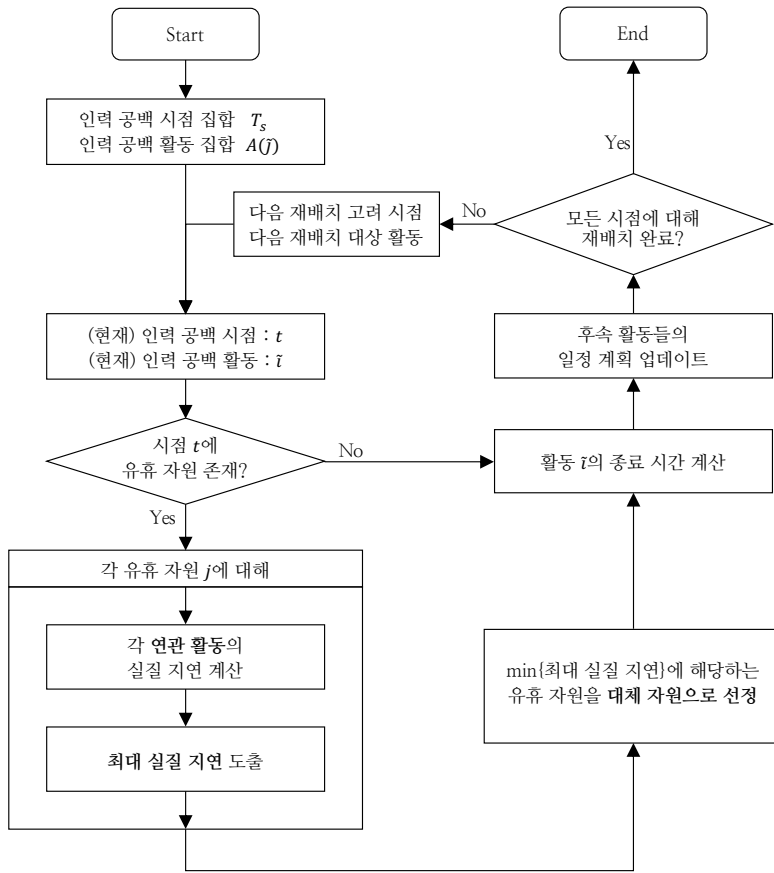


그림 3.2: 실질 지연 기반의 재스케줄링 휴리스틱 흐름도

수행하고 있던 활동  $i$ 와 이후에 계획되어 있던 활동들을 포함하는 집합이다.

본 논문에서 제안하는 실질 지연 기반 재스케줄링 방법론은 아래와 같은 내용을 가정한다.

- I. 프로젝트의 일정 계획에서 각 활동에 배정된 참여 자원은 유지된다.
- II. 프로젝트를 나간 자원은 프로젝트가 종료할 때까지 돌아오지 않고, 새로운 자원의 충원은 고려하지 않는다.

III. 기존에 대체 자원이 수행하기로 계획되어 있던 활동들은, 해당 자원이 대체 활동을 마치고 돌아온 시점에 시작한다.

IV. 두 개의 활동이 서로 연결되어 있는 경우, 선행 활동이 끝난 이후에 후행 활동을 시작한다.

### 3.2.1 연관 활동

가정 IV에 따라 인력 공백이 일어난 활동  $i$ 로부터 정보를 받아야 하거나, 앞선 활동에서 활용한 자원 중 일부가 필요한 활동들은 선행하는 활동이 끝난 이후에 작업을 시작하게 된다. 이처럼 인력 공백 활동  $i$ 와 연결되어 있어 활동  $i$ 에 의해 일정 계획이 영향 받는 후행 활동들을 ‘연관 활동’이라고 정의하고, 본 연구에서는 선행 관계 및 자원 공유 관계를 통틀어 *연관 관계*라고 한다. 시점  $t$ 에 어느 활동에도 참여하지 않고 대기 중인 유휴 자원이 인력 공백 활동에 투입된 후 활동  $i$ 의 종료 시간이 연관 활동의 계획된 시작 시간을 넘어 지연되면 그만큼 연관 활동  $i$ 의 시작 시간  $t_i^s$ 가 늦어진다.

한편, 유휴 자원은 기존 프로젝트 일정 계획 상에서 시점  $t$  이후에 수행해야 하는 활동들이 정해져 있다. 유휴 자원  $j$ 가 대체 자원으로 선정되어 시점  $t$ 부터 활동  $i$ 를 수행하게 된다면 가정 III에 의해 기존에 작업하기로 계획된 후속 활동 중 시작 시간이 가장 이른 활동인 ‘다음 활동’  $i_{next}(j, t)$ 는 선행 활동인  $i$ 가 끝날 때까지 시작할 수 없다. 인적 자원의 이동으로 인해 활동  $i$ 와 다음 활동  $i_{next}(j, t)$  사이에 새로운 자원 공유 관계가 형성되는 경우에 다음 활동  $i_{next}(j, t)$ 도 활동  $i$ 의 연관

활동에 포함된다. 다음 활동  $i_{next}(j, t)$ 는 다음의 식 (3.1)과 같이 나타낼 수 있다.  $A(j)$ 는 시점  $t$  이후 자원  $j$ 가 수행해야 하는 활동들을 포함하는 집합이다.

$$i_{next}(j, t) = \underset{i}{\operatorname{argmin}}\{t_i^s\}, \quad (3.1)$$

where  $i \in A(j)$

예를 들어, 예시 프로젝트에서  $t = 2$ 에 자원 10이 프로젝트를 나가는 경우를 생각해 보자. 이 경우, 활동 2에 즉각적으로 인력 공백이 발생한다. 그림 3.1과 표 3.3의 일정 계획을 참고하면 활동 2의 종료 시간에 영향을 받는 연관 활동들은 활동 4(자원 공유 관계)와 8(선행 관계)이고,  $t = 2$ 에 대기 중인 유휴 자원은 자원 4, 5, 6, 7과 8이다. 각각의 유휴 자원이 수행해야 하는 다음 활동  $i_{next}$ 를 살펴 보면 자원 4, 5, 6, 그리고 7은 활동 5를, 자원 8은 활동 4를 하기로 계획되어 있다. 활동 4는 활동 2의 연관 활동인 반면 활동 5는 활동 2와 선행 관계로 이어져 있지도, 자원을 공유하지 않는다. 따라서 자원 4, 5, 6과 7 중 한 명이 선택되어 활동 2에 투입된다면 새로운 자원 공유 관계의 형성으로 인해 활동 5가 활동 2의 연관 활동에 추가되고, 활동 4와 8에 더불어 활동 5의 시작 시간 또한 활동 2의 종료 시간에 의해 결정된다.

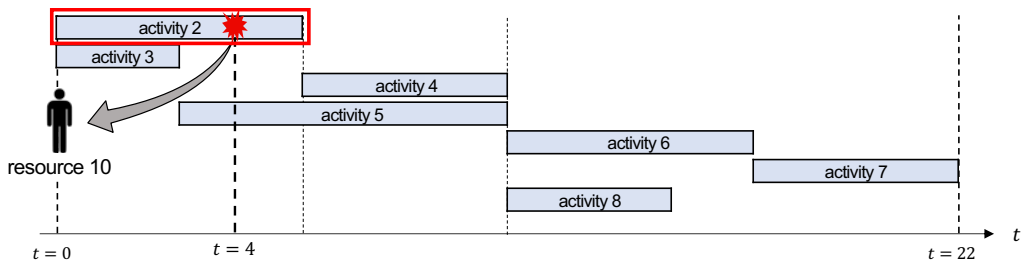
### 3.2.2 연관 활동의 실질 지연 도출

앞서 설명한 것처럼 인력 공백 활동  $i$ 가 끝난 이후에 연관 활동들을 시작할 수 있고, 대체 자원으로 선정된 유휴 자원의 역량에 따라 활동  $i$ 의 작업에 소요되는 시간이 정해진다. 따라서 재스케줄링 이후 프로젝트의 총 지연 시간은 유휴 자원이 대체 자원으로써 투입된 활동  $i$ 의 소요 시간이 지연되는 정도와 활동  $i$ 의 연관 활동이

지연되는 정도를 비교하여 예측할 수 있다.

기존의 프로젝트 일정 계획은 각 활동을 수행하는 자원들 모두 상응하는 활동에서 요구하는 기술들을 모두 전문적인 수준으로 가지고 있는 상태이다. 만약 대체 자원이 인력 공백 활동  $i$ 에 필요한 인적 자원의 기술 역량을 충분히 보유하고 있지 않다면 해당 자원은 활동  $i$ 에 남아 있는 다른 자원에 비해 생산성이 떨어지며, 생산성의 차이로 인해 활동  $i$ 가 실제 종료되는 데까지 걸리는 시간은 프로젝트 활동 정보에 나타난 소요 시간보다 더 길어진다.

예시 프로젝트에서  $t = 4$ 에 자원 10이 프로젝트를 나가는 경우 활동 2는 표 3.3과 같이  $t = 0$ 에 작업을 시작하여  $t = 6$ 까지 수행해야 완료되는, 활동의 작업 도중 인력 공백이 발생한다. 인력 공백이 일어난 활동 2에는 자원 9, 11, 그리고 12가 잔류하게 되고 표 3.4의 프로젝트 일정 계획에 따라  $t = 4$ 에 대기 중인 유휴 자원이 1과 8임을



Activity	Skill requirement					Resource requirement	Shortage amount
	1	2	3	4	5		
1	-	-	-	-	-	0	-
<b>2</b>	<b>0</b>	<b>0</b>	<b>0</b>	<b>0</b>	<b>0</b>	<b>4</b>	<b>1</b>
...	...	...	...	...	...	...	...
8	1	1	0	0	0	3	0
9	-	-	-	-	-	0	-

Resource pool	Skill capability					Status
	1	2	3	4	5	
<b>1</b>	<b>1</b>	<b>1</b>	<b>1</b>	<b>0</b>	<b>0</b>	<b>idle</b>
<b>8</b>	<b>0.5</b>	<b>1</b>	<b>1</b>	<b>1</b>	<b>0.5</b>	<b>idle</b>
2	1	1	1	0.5	0	busy
3	1	1	1	0	0.5	busy
...	...	...	...	...	...	...
12	0.5	0.5	1	1	1	busy

그림 3.3: 인력 공백 활동 및 유휴 자원 파악

알 수 있다. 그림 3.3에서 확인할 수 있는 것처럼 활동 2는 총 5개의 기술 중 마지막 기술이 작업에 필요하고 표 3.2에 따라 자원 9, 10, 11과 12가 모두 5번 기술에 대해 전문적인 수준의 기술을 보유하고 있다. 반면 그림 3.3에 따라 자원 1은 5번 기술을 보유하지 않고 자원 8은 5번 기술에 대한 기술 수준이 0.5로, 두 유휴 자원 모두 활동 2의 기술 요구 사항을 모두 완벽하게 만족하지 못하는 자원이라고 할 수 있다. 즉, 프로젝트를 이탈한 자원 10에 비해 활동 2에서 요구하는 기술 역량을 갖추지 못한 자원 1 또는 8이 대체 자원으로 선정되었을 때 활동 2가 종료될 것으로 예상되는 시간은 기존에 계획된 종료 시간  $t_2^f = 6$ 보다 늦어져 지연이 발생한다.

활동의 기술 요구 사항과 자원이 가진 기술 역량이 일치하는 정도를 토대로 자원이 활동을 수행하기에 얼마나 숙련된 사람인지를 판단할 수 있고, 더 숙련된 자원일수록 활동에 더 적합하다는 것을 의미한다. 활동  $i$ 에 대한 자원  $j$ 의 적합도(suitability)  $f_{ij}$ 는 활동  $i$ 의 기술 요구 사항  $S_i$ 와 자원  $j$ 의 기술 역량  $C_j$ 를 바탕으로 수식 (3.2)과 같이 계산할 수 있다(Otero 등, 2009).

$$f_{ij} = \sum_{k=1}^K s_{ik} \cdot c_{jk}, \quad (3.2)$$

$$s_{ik} \in S_i, c_{jk} \in C_j$$

각 활동마다 자원 적합도의 최댓값은 활동이 필요로 하는 기술 종류 개수의 합이 되고, 이 값보다 작은 적합도를 가진 자원은 해당 활동을 수행하기에 미숙한, 부분적으로 숙련된(partially skilled) 자원이라고 할 수 있다. 앞선 예시에서 이탈한 자원 10과 활동 2를 수행해야 하는 잔류 자원인 9, 11, 그리고 12는 적합도가 1로 최댓값을 가지는 한편, 유휴 자원 1과 8의 적합도는 각각 0과 0.5이고 둘 모두 자원



10에 비해 부분적으로 숙련된 자원이다. 이처럼 인력 공백 활동  $i \in A(j)$ 에 대해 최대 적합도  $f_{i,j}$ 를 가진 자원  $j$ 가 나간 공백을  $f_{i,j}$ 의 적합도를 가진 자원  $j$ 가 대신하게 된다면, 활동  $i$ 는 최대 적합도  $f_{i,j}$ 의 자원  $(n_i - 1)$ 명과 적합도  $f_{i,j}$ 의 자원 1명이 함께 작업하게 된다.

부분적으로 숙련된 대체 자원  $j$ 의 투입으로 서로 다른 적합도의 인력들이 협업하게 된다면 적합도에서 비롯된 생산성의 차이로 인해 예정보다 활동의 소요 시간이 길어지게 되고, 이 때 활동  $i$ 에 대해 기존 종료 시점( $t_i^f$ )에 비해 늘어난 시간인 지연 시간  $d_i$ 는 다음의 식 (3.3)에 따라 구할 수 있다.

$$d_i(j, \tilde{j}, t) = \frac{\left(1 - \frac{f_{i,j}}{f_{i,\tilde{j}}}\right)(t_i^f - t)}{n_i - 1 + \frac{f_{i,j}}{f_{i,\tilde{j}}}} \quad (3.3)$$

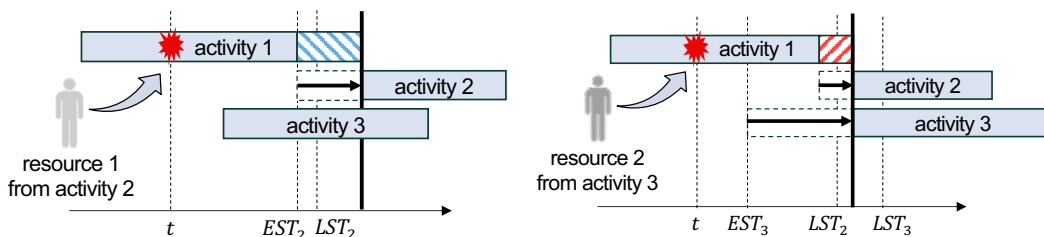
$(t_i^f - t)$ 는 활동  $i$ 에 대해서 작업이 이루어지지 않고 남은 기간을 말한다. 즉, 시점  $t$ 로부터 활동의 계획된 종료 시간  $t_i^f$ 까지의 기간을 의미한다. 대체 자원의 투입 이후 활동  $i$ 의 종료 시간인  $t_i^f$ 는  $(t_i^f + d_i(j, \tilde{j}, t))$ 로 업데이트된다.

프로젝트를 나간 자원  $j$ 의 적합도와 동일한 적합도를 가진 유휴 자원이 투입된다면 식 (3.2)에서 분자가 0이 되어 활동에 지연이 발생하지 않으며, 부분적으로 숙련된 인력이 투입되었을 때는 적합도와 지연 시간이 서로 반비례한다. 다시 말해, 유휴 자원의 적합도가 작을수록 활동  $i$ 에 대해 지연 시간  $d_i$ 의 크기는 점점 커진다. 적합도가 0의 값을 가질 때 지연 시간의 크기가 가장 크고, 이 때의 지연 시간은  $\frac{t_i^f - t}{n_i - 1}$ 으로 활동  $i$ 가 기존의 계획보다 한 명 줄어든 상태로 작업을 수행했을 때 발생하는 지연 시간과 동일하다. 즉 적합도가 0인 대체 자원을 투입하더라도 활동  $i$ 를

한 명 부족한 상태로 작업하는 것과 일치하며, 이러한 이유로 작업에 도움을 주지 못하는 적합도 0의 유휴 자원은 대체 자원으로 고려하지 않는다. 만약 시점  $t$ 에 유휴 자원이 전혀 없거나 유휴 자원들의 적합도가 모두 0이라면 인력 공백 활동  $i$ 는 대체 자원 없이, 인원이 부족한 상태로 활동을 수행하게 된다.

한편, 프로젝트를 구성하는 각 활동마다 계획된 시작 시간(earliest start time, EST)와 더불어 프로젝트의 완료 시점을 지연시키지 않는 선에서 가장 늦게 시작할 수 있는 시간인 latest start time(LST)가 존재한다(Ahuja 등, 1994). 후진 계산법(backward pass processing)을 통해 각 활동의 LST를 계산할 수 있고 대체 자원이 투입된 활동의 종료 시간이 연관 활동의 LST보다 이르면 후행 활동의 시작 시간이 미뤄지더라도 프로젝트에 지장을 전혀 주지 않는다는 것을 의미한다.

예를 들어, 그림 3.4에서 묘사된 것처럼 시점  $t$ 에 인력 공백이 발생한다면 해당 시점에서 대기 중인 유휴 자원 1과 2 모두 활동 1에 대한 적합도가 0보다 큰 대체 자원 후보가 된다. 자원 1은 다음 활동 2를, 자원 2는 다음 활동 3을 수행하기로 계획되어 있으며, 프로젝트 일정 계획 상 활동 2만 활동 1의 연관 활동에 해당한다. 그림 3.4의 (b)처럼 자원 2를 대체 자원으로 선정한 경우에는 활동 3도 활동 1의 연관 활동에 새롭게 추가된다. 활동 1에서 요구하는 적합도에 비해 대체 자원이 가진 낮은



(a) 후보 자원 1의 다음 활동 : 활동 2      (b) 후보 자원 2의 다음 활동 : 활동 3

그림 3.4: 대체 자원 후보들의 다음 활동 및 연관 활동 분석

적합도로 인해 인력 공백 활동 1이 지연되고, (a)와 (b) 모두 활동 1의 종료 시간이 활동 2의 LST를 초과하기 때문에 2가지 경우 모두 활동 2의 시작 시간이 LST를 초과한 부분만큼 프로젝트 전체가 지연된다. 한편 그림 3.4의 (b)를 보면 새로운 연관 활동인 활동 3도 활동 1의 지연으로 시작 시간이 뒤로 밀리지만, 밀린 시점이 활동 3의 LST 전인 것을 확인할 수 있고, 따라서 활동 3으로부터 프로젝트의 지연이 발생하지 않는다.

선행 활동의 종료 시간이 후행하는 연관 활동의 LST를 초과한 부분만큼 프로젝트 전체의 지연으로 이어지고, 이를 반영하여 본 연구에서는 연관 활동에 발생하는 ‘실질 지연’을 다음의 식 (3.4)과 같이 정의한다.

$$\begin{aligned}
 ND_i(\tilde{i}, j, \tilde{j}, t) &= (t_i^f + d_i(j, \tilde{j}, t)) - LST_i, \\
 \text{where } i &\in Succ(\tilde{i}), \\
 \tilde{i} &\in A(\tilde{j})
 \end{aligned}
 \tag{3.4}$$

프로젝트의 지연에 기여하는 연관 활동  $i$ 의 실질 지연  $ND_i$ 는 인력 공백 활동  $\tilde{i}$ , 유휴 자원  $j$ , 프로젝트 이탈 자원  $\tilde{j}$ , 그리고 시점  $t$ 가 주어졌을 때, 활동  $i$ 가 종료될 것으로 예상되는 시간  $(t_i^f + d_i(j, \tilde{j}, t))$ 과 연관 활동  $i$ 의  $LST_i$ 의 차이로 정의된다. 실질 지연  $ND_i$ 가 0보다 큰 값을 가지게 될 경우 실질 지연의 크기만큼 프로젝트가 지연되지만, 0 이하의 값에 대해서는 프로젝트에 지장을 주지 않는다. 활동  $\tilde{i}$ 의 연관 활동 집합  $Succ(\tilde{i})$ 에 포함된 모든 활동에 대해 실질 지연을 구할 수 있고, 대체 자원으로 선정된 유휴 자원의 적합도에 따라 활동  $\tilde{i}$ 의 종료 시간이 결정되며, 결과적으로 연관 활동에 발생하는 실질 지연의 크기가 달라진다.

### 3.2.3 실질 지연을 활용한 대체 자원 선정 및 일정 업데이트

그림 3.2에서 묘사한 것처럼, 본 연구에서 제안하는 재스케줄링 휴리스틱은 자원 이탈이 발생한 시간을 기점으로 시점을 옮겨 가며 모든 인력 공백이 생긴 활동들을 대상으로 적합도가 0보다 큰 유휴 자원 중 한 명을 선정하는 과정을 반복한다. 집합  $T_s$ 의 각 인력 공백 시점  $t$ 마다 선정된 유휴 자원의 적합도에 따른 연관 활동의 실질 지연 값이 쌓여 최종적으로 모든 활동의 수행 이후 프로젝트 전체가 지연된 시간이 결정된다. 프로젝트의 지연을 최소화하기 위해서는 각 시점  $t$ 에서 인력 공백 활동  $i$ 의 지연 시간  $d_i$ 를 줄이는 것과 함께 연관 활동  $i$ 에 발생하는 실질 지연  $ND_i$ 를 최소화할 수 있는 유휴 자원을 대체 자원으로 선정해야 한다. 앞서 언급했듯이, 본 연구의 재스케줄링 휴리스틱은 최소최대 알고리즘(minmax algorithm)을 적용하여 유휴 자원을 투입했을 때 연관 활동의 실질 지연의 최댓값을 도출한 이후, 도출된 최대 실질 지연의 값이 가장 작은 경우에 해당하는 자원을 대체 자원으로 선정한다. 그림 3.2에 나타난 휴리스틱의 흐름을 구체적으로 표현하면 다음 그림 3.5와 같이 정리할 수 있다.

대체 자원 선정이 고려되는 시점  $t$ 에 대기 중인 유휴 자원이 존재하는지 먼저 판별한다. 유휴 자원이 없거나, 존재하더라도 모든 유휴 자원의 적합도가 0인 경우에는 대체 자원 없이 부족한 인원 대로 인력 공백 활동  $i$ 의 작업이 이루어지고, 이 때 발생하는 활동  $i$ 의 지연 시간은  $d_i = \frac{t_i^f - t}{n_i - 1}$ 의 방식으로 계산되며, 대체 자원이 없기 때문에 실질 지연은 해당 시점에서 고려되지 않는다. 반면, 적합도가 0보다 큰 유휴 자원이 한 명 이상 존재한다면 연관 활동의 실질 지연 값을 기준으로 적절한 대체 자원을 고른다.

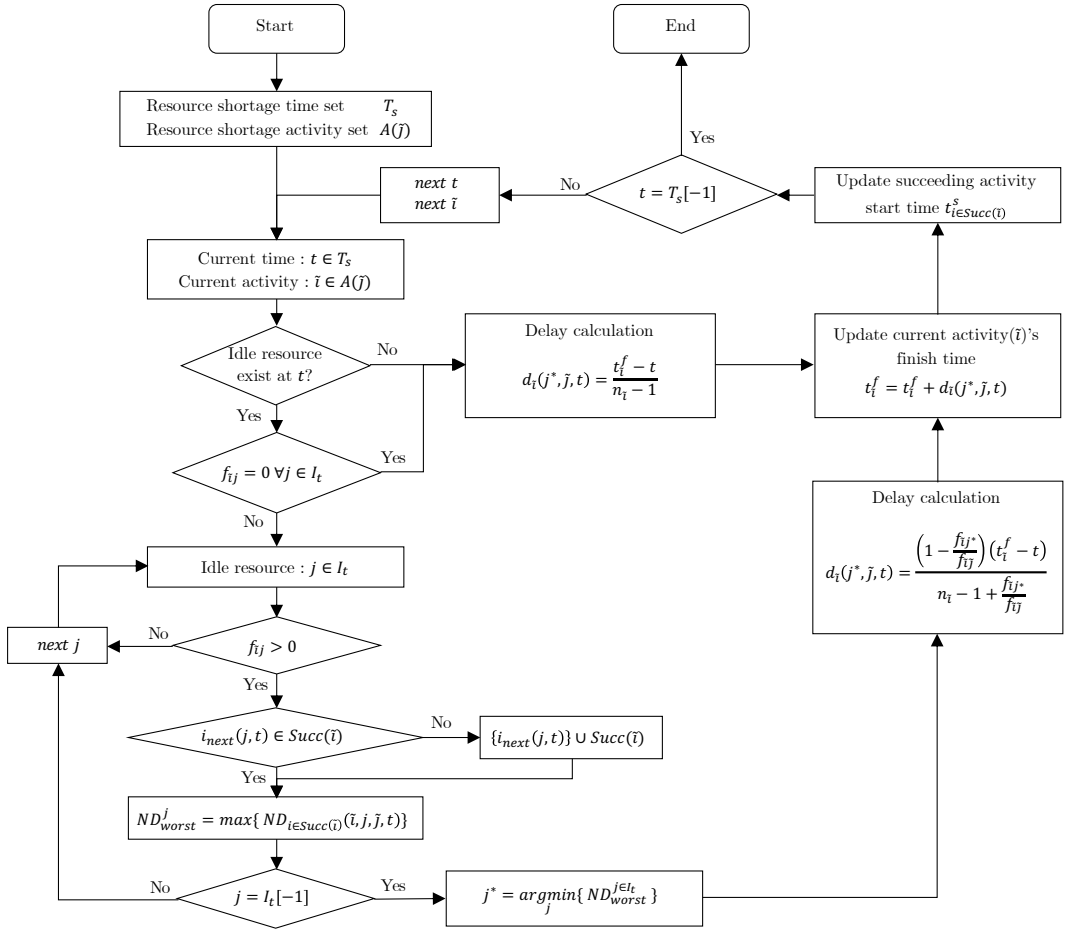


그림 3.5: 실질 지연 기반의 재스케줄링 휴리스틱 세부 흐름도

집합  $I_t$ 는 시점  $t$ 에서 대체 자원 후보가 되는 유휴 자원으로 구성된 집합을 말하고, 활동  $\bar{i}$ 에 대한 적합도가 0보다 큰 자원들만을 대상으로 연관 활동의 실질 지연을 계산한다. 대체 자원 후보 집합  $I_t$  내 유휴 자원  $j$ 의 다음 활동  $i_{next}(j, t)$ 까지 활동  $\bar{i}$ 의 연관 활동에 포함되고, 모든 연관 활동에 대해 유휴 자원  $j$ 를 활동  $\bar{i}$ 에 투입했을 때 발생하는 실질 지연을 계산한다. 다음의 식 (3.5)와 같이 정의되는  $ND_{worst}^j$ 는 연관 활동의 실질 지연 중 최댓값을 나타내고, 유휴 자원  $j$ 로 인해 프로젝트 전체가 가장 늦어지는 최악의 상황을 의미한다.

$$ND_{worst}^j = \max \{ND_{i \in Succ(i)}(\tilde{i}, j, \tilde{j}, t)\} \quad (3.5)$$

where  $j \in I_t$

위의 식 (3.5)를 활용하여 모든 유휴 자원  $j$ 에 대해 최대 실질 지연  $ND_{worst}^j$ 를 구할 수 있다. 최대 실질 지연의 값을 가장 최소화하는 유휴 자원이 선정되어 인력 공백 활동  $i$ 에 투입되어야 하고, 따라서 최종적으로 선정되는 유휴 자원  $j^*$ 는 다음의 식 (3.6)에 따라 도출된다.

$$j^* = \underset{j}{\operatorname{argmin}} \{ ND_{worst}^{j \in I_t} \} \quad (3.6)$$

대체 자원  $j^*$ 의 적합도에 따라 인력 공백 활동  $i$ 의 지연 시간  $d_i(j^*, \tilde{j}, t)$ 는 식 (3.3)에 따라 결정되고, 이에 따라 활동  $i$ 의 종료 시간  $t_i^f$ 는  $(t_i^f + d_i(j^*, \tilde{j}, t))$ 로 업데이트된다. 대체 자원  $j^*$ 로 인해 활동  $i$ 와 연결된 연관 활동들의 시작 시간이 활동  $i$ 의 업데이트된 종료 시간보다 이른 경우,  $t_{i \in Succ(i)}^s$ 는 활동  $i$ 의 실제 종료 시간으로 수정된다.

$T_s$  내 모든 인력 시점  $t$  및  $A(j)$  내 모든 인력 공백 활동에 대해 위와 같은 방식으로 유휴 자원 중 적합한 대체 자원을 선정하고, 지연 시간을 계산하여 프로젝트의 스케줄을 업데이트하면 프로젝트는 종료하며 재스케줄링 휴리스틱을 마친다.

## 제 4 장 실험 및 분석

본 연구에서 제안하는 재스케줄링 휴리스틱을 2개의 프로젝트 상황에 적용하여 실질 지연에 기반한 자원 재배치 전략이 가장 효과적으로 프로젝트 지연을 줄일 수 있음을 보인다. 휴리스틱의 효과를 실험하기 위해 조건이 서로 다른 2가지 예제를 구성하였다.

### 4.1 실험 데이터

본 연구에서 제시하는 재스케줄링 휴리스틱에 대해 실험하고 결과를 분석하기 위해 Kolisch와 Sprecher(1997)가 고안한 PSPLIB(Project Scheduling Problem Library, <https://www.om-db.wi.tum.de/psplib/>) 정보를 활용하였다. 해당 라이브러리에는 다양한 유형의 자원 제약 프로젝트 스케줄링 문제(RCPSP)를 모델링하고 해결하는 데 사용할 수 있는 벤치마크 문제들이 포함되어 있다. 다양한 표준 문제들이 있지만 그 중에서 활동의 개수가 30개인 J30 문제를 대상으로 하였고, 문제에 주어진 정보들을 토대로 생성한 프로젝트 일정에 대해 실험을 진행하였다. 기존에 제시된 PSPLIB의 문제들은 참여하는 자원들의 기술 역량, 그리고 활동마다 요구하는 기술 요구 사항에 대해서 다루지 않는다. 따라서 본 연구의 휴리스틱을 실험에 적용할 수 있도록 문제에서 주어진 정보들에 다음과 같은 내용을 추가하여 프로젝트를 생성하였다.

[1] 프로젝트의 진행에 필요한 기술의 종류는 프로젝트에 참여하는 자원의 종류에 대응된다. 즉, 4가지 종류의 자원들의 조합으로 작업이 이루어지는 활동들은 마찬가지로 4가지 종류의 기술에 대한 요구사항을 가진다. 예를 들어, 어떤 활동은 4가지 중 2번 기술을 제외한 나머지 3가지 기술만을 필요로 하고, 이 경우 해당 활동의 기술 요구사항은 (1, 0, 1, 1)과 같다. 각 활동에 참여하는 자원의 종류에 맞게 기술 요구사항이 결정된다.

[2] 프로젝트에는 총 4가지 종류의 자원이 참여하게 되고, 각 종류의 자원마다 특정한 기술을 공통적으로 보유하게 된다. 예를 들어, 첫 번째 유형의 자원들은 모두 1번 기술을, 두 번째 유형의 자원들은 2번 기술을 모두 공통적으로 보유한다. 기본적으로 보유하게 되는 기술 외에 나머지 종류의 기술들에 대해서는 기술의 보유 여부 및 기술의 수준이 임의로 정해진다. 3장에서 설명한 것처럼 나머지 기술들에 대한 기술 수준( $c_{jk}$ )은  $\{0, 0.5, 1\}$  중 하나의 값이 할당된다.

J30 문제에는 30개의 활동에 대해 각 활동 간 선행 관계 및 활동에 대한 자원 제약이 사전에 정의되어 있고, 선행 관계의 구성 그리고 자원 제약이 설정된 방식에 따라 여러 가지 유형의 문제들이 포함되어 있다. 활동 간 선행 관계 구성이 다르고 프로젝트에 참여하는 자원의 수도 다른 2개의 문제를 선정하여 각각에 대해 프로젝트 일정을 생성하였고, 서로 다르게 구성되어 있는 프로젝트들에 대해서도 본 연구에서 제시하는 재스케줄링 휴리스틱이 지연 시간을 효과적으로 단축시킬 수 있는지를 검증하고자 하였다.



표 4.1: 프로젝트 1 활동 정보

활동 ( $i$ )	기술 요구 사항 ( $S_i$ )	필요 자원 수 ( $n_i$ )	소요 시간 ( $d_i$ )	후행 활동
1	$S_1 = \{0, 0, 0, 0\}$	0	0	2, 3, 4
2	$S_2 = \{1, 0, 1, 1\}$	9	9	6, 8, 10
3	$S_3 = \{0, 1, 1, 0\}$	9	9	12, 20, 29
4	$S_4 = \{1, 1, 1, 0\}$	2	2	5, 10, 23
5	$S_5 = \{1, 1, 1, 1\}$	5	5	7, 9, 18
6	$S_6 = \{1, 1, 0, 1\}$	9	9	7, 13, 15
7	$S_7 = \{1, 1, 1, 1\}$	5	5	14, 22, 26
8	$S_8 = \{0, 1, 1, 1\}$	10	10	9, 11, 13
9	$S_9 = \{1, 1, 1, 1\}$	9	9	17, 26, 27
10	$S_{10} = \{1, 1, 0, 1\}$	1	1	12, 16, 19
11	$S_{11} = \{0, 1, 1, 1\}$	2	2	15, 20, 27
12	$S_{12} = \{1, 1, 0, 1\}$	9	9	14, 17
13	$S_{13} = \{1, 1, 0, 1\}$	6	6	17, 19, 21
14	$S_{14} = \{1, 0, 0, 0\}$	5	5	24, 30
15	$S_{15} = \{1, 1, 1, 1\}$	4	4	18, 19, 23
16	$S_{16} = \{1, 0, 1, 1\}$	2	2	21, 27, 31
17	$S_{17} = \{1, 1, 0, 1\}$	10	10	22, 30
18	$S_{18} = \{1, 1, 1, 1\}$	5	5	26, 28, 31
19	$S_{19} = \{1, 0, 1, 1\}$	9	9	22, 25, 28
20	$S_{20} = \{0, 1, 1, 1\}$	2	2	21, 23
21	$S_{21} = \{1, 1, 1, 0\}$	4	4	25
22	$S_{22} = \{1, 1, 1, 1\}$	7	7	24
23	$S_{23} = \{1, 1, 1, 0\}$	3	3	24, 25
24	$S_{24} = \{1, 1, 1, 1\}$	4	4	31
25	$S_{25} = \{0, 0, 1, 1\}$	2	2	30
26	$S_{26} = \{1, 1, 0, 0\}$	9	9	29
27	$S_{27} = \{1, 1, 0, 1\}$	6	6	28
28	$S_{28} = \{1, 0, 1, 1\}$	9	9	29
29	$S_{29} = \{1, 1, 0, 1\}$	8	8	32
30	$S_{30} = \{0, 1, 1, 0\}$	8	8	32
31	$S_{31} = \{1, 0, 1, 1\}$	10	10	32
32	$S_{32} = \{0, 0, 0, 0\}$	0	0	-

예를 들어, 표 4.1과 표 4.2는 은 본 연구에서 실험을 위해 생성한 프로젝트 1의

표 4.2: 프로젝트 1 인적 자원 정보

자원 (j)	기술 역량 (C <sub>j</sub> )	자원 (j)	기술 역량 (C <sub>j</sub> )	자원 (j)	기술 역량 (C <sub>j</sub> )
1	C <sub>1</sub> = {1, 0.5, 1, 0.5}	20	C <sub>20</sub> = {1, 1, 0, 0}	39	C <sub>39</sub> = {0, 0, 1, 0}
2	C <sub>2</sub> = {1, 0.5, 1, 1}	21	C <sub>21</sub> = {0, 1, 0.5, 1}	40	C <sub>40</sub> = {0.5, 0.5, 1, 0}
3	C <sub>3</sub> = {1, 0, 0, 1}	22	C <sub>22</sub> = {0.5, 1, 0, 0}	41	C <sub>41</sub> = {0, 0.5, 1, 0.5}
4	C <sub>4</sub> = {1, 0.5, 1, 1}	23	C <sub>23</sub> = {0, 1, 0, 1}	42	C <sub>42</sub> = {0.5, 1, 1, 0}
5	C <sub>5</sub> = {1, 0, 0, 0.5}	24	C <sub>24</sub> = {0, 1, 0.5, 1}	43	C <sub>43</sub> = {0.5, 0.5, 0, 1}
6	C <sub>6</sub> = {1, 0.5, 0, 0}	25	C <sub>25</sub> = {0.5, 1, 1, 0.5}	44	C <sub>44</sub> = {0, 0.5, 1, 1}
7	C <sub>7</sub> = {1, 1, 1, 1}	26	C <sub>26</sub> = {0, 1, 0.5, 1}	45	C <sub>45</sub> = {1, 0.5, 1, 1}
8	C <sub>8</sub> = {1, 0, 1, 0.5}	27	C <sub>27</sub> = {0.5, 1, 1, 0}	46	C <sub>46</sub> = {0.5, 0.5, 1, 1}
9	C <sub>9</sub> = {1, 0.5, 1, 1}	28	C <sub>28</sub> = {1, 1, 0.5, 0}	47	C <sub>47</sub> = {1, 0, 0, 1}
10	C <sub>10</sub> = {1, 1, 1, 0}	29	C <sub>29</sub> = {0, 1, 1, 0}	48	C <sub>48</sub> = {1, 0, 0.5, 1}
11	C <sub>11</sub> = {1, 1, 0, 1}	30	C <sub>30</sub> = {0, 0.5, 1, 0}	49	C <sub>49</sub> = {0, 0, 1, 1}
12	C <sub>12</sub> = {1, 0, 0, 0}	31	C <sub>31</sub> = {0.5, 0, 1, 1}	50	C <sub>50</sub> = {0.5, 0, 0.5, 1}
13	C <sub>13</sub> = {1, 0, 0, 1}	32	C <sub>32</sub> = {0, 0, 1, 0}	51	C <sub>51</sub> = {1, 1, 1, 1}
14	C <sub>14</sub> = {1, 0, 0.5, 0.5}	33	C <sub>33</sub> = {1, 0, 1, 1}	52	C <sub>52</sub> = {1, 0, 0, 1}
15	C <sub>15</sub> = {1, 0.5, 1, 0}	34	C <sub>34</sub> = {0, 0, 1, 1}	53	C <sub>53</sub> = {1, 0, 0.5, 1}
16	C <sub>16</sub> = {0, 1, 0.5, 1}	35	C <sub>35</sub> = {0, 1, 1, 0}	54	C <sub>54</sub> = {0.5, 1, 0, 1}
17	C <sub>17</sub> = {0.5, 1, 1, 0.5}	36	C <sub>36</sub> = {0, 0, 1, 1}	55	C <sub>55</sub> = {0, 1, 0.5, 1}
18	C <sub>18</sub> = {0.5, 1, 0, 0.5}	37	C <sub>37</sub> = {1, 0.5, 1, 1}	-	-
19	C <sub>19</sub> = {1, 1, 0, 1}	38	C <sub>38</sub> = {0.5, 0, 1, 1}	-	-

활동에 대한 정보 및 인적 자원 풀에 대한 정보를 나타낸다. 표 4.1에는 프로젝트를 구성하는 활동 별로 요구되는 기술과 자원의 수, 활동의 작업 소요 시간, 그리고 선행 관계에 의해 연결되어 있는 후행 활동들의 정보가 포함되어 있다. 가상 활동에 해당하는 활동 1과 32의 정보까지 같이 포함되어 있고, 시작과 종료를 의미하는 각 활동들은 표에 나와 있는 것처럼 필요 자원 수와 소요 시간이 모두 0이며, 어떠한 기술도 요구하지 않는다. 해당 프로젝트에는 총 55명의 인적 자원이 참여하고 각각의

인적 자원들은 표 4.2에 따라 서로 다른 기술 역량을 가지게 된다.

프로젝트 활동 정보와 인적 자원 정보, 이 두 가지 정보를 바탕으로 프로젝트의 일정 계획이 수립된다. 활동 간 선행 관계가 더 복잡하게 구성되어 있으며 프로젝트 자원이 더 많은 프로젝트 1의 경우 활동 32를 마지막으로 총 121일 만에 프로젝트가 마무리되고, 상대적으로 덜 복잡한 프로젝트 2의 경우 프로젝트 완료까지 총 75일이 소요된다. 두 개의 프로젝트 모두 각 활동을 수행하는 자원들이 상응하는 활동에 대한 적합도가 최댓값을 가지도록 배치되어 있는 상태이고, 프로젝트 2의 활동 및 인적 자원에 대한 정보, 그리고 모든 프로젝트의 일정이 계획된 결과는 [부록]에서 확인할 수 있다.

## 4.2 실험 방법

인력 공백으로 인해 스케줄 변동이 일어난 상황에서 프로젝트가 최소한으로 지연되도록 하기 위해서는 빈 자리를 대체할 인적 자원을 선정하는 기준이 필요하고, 본 연구에서 제안하는 재스케줄링 방법론의 효과와 비교하기 위해 다음 표 4.3에 정리된 대체 자원 선정 규칙들을 활용하였다. 본 논문에서 다루는 자원 재배치 방법론과 마찬가지로, 매 재배치 고려 시점마다 다음에서 설명하는 각각의 규칙들을 통해 유휴 자원 중 재배치 대상 활동에 투입할 대체 자원을 선정하고, 유휴 자원이 없는 경우에는 자원 재배치를 수행하지 않는다.

[1] Maximum suitability :

재배치 고려 시점에 대기 중인 유휴 자원 중 적합도의 값이 클수록 우선순위가 더 높은 규칙이다.

[2] Largest idle time :

유티 시간(idle time)은 재배치 고려 시점으로부터 유티 자원의 다음 활동이 계획대로 시작해야 하는 시간까지의 기간으로, 다음의 식 (4.1)을 통해 구할 수 있다. 유티 시간( $idle_j$ )이 긴 유티 자원일수록 우선순위가 더 높은 규칙이다.

$$\begin{aligned} idle_j &= t_{i_{next}(j,t)}^s - t, \\ \text{where } i_{next}(j,t) &= \underset{i}{\operatorname{argmin}}\{t_i^s\}, i \in A(j) \end{aligned} \quad (4.1)$$

[3] Largest reserve time :

활용 가능 시간(reserve time)은 재배치 고려 시점으로부터 다음 활동이 가장 늦게 시작할 수 있는 시간 LST까지의 기간을 말하고, 프로젝트를 지연시키지 않도록 유티 자원을 최대한 활용할 수 있는 기간에 해당된다. 활용 가능 시간( $\delta_j$ )이 긴 유티 자원일수록 우선순위가 더 높은 규칙이다.

$$\begin{aligned} \delta_j &= LST_{i_{next}(j,t)} - t, \\ \text{where } i_{next}(j,t) &= \underset{i}{\operatorname{argmin}}\{t_i^s\}, i \in A(j) \end{aligned} \quad (4.2)$$

[4] Least number of involved activities :

재배치 고려 시점 이후에 시작하는 활동들 중 재배치를 위해 선정된 유티 자원이 투입되는 활동의 개수가 적을수록 우선순위가 더 높은 규칙이다.

표 4.3: 비교 검증을 위해 활용한 대체 자원 선정 규칙

<i>Rule</i>	<i>Activities sorted by</i>
Maximum suitability	<i>Non-increasing order</i>
Largest idle time	<i>Non-increasing order</i>
Largest reserve time	<i>Non-increasing order</i>
Least number of involved activities	<i>Non-decreasing order</i>

인력 공백에 대처하기 위한 방안으로 위에서 설명한 4가지 대체 자원 선정 규칙에 비해 본 연구를 통해 제시하는 규칙으로 유휴 자원 중 대체 자원을 선정했을 때 발생하는 프로젝트의 지연이 가장 작다는 것을 보이고자 한다. 이를 위해 생성된 프로젝트 일정 계획에 대해서 정해진 시점에 작업 중인 인적 자원 한 명이 프로젝트를 이탈한 상황을 가정하였고 본 연구의 규칙을 포함한 5가지 규칙을 적용했을 때 선정된 대체 자원을 재배치한 이후 프로젝트의 완료 시점을 서로 비교하였다. 2개의 프로젝트 모두 착수 시점인  $t = 0$  부터 계획 마감일까지 10일 주기로 인력 공백 시점을 변경하며 실험을 진행하였고, 각 시점에서 작업 중인 모든 인적 자원에 대해 한 명씩 이탈 인원으로 지정한 후, 각 이탈 인원에게 대해 프로젝트 완료 시점까지 해당 인원을 대체할 수 있는 자원을 선정하고 후행 활동들의 일정 계획을 수정하며 프로젝트의 실제 완료 시점을 도출하였다. 그 결과, 프로젝트 1의 경우 총 309번, 프로젝트 2의 경우 총 250번의 실험을 진행하였고 다음 절에서 실험 후 도출된 결과에 대한 분석을 진행한다.

### 4.3 실험 결과 분석

적용한 대체 자원 선정 규칙 별로 프로젝트에 발생한 지연 시간 결과를 정리하면 표 4.4와 같다. 비교 검증을 위해 활용한 규칙들은 표 4.3에 기재된 순서대로 [1] ~ [4]로 표기하였고, 본 연구에서 제시한 대체 자원 선정 규칙은 [5]로 표기하였다.

표 4.4에 따르면 2개의 프로젝트 각각 309번, 250번 실험을 수행한 결과 대체 자원을 활용하지 않았을 때보다 규칙에 상관없이 대체 자원을 선정하여 임시로 공백에 투입한 경우 평균적으로 프로젝트의 완료 시점을 더 단축시키는 것을 관찰할 수 있다. 인력 공백의 상황에서 아무런 조치를 취하지 않은 경우와 비교했을 때 프로젝트 1의 경우 [1] ~ [5]의 모든 규칙에 대해 309개의 결과 중 프로젝트 총 소요 시간이 더 적게 도출된 비율이 98.7% 이상이고 대체 자원을 선정하기 위해 적용한 규칙에 따라 평균적으로 최소 0.234일부터 최대 0.343일만큼 프로젝트의 완료 시점이 앞당겨졌다. 프로젝트 2의 경우에는 [1] ~ [5]의 모든 규칙에 대해 250개의 결과 중 프로젝트 총 소요 시간이 감소한 비율이 97.3% 이상이며, 대체 자원을 선정하기 위해 적용한 규칙에 따라 평균적으로 최소 0.759일부터 최대 0.902일만큼 프로젝트의 완료 시점이 앞당겨졌다.

대체 자원 선정을 위해 적용한 규칙에 따라 프로젝트 지연의 개선 효과가 다르게 나타나고, 이 중 본 연구에서 제안하는 실질 지연 기반의 대체 자원 선정 전략을 활용하여 프로젝트 재스케줄링을 수행했을 때 인력 공백으로 인한 지연 시간이 프로젝트 1과 2 모두에 대해 가장 적게 발생하는 동시에 대체 자원을 고려하지 않는 경우 대비 다른 규칙들에 비해 지연 시간을 최대로 줄이게 된다. 본 연구의 전략을 바탕으로 재스케줄링 휴리스틱을 적용했을 때 프로젝트 1은 평균적으로 약 138.444일에 종료하였고, 사전에 계획된 시간인 121일과 비교했을 때 약 17.444일의

표 4.4: 대체 자원 선정 규칙에 따른 프로젝트 지연 결과 비교

프로젝트 (계획 완료 기간(일))	대체 자원 X	대체 자원 선정 규칙				
		[1]	[2]	[3]	[4]	[5]
<b>프로젝트 1</b>						
(121)						
완료 시간 평균 (일)	138.787	138.505	138.553	138.553	138.524	138.444
[지연 시간 (일)]	[17.787]	[17.505]	[17.553]	[17.553]	[17.524]	[17.444]
‘대체 자원 X’ 대비 시간 차이 평균 (일)	-	0.282	0.234	0.234	0.263	0.343
‘대체 자원 X’ 대비 시간 초과 비율 (%)	-	98.7	98.7	98.7	98.7	100
[5]의 결과 대비 시간 초과 비율 (%)	-	90.9	90.9	91.6	90.9	-
<b>프로젝트 2</b>						
(75)						
완료 시간 평균 (일)	83.896	83.004	83.123	83.137	83.123	82.995
[지연 시간 (일)]	[8.896]	[8.004]	[8.123]	[8.137]	[8.123]	[7.995]
‘대체 자원 X’ 대비 시간 차이 평균 (일)	-	0.892	0.773	0.759	0.773	0.902
‘대체 자원 X’ 대비 시간 초과 비율 (%)	-	97.3	97.9	97.9	97.9	98.4
[5]의 결과 대비 시간 초과 비율 (%)	-	93.0	92.5	93.6	92.5	-

지연 시간이 발생하였다. 프로젝트 2는 평균적으로 약 82.995일에 종료하였으며 사전에 계획된 시간인 75일보다 약 7.995일만큼 지연되었다. 이는 규칙 [1] ~ [4]를 적용했을 때 도출되는 가장 이른 평균 종료 시간인 138.505일(프로젝트 1),

83.004일(프로젝트 2)에 비해 프로젝트의 종료 시간이 각각 0.061일, 0.009일만큼 더 빠르고, 대체 자원을 고려하지 않은 상황에서 나타나는 평균 종료 시간인 138.787일(프로젝트 1), 83.896일(프로젝트 2)보다 각각 0.343일, 0.902일 앞당긴 본 연구의 규칙 [5]는 최대 0.282일, 0.892일 앞당긴 규칙 [1] ~ [4]보다 더욱 개선된 결과를 도출한다. 프로젝트 1과 2 각각 90.9%, 92.5% 이상 규칙 [1] ~ [4]에 비해 규칙 [5]의 적용 이후 지연 시간이 단축된 실험 결과를 확인할 수 있고, 이를 통해 본 연구에서 제시하는 대체 자원 선정 규칙이 인력 공백으로 인한 프로젝트의 지연을 최소화하는데 가장 우월하다는 결론을 내릴 수 있다.

프로젝트 1과 2 모두 공통적으로 본 연구에서 제시한 규칙을 제외하면 유휴 자원의 적합도를 기준으로 최대 적합도를 보유한 자원을 우선적으로 인력 공백의 자리에 배치하는 규칙 [1](maximum suitability)을 적용했을 때 프로젝트의 지연 시간이 가장 작게 발생한다. 이러한 결과가 관찰되는 이유는 다른 규칙에서 활용한 기준에 비해 규칙 [1]에서 활용하는 기준인 적합도는 인력 공백이 발생한 활동의 지연을 결정짓는 가장 직접적인 요인이기 때문이다. 다시 말해, 가장 높은 적합도를 보유한 대체 자원을 통해 인력 공백이 발생한 활동이 계획된 종료 시간보다 최소한으로 지연되도록 할 수 있고, 이로 인해 후행하는 활동들의 시작 시간을 최소로 변동시키며 궁극적으로 계획에 최대한 가깝게 프로젝트를 완료할 수 있다.

추가로, 표 4.4에 나타나는 실험 결과를 통해 프로젝트 1이 프로젝트 2에 비해 인력 공백으로 인한 지연 시간이 전반적으로 더 큰 것을 관찰할 수 있다. 프로젝트 1의 경우 별다른 조치 없이 인력 공백으로 인한 지연 시간이 17.787일이고 본 연구의 규칙을 기반으로 재스케줄링을 수행했을 때 지연 시간이 17.444일까지 개선된다. 프로젝트 2의 경우 인력 공백으로 인한 지연 시간인 8.896일을 본 연구의 규칙을



바탕으로 7.995일까지 개선시킬 수 있다. 프로젝트 1은 계획된 완료 시점인 121일에 비해 평균 지연 시간이 14.4%부터 14.7%까지 나타나는 데 반해, 프로젝트 2는 계획 완료 시점은 75일에 비해 10.7%부터 11.9%의 지연이 평균적으로 발생한다. 즉, 프로젝트에 참여하는 인원이 많아지고 프로젝트 내 활동들 간 선행 관계가 복잡해질수록 지연이 증가하게 된다. 프로젝트 2는 프로젝트 1과 비교했을 때 상대적으로 프로젝트 참여 인원 대비 각 활동에서 요구하는 인원의 수가 적고, 활동 간 선행 관계가 복잡하게 얽혀 있지 않다. 따라서 복잡하게 연결되어 있지 않는 활동 간 선행 관계에 따라 인력 공백에 대처하기 위한 대체 자원 없이 부족한 인원으로 활동을 수행하더라도 이 때 발생한 지연이 다른 활동으로 쉽게 전파되지 않고 프로젝트 일정 변동을 적게 발생시킨다. 나아가 임의의 대체 자원 선정 규칙을 활용하더라도 인력 공백에 대응해야 하는 시점에 다음 활동까지 대기 중인 유휴 자원의 수가 프로젝트 2가 상대적으로 더 많고 인력 공백의 자리에 대체 자원으로써 임시로 투입할 수 있는 적정 인력을 탐색하는 것이 더 용이하여 재스케줄링 휴리스틱에 기인한 프로젝트 지연의 개선 효과가 프로젝트 1과 비교했을 때 더 두드러지게 나타난다.

## 제 5 장 결론 및 추후 연구 방향

### 5.1 결론

본 논문에서는 프로젝트에 인력 공백이 발생하여 계획한 대로 작업이 진행되기 어려운 상황에서 프로젝트 내부에 남아 있는 인력을 인력 공백 활동에 임시로 배치하여 프로젝트가 지체되는 것을 막는 방법에 대해 연구하였고 적합한 대체 인력을 선정하는 방법론을 개발하였다.

프로젝트를 나간 사람과 동일한 기술과 능력을 보유한 인력이 존재할 경우 해당 인력을 투입하여 공백이 발생한 활동의 작업이 지연되지 않도록 할 수 있지만 만약 그러한 인력이 이미 다른 활동에 투입되어 작업 중이거나 다음 활동까지 대기 중인 상태라고 해도 대기 시간이 충분하지 못한 경우 인적 자원 이동으로 인해 오히려 작업하기로 예정된 다음 활동에 지장을 줄 수 있다. 다시 말해, 다음 활동까지 대기 시간이 충분한 유휴 자원 중 프로젝트를 이탈한 사람의 능력에 가장 가까운 인력을 대체 자원으로 선정하는 동시에 대체 자원의 투입 이후 대상 활동에 발생한 지연이 프로젝트 전체의 지연에 기여하는 정도로 정의된 ‘실질 지연’을 최소화한다. 즉, 대체 자원이 투입된 활동과 연결된 연관 활동이 가장 늦게 시작할 수 있는 시간을 넘어 전자의 활동에 지연이 발생하게 되면(실질 지연의 크기가 0보다 커진다면) 프로젝트의 전체의 지연으로 이어지기 때문에 실질 지연을 줄일 수 있는 자원을 선정해야 한다. 인적 자원이 보유한 기술 역량과 활동에서 요구하는 기술을 바탕으로 특정 유휴 자원이 대상 활동에 대해 적합한 정도인 적합도를 구할 수 있고, 적합도를 통해 대체

자원의 투입 이후 활동에 발생하는 지연 시간 및 실질 지연을 계산하였다. 본 연구에서는 프로젝트 진행 중 인력 공백에 대처해야 하는 시점마다 유휴 자원 별로 대체 자원이 투입된 활동의 지연으로 인한 실질 지연의 최댓값을 구한 이후 이 값을 최소로 만드는 인적 자원을 최종적인 대체 자원으로 선정하는 전략을 제안하였으며, 이것은 대체 자원으로 인해 프로젝트 전체에 발생할 수 있는 최악의 상황을 가장 줄일 수 있는 사람이 프로젝트가 지연되는 것을 막을 수 있는 적절한 인력이라는 것을 의미한다.

PSPLIB를 기반으로 한 가상의 프로젝트 일정에 대해 실질 지연 기반의 대체 자원 선정 규칙을 적용하여 재스케줄링 휴리스틱을 수행하였고, 4장에서 서술한 다른 4가지의 규칙과 비교했을 때 본 연구의 방법론이 지연 최소화에 더 효과적이라는 것을 입증하였다. 이에 더해, 활동에서 필요로 하는 기술 중 일부를 자원이 보유하고 있는 여부만을 판별하여 프로젝트 이탈 인원을 완벽하게 대체할 수 있는 자원을 찾는 방법에 대해 고민한 기존의 연구들과 달리 본 연구를 통해 각 인적 자원 및 활동의 기술 조합으로 자원과 활동 간 적합도를 계산하였고 대체 자원이 투입되는 활동으로 인해 연관 관계에 있는 활동, 나아가 프로젝트 전체에 미치는 영향을 정량적으로 측정하였으며, 이를 바탕으로 프로젝트의 완료 시점을 최대한 앞당길 수 있는 휴리스틱을 고안했다는 점에서 차별성을 갖는다.

## 5.2 한계점 및 추후 연구 방향

본 연구에서 변경 영향 기반의 효과적인 자원 재배치 방법론을 개발하였지만 다음의 한계점을 가진다.

첫째, 방법론을 제시하기에 앞서 재배치 후보로 특정 시점에 작업 중이지 않고

다음 작업까지 대기 중인 유휴 자원을 고려하였고, 이 중 하나의 자원이 선택되어 재배치가 이루어진 후에는 해당 활동이 완전히 끝날 때까지 원래 투입이 계획되었던 다음 작업이 기다려야 한다고 가정하였다. 계획된 시간에 작업을 시작할 수 없는 대체 자원의 다음 작업 또한 다른 인력의 공백으로 인해 영향을 받고, 자원 재배치의 대상이 될 수 있는 활동이 된다. 본 연구에서는 문제의 복잡도를 줄이기 위해 유휴 자원에 한정하여 공백을 대체할 잔류 인력을 탐색하였지만 자원의 재배치로 인해 프로젝트 내 다른 활동에 또다른 인력 공백이 생겼을 경우, 이를 위한 대체 자원을 선정하는 규칙 및 전략을 개발하는 연구가 필요하다.

둘째, 대체 자원 없이 인력 공백에 대처하지 않았을 때 프로젝트에 발생하는 지연 시간이 계획된 프로젝트의 기간의 약 10% 이상인 반면에, 모종의 규칙을 바탕으로 대체 자원을 선정하여 인력 공백의 자리에 투입했을 때 프로젝트의 지연이 개선된 정도는 1일보다 작은 것을 표 4.4를 통해 파악 가능하다. 이는 실험에 활용된 경우처럼 프로젝트를 구성하는 각 활동에 투입되는 인원이 많다면 필요한 인원보다 한 명 부족한 상태로 작업을 수행해도 활동에 소요되는 시간이 크게 늘어나지 않기 때문에 재스케줄링 휴리스틱을 수행하는 과정 속에서 대체 자원을 활용하더라도 개선의 효과가 작게 나타난다. 프로젝트를 이탈한 인적 자원이 한 명이 아니라 여러 명이라면 프로젝트에 발생하는 지연이 더 커지면서, 인력 공백에 대처하기 위해 재스케줄링 휴리스틱을 적용했을 때 프로젝트 지연을 더 효과적으로 개선시킬 수 있을 것으로 기대된다. 결국, 프로젝트에서 다수의 인적 자원이 나간 상황에서 프로젝트가 최소한으로 지연될 수 있도록 내부에서 대체 자원을 선정하는 방법에 대한 연구도 진행할 수 있다.

셋째, 본 연구의 대체 자원 선정 전략 및 재스케줄링 휴리스틱은 이미 계획되어

진행 중인 프로젝트를 대상으로 한다. 달리 말하면, 휴리스틱을 적용하기 이전에 프로젝트의 일정을 처음부터 계획하는 방법에 대해서는 다루어지지 않았다. 프로젝트 스케줄링 방식에 따라 프로젝트 내 활동들의 시작 시간, 각 활동에 할당되는 자원의 종류가 전부 달라질 수 있고, 프로젝트 생성 알고리즘에 따라 본 연구에서 제안하는 재스케줄링 휴리스틱을 적용한 결과가 달라질 수 있다는 것을 의미한다. 프로젝트 일정 생성 알고리즘에 무관하게 새로운 방법론의 성능이 기존 연구에 비해 효과적인지 추가적으로 검증하는 과정이 반드시 필요하고, 이에 더해 실질 지연 기반의 대체 자원 선정 및 재스케줄링 휴리스틱의 효과를 극대화할 수 있는 proactive scheduling을 토대로 한 방법론을 새롭게 개발하는 연구 또한 진행할 수 있을 것이다.

마지막으로 미숙한 인력이 다른 기술을 습득하거나 기존에 가지고 있는 기술에 대한 전문성이 시간에 따라, 경험에 따라 더 향상되고, 이에 따라 재배치 시 작업의 지연을 줄이는 인적 자원의 학습효과까지 모델에 반영하여 방법론을 더 정교하게 발전시킬 수 있을 것으로 기대된다.

## 참고 문헌

- Abbasi, B., Shadrokh, S., & Arkat, J. (2006). Bi-objective resource-constrained project scheduling with robustness and makespan criteria. *Applied mathematics and computation*, 180(1), 146-152.
- Abrahamsson, P., Salo, O., Ronkainen, J., & Warsta, J. (2017). Agile software development methods: Review and analysis. *arXiv preprint arXiv:1709.08439*.
- Afshar-Nadjafi, B. (2021). Multi-skilling in scheduling problems: A review on models, methods and applications. *Computers & Industrial Engineering*, 151, 107004.
- Ahmadpour, S., & Ghezavati, V. (2019). Modeling and solving multi-skilled resource-constrained project scheduling problem with calendars in fuzzy condition. *Journal of Industrial Engineering International*, 15, 179-197.
- Ahuja, H. N., Dozzi, S. P., & Abourizk, S. M. (1994). *Project management: techniques in planning and controlling construction projects*. John Wiley & Sons.
- Al-Fawzan, M. A., & Haouari, M. (2005). A bi-objective model for robust resource-constrained project scheduling. *International Journal of production economics*, 96(2), 175-187.
- Artigues, C., & Roubellat, F. (2000). A polynomial activity insertion algorithm in a multi-resource schedule with cumulative constraints and multiple modes. *European Journal of Operational Research*, 127(2), 297-316.
- Aytug, H., Lawley, M. A., McKay, K., Mohan, S., & Uzsoy, R. (2005). Executing production schedules in the face of uncertainties: A review and some future directions. *European Journal of Operational Research*, 161(1), 86-110.
- Azizi, N., & Liang, M. (2013). An integrated approach to worker assignment, workforce flexibility acquisition, and task rotation. *Journal of the Operational Research*

Society, 64(2), 260-275.

- Bellenguez, O., & Néron, E. (2005). Lower bounds for the multi-skill project scheduling problem with hierarchical levels of skills. In Practice and Theory of Automated Timetabling V: 5th International Conference, PATAT 2004, Pittsburgh, PA, USA, August 18-20, 2004, Revised Selected Papers 5 (pp. 229-243). Springer Berlin Heidelberg.
- Bellenguez-Morineau, O., & Néron, E. (2007). A branch-and-bound method for solving multi-skill project scheduling problem. *RAIRO-operations Research*, 41(2), 155-170.
- Brooks Jr, F. P. (1995). *The mythical man-month: essays on software engineering*. Pearson Education.
- Browning, T. R., & Ramasesh, R. V. (2007). A survey of activity network-based process models for managing product development projects. *Production and operations management*, 16(2), 217-240.
- Buddhakulsomsiri, J., & Kim, D. S. (2007). Priority rule-based heuristic for multi-mode resource-constrained project scheduling problems with resource vacations and activity splitting. *European Journal of Operational Research*, 178(2), 374-390.
- Chua, D. K. H., & Hossain, M. A. (2011). A simulation model to study the impact of early information on design duration and redesign. *International journal of project management*, 29(3), 246-257.
- Correia, I., Lourenço, L. L., & Saldanha-da-Gama, F. (2012). Project scheduling with flexible resources: formulation and inequalities. *OR spectrum*, 34, 635-663.
- Deblaere, F., Demeulemeester, E., & Herroelen, W. (2011). Reactive scheduling in the multi-mode RCPSP. *Computers & Operations Research*, 38(1), 63-74.
- ElMekawy, T. Y., & ElMaraghy, H. A. (2003). Real-time scheduling with deadlock avoidance in flexible manufacturing systems. *The International Journal of*

- Advanced Manufacturing Technology, 22, 259-270.
- Eppinger, S. D., Whitney, D. E., Smith, R. P., & Gebala, D. A. (1994). A model-based method for organizing tasks in product development. *Research in engineering design*, 6, 1-13.
- Firat, M., & Hurkens, C. A. (2012). An improved MIP-based approach for a multi-skill workforce scheduling problem. *Journal of Scheduling*, 15(3), 363-380.
- Goldratt, E. (1997). *Critical Chain*; North River Press: Great Barrington, MA, USA.
- Heimerl, C., & Kolisch, R. (2010). Scheduling and staffing multiple projects with a multi-skilled workforce. *OR spectrum*, 32, 343-368.
- Herroelen, W., Demeulemeester, E., & De Reyck, B. (1999). A classification scheme for project scheduling (pp. 1-26). Springer US.
- Herroelen, W., & Leus, R. (2005). Project scheduling under uncertainty: Survey and research potentials. *European journal of operational research*, 165(2), 289-306.
- Katragjini, K., Vallada, E., & Ruiz, R. (2013). Flow shop rescheduling under different types of disruption. *International Journal of Production Research*, 51(3), 780-797.
- Kolisch, R., & Sprecher, A. (1997). PSPLIB-a project scheduling problem library: OR software-ORSEP operations research software exchange program. *European journal of operational research*, 96(1), 205-216.
- Lambrechts, O., Demeulemeester, E., & Herroelen, W. (2008). Proactive and reactive strategies for resource-constrained project scheduling with uncertain resource availabilities. *Journal of scheduling*, 11(2), 121-136.
- Lambrechts, O., Demeulemeester, E., & Herroelen, W. (2011). Time slack-based techniques for robust project scheduling subject to resource uncertainty. *Annals of operations Research*, 186(1).
- Li, H., & Womer, K. (2009). Scheduling projects with multi-skilled personnel by a hybrid MILP/CP benders decomposition algorithm. *Journal of Scheduling*, 12, 281-298.



- Liu, S. S., & Shih, K. C. (2009). Construction rescheduling based on a manufacturing rescheduling framework. *Automation in Construction*, 18(6), 715-723.
- Nevins, J. L., & Whitney, D. E. (1989). *Concurrent design of products and processes: a strategy for the next generation in manufacturing*. McGraw-Hill Companies.
- Otero, L. D., Centeno, G., Ruiz-Torres, A. J., & Otero, C. E. (2009). A systematic approach for resource allocation in software projects. *Computers & Industrial Engineering*, 56(4), 1333-1339.
- Sadeh, N., Otsuka, S., & Schnelbach, R. (1993). Predictive and reactive scheduling with the Micro-Boss production scheduling and control system. In *Proceedings, IJCAI-93 Workshop on Knowledge-Based Production Planning, Scheduling and Control*.
- Snauwaert, J., & Vanhoucke, M. (2022). Mathematical formulations for project scheduling problems with categorical and hierarchical skills. *Computers & Industrial Engineering*, 169, 108147.
- Ulrich, K. T., Eppinger, S. D., & Yang, M. C. (2008). *Product design and development* (Vol. 4, pp. 1-3). Boston: McGraw-Hill higher education.
- Van de Vonder, S., Demeulemeester, E., Leus, R., & Herroelen, W. (2006). Proactive-reactive project scheduling trade-offs and procedures. *Perspectives in modern project scheduling*, 25-51.
- Van de Vonder, S., Demeulemeester, E., & Herroelen, W. (2007). A classification of predictive-reactive project scheduling procedures. *Journal of scheduling*, 10, 195-207.
- Van de Vonder, S., Ballestin, F., Demeulemeester, E., & Herroelen, W. (2007). Heuristic procedures for reactive project scheduling. *Computers & Industrial Engineering*, 52(1), 11-28
- Vieira, G. E., Herrmann, J. W., & Lin, E. (2003). Rescheduling manufacturing systems: a framework of strategies, policies, and methods. *Journal of scheduling*, 6, 39-62.

- Wang, M., Liu, G., & Lin, X. (2022). Dynamic Optimization of the Multi-Skilled Resource-Constrained Project Scheduling Problem with Uncertainty in Resource Availability. *Mathematics*, 10(17), 3070.
- Xiao, J., Osterweil, L. J., Wang, Q., & Li, M. (2010). Dynamic resource scheduling in disruption-prone software development environments. In *Fundamental Approaches to Software Engineering: 13th International Conference, FASE 2010, Held as Part of the Joint European Conferences on Theory and Practice of Software, ETAPS 2010, Paphos, Cyprus, March 20-28, 2010. Proceedings 13* (pp. 107-122). Springer Berlin Heidelberg.
- Zhu, G., Bard, J. F., & Yu, G. (2005). Disruption management for resource-constrained project scheduling. *Journal of the Operational Research Society*, 56(4), 365-381.

## 부록

본 부록에서는 4 장의 실험에 활용된 2 개의 프로젝트 중 프로젝트 2 의 활동 및 인적 자원 정보, 그리고 각각의 프로젝트 생성 결과 도출된 일정 계획에 대해 보이며, 그 결과는 아래 표 A.1 ~ 표 A.4와 같다.

표 A.1: 프로젝트 2 활동 정보

활동 ( $i$ )	기술 요구 사항 ( $S_i$ )	필요 자원 수 ( $n_i$ )	소요 시간 ( $d_i$ )	후행 활동
1	$S_1 = \{0, 0, 0, 0\}$	7	0	2, 3, 4
2	$S_2 = \{0, 0, 0, 1\}$	13	7	5, 14, 23
3	$S_3 = \{1, 0, 1, 0\}$	13	6	8, 12, 20
4	$S_4 = \{0, 0, 1, 1\}$	13	5	8, 12, 18
5	$S_5 = \{1, 0, 0, 0\}$	5	10	6, 7, 21
6	$S_6 = \{0, 1, 0, 1\}$	14	9	11
7	$S_7 = \{1, 0, 1, 0\}$	9	7	15, 16
8	$S_8 = \{0, 1, 0, 1\}$	11	10	9, 11, 28
9	$S_9 = \{0, 0, 0, 1\}$	9	2	10, 22, 26
10	$S_{10} = \{0, 1, 1, 0\}$	4	9	13
11	$S_{11} = \{0, 0, 1, 1\}$	7	10	17, 22
12	$S_{12} = \{1, 0, 1, 1\}$	18	3	23
13	$S_{13} = \{0, 1, 1, 0\}$	10	2	21
14	$S_{14} = \{1, 0, 1, 0\}$	17	8	19, 25, 29
15	$S_{15} = \{1, 1, 0, 1\}$	16	5	20
16	$S_{16} = \{1, 1, 1, 0\}$	6	8	19
17	$S_{17} = \{0, 1, 1, 1\}$	15	4	25, 29, 30
18	$S_{18} = \{1, 0, 1, 1\}$	18	3	20, 27
19	$S_{19} = \{0, 0, 0, 1\}$	6	6	22, 24, 26
20	$S_{20} = \{1, 1, 1, 1\}$	12	5	24, 25
21	$S_{21} = \{1, 1, 1, 0\}$	19	9	30
22	$S_{22} = \{1, 0, 0, 1\}$	9	9	31
23	$S_{23} = \{1, 0, 0, 1\}$	11	2	24, 27, 29
24	$S_{24} = \{0, 0, 1, 1\}$	8	8	28
25	$S_{25} = \{1, 0, 0, 0\}$	8	6	26
26	$S_{26} = \{0, 1, 0, 1\}$	7	2	31
27	$S_{27} = \{0, 1, 0, 1\}$	14	9	28, 30
28	$S_{28} = \{0, 0, 1, 0\}$	4	1	31
29	$S_{29} = \{1, 0, 1, 1\}$	12	2	32
30	$S_{30} = \{1, 0, 0, 0\}$	8	8	32
31	$S_{31} = \{0, 0, 0, 1\}$	3	6	32
32	$S_{32} = \{0, 0, 0, 0\}$	0	0	-

표 A.2: 프로젝트 2 인적 자원 정보

자원 ( $j$ )	기술 역량 ( $C_j$ )	자원 ( $j$ )	기술 역량 ( $C_j$ )	자원 ( $j$ )	기술 역량 ( $C_j$ )
1	$C_1 = \{1, 1, 0, 0.5\}$	20	$C_{20} = \{0, 1, 0.5, 0.5\}$	39	$C_{39} = \{0, 0, 0.5, 1\}$
2	$C_2 = \{1, 0.5, 1, 0\}$	21	$C_{21} = \{1, 1, 0.5, 0.5\}$	40	$C_{40} = \{1, 1, 1, 1\}$
3	$C_3 = \{1, 0, 0.5, 0.5\}$	22	$C_{22} = \{0, 1, 0, 0.5\}$	41	$C_{41} = \{0, 0, 0.5, 1\}$
4	$C_4 = \{1, 0.5, 1, 0\}$	23	$C_{23} = \{1, 1, 1, 0.5\}$	42	$C_{42} = \{0.5, 1, 0.5, 1\}$
5	$C_5 = \{1, 0, 1, 0.5\}$	24	$C_{24} = \{0.5, 1, 0.5, 0\}$	43	$C_{43} = \{0, 1, 1, 1\}$
6	$C_6 = \{1, 0.5, 0, 0\}$	25	$C_{25} = \{1, 1, 1, 0.5\}$	44	$C_{44} = \{0, 0, 1, 1\}$
7	$C_7 = \{1, 1, 0.5, 0\}$	26	$C_{26} = \{0.5, 1, 0, 0\}$	45	$C_{45} = \{0, 1, 1, 1\}$
8	$C_8 = \{1, 0.5, 0, 1\}$	27	$C_{27} = \{1, 0.5, 1, 1\}$	46	$C_{46} = \{0.5, 0, 0, 1\}$
9	$C_9 = \{1, 0.5, 0.5, 0.5\}$	28	$C_{28} = \{0, 1, 1, 1\}$	47	$C_{47} = \{1, 0.5, 0, 1\}$
10	$C_{10} = \{1, 0.5, 1, 0.5\}$	29	$C_{29} = \{0, 0.5, 1, 0\}$	48	$C_{48} = \{0.5, 0.5, 0, 1\}$
11	$C_{11} = \{1, 0.5, 0, 1\}$	30	$C_{30} = \{0.5, 0.5, 1, 1\}$	49	$C_{49} = \{1, 1, 1, 1\}$
12	$C_{12} = \{1, 0.5, 1, 0.5\}$	31	$C_{31} = \{0, 0, 1, 0.5\}$		
13	$C_{13} = \{1, 0, 0.5, 0\}$	32	$C_{32} = \{0, 1, 1, 0\}$		
14	$C_{14} = \{1, 0.5, 1, 0.5\}$	33	$C_{33} = \{0, 0, 1, 0\}$		
15	$C_{15} = \{1, 1, 1, 0\}$	34	$C_{34} = \{1, 1, 1, 0.5\}$		
16	$C_{16} = \{0.5, 1, 1, 1\}$	35	$C_{35} = \{1, 0.5, 1, 1\}$		
17	$C_{17} = \{0, 1, 0.5, 1\}$	36	$C_{36} = \{0, 0.5, 1, 0\}$		
18	$C_{18} = \{0, 1, 0, 0.5\}$	37	$C_{37} = \{0, 1, 1, 1\}$		
19	$C_{19} = \{0.5, 1, 0, 0\}$	38	$C_{38} = \{0.5, 1, 1, 1\}$		

표 A.3: 프로젝트 1 일정 계획

활동 ( $i$ )	참여 자원 ( $R_i$ )	시작 시간 ( $t_i^s$ )	종료 시간 ( $t_i^f$ )	후행 활동
1	-	0	0	2,3,4
2	1,2,3,4,30,31,32,33,34,35,36,43,44	0	9	6,8,10
3	19,20,21,22,23,24,30,31,32,37,38,39,40,41,42	9	18	12,20,29
4	1,2,3,4,5,6,7,15,18,19,20,21,22,23,24,30,31,32,33,34,40,41,42	18	20	5,10,23
5	8,9,10,11,25,26,27,28,29,35,36,44,45,46,47,48,49,50,51,52,53	30	35	7,9,18
6	5,6,7,8,9,10,16,45,46,47	9	18	7,13,15
7	13,14,15,16,17,18,19,20,25,26,27,28,29,30,31, 32,33,34,41,42,43,44,45,46,47,48,49,50,54,55	44	49	14,22,26
8	17,18,19,20,21,22,23,24,25,26,37,48,49,50,51,52,53	20	30	9,11,13
9	1,2,3,4,5,6,7,21,22,23,24,25,35,51	49	58	17,26,27
10	1,2,15,19,20,21,22,23,43,44,45,46,47,48,49,50,54,55	35	36	12,16,19
11	16,17,18,19,20,21,22,27,28,29,38,54,55	36	38	15,20,27
12	3,4,5,6,7,16,24,25,26,27,28,29,51	49	58	14,17
13	5,6,7,8,9,10,11,12,13,14,16,25,26,27,28,29,51,52	38	44	17,19,21
14	3,4,5,6,7,8,9,10	62	67	24,30
15	1,2,3,4,11,12,13,14,15,16,17,18,23,24,25,26,27,28,29,30,31, 32,33,34,35,36,39,40,41,42,43,44,45,46,47,48,49,50	58	62	18,19,23
16	1,2,3,13,14,15,30,31,32,33,34,35,45,46	36	38	21,27,31
17	11,12,21,22,23,43,44,48,49,50,51,52,53,54,55	62	72	22,30
18	8,9,10,11,12,13,14,16,17,18,26,27,28,29,36,37,38,39,40,41,52,53	68	73	26,28,31
19	8,9,10,11,42,43,44,45,46,47,52,53,54,55	72	81	22,25,28
20	17,33,34,35,36,37,38,39,43,53,54,55	38	40	21,23
21	4,5,6,7,8,9,10,11,16,17,18,19,24,25,26,27,28,29,36	73	77	25
22	2,3,4,5,6,16,17,18,19,20,21,22,23,24, 33,34,35,36,37,38,39,40,41,42,47,48	81	88	24
23	12,16,17,18,19,20,21,22,23,24,33, 34,35,36,37,38,39,40,41,42,47,48	77	80	24,25
24	1,2,16,17,18,19,25,26,27,28,29,31,32,33,34,35,36,43,52,53,54,55	90	94	31
25	34,35,36,37,38,39,40,41,43,44,45,46,47,48,49,50,51,52	88	90	30
26	1,2,12,13,14,15,17,18,19,20	77	86	29
27	1,12,13,14,15,20,21,22,23,24,25,26,27, 28,29,30,31,32,37,38,39,40,41,42	62	68	28
28	7,8,9,10,11,12,13,14,15,30,49,50,51	86	95	29
29	3,4,5,6,7,20,21,22,23,24,25,44,45,46,47,48,49,50,51,52	95	103	32
30	16,17,18,19,20,21,26,27,28,29,30,31,32,33,34,35,42	103	111	32
31	1,8,9,10,11,12,13,14,15,30,32,33,37,38,39,40,41,42,53,54,55	111	121	32
32	-	121	121	-

표 A.4: 프로젝트 2 일정 계획

활동 ( $i$ )	참여 자원 ( $R_i$ )	시작 시간 ( $t_i^s$ )	종료 시간 ( $t_i^f$ )	후행 활동
1	-	0	0	2,3,4
2	38,39,40,41,42,43,44	0	7	5,14,23
3	1,2,3,4,5,6,7,8,9,10,27,28,29	0	6	8,12,20
4	30,31,32,33,34,35,36,37,45,46,47,48,49	0	5	8,12,18
5	1,11,12,13,14	7	17	6,7,21
6	15,16,17,18,19,20,21,22,23,38,39,40,41,42	18	27	11
7	1,2,9,10,11,12,13,14,27	21	28	15,16
8	15,16,17,18,22,23,24,25,26,45,46	6	16	9,11,28
9	39,40,41,42,43,44,45,46,47	27	29	10,22,26
10	19,20,27,37	30	39	13
11	28,29,30,38,39,48,49	29	39	17,22
12	1,2,9,10,11,12,13,14,32,38,39,40,41,42,43,44,45,49	15	18	23
13	15,16,17,18,19,20,24,25,26,32	52	54	21
14	2,3,4,5,6,7,8	7	15	19,25,29
15	3,4,5,6,15,16,17,18,19,20,24,25,26,43,44	29	34	20
16	8,9,21,28,29,30	30	38	19
17	20,21,22,23,28,29,30,31,42,43,44,45,46,47,48	48	52	25,29,30
18	1,2,10,11,12,13,14,31,33,34,35,36	18	21	20,27
19	38,45,46,47,48,49	52	58	22,24,26
20	3,4,21,22,23,31,40,41,42,43,44,45	34	39	24,25
21	5,6,7,15,16,17,18,19,21, 22,23,24,26,27,33,34,35,36,37	58	67	30
22	8,38,39,40,41,46,47,48,49	58	67	31
23	3,4,5,6,7,8,9,10,11,46,47	28	30	24,27,29
24	33,34,35,36,38,39,48,49	58	66	28
25	3,4,5,6,7,8,9,10	52	58	26
26	24,25,26,44,45,46,47	67	69	31
27	15,16,17,18,19,38,39,40,41,42,43,44,48,49	39	48	28,30
28	29,30,31,32	66	67	31
29	1,2,12,13,14,27,28,37,40,41,42,43	52	54	32
30	1,2,3,11,12,13,14	67	75	32
31	45,46,47	69	75	32
32	-	75	75	-

# Abstract

## Human Resource Rescheduling Methodology for Workforce Absence Situation using Net Delay Analysis

Yoon Hyung Joh

Department of Industrial Engineering

The Graduate School

Seoul National University

The product development process can be seen as a series of activities where teams with different expertise collaborate to rapidly develop and launch more innovative products, making diverse decisions. The complex collaboration of these activities, guided by teams with diverse skills, can be considered as a multi-skilled resource-constrained project scheduling problem (MSRCPSP). In this context, minimizing the project duration becomes the ultimate goal of product development. However, factors such as frequent employee turnover, employee absenteeism due to diseases like COVID-19, and the need for maintenance of machinery and equipment can lead to the absence of required resources at critical moments, causing the planned project schedule to be infeasible and resulting in project delays beyond the predetermined due date.

This study aims to minimize project delays by modifying project schedules in response to changes, especially resource disruptions(absences) that occur during the project. It proposes a method for resource rescheduling without introducing new personnel from outside. The proposed methodology in this study is based on net delay,



which is defined as the contribution of activity delay to the overall project delay resulting from the deployment of substitute resource from inside the project. It then selects the most suitable substitute resources to temporarily allocate to activities experiencing resource shortages. The rescheduling heuristic from this study derives a strategy for selecting the most suitable resource based on two criteria: suitability and net delay. Furthermore, through virtual examples from PSPLIB, this study demonstrates that the proposed methodology is more effective in minimizing delays compared to other resource rescheduling strategies.

Keywords: Project scheduling, Workforce absence, Resource disruption, Project rescheduling, Net delay, Industrial engineering

Student Number: 2021-21043