



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

설비중심공정의 인력 효율화를 위한
업무 재구조화 방법론

Work Restructuring Methodology for
Workforce Efficiency in Facility-Centric Process

2020 년 7 월

서울대학교 대학원

산업공학과

조 성 원

초록

화학공정의 반응기가 자율적인 작동이 가능한 것처럼, 작업자 없이 설비에 의해서만 진행되는 작업의 시간은 같은 공정에 함께 배치된 작업자 자원에 대한 유향 시간(Idle Time)이 된다. 설비와 작업자 자원 사이에 존재하는 이런 작업량 불균형은 설비의 작업을 전체 공정의 주경로(Critical Path)로 만들고 그에 따라 작업자 자원의 비효율성이 전체 근무시간 중 작업자 대기시간이 차지하는 높은 비중, 그리고 작업자 효율(Utilization)의 낮은 지표로 나타나게 된다.

설비중심공정(Facility-Centric Process)이란 화학 공장에서 사용되는 반응기의 작업과 같이, 작업자의 작업보다는 설비 자원이 담당하는 작업이 전체 공정의 주경로인 공정이다. 설비중심공정에서 발생하는 자원 간 업무량의 불균형 및 그로 인한 작업자 자원의 낮은 효율을 개선하기 위해서는 작업자 대기시간 안에 다른 공정의 작업을 배치하여 작업자 자원의 효율을 개선하는 업무 재구조화가 필요하다.

본 연구는 이를 위한 실증적인 방법론으로써 배치 자유도에 따른 작업 유형 분류 체계와, 그에 기반한 작업 유형별 순차적 배치 알고리즘을 제시한다. 작업 유형 분류 체계는 작업 배치 제약을 결정하는 작업 특징에 따라 전체 작업을 필수작업, 유동작업, 부수작업의 세가지 유형으로 분류한다. 이에 따라 분류된 개별 작업 단위들은 작업 간 선후 관계와 배치 제약을 반영한 정수 계획 문제(Integer Programming)을 통해 작업자 근무시간에 배치된다. 순차적 배치 알고리즘은 작업들을 필수, 유동, 부수작업의 순서대로 작업자 근무시간에 배치하는 알고리즘으로써, 배치 자유도가 적은 유형인 필수작업만을 대상으로만 정수 계획 문제를 구성하는 것으로 전체 작업 배치 문제의 규모 및 그에 따르는 계산량을 축소한다.

제시된 방법론의 인력 효율화 효과는 UDCA 생산 공정의 업무 재구조화 사례 연구를 통해 검증되었다.

주요어: 설비중심공정, 작업 분류, 정수 계획법, 업무 재구조화, 산업공학

학번: 2018-25619

목차

초록	i
목차	iii
표 목차	v
그림 목차	vi
제 1 장 서론	1
1.1 연구 배경 및 목적.....	1
1.2 관련 연구.....	3
1.3 문제정의.....	6
1.4 논문구성.....	8
제 2 장 작업 유형 분류와 배치 단위 정립	9
2.1 작업의 특성 및 작업 간 관계 정의.....	9
2.2 작업 유형 분류 체계.....	16
2.3 작업 배치 단위 정립.....	19
제 3 장 업무 재구조화 휴리스틱 알고리즘	22
3.1 작업 배치의 표현.....	22
3.2 작업 유형별 배치 제약과 순차적 배치.....	24
3.3 필수작업 배치를 위한 정수 계획 문제.....	26

3.4	유동작업과 부수작업의 배치.....	31
제 5 장	사례연구 : UDCA 생산공정의 재구조화	33
4.1	사례연구 개요.....	33
4.2	작업 유형 분류 및 작업 관계 모델링.....	34
4.3	업무 재구조화 결과 분석.....	37
제 5 장	결론 및 향후 연구 방향	40
5.1	결론.....	40
5.2	향후 연구 방향.....	41
	참고문헌	42
	Abstract	45

표 목차

표 2.1	작업 간 간격 제약의 유형.....	11
표 2.2	업무 재구조화에 고려해야 할 작업 특성.....	15
표 3.1	작업 배치 과정에 필요한 파라미터.....	27
표 4.1	농축 공정과 여과-세척 공정의 작업 유형 분류 결과.....	36

그림 목차

그림 1.1	UDCA 생산공정 업무의 작업 유형별 비중.....	2
그림 1.2	작업 배치를 제약하는 선후 관계와 간격.....	3
그림 1.3	기존 작업 단위의 예시.....	3
그림 1.4	작업자-기계공정도와 갱 공정도의 예시 (Freivalds et al., 2009).....	5
그림 1.5	업무 재구조화 방법론의 구성.....	7
그림 2.1	시간 간격 제약을 표현한 네트워크 (Dechter et al., 1991).....	10
그림 2.2	작업 간 시간 간격 제약에 따른 배치 제약.....	11
그림 2.3	작업 간의 고정된 간격에 의한 배치 제약.....	12
그림 2.4	분할 배치를 통한 간격 제약 극복.....	13
그림 2.5	작업과 작업 사이의 선후 관계 및 간격 제약 행렬의 예시.....	16
그림 2.6	작업 유형 분류 체계.....	18
그림 2.7	작업 배치 단위, 작업 모듈의 구성.....	20
그림 2.8	작업 배치 단위 간 간격 제약 행렬의 예시.....	21
그림 3.1	이진 변수 x , u , v 를 활용한 작업 배치 상황 표현.....	23
그림 3.2	작업 유형별 순차적 배치 휴리스틱 알고리즘.....	25
그림 4.1	UDCA 생산의 세부 공정.....	33
그림 4.2	여과-세척 공정의 작업 분석 예시.....	34
그림 4.3	농축 공정의 배치 단위 간 간격 제약 행렬.....	36
그림 4.4	정수 계획을 위한 Python 코드의 일부.....	37
그림 4.5	정수 계획에 따른 작업 모듈 배치 결과.....	38
그림 4.6	재구조화에 따른 업무 구조의 변화.....	38

제 1 장 서론

1.1 연구 배경 및 목적

자동화된 생산 공정이 드물지 않은 오늘날에도, 공정 진행의 관리 및 감독을 위해, 또는 여전히 자동화하기 어려운 작업의 존재로 인해 작업자가 담당하는 역할은 생산 현장에서 여전히 중요한 위치를 차지하고 있다(Joe, 2017).

화학공장의 반응기처럼 작업을 진행하는데 작업자의 실시간 통제와 감시를 필요로 하지 않는 자동화된 설비에 대해서 작업자는 설비의 시동이나 설비 내부 모니터링처럼 공정 진행에 대한 보조적, 간접적 업무를 담당한다. 이런 자동화된 설비와 협업하는 작업자 작업은 제품이나 물질의 취급, 공정 중 원료에 일어나는 변화와 시간적, 공간적인 거리를 둔다는 점에서 설비 자원의 작업 일정과 분명하게 구분된다. 이에 따라, 같은 공정에 배치되었더라도 설비가 자율적으로 작동하는 동안에 작업자 일정 상의 공백이 발생한다.

공정에서 발생하는 설비와 작업자 간의 업무량 불균형 및 그로 인한 작업자 대기시간의 발생은 작업자 자원의 효율성 저하를 불러온다(Freivalds et al., 2009). 작업자 대기시간이란 근무시간 도중에 작업자가 시행할 수 있는 작업이 없어 아무 작업도 하지 않는 상태로 보내는 시간이다. 그동안 작업자는 작업장을 벗어나 휴식하거나 다른 작업자를 돕는 등의 부차적인 활동을 할 뿐, 배치된 공정의 진행에 기여하지 않는다. 자원의 유휴 시간은 여러 스케줄링 선행 연구에서 자원 활용의 비효율을 나타내는 지표로 사용되고 있다(Caliş, 2015). 본 논문은 이런 제품 생산의 주경로(Critical Path)가 작업자의 업무보다는 설비의 작업 진행에 있는 공정에 대하여 ‘설비중심공정’이라 정의한다.

설비중심공정의 작업자 대기시간 발생 문제는 실제 업무 관찰 데이터를 통해서 확인된다. 그림 1.1은 제약 원료인 UDCA 생산 공정의 업무를 유형별로

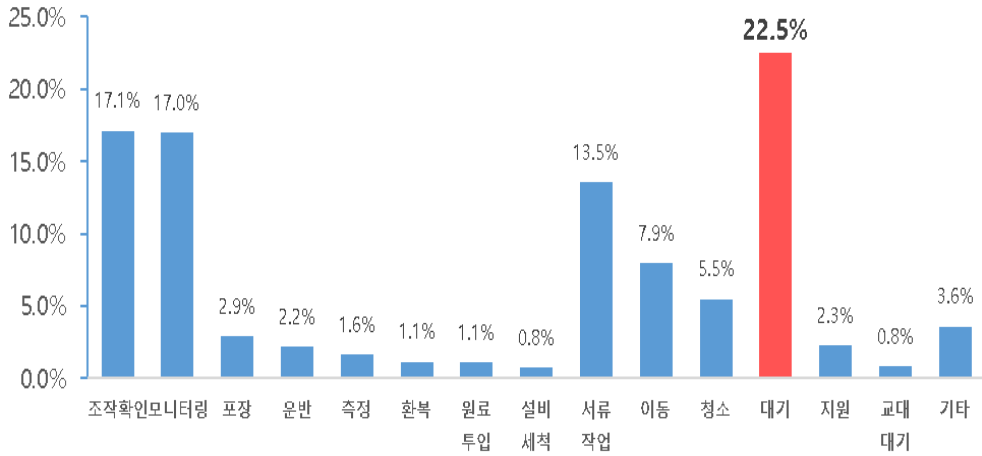


그림 1.1 UDCA 생산공정 업무의 작업 유형별 비중

나는 뒤 그 시간 비중을 나타낸 그래프이다. 대기시간은 설비 조작, 모니터링 등 많은 업무 중에서도 22.5%의 높은 비중을 차지하고 있다.

업무시간 내 작업자 대기시간의 활용은 해당 유휴 시간에 작업할 새로운 작업내용을 마련하는 것을 통해서 이루어질 수 있다(Freivalds et al., 2009). 공정 업무에 대한 작업 단위로의 분해 및 재배열은 기존 업무를 기반으로 새로운 업무 루틴(Routine)을 만든다는 면에서 ‘업무 재구조화’이다. 이런 업무 재구조화는 그림 1.2와 같이 같은 공정 안에서는 개별 작업 사이의 작업 진행 선후 관계 제약을, 서로 다른 공정 사이에서는 배치될 작업 크기가 배치되는 간격의 크기보다 작아야 하는 배치 제약의 영향을 받는다.

작업 배치에 고려해야할 이런 난점들과 관련하여, 기존의 업무 현장에서 사용되는 작업에 대한 정보와 개별 작업 단위는 작업의 소요시간과 수행 순서만을 나타낼 뿐, 작업 배치에 영향을 주는 앞의 특성들을 전혀 반영하지 않아 업무 재구조화에 사용되기에는 적절하지 않다(그림 1.3). 따라서 설비중심공정 내에서 필연적으로 발생하는 작업자 유휴 대기시간을 활용하기 위해, 작업 간 선후 관계와 간격 제약을 고려한 체계적인 업무 재구조화 방법론이 필요하다.

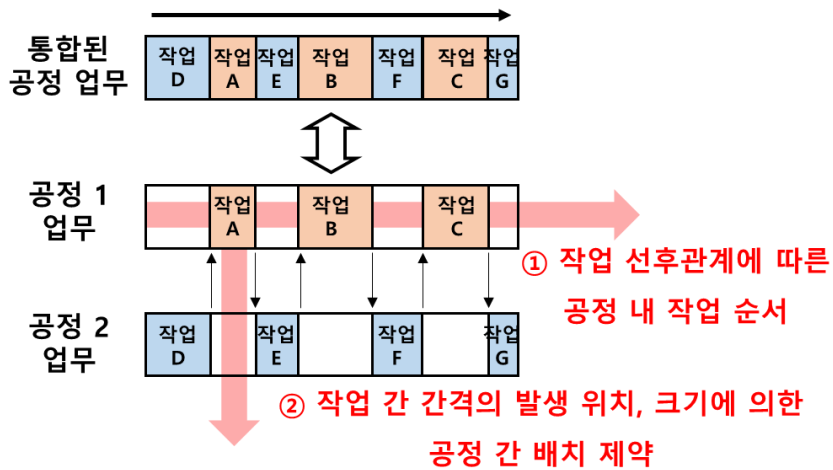


그림 1.2 작업 배치를 제약하는 선후 관계와 간격

작업 내용	작업시간 (시간:분)
대기	0:15
서류 작업	0:14
황산 투입 개시	0:04
황산 투입 모니터링	0:05
서류 작업	0:02

그림 1.3 기존 작업 단위의 예시

1.2 관련 연구

1.2.1 작업의 재분배

임의의 작업 공정에 투입되는 작업자 자원과 설비 자원은 해당 자원의 종류나 공정의 내용에 따라 각자 다른 역할을 수행한다. 개별 설비와 작업자들

사이에는 맡은 공정의 종류에 따라 주어지는 작업량에 차이가 생기고, 이로 인해 일부 자원은 전체 작업 진행에 대한 병목 공정으로 전락하여 전체 공정 지연을 일으킨다. 그에 대한 대표적인 대처로는 개별 자원이 담당할 작업을 새로 분배하는 것을 통해 전체 자원의 가동률을 올리는 방법론인 라인 밸런싱(Line Balancing)이 있다(Thomopoulos, 1967).

개별 작업 사이의 선후 관계가 주어진 공정 작업들을 작업 주체에게 고르게 분배하기 위하여 선행 행렬(Precedence Matrix)을 사용한 알고리즘을 제안한 Hoffmann(1963)를 비롯하여 여러 연구들이 진행되어왔다(Boysen, 2007). 그러나 라인 밸런싱은 전체 공정을 가장 빨리 마칠 수 있도록 작업량을 분배하는 것이 목적이기 때문에, 분배된 이후의 작업들이 자원들에 의해 구체적으로 어떤 방식으로 진행될 지에 대한 고려가 부족하다.

머신 커플링은 작업자의 업무량이 같은 공정에 함께 배정된 기계설비의 업무량보다 더 적을 경우에, 한 작업자가 여러 개의 설비를 동시에 운용할 수 있도록 작업자의 업무를 배정하는 것이다(Freivalds et al., 2009). Freivalds et al.(2009)는 머신 커플링을 위한 방법으로, 공정에 투입되는 자원의 작업 내용을 시간에 따라 비교하여 양 자원 작업 주기 사이의 시간 관계를 분석하는 작업자-기계공정도와 갭 공정도를 소개하고 있다(그림 1.4).

그러나 머신 커플링은 할당된 자원의 수를 통해 개별 자원에게 할당된 업무량을 조정하는 것에 그치기 때문에, 각 자원의 업무 내부에서 발생하는 대기시간은 다룰 수 없다는 한계가 있다. 이를 활용하기 위해서는 기존 업무를 대기가 실제 발생하는 수준까지 분해하여 분석할 수 있는 방법론이 필요하다.

1.2.2 Work Breakdown Structure

WBS(Work Breakdown Structure)란 규모가 크고 복잡한 업무를 관리할 수 있는 수준의 작은 단위인 워크 패키지(Work Package)로 분해한 뒤 이들 사이의

Worker and Machine Process Chart

Subject Charted Milling slot in retractor clamp Chart No. 807
 Drawing No. 1-1492 Part No. 11492-1 Chart of Method Proposed
 Chart Begins Loading mchs. for milling Charted by C.A. Anderson
 Chart Ends Unloading milled clamp Date 8-27 Sheet 1 of 1

Element description	Operator	B.&S. Hor. Mill	
		Machine 1	Machine 2
Stop machine #1	.0004		
Return table mch #1 5 inches	.0010	Unloading .0024	Mill slot .0040
Loosen vise remove part and lay aside (mch. #1)	.0010		
Pickup part and tighten vise mch. #1	.0018		
Start machine #1	.0004	Loading .0032	Idle
Advance table and engage feed mch. #1	.0010		
Walk to machine #2	.0011		
Stop machine #2	.0004		
Return table mch #2 5 inches	.0010	Mill slot .0040	Unloading .0024
Loosen vise remove part and lay aside (mch. #2)	.0010		
Pick up part and tighten vise mch. #2	.0018		
Start machine #2	.0004		Loading .0032
Advance table and engage feed mch. #2	.0010	Idle	
Walk to machine #1	.0011		
Idle man time per cycle	.0000	Idle hours machine #1	.0038
Working time per cycle	.0134	Productive hours mch. #1	.0096
Man-hours per cycle	.0134	Machine #1 cycle time	.0134

GANG PROCESS CHART PROPOSED METHOD
 Hydraulic Extrusion Press Dept. II Bellefonte, Pa. Plant
 Charted by B.W.N 4-15 Chart G-85

MACHINE OPERATION TIME	MACHINE OPERATION TIME	ASSISTANT PRESS OPERATOR OPERATION TIME	DUMMY KNOCKER OPERATION TIME	PULL-OUT MAN OPERATION TIME
Elevate billet .07	Elevate billet .07	Grease die & position back in die head .12	Position shell on small press .10	Pull rod toward cooling rack .20
Position billet .08	Position billet .08	Walk to furnace .05	Press dummy out of shell .12	Walk back toward press .15
Position dummy .04	Position dummy .04	Rearrange billets in furnace .20	Dispose of shell .18	
Build pressure .05	Build pressure .05	Return to press .05	Dispose of dummy and lay aside tongs .12	Grab rod with tongs and pull out .45
Extrude .45	Extrude .45	Idle time .09	Idle time .23	
Unlock die .06	Unlock die .06	Open furnace door & remove billet .19		
Loosen & push out shell .10	Loosen & push out shell .10	Ram billet from furnace & close furnace door .10	Grab tongs & move to position .05	
Withdraw ram & lock die in head .15	Withdraw ram & lock die in head .15	Run head & shell .11	Guide shell from shear to small press .20	Straighten rod end with mallet .11
		Shear rod from shell .04	Hold rod while die removed at press .09	
		Pull die off end of rod .05		
Working time 1.00 Min.	1.00 Min.	.91 Min.	.77 Min.	1.00 Min.
Idle time 0	0	.09 Min.	.23 Min.	0

그림 1.4 작업자-기계공정도와 갱 공정도의 예시 (Freivalds et al., 2009)

관계를 체계적으로 분석하는 프로젝트 관리 기법이다(Norman et al., 2008). 전체 프로젝트는 체계적인 기준에 따라 계층적 구조로 분해되며, 프로젝트 관리의 목적에 따라 WBS 구조와 그 워크 패키지의 크기 및 내용이 달라질 수 있다(Globerson, S., 1994, Li et al., 2019).

Lee et al.(2010)는 프로젝트를 구성하는 개별 작업 및 그를 담당하는 조직 사이에 존재하는 업무적 상관 관계의 강도를 DSM(Design Structure Matrix)를 통해 표현한 뒤, 서로 밀접한 관련이 있는 작업들을 하나로 묶쳐 WBS 및 워크 패키지를 구성했다. 워크 패키지에 대한 클러스터링은 많은 작업으로 구성된 복잡한 프로젝트에 대한 관리를 용이하게 만든다. Sequeira(2015)는 WBS와 각 워크 패키지를 담당하는 인원에 대한 인건비와 작업 소요 기간에 대한 정보를 통해 프로젝트의 비용 구조를 분석하는 방법론을 제시했다. Supriadi et al.(2017)는 공항 건설 프로젝트의 개별 워크 패키지에서 발생 가능할 수 있는 작업 지연에 대한 위험성을 위험성의 크기와 위험 발생 빈도라는 두 인자의 조합으로 표현하는 것을 통해서 워크 패키지 및 WBS의 계층 구조를 이루는 여러

요소들에 대한 위험성 분석을 시도한 바 있다.

WBS를 활용한 프로젝트 스케줄링 역시 여러 연구에 의해 다뤄져 왔다. Deckro(1992)는 개별 워크 패키지의 비용과 기간을 프로젝트 스케줄링의 인자로 반영한 수리 계획(Mathematical Programming) 방법론을 제시했다. Hassanein(2004)은 고속도로 건설 프로젝트의 구성 작업들이 상호 간에 가질 수 있는 진행 선후 관계의 요인을 분석하는 것으로 WBS를 구축한 뒤, 그를 통해 스케줄링에 대한 실증적인 연구를 진행한 바 있다. Zhang(2008)은 항공기 조립 공정의 스케줄링을 위해 작업들이 가진 조립 순서, 필요한 자원의 종류, 진행에 필요한 소요시간 등의 정보를 반영할 수 있는 WBS를 구축했다.

이렇게 WBS 방법론은 다양한 상황에서 유연하게 활용될 수 있지만 프로젝트 관리의 목적과 해당 프로젝트가 위치한 도메인의 지식에 따라 WBS의 구조 및 워크 패키지로의 분해 기준이 달라지기 때문에, 설비중심공정의 업무 재구조화는 그에 맞는 새로운 WBS를 통해 진행되어야 한다. 본 연구에서는 설비중심공정의 작업자 대기시간을 활용하기 위해 공정 업무에 대한 WBS를 구성하고 이를 통해 새로운 업무 재구조화 대안을 생성하는 방법론을 제시할 것이다.

1.3 문제정의

특정 근무시간에 대한 작업 배치는 작업자 자원이 온전히 한가지 작업에 점유되기 때문에 다른 작업이 동시에 진행될 수 없는, 자원이 한정된 프로젝트 스케줄링 문제(Resource-Constrained Project Scheduling Problems)이다(Rodammer et al., 1988, Abdolshah, 2014). 또한 한 작업자가 여러 작업을 동시에 수행할 수 없으므로, 작업이 배치되기 위해서는 해당 작업의 소요시간이 그가 배치될 대기시간의 크기보다 더 작아야 하는 ‘간격 제약’이 존재한다. 실행 가능한 작업 배치 대안을 위해서는 우선 개별 작업의 배치 가능성을 결정하는 요인에

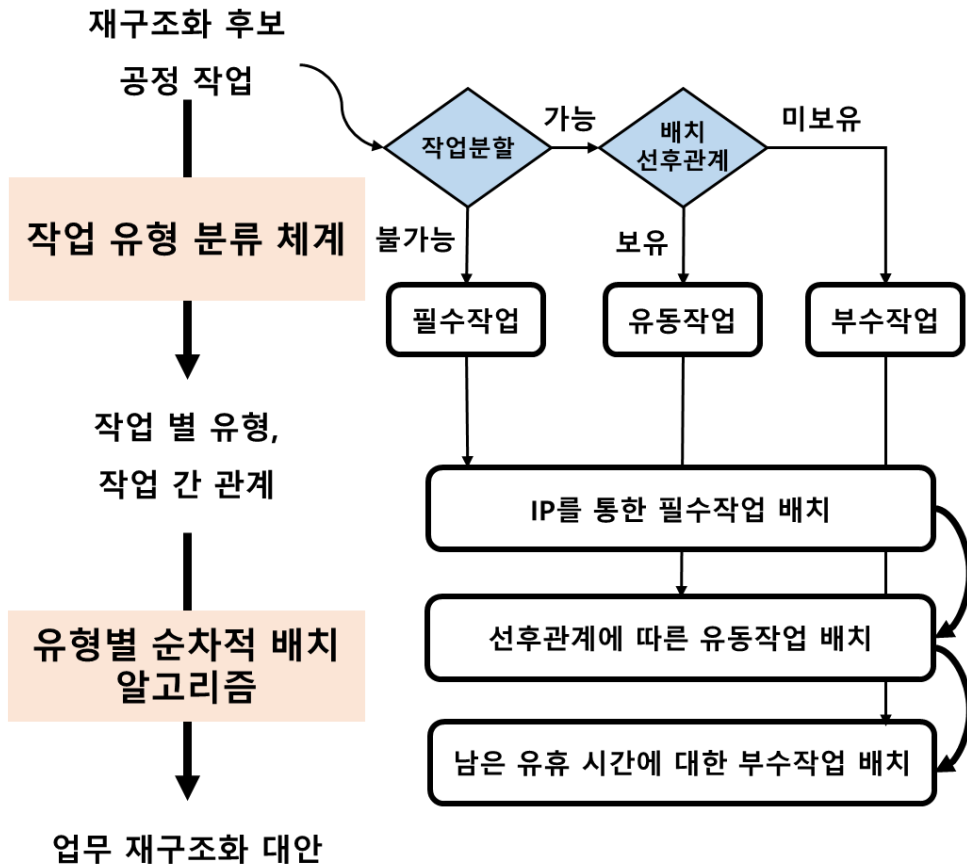


그림 1.5 업무 재구조화 방법론의 구성

대한 분석이 필요하다.

작업 사이의 진행 선후 관계, 그리고 배치되는 작업의 분할 진행 가능성은 근무시간 중 특정 시간에 대한 작업의 배치 가능성을 결정하는 큰 요소이다. 작업 간 진행 선후 관계란 각 작업의 시작에 요구되는 설비나 원료의 상태에 따라 같은 공정 내의 두 작업 중에서 특정한 한 작업이 시간적으로 먼저 진행되어야 한다는, 작업 배치의 제약조건이다. 작업의 분할 진행 가능성이란 개별 작업의 소요시간이 여러 번 나눠 진행될 수 있는지의 여부를 나타낸다.

분할 진행할 수 없는 작업은 그 자신보다 작은 크기의 간격에는 배치될 수 없지만, 분할 진행 가능한 작업은 짧은 크기의 작업 간 간격에도 분할되어 배치될 수 있다는 점에서 배치의 유연성을 가지고 있다.

본 연구에서 제안하는 업무 재구조화 방법론은 설비중심공정의 업무에서 발생하는 작업자 대기시간에 다른 공정의 작업을 배치하여, 작업자 자원 효율을 향상시키면서 공정에 투입되는 인력의 수를 줄이는, 실행 가능한 새로운 업무 루틴을 생성하기 위한 것이다. 그림 1.5와 같이, 업무 재구조화 방법론은 배치 제약에 따라 작업들을 분류하고 배치 단위를 재정립하는 ‘작업 유형 분류 체계 정립’, 그리고 재정립된 작업 단위를 기반으로 실제 업무 대안을 생성하는 ‘작업 유형별 순차적 배치 알고리즘 제안’의 두 단계로 구성된다. 또한 순차적 배치 과정에서의 작업 배치는 정수 계획(Integer Programming) 모형을 통해 진행된다.

1.4 논문구성

본 논문은 서론을 포함한 총 5장으로 구성된다. 제2장에서는 본격적인 재구조화에 앞서 개별 작업의 특성 및 작업 간 관계의 특성을 작업 배치의 관점에서 정의한다. 또한 이를 사용해 작업의 유형을 분류 체계를 제안하여 작업의 배치 단위를 새로 정립한다. 그에 이어 제3장은 재정립된 작업 단위를 사용한 본격적인 업무 재구조화에 대해 기술하며, 그 과정에서 유형별 순차적 배치 휴리스틱 알고리즘과 정수 계획 모형을 소개한다. 제4장은 제약 공정인 UDCA 생산 공정의 작업 관찰 데이터를 활용해 앞선 방법론을 검증하는 내용을 담고 있다. 마지막으로 제5장에서는 본 연구 내용의 요약 및 의의, 그리고 추후 연구과제에 대해서 다룬다.

제 2 장 작업 유형 분류와 배치 단위 정립

2.1 작업의 특성 및 작업 간 관계 정의

설비중심공정의 작업자가 수행하는 업무는 설비의 조작이나 원료 투입 등 여러 세부 작업으로 구성되어 있으며, 근무시간 동안 작업자는 이 고정된 루틴에 따라 정해진 내용의 작업을 순차적으로 실행한다. 업무 재구조화의 목적은 이런 공정 업무의 구성 작업들을 분리한 뒤, 한 작업자가 수행할 수 있도록 새로운 업무 대안 루틴으로 조립하는 것이다. 업무 루틴 안에 작업이 배치되기 위해서는 배치와 관련된 작업의 특성에 대한 분석이 필요하다. 이를 위해 작업의 배치 가능성을 결정하는 작업 선후 관계와 분할 가능성을 토대로 재구조화에 사용될 개별 작업 및 작업 간 관계를 모델링할 것이다.

2.1.1 작업 간의 진행 선후 관계

공정의 구성 작업은 원재료의 완제품으로의 변화를 위해 수행된다. 이에 따라 시간적으로 뒤에 진행되는 작업은 직전 작업에 따른 원료와 설비의 상태를 넘겨받게 되고, 다른 상태로 변화시킨 뒤, 다음 배치된 작업으로 전달한다. 한 공정 내의 작업들은 이 설비와 원료의 상태를 매개로 서로의 시행에 의존성을 가지고 있다. 따라서 작업의 온전한 수행을 위해서는, 더 나아가 온전한 제품을 만들기 위해서는 공정이 작업과 작업 사이의 시행 조건을 고려하여 진행되어야 한다

배치 선후 관계란 각 작업에 요구되는 원료, 설비의 여러 조건에 따라, 공정 내의 두 작업 중 한 작업이 시간적으로 먼저 진행되어야 한다는 제약조건이다. 선후 관계를 가진 두 작업은 서로가 배치 가능한 시간의 범위를

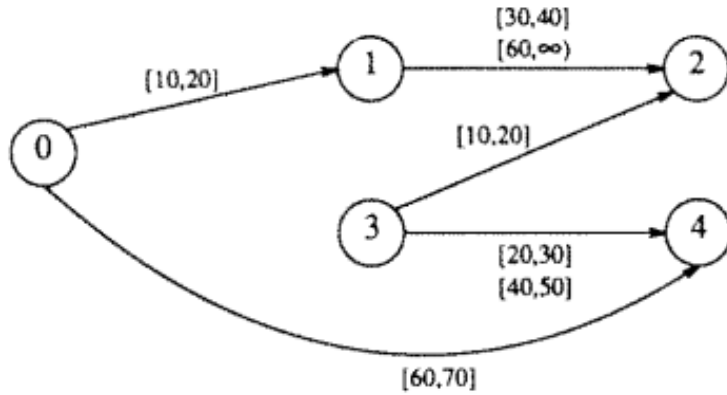


그림 2.1 시간 간격 제약을 표현한 네트워크(Dechter et al., 1991)

제약한다. Hoffmann(1963)은 이런 전체 공정의 구성 작업 간 선후 관계를 방향 그래프(Directed Network)와 그에 대응하는 행렬을 통해 표현했다.

작업 간에 발생하는 선후 관계는 업무의 진행 순서일 뿐 아니라 서로의 시간적 간격의 크기 역시 제약한다. Allen(1981)은 사건(Event)과 사건 사이에서 발생할 수 있는 시간적 관계를 분류하고 이를 일관성 있게 표현하는 방법을 제시했다. Dechter et al.(1991)은 선후 관계를 가진 여러 사건들 사이의 시간 간격 크기 제약을 그림 2.1과 같이 수식과 선후 관계 네트워크로 표현한 바 있다.

본 연구에서 대상으로 하는 설비중심공정의 구성 작업들은 설비의 자율적인 작동 시간과 앞뒤로 연계되어 진행되기 때문에, 작업자 관점에서 설비의 자율 작동 시간은 작업 사이의 간격이다. 이에 따라 주경로인 설비 작업의 진행을 지연시키지 않기 위해 그 간격이 고정되는 관계가 발생하므로, 이를 작업 배치에 고려해야 한다.

이를 위해 작업 간 간격 크기 제약을 표 2.1과 같이 상수 L 에 대하여, L 이상의 간격, L 로 고정된 간격, L 보다 작은 간격, 그리고 간격 크기의 제약이 없는 단순 선후 관계로 분류했다. 주어진 제약은 그림 2.2와 같이 근무시간 중 개별 작업이 배치될 수 있는 구간을 제약한다.

표 2.1: 작업 간 간격 제약의 유형

간격 제약의 유형	내용
간격 $\geq L$	두 작업 사이에 일정 길이 이상의 간격이 필요한 경우
간격 = L	두 작업 사이에 고정된 크기의 간격이 필요한 경우
간격 $\leq L$	후행 작업이 선행 작업 종료 후 일정 시간 안에 진행되어야 하는 경우
단순 선후 관계	두 작업 간에 간격 크기의 제약이 없는 경우

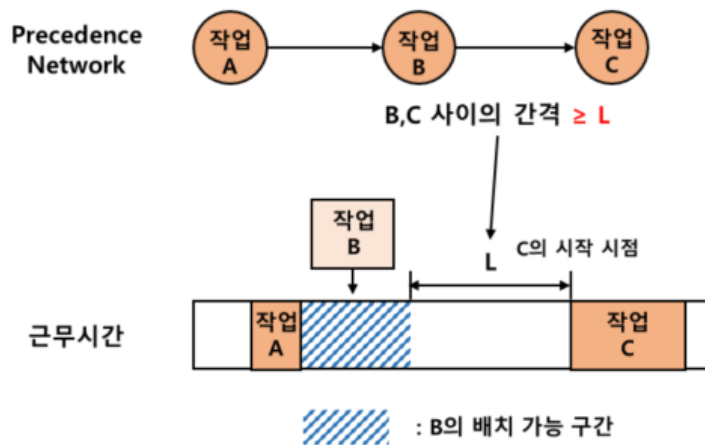


그림 2.2 작업 간 시간 간격 제약에 따른 배치 제약

2.1.2 작업 간 간격 크기 제약과 분할 진행 가능성

업무 재구조화에서 작업의 배치는 서로 다른 공정 업무 사이에서 일어나는 작업과 대기시간 사이의 대응이다(그림 2.3). 작업자가 여러 작업을 동시에 진행할 수 없으므로, 작업이 성공적으로 배치되기 위해서는 배치되는 작업의 크기는

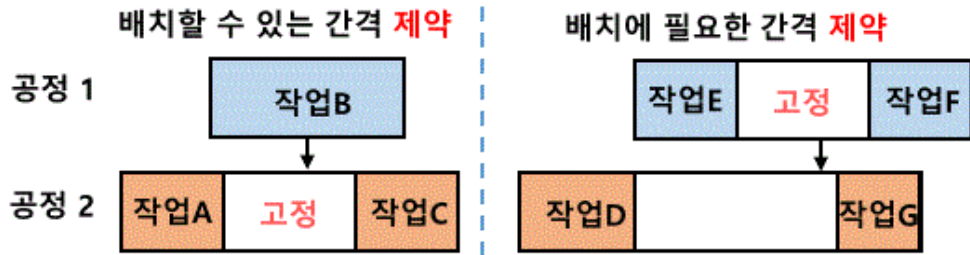


그림 2.3 작업 간의 고정된 간격에 의한 배치 제약

배치될 간격의 크기보다 더 작아야 한다. 이런 간격 제약은 작업 사이의 간격이 유연하게 조정될 수 있는 잡 샵 스케줄링(Job Shop Scheduling)과 같은 단순 배열에서는 나타나지 않는다(Calıř, 2015).

이것이 개별 작업 배치의 제약 조건으로 적용되는 상황은 배치되는 작업의 크기, 그리고 배치가 일어나는 간격의 크기가 제한될 때이다. 간격 제약에 따라, 고정된 작업 간 간격은 그림 2.3의 왼쪽 그림과 같이 그에 배치될 수 있는 다른 공정 작업의 크기를 제한한다. 배치될 간격이 고정되었던 상황과는 반대로 배치될 작업 쪽의 간격이 고정된 경우, 다른 작업과 고정된 간격을 가지고 함께 배치되므로 그림 2.3의 오른쪽 그림과 같이 해당 작업이 배치될 간격의 크기를 제약 받는다. 업무 재구조화 과정에서 고려할 간격 제약이 반드시 간격 크기의 고정일 필요는 없다. 재구조화 대상 공정의 맥락과 내용에 따라 간격 제약은 특정 값이 아니라, 작업 간에 필요한 최소한의 간격이나 작업 간에 발생 가능한 간격 크기의 상한과 같은, 간격 크기에 대한 범위 제약이 될 수 있다.

간격 제약을 실질적인 작업 배치의 제약조건으로 만드는 데 있어 또다른 결정 요인은 배치되는 작업의 분할 진행 가능성이다. 공정 업무를 구성하는 작업 중에는 작업 현장의 바닥 청소나 공정과 별개로 주어진 서류 작성과 같이 작업자가 도중에 중단하더라도 이후 여유로운 시간에 그대로 이어서 진행할 수 있는 작업이 존재한다.

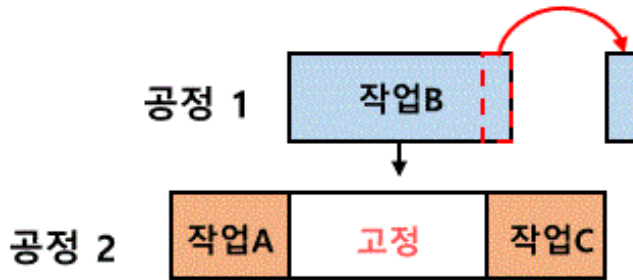


그림 2.4 분할 배치를 통한 간격 제약 극복

다른 작업을 배치하기 위해 진행 중인 작업을 중단하는 상황은 선점형 스케줄링(Preemptive Scheduling)에 대한 기존 연구에 의해 다루어져 왔다(Muntz et al., 1969). 작업 선점이란 진행 중인 작업을 중단하고 그 자원에 다른 작업을 배치하는 결정을 내리는 것이고, 이를 통해 스케줄링에 작업의 우선도를 반영하면서 개별 작업들의 평균 종료 시간(Average Turnaround Time)을 단축시키는 효과를 얻을 수 있다(DasGupta et al., 2001, Kettimuthu, 2005). 그러나 업무 재구조화에서의 작업 분할은 배치를 결정하는 시점에 다른 작업에 대한 정보가 주어져 있다는 점에서 미래에 배치될 작업에 대한 정보가 불확실한 실시간 스케줄링과 차이가 있다.

본 연구에서는 이렇게 업무 중 몇 번이고 끊어서 진행할 수 있는 작업을 분할 진행 가능하다고 정의한다. 이에 따라, 작업이 분할되는 패턴은 업무 배치에 따라 유연하게 결정될 수 있고, 분할된 각 부분들은 그 구체적인 세부 내용이 특정되어 독립적인 배치 단위로 다루어 지기보다는 원 작업의 총 소요 시간을 분담하는 원본 작업의 일부로 취급할 것이다. 따라서 재구조화 과정에서 분할된 작업 부분들은 모두 동일한 배치 제약을 가지고 있으며, 상호 간에는 선후 관계를 가지지 않는다.

이런 분할 진행 가능한 작업은 간격의 크기보다 작은 크기로 분할될 수

있기 때문에 그림 2.4와 같이 배치되는 간격 크기의 제약에 영향을 받지 않는다. 그 배치 과정에서 제약된 간격에 대해 배치 가능한 양의 작업은 분할되어 배치되고, 분할된 나머지 작업량은 이후 등장하는 다른 작업 간 간격에 배치된다.

2.1.3 작업과 작업 간 관계의 정의

본 단락에서는 업무 재구조화 방법론에 앞서 작업 및 작업 과 작업 사이의 관계의 특성에 대해서 정의한다. 작업의 선후 관계 보유 여부와 작업의 분할 진행 가능성은 근무 시간 위의 특정 시점에 대한 작업 배치 가능성을 결정하는 두 요소이다. 선후 관계를 보유한 작업은 선행 작업과 후행 작업이 언제 배치되었는지 고려해야한다. 선행 작업과 후행 작업의 배치 과정이 재구조화 과정에서 분리되어 일어나더라도, 먼저 배치되는 작업은 이후에 배치될 선후 관계 작업을 고려하여 그 작업 시점이 결정되어야 한다. 분할 진행 가능한 작업은 작업 배치에 있어서 배치되는 간격의 크기에 대한 제약을 받지 않기 때문에 분할 진행 불가능한 작업보다 배치할 수 있는 시점의 제약이 더 적다.

개별 작업을 구성하는 특성들을 표 2.2와 같이 작업 소요시간, 공정 내 다른 작업과의 선후 관계 보유 여부, 작업의 분할 진행 가능성, 그리고 선후 작업의 연속 진행 여부로 정의한다. 작업의 소요시간이란 공정을 구성하는 각 작업의 내용을 완전히 실행하기 위해서 작업자 자원이 투입되어야 하는 시간의 길이이다. 실행 가능한 업무 재구조화 대안에서 작업자는 모든 작업들을 필요한 소요시간만큼 진행해야 한다. 작업의 진행 가능성은 연속된 두 작업 사이의 간격 발생 가능성을 의미하는 것으로 다른 작업의 배치 가능성을 제한한다. 또한 연속하여 진행되는 작업들은 각자의 배치가 연속된 다른 작업의 배치 상황을 완전히결정하므로 업무 재구조화에 있어서 작업 배치의 단위가 된다.

표 2.2: 업무 재구조화에 고려해야 할 작업 특성

작업 특성	내용
작업 소요시간	작업 내용의 완전한 수행에 필요한 작업 시간
선후 관계 보유 여부	작업 배치에 있어 고려해야하는 공정 내 다른 작업과의 배치 선후 관계의 보유 여부
분할 진행 가능성	해당 작업의 진행이 도중에 중단되거나 이후에 다시 시작 될 수 있는 지의 여부
연속 진행 필요성	작업의 시작, 종료 시점과의 사이에 대기시간 없이 연속하여 진행되어야 하는 선행 작업, 후행 작업의 보유 여부

작업 배치 과정에 있어서 작업 시간에 대한 정보 만이 작업과 작업이 상호작용할 수 있는 통로이므로, 두 작업 사이에 존재하는 모든 종류의 배치 관계는 상대 작업의 진행 시점과 관련되어 있다. 설비중심공정에서 진행되는 작업은 설비나 원료 상태에 대한 비 가역적인 변화를 수반하므로, 두 작업의 관계는 둘 중 한 쪽 작업이 항상 먼저 배치되어야 하는 선후 관계를 포함하고 있다고 가정한다.

DSM은 시스템을 구성하는 여러 요소 간의 관계를 해당 행렬 요소의 내용으로 표현하는 것으로 복잡한 시스템 내부 관계를 분석에 용이하도록 간결하게 나타내는 기법이다(Browning, 2015). DSM은 그 형태가 정해져 있지 않기 때문에, 분석의 목적에 따라, 요소의 내용에 따라, 또는 표현하고자 하는 관계의 유형에 따라 수많은 형태를 가질 수 있다. Maheswari et al.(2005)는 프로젝트의 스케줄링을 위해, 그를 구성하는 세부 작업들이 같은 시간대에 서로 중첩(Overlapping)되어 진행될 수 있는 정도를 수치로 표현하는 DSM을 제시했고, Lee et al.(2010)은 프로젝트를 담당하는 조직 간의 업무적 상관관계의 강도를 DSM을 통해 표현한 바 있다.

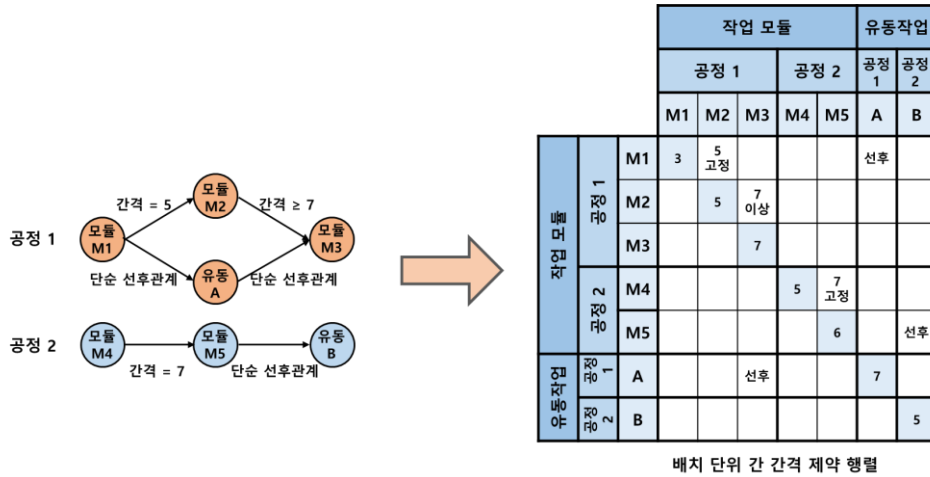


그림 2.5 작업과 작업 사이의 선후 관계 및 간격 제약 행렬의 예시

본 연구는 공정을 구성하는 모든 작업 사이의 간격 제약 관계를 표현하기 위해서 ‘간격 제약 행렬’을 정의했다. 그림 2.5는 업무를 구성하는 작업과 작업 사이의 관계를 표현한 네트워크와 그에 대응하는 간격 제약 행렬이다. 행렬의 각 행과 열은 공정을 구성하는 작업을 나타낸다. 행렬의 대각요소는 해당 행의 진행에 필요한 작업 소요시간을 나타내고, 비 대각요소는 해당 행과 열의 두 작업 간에 존재하는 관계에 대해서, 해당 관계가 가진 간격 제약의 내용을 수치와 단어로 나타낸다.

2.2 작업 유형 분류 체계

설비 중심 공정의 업무 재구조화에서 개별 작업이 가지고 있는 선후 관계 제약, 그리고 해당 작업의 분할 진행 가능성은 각 시점에서 해당 작업에 대한 배치 가능성을 결정한다. 본 연구는 작업 배치 제약 조건의 내용을 구별하는 이 두 특징에 따라 모든 작업을 크게 세가지 유형으로 구별한다.

개별 작업의 분할 진행 가능성은 작업 배치에 대한 간격 제약의 적용 여부를 결정한다. 앞의 그림 2.3과 같이 분할 진행할 수 없는 작업은 크기가 제약된 간격에 대해 자신의 길이보다 긴 간격에만 배치될 수 있는 반면, 분할 진행이 가능한 작업은 배치되는 간격의 제약 여부와 관계없이 짧게 분할되어 배치될 수 있다. 작업이 가진 선후 관계는 해당 작업이 배치될 수 있는 근무 시간 중의 위치를 결정한다. 작업의 배치에 있어 선행 작업은 후행 작업보다 반드시 먼저 진행되어야 하고 후행 작업은 먼저 배치된 선행 작업보다 이후의 시점에 배치되어야 한다.

이렇게 작업 별로 적용되는 배치 제약의 유형이 다르므로 업무 재구조화 과정에서 모든 작업에 대해 필요 없는 배치 제약까지 일괄적으로 적용하여 재구조화 대안을 생성하는 것은 비효율적인 방법이다. 또한 이후 순차적 배치 알고리즘에서 사용할 정수 계획법은 문제의 크기가 커질수록 계산 복잡성이 급격하게 증가하기 때문에 모든 작업에 대한 일괄적인 배치는 현실적이지 않다. 본 단락에서는 이후에 진행될 업무 재구조화 과정에서 배치 자유도가 작은 작업에 대해서만 정수 계획 문제를 만드는 것으로 실행 가능한 대안을 빠른 시간 안에 만들 것이다. 또한 그에 이어 제약 조건의 유형에 따라 다른 알고리즘을 통해 작업을 배치하기 위해 작업의 유형을 분류할 것이다.

이후에 진행할 순차적 배치 알고리즘을 위하여, 업무를 구성하는 작업들을 앞의 두 특징을 기준으로 그림 2.6과 같이 필수작업, 유동작업, 그리고 부수작업으로 분류했다. 필수작업은 분할 진행할 수 없어 배치 시 반드시 일정 길이 이상의 연속된 시간을 동시에 점유하게 되는 작업 유형으로, 설비에 대한 조작이나 원료의 투입과 같은 작업들이다. 따라서 필수작업은 배치 시 작업 사이의 간격 크기에 따른 배치 제약을 적용 받는 유일한 유형이며, 선후 관계를 가진 작업과 가지지 않는 작업을 모두 포함하고 있다.

필수작업의 분류 뒤 남은 작업은 모두, 분할 진행 가능한 작업들이다. 이들은 작업 배치 시에 작업 소요시간 만큼의 연속된 근무시간을 함께 점유할

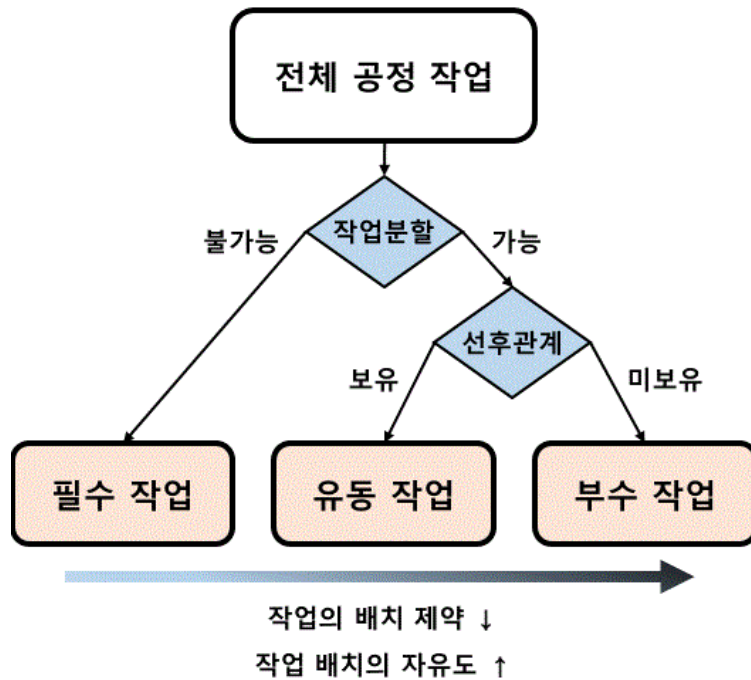


그림 2.6 작업 유형 분류 체계

필요가 없기 때문에 필수작업과 적용 받는 제약이 다르다. 유동작업은 이중에서도 다른 작업과의 선후 관계를 최소한 하나 가지고 있는 작업들이다. 선후 관계를 가지면서도 분할 진행 가능하기 때문에 작업의 시작, 종료 시점이 유동적으로 변할 수 있어, 작업 간 간격의 고정과 같은, 그를 기준으로 한 작업 간 간격 크기에 대한 제약조건을 정의하기 힘들다.

따라서 본 연구에서는 유동작업들이 간격 크기에 대한 제약이 없는 단순 선후 관계만을 가진다고 가정했다. 유동작업의 주요 예시로는 화학 공정에서 안정화 단계에 이른 설비 내용물에 대한 모니터링이 있다. 이들은 공정의 흐름에서 다른 작업들과 내용상의 선후 관계를 가지면서도 작업자가 계속 설비 앞에 있을 필요 없이 유연하게 배치될 수 있다.

부수작업은 분할 진행 가능하면서도 선후 관계를 갖지 않는 작업으로 필수작업과 유동작업이 배치되고 남은 나머지 모든 작업들이다. 이들은 작업 배치에 적용되는 어떠한 제약 조건도 없기 때문에, 업무 재구조화 대안에서 각 작업이 근무시간 중에서 점유하는 시간의 총량 만이 중요하다. 바닥에 대한 청소나 서류 작성 작업과 같이 공정 내용에서 벗어난 작업들이 부수작업 유형에 속한다.

이 세 가지 작업 유형들은 그림 2.6 아래 부분의 화살표와 같이, 필수작업, 유동작업, 부수작업의 순서대로 배치에 적용되는 제약 조건이 줄어들고 그 작업 배치의 자유도가 증가한다. 본 연구에서는 작업 유형들의 배치 자유도 차이를 활용하여 작업 배치의 복잡성을 줄이기 위한 순차적 배치 알고리즘을 제안할 것이다.

2.3 작업 배치 단위 정립

본 단락은 개별 작업의 배치 제약에 따라 분류된 업무 구성 작업의 유형 분류 결과를 바탕으로 업무 재구조화에서 사용될 작업의 배치 단위와 그들 사이의 상호 배치 관계를 새로 정의한다. 작업의 배치는 작업을 다른 공정의 작업 사이에 존재하는 간격에 집어넣는 것이다. 따라서 작업 배치를 진행하여 업무 재구조화 대안을 생성하기 위해서는 작업이 배치될 간격이 발생하는 위치에 대해 분석해야 한다.

업무에서 발생 가능한 간격은 그 발생 위치에 따라 작업 내부 간격과 작업 외부의 간격으로 구분할 수 있다. 내부에서 발생하는 간격은 각 작업의 분할 진행 가능성에 따라 단일 작업이 분할 되는 것을 통해 발생한다. 따라서 분할 진행 가능한 작업은 배치 과정에서 내부 간격 발생의 가능성을 내재하고 있다. 작업 외부의 간격은 서로 다른 작업들 사이의 경계에서 발생하는 간격이다. 이는 작업 간 간격의 크기가 0으로 고정되어 서로 연속해서 진행되어야 하는

작업 사이에서는 발생할 수 없기 때문에, 관찰된 개별 작업 단위에서의 경계와는 차이가 있다. 따라서 업무 재구조화에서 작업 배치의 의사결정은 관찰 데이터를 구성하는 작업의 관찰 단위에서 일어나는 것이 아니다.

작업 배치가 용이하도록 간격이 발생할 수 있는 위치를 통제하기 위해서, 작업 배치의 의사 결정은 내부에서 간격이 발생할 수 없고 전체가 항상 연속하여 배치되어야 하는 작업, 즉 배치 자유도가 작은 필수작업을 중심으로 이루어져야 한다. 업무 관찰 데이터에서의 개별 작업들은 작업 배치의 결과일 뿐이라는 점에서 이와는 차이가 있다. 유동, 부수작업은 분할 될 수 있으므로, 앞서 분류된 작업 유형 중 필수작업만이 이런 작업 배치의 기준이 될 수 있다. 이를 위해서 내부에 간격이 발생할 수 없는 작업 유형인 필수작업을 주변 작업과의 연속 진행 필요성에 따라 작업 배치 단위로 재구성할 필요가 있다(그림 2.7). 본 연구에서는 이렇게 구성된 작업 배치의 새로운 단위를 작업 모듈(Module)이라 정의한다.

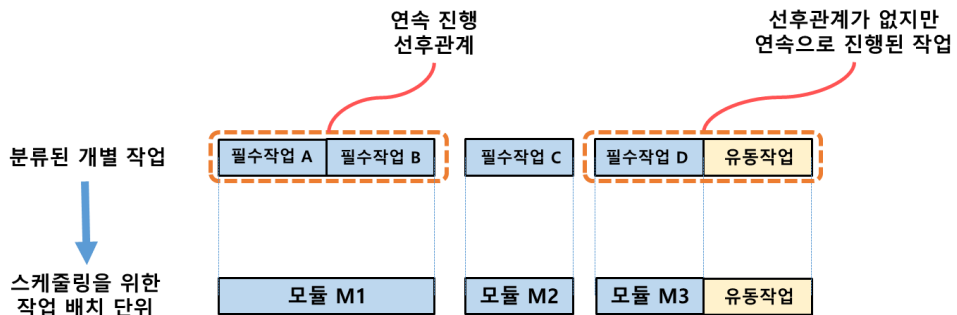


그림 2.7 작업 배치 단위, 작업 모듈의 구성

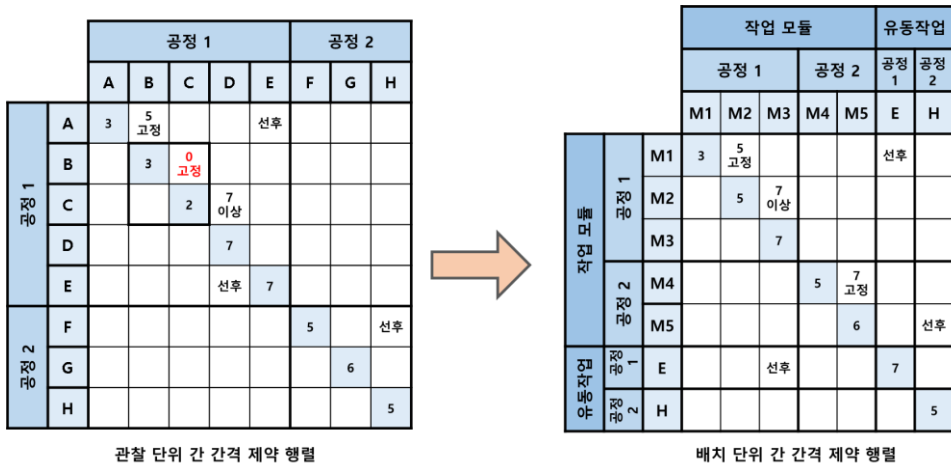


그림 2.8 작업 배치 단위 간 간격 제약 행렬의 예시

업무 재구조화에 필요한 작업 간의 선후 관계, 간격 제약 역시 재정립되어야 한다. 그림 2.8은 관찰 단위 사이의 간격 관계로부터 모듈 및 유동작업 사이에 존재하는 관계를 나타내는 간격 제약 행렬로의 변환을 나타낸 그림이다. 앞의 그림 2.5와 같이 배치 단위 간 간격 제약 행렬은 행과 열이 각각 선행, 후행 작업을 나타내고 그 비 대각요소는 관계의 유형을 나타낸다. 행렬의 대각요소는 해당 단위의 소요시간을 나타낸다. 사이의 간격이 0으로 고정된 필수작업 B와 C가 M2로 통합된 것을 확인할 수 있다.

제 3 장 업무 재구조화 휴리스틱 알고리즘

본 단락은 재정립된 작업 배치 단위를 바탕으로 실행 가능한 업무 재구조화 대안을 생성하기 위한 작업 배치 알고리즘을 제안한다. 업무 재구조화에서의 작업 배치는 작업자 근무시간 중의 특정 구간을 각 작업들의 진행에 할당하는 것이다. 공정에 투입되는 작업자의 수가 한정되어 있으므로, 한 작업자가 동시에 여러 작업을 진행할 수 없다는 가정 하에, 업무 대안에서 동시간에 진행되는 작업의 숫자는 제한된다. 따라서 작업자 자원이 각 시점에 다른 작업에 의해 점유되는지 여부가 개별 작업의 배치에 결정적인 영향을 미친다. 이진 변수(Binary Variable)인 작업자 자원의 점유 여부를 다루기 위해 알고리즘에 정수 계획법을 도입했다.

3.1 작업 배치의 표현

정수 계획법을 통한 작업 배치를 위해 배치에서 다룰 시점(Time)을 일정한 시간 간격(1분)으로 구분된 근무시간으로 정의한다. 또한 업무 재구조화에서 동시간대에 진행되는 작업의 수를 통제하기 위해서 식(3.1)과 같이 총 근무시간 T 안의 시점 t 에서의 작업 모듈 k 의 진행 여부를 표현하는 이진 변수 $x(t, k)$ 를 정의한다. $x(t, k) = 1$, 즉 시점 t 에 k 가 배치되는 사건은 해당 시간 동안 작업자가 다른 작업을 수행할 수 없음을 의미함과 함께, 해당 작업의 수행에 필요한 시간 중 1분이 완료되는 것을 의미한다. 따라서 변수 x 에 대한 제약조건을 작성하는 것을 통해 정수 계획법에서 동시에 진행되는 작업의 수와 개별 작업에 투입되는 작업자 근무시간을 통제할 수 있다.

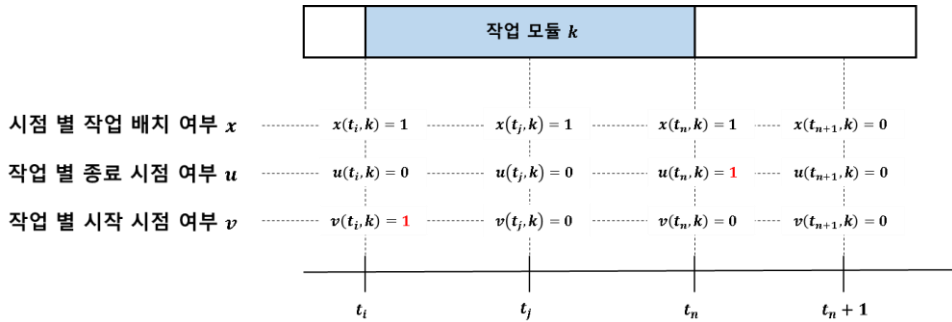


그림 3.1 이진 변수 x , u , v 를 활용한 작업 배치 상황 표현

$$x(t, k) = \begin{cases} 1, & \text{if task } k \text{ was placed at time } t \\ 0, & \text{otherwise} \end{cases} \quad (3.1)$$

업무 재구조화 과정에서 공정 내 작업 사이의 작업 진행 선후 관계를 반영하여 작업을 배치하기 위해서, 개별 작업이 시행되는 시점에 대한 정보가 필요하다. 그를 위해 변수 x 의 정의와 비슷하게, t 시점의 작업 모듈 k 의 종료 시점 여부 및 시작 시점 여부 $u(t, k), v(t, k)$ 를 식(3.2), 식(3.3)과 같이 정의한다.

$$u(t, k) = \begin{cases} 1, & \text{if task } k \text{ finishes at time } t \\ 0, & \text{otherwise} \end{cases} \quad (3.2)$$

$$v(t, k) = \begin{cases} 1, & \text{if task } k \text{ starts at time } t \\ 0, & \text{otherwise} \end{cases} \quad (3.3)$$

이렇게 정의된 이진 변수 x, u, v 의 값은 그림 3.1과 같이 작업자 근무시간 중 각 시점의 개별 작업의 배치 상황과 일대일 대응한다. 따라서 이진 변수들의 값을 결정하는 정수 계획 문제는 업무 재구조화 대안을 생성하는 것과 동일한 문제이다.

3.2 작업 유형별 배치 제약과 순차적 배치

심플렉스(Simplex Method)라는 일반적인 해법이 존재하는 선형 계획 문제(Linear Programming)와는 달리, 정수 계획 문제는 정수 제약에 따른 NP-hard 문제로서 최적해를 탐색하는데 많은 연산이 요구된다. 변수 및 제약 조건의 수가 증가하여 문제의 규모가 커지면 그 해결에 필요한 연산 시간이 기하급수적으로 증가하기 때문에, 업무 재구조화를 구성하는 모든 시점에 대한 모든 작업들의 이진 변수들을 한번에 계산하는 것은 현실적인 방법이 아니다. 본 연구에서는 여러 작업 유형 중에서도 작업 배치의 제약이 큰 필수작업 유형만을 대상으로 정수 계획법을 진행하는 것을 통해 실행 가능한 대안에 대해 빠르게 탐색하는 순차적 배치 알고리즘을 제안하고자 한다.

3.2.1 작업 유형에 따른 배치 제약의 차이

제2장에서 공정들을 구성하는 개별 작업들은 작업 배치의 자유도를 결정하는 작업 분할 가능성, 그리고 작업 선후 관계 여부에 따라 필수, 유동, 부수의 세 유형으로 분류되었다. 유동작업이나 부수작업과는 달리 분할 진행할 수 없는 필수작업, 작업 모듈은 진행에 필요한 총 소요시간만큼의 연속된 구간을 동시에 점유해야 한다. 작업 모듈의 집합 M 의 원소인 작업 모듈 k 는 그 소요시간 $d(k)$ 동안 연속하여 진행되고 이에 따라 식(3.4)와 같이 시작 시점과 종료 시점이 정확하게 $d(k)$ 만큼의 차이를 가져야한다. 유동작업과 부수작업 역시 작업 배치의 선후 관계 유무에 따라 분류되었으므로, 사이에 선후 관계 제약의 적용에 따른 배치 자유도 차이가 존재한다.

$$\sum_{t=1}^T (t \cdot u(t, k)) - \sum_{t=1}^T (t \cdot v(t, k)) = d(k) \quad \forall k \in M \quad (3.4)$$

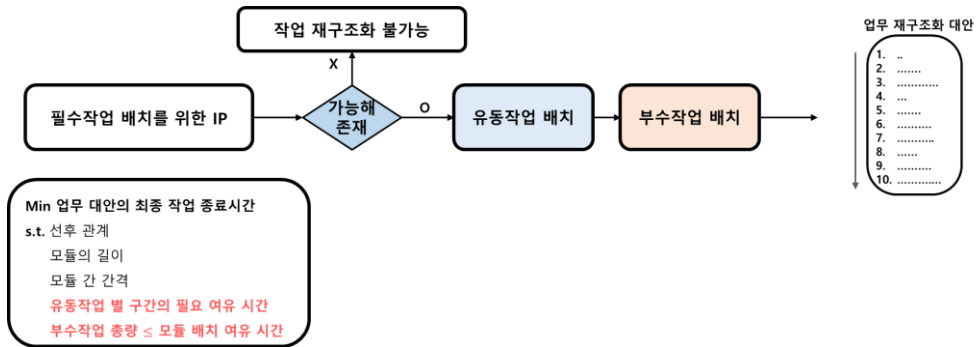


그림 3.2 작업 유형별 순차적 배치 휴리스틱 알고리즘

3.2.2 순차적 배치 알고리즘

그림 3.2와 같이 본 연구에서 제안하는 순차적 배치 알고리즘은 개별 작업들을 작업 배치의 자유도에 따라 필수작업, 유동작업, 부수작업의 순서로 근무시간의 각 시점에 배치한다. 이 과정에서 필수작업의 배치에 필요한 제약조건으로 구성된 정수 계획 문제를 먼저 진행한다. 필수작업의 배치 문제에 대한 가능해가 존재하지 않는다면, 전체 업무 재구조화 대안은 존재할 수 없다. 반대로 필수작업 배치에 대한 가능해의 존재는 전체 업무 재구조화 문제의 가능해가 존재한다는 것을 의미한다.

필수작업 배치에 대한 최적해를 발견한 뒤, 배치된 필수작업들 사이에 유동작업을 배치한다. 각 유동작업의 배치 가능 시점은 그와 선후 관계를 갖는 필수작업 진행 시간에 따라 결정되고 이에 따라 필수작업의 배치는 유동작업이 선후 관계를 지켜 배치될 수 있도록 진행되어야 하기 때문에, 아무 관계가 없는 부수작업과 같이, 양 쪽을 완전히 떼어 놓고 생각하는 것은 불가능하다. 본 알고리즘은 유동작업의 배치에 필요한 간격을 필수작업의 배치 제약에 미리 반영하는 것을 통해, 필수작업과 유동작업의 배치를 시간적으로 분리하면서도 실행 가능한 대안을 생성한다. 이 제약에 따라, 유동작업 배치 시점에서는 모든

유동작업의 배치 가능성이 보장된다.

부수작업은 선후 관계도 없으며 분할 진행할 수 있으므로 필수작업과 유동작업의 배치 이후 비어 있는 간격의 양 만이 배치 가능성을 결정한다. 따라서 전체 근무시간에서 부수작업의 배치 가능성은 필수작업과 유동작업의 배치 결과에 대한 영향을 받지 않는다. 반대로 필수작업과 유동작업의 근무시간 내 배치 가능성은 부수작업이 배치된 위치에 따라 변동할 수 있다. 이에 따라 필수작업과 유동작업이 배치되고 남은 근무 시간에 대하여 부수작업을 배치하는 것을 통해 최종 업무 재구조화 대안을 생성한다.

3.3 필수작업 배치를 위한 정수 계획 문제

필수작업의 배치는 표 3.1과 같이 제2장에서 재정립된 개별 작업 배치 단위들에 대한 정보를 기반으로 앞서 설명한 이진 변수 x, u, v 의 값을 결정하는 문제이다. 총 근무시간 T 에 대하여, 근무시간을 구성하는 각 시점은 H 의 원소이다. 또한 총 작업 모듈의 수 m , 총 유동작업의 수 n , 총 부수작업의 수 o 에 대하여, S 는 재구조화의 대상인 두 업무를 구성하는 각 작업 배치 단위의 집합이다. M 은 작업 모듈만으로 구성된 집합이다.

각 작업 배치 단위의 소요시간과 작업 간 선후 관계는 d 와 p, f, l 을 통해 표현된다. 식(3.5)와 같이 p 는 서로 다른 두 작업에 대하여, 한 작업 단위가 먼저 진행되어야 하는 경우에만 1의 값을 갖는다. f 는 식(3.6)과 같이 두 작업 사이의 선후 관계가 고정된 간격을 갖는 경우에 1의 값을 갖는다. l 은 작업 간 간격이 제약된 선후 관계에서 제약의 기준이 되는 간격의 크기를 나타낸다.

$$p(i, j) = \begin{cases} 1, & \text{if task } i \text{ must precede task } j \\ 0, & \text{otherwise} \end{cases} \quad (3.5)$$

$$f(i, j) = \begin{cases} 1, & \text{if interval between task } i \text{ and } j \text{ is fixed} \\ 0, & \text{otherwise} \end{cases} \quad (3.6)$$

표 3.1: 작업 배치 과정에 필요한 파라미터

파라미터	내용
H	분 단위로 분할된 작업자 근무시간 중 시점의 집합 $H = \{1, 2, \dots, T\}$
S	재구조화 대상 업무를 구성하는 작업 모듈과 유동작업, 부수작업의 집합 $S = \{1, 2, \dots, m, m + 1, \dots, m + n, \dots, m + n + o\}$
M	양 업무의 작업 모듈의 집합 $M = \{1, 2, \dots, m\}$
$d(k)$	재구조화 대상 업무를 구성하는 작업 k 의 총 소요시간
$p(i, j)$	작업 모듈 및 유동작업 사이의 선후 관계 여부, binary
$f(i, j)$	작업 모듈 사이의 고정된 선후 관계 여부, binary
$l(i, j)$	간격이 고정된 작업 모듈 i 와 j 사이의 간격 크기

3.3.1 필수작업 배치의 제약조건

근무시간에 배치되는 필수작업들이 만족시켜야 하는 조건은 각자 각 시점에서의 작업 배치 상태를 나타내는 이진 변수 x, u, v 에 대한 제약식을 통해 표현할 수 있다. 배치된 업무에 투입되는 작업자 자원 수의 제한으로 인해 동시에 진행되는 작업의 수가 제한된다. 이는 특정 시점에서 진행되는 모든 작업 진행 여부의 합이 1보다 작거나 같아야 하는 식(3.7)처럼 나타난다. 또한 작업의 온전한 진행을 위해서, 전체 근무시간 동안에 작업 별 소요 시간의 길이만큼의 시점이 각 작업의 진행에 할당되어야 한다(식(3.8)).

$$\sum_{k=1}^m x(t, k) \leq 1 \quad \forall t \in H \quad (3.7)$$

$$\sum_{t=1}^T x(t, k) = d(k) \quad \forall k \in M \quad (3.8)$$

필수작업 및 그로 구성된 작업 모듈들은 분할 배치되지 않고 항상 연속해서 진행되기 때문에 식(3.9), 식(3.10)과 같이 그 시작과 종료 사건이 전체 근무시간 중에서 각각 단 한 시점에서만 발생해야 한다. 이 제약조건은 식(3.4)와 함께, 여러 작업 유형 중에서 필수작업의 배치에만 존재하는 제약조건이다.

$$\sum_{t=1}^T u(t, k) \leq 1 \quad \forall k \in M \quad (3.9)$$

$$\sum_{t=1}^T v(t, k) \leq 1 \quad \forall k \in M \quad (3.10)$$

마지막으로 작업 배치 단위 사이에 존재하는 선후 관계와, 그 사이 간격에 대한 제약식이 필요하다. 식(3.11)은 작업 간에 존재하는 선후 관계가 존재하는 작업 모듈 i, j 에 대하여, 선행 작업의 종료 시점은 후행 작업의 시작 시점보다 빨라야 한다는 제약조건을 나타낸다. 작업 단위 간에 필요한 간격에 대한 제약은 식(3.12)와 같이 선후 관계를 구성하는 작업 중 선행 작업의 종료 시간과 후행 작업의 종료 시간 사이의 차이와 필요 간격 크기에 대한 식으로 나타낼 수 있다.

$$\sum_{t=1}^T t \cdot u(t, i) \cdot p(i, j) \leq \sum_{t=1}^T t \cdot v(t, j) \cdot p(i, j) \quad \forall i, j \in M \quad (3.11)$$

$$f(i, j) = \sum_{t=1}^T t \cdot v(t, j) \cdot f(i, j) \quad \forall i, j \in M \quad (3.12)$$

3.3.2 유동작업 배치의 제약조건화

순차적 배치의 필수작업 배치 단계에서 이후에 배치될 유동작업의 자리를 미리 마련하기 위해서, 유동작업이 가진 선후 관계를 정수 계획 문제의

제약조건으로 표현하는 것이 필요하다. 근무시간 내 각 시점에서의 유동작업의 배치 가능성을 결정하는 것은 그와 선후 관계를 가진 필수작업들의 배치 위치와 그 사이의 간격의 크기이므로, 정수 계획에 의해 결정된 작업 모듈들의 배치 순서가 제약 조건에 반영될 수 있어야 한다. 이를 위해 식(3.13)과 같이 임의의 배치된 두 작업 모듈 사이의 선후 관계를 의미하는 이진 변수 \mathbf{b} 를 $\mathbf{x}, \mathbf{u}, \mathbf{v}$ 에 이어 추가로 도입했다.

$$\mathbf{b}(i, j) = \begin{cases} \mathbf{1}, & \text{if task } i \text{ precedes task } j \\ \mathbf{0}, & \text{otherwise} \end{cases} \quad (3.13)$$

\mathbf{b} 는 정수 계획에 따른 필수작업 배치 결과의 작업 순서를 나타내기 위한 변수로서, 작업 배치의 제약 조건이자 상수로 고정되어 있는 선후 관계 \mathbf{p} 와 차이가 있다. \mathbf{b} 가 배치 결과의 작업 순서를 나타내도록 하기 위해 몇 가지 제약식이 더 필요하다. 식(3.14)와 식(3.15)은 서로 다른 모듈 i, j 에 대해서 $\mathbf{b}(i, j)$ 와 $\mathbf{b}(j, i)$ 중 하나 만 1의 값을 가지도록 만드는 제약조건이다. 이어 식(3.16)을 통하여 $\mathbf{b}(i, j)$ 가 1의 값을 가질 때만 작업 i 가 j 보다 먼저 종료될 수 있도록, 그리고 반대로 작업 i 가 j 보다 선행할 때만 $\mathbf{b}(i, j)$ 가 1의 값을 가질 수 있도록 만들었다.

$$\mathbf{b}(i, i) = \mathbf{0} \quad \forall i \in M \quad (3.14)$$

$$\mathbf{b}(i, j) + \mathbf{b}(j, i) = \mathbf{1} \quad \forall i, j \in M, i \neq j \quad (3.15)$$

$$\sum_{t=1}^T (t \cdot (\mathbf{u}(t, i) - \mathbf{u}(t, j)) + T \cdot \mathbf{b}(i, j)) \geq \mathbf{0} \quad \forall i, j \in M, i \neq j \quad (3.16)$$

\mathbf{b} 를 활용하면 특정 작업 모듈보다 먼저 배치되거나, 나중에 배치된 작업들을 찾아낼 수 있다. 식(3.17)는 작업 모듈 j 보다 먼저 위치하는 작업들과 선후 관계를 가져서 j 보다 먼저 배치되어야 하는 유동작업들의 총 소요시간보다 j 앞에 존재하는 빈 근무 시간의 양이 많아야 한다는 제약이다.

식(3.18)은 반대로 어떤 작업 i 보다 뒤에 존재해야 하는 간격에 대한 제약식이다. 이 제약들을 모두 만족시키는 필수작업 배치 대안은 이후에 모든 유동작업이 배치될 수 있는 여유 간격을 가지고 있다.

$$\begin{aligned} \sum_{t=1}^T t \cdot v(t, j) - \sum_{k=1}^m b(k, j) \cdot d(k) \geq \\ \sum_{i=m+1}^{m+n} p(i, j) \cdot d(i) + \sum_{i=m+1}^{m+n} \sum_{k=1}^m p(i, k) \cdot b(k, j) \cdot d(i) \quad \forall j \in M \end{aligned} \quad (3.17)$$

$$\begin{aligned} T - \sum_{t=1}^T t \cdot u(t, i) - \sum_{k=1}^m b(i, k) \cdot d(k) \geq \\ \sum_{j=m+1}^{m+n} p(i, j) \cdot d(j) + \sum_{j=m+1}^{m+n} \sum_{k=1}^m p(k, j) \cdot b(i, k) \cdot d(j) \quad \forall i \in M \end{aligned} \quad (3.18)$$

3.3.3 목적함수

분할되어 근무시간의 어느 시점이든 자유롭게 배치될 수 있는 부수작업은 필수, 유동작업 배치의 배치 변화에 따른 전체 업무 시간 변화에 대한 버퍼 역할을 수행할 수 있다. 또한 정수 계획의 특성상 천문학적인 양의 연산이 필요하기 때문에 업무 재구조화에서 우선시되어야 하는 것은 최적해 탐색 그 자체보다는, 모든 제약 조건을 만족시키는 실행 가능한 대안, 즉 가능해를 찾아내는 것이다.

그럼에도 불구하고 정수 계획법의 구성 요소로서, 그리고 여러 대안 중 하나의 대안을 선택하기 위한 기준으로서 목적함수가 필요하다. 순차적 배치 알고리즘에서 필수작업을 배치하는 정수 계획 문제의 최적해를 발견한 뒤에도 유동작업과 부수작업의 배치가 남아있기 때문에, 이후에 배치될 작업들과 최종 재구조화 대안을 고려하여 목적 함수를 선택해야 한다.

본 연구가 대상으로 하는 설비 중심 공정에는 설비 고장과 같은 작업 진행 중의 사고, 화학 반응 자체에 내재된 개별 작업 시간의 변동 등 개별 작업의

진행에 여러 불확실성이 존재한다. 재구조화 과정에서 사이 간격없이 빠르게 배치되는 업무 대안에서 발생하는 작업 지연은, 전체 업무의 지연을 불러올 수밖에 없다. 따라서 업무 시간이 작업자 근무시간을 초과하는 상황을 방지하기 위해서, 최대한 빠른 시간 안에 업무를 마치는 것을 통해 업무 지연에 대한 버퍼를 마련하는 것이 유효하다.

이를 위해 식(3.19)과 같이 업무에서 마지막으로 배치된 필수작업의 종료시간 e 를 목적 함수로 설정했다. 유동작업의 위치는 필수작업의 배치 단계에서 이미 고려되는 것이고, 부수작업을 필수작업과 유동작업이 배치되고 남은 빈 근무시간에 앞에서부터 배치하는 것으로 가장 빠르게 마치는 업무 대안을 만들 수 있다.

$$\sum_{t=1}^T (t * x(t, k)) \leq e \quad \forall k \in M \quad (3.19)$$

3.4 유동작업과 부수작업의 배치

필수작업 배치를 위한 정수 계획 문제의 최적해가 주어진 이후, 최종 업무 재구조화 대안의 생성을 위하여 필수작업이 배치되고 남은 근무시간에 유동작업과 부수작업을 배치해야 한다. 필수작업의 배치과정에서 유동작업의 배치에 필요한 간격이 제약조건으로 반영되었기 때문에 이들이 배치될 위치는 근무시간 내에 마련되어 있다. 따라서 필수작업 배치 뒤 남은 근무시간에서 유동작업의 각 부분이 분할되어 배치될 위치는 필수작업의 배치 단계에서 이미 결정되어 있다.

유동작업은 분할 진행할 수 있으면서 선후 관계를 가지고 있는 작업 유형이다. 유동작업의 배치를 위해서 우선 배치된 필수작업들의 위치에 대하여 각 유동작업의 배치될 수 있는 근무시간의 범위를 찾아야 한다. 각 유동작업보다 선행해야 하는 필수작업의 종료 시간이 그 배치 가능 구간의

시작 시점이 되고, 후행해야 하는 작업의 시작 시점이 배치 가능 구간의 종료 시점이 된다.

그에 이어서, 비어 있는 근무시간 중 빠른 시점부터 살피며 유동작업이 배치될 위치를 결정한다. 앞서 파악한 유동작업의 배치 가능 구간을 사용하여, 각 시점에서 배치 가능한 유동작업을 찾을 수 있다. 그 중에서 매 시점에서 배치할 작업이 남아있으면서도 가장 먼저 종료되어야 하는 유동작업을 이 시점의 근무시간에 배치하는 것을 반복하는 것으로 모든 유동작업을 배치할 수 있다.

유동작업을 배치하고도 남은 근무시간에 대해서, 부수작업을 배치한다. 부수작업은 배치에 고려해야하는 제약조건이 없기 때문에, 필수작업 배치 단계에서의 목적함수와 같이 최종 업무 대안이 가능한 한 빠른 시점에서 종료될 수 있도록 이른 시점의 여유 간격부터 차례로 배치한다.

제 4 장 사례연구 : UDCA 생산공정의 재구조화

4.1 사례연구 개요

본 연구는 제약회사의 UDCA(Ursodeoxycholic Acid) 생산공정의 관찰 데이터를 활용하여 사례연구를 진행했다. 이름의 접두사에서 알 수 있듯 UDCA는 곰의 쓸개인 웅담에서 발견되는 화학물질로서, 담즙의 분비를 촉진시키면서 간의 대사활동을 증진하는 효과가 있어 다양한 의약품의 원료로 사용되고 있다. 본 연구에서 대상으로 한 UDCA 생산공정은 수급이 힘든 웅담이 아닌 닭이나 돼지의 쓸개에서 획득할 수 있는 CDCA(Chenodeoxycholic Acid)를 원료로 UDCA를 생산한다.

CDCA는 그 수급이 원활한 대신 그림 4.1과 같이, UDCA로 완전히 합성되기까지 전처리 공정, 반응 공정 등 여러 세부 공정을 거둬하며 여러 중간

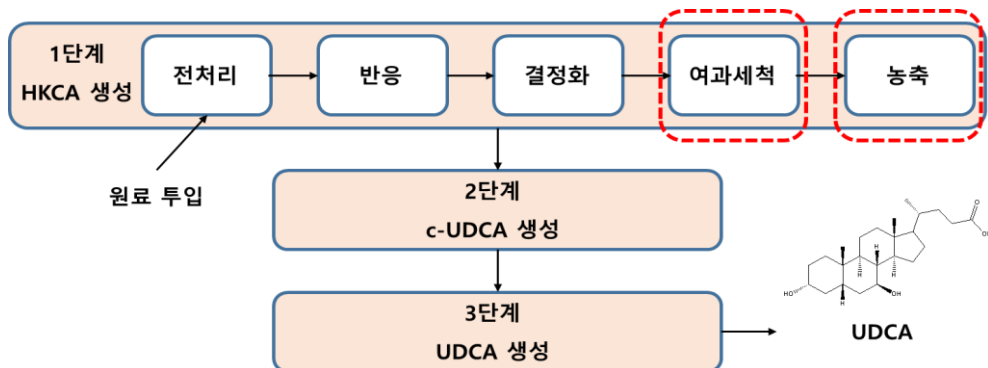


그림 4.1 UDCA 생산의 세부 공정

단계 상태를 순차적으로 거친다. 각 세부 공정들은 원료가 위치한 중간 물질 상태를 기준으로 크게 세 단계로 분류되며, 개별 공정은 주경로가 설비 작업에 있는 설비중심공정의 전형이다. 각 세부 공정에는 설비 및 해당 세부 공정에 할당된 작업자 1~2명이 배정되고 공정 별로 고정된 업무 루틴에 의해 진행된다.

사례연구는 UDCA 생산의 여러 공정 중, 대기시간 비중이 크면서 동 시간대(20:00 ~ 08:00)에 진행되고 있어 재구조화 필요성과 실행 가능성이 큰 HKCA 농축 공정과 HKCA 여과-세척 공정을 대상으로 한다. 이를 위해 실제 공정이 진행되는 현장에 배치된 작업자에 대한 추적 관찰을 진행하는 것으로 개별 작업의 진행 시간과 내용에 대한 데이터를 획득했다.

4.2 작업 유형 분류 및 작업 관계 모델링

본격적인 재구조화 과정에 앞서 관찰된 순서와 내용을 기반으로 개별 작업의 분할 진행 가능성과 선후 관계, 연속 진행 필요성을 분석했다. 그림 4.2는 실제 작업 관찰 데이터 및 분석 결과의 일부이다.

Btime	Etime	세부내용	작업시간	조작 확인	필수 관찰	대기 관찰	설비 세척	정소	운반	원료 투입	포장	측정	서류 작업	이동	환복	대기	교대 대기	지원	휴식	기타	선후 관계	분할 가능성	연속 진행	
19:35	19:38	패역 별보 점검	3	1																		○	X	
19:38	19:40	세척역 역송	2	1																		○	X	
19:40	19:43	역송 모니터링	3		1																	○	X	
19:43	19:48	역송 조절	5	1																		○	X	
19:48	19:50	반응기 교반 조정	2	1																		○	X	
19:50	19:53	세척역 역송 종료	3	1																		○	X	
19:53	19:56	질소가입 및 여과 진행	3	1																		○	X	
19:56	19:57	서류작업	1										1									X	○	
19:57	19:59	패역 탱크 확인	2	1																		○	X	
19:59	20:03	서류작업	4										1									X	○	
20:03	20:06	서류작업	3										1									X	○	
20:06	20:12	서류작업	6										1									X	○	
20:12	20:14	반응기 확인	2	1																		○	X	
20:24	20:28	복도 바닥 청소	4					1														X	○	
20:28	20:57	세척역 배출되는 동안 대기	29			1																○	○	
21:08	21:11	패역탱크 확인	3	1																		○	○	

그림 4.2 여과-세척 공정의 작업 분석 예시

업무 재구조화에 필요한 개별 작업의 특성은 해당 작업의 시작 및 종료 시간, 작업 세부내용, 선후 관계의 유무, 그리고 분할 진행 가능성으로 구성된다. 서류작성이나 청소와 같이 공정 내용, 설비와 직접 관련이 없는 작업은 배치에 있어 다른 작업들과 선후 관계가 없었지만 그를 제외한 대부분의 작업은 업무 내용에 있어 앞뒤 작업들과 진행의 선후 관계를 가졌다. 이는 공정 내용과 관련된 모든 작업은 설비 내부에서 원료가 반응하여 온도나 성분의 변화를 일으키기 위한 순차적 단계이고 이에 따라 그 시작 시점에서 설비와 내부 반응물이 특정한 상태로 준비될 것을 요구하기 때문이다.

앞서 언급한 서류 작성과 청소 작업 외의 일부 모니터링 작업 역시 분할 진행 가능성을 가진다. 공정에서의 모니터링은 설비 조작 작업과 같은 설비나 원료의 직접적인 취급과 달리 일정시간 동안, 혹은 일정 시간 주기로 설비 내용물의 상태를 감시하는 작업이다. 그 중에서도 반응이 마무리되어 설비 내부가 안정화된 이후의 모니터링은 격렬하고 위험한 반응의 모니터링과 달리 작업자가 중간에 잠시 자리를 비울 수 있다. 실제로 작업자들은 이런 여유로운 모니터링 시간을 이용해 틈틈이 휴식을 취하거나 부차적인 업무를 수행하고 있었으므로 이런 부류의 모니터링 작업에 대해서는 분할되어 진행될 수 있는 것을 확인할 수 있었다.

표 4.1은 개별 작업의 분할 진행 가능성과 선후 관계를 기반으로 한, 1단계 HKCA 농축 공정과 여과-세척 공정 구성 작업들에 대한 작업 유형 분류 결과이다. 근무 중 보장된 90분의 휴식(00:00 ~ 01:00, 06:00~06:30)을 제외한 실질적인 근무 시간 630분에 대해서, 재구조화 대상인 두 공정 모두 작업자의 효율이 50%를 밑도는 것을 확인할 수 있었다. 이어서 사이에 대기시간 없이 연속 진행해야 하는, 간격이 0으로 고정된 필수작업들을 하나의 작업 모듈로 병합했다. 그림 4.3은 1단계 HKCA 농축 공정을 구성하는 작업 모듈과 유동작업들에 대한 배치 간격 제약 행렬이다.

표 4.1: 농축 공정과 여과-세척 공정의 작업 유형 분류 결과

공정	작업모듈	유동작업	부수작업	작업시간	효율
농축	32 회	5 회	11 회	300 분	47.62%
여과세척	31 회	11 회	16 회	304 분	48.09%

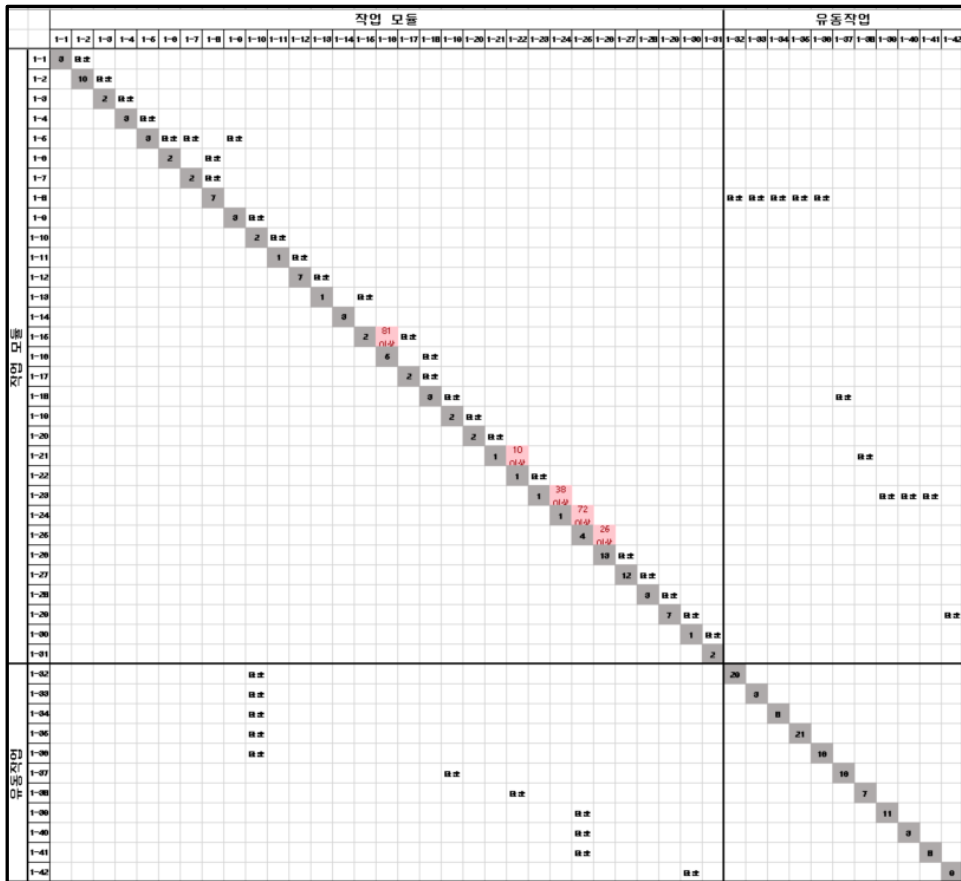


그림 4.3 농축 공정의 배치 단위 간 간격 제약 행렬

```

try:
    m = Model("Scheduling")
    m.setParam("MIPGap", 0.1)
    m.setParam("TimeLimit", 15000)

    x = m.addVars(times, modules, name='x', vtype=GRB.BINARY) # 만일 time에 job를 하는지 유무
    u = m.addVars(times, modules, name='u', vtype=GRB.BINARY) # job의 종료시점 유무
    v = m.addVars(times, modules, name='v', vtype=GRB.BINARY) # job 시작시점 유무
    b = m.addVars(modules, modules, name='b', vtype=GRB.BINARY) # 배치된 모듈의 순서 관계
    maxt = m.addVar(name='maxt') # 가장 나중에 마치는 모듈 종료시간

    m.setObjective( x.sum() + maxt ) # 목적함수, x가 최대한 작아지게

    m.addConstrs((quicksum(t*x[t, k] for k in modules) <= maxt for t in times), name='maxt=argmax')
    m.addConstrs((sum(x[t, k] for k in modules) <= 1 for t in times), name='one.body') # 한사람은 동시에 한가지 일까지만
    m.addConstrs((sum(v[t, k] for t in times) == 1 for k in modules), name='start.once')
    m.addConstrs((sum(u[t, k] for t in times) == 1 for k in modules), name='finish.once')

    for k in modules:
        m.addConstrs((u[t, k] == 0 for t in range(int(dur[k]))), name='min.duration<u[k]')
        m.addConstrs((v[t, k] == 0 for t in range(int(T - dur[k]), int(T))), name='T-min.duration>v[k]')
    for k in modules:
        m.addConstr((quicksum(t * v[t, k] for t in times) + dur[k]
                    == quicksum(t * u[t, k] for t in times)), name='start+dur=finish')
    for k in modules:
        for t in range(int(T - dur[k])):
            m.addConstrs((x[t + j, k] >= v[t, k]
                            for j in range(int(dur[k]))), name='x.for.v resource') #자원 점유

```

그림 4.4 정수 계획을 위한 Python 코드의 일부

4.3 업무 재구조화 결과 분석

재정립된 작업 배치 단위와 그 사이의 간격 제약을 사용하여 순차적 배치 알고리즘을 구현하는 Python 코드를 작성했다(그림 4.4). 순차적 배치 알고리즘 중 필수작업에 대한 정수 계획 문제의 구현과 그 해결에는 Python의 gurobi 라이브러리가 사용되었다. 이를 통한 순차적 배치 결과, 작업들이 가진 다양한 배치 제약을 모두 만족하면서 실행 가능한 최적 업무 대안을 생성할 수 있었다.

그림 4.5는 정수 계획에 따른 작업 모듈 배치 결과를 나타낸 간트 차트(Gantt Chart)이다. 유동작업이 가진 선후 관계를 정수 제약 조건에 반영했기 때문에, 유동작업이 배치될 간격이 작업 모듈 배치 시점부터 미리 마련된 것을 확인할 수 있다. 각 간격에는 그에 대응하는 유동작업들이 배치되고, 그에 이어

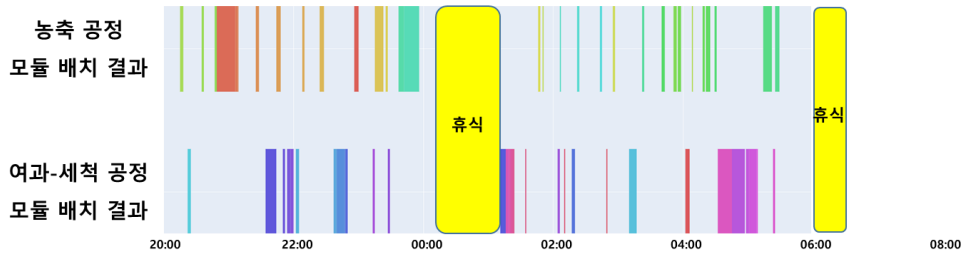


그림 4.5 정수 계획에 따른 작업 모듈 배치 결과

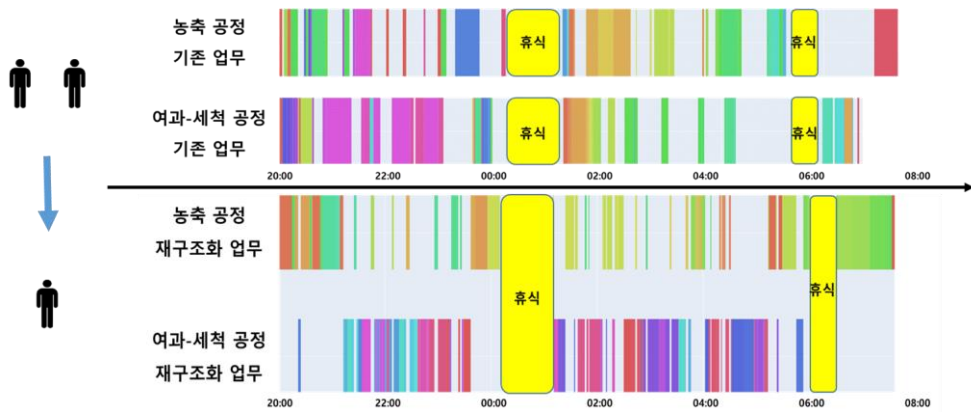


그림 4.6 재구조화에 따른 업무 구조의 변화

아무런 배치 제약을 갖지 않은 부수작업들이 필수작업과 유동작업이 채우고 남은 간격과 근무 종료 시점까지 남아있는 여유시간을 차지한다.

재구조화에 따른 작업자 업무 내용의 변화는 그림 4.6과 같다. 이를 통해 두 작업자가 배정되었던 공정이 한 작업자에 의해 진행될 수 있음을 보임과 함께, 본 연구에서 제시한 업무 재구조화 방법론이 실행 가능한 업무 대안을 성공적으로 탐색할 수 있음을 검증할 수 있었다. 재구조화 된 공정 업무를 진행하는 작업자의 효율은 총 604분의 작업 시간이 한 작업자에 의해서만 진행되므로 95.87%로 개선되었다. 이는 작업자 대기시간을 활용하여 다른

공정의 작업을 효율적으로 배치한 것에 대한 결과이다. 작업자 효율의 개선과 더불어 두 공정 진행에 대한 배치 인원이 2명에서 1명으로 축소되는 것으로 여유 인력을 공장 내의 다른 업무에 투입하거나 작업자 인건비를 절감할 수 있다.

제 5 장 결론 및 향후 연구 방향

5.1 결론

설비와 작업자가 함께 투입되는 공정에서 두 자원 사이의 작업량의 불균형이 발생하는 상황은 다른 한쪽에 대한 비효율로 이어지게 된다. 설비 작업이 공정의 주경로에 있는 설비 중심 공정에서 작업자가 필요 없는 설비의 자율 작동 시간은 작업자 근무 시간 내에서 대기시간을 발생시키므로 작업자 자원의 효율 개선을 위해서 여러 공정의 작업들을 대기시간에 배치하는 업무 재구조화가 필요하다.

본 연구는 제약 공정의 사례를 기반으로 설비중심공정의 생산성 향상 및 인력 효율화를 위한 실무적인 업무 재구조화 방법론을 제안한다. 이를 위해 작업 배치 제약의 종류에 따른 작업 유형 분류 체계를 정립한 뒤, 그를 활용해 실행 가능한 업무 대안을 생성하는 순차적 작업 배치 알고리즘을 제시했다. 작업 유형 분류 체계는 작업의 배치 가능성을 결정하는 두가지 요소인 작업 간 선후 관계의 유무와 작업의 분할 진행 가능성에 기반한다.

필수작업, 유동작업, 부수작업으로 분류된 작업들은 유형에 따라 서로 다른 배치 제약과 자유도를 가지고, 순차적 배치 알고리즘은 재구조화 대상 작업들을 배치 자유도가 낮은 유형부터 배치하는 것을 통해 실행 가능한 업무 대안을 빠르게 탐색한다. 그 과정에서 업무 대안을 생성하는 정수 계획 문제가 필수작업만을 대상으로 진행되도록 하는 것으로, 가능해 판별과 최적해 탐색에 필요한 연산의 규모를 줄일 수 있다.

제안된 방법론은 UDCA 생성 공정의 작업자 업무 관찰 데이터를 활용한 사례 연구를 통하여 검증되었으며, 재구조화를 통해 작업자 효율이 크게

개선됨과 함께 공정 진행에 필요한 작업자 수를 줄일 수 있음을 확인할 수 있었다.

5.2 향후 연구 방향

본 연구는 작업자의 비효율을 부른다는 점에서, 공정에 존재하는 대기시간을 없앨수록 좋은 악으로 보고 이를 제거하는 것을 목표로 했다. 그러나 작업 현장에는 강도 높은 육체노동 작업 뒤, 이후 작업의 원활한 진행을 위해 대기시간을 배정하는 것이 필요하다. 실제로 본 연구의 사례연구 대상 공정에서는 20kg 이상의 원료 용기를 수십 번 운반하는 원료 투입 작업 직후 휴식 시간을 배정하고 있다. 현실의 공정에서 대기시간은 작업자에게 반드시 필요한 휴식이기도 하다. 따라서 업무 재구조화에 있어 대기시간의 구체적인 내용과 목적에 대해 고려할 필요가 있다.

개별 작업 소요시간의 불확실성 역시 실행 가능한 업무 대안을 위해 고려해야할 요소이다. 본 연구에서는 업무 대안에 배치되는 작업들이 고정된 소요시간을 가졌다는 가정하에 재구조화를 진행했다. 그러나 설비중심공정의 대표적인 예시인 화학공정의 화학반응처럼, 개별 작업의 진행 시간은 매번 달라질 수 있다. 대기시간 없이 빠곡하게 배치된 업무 대안은 개별 작업에서 발생하는 작은 지연 만으로도 작업 간 선후 관계, 배치 간격 제약을 준수하는데 문제가 생길 수 있다. 작업 사이에 배치된 대기시간은 작업시간의 불확실성에 대한 버퍼 역할을 수행하는 것으로 작업 간 선후 관계 및 간격 제약으로 인해 발생하는 업무 실패 리스크를 줄인다.

이런 대기시간의 긍정적 역할을 고려한 업무 재구조화 연구는 실행 가능하고 설득력 있는 업무 대안을 만들고, 이를 실제로 적용하여 작업자 업무를 개선하는데 많은 도움을 줄 수 있을 것이다.

참고 문헌

- Freivalds, A., & Niebel, B. (2009). **Niebel의 작업관리** (12th ed.). (조문수, 윤훈용, 박성하, 역). 서울: 한국맥그로힐. (원서출판 2008).
- Abdolshah, M. (2014). A review of resource-constrained project scheduling problems (RCPSP) approaches and solutions. *International Transaction Journal of Engineering, Management, & Applied Sciences & Technologies*, 5(4), 253-286.
- Allen, J. F. (1981). An interval-based representation of temporal knowledge. In *Proceedings of the 7th international joint conference on Artificial intelligence-Volume 1*(pp. 221-226). San Francisco, CA: Morgan Kaufmann Publishers Inc.
- Boysen, N., Fliedner, M., & Scholl, A. (2007). A classification of assembly line balancing problems. *European journal of operational research*, 183(2), 674-693.
- Browning, T. R. (2015). Design structure matrix extensions and innovations: a survey and new opportunities. *IEEE Transactions on Engineering Management*, 63(1), 27-52.
- Çaliş, B., & Bulkan, S. (2015). A research survey: review of AI solution strategies of job shop scheduling problem. *Journal of Intelligent Manufacturing*, 26(5), 961–973
- DasGupta, B., & Palis, M. A. (2001). Online real-time preemptive scheduling of jobs with deadlines on multiple machines. *Journal of Scheduling*, Volume 4, Number.
- Dechter, R., Meiri, I., & Pearl, J. (1991). Temporal constraint networks. *Artificial intelligence*, 49(1-3), 61-95.

- Deckro, R. F., Hebert, J. E., & Verdini, W. A. (1992). Project scheduling with work packages. *Omega*, 20(2), 169-182.
- Globerson, S. (1994). Impact of various work-breakdown structures on project conceptualization. *International Journal of Project Management*, 12(3), 165-171.
- Hassanein, A., & Moselhi, O. (2004). Planning and scheduling highway construction. *Journal of Construction Engineering and Management*, 130(5), 638-646
- Hoffmann, T. R. (1963). Assembly line balancing with a precedence matrix. *Management Science*, 9(4), 551-562.
- Joe, K. (2017, September 14). Why robots will improve manufacturing jobs. *Time*. Retrieved from <https://time.com>.
- Kettimuthu, R., Subramani, V., Srinivasan, S., Gopalsamy, T., Panda, D. K., & Sadayappan, P. (2005). Selective preemption strategies for parallel job scheduling. *International Journal of High Performance Computing and Networking*, 3(2-3), 122-152.
- Lee, J., Deng, W. Y., Lee, W. T., Lee, S. J., Hsu, K. H., & Ma, S. P. (2010). Integrating process and work breakdown structure with design structure matrix. *Simulation*, 7, 8.
- Li, C. L., & Hall, N. G. (2019). Work package sizing and project performance. *Operations Research*, 67(1), 123-142.
- Maheswari, J. U., & Varghese, K. (2005). Project scheduling using dependency structure matrix. *International Journal of Project Management*, 23(3), 223-230.
- Muntz, R. R., & Coffman, E. G. (1969). Optimal preemptive scheduling on two-processor systems. *IEEE Transactions on Computers*, 100(11), 1014-1020.
- Norman, E. S., Brotherton, S. A., & Fried, R. T. (2008). *Work breakdown structures: the*

foundation for project management excellence. Hoboken, NJ: John Wiley & Sons.

Rodammer, F. A., & White, K. P. (1988). A Recent survey of production scheduling. *IEEE Transactions on Systems, Man and Cybernetics*, 18(6), 841–851.

Sequeira, S., & Lopes, E. (2015). Simple method proposal for cost estimation from work breakdown structure. *Procedia Computer Science*, 64, 537-544.

Supriadi, L. S. R., Latief, Y., Susilo, B., & Rajasa, M. (2017). Development of risk-based standardized WBS (Work Breakdown Structure) for cost estimation of apartment's project. *International Journal of Civil Engineering and Technology*, 822-833.

Thomopoulos, N. T. (1967). Line balancing-sequencing for mixed-model assembly. *Management Science*, 14(2), B-59.

Zhang, J., Li, Y., Yu, J. F., & Zhang, K. F. (2008). A new method for generating work breakdown structure on aircraft assembly scheduling. In *Proceedings of 2008 4th International Conference on Wireless Communications, Networking and Mobile Computing* (pp. 1-4). Piscataway, NJ: IEEE.

Abstract

Work Restructuring Methodology for Workforce Efficiency in Facility-Centric Process

Seongwon Cho

Department of Industrial Engineering

The Graduate School

Seoul National University

Like the reactor facilities in the chemical production process can operate autonomously, the task done only by the facility without the operator is the idle time of the workers placed together in the same process. This workload imbalance, which exists between the facility and worker resources, makes the tasks of the facility critical passes through the entire process, resulting in inefficiency of the workers, high percentages of worker's idle time during the entire working hours, and low workers' utilizations.

A facility-centric process is a process in which the tasks of facilities rather than workers, such as chemical processes, is critical passes to the entire process. In order to improve the imbalance of workload between resources arising from the facility-centered processes and the resulting low efficiencies of workers' resources, it is necessary to restructure the entire works to improve the efficiencies of workers' resources by placing the tasks of other processes in the workers' idle time.

This thesis presents a task type classification according to the freedom of task placement as an empirical methodology for them, and a sequential placement

algorithm based on the task types. The task type classification categorizes the entire tasks into three types: essential, floating, and incidental tasks, depending on the task characteristics that determine the constraints of task placement. The classified individual task units are placed during worker's working hours through the integer programming that reflects the preceding and succeeding task relationships and placement constraints. Sequential placement algorithm is the algorithm that places tasks in the order of essential, floating, and incidental task during workers' working hours. It constructs integer programming problem only for essential tasks, which are of the type with less freedom of placement, reducing the scales of the overall task placement problem.

The proposed methodology has been validated in the case study of the work restructuring of the UDCA production processes.

Keywords: Facility-Centered Process, Task Classification, Integer Programming,
Work Restructuring, Industrial Engineering
Student Number: 2018-25619