
경영과학 연구실 워크샵 발표

김변민

목차

1. 클러스터 장비 내 웨이퍼 품질 향상을 위한 스케줄링
2. Last -mile delivery station with heterogeneous vehicles: modeling, solution approaches, and implications
3. 생산 스케줄링에 대한 강화학습 적용을 위한 시뮬레이션 프레임워크에 관한 연구
4. 대학 지원 최적화 문제
5. 메타 휴리스틱 기반 조선 작업장 내 임의의 형상을 가진 3D 블록의 공간배치 알고리즘 연구

제목

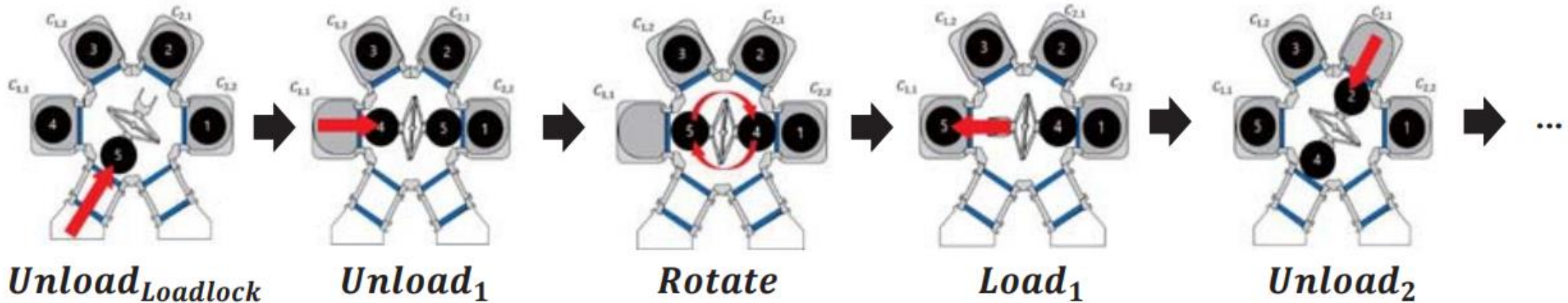
클러스터 장비 내 웨이퍼 품질향상을 위한 스케줄링: 웨이퍼 지연 및 흐름시간 제어

저자

김민찬 외 1 인

내용

반도체 공정에서 Partial loading 을 이용한 스케줄링 기법



Robotized cluster tools

Deposition, Etching, wafer cleaning 공정에 주로 사용됨

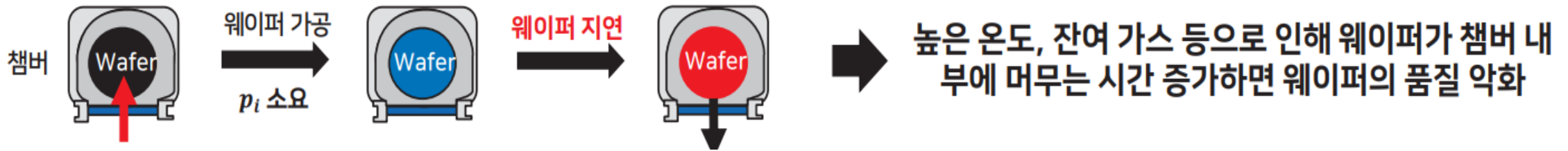
장비 내부에 buffer space 가 없음

가격 : 3~400 million 달러

공정 단계 순서대로 로봇은 챔버에서 웨이퍼를 꺼내고, 다시 채워 넣는 방법을 반복

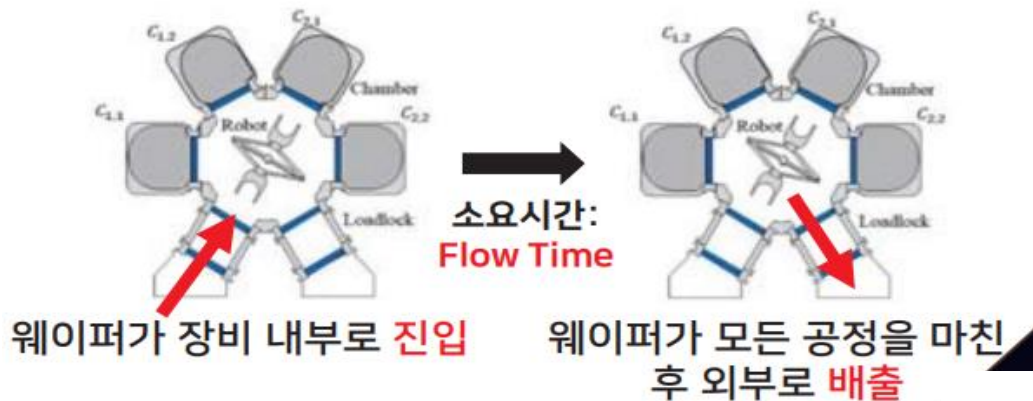
웨이퍼 지연

챔버에서 가공을 마친 웨이퍼가 챔버 내부에서 머무는 시간



플로우 타임

로드락에서 장비로 투입된 웨이퍼가 클러스터 장비에서 머무는 시간



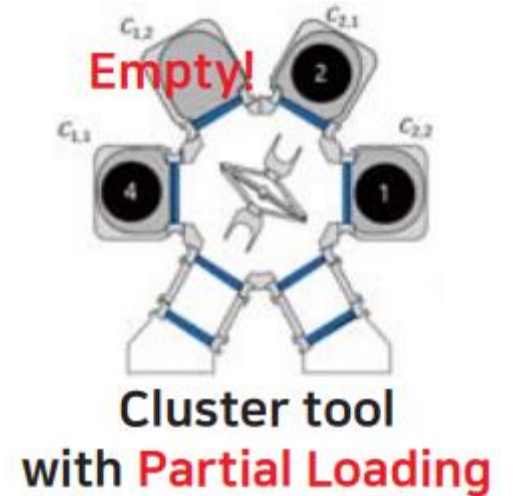
*****웨이퍼 지연과 플로우타임은 모두*****
작업부하와 밀접한 관련이있음

부분 적재

일부 챔버를 비운채로 운영 - > 챔버 개수를 유지하면서 작업부하를 balancing

장비의 생산성 측면에서는 부분적재 전략의 장점 X

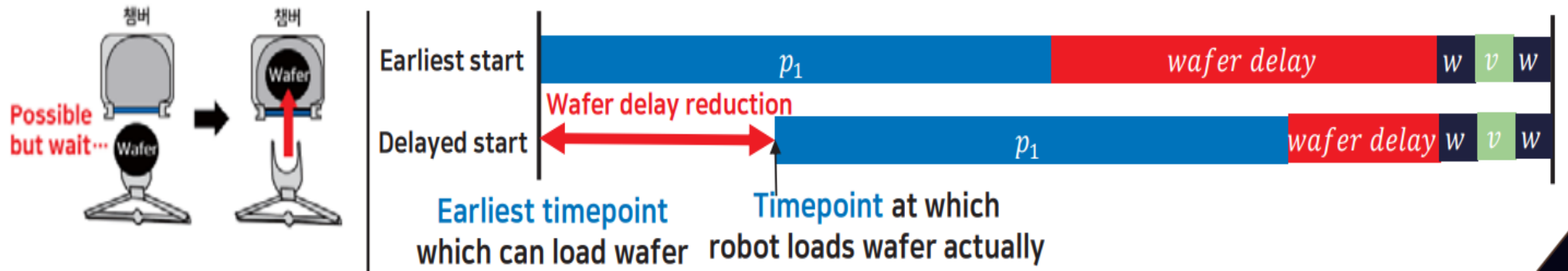
하지만 과도한 병렬화가 있는 공정에서는 부분적재 활용



장비의 생산성을 유지하면서 불필요한 웨이퍼 지연 및 흐름 시간을 감소

기존의 해법 (timing control method)

Cycle time 을 증가시키지 않는 선에서 로봇의 웨이퍼 로딩작업을 지연시킴

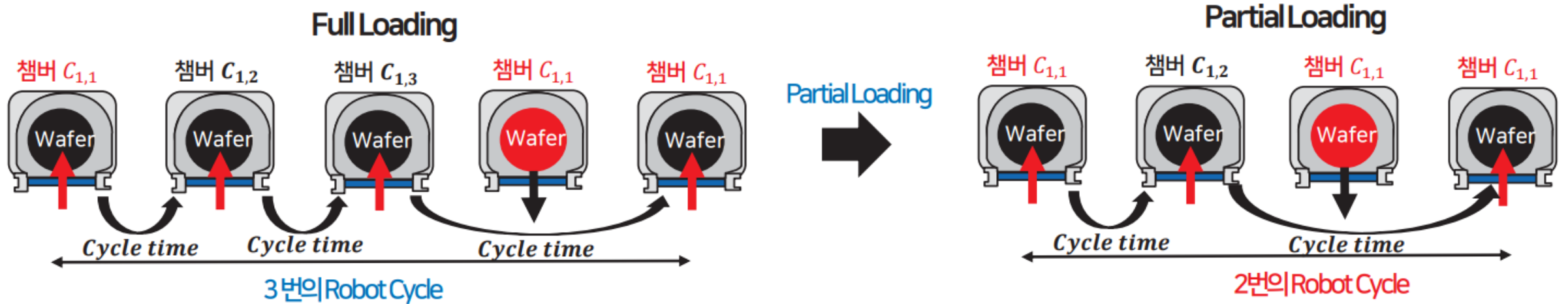


한계

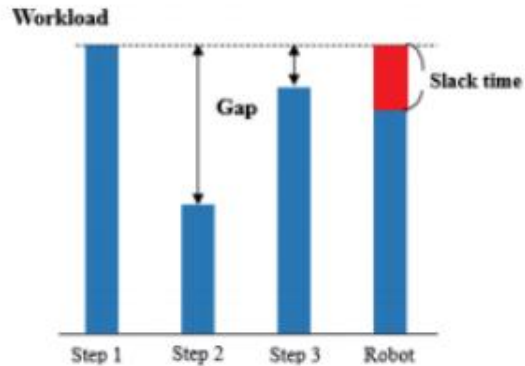
Slack time 이상으로 로봇행동이 지연된다면 다른 공정 단계의 웨이퍼 딜레이도 함께 증가

즉, Timing Control Method는 **작은 Wafer Delay를 제어하기 위해 효과적**

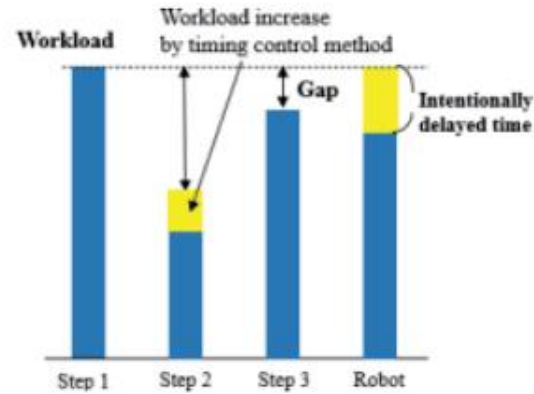
만약 Partial loading이 적용된다면....



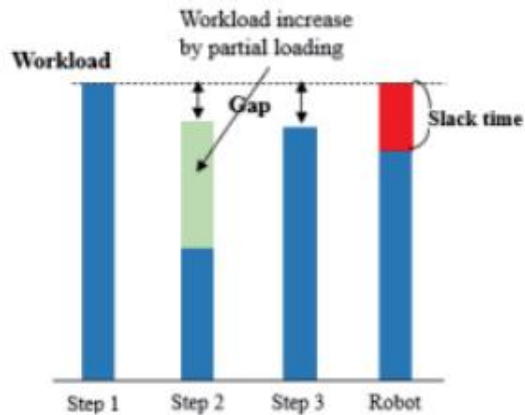
웨이퍼가 기다려야하는 로봇 사이클의 횟수를 줄일 수 있음



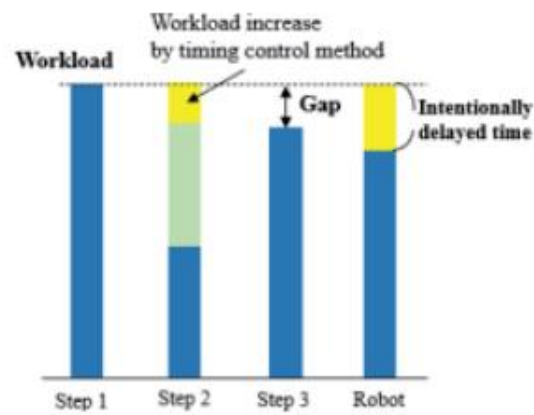
(a) Swap(0) is considered.



(b) timing control method is applied to (a).



(c) Partial loading (i.e., Swap(z)) is applied to (a).



(d) timing control method is applied to (c).

Description

(a) 최초 상태

- 병목공정 (Step 1)과 다른 리소스들의 작업부하 차이가 Wafer Delay를 유발

(b) Timing Control Method 적용

- 로봇의 Slack time → Step 2의 Workload를 의도적으로 증가
- 하지만 대부분의 Wafer Delay는 해결 X

(c) Partial Loading을 대신 적용

- Step 2의 챔버 하나를 비움으로써 Workload Balancing
- 하지만 대부분의 Wafer Delay 해결 완료

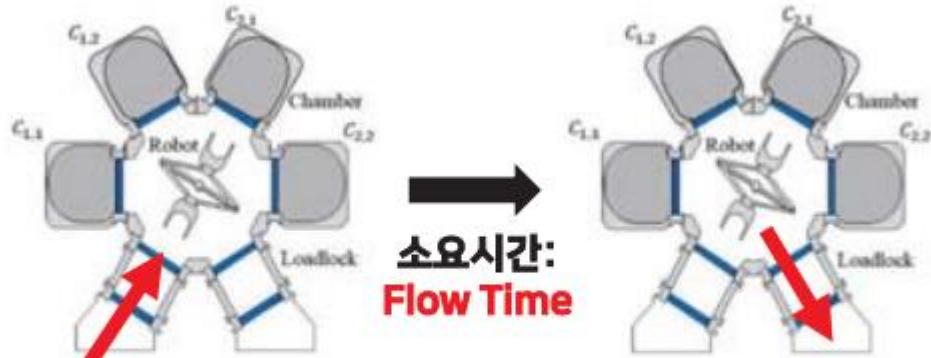
(d) 추가적으로 Timing Control Method도 적용

- Slack Time → 나머지 Step 2의 Workload Balancing
- 장비 내부에 웨이퍼 딜레이는 거의 존재하지 않음

→ 결과적으로 두 방법을 동시에 활용할 수 있다는 장점 존재

Flow time

장비로 투입된 웨이퍼가 모든 공정을 마치고 로드락으로 다시 돌아오는데 소요되는 시간




웨이퍼가 장비 내부로 **진입** 웨이퍼가 모든 공정을 마친 후 외부로 **배출**

$$\therefore \text{Flow Time} = (\text{장비내부웨이퍼의개수} + 1) \times \text{Cycletime} - \text{로봇작업}$$

Cycle time = 웨이퍼 배출 간격

Flow time 은 클러스터 장비 내부 웨이퍼의 개수에 의해 결정됨

Partial loading 은 장비 내부에 있는 웨이퍼를 줄이는 것  Partial loading은 flow time 을 효과적으로 제어 할 수 있음

Objective

Wafer delay, Flow time

Summary

공정 단계간의 작업부하 불균형 -> wafer delay 및 flow time 증가

Partial loading 을 통해 작업부하 불균형을 해소

Partial loading 은 wafer delay 와 Flow time 을 동시에 제어 가능

Partial Loading → Wafer Delay와 Flow time을 동시에 제어

- **Timing Control Method**: 로봇이 웨이퍼를 더 오래 '들고 있어주는 것' → Flow time에 영향 X
- **Partial Loading**: 작업부하 불균형으로 발생하는 Wafer delay를 직접 줄이는 것 → Flow time 감소

제목

Last-mile delivery station with heterogeneous vehicles: modeling, solution approaches, and implications

저자

최병일 외 3인

내용

Clustering 과 Reassignment를 이용해 UAV 와 delivery 로봇이 각 station에 몇 개를 비치해 놓을 지와 경로를 정함

Terms

Last-mile delivery = 유통업체의 상품이 목적지에 도착하기까지의 전 과정을 뜻하는 용어

UAV = 무인 비행기 (드론)

Delivery robot = 운반로봇



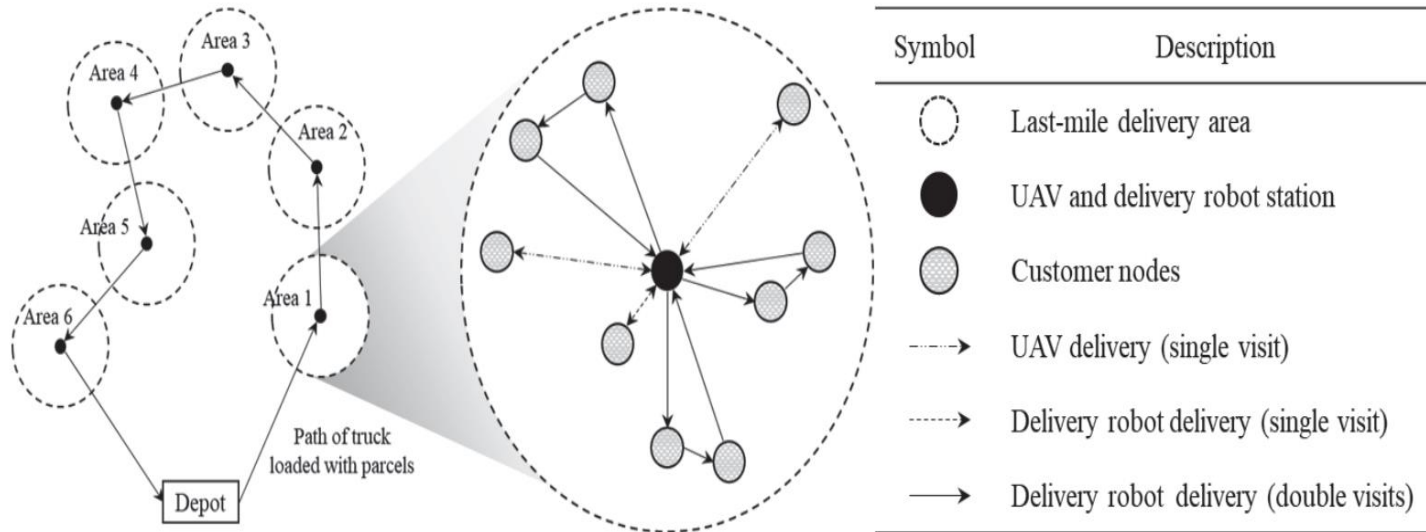
Background of the logistics industry

E-commerce market 은 계속 성장 중이며 배달 서비스에 대한 수요도 증가

E-commerce 회사에서 자체적으로 물류 시스템을 개발중임

E-commerce 회사에서 경쟁력을 가지기위해 효율적인 물류 시스템을 적용하기 원하고 있음

The problem in this study



목표

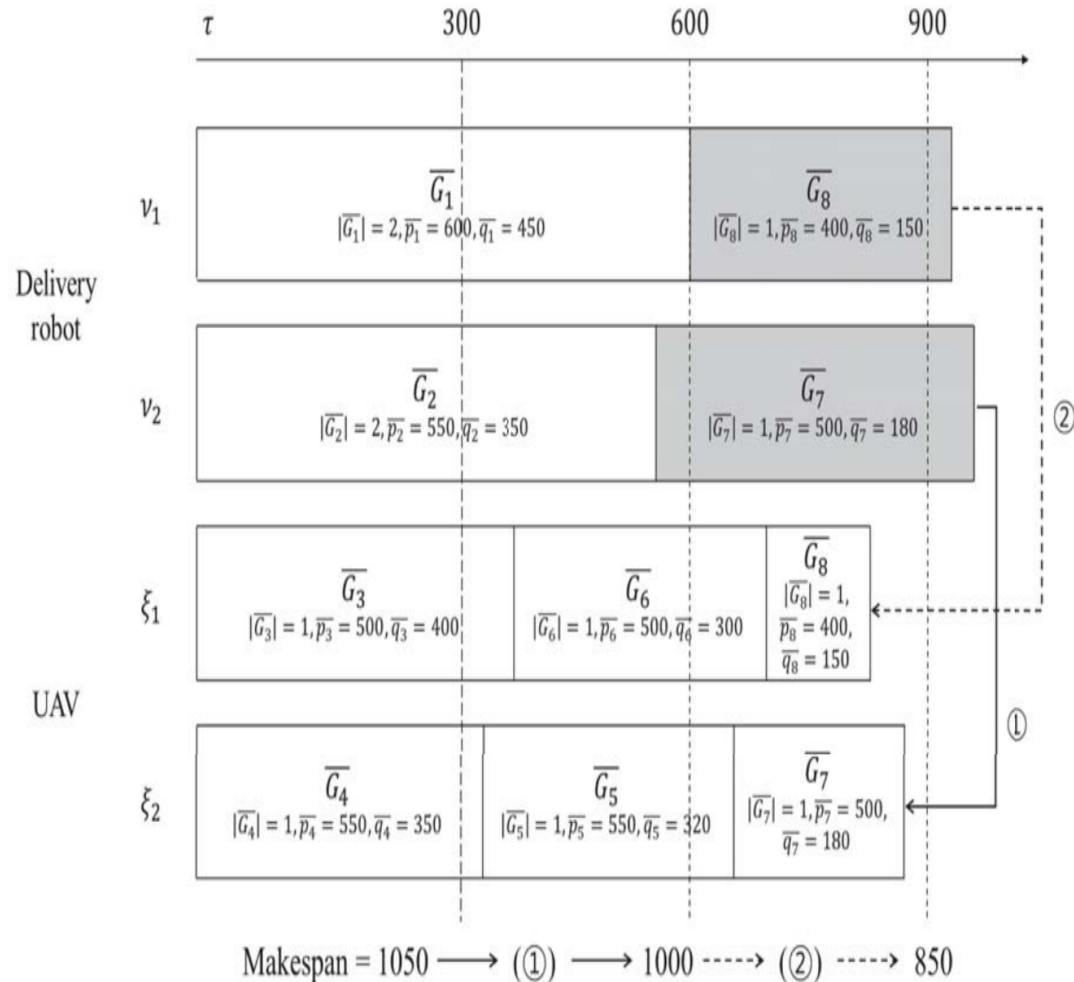
Delivery station에 UAV 와 delivery robot 이 얼마나 있어야 할지 정함

Delivery cost를 최소화 할 수 있는 UAV 와 delivery robot의 루트를 찾음

Heuristic Approach

Last-mile delivery station with heterogeneous vehicles: modeling, solution approaches, and implications

- \bar{G}_i : cluster, \bar{p}_i : delivery robot travel time, \bar{q}_i : UAV travel time, and i indicates the order in a sequence list



매커니즘

1. K-means 를 이용해 각 노드를 군집화
2. 각 cluster 를 UAV나 delivery robot에 할당
3. Makespan의 향상을 위해 2번에서 했던 할당 중 마지막 할당을 조정
4. UAV 와 delivery robot 의 수를 교체
5. 최적방안 도출

$$\text{Gap to MILP (\%)} = \frac{\text{Objective value of CRA++ heuristic} - \text{Optimal value of MILP}}{\text{Optimal value of MILP}} \times 100$$

# of nodes	MILP		CRA++ heuristic		
	Avg. optimal value	Avg. cplex time (sec)	Avg. optimal value	Avg. gap to MILP (%)	Avg. run time (sec)
10	1380.8	0.5	1446.8	6.0	0.03
15	1389.0	5.0	1486.5	6.7	0.19
20	1952.0	812.8	2069.2	6.6	0.56
50	-	-	3416.4	-	22.51
100	-	-	4745.2	-	1075.52

MILP = 일반적인 수리모델을 이용한 최적방안

CRA++heuristic = 본 발표에서 제안한 휴리스틱 방법론

- 2 기종 차량을 동시에 사용하는 새로운 delivery station 모델 소개
- Practical size 의 문제를 풀기위해서는 MILP 모델보다 CRA++ heuristic 모델이 적절함
- 적은 비용과 빠른 배송으로 e-commerce 회사의 니즈를 충족시킬 수 있음
- 향후 연구로 특성이 다른 유형의 차량을 추가해서 모델을 풀 수 있음

제목

생산 스케줄링에 대한 강화학습 적용을 위한 시뮬레이션 프레임워크에 관한 연구

저자

오승헌 외 5인

내용

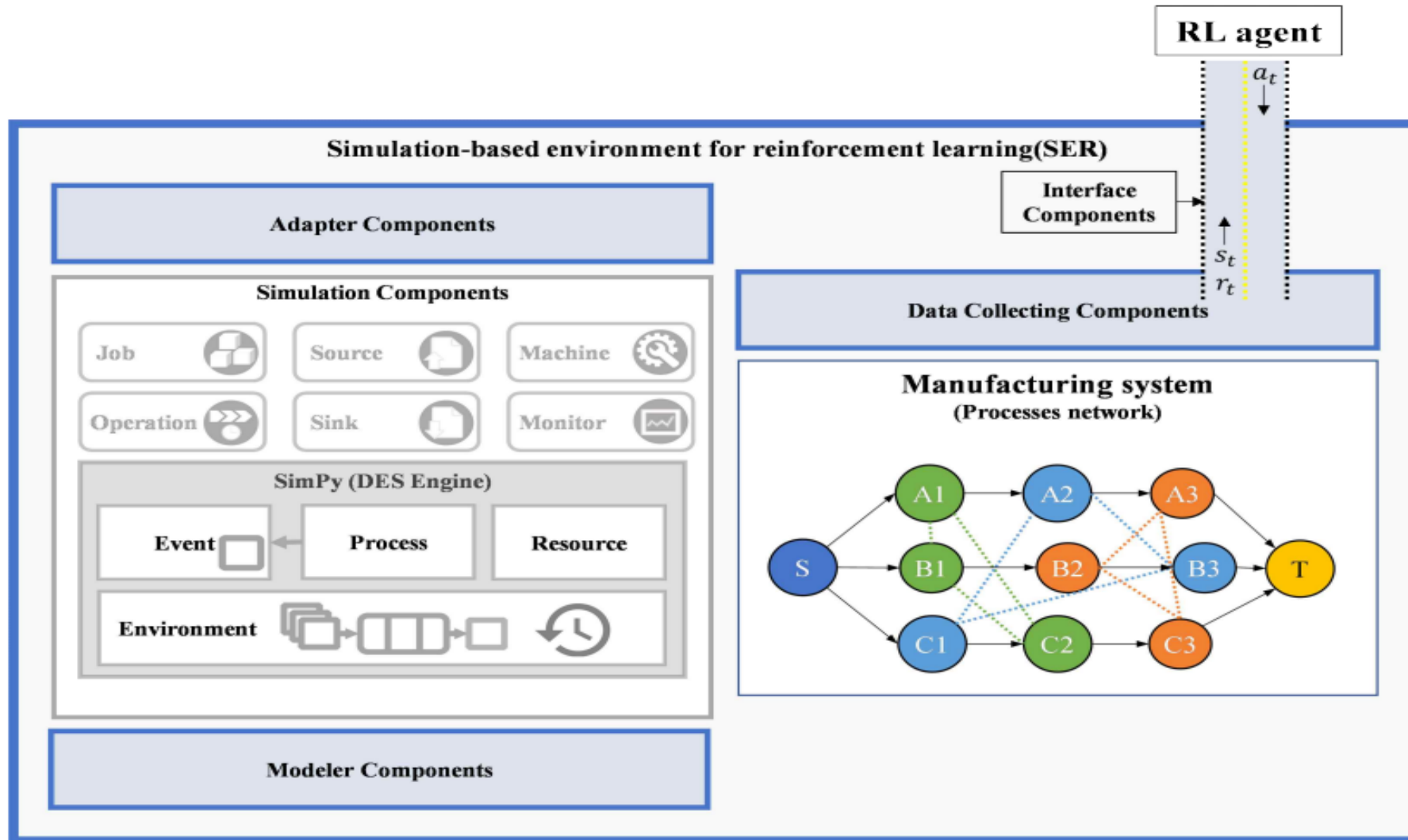
시뮬레이션 기반 강화학습 환경 개발 도구들이 지닌 특성을 토대로 강화학습을 효과적으로 적용하기 위한 고려요소 및 이를 개발 프레임워크를 제안하고 적용사례 소개

시뮬레이션 기반 강화학습을 위한 4가지 요소

- 신뢰성
- 신속성
- 상호운용성
- 비용 효율성



Simpy 적용



Application

생산 스케줄링에 대한 강화학습 적용을 위한 시뮬레이션 프레임워크에 관한 연구

	Case1	Case2	Case3	Case4
Production system	Flowshop	Parallel machine	Flexible jobshop	Flexible jobshop
Example	Panel block assembly line	Pre-painting	Block logistics	Quay scheduling
Scheduling problem	Sequencing	Job-machine allocation	Job-machine allocation	Quay assignment
Objective	Minimizing makespan	Minimizing weighted tardiness	Minimizing makespan	Minimizing ship movements and maximizing quay

Case 1

Makespan 이 랜덤 하게 블록을 투입하는 것보다 13.1%, 휴리스틱보다 6.87% 가량 성능이 향상됨

Case 2

Mean weighted tardiness 관점에서 우선규칙을 단독 사용하는 경우보다 최소 0.45%에서 최대 39%의 성능 향상을 보임

Case 3

최소 60개에서 최대 350개의 job에 대한 스케줄링을 수행하였고 약 2.5초 내에 디스패칭 룰 대비 양질의 솔루션을 산출

Case 4

조선소에서 목표로 하는 기준치 대비 최소 1.1% 에서 15%까지 이동량을 감소시켜 주는 개선효과 보임

제목

대학 지원 최적화 문제

저자

홍성필 외 1 인

내용

대학 지원 전략을 최적화 문제로 모형화
문제의 계산 복잡도 분석, 해법제시, 알고리즘 구현

1. 배분적 휴리스틱

입학 컨설팅 산업에서 주로 "상향, 소신, 안정 지원 학교" 각각 균일하게 지원함
위험 회피적인 전략임

2. 선형화 휴리스틱

$$\text{maximize } \sum_{j \in \mathcal{X}} f_j t_j \quad \text{subject to } \sum_{j \in \mathcal{X}} g_j \leq H$$

이는 C_j 에는 지원했지만 C_j 보다 선호하는 학교에 합격했을 때를 고려하지 않음

Parameter

- $C = \{1 \dots m\}$
- $C_j = j$ 번째 학교 이름
- $f_j = j$ 번째 학교 합격확률
- $Z_j \sim \text{Bernoulli}(f_j) =$ 학생이 합격하면 1, 아니면 0
- $g_j = C_j$ 의 지원비용(지원횟수, 전형료예산, 원서작성시간)
- $t_j = C_j$ 에 다닐 때의 효용

효용모형

$$\max \{t_0, \max\{t_j Z_j : j \in \mathcal{X}\}\}$$

기본 최적화 모형

$$\begin{aligned} &\text{maximize } v(\mathcal{X}) = \sum_{j \in \mathcal{X}} \left(f_j t_j \prod_{\substack{i \in \mathcal{X}: \\ i > j}} (1 - f_i) \right) \\ &\text{subject to } \mathcal{X} \subseteq C, \quad \sum_{j \in \mathcal{X}} g_j \leq H \end{aligned}$$

정수 비선형 계획 모형

$$\begin{aligned} &\text{maximize } v(x) = \sum_{j=1}^m \left(f_j t_j x_j \prod_{i>j} (1 - f_i x_i) \right) \\ &\text{subject to } x_j \in \{0, 1\}, j \in C; \quad \sum_{j=1}^m g_j x_j \leq H \end{aligned}$$

Branch and Bound

일반적인 INLP문제에 대해 자주 쓰이는 해법

총 지원 비용 기반 동적계획

의사다항 시간에 정확한 해를 구하며, g_j 가 작은 정수가 되는 전형적 인스턴스에 대해 매우 효율적인 해법

Simulated annealing 휴리스틱

매우 빠르고 대부분 98%이상의 최적성을 얻음

제목

메타 휴리스틱 기반 조선 작업장 내 임의의 형상을 가진 3D 블록의 공간배치 알고리즘 연구

저자

이승엽 외 3인

내용

조선 작업장 내 3D 블록의 공간배치 알고리즘 연구

조선 산업에서 공간배치의 필요성

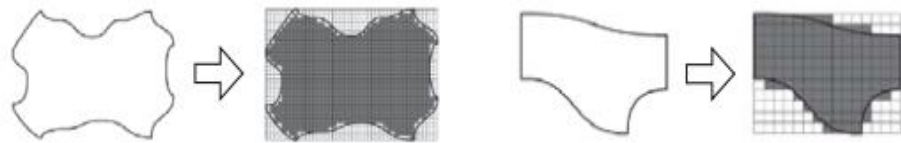
선박 제조에 필요한 블록들의 효율적인 배치는 상당한 생산성 증대를 가져옴

3D 비정형 형상 및 작업장의 공간 배치의 어려움

주로 2차원 좌표평면에서 직사각형 혹은 표준 형상의 블록과 작업장 공간을 가정
블록 회전, 장애물, 구역 구분 등의 조선 산업의 특징을 반영한 연구가 필요

비정형 형태의 표현

Raster method



주어진 도형을 grid 로 쪼개어 격자화

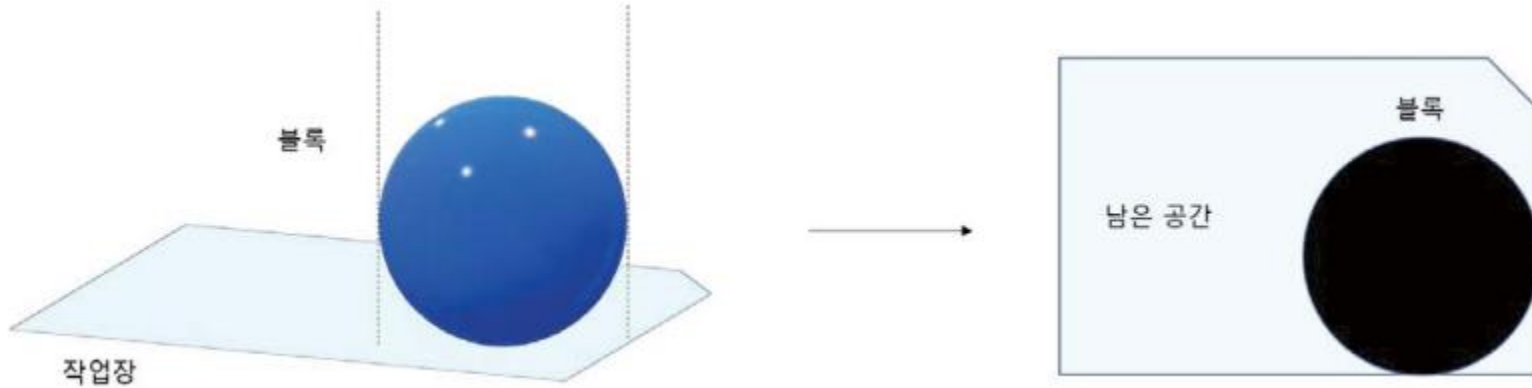
Polygonal method



주어진 도형을 있는 그대로 벡터로 표현

목적

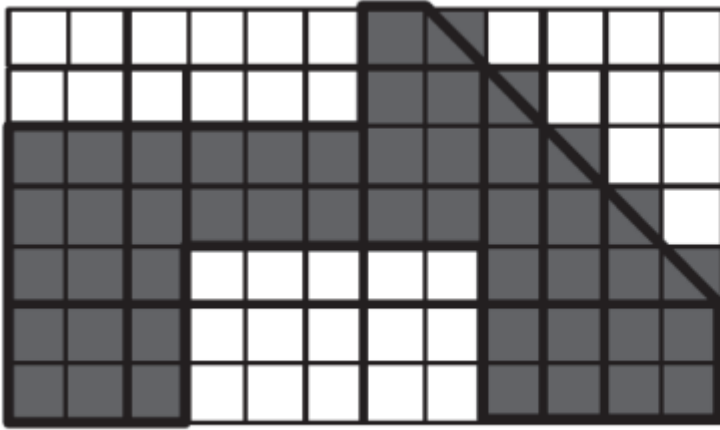
작업장의 면적 활용률을 최대화 (면적 활용률 = 블록이 배치된 면적 / 작업장 면적)



배치 제약조건

- 모든 블록을 배치할 수 없음을 가정
- 이미 배치되어 있는 블록 존재
- 제외되어야 하는 블록 존재
- 블록은 단층으로 배치

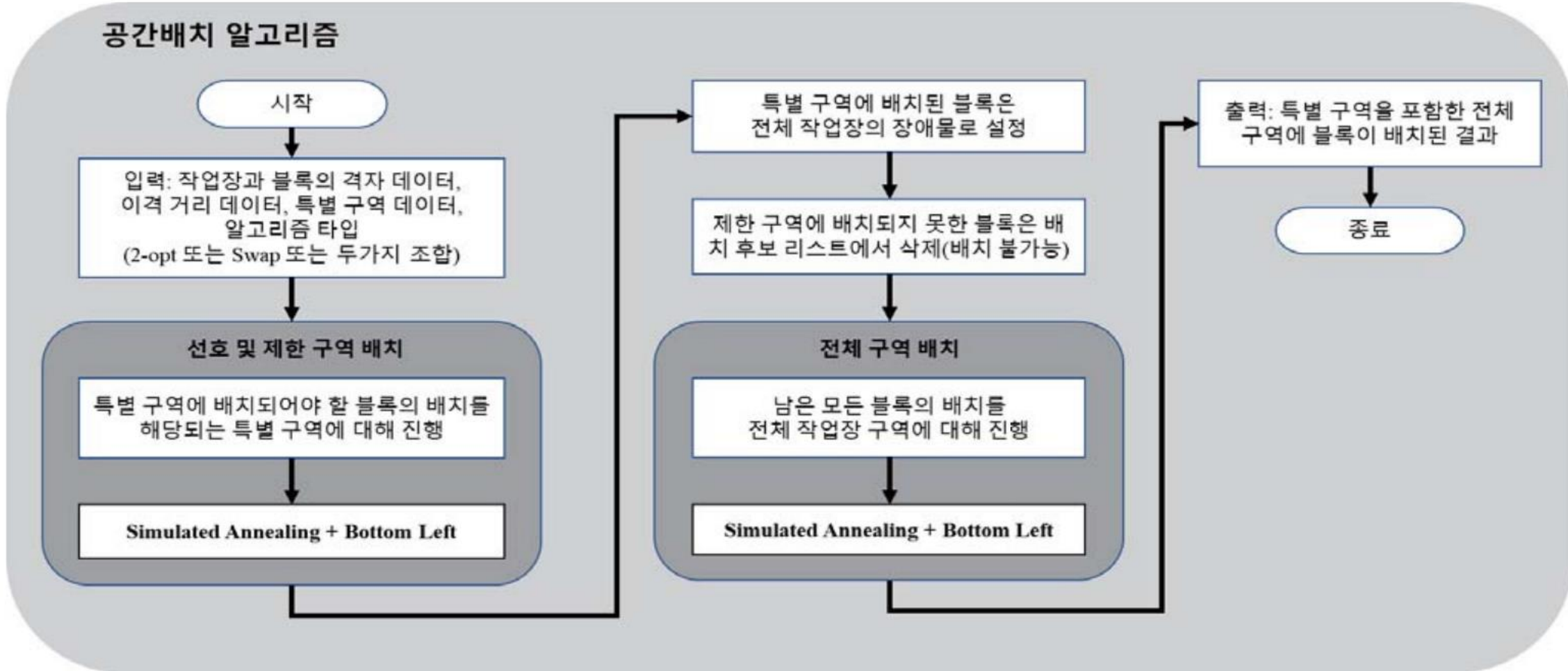




0	0	0	0	0	0	1	1	0	0	0	0
0	0	0	0	0	0	1	1	1	0	0	0
1	1	1	1	1	1	1	1	1	1	0	0
1	1	1	1	1	1	1	1	1	1	1	0
1	1	1	0	0	0	0	0	1	1	1	1
1	1	1	0	0	0	0	0	1	1	1	1
1	1	1	0	0	0	0	0	1	1	1	1

0-1 raster representation 이용

- 블록을 표현하는 각 꼭지점 좌표가 순서대로 주어짐
- 블록을 감싸는 가장 작은 격자 직사각형 생성
- 연속된 꼭지점이 나타나는 블록의 모서기를 바탕으로 격자점에 블록의 포함여부를 저장
- 단위 높이마다 3D 블록의 단면을 해당 블록 격자화로 표현



아래 순서 기준으로 배치 후 Bottom left 결과가 가장 좋은 순서 선택

- 임의 순서
- 가로 길이
- 블록 넓이
- 빈공간 넓이
- 직사각형 유사도



Simulated annealing 을 통해 블록 배치 순서 개선

Simulated annealing = 고온 물질의 분자가 식어가면서 점차 안정화되어 가는 과정을 묘사하여 광역적 최적화를 수행하는 알고리즘

- 2D 블록 배치
 - ▶ 99개 블록 중 40개 배치 / 계산 시간: 378.8초 / 면적 활용률: 82.1%



- 3D 블록 배치
 - ▶ 99개 중 45개 배치 / 계산 시간: 413.3초 / 면적 활용률: 85.8%



- 알고리즘 적용 후 2D 에서는 면적 활용률이 3.7%가량 상승