

# 재구성제조시스템에서 설비배치와 Job Shop 스케줄링의 통합 최적화에 관한 연구

김재혁 · 서윤호<sup>†</sup>

고려대학교 산업경영공학부

## Integrated Optimization of Facility Layout and Job Shop Scheduling in Reconfigurable Manufacturing System

Jaehyuk Kim · Yoonho Seo

Department of Industrial Management Engineering, Korea University

In manufacturing systems, the facility layout problem (FLP) and the job shop scheduling problem (JSSP) are two main factors influencing on productivity and efficiency of the systems, and a number of previous researches have dealt with these independently. However, FLP and JSSP need to be considered concurrently because the two factors affect each other's performance, and the modularity of production lines in reconfigurable manufacturing system (RMS) makes it easy and cost-effective to change the facility layout. In this paper, we proposed an integrated algorithm considering FLP and JSSP simultaneously, and it was compared with a method that considers JSSP only. As a result, it was found that the modularity of RMS was worthwhile and the proposed integrated algorithm was effective in deriving optimal solution combination of FLP and JSSP.

**Keywords:** Facility Layout Problem (FLP), Job Shop Scheduling Problem (JSSP), Reconfigurable Manufacturing System (RMS), Integrated Optimization

### 1. 서론

현재의 스마트제조시스템 (smart manufacturing system)이 등장하기까지 제조산업에서 중요시되었던 요소는 지속적으로 변경되어왔으며, 그에 따른 다양한 제조시스템이 등장하였다. 1990년대 초에는 비용효율성 (cost-effectiveness)이 중요한 요소였으며 컨베이어 벨트를 활용한 포드의 대량생산 시스템 (Ford's mass production system)을 통해 규모의 경제를 실현하였다. 그 후에는 대량생산뿐만 아니라 제품의 품질향상이 중요한 요소였고 린 생산시스템 (lean production system)의 적용을 통해 제품생산 시 발생하는 낭비요소를 최소화하여 제품의 품질을 향상시켰다.

현재는 고객의 니즈가 다양해지고 급격히 변화함에 따라 개별 고객을 위한 맞춤형 제품생산 (customization)과 급격한 시장

변화에 대응할 수 있는 능력이 중요시되고 있으며, 과거 공급자 중심의 대량생산시스템에서 소비자 중심의 다품종 소량생산 시스템으로 전환되고 있다. 이러한 고객 니즈의 다양화와 빠른 시장변화에 대응하기 위해서는 높은 유연성과 생산성을 동시에 제공할 수 있는 생산시스템이 요구된다. 또한 신제품의 생산이나 기존 제품의 주문변경 또는 고객 맞춤형 제품생산의 필요성에 따라 서로 다른 작업공정순서를 가지는 다양한 제품을 생산하기 위해서는 생산라인을 자주 변경할 수 있어야 한다. 이를 충족하기 위해서 1990년대 중반에 재구성제조시스템 (Reconfigurable Manufacturing System, RMS)이 등장했다 (Koren *et al.*, 2017).

재구성제조시스템이란 빠른 시장 변화와 기술환경 변화에 대응하고 다양한 제품이 생산될 수 있도록 처음부터 설계된 시스템을 말하며 시스템의 모듈화를 특징으로 한다. 재구성제

<sup>†</sup> 연락저자 : 서윤호 교수, 02841 서울특별시 성북구 안암로 145 고려대학교 공과대학 산업경영공학부, Tel : 02-3290-3393, Fax : 02-3290-4550, E-mail : yoonhoseo@korea.ac.kr

2021년 6월 25 접수; 2021년 8월 15일, 2021년 11월 2일 수정본 접수; 2021년 11월 30일 게재 확정.

조시스템의 모듈화는 시스템의 재구성 가능성을 향상시킬 뿐만 아니라, 프로세스 변경 요구사항을 충족시켜 시스템의 유연성을 높인다. 특히 생산라인의 모듈화는 라인의 변경을 용이하게 하며, 하나 이상의 모듈을 수정함으로써 기존의 설비배치를 효율적으로 변경할 수 있다(Lameche *et al.*, 2017).

제조시스템에서 설비배치와 스케줄링은 생산성과 비용 효율성에 영향을 미치는 주요 요인들이며, 제조시스템의 총 운영비용 중 20~50%는 자재취급비용(material handling cost)으로 효율적인 설비배치를 통해 이 비용의 10~30%를 줄일 수 있는 것으로 알려져 있다(De Carlo *et al.*, 2013; McKendall and Liu, 2012). 또한 고객 니즈의 다양화와 빠른 시장변화에 대응하기 위해서 작업의 스케줄링은 효율적인 생산 프로세스를 위한 필수적인 요소이다.

기존 제조시스템의 경우 설비배치가 한번 이루어지고 나면 문제점이 대두되더라도 설비의 재배치를 위해서는 많은 시간과 비용이 소요되기 때문에 단기간에 변경되기 어려우며(Kulturel-Konak, 2007), 정해진 설비배치에 대한 작업 스케줄링 변경을 통해 제조시스템의 효율적인 운영을 하였다. 이로써 설비배치 문제(Facility Layout Problem, FLP)와 다품종 소량 생산 환경의 잡샵 스케줄링 문제(Job Shop Scheduling Problem, JSSP)에 관한 대다수의 연구가 독립적으로 진행되고 있다. 하지만, 재구성제조시스템(RMS)의 경우 생산라인 모듈화로 인해 설비배치의 재구성이 용이할 뿐만 아니라 설비배치와 스케줄링 각각의 요인들에 대한 결과는 상호작용하며 서로의 결과에 영향을 미치기 때문에 함께 고려되어야 한다(Ripon *et al.*, 2012).

본 연구의 목적은 설비배치 문제와 스케줄링 문제를 동시에 고려하여 전체 작업의 자재흐름거리(material flow distance)와 공정완료시간(makespan)을 최소화하는 설비배치 및 작업순서 조합을 도출하는 것이다. 이를 위해서 본 연구에서는 두 가지 문제를 통합적으로 고려하여 최적의 조합을 도출할 수 있는 통합 알고리즘을 개발하였다. 통합 알고리즘은 크게 두 단계로 구성되어있으며, 첫 번째 단계에서는 설비배치와 스케줄링 해 조합의 다양한 해공간(solution space) 탐색을 위해 임의 탐색 방법(Random Search Method)을 적용하였고, 두 번째 단계에서는 지역탐색을 강화하기 위해 타부 서치 방법(Tabu Search Method)을 적용하였다.

본 논문의 구성은 다음과 같다. 제2장에서는 설비배치 문제와 스케줄링 문제에 관해 다룬 기존의 연구를 살펴보고, 제3장에서는 본 연구에서 다루고자 하는 설비배치 및 스케줄링 문제에 대한 문제 정의를 기술한다. 제4장에서는 본 연구에서 제안하는 방법의 해 표현 방법에 대해서 기술하며, 제5장에서는 설비배치와 스케줄링 문제를 해결하기 위한 통합 알고리즘에 대해 기술한다. 제6장에서는 본 연구에서 제안한 방법과 주어진 설비배치에 대하여 스케줄링 문제만 고려한 경우의 비교실험을 통하여 통합 알고리즘의 효율성을 입증하고, 마지막으로 제7장에서는 결론 및 향후 연구과제에 대해 기술한다.

## 2. 기존 연구 고찰

제조시스템의 최적운영을 위해서 지난 수십 년 동안 여러 방면에서 연구가 진행되어왔으며, 특히 NP-hard 문제로 알려진 설비배치 문제와 스케줄링 문제를 해결하기 위한 다수의 연구들이 진행되었다(Applegate and Cook, 1991; Drira *et al.*, 2007). 설비배치와 스케줄링 문제는 NP-hard 특성상 문제 사이즈가 커질수록 제한된 시간 내에 최적해를 도출해내는 것이 거의 불가능하며, 최적 근사해를 도출하기 위한 다양한 휴리스틱 알고리즘(heuristic algorithm)이 연구되었다. 이에 대한 기존의 연구로는 크게 두 문제를 각각 독립적으로 고려한 것과 통합적으로 고려한 것으로 나누어 볼 수 있다.

설비배치와 스케줄링 문제를 독립적으로 다루며 휴리스틱 방법을 사용한 기존의 연구는 다음과 같다. 먼저 설비배치 문제를 다룬 기존의 연구로 El-Baz(2004)는 여러 자재흐름(Material flow) 유형에 대하여 자재취급비용을 최소화하기 위해서 유전알고리즘(Genetic Algorithm, GA)을 제안하였으며, Nasab and Mobasheri(2013)는 자재의 이동 거리를 최소화하기 위해 시뮬레이티드 어닐링 알고리즘(Simulated Annealing algorithm, SA)을 제안하였다. McKendall and Liu(2012)는 동적설비배치문제(Dynamic Facility Layout Problem, DFLP)에서 자재취급비용과 설비 재배치 비용을 최소화하기 위해 타부서치(Tabu Search, TS) 알고리즘을 제안하였고, Jeong and Seo(2007)는 입출력점이 고정된 설비를 가정하여 설비간 직각거리와 물류량의 곱이 최소가 되는 설비배치를 찾기 위해 배치 알고리즘과 시뮬레이티드 어닐링 알고리즘으로 구성된 휴리스틱 방법을 제안하였다. Xiao *et al.*(2013)은 서로 다른 크기의 설비배치에 대하여 총 자재흐름 거리를 최소화하기 위해서 첫 번째 단계에서는 zone algorithm과 시뮬레이티드 어닐링 알고리즘을 적용하여 해를 도출하고, 두 번째 단계에서는 혼합정수계획법(Mixed Integer Programming, MIP)을 사용하여 설비의 위치와 방향을 조절함으로써 이전 단계의 해 개선 및 최적의 해를 도출하는 두 단계의 휴리스틱 알고리즘을 제안하였다.

스케줄링 문제를 다룬 기존의 연구는 다음과 같다. Salido *et al.*(2016)는 총 공정완료시간과 설비 가동을 위한 에너지 소모량을 고려하여 이를 최소화하기 위해 유전알고리즘을 제안하였고, Zhang and Wu(2011)는 총 가중 납기지연(total weighted tardiness)을 고려하여 이를 최소화하기 위해 시뮬레이티드 어닐링 알고리즘을 제안하였다. Zhu *et al.*(2010)은 적시생산시스템(Just-In-Time production system, JIT) 환경에서 재공품재고비용, 재고유지비용, 납기지연비용을 고려하여 비용을 최소화하기 위해 타부서치 알고리즘을 제안하였으며, Jia and Seo(2013)는 자원제약이 있는 프로젝트 스케줄링 문제에서 마지막 dummy activity의 완료시간이 최소화되는 프로젝트 스케줄링 해를 도출하기 위한 입자군집최적화(Particle Swarm Optimization, PSO) 기법을 제안하였다. Moon and Seo(2005)는 다중플랜트의 공정 계획과 스케줄링을 함께 고려한 APPS(Advanced Process

Planning and Scheduling) 모델을 제안하였으며 진화 알고리즘을 적용하여 총 공정완료시간을 최소화하였다. Zheng *et al.*(2014)은 설비 및 무인운반차(Automated Guided Vehicle, AGV)에 대한 스케줄링 문제를 동시에 해결하기 위하여 각각 서브 문제로 구성된 하나의 혼합정수계획법 모델을 제안하였고, 타부서치 방법을 이용하여 총 공정완료시간을 최소화하였다.

설비배치와 스케줄링 문제를 통합적으로 다룬 기존의 연구는 다음과 같다. Mallikarjuna *et al.*(2016)은 유연생산시스템(Flexible Manufacturing System, FMS)의 루프 레이아웃(loop layout) 최적화 문제를 다루었으며, 스케줄링의 목적함수인 총 공정완료시간을 제약조건으로 설정하여 총 운반비용(transportation cost)을 최소화하기 위해 유전알고리즘과 시뮬레이티드 어닐링 알고리즘을 각각 적용하여 비교하였다. Ranjbar and Razavi(2012)는 다품종 소량생산의 잡샵 환경 내에서 설비배치와 스케줄링 문제를 함께 다루고 있지만, 스케줄링의 목적함수인 총 공정완료시간만을 고려하여 이를 최소화하기 위해 scatter search 알고리즘을 제안하였다. Potts and Whitehead(2001)는 병목작업을 조절하여 처리량(throughput)을 최대화하고 자재운반거리를 최소화하기 위해 스케줄링 문제와 루프 레이아웃 문제를 함께 고려하였다. 위 연구에서는 이를 해결하기 위해 3단계를 거치는 모델을 제안하였는데, 첫 번째 단계에서는 병목현상이 일어나는 설비의 부하분산(load balancing)을 통해 작업량을 조절하고, 두 번째 단계에서는 이전 단계에서 수행한 부하 균형을 유지하면서 설비 간 이동을 최소화하며, 세 번째 단계에서는 루프 레이아웃에서 설비배치가 가능한 위치에 설비를 할당하여 제품이 만들어지는데 필요한 루프 회전 횟수를 최소화하는 3단계의 정수계획법(Integer Programming, IP) 모델을 제안하였다.

기존에 진행된 설비배치 문제와 스케줄링 문제에 대한 다수의 연구는 두 문제를 독립적으로 다루었으며, 설비배치 문제에 대한 최적해만을 도출하거나 주어진 설비배치에 대하여 스케줄링 문제에 대한 최적해만을 도출하였다(Ripon and Torresen, 2014). 또한 두 문제를 동시에 고려했다라도 하나의 문제를 제약조건으로 설정하고 다른 하나의 문제를 다루거나, 하나의 문제에 대한 목적함수만을 고려하여 문제를 다룬 연구들이 많다. 하지만, 스케줄링을 통한 전체 작업의 처리순서 및 작업별 공정순서는 생산 프로세스의 물동량과 자재흐름에 영향을 미치기 때문에 이를 고려한 설비배치 문제의 최적해를 구하는 것이 매우 중요하다. 반대로, 설비배치를 통해 결정되는 설비 간 이동 거리가 작업의 공간 간 이동으로 인한 시간지연을 발생시키고, 이는 전체 작업의 완료 시간에 영향을 미치기 때문에 이를 고려하여 스케줄링 최적해를 구하는 것 또한 매우 중요하다. 설비배치 문제와 스케줄링 문제에 대한 각각의 결과는 서로 상호영향을 미치며, 재구성제조시스템에서의 생산라인 모듈화는 설비 재배치를 용이하게 하기 때문에 제조시스템의 생산성과 비용효율성을 향상시키기 위해서는 이 두 요인을 함께 고려하여 최적해조합을 도출해야 한다. 따라서,

본 연구에서는 설비배치 문제와 스케줄링 문제를 동시에 고려하여 최적해조합을 찾는 방법을 제안한다.

### 3. 문제 정의

본 연구는 작업장 크기, 구역간 이동 거리 및 시간, 작업의 개수, 작업의 공정별 시간, 선행공정 제약조건이 주어지면 총 자재흐름거리와 전체 작업완료시간이 최소화되도록 설비배치 문제와 스케줄링 문제의 최적해조합을 찾는 문제를 다루며, 작업장의 각 구역에 설비를 배치하고 전체 작업에 대하여 작업 및 공정순서를 스케줄링한다. 이때 최적해조합을 찾는 과정에서 설비배치의 결과로 나오는 설비간 거리 및 공간간 이동시간 요소가 스케줄링에 영향을 미치고, 반대로 스케줄링 결과로 나오는 전체 작업의 처리 순서, 작업별 공정순서, 물동량의 변화가 설비배치에 영향을 미치므로 이러한 요인들을 반영하였다.

본 연구를 수행함에 있어서 다음의 사항들을 가정하였다.

- 1) 작업장 각 구역마다 하나의 설비만 배치될 수 있으며 설비간 거리 계산 시 각 설비 중심 사이의 직각거리를 고려하였다.
- 2) 모든 설비는 한 번에 하나의 작업만 처리할 수 있으며 작업 진행 중 다른 작업이 투입될 수 없다.
- 3) 모든 작업은 선행공정에 대한 제약이 있으며 작업 스케줄링 시 이를 기준으로 작업순서 및 공정순서를 변경함으로써 자재흐름과 물동량에 변화를 준다.
- 4) 작업 운반능력이 제약이 없으며 서로 다른 여러 반제품들은 동시에 운반가능하다.
- 5) 모든 작업의 공간간 운반속도는 단위시간으로 모두 일정하고, 이전 공정이 완료된 후 바로 다음 공정으로 이동한다.

본 연구의 목적함수는 설비배치와 스케줄링의 해조합(X, Y)에 대하여 설비배치의 목적함수값( $F_{FLP}$ )과 스케줄링의 목적함수값( $F_{JSSP}$ ) 각각에 대하여 가중치를 부여한 것으로 정의되며, 설비배치와 스케줄링의 해조합(X, Y)을 개선하여 목적함수값을 최소화하는 것으로써 식 (1)로 표현된다. 설비배치 문제의 목적함수값은 모든 작업에 대한 자재흐름거리의 총합으로 계산되며 설비 간의 물류량( $f_{mn}$ )과 거리( $d_{mn}$ )의 곱으로 표현된다(식 (2)). 이때 설비간의 거리( $d_{mn}$ )는 설비의 중심 간 직각거리로 계산하며 식 (3)으로 표현된다. 또한 스케줄링 문제의 목적함수값은 전체 작업이 완료되는 시간인 총 공정완료시간이며, 개별 작업들의 공정완료시간 중 최댓값으로 표현된다(식 (4)). 이때 개별 작업의 공정완료시간은 공정시간( $pt_{ij}$ )과 공간간 이동시간( $mt_{ij}$ )의 합으로 표현되며(식 (5)), 공간간 이동시간( $mt_{ij}$ )은 설비간 거리( $d_{mn}$ )에 비례한다.

$$\text{Minimize } F(X, Y, w_1, w_2) = \min(w_1 \times F_{JSSP}(X, Y) + w_2 \times F_{FLP}(X, Y)) \quad (1)$$

**Table 1.** Notations

Notations	Description
N	The number of jobs
M	The number of machines
$C_i$	Completion time of $i^{th}$ job
$pt_{ij}$	Processing time of $j^{th}$ operation in $i^{th}$ job
$mt_{ij}$	Moving time between $j^{th}$ and $(j+1)^{th}$ operation in $i^{th}$ job
$f_{mn}$	Volume of material flow between machine m and n
$d_{mn}$	Distance between machine m and n
$w_k$	Objective weight for $k^{th}$ objective. Subscript 1 and 2 refer to makespan and total material handling distance respectively
$(x_m, y_m)$	Centroid coordinates of machine m
$X_0, Y_0$	Initial solution of FLP, JSSP
$X_{cur}, Y_{cur}$	Current solution of FLP, JSSP
$X^*, Y^*$	Best solution of FLP, JSSP
$\bar{X}, \bar{Y}$	Best solution among the neighborhood solutions of FLP, JSSP
$TL_{FLP/JSSP}$	Tabu list of FLP, JSSP
$F_{FLP/JSSP}$	Objective function of FLP, JSSP

$$F_{FLP} = \sum_{m=1}^M \sum_{n=1}^M f_{mnn} \times d_{mnn} \quad (2)$$

$$d_{mnn} = |x_m - x_n| + |y_m - y_n| \quad (3)$$

$$F_{JSSP} = \max_{i=1}^N [C_i] \quad (4)$$

$$C_i = \sum_{j=1}^M (pt_{ij} + mt_{ij}) \quad (5)$$

본 연구에서 제안하는 방법에 사용되는 기호와 변수들을 정리하면 <Table 1>과 같다.

### 4. 해 표현 방법

본 연구에서 사용되는 해는 설비배치와 스케줄링 두 부분으로 구성되어있으며 정수 문자열(string of integers) 형태로 표현된다. 해의 설비배치 부분에서는 M개의 설비에 대하여 1부터 M까지의 숫자가 한 번씩 입력되며, 입력된 숫자의 순서는 작업장의 첫 번째 영역부터 배치되는 설비의 번호를 의미한다. 또한 해의 스케줄링 부분에서는 Ripon and Torresen(2014)에서 제안한 작업 및 공정 순서 중심의 해 표현 방법을 적용하였으며, 문자열마다 (작업번호, 공정번호) 형태로 작업의 번호와 각 작업별 공정번호를 하나의 쌍(pair)으로 입력한다. 이 방법은 다양한 순서조합으로 합리적인 스케줄링 해를 생성할 수 있고, 생성된 스케줄링 해에 대하여 해의 왼쪽부터 차례로 작업 및 공정 순서에 대한 디코딩을 통해 간트 차트(Gantt Chart)로 나타내어 간단하게 유효한 스케줄을 생성한다. 작업번호란

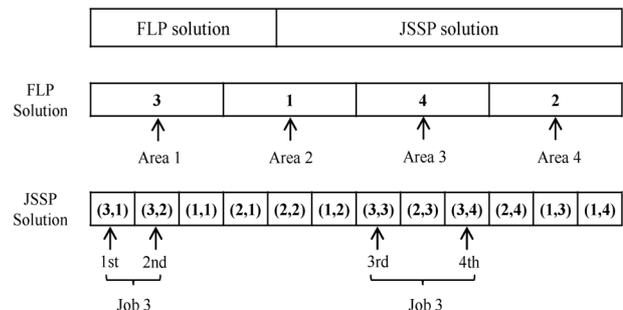
에 입력되는 숫자는 N개의 작업에 대하여 1부터 N까지의 숫자가 M개의 설비개수만큼 반복해서 입력되며, 공정번호란에는 M개의 설비에 대하여 1부터 M까지의 숫자가 입력된다. 따라서 해의 스케줄링 부분에 입력된 (작업번호, 공정번호) 쌍의 순서는 작업의 처리순서와 해당 작업의 공정순서를 의미한다.

해 표현 방법에 대한 예시를 통해 추가적인 설명을 진행한다. <Table 2>는 4대의 설비와 3개의 작업에 대하여 작업별 담당 설비 및 공정시간에 대한 정보를 나타낸 것이다. 본 예시에서 작업장은 각 구역의 가로와 세로의 길이가 10m인 동일한 크기의 4개의 영역(2x2)으로 구성되도록 설정하였으며, 각 작업에 대하여 이전 공정을 다음 공정의 선행공정으로 가정하여 첫 번째 공정부터 마지막 공정까지 차례로 진행하는 것으로 설정하였다.

<Figure 1>은 설비배치와 스케줄링 문제에 대하여 도출 가능한 하나의 해 표현에 대한 예시이다. 해의 설비배치 부분에 입력된 첫 번째 숫자 3은 작업장의 첫 번째 영역에 3번 설비가 배치된 것을 의미하며 그 다음 두 번째 영역부터 차례로 1번, 4번, 2번 설비가 배치되는 것을 의미한다. 해의 스케줄링 부분에 입력된 첫 번째 쌍 (3,1)은 3번 작업의 첫 번째 공정을 수행하는 것을 의미하며, 두 번째 쌍 (3,2)는 3번 작업의 두 번째 공정을 수행하는 것을 의미한다. 그 다음에 입력된 쌍들도 이와 동일하게 처리되는 작업번호와 해당 작업에 대한 공정번호를 의미한다. 이에 대한 결과로, <Figure 2>는 각 영역에 설비들을 배치한 결과를 그림으로 나타낸 것이며, <Figure 2>에서 그려진 점선은 스케줄링 결과로 인한 각 작업의 처리순서를 반영하여 전체 작업의 자재흐름을 나타낸 것이다. 또한 <Figure 3>은 스케줄링 결과에 대한 작업순서를 간트 차트로 나타낸 것으로, 이때 그래프에 그려진 각 작업의 막대에는 공정시간과 공정간 이동시간이 포함되어 있다.

**Table 2.** An Example of a 3x4 JSSP

	Machine/Processing time			
	Op1	Op2	Op3	Op4
Job1	1/3	3/4	2/9	4/2
Job2	1/4	2/5	3/3	4/5
Job3	2/4	3/6	4/4	1/3



**Figure 1.** A Possible Solution Representation of FLP and JSSP

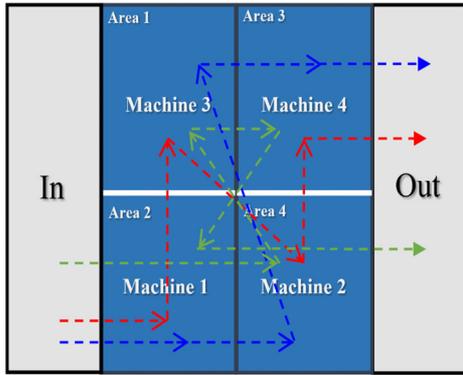


Figure 2. Facility Layout of FLP Solution Example

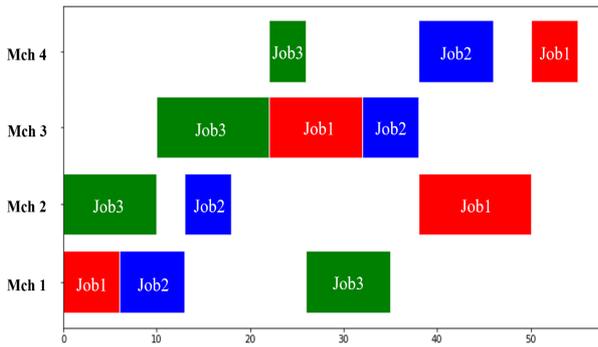


Figure 3. Gantt Chart of JSSP Solution Example

## 5. 통합 알고리즘

본 연구에서 제안하는 통합 알고리즘은 설비배치 문제와 스케줄링 문제를 동시에 고려하여 최적해조합을 도출하는 알고리즘으로 통합 알고리즘의 절차는 크게 두 단계로 나뉘어진다. 본 장에서는 이에 대해서 설명하며 제5.1절에서는 통합 알고리즘의 초기 절차로 목적함수의 가중치를 도출하기 위한 평균가중(mean weight) 방법을, 제5.2절에서는 통합 알고리즘의 첫 번째 절차인 임의 탐색(random search) 방법을, 제5.3절에서는 통합 알고리즘의 두 번째 절차인 타부 서치(tabu search) 방법을, 제5.4절에서는 통합 알고리즘의 전체 절차를 설명한다.

### 5.1 평균 가중 방법

본 연구에서는 설비배치 목적함수값과 스케줄링 목적함수값 각각에 대한 가중치 계산을 위하여 Singh and Singh(2010)에서 제안한 평균 가중 방법을 활용하였으며, 평균 가중 방법은 서로 다른 개별 목적함수값들이 동일한 중요도를 가지는 것으로 가정한다. 전체 목적함수값 계산 시 개별 목적함수값에 대하여 가중치를 적용하는 이유로는 입력되는 설비간 이동거리, 작업별 공정시간 데이터의 상대적 크기에 따라 결과값에 영향을 주기 때문이다. 예를 들어 설비간 이동거리 정보가 공정시간 정보 대비 상대적으로 클 경우, 설비배치 목적함수

값이 상대적으로 크게 도출이 되며 이에 따라 한쪽으로 치우친 결과값이 도출되게 된다. 이에 대한 보완과 서로 다른 개별 목적함수값들을 함께 고려하기 위해 평균 가중 방법에서는 각 데이터에 대한 정규화를 진행하고, 정규화된 데이터에 대한 개별 목적함수값을 도출하여 각각에 대한 가중치를 계산한다. 평균 가중 방법을 적용하여 가중치를 계산하는 절차는 다음과 같다.

- Step 1. 설비배치 및 스케줄링 관련 데이터 각각에 대하여 최소-최대 정규화를 진행한다.
- Step 2. 정규화 된 데이터를 사용하며 초기 설정된 설비배치 해와 스케줄링 해에 대한 개별 목적함수값( $F_{JSSP}$ ,  $F_{FLP}$ )을 도출한다.
- Step 3. 식 (6), (7)에 따라 개별 목적함수에 대한 가중치( $w_1$ ,  $w_2$ )를 계산하여 적용한다.

$$w_1 = \frac{F_{JSSP}}{F_{JSSP} + F_{FLP}} \quad (6)$$

$$w_2 = \frac{F_{FLP}}{F_{JSSP} + F_{FLP}} \quad (7)$$

### 5.2 임의 탐색 방법

통합 알고리즘의 첫 번째 단계에서는 임의 탐색 방법을 적용하여 해공간의 다양한 해 탐색을 진행한다. 첫 번째 단계에서 임의 탐색 방법을 적용하는 이유는 다음과 같다. 먼저, 다음 단계에서 적용할 타부서치 방법의 경우 지역탐색에는 강하지만 해공간 탐색능력이 비교적 약한 점을 보완하기 위해서이다. 그리고 타부서치 방법의 성능은 초기해의 영향을 받으므로 해공간의 임의 탐색을 통해 가장 개선된 해를 두 번째 단계의 초기해로 설정하기 위함이다(Geyik and Cedimoglu, 2004). 마지막으로 임의 탐색 방법은 제한된 시간 내에 넓은 범위의 해공간을 탐색하는데 유용하게 활용될 수 있기 때문이다(Bergstra and Bengio, 2012). 따라서 본 연구에서는 통합 알고리즘의 첫 번째 단계에서 넓은 해공간을 탐색하기 위해 임의 탐색 방법을 적용하고, 가장 개선된 목적함수값 갖는 설비배치와 스케줄링의 해조합을 도출하여 두 번째 단계의 초기해로 설정한다.

통합 알고리즘의 첫 번째 단계인 임의 탐색 방법에 대한 절차는 다음과 같다.

- Step 1. 설비배치와 스케줄링 문제의 초기해를 현재해로 설정하고 초기해조합의 목적함수값  $F(X_{cur}, Y_{cur}, w_1, w_2)$ 를 계산한다.
- Step 2. 설비배치 영역과 스케줄링 영역을 임의로 탐색하여 새로운 해  $X_{new}, Y_{new}$ 를 생성한다.
- Step 3. <Figure 4>와 같이 설비배치와 스케줄링의 현재 해와

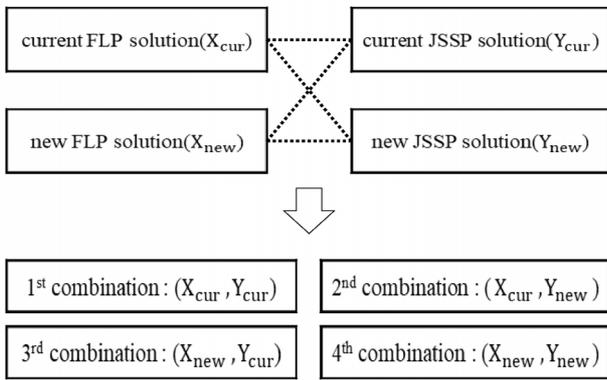


Figure 4. Solution Combinations of FLP and JSSP

임의로 생성된 새로운 해에 대하여 4가지 해조합을 구성하고, 해조합들의 목적함수값  $F(X_{cur}, Y_{cur}, w_1, w_2)$ ,  $F(X_{cur}, Y_{new}, w_1, w_2)$ ,  $F(X_{new}, Y_{cur}, w_1, w_2)$ ,  $F(X_{new}, Y_{new}, w_1, w_2)$ 를 계산한다.

- Step 4. 이전 단계에서 계산된 해조합들의 목적함수값을 비교한다.
- Step 5. 가장 개선된 목적함수값의 해조합을 현재해로 갱신한다.
- Step 6. 초기 지정한 횟수만큼 반복 수행을 완료했다면 프로세스를 종료하고, 그렇지 않다면 <Step 2>로 돌아간다.
- Step 7. 설비배치와 스케줄링 문제의 가장 개선된 해조합을 도출한다.

다음은 이에 대한 pseudo code이다.

**1<sup>st</sup> Phase of Integrated algorithm : Random Search method**

1. **Set** Initial FLP solution is given;
2. Initialize JSSP solution randomly;
3. Objective weights for FLP and JSSP are calculated;
4. Update FLP and JSSP initial solution to current;
5. **Repeat**
6. **for**  $iter_1 = 1$  to iteration;
7. Generate new FLP and JSSP solutions randomly;
8. Compare the four solution combinations consisting of current and new solution;
9. Find the best combination and update to current;
10. **End**
11. **Return** best solution combination

**5.3 타부 서치 방법**

통합 알고리즘의 두 번째 단계에서는 타부서치 방법을 활용하였다. 타부서치 방법은 Glover and Laguna(1998)에 의해 처음 제안되었으며, 조합최적화(combinatorial optimization) 문제를 해결하기 위해 고안된 인간의 기억과정을 모방한 메타휴리스틱 방법이다. 타부서치 방법의 주요 특징으로는 세 가지가 있다. 먼저, 현재해의 이웃에서 후보해를 탐색하여 단일해를 개선해나가는 점 기반 이웃탐색(point based neighborhood

search) 기법으로 특정 지역의 지역 최적해를 도출하는데 강점이 있다. 다음으로, 해의 탐색 과정에서 타부 리스트를 이용하여 타부 리스트에 해당되는 이웃해의 재탐색을 금지함으로써 해의 순환을 방지한다. 마지막으로, 해가 개선되지 않아도 해의 이동을 허락하거나 타부 리스트에 해당되는 이웃해 일지라도 열망 조건(aspiration criterion)에 의해 해의 이동을 허락함으로써 해의 수렴을 방지한다.

5.3.1 타부서치 파라미터

타부서치 방법의 주요 파라미터로는 타부목록의 크기(size of tabu list), 이웃 후보해(neighborhood candidate list) 생성 개수, 알고리즘의 반복 횟수가 있다. 타부서치 방법의 파라미터 설정값에 따라 해공간의 탐색정도와 지역최적을 위한 이웃 탐색정도가 달라지며 알고리즘의 성능도 달라진다. 이에 본 연구에서는 Ponnambalam *et al.*(2000)을 참조하여 타부목록의 크기와 이웃 후보해 생성 개수를 설정하였고, 문제사이즈가  $n$ 인 경우에 타부목록의 크기는  $(\sqrt{n})$ 으로 이웃 후보해 생성 개수는  $(n-1)$ 로 지정하였다. 또한 반복 횟수는 첫 번째 단계와 동일하게 설정하였다.

5.3.2 이웃 후보해 생성 방법

단일해 기반의 타부서치 방법은 현재 해의 지역변환(local transformation)을 통해 이웃 후보해 생성이 이루어지며, 지역변환의 방법으로는 교환, 삽입, 역순 이동 등이 있다. 본 연구에서는 현재 해의 설비배치 혹은 스케줄링 부분 각각에 대하여 임의로 두 인자를 선택하고, 선택된 두 인자를 서로 교환하는 쌍대교환(pairwise interchange) 방법을 사용하여 이웃 후보해를 생성한다. <Figure 1>의 해의 설비배치 부분에 쌍대교환 방법을 적용한 예시는 <Figure 5>와 같다.

5.3.3 타부서치 절차

통합 알고리즘의 두 번째 단계에서는 이전 단계에서 도출된 해조합에 대하여 지역최적 해조합을 도출하기 위해 초기 설정한 반복 횟수만큼 설비배치 문제와 스케줄링 문제에 타부서치 방법을 교대로 적용하면서 이웃 탐색을 진행하고 지역 최적해

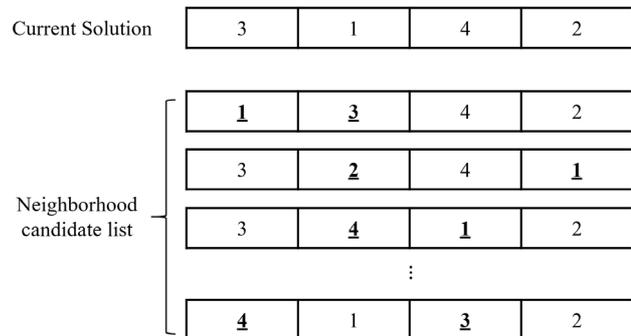


Figure 5. Pairwise Interchange Method for Generating Neighborhoods

조합을 도출한다. 설비배치와 스케줄링 문제에 적용하는 타부서치 방법은 동일하므로 다음은 스케줄링 문제에 타부서치 방법을 적용한 것만 기술하며, 이에 대한 절차는 다음과 같다.

- Step 1. 스케줄링의 문제사이즈(n)를 고려하고 쌍대교환 방법을 적용하여 현재의 스케줄링 해에 대한 (n-1)개의 이웃 후보해  $\{Y_1, \dots, Y_{n-1}\}$ 를 생성한다.
- Step 2. 생성된 (n-1)개의 이웃 후보 해조합 중 가장 개선된 목적함수값  $F(X_{cur}, \bar{Y}, w_1, w_2)$ 를 갖는 이웃해  $\bar{Y}$ 를 도출한다.
- Step 3. 타부 리스트를 확인하여 이웃해  $\bar{Y}$ 로의 갱신 가능 여부를 확인한다.
- Step 4.  $\bar{Y}$ 로의 갱신이 가능할 경우 <Step 9>로, 갱신이 불가능할 경우 다음 단계로 이동한다.
- Step 5. 현재 해조합의 목적함수값  $F(X_{cur}, Y_{cur}, w_1, w_2)$ 와 도출된 이웃 후보 해조합의 목적함수값  $F(X_{cur}, \bar{Y}, w_1, w_2)$ 를 비교한다.
- Step 6. 이웃 후보 해조합의 목적함수값이 현재 해조합의 목적함수값보다 더 좋다면 열망 조건에 의해 <Step 10>으로 이동하고, 그렇지 않다면 다음단계로 이동한다.
- Step 7. 이웃 후보해에서  $\bar{Y}$ 를 제거하고 이웃 후보해에서 그 다음으로 가장 좋은 목적함수값을 갖는 이웃해  $\bar{Y}$ 를 도출한다.
- Step 8. 도출된 이웃해에 대하여 타부 리스트를 확인하고 갱신이 가능할 경우 다음 단계로, 그렇지 않을 경우 이전 단계로 이동한다.
- Step 9. 타부 리스트를 갱신한다. 이때 타부 리스트가 모두

채워져 있을 경우 가장 오래된 타부목록을 제거하고 타부 리스트를 갱신한다.

Step 10. 이웃해  $\bar{Y}$ 를 현재 해로 갱신한다.

Step 11. 업데이트 된 현재 해조합  $X_{cur}, Y_{cur}$ 을 도출한다.

다음은 이에 대한 pseudo code이며, 설비배치 문제에도 동일한 절차로 타부서치 방법을 적용하였다.

**2<sup>nd</sup> Phase of Integrated algorithm : Tabu Search method**

```

1. Generate the (n-1) numbers of JSSP solution's neighborhoods;
2. Find the best solution  $\bar{Y}$  among neighborhoods  $\{Y_1, \dots, Y_{n-1}\}$ ;
3. If  $\bar{Y}$  is in  $TL_{JSSP}$ ;
4.   If  $F(X_{cur}, \bar{Y}, w_1, w_2) < F(X_{cur}, Y_{cur}, w_1, w_2)$ ; // Aspiration criterion is satisfied
5.     update  $\bar{Y}$  to  $Y_{cur}$ ;
6.   else;
7.     While ( $\bar{Y}$  is in  $TL_{JSSP}$ ) do;
8.       delete  $\bar{Y}$  from the neighborhood solution;
9.       find the next best neighborhood solution  $\bar{Y}$ ;
10.    End
11.    update  $\bar{Y}$  to  $Y_{cur}$ ;
12.    If  $TL_{JSSP}$  is full;
13.      remove the oldest tabu in  $TL_{JSSP}$ ;
14.    update  $TL_{JSSP}$ ;
15.  End
16. else;
17.   update  $\bar{Y}$  to  $Y_{cur}$ ;
18.   If  $TL_{JSSP}$  is full;
19.     remove the oldest tabu in  $TL_{JSSP}$ ;
20.   update  $TL_{JSSP}$ ;
21. End
22. Return  $X_{cur}, Y_{cur}$ 
    
```

**5.4 통합 알고리즘 절차**

본 연구에서 제안하는 통합 알고리즘의 절차는 다음과 같다. 먼저 통합 알고리즘의 초기 절차로 설비배치와 스케줄링 개별

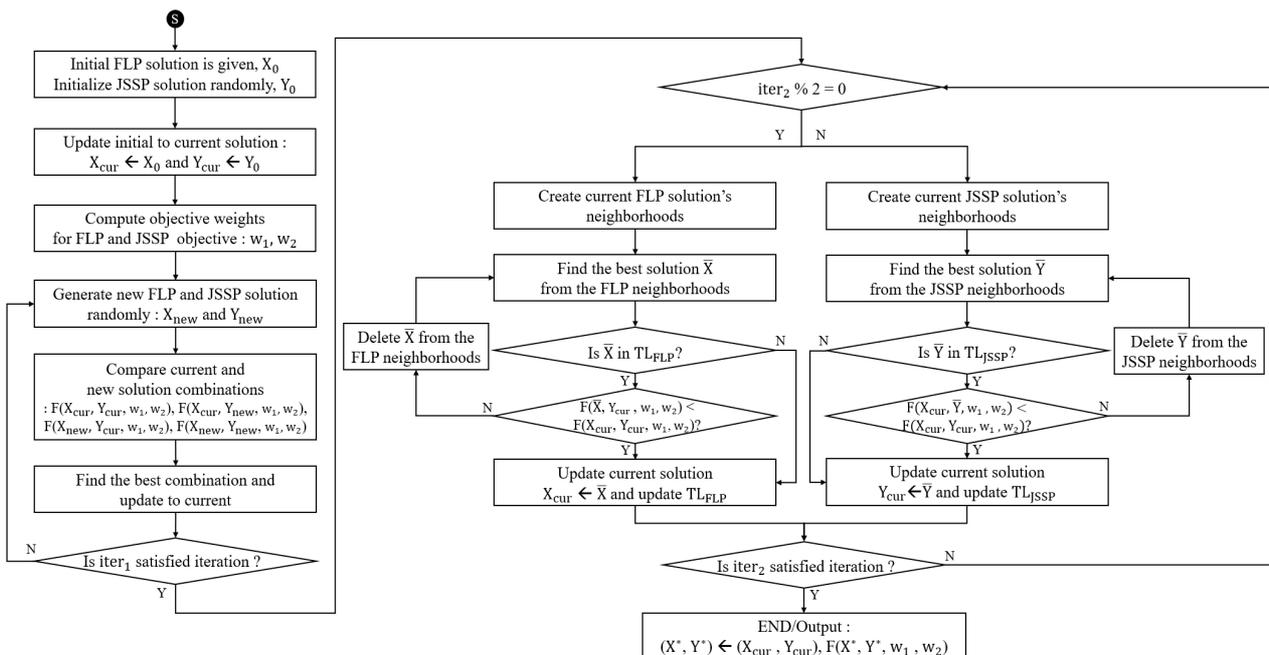


Figure 6. Flow Chart of Integrated Algorithm

문제를 통합적으로 고려하기 위해 평균 가중 방법을 사용하여 각 목적함수에 대한 가중치를 계산한다. 다음으로 다양한 해공간의 탐색과 타부서치 방법의 초기해 영향을 개선하기 위해 제5.2절에서 기술한 임의의 탐색 방법을 활용하여 임의의 해공간을 탐색하고 개선된 해조합(solution combination)을 도출한다. 마지막으로 제5.3절에서 기술한 지역탐색을 강화하기 위해 이전 단계에서 도출된 해조합에 대하여 설비배치 및 스케줄링 각각에 교대로 타부서치 방법을 적용하여 이웃탐색을 진행하고 개선된 지역최적(local optimum) 해조합을 도출한다.

<Figure 6>은 최종적으로 본 연구에서 제안하는 통합 알고리즘에 대한 전체 Flow Chart이다.

### 6. 실험 및 결과

제6장에서는 본 연구에서 제안한 통합 알고리즘의 성능 검증 실험이 수행되며, 설비배치 문제와 스케줄링 문제를 동시에 고려하는 경우와 주어진 설비배치에 대하여 스케줄링만 고려하는 경우에 대한 실험 결과를 비교분석한다.

본 연구의 실험에서는 제안한 통합 알고리즘의 성능 검증을 위해 설비배치와 스케줄링 각각의 문제 사이즈를 <Table 3>과 같이 설정하여 실험을 진행하였다. 설비배치의 경우 각 구역의 가로와 세로의 길이가 20m인 동일한 크기의 9개의 영역(3×3)으로 구성된 작업장을 설정하였으며, 모든 작업은 <Figure 7>과 같이 선행공정에 대한 제약이 있는 것으로 설정하였다. 스케줄링 시 <Figure 7>을 기준으로 작업순서 및 공정순서를 변경함으로써 자재흐름과 물동량에 변화를 주었다. 스케줄링의 경우 다양한 작업사이즈 및 문제에 대하여 실험을 진행하기 위해 Or-library를 참조하였으며, abz5, abz6, orb03, la25, la27 문제의 각 작업 및 공정에 대한 수행시간과 설비 데이터를 활용하였다(Beasley, 1990).

본 연구의 비교 실험으로는 설비배치가 한 번 이루어지고 나면 단기간 내 설비 재배치가 이루어지기 어려운 기존의 제조시스템을 가정하였다. 따라서 비교실험에서 설비배치는 정해진 것으로 가정하여 변경되지 않으며, Jawahar *et al.*(1998)에서 제안한 유전알고리즘 방법을 적용하여 스케줄링 최적화를 수행하였다. 비교 실험의 조건으로 작업장 크기, 구역간 이동거리, 작업의 개수, 작업의 공정별 시간, 선행공정 제약조건,

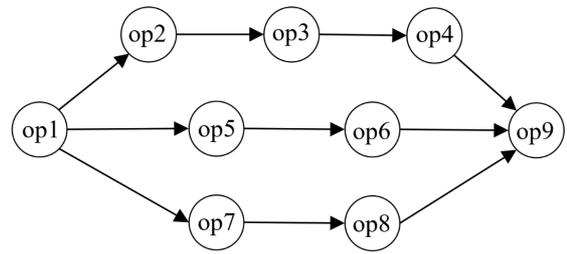


Figure 7. Operation Precedence Network

abz5	5	1	8	9	2	3	6	7	4
abz6	9	5	2	6	8	4	1	3	7
orb03	2	6	1	3	5	4	7	8	9
la25	3	5	9	2	4	6	8	7	1
la27	5	4	1	9	3	7	6	8	2

Figure 8. Initial FLP Solution

반복 횟수 모두 본 연구에서 제안하는 통합 알고리즘의 실험 조건과 동일하다. 또한, 초기 설비배치 해의 경우 제5.3절의 타부서치 방법을 적용하여 초기 설정된 작업순서 조건을 고려한 최적의 설비 배치가 이루어진 것으로 가정하였으며, 휴리스틱 알고리즘 특성상 초기 설정된 설비배치와 스케줄링 해에 따라 결과값이 왜곡될 수 있으므로 비교 실험과 제안 실험에서 동일한 초기해를 설정하여 실험을 진행하였다. 각 문제에 대하여 설정된 초기 설비배치는 <Figure 8>과 같다.

본 연구의 모든 문제에 대하여 제안 알고리즘의 실험과 비교 알고리즘의 실험 모두 30번씩 반복 수행하였다. 실험의 결과로는 목적함수값의 평균값, 목적함수값의 표준편차, 가장 좋은 최적해조합의 목적함수값, 가장 좋은 최적해조합의 가중치, 목적함수 평균값의 개선율을 도출하였다. 그 결과 모든 작업사이즈에 대하여 본 연구에서 제안한 통합 알고리즘의 경우 목적함수 평균값, 표준편차, 가장 좋은 최적해조합의 목적함수값 모두 비교 실험 대비 더 우수한 결과값이 도출되었다. 제안하는 통합 알고리즘은 목적함수 평균값에서 3.82~5.21%의 개선율을 가지는 해조합 결과를 도출하였다. 또한 가장 좋은 최적해조합의 목적함수값에서 3.87~7.20%의 개선율을 가지는 해조합을 도출하였으며, 설비배치와 스케줄링 각각의 목적

Table 3. Size of FLP and JSSP for Each Instance

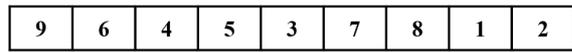
Problem number	Instance	# of jobs	FLP size (# of machines)	JSSP size (# of machines × # of jobs)
1	abz5	10	9	9 × 10
2	abz6	10	9	9 × 10
3	orb03	10	9	9 × 10
4	la25	15	9	9 × 15
5	la27	20	9	9 × 20

Table 4. Experimental Results

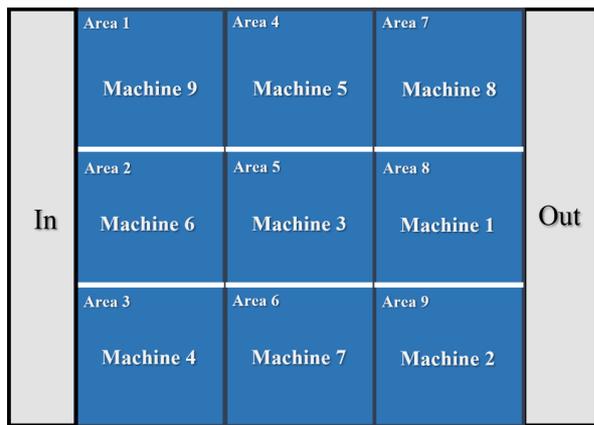
Job size	Instance	Comparison Algorithm (JSSP only)			Proposed Algorithm (Integrated FLP&JSSP)			Weights of Best Combination (W <sub>FLP</sub> /W <sub>JSSP</sub> )	Average Improvement (%)	Best Combination Improvement (%)
		Average	Standard Deviation	Value of Best Combination (F <sub>FLP</sub> /F <sub>JSSP</sub> )	Average	Standard Deviation	Value of Best Combination (F <sub>FLP</sub> /F <sub>JSSP</sub> )			
10	abz5	4449	96.51	4268 (1760/2508)	4241.91	82.06	4103 (1700/2403)	(0.58/0.42)	4.65	3.87
10	abz6	3998.37	93.47	3904 (1890/2014)	3821.31	70.36	3623 (1800/1823)	(0.53/0.47)	4.43	7.20
10	orb03	3847.32	81.98	3688 (1640/2140)	3700.47	95.40	3518 (1580/1938)	(0.51/0.49)	3.82	4.61
15	la25	5532.47	97.65	5378 (2820/2558)	5315.68	88.83	5142 (2640/2502)	(0.59/0.41)	3.92	4.39
20	la27	7597.27	114.89	7369 (3810/3559)	7201.67	95.90	6979 (3680/3299)	(0.57/0.43)	5.21	5.29

함수값에서도 개선된 결과값이 도출되었다. 이에 대한 실험 결과는 <Table 4>에 정리하였다.

또한 <Figure 9>와 <Figure 10>은 abz6 문제에 대하여 통합 알고리즘을 적용하고 도출된 가장 좋은 최적해조합의 설비배치 결과와 작업순서에 대한 스케줄링 결과를 그림과 간트차트로 나타낸 것이다.



(a)

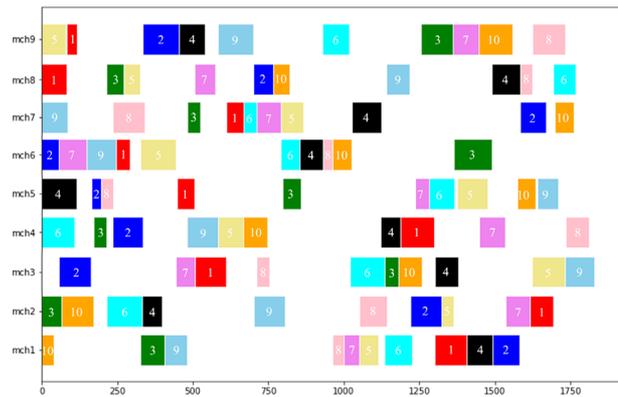


(b)

Figure 9. (a) FLP Solution and (b) Facility Layout of Best Combination in Instance abz6

(4,1)	(10,1)	(3,1)	(10,5)	(2,1)	(1,1)	(6,1)	(2,2)	(2,5)	(8,1)	(7,1)	(5,1)	(3,7)	(9,1)	(9,2)
(1,2)	(3,2)	(5,2)	(3,5)	(1,3)	(8,2)	(6,5)	(9,5)	(3,3)	(5,7)	(2,7)	(7,2)	(4,7)	(2,3)	(7,3)
(4,2)	(1,5)	(1,7)	(1,6)	(6,6)	(9,6)	(9,3)	(9,7)	(8,5)	(5,3)	(7,4)	(2,6)	(6,2)	(4,3)	(6,7)
(10,2)	(8,3)	(8,6)	(10,3)	(10,6)	(7,5)	(8,7)	(3,6)	(6,3)	(5,8)	(5,5)	(6,8)	(3,4)	(7,6)	(4,5)
(4,4)	(9,4)	(10,4)	(3,8)	(3,9)	(2,4)	(7,7)	(6,4)	(5,6)	(1,4)	(4,6)	(1,8)	(7,8)	(5,4)	(4,8)
(4,9)	(8,8)	(10,7)	(2,8)	(2,9)	(7,9)	(10,8)	(1,9)	(5,9)	(8,4)	(8,9)	(9,8)	(10,9)	(6,9)	(9,9)

(a)



(b)

Figure 10. (a) JSSP Solution and (b) Gantt chart of Best Combination in Instance abz6

### 7. 결론

소비자의 니즈가 다양해지고 개별제품의 생산요구량이 증대됨에 따라 높은 유연성과 생산성을 동시에 제공할 수 있는 제조시스템이 요구되고 있다. 또한 빠른 시장변화에 대응하기 위해서는 생산라인의 모듈화를 통해 설비배치의 변경을 용이하게 함으로써 다양한 제품을 한정된 자원 내에서 효율적으로 생산할 수 있어야 한다.

본 연구에서는 재구성제조시스템(RMS)의 생산라인 모듈화에 대한 가치를 보여주는 데 의미가 있으며 모듈화로 인한 설비배치 변경이 용이한 경우를 가정하였다. 한편 설비배치가 결정되고 나서 장기간 변경이 이루어지지 않는 기존 제조시스템과 달리 재구성제조시스템의 경우 보다 효율적인 설비배치의 재구성이 가능하며 설비배치 문제와 스케줄링 문제에 대한 각각의 결과는 서로 상호영향을 미치기 때문에 이 두 가지 문

제를 통합적으로 고려한 방법을 제안하였으며, 제안된 통합 알고리즘을 적용한 실험과 스케줄링 요소만 고려하여 유전 알고리즘을 적용한 비교실험을 진행하였다.

결과적으로 설비배치와 스케줄링 두 요소를 통합적으로 고려하여 재구성(reconfigure)할 경우 전체 작업의 물류흐름 측면과 총 공정완료시간 측면에 대하여 목적함수값의 평균이 3.82~5.21% 개선된 결과값을 도출하였다. 또한 가장 좋은 최적해 조합의 목적함수값이 3.87~7.20% 개선되었으며 설비배치와 스케줄링 각 요소에 대하여 개선된 결과값을 도출하였다. 현재도 다수의 연구에서 설비배치와 스케줄링을 독립적으로 다루고 있지만, 설비배치와 스케줄링은 서로의 결과에 영향을 미친다. 이는 설비재구성으로 인해 설비간 거리 및 재공품 이동 시간이 변경되고 스케줄링으로 인해 물동량 및 자재흐름이 변경되기 때문이며, 설비배치와 스케줄링 요소를 동시에 고려하고 개선하는 것이 전체 생산프로세스의 전역 최적해를 도출하는데 필수적이다.

이로써 본 연구에서 제안하는 통합 알고리즘은 모듈화된 공장에서 일부 모듈의 생산설비 고장 혹은 새로운 생산설비가 도입되어 설비배치 재구성을 해야 하거나 생산 제품이 변경되어 시스템을 재구성해야 하는 경우, 스케줄링뿐만 아니라 설비의 재배치도 함께 고려하여 최적 솔루션을 도출하고 이를 기반으로 의사결정을 수행하는 데 도움이 될 것이다. 이 외에도 설비배치 혹은 스케줄링 변경 시 두 요소를 통합적으로 고려하여 여러 생산시나리오를 테스트하고 실제 현장 상황에 가장 적합한 솔루션을 도출하여 의사결정을 수행해야 할 때 효과적으로 사용될 수 있을 것이다.

실제 공장에서 물류이동은 AGV, 지게차(forklift)와 같은 운송수단에 의해 이루어지며, 운송수단의 대수, 운송가능 경로, 처리해야할 작업의 양 등에 따라 운반능력에 제약이 생긴다. 또한 설비 레이아웃 및 생산 프로세스 재구성 시 자재흐름, 총 공정완료시간 등이 개선되지만 추가적인 비용과 시간 발생으로 인한 트레이드오프(trade-off)가 발생하게 된다. 따라서 향후 연구에는 운송수단에 대한 제약뿐만 아니라 재구성으로 인해 발생하는 추가적인 비용 및 시간 정보가 고려된 추가적인 연구가 필요할 것으로 보인다.

## 참고문헌

- Applegate, D. and Cook, W. (1991), Computational Study of the Job-Shop Scheduling Problem, *ORSA Journal on Computing*, **3**(2), 149-156.
- Beasley, J. E. (1990), OR-Library: Distributing Test Problems by Electronic Mail, *The Journal of the Operational Research Society*, **41**(11), 1069-1072.
- Bergstra, J. and Bengio, Y. (2012), Random Search for Hyper-Parameter Optimization, *Journal of Machine Learning Research*, **13**, 281-305.
- De Carlo, F., Arleo, M. A., Borgia, O., and Tucci, M. (2013), Layout Design for a Low Capacity Manufacturing Line: A Case Study, *International Journal of Engineering Business Management*, **5** (SPL.ISSUE), 1-10.
- Drira, A., Pierreval, H., and Hajri-Gabouj, S. (2007), Facility Layout Problems: A survey, *Annual Reviews in Control*, **31**(2), 255-267.
- El-Baz, M. A. (2004), A Genetic Algorithm for Facility Layout Problems of Different Manufacturing Environments, *Computers and Industrial Engineering*, **47**(2-3), 233-246.
- Geyik, F. and Cedimoglu, I. H. (2004), The Strategies and Parameters of Tabu Search for Job-Shop Scheduling, *Journal of Intelligent Manufacturing*, **15**(4), 439-448.
- Glover, F. and Laguna, M. (1998), *Tabu Search* (Vol. 3), Kluwer Academic.
- Jawahar, N., Aravindan, P., and Ponnambalam, S. G. (1998), A Genetic Algorithm for Scheduling Flexible Manufacturing Systems, *International Journal of Advanced Manufacturing Technology*, **14**(8), 588-607.
- Jeong, D. and Seo, Y. (2007), Heuristic Algorithm for Facility Layout Design with Fixed Input and Output Points, *IE Interfaces*, **20**(2), 121-132.
- Jia, Q. and Seo, Y. (2013), An Improved Particle Swarm Optimization for the Resource-Constrained Project Scheduling Problem, *International Journal of Advanced Manufacturing Technology*, **67**(9-12), 2627-2638.
- Koren, Y., Gu, X., and Guo, W. (2017), Reconfigurable Manufacturing Systems: Principles, Design, and future Trends, *Frontiers of Mechanical Engineering*, **13**(2), 121-136.
- Kulturel-Konak, S. (2007), Approaches to Uncertainties in Facility Layout Problems: Perspectives at the Beginning of the 21st Century, *Journal of Intelligent Manufacturing*, **18**(2), 273-284.
- Lameche, K., Najid, N. M., Castagna, P., and Kouiss, K. (2017), Modularity in the Design of Reconfigurable Manufacturing Systems, *IFAC-PapersOnLine*, **50**(1), 3511-3516.
- Mallikarjuna, K., Veeranna, V., and Reddy, K. H. (2016), A New Meta-Heuristics for Optimum Design of Loop Layout In Flexible Manufacturing System with Integrated Scheduling, *International Journal of Advanced Manufacturing Technology*, **84**(9-12), 1841-1860.
- McKendall, A. R. and Liu, W. H. (2012), New Tabu Search Heuristics for the Dynamic Facility Layout Problem, *International Journal of Production Research*, **50**(3), 867-878.
- Moon, C. and Seo, Y. (2005), Evolutionary Algorithm for Advanced Process Planning and Scheduling in a Multi-Plant, *Computers and Industrial Engineering*, **48**(2), 311-325.
- Nasab, H. H. and Mobasheri, F. (2013), A Simulated Annealing Heuristic for the Facility Location Problem, *International Journal of Mathematical Modelling and Numerical Optimisation*, **4**(3), 210-224.
- Ponnambalam, S. G., Aravindan, P., and Rajesh, S. V. (2000), A Tabu Search Algorithm for Job Shop Scheduling, *The International Journal of Advanced Manufacturing Technology*, **16**, 765-771.
- Potts, C. N. and Whitehead, J. D. (2001), Workload Balancing and Loop Layout in the Design of a Flexible Manufacturing System, *European Journal of Operational Research*, **129**(2), 326-336.
- Ranjbar, M. and Razavi, M. N. (2012), A Hybrid Metaheuristic for Concurrent Layout and Scheduling Problem in a Job Shop Environment, *International Journal of Advanced Manufacturing Technology*, **62**(9-12), 1249-1260.
- Ripon, K. S. N. and Torresen, J. (2014), Integrated Job Shop Scheduling and Layout Planning: A Hybrid Evolutionary Method for Optimizing

Multiple Objectives, *Evolving Systems*, **5**(2), 121-132.

Ripon, K. S. N., Glette, K., Hovin, M., and Torresen, J. (2012), Job Shop Scheduling with Transportation Delays and Layout Planning in Manufacturing Systems: A Multi-Objective Evolutionary Approach, *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 7326 LNAI, 209-219.

Salido, M. A., Escamilla, J., Giret, A., and Barber, F. (2016), A Genetic Algorithm for Energy-Efficiency in Job-Shop Scheduling, *International Journal of Advanced Manufacturing Technology*, **85**(5-8), 1303-1314.

Singh, S. P. and Singh, V. K. (2010), An Improved Heuristic Approach for Multi-Objective Facility Layout Problem, *International Journal of Production Research*, **48**(4), 1171-1194.

Xiao, Y., Seo, Y., and Seo, M. (2013), A Two-Step Heuristic Algorithm for Layout Design of Unequal-Sized Facilities with Input/Output Points, *International Journal of Production Research*, **51**(14), 4200-4222.

Zhang, R. and Wu, C. (2011), A Simulated Annealing Algorithm based on Block Properties for the Job Shop Scheduling Problem with Total Weighted Tardiness Objective, *Computers and Operations Research*, **38**(5), 854-867.

Zheng, Y., Xiao, Y., and Seo, Y. (2014), A Tabu Search Algorithm for

Simultaneous Machine/AGV Scheduling Problem, *International Journal of Production Research*, **52**(19), 5748-5763.

Zhu, Z. C., Ng, K. M., and Ong, H. L. (2010), A Modified Tabu Search Algorithm for Cost-Based Job Shop Problem, *Journal of the Operational Research Society*, **61**(4), 611-619.

## 저자소개

**김재혁**: 중앙대학교 기계공학과와 전자전기공학부에서 2019년 학사학위를 취득하고 고려대학교 산업경영공학부에서 석사과정에 재학 중이다. 연구분야는 제조시스템 및 휴리스틱 알고리즘 개발이다.

**서윤호**: 고려대학교 산업경영공학부에서 1984년 학사, Pennsylvania State University 산업공학과에서 1989년 석사, 1993년 박사학위를 취득하였다. 2003년부터 고려대학교 산업경영공학부 교수로 재직하고 있다. 연구분야는 제조, 조립 및 물류 시스템 설계 및 시뮬레이션, 최적화, 경로 계획 알고리즘 개발이다.