Journal of Korean Society of Industrial and Systems Engineering
https://doi.org/10.11627/jksie.2024.47.2.116

Vol. 47, No. 2 : 116-123, June 2024
ISSN : 2005-0461 / EISSN : 2287-7975

A Reinforcement Learning Model for Dispatching System through Agent-based Simulation

Minjung Kim · Moonsoo Shin†

Department of Industrial and Management Engineering, Hanbat National University

에이전트 기반 시뮬레이션을 통한 디스패칭 시스템의 강화학습 모델

김민정 · 신문수†

국립한밭대학교 산업경영공학과

In the manufacturing industry, dispatching systems play a crucial role in enhancing production efficiency and optimizing production volume. However, in dynamic production environments, conventional static dispatching methods struggle to adapt to various environmental conditions and constraints, leading to problems such as reduced production volume, delays, and resource wastage. Therefore, there is a need for dynamic dispatching methods that can quickly adapt to changes in the environment. In this study, we aim to develop an agent-based model that considers dynamic situations through interaction between agents. Additionally, we intend to utilize the Q-learning algorithm, which possesses the characteristics of temporal difference (TD) learning, to automatically update and adapt to dynamic situations. This means that Q-learning can effectively consider dynamic environments by sensitively responding to changes in the state space and selecting optimal dispatching rules accordingly. The state space includes information such as inventory and work-in-process levels, order fulfilment status, and machine status, which are used to select the optimal dispatching rules. Furthermore, we aim to minimize total tardiness and the number of setup changes using reinforcement learning. Finally, we will develop a dynamic dispatching system using Q-learning and compare its performance with conventional static dispatching methods.

Keywords : Smart Manufacturing, Rule-based Dispatching, Reinforcement Learning, Q-learning, Agent-based Simulation

1. 서 론

제조업 분야에서 디스패칭 시스템은 생산 공정의 효율성과 생산량 최적화를 위해 필수적인 역할을 한다[12]. 디스패칭 시스템은 작업을 적절한 시점에 적절한 자원에 할당함으로써 전체 생산 공정을 최적화하고, 생산 속도와 품질을 향상시키는 데 기여한다. 흐름생산 공정(flow shop)에서는 일반적으로 하나의 가공품이 확정적인 순서에 따라 여러 공정을 거치기 때문에 투입 단계에서의 디스패칭은 전체 생산 속도와 품질에 직접적인 영향을 미친다. 개별생산 공정(job shop)의 경우에는 각각의 가공품이 다양한 경로를 통해 여러 공정을 거치는 과정에서 작업 순서와 가용 생산자원의 선택 범위가 넓고 복잡하므로, 이를 효율적으로 관리하기 위한 디스패칭 시스템이 더욱 중요하다.

그런데 오늘날의 제조 환경은 매우 동적이며, 예측 불가능한 다양한 변수들을 내포하고 있다[10]. 하지만 기존

Received 31 May 2024; Finally Revised 17 June 2024;
Accepted 18 June 2024
† Corresponding Author : shinms@hanbat.ac.kr

의 확정적 규칙 기반의 정적인 디스패칭 방법은 많은 한계를 보이고 있다[9]. 실제로 기존의 정적인 디스패칭 방법은 예기치 못한 기계 고장, 주문량의 급격한 변화, 긴급 주문 등과 같은 동적인 환경 변화에 적절히 대응하지 못한다. 이에 따라 생산량 감소와 납기 지연, 자원 낭비 등의 문제가 발생하며, 생산 시스템의 유연성이 저하되고, 생산 계획을 최적으로 유지하기 어렵다[6].

일반적으로 생산성 극대화를 위해서는 생산 품목의 변경을 최소화하는 것이 중요하다[3]. 동시에 고객의 요구를 만족시키기 위해서는 납기 준수 또한 매우 중요하다. 그러나 이러한 두 요소는 상충 관계에 있다. 생산 품목의 변경을 최소화하는 과정에서 개별 생산 품목에 대한 납기를 적절히 준수하지 못하는 상황이 빈발할 수 있다. 또한 개별 품목에 대한 납기를 준수하기 위해서는 생산 품목의 잦은 변경을 감수해야 한다. 이러한 상충관계로 인해 두 요소를 동시에 최적화하는 것은 매우 어렵다. 주어진 상황에 따라 보다 중요한 요소를 선택할 수 있는 동적인 접근법이 필요하다.

최근 강화학습 알고리즘은 이러한 문제를 해결할 수 있는 접근 방법으로 주목받고 있다[8]. 강화학습을 통해 에이전트는 실시간으로 환경과 상호작용하며, 최적의 디스패칭 규칙을 학습하고 적용할 수 있다. 이는 기존 방법과 달리 다양한 환경 변화에 적응하고, 실시간으로 최적의 결정을 내릴 수 있도록 도와준다. 강화학습은 기계 학습의 한 분야로, 에이전트가 환경과 상호작용하면서 최적의 행동을 학습하는 과정이다. 이는 복잡한 의사결정 문제를 해결하는 데 널리 사용되고 있다. 특히 불확실성이 높은 환경에서 효과적으로 작동하며, 지속적인 학습과 적응을 통해 성능을 향상시킬 수 있다.

본 연구의 궁극적인 목표는 동적으로 변화하는 제조 환경에 유연하게 대응할 수 있는 지능적인 디스패칭 시스템을 개발하는 것이다. 이를 위해 기존의 정적인 규칙 기반의 디스패칭 방식이 가지는 한계를 극복하고, 실시간으로 최적의 결정을 내릴 수 있는 강화학습 모델을 제시하고자 한다. 특히 생산 품목의 변경을 위한 셋업 횟수와 총 납기 지연(total tardiness)의 최소화를 동시에 지향한다.

논문의 이후 구성은 다음과 같다. 제2장에서는 강화학습 기반의 디스패칭 관련 연구 현황과 이론을 간략히 소개한다. 제3장에서는 본 연구에서 다루는 문제를 정의하고, 이를 해결하기 위한 MDP (Markov decision process) 모델을 제시한다. 제4장에서는 에이전트 기반 디스패칭 시스템에 대해 간략히 소개하고, 제5장에서는 이를 구현하는 시뮬레이션 테스트베드와 이에 기반한 실험 결과를 제시한다. 마지막으로 제6장에서는 결론 및 추후 연구 방향에 대해 기술한다.

2. 관련 연구

2.1 강화학습 기반 디스패칭

디스패칭 시스템의 성능 향상을 위해 강화학습 알고리즘을 적용하는 연구가 활발하게 이루어지는 추세이다. Zhang et al.[14]은 flow shop 문제에 대해 Q-learning 알고리즘을 사용하여 최적의 디스패칭 규칙을 선택함으로써 총 완료 시간 (makespan)을 최소화하였다. 이 연구는 총 완료 시간 최소화를 단일 목적으로 두고 있다.

Yoo et al.[13]은 병렬 기계에 작업을 할당하기 위해 심층 Q-learning 알고리즘을 사용하여 기계를 선택하며, 총 완료 시간과 셋업 비용, 설비 유휴시간의 3가지 측면에서 최적화를 추구하였다. 이 연구는 복잡한 병렬 기계 환경에서 다양한 작업 조건을 고려하여 효율적인 자원 배분과 작업 스케줄링을 수행하였다. 특히, 간트리스 게임 환경에서 알고리즘의 실효성을 입증하였다.

Cho et al.[1]은 flow shop 문제에 대해 DDQN (double deep Q-network) 알고리즘을 사용하여 투입할 제품을 결정하는 스케줄링 알고리즘을 개발하였으며, 총 리드타임 최소화라는 목적함수를 고려하였다. 이 연구는 DDQN을 통해 다양한 제품의 리드타임을 최소화하는 데 중점을 두고 있다.

본 연구에서는 이러한 선행 연구를 참고하여 제조 공정에서 제품의 투입 순서를 결정하는 디스패칭 문제에 강화학습 알고리즘을 적용한다. 특히 생산 품목의 변경 횟수와 총 작업 지연 최소화를 목적으로 학습을 진행하고자 한다. 대부분의 선행 연구에서는 두 가지 이상의 목적을 동시에 고려할 때 각각의 목적에 대한 보상 값들의 가중합으로 총 보상 값을 산출한다. 이러한 접근 방식은 가중치를 통해 각 보상의 중요도를 반영하지만, 보상의 상대적 크기와 중요도를 정확하게 반영하기 어렵다는 한계가 존재한다[2].

본 연구에서는 기존 선행 연구들과 달리 보상 설정 방식에 차별성을 둔다. 에이전트 기반 시뮬레이션을 통해 동적으로 변하는 상황에 대응하여 각각의 목적에 대응하는 보상 값을 구하고, 그 크기를 비교하여 상대적으로 더 큰 보상을 부여하는 액션을 선택한다. 이를 통해 다목적 최적화 문제에서 보상의 상대적 크기와 중요도를 더 정확하게 반영하고자 한다. 이는 선행 연구들과의 주요 차별점이다.

2.2 강화학습

강화학습은 행동의 주체인 에이전트가 시행착오를 통해 행동에 따른 보상을 최대화하는 방향으로 학습하는 알고리즘이다. 제조 공정에서의 디스패칭 문제에 강화학

습 알고리즘을 적용하기 위해서는 이 문제를 MDP로 정의해야 한다. MDP는 상태, 행동, 보상의 세 가지 요소로 에이전트와 환경의 상호작용으로 정의한다. 강화학습은 이러한 상호작용을 통해 정책(policy)을 개선하여 보상을 최대화하는 방향으로 학습된다. 에이전트는 상태 및 행동에 대한 가치를 평가하고, 이를 기반으로 다음 행동을 선택한다. 강화학습 알고리즘은 이러한 가치를 추정하고 업데이트하는 과정을 수행하며 최적의 행동을 학습한다.

2.3 Q-learning 알고리즘

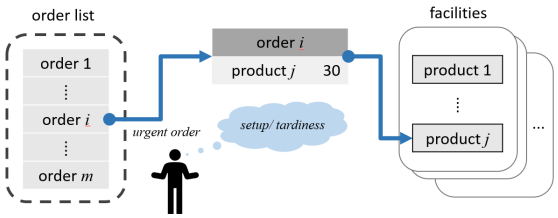
Q-learning은 강화학습의 한 종류로, 에이전트가 환경과 상호작용하면서 얻은 경험을 토대로 상태와 행동 간 최적 가치 함수를 학습하는 알고리즘이다[5]. 이는 주어진 상태에서 가능한 모든 행동에 대한 Q-value를 추정하고, 이를 통해 최적의 행동을 선택한다. 이때, Q-value는 현재 상태와 특정 행동을 선택했을 때 기대되는 미래 보상을 의미한다.

Q-learning은 주어진 상태-행동 쌍에 대한 Q-value를 업데이트하는 과정에서 벨만 최적 방정식을 사용한다. 기본적으로 벨만 방정식을 통해 현재 시점에서의 Q-value와 미래 시점에서의 Q-value 사이의 관계를 나타낼 수 있다. 이를 기반으로 주어진 상태-행동 쌍에 대해 미래에 얻을 것으로 기대되는 보상의 총합을 Q-value로 정의하고, Q-value가 가장 큰 행동을 최적의 행동으로 판단한다. 또한 현재의 상태-행동 쌍에서의 Q-value와 이어지는 다음 상태에서의 최대 Q-value의 차이를 기반으로 현재의 Q-value를 점진적으로 업데이트한다. 이러한 업데이트는 에이전트가 환경과 상호작용하며 새로운 정보를 얻을 때마다 수행된다. 즉, Q-learning은 TD(temporal difference) 학습의 특성이 있어 환경이 변할 때마다 자동으로 업데이트된다. 이는 실시간으로 변화하는 제조 환경에서 유용하게 적용될 수 있는 특징 중 하나이다. 정리하면, Q-learning은 디스패칭 시스템이 환경의 변화에 빠르게 적응하고 최적의 행동을 학습할 수 있도록 지원한다.

3. 강화학습 모델

3.1 문제 정의

본 연구는 제조시스템에서의 전형적인 디스패칭 문제를 다룬다. 특히 주어진 생산 주문 목록에서 가장 긴급한 주문을 선정하고, 선정된 주문이 요구하는 제품을 작업장에 투입하는 상황을 가정한다. 이때 주문 목록의 크기와 생산 가능 제품의 종류는 각각 m과 n이며, 각 주문은 서로 다른 납기와 요구사항을 갖는다(<Figure 1>).



<Figure 1> Dispatching Problem

제조 공정의 효율성을 저해하는 주요 요인 중 하나는 작업물 변경으로 인한 작업 준비 비용이다. 이는 특정 제품을 생산한 후 다른 제품을 생산하기 위해 설비를 변경하고 조정해야 하므로, 작업 시간과 생산 비용이 증가하는 문제가 발생한다. 즉, 생산 장비를 효율적으로 활용하기 위해서는 생산 품목의 변경을 위한 셋업 횟수를 최소화해야 한다. 그러나 셋업 횟수를 줄이기 위해 품목 변경을 최소화할 경우 작업 지연이 유발될 수 있다. 작업 지연은 고객 만족도 및 신뢰도 하락과 지연된 주문 처리에 의한 각종 추가 비용(예, 초과 근무 수당, 긴급 배송 등)을 발생시킨다. 본 논문은 이와 같은 트레이드 오프 관계에 있는 두 가지 요구사항을 동시에 만족할 방법을 모색하기 위해 강화학습 기법을 활용한다.

3.2 Markov Decision Process

강화학습 모델은 기본적으로 MDP를 기반으로 한다. 특히 디스패칭 문제를 다루기 위해서는 MDP의 기본 요소인 상태와 행동, 보상을 디스패칭 문제에 적합한 형태로 설계해야 한다.

3.2.1 상태(state)

본 연구는 셋업 횟수와 작업 지연을 동시에 최소화하는 것을 목적으로 최적의 행동을 탐색하고 학습하는 모델을 제한다. 위의 두 가지 목적을 달성하기 위한 에이전트의 행동 결정의 기준이 되는 상태 정의는 총 세 가지 변수를 활용한다.

첫 번째 상태 변수(f_1)는 직전에 투입한 제품의 종류에 관한 정보로 식 (1)과 같이 제품 종류를 나타낸다. 이는 에이전트가 다음에 어떤 제품을 투입할지 결정하는 데 도움을 준다. 예를 들어, 동일 제품을 연속해 투입함으로써 셋업 발생을 최소화하는 방향으로 유도하는 변수이다.

$$f_1 = 1, 2, 3, \cdots, n \tag{1}$$

n: 가용 제품 종류의 수

두 번째 상태 변수(f_2)는 직전에 투입한 제품의 투입 잔량 유무에 관한 정보로, 식 (2)와 같이 정의한다. 첫 번째 상태 변수(f_1)가 직전에 투입한 제품 정보를 제공하나, 해당 제품이 더 이상 남아있지 않을 때 어쩔 수 없이 다른 제품을 투입해야 한다. 이로 인해 학습이 제대로 되지 않는 것을 방지하고자, 이 변수는 이를 고려한다.

$$f_2 = \begin{cases} 1 & \text{if } rQty(type_{t-1}) > 0 \\ 0 & otherwise \end{cases} \quad (2)$$

$type_{t-1}$: 직전(즉, $t-1$시점)에 투입한 제품
$rQty(type_{t-1})$: 제품 $type_{t-1}$에 대한 투입 잔량

세 번째 상태 변수(f_3)는 현재 공정 상황의 긴급 여부를 나타내는 정보로 식 (3)과 같이 나타낸다. 모든 주문별 여유 시간를 구한 후, 그 중 최솟값이 특정 임계값(본 연구에서는 5,000초를 적용) 이하인 경우 납기 지연 가능성이 높은 것으로 판단하고, 긴급한 상황으로 간주한다.

$$f_3 = \begin{cases} 1 & \text{if } \min_j((d_j - now) - p_j) > 5000 \\ 0 & otherwise \end{cases} \quad (3)$$

d_j: 주문 j의 납기 시간
p_j : 주문 j의 남은 처리 시간
now: 현재 시각

3.2.2 행동(action)

에이전트의 행동은 동적으로 변화하는 제조 환경 속에서 주어진 상태에 대해 가장 적합한 우선순위 규칙을 선정하는 것으로 정의할 수 있다. 본 연구에서 대상으로 삼고 있는 기본적인 우선순위 규칙은 1) FIFO(first in first out), 2) SPT(shortest processing time), 3) LPT(longest processing time), 4) SDT(smallest due time), 5) SA(setup aware), 6) CR(critical ratio) 등이다.

FIFO는 가장 기본적인 우선순위 규칙으로, 작업이 도착한 순서대로 처리하는 방식이다. 즉, 가장 먼저 도착한 작업을 가장 먼저 처리한다. SPT는 공정 처리 시간이 가장 짧은 작업을 가장 먼저 처리하는 방식이다. 작업의 처리 시간을 고려하여 공정 처리 시간이 짧은 작업부터 먼저 처리한다. LPT는 공정 처리 시간이 가장 긴 작업을 먼저 처리하는 방식으로, 작업 시간이 긴 작업부터 처리한다. SDT는 납기 일자가 가장 빠른 작업을 가장 먼저 처리하는 방식이다. 작업의 납기를 고려하여 납기 일자가 빠른 순서대로 처리한다. SA는 셋업 발생을 최소화하기 위한 방식으로 셋업이 필요하지 않은 주문을 우선 선정한다. 따라서 직전에 투입한 제품과 동일한 제품을 우선 처리하며, 만약 동일한 제품이 여러 개 있거나 없으면 FIFO를 기준으로 처리한다. CR은 주문별 긴급한 정도를 식 (4)와 같이 잔여 처리시간 대비 여유시간으로 계산하여 처리하는 방식이다. 이때 CR_j는 주문 j에 대한 긴급률 값을 의미한다. 즉, 잔여 처리시간 대비 여유시간의 비율로 계산하여 긴급도가 가장 높은 주문을 가장 먼저 처리한다.

$$j^* = \arg\max_j(CR_j) = (d_j - now)/p_j \quad (4)$$

3.2.3 보상(reward)

보상은 학습의 성능을 좌우하는 핵심 요소로서 선택된 행동의 결과로 초래되어 산정되는 수치화된 값이다[12]. 본 연구에서는 셋업 횟수 및 작업 지연의 최소화라는 목적을 함께 반영하기 위해 다음 두 가지 관점에서 각각의 보상 값을 정의한다.

셋업 횟수 최소화 관점의 보상 r_1은 식 (5)와 같다. 선택된 액션, 즉 선택된 우선순위 규칙에 의해 선정된 주문에 해당하는 제품이 작업장에 투입될 경우 셋업이 발생하면 -1의 보상을 부여하고, 그렇지 않으면 0의 보상을 준다. 작업 지연 최소화 관점에서의 보상 r_2는 식 (6)과 같다. 납기 지연이 발생하면 -1의 보상을 부여하고, 그렇지 않으면 0의 보상을 주도록 설정한다.

이와 같은 보상 체계를 통해 디스패칭 에이전트는 보상을 극대화하기 위한 행동을 선택하는 방법을 학습한다. 각 관점에서 보상의 크기를 계산한 후, 두 보상 중 더 큰 값을 가지는 쪽의 행동을 취하도록 한다(식 (7)). $r_1(i)$와 $r_2(i)$는 각각 우선순위 규칙 i를 적용했을 때 기대되는 r_1과 r_2를 의미하며, i^*는 최적의 우선순위 규칙을 의미한다. 이러한 보상 구조는 에이전트가 특정 목적에만 집중하지 않고, 하나 이상 복수의 관점에서 보상 체계를 종합적으로 고려하여 균형 잡힌 결정을 내릴 수 있도록 지원한다.

$$r_1 = \begin{cases} -1 \text{ if } type_{t-1} \neq type_t \\ 0 \; otherwise \end{cases} \quad (5)$$

$$r_2 = \begin{cases} -1 \text{ if } d_j < c_j \; (c_j : \text{주문 } j\text{의 완료시간}) \\ 0 \; otherwise \end{cases} \quad (6)$$

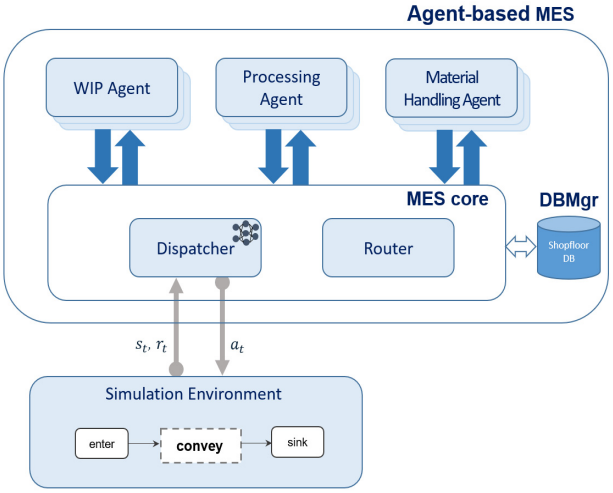
$$i^* = \arg\max_i(\max(r_1(i), r_2(i))) \quad (7)$$

4. 에이전트 기반 디스패칭 시스템

본 연구에서는 각 작업이 투입되기 직전 시점을 의사

결정 시점으로 설정하여, 에이전트가 그 시점의 환경 상태를 제공받아 행동을 결정하도록 한다. 또한, 에이전트 기반 시뮬레이션 모델을 활용하여 학습 환경을 구현함으로써 상태변환 확률을 따로 설정하지 않고 시뮬레이션을 통해 다음 상태와 보상을 도출하고자 한다.

제조 공정에서의 디스패칭 문제에 대한 학습 프레임워크는 <Figure 2>와 같이 구성된다. 에이전트 기반 MES(manufacturing execution system)는 WIP Agent와 Processing Agent, Material Handling Agent 등의 다양한 에이전트들 간의 협업을 통해 작동된다. 이런 협업을 통해 각 에이전트 간 자율적인 상호작용이 이루어지며, 동적인 상황에 대응할 수 있다.



<Figure 2> Agent-based Manufacturing Execution Framework

에이전트 기반 MES는 에이전트들의 상태 정보를 관리하며, 이 중 WIP Agent는 투입할 제품을 담당한다. Dispatcher는 최적의 작업을 할당하고 관리하는 에이전트 모듈로, MES로부터 주문 및 상태 정보를 제공받아 디스패칭 규칙을 선택하고 작업을 진행한다. 에이전트 간 상호작용을 통해 동적으로 변하는 상황에 적응하고 최적의 결정을 내릴 수 있다.

에이전트 기반 시뮬레이션은 다음과 같은 방식으로 작동된다. 먼저, MES로부터 실시간으로 현재 상태 정보를 수집한다. 그런 다음, 에이전트는 수집된 상태 정보를 바탕으로 최적의 디스패칭 규칙을 선택한다. 선택된 행동에 따라 시뮬레이션을 통해 다음 상태와 보상을 구한다. 마지막으로, 계산된 보상을 통해 강화학습 알고리즘을 업데이트하여 에이전트의 성능을 향상시킨다. 이를 통해 에이전트 기반 디스패칭 시스템은 동적인 제조 환경에서 효율적으로 작동하며, 생산성을 극대화하고 납기 준수율을 향상시킬 수 있다.

5. 시뮬레이션 실험

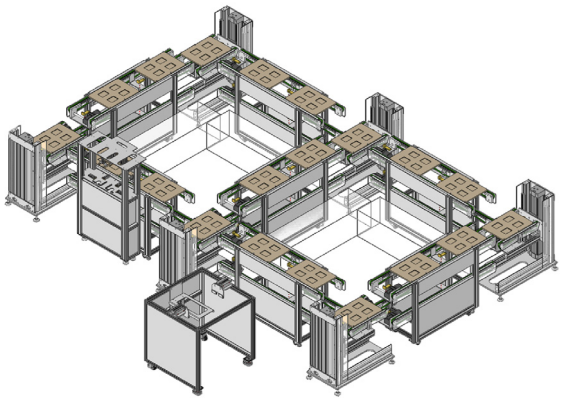
5.1 시뮬레이션 테스트베드

본 논문은 제안된 강화학습 모델을 실증하기 위한 목적으로 시뮬레이션 테스트베드를 개발하여 제시한다. 개발된 테스트베드는 Dispatcher가 선택한 액션(즉, 디스패칭 규칙)에 따라 선정된 제품을 작업 현장에 투입하고, 이로 인해 귀결되는 상태전이와 보상 값을 보고받기 위한 작업 현장의 디지털 트윈으로서의 기능을 수행한다.

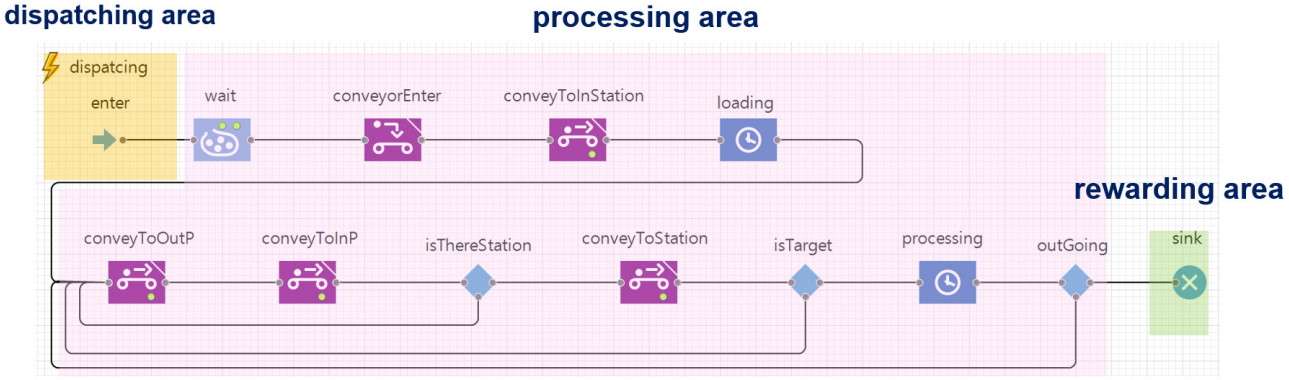
시뮬레이션 테스트베드 개발은 에이전트 기반 시뮬레이션 툴인 AnyLogic[11]을 기반으로 하며, 국립한밭대학교가 구축하여 운영 중인 스마트팩토리 테스트베드 시스템[4]을 대상으로 수행되었다. 대상 시스템은 <Figure 3>에 나타난 바와 같이 입고 및 출고 공정과 가공 공정을 수행하는 총 7개의 스테이션으로 구성된다. 또한 비동기식 컨베이어 시스템을 통해 자유경로 시스템을 지원한다.

<Figure 4>는 대상 시스템의 작동 모형을 AnyLogic으로 모델링한 흐름 구조를 보이고 있으며, 전반적인 흐름 구조는 다음 세 가지 영역으로 구분된다.

- 작업 할당(diapatching): 선정된 디스패칭 규칙에 따라 선택된 제품을 투입하는 단계. 이전 제품과 현재 투입된 제품을 비교하여 셋업 발생 여부를 확인
- 가공 처리(processing): 개별 WIP의 공정 진행을 묘사하는 단계. 각 WIP이 생산 라인 내에서 이동하고 작업이 진행되는 과정을 묘사
- 보상 처리(rewarding): 개별 WIP의 작업 완료를 처리하는 단계. 작업이 완료된 제품의 납기 시간과 작업이 완료된 시간을 비교하여 납기 만족 여부를 확인하고, 보상 값을 업데이트



<Figure 3> Hanbat Smart Factory



<Figure 4> AnyLogic Process Model

5.2 시뮬레이션 시나리오

본 연구는 3종의 제품에 대해 50건의 주문에 해당하는 총 500개의 제품의 작업 순서를 결정하는 문제로 시뮬레이션 시나리오를 구성한다. 각 제품은 동일한 작업 순서를 따르지만, 공정별 소요 시간은 제품 종류에 따라 다르며, 주문 시점과 납기 시점, 주문 수량은 모두 다르다. 주요 가정은 다음과 같다. 첫째, 항상 최적의 설비를 사용하여 불량품이 발생하지 않는다. 둘째, 모든 작업물의 설비 간 이동은 최단 경로를 따른다. 마지막으로 강화학습 초기에는 무작위로 디스패칭 규칙을 선택한다.

본 연구에서 적용한 Q-learning 알고리즘의 하이퍼 파라미터는 <Table 1>과 같다. 강화학습은 총 100회의 에피소드를 통해 진행한다. 이때 입실론(epsilon) 값은 탐색하지 않았던 해 공간을 탐험하기 위해 무작위로 행동을 선택하는 비율을 의미하며, 학습이 진행됨에 따라 점차 감소한다. 할인율(discount ratio)은 미래의 보상 값에 부여하는 값으로 현재의 보상 값에 대한 상대적 가중치를 의미한다. 학습률(learning rate)은 Q-value를 업데이트하는 과정에 새로운 정보가 반영되는 비율을 의미한다.

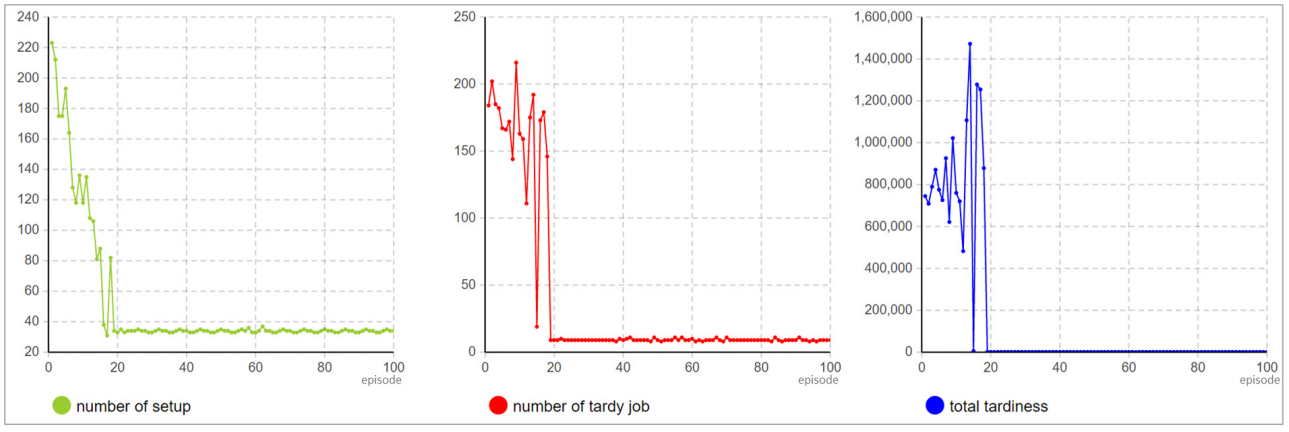
<Table 1> Hyper-parameters

hyper-parameters	value
number of episode	100
initial value of epsilon	0.9
discount ratio	1.0
learning rate	0.1

5.3 실험 결과

실험 결과는 <Figure 5>와 <Table 2>에 나타낸 바와 같다. <Figure 5>는 강화학습 에피소드가 진행됨에 따라 주요 성능 지표(셋업 횟수, 지연 작업의 개수, 총 지연시간)가 개선되는 과정을 보이고 있다. 특히 셋업 횟수와 지연 작업의 개수 및 총 지연시간 모두가 최소화된 수준에서 수렴함을 확인할 수 있다.

<Table 2>는 기존의 정적인 디스패칭 규칙 기반의 실



<Figure 5> Measurements of Performances(i.e. number of setups and tardy jobs, total tardiness)

<Table 2> Simulation Result

dispatching rule	FIFO	SPT	LPT	SDT	SA	CR	random selection (average)	proposed model
number of setups	35	22	21	44	3	33	204	34
number of tardy jobs	198	181	201	9	176	95	184	9
total tardiness (sec)	1,937,000	1,735,000	1,438,000	1,018	1,632,000	552,080	1,028,523	1,110

험 결과와 비교하여 보이고 있다. 셋업 횟수 관점에서는 SA 규칙이 가장 좋은 결과(3회)를 보이고 있다. 반면, 본 연구에서 제안된 강화학습 모델의 경우에는 34회의 셋업이 발생하였음을 확인할 수 있다. 하지만, 납기 지연 관점의 성능지표를 살펴보면 제안된 모델이 최선의 결과를 보인 SDT 규칙과 거의 유사한 수준의 결과를 보이고 있다. SDT 규칙과 SA 규칙이 각각 셋업 횟수 관점과 납기 지연 관점에서는 좋지 않은 성과를 보이고 있음을 고려할 때 제안된 강화학습 모델에 의한 결과가 가장 우수한 것으로 판단할 수 있다.

제시된 강화학습 모델이 우수한 성능지표를 보이는 이유는 에피소드 초반과 후반 각각에서 의사결정 패턴이 동적으로 변화하기 때문으로 파악된다. 에피소드 초반에는 모든 주문 건이 상대적으로 여유가 있으므로 납기 지연보다는 셋업을 최소화하는 방향의 의사결정(즉, SA 규칙을 선택)하고, 후반으로 갈수록 납기 지연을 최소화하기 위한 의사결정(즉, SDT 규칙을 선택)을 수행하기 때문이다.

6. 결 론

본 연구에서는 강화학습 기반 디스패칭 모델을 개발하고, 기존의 정적인 규칙 기반의 디스패칭 시스템과 성능을 비교하여 검증하였다. 이를 통해 효과적인 작업 투입 순서를 결정함으로써 제조 공정의 생산성을 개선할 수 있다. 이때 생산성 개선은 생산량 최대화와 고객 만족 최대화라는 두 가지 측면에서 고려되며, 이는 각각 셋업 횟수 최소화와 작업 지연 최소화의 형태로 정의된다. 또한 본 연구에서는 상호 상충관계에 있는 위의 두 측면을 동시에 고려하여 최적의 정책을 학습하기 위한 MDP를 정의하였다. 특히 Q-learning 알고리즘을 기반으로 강화학습 모델을 개발하였으며, AnyLogic 기반의 시뮬레이션 테스트베드를 구축하여 상태 변이와 보상치 산정에 연계하였다. 개발된 모델은 기존 우선순위 규칙과의 비교를 통해 성능을 평가하였다. 결과적으로 셋업 횟수와 지연작업의 개수 및 총 작업 지연시간의 세 가지 평가 지표를 종합적으로 고려하였을 때, 정적인 우선순위 규칙보다 제안된 디스패칭 모델이 더 효과적임을 확인하였다.

본 연구에서는 최소화된 상태공간과 보상구조를 기반으로 동적인 디스패칭 규칙의 선정 메커니즘을 제안하고 있다. 이로 인해 셋업 횟수와 지연 작업의 개수, 그리고 총 작업 지연시간을 개별적으로 해석할 수 밖에 없는 한계가 있다. 추후에는 보다 면밀하고 다양한 상태공간 및 보상구조를 적용함으로써 일반화된 환경에서의 디스패칭 문제에 적용할 수 있도록 개선할 필요가 있다. 특히 셋업 횟수 최소화와 작업 지연 최소화를 동시에 지향할 수 있는 정교하고 복합적인 보상구조의 정의가 필요하다. 그리고 본 연구에서는 실험 환경의 단순화를 위해 확률적 변이에 의한 불확실성 요소를 최대한 배제하였으며, 작업물의 이송 또한 최단 거리 기준의 확정적 경로를 적용하고 있다. 추후에는 확률적 변동성을 고려한 연구가 필요하다. 또한 디스패칭 알고리즘의 학습 속도의 향상을 위해 하이퍼 파라미터 최적화에 대한 추가 연구가 필요하다. 마지막으로 보다 다양한 시나리오에서의 검증과 개선을 위한 추후 연구가 필요하다.

Acknowledgement

This research was supported by the research fund of Hanbat National University in 2023.

References

[1] Cho, Y.I., Nam, S.H., and Woo, J.H., Development of the Reinforcement Learning-based Adaptive Scheduling Algorithm for Panel Block Shop, *Korean Journal of Computational Design and Engineering*, 2021, Vol. 26, No. 2, pp. 81-92.

[2] Cho, Y.I., Oh, S.H., Kwak, D.H., Choi, J.H., and Woo, J.H., Development of Quay Scheduling Algorithm Based on Reinforcement Learning, *Korean Journal of Computational Design and Engineering*, 2022, Vol. 27, No. 2, pp. 98-114.

[3] Choi, H.W., Byeon, H.J., Yoon, S.H., Kim, B.S., and Hong, S.D., Analysis of Workforce Scheduling Using

Adjusted Man-machine Chart and Simulation, *Journal of Korean Society of Industrial and Systems Engineering*, 2024, Vol. 47, No. 1, pp. 20-27.

[4] Hanbat National University, https://www.youtube.com/watch?v=U-Zsqq0I2X8&t=400s.

[5] Kang, B.W., Kang, B.M., and Hong, S.D., A Dynamic OHT Routing Algorithm in Automated Material Handling Systems, *Journal of Korean Society of Industrial and Systems Engineering*, 2022, Vol. 45, No. 3, pp. 40-48.

[6] Kim, S.J., Yoo, W.S., and Kim, G.H., Real-time scheduling using CNN on a parallel machine with setup cost, *Proceedings of the 2018 Korean Institute of Industrial Engineers Fall Conference*, 2018, Seoul, pp. 1385-1401.

[7] Kim, H.H., Kim, J.H., Kong, J.H., and Kyung, J.H., Reinforcement Learning-based Dynamic Weapon Assignment to Multi-Caliber Long-Range Artillery Attacks, *Journal of Korean Society of Industrial and Systems Engineering*, 2022, Vol. 45, No. 4, pp. 42-52.

[8] Levine, S., Reinforcement Learning and Control as Probabilistic Inference: Tutorial and Review, 2018, arXiv Preprint:1805.00909.

[9] Nam, S.H., Cho, Y.I., and Woo, J.H., Reinforcement Learning for Minimizing Tardiness and Set-Up Change in Parallel Machine Scheduling Problems for Profile Shops in Shipyard, *Journal of the Society of Naval Architects of Korea*, 2023, Vol. 60, No. 3, pp.202-211.

[10] Priore, P., Gómez, A., Pino, R., and Rosillo, R., Dynamic scheduling of manufacturing systems using machine learning: An updated review, Artificial Intelligence for Engineering Design, *Analysis and Manufacturing*, 2014, Vol. 28, No. 1, pp. 83-97.

[11] The AnyLogic Company, https://www.anylogic.com.

[12] Waschneck, B., Reichstaller, A., Belzner, L., Altenmüller, T., Bauernhansl, T., Knapp, A., and Kyek, A., Deep reinforcement learning for semiconductor production scheduling, *Proceedings of the 29th Annual SEMI Advanced Semiconductor Manufacturing Conference (ASMC)*, Saratoga Springs, NY, USA, 2018, pp. 301-306.

[13] Yoo, W.S., Kim, S.J., and Kim, G.H., Real-Time Scheduling Scheme based on Reinforcement Learning Considering Minimizing Setup Cost, *The Journal of Society for e-Business Studies*, 2020, Vol. 25, No. 2, pp.15-27.

[14] Zhang, Z., Wang, W., Zhong, S., and Kaishun, H.U., Flow Shop Scheduling with Reinforcement Learning, *Asia-Pacific Journal of Operational Research*, 2013.

ORCID

Minjung Kim | http://orcid.org/0009-0001-8885-8460

Moonsoo Shin | http://orcid.org/0000-0001-6318-9662