소포물 분류 시스템의 다중 에이전트 강화 학습 기반 행동 제어

최호빈*, 김주봉*, 황규영*, 한연희*†
*한국기술교육대학교 컴퓨터공학과, 미래융합공학전공
{chb3350, rlawnqhd, to6289, yhhan}@koreatech.ac.kr

Multi-Agent Reinforcement Learning-based Behavior Control of Parcel Sortation System

Ho-Bin Choi*, Ju-Bong Kim*, Gyu-Young Hwang*, Youn-Hee Han*†
*Future Convergence Engineering Major,
Dept. of Computer Science Engineering, KoreaTech University

요 약

인공지능은 스스로 학습하며 기존 통계 분석보다 탁월한 분석 역량을 지니고 있어 스마트팩토리 혁신에 새로운 전기를 마련할 것으로 기대된다. 이를 증명하듯 스마트팩토리의 주요 분야인 공정 간 연계 제어, 전문가 공정 제어, 로봇 자동화 등에서 활발한 연구가 이어지고 있다. 본 논문에서는 소포물 분류 시스템에 전통적인 룰 기반의 제어 방식 대신 다중 에이전트 강화 학습 제어 방식을 설계 및 적용하여 효과적인 행동 제어가 가능함을 입증한다.

1. 서론

내부 물류(Intralogistics) 산업은 공장의 운영 자동화와 전체 프로세스 절차의 디지털화로 인해 가장 빠르게 성장하는 산업 중 하나이다. 물류(Logistics)가 자원이 필요한 장소에 있는지 확인하기 위해 계획하고 구성하는 프로세스라면, 내부 물류는 창고와 같은 내부 장소에서 생산 및 유통 프로세스를 관리하고 최적화하는 것이다 [1]. 또, 물류는 한 지점에서 다른 지점으로 제품을 이동하는 방법인 반면, 내부 물류는 비슷한 개념이지만 창고나 유통 센터 내에서 공급, 생산 및 유통의 모든 물리적 물류 흐름을 최적화하는 것이다 [2].

스마트 제조 현장에 설치된 각종 센서 또는 사물인터넷(IoT; Internet of Things)으로부터 생성되는 대용량의 데이터를 기반으로 제조공정에서 필요로 하는 최적 제어 운용 기술은 매우 중요한 이슈이다. 강화 학습(Reinforcement Learning) 에이전트는 제어하려는 환경(Environment) 안에서 현재의 상태(State)를 인식하여 행동(Action)을 선택하고, 환경은 에이전트가 취한 행동에 대한 보상(Reward)을 반환한다. 에이전트는 이 과정을 반복하여 보상을 최대화하는 제어 정책을 학습한다. 이러한 최적 제어에 특화된 강화 학습을 통하여 내부 물류를 제어한다면, 높은 효율성과 그에 따른 공정비용 절감을 불러올 것으로 예측된다.

본 논문에서는 소포물 분류 시스템에 전통적인 룰 기반의 제어 방식 대신 다중 에이전트 강화 학습 제어 방식을 설계 및 적용하여 효과적인 행동 제어가 가능함을 입증한다. 실험은 다중 입출력 레이아웃을 구성하여 진행한다.

2. 강화 학습

2.1 환경 및 시나리오

그림 1은 본 연구의 실험에 사용된 레이아웃이다. 각 셀은 모듈이 놓일 수 있으며 셀의 색을 통해 어떤 모듈인지 구분할 수 있다. 먼저, 회색 셀은 모듈이 존재하지 않는다. 녹색 셀은 Emitter로서 분류될 parcel이 들어오며, parcel은 1, 2, 3의 타입 중 무작위로 하나의 타입을 갖는다. 여기서 Emitter로 들어오는 parcel의 수는 무한하다고 가정하였다. 주황색 셀은 Sorter로서 parcel이 목적지를 향해 분류될 수 있도록 한다. 적색 셀은 Remover로서 parcel의 목적지가 되며 도착해야 할 parcel의 타입은 그림 1과 같이 한 가지로 정해져 있다.

† 이 논문은 정부(교육부)의 재원으로 한국연구재단의 지원을 받아 수행된 기초연구사업임 (No. 2018R1A6A1A03025526 및 No. NRF-2020R1I1A3065610).

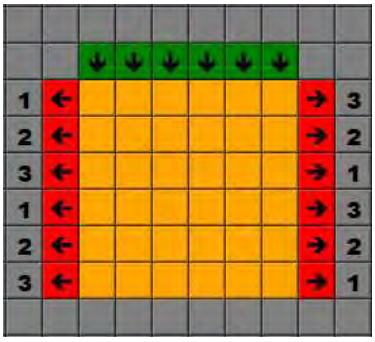

그림 1. 실험에 사용된 레이아웃

2.2 학습 목표 및 에이전트 설계

본 연구의 학습 목표는 주어진 레이아웃에서 분류 대기 중인 무한한 parcel들에 대해 시간 대비 분류 수를 가능한 한 높이는 것이다. 학습 목표를 달성하기 위해 다음과 같이 다중 에이전트를 설계하였다.

Emitter와 Sorter 같은 모듈들이 에이전트가 되는 것이 아니라 parcel 각각이 에이전트가 되어 본인의 observation을 보고 행동을 결정한다. 에이전트는 두 종류가 있으며 parcel이 Emitter 위에 있을 때는 Emitting 에이전트가 되며, Sorter 위에 있을 때는 Sorting 에이전트가 된다. 마찬가지로, 모델도 두 종류로 나뉘며 parcel의 타입이 같으면 모델을 공유한다. 따라서, parcel의 타입이 3개이므로 모델은 총 6개가 존재하며 모두 CNN (Convolutional Neural Network) + FC (Fully Connected) Layer의 구조로 이루어져 있다. 알고리즘은 모두 DQN (Deep Q-Network)을 사용하였다 [3].

2.3 State, Action, Reward

1) Emitting 에이전트

각 Emitting 에이전트는 State로 레이아웃 전체의 정보를 받는다. Action은 [‘stop’, ‘emission’] 중에 선택한다. Reward는 전체 Sorter 수 대비 parcel이 올려져 있는 Sorter 수의 비율에 따라 높을수록 -1.0을 받고 낮을수록 +1.0을 받는다.

2) Sorting 에이전트

각 Sorting 에이전트는 State로 자신의 위치를 중심으로 한 5x5 크기의 정보를 받는다. Action은 [‘stop’, ‘up’, ‘right’, ‘down’, ‘left’] 중에 선택한다. Reward는 자신의 타입과 동일한 타입의 Remover로 분류되면 +1.0을 받는다.

3. 실험

전통적인 룰 기반의 제어 방식이 아닌 다중 에이전트 강화 학습 제어 방식으로 효과적인 행동 제어가 가능함을 입증하기 위해 그림 1에 제시된 레이아웃을 대상으로 성능을 평가한다. 평가 지표로는 수식 (1)에 제시된 SPI (Sortation Performance Index)를 사용한다.

$$SPI_t = \frac{E_t + R_t}{N_{Emitters} + N_{Removers}} \quad (1)$$

SPI는 타임 스텝마다 측정된다. 분모에 있는 $N_{Emitters}$와 $N_{Removers}$는 각각 레이아웃에 존재하는 Emitter의 수와 Remover의 수를 나타내며, 본 실험에서는 하나의 레이아웃에 대해서만 성능을 평가하므로 SPI의 분모 값은 변하지 않는다. 분자에 있는 E_t와 R_t는 각각 타임 스텝마다 Emitting 되는 parcel의 수와 Removing 되는 parcel의 수이다. 즉, SPI는 현재 주어진 레이아웃에서 Emitter와 Remover의 수를 고려하여 타임 스텝마다 얼마나 많은 parcel이 Emitting 되거나 Removing 되는지를 나타낸다. SPI는 0 이상 1 이하의 값을 가지며, 0에 가까울수록 Emitting과 Removing이 적게 되었음을 나타내고 1에 가까울수록 많이 되었음을 나타낸다.

그림 2는 그림 1에 제시된 레이아웃에 대해 본 연구에서 설계한 다중 에이전트 강화 학습을 적용해 학습한 SPI 그래프이다. x축은 타임 스텝, y축은 해당 타임 스텝의 SPI를 나타낸다. 학습이 진행될수록 SPI가 점차 증가해 수렴하는 것을 확인할 수 있다.

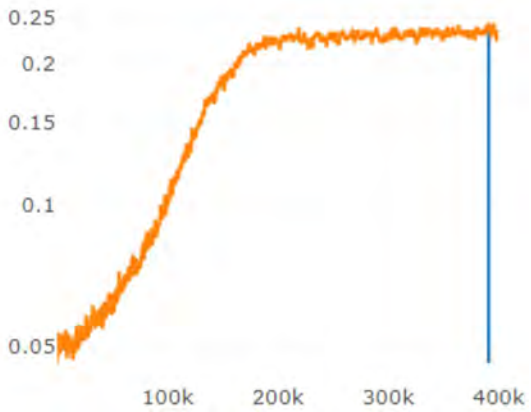

그림 2. Sortation Performance Index

참고문헌

[1] Hafner, N., & Lottersberger, F. "Intralogistics Systems - Optimization of Energy Efficiency." FME transactions 44.3 (2016): 256-266.

[2] Cavalcante, T. R. F., de Bessa, I. V., & Cordeiro, L. C. "Planning and evaluation of uav mission planner for intralogistics problems." 2017 VII Brazilian Symposium on Computing Systems Engineering (SBESC). IEEE, 2017.

[3] Mnih, Volodymyr, et al. "Human-level control through deep reinforcement learning." nature 518.7540 (2015): 529-533.