소포물 분류를 위한 그리드 타입 시스템의 강화 학습 기반 행동 제어

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

Reinforcement learning-based behavior control of a grid-type system for sorting parcels

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

요 약

공정 데이터를 실시간으로 수집할 수 있는 스마트 팩토리의 장점을 활용하여, 일반적인 기계 학습 대신 강화 학습을 사용한다면 미리 요구되는 훈련 데이터 없이 행동 제어를 할 수 있다. 하지만, 현실 세계에서는 물리적 마모, 시간적 문제 등으로 인해 수천만 번 이상의 반복 학습이 불가능하다. 따라서, 본 논문에서는 시뮬레이터를 활용해 스마트 팩토리 분야에서 복잡한 환경 중 하나인 이송 설비에 초점을 둔 그리드 분류 시스템을 개발하고 협력적 다중 에이전트 기반의 강화 학습을 설계하여 효율적인 행동 제어가 가능함을 입증한다.

1. 서론

물류 관리 분야에서, 분류는 제품(상품, 수하물, 우편물 등)을 식별하여 특정 목적지로 전환하는 프로세스이다. 전통적인 분류기는 컨베이어를 기반으로 하여 많은 시스템에 응용되어왔다. 컨베이어 기반의 분류 시스템은 장비의 성능이나 장비끼리의 호환성 등을 중요시하였다면, 최근에는 컨베이어를 활용하여 시스템 전체의 성능을 최적화하는 다양한 분류 시스템이 연구되고 있다. 그러한 연구들은 장비들의 배치 구성이나 작업 처리 알고리즘과 관련되어 있다. 특히, 그리드 구조의 분류 시스템 연구가 활발하며 실제 상업 제품으로 사용되고 있어 실용성이 입증되었다. 그리드 구조의 분류 시스템은 전통적인 분류 시스템에 비해 더 높은 처리량을 보여주며 더 적은 공간으로 시스템을 구성할 수 있다.

본 논문에서는 Real Games사에서 제공하는 3D Simulation Software 중 스마트 팩토리 분야에 해당하는 Factory I/O를 사용하여 스마트 팩토리 분야에서 복잡한 환경 중 하나인 이송 설비에 초점을 둔 그리드 분류 시스템을 개발한다[1]. 개발한 그리드 분류 시스템의 소형 버전에 협력적 다중 에이전트 기반 강화 학습 환경을 설계하고 적용하여 복잡한 규칙 기반의 알고리즘 없이 효율적인 행동 제어가 가능함을 입증한다.

2. 개발한 시스템

그림 1은 본 연구에서 개발한 3×3개의 Chain Transfer가 그리드 구조로 중앙에 구성된 3-Grid Sortation System이다. 본 시스템은 N×N개의 Chain Transfer로 구성되는 N-Grid Sortation System의 간단한 버전이며 쉽게 확장 가능하다. 시스템의 상단과 하단에는 2N 개의 Emitter가 존재하며 좌측과 우측에는 2N 개의 Remover가 존재한다.

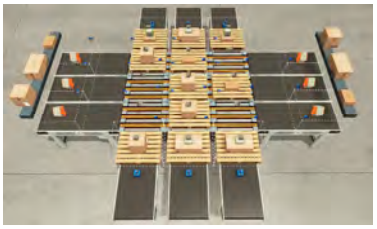

그림 1. 3-Grid Sortation System

3. 강화 학습 설계

그림 2는 본 연구에서 사용한 협력적 다중 에이전트 강화 학습 구성이다. 각 셀의 윗줄은 Factory I/O의 Parts를 나타내며 각 셀의 아랫줄은 강화 학습에서의 역할을 의미한다. D는 분류 목적지를 나타

내며, 각 Chain Transfer는 독립적인 Sorting 에이전트가 제어하고 6개의 Emitter는 하나의 Emitting 에이전트가 제어한다. 모든 에이전트는 서로 다른 CNN을 가지며, Sorting 에이전트들은 DQN 알고리즘을 사용하였고 Emitting 에이전트는 PPO 알고리즘을 사용하였다[2, 3].

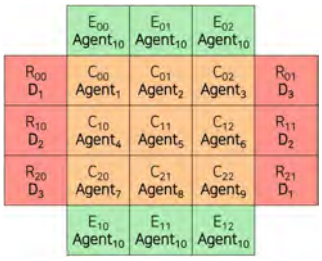

그림 2. RL Configuration

3.1 Environment

본 연구에서 설정한 강화 학습의 에피소드 시나리오는 다수의 Emitter에서 분류 대기 중인 무작위 타입 500개의 상자를 타입에 맞게 올바른 목적지로 신속하게 이동 분류하는 것이다. 분류할 상자의 타입은 Small, Medium, Large로 총 세 가지가 있으며 각각의 목적지는 순서대로 D_1, D_2, D_3가 된다. 최종 학습 목표는 높은 분류 정확도를 유지하며 최대한 빠르게 모든 상자를 분류하는 것이다.

3.2 State, Action, Reward

1) Sorting Agents

state로 전체 상자들의 위치와 자신의 위치를 사용한다. action은 타임 스텝마다 정지할 것인지 상자를 인접한 Chain Transfer 또는 Remover로 보낼 것인지 선택한다. reward는 분류 정확성과 충돌 여부를 고려하여 결정된다.

2) Emitting Agent

state로 전체 상자들의 위치와 Sorting Agent들의 액션 정보를 사용한다. action은 타임 스텝마다 6개의 Emitter 각각에 대해서 정지할 것인지 상자를 인접한 Sorting 에이전트에게 보낼 것인지 선택한다. reward는 emission 양, 분류량 및 충돌 여부를 고려하여 결정된다.

4. 실험

수식 1은 본 연구에서 설계한 강화 학습의 성능 평가를 위한 지표 PI (Performance Index)이며 에피소드마다 측정이 된다.

$$PI = \alpha \frac{R_{right} - R_{wrong}}{N_{destination}} + (1-\alpha) \frac{R_{emission}}{N_{emitter}} \quad (1)$$

PI는 두 항의 합으로 이루어져 있으며 α를 통해 두 항의 가중치를 조절한다. 첫 번째 항에 포함된 R_{right}는 해당 에피소드에서 타임 스텝당 올바르게 분류한 상자의 수이고, R_{wrong}은 해당 에피소드에서 타임 스텝당 올바르지 않게 분류한 상자의 수이다. $N_{destination}$은 분류 목적지의 수로 정규화의 역할을 한다. 따라서, 첫 번째 항은 상자가 올바르게 분류될수록 1에 가깝고 올바르지 않게 분류될수록 -1에 가까운 값이 계산된다. 두 번째 항에 포함된 $R_{emission}$은 해당 에피소드에서 타임 스텝당 6개의 Emitter가 한 Emission 횟수이다. $N_{emitter}$는 Emitter의 수이며 정규화의 역할을 한다. 따라서, 두 번째 항은 Emission 되는 횟수가 많을수록 1에 가깝고 적을수록 0에 가까운 값이 계산된다.

그림 3은 에피소드에 따른 PI 값의 변화 그래프를 나타내며 PI 값이 일정 기간 오르지 않으면 학습을 종료시켰다. 학습은 약 550 에피소드에서 종료되었으며 학습이 진행될수록 PI 값이 증가하는 것을 확인할 수 있다. 본 실험은 3-Grid Sortation System으로 수행한 것이기 때문에 상자들이 이동할 수 있는 버퍼가 매우 부족해 많은 제약사항이 존재한다. 따라서, Grid의 수를 늘릴수록 PI 값이 커질 것으로 사료된다.

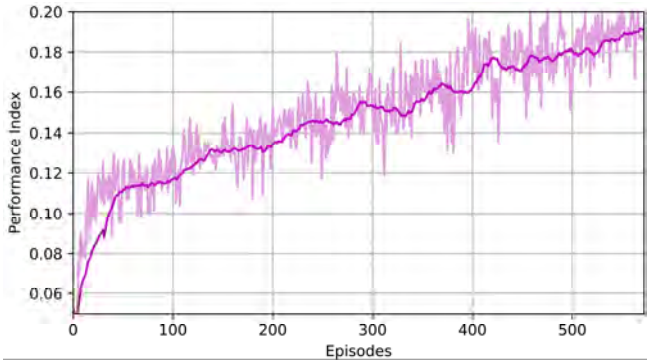

그림 3. Performance Index

ACKNOWLEDGMENT

†: 교신저자 한연희

이 논문은 2018년도 정부(교육부)의 재원으로 한국연구재단의 지원을 받아 수행된 기초연구사업임 (No. 2018R1A6A1A03025526).

참고문헌

[1] https://factoryio.com.

[2] V. Mnih, K. Kavukcuoglu, D. Silver, A. Rusu, J. Veness, M. Bellemare, A. Graves, M. Riedmiller, A. Fidjeland, G. Ostrovski, S. Petersen, C. Beattie, A. Sadik, I. Antonoglou, H. King, D. Kumaran, D. Wierstra, S. Legg, and D. Hassabis, "Human-level control through deep reinforcement learning," Nature, vol. 518, pp. 529-533, Feb. 2015.

[3] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, Oleg Klimov, "Proximal Policy Optimization Algorithms," arXiv:1707.06347, Jul. 2017.