

항만 물류 환경에서 강화학습 기반 컨테이너 다단 적재 모델링 방법

장우석*¹, 이효준*², 서민성*³, 이성진**^{*}, 김동규***^{*}

Reinforcement Learning-based Container Multi-stage Loading Modelling Method in Port Logistics

Woo-Seok Jang*¹, Hyo-June Lee*², Min-Seoung Seo*³, Seong-Jin Lee**^{*}, and Dong-Gyu Kim***^{*}

본 연구는 중소벤처기업부의 재원으로 중소기업기술정보진흥원의 지원을 받아 수행된 연구임
(NTIS고유번호 [1425168887])

요 약

항만 물류 환경에서 한정된 공간에 최대한 많은 컨테이너를 적재하기 위하여 컨테이너 다단 적재를 하고 있다. 다단 적재 상황에서 저단 화물을 취급하려면 상단에 적재된 화물을 다른 장소로 이동시켜야 한다. 이러한 과정을 컨테이너 재취급이라 하며, 재취급 과정에서 비용이 소모되므로 컨테이너 장치장 운영사는 재취급을 최소화하고자 한다. 본 논문에서는 이러한 재취급을 최소화하여 컨테이너를 다단 적재하도록 적재 위치를 추천, 선택하는 강화학습 모델을 제안한다. 제안된 강화학습 모델은 컨테이너의 야드 출고 일자를 고려하여 적재순서를 최적화함으로써 하단에 적재된 컨테이너가 상단에 적재된 컨테이너보다 먼저 출고되는 일이 없도록 관리한다. 이를 통해 배후단지 야드의 컨테이너 보관과정에서 발생하는 물류비용 및 작업 시간을 줄일 수 있다.

Abstract

In a port logistics environment, containers are stacked in multiple layers to maximize the use of limited space. To handle the lower-layer cargo in a multi-layer stacking situation, the cargo stacked on top must be moved to another location. This process is called container re-handling, and as it incurs costs, container terminal operators aim to minimize re-handling. This paper proposes a reinforcement learning model that recommends and selects loading positions to minimize re-handling while stacking containers in multiple layers. The proposed reinforcement learning model optimizes the loading order by considering the yard departure date of the containers, ensuring that the containers in the lower layer do not depart before those in the upper layer. This can reduce logistics costs and time during the container storage process.

Keywords

port logistics, container yard, container loading, optimizing container handling, reinforcement learning

* (주)컨테인어스 개발자

- ORCID¹: <https://orcid.org/0000-0001-5838-1932>

- ORCID²: <https://orcid.org/0000-0001-8026-3510>

- ORCID³: <https://orcid.org/0000-0002-1307-0521>

** (주)컨테인어스 대표이사

- ORCID: <https://orcid.org/0000-0002-9843-5605>

*** (주)컨테인어스 연구소장(교신저자)

- ORCID: <https://orcid.org/0000-0001-7131-4398>

• Received: Jan. 12, 2023, Revised: Apr. 05, 2023, Accepted: Apr. 08, 2023

• Corresponding Author: Dong-gyu Kim

ContainUs Co., 49-1, Haegwan-ro, Jung-gu, Busan, Republic of Korea

Tel.: +82-70-4940-2040, Email: donggyukim@containus.kr

I. 서 론

항만 물류 환경에서 컨테이너 적재 업무 시 한정된 공간에 최대한 많은 컨테이너를 적재하기 위하여 컨테이너를 다단 적재한다. 컨테이너는 컨테이너 야드에 입고되는 순서대로 적재되므로 출고일을 고려하지 않고 혼재되어 적재된다.

다단 적재되는 경우, 하단에 적재된 컨테이너가 운송되어야 하는 경우가 발생하고 이 경우 해당 컨테이너 상단에 적재된 다른 컨테이너를 모두 옮긴 뒤에 운송할 수 있게 된다. 이때, 발생하는 컨테이너 재취급 작업은 컨테이너 야드 운영자에게 비용을 소모하게 한다. 재취급은 올바른 위치와 순서로 적재된 컨테이너가 반출될 때 작업 소요가 발생하므로 컨테이너가 입고되어 적재될 때 올바른 위치에 적재하면 재취급 작업을 줄일 수 있다. 하지만 순차적으로 입고될 컨테이너의 상세를 예측할 수 없으므로 조합 최적화와 같은 문제로 접근하여 적재 계획을 통하여 최적 적재할 수 없다.

컨테이너 야드에 입고되는 컨테이너의 상세에 따라 현 상태를 고려하여 컨테이너 입고 위치를 결정하도록 하는 강화학습 모델링 방법을 제안하며 그 분석 결과를 제시한다.

II. 관련 연구

2.1 컨테이너 배치 최적화

컨테이너 재취급은 비용을 소모하는 작업이다. 따라서 재취급 최소화의 중요성과 필요성은 꾸준히 증가하고 있다.

컨테이너의 재취급을 최소화하기 위하여 D. Hong은 컨테이너 야드의 여러 베이를 클러스터로 구성하여 동일한 특성이 있는 컨테이너를 적재하도록 컨테이너 야드의 베이 스택을 수치화하여 최적해를 찾는 방식의 최적화 방법을 제안하였다[1]. 제시된 방법은 컨테이너 야드의 베이 스택을 수치화하여 적하 계획을 수립한 뒤 수치화된 베이 스택을 최적이 되도록 적재순서를 결정하고 있다. Hong이 제안한 방법은 적재할 모든 컨테이너의 정보를 알

고 있으므로 적재 시에 전체적인 계획 수립을 통한 적재 최적화를 달성하고 있다. 따라서 임의의 컨테이너가 불규칙적으로 입고되는 컨테이너 야드에서는 사용할 수 없다.

컨테이너 야드 운영자의 경험으로 수립되고 있는 컨테이너 장치장 운용 계획을 대체 하고자 K. T. Kim et al이 제안한 정수 계획모형 또한 컨테이너 재취급을 최소화하고자 제안된 모델이다[2]. Kim은 컨테이너가 장치되면 이후 반출 시점까지 위치가 변동되지 않는다는 가정하에 동작하는 최적화 모형을 제시하고 있으므로 컨테이너의 입고 상황에 따라 효율적으로 장치하기 위한 재취급 학업을 허용하지 않는 제약사항이 있으므로 실제 컨테이너 야드에서 사용할 수 없다.

2.2 컨테이너 처리 속도 최적화

T. Kim et al이 제안한 자동화 컨테이너 터미널 크레인 작업 할당 전략은 터미널 크레인의 컨테이너 작업 시 최대한의 작업 효율을 위하여 제안되었다[3]. 제안된 전략에서는 의도적인 컨테이너 재취급을 통하여 컨테이너 야드의 화물 처리 속도를 향상하고 있다. 컨테이너의 재취급을 통하여 화물의 위치를 최적화하고 화물 처리 속도의 최대화를 목적으로 하므로 재취급 작업을 최소화하는 데에 목적을 두고 있지는 않다. 하지만 화물 처리 속도의 향상을 위하여 재취급 작업의 방지를 위하여 컨테이너 재배치를 실행하고 있다.

2.3 강화학습 알고리즘

Proximal policy optimization[4] 알고리즘은 policy gradient 방법을 사용하는 알고리즘으로 trust region policy optimization[5] 알고리즘의 surrogate suggestion function을 컴퓨터 친화적으로 단순화 시킨 알고리즘이다.

Proximal policy optimization 알고리즘은 초기 파라미터 θ_0 , 초기 value function 파라미터를 ϕ_0 로 두고 스텝을 0에서 증가시키며 정책 $\pi_k = \pi(\theta_k)$ 를 활용하여 trajectory $D_k = \{T_t\}$ 를 누적한다.

T_t 를 누적하는 과정에서 이전 상태에 관한 정보를 획득한다. 또한 동작 이후 획득한 보상인 rewards-to-go \hat{R}_t 를 계산한다. 이후 현재 상태의 가치를 계산하는 value function V_{ϕ_t} 를 활용하여 advantage \hat{A}_t 를 계산한다. 계산한 \hat{A}_t 에 따라 PPO-clip[6]을 최대화하여 정책을 업데이트한다.

따라서 모델은 현재 강화학습 환경에서 자신의 상태를 고려한다. 현재 상태를 기반으로 미래의 누적 보상이 최대가 되는 동작을 선택하고 동작은 환경에 영향을 미치며 동작에 따른 보상을 획득한다. 학습이 진행되면서 모델은 미래에 획득할 보상이 최대가 되도록 매 스텝 동작을 선택하도록 업데이트된다.

III. 강화학습 모델 제안

강화학습 모델은 환경, 행위자, 동작과 동작에 따른 보상 총 4가지 요소를 모델링 하여야 한다. 최적 적재 모델을 위하여 강화학습 모델의 4가지 요소를 다음과 같이 설정하였다. 또한 proximal policy optimization[4] 알고리즘을 활용하여 현재 상태를 고려한 동작을 취할 수 있도록 하였다.

3.1 제안 강화학습 모델의 환경 정의

컨테이너 야드는 5행 5열의 컨테이너 야드 형태로 설정 후 최대 5단까지만 적재하도록 제한하였다 따라서 장치율 100%일 때 컨테이너 최대량은 125개이다. 가상 컨테이너 야드는 그림 1과 같이 구성되었다.

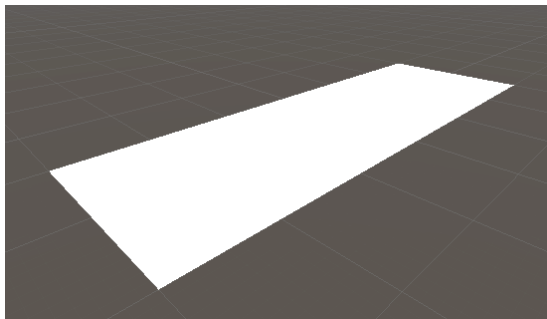


그림 1. 5x5x5 가상 컨테이너 야드
Fig. 1. 5x5x5 virtual container yard

컨테이너 야드에서 평균적으로 공(Empty) 컨테이너, 적(Full) 컨테이너 장치율 75%[7]를 유지하며 컨테이너를 보관하지만 제안하는 모델은 적 컨테이너의 적재만 고려하므로 가상 컨테이너 야드의 장치율 60% 수준까지 컨테이너를 입고하도록 하였다. 따라서 입고될 컨테이너는 125의 60%인 75개이다.

부산신항의 경우 수출 컨테이너의 항만 컨테이너 야드 반입 허용 기한은 모선 입항 3일 전이다. 항만 배후단지의 컨테이너 야드에 같은 조건을 적용하여 항만 컨테이너 야드의 반입 허용 기한과 동일한 3일과 모선의 사정상 입항이 지연될 것을 고려하여 1일의 여유를 포함한 출고 일자 최대 4일 이후까지 인 컨테이너가 보관되도록 하였다. 또한 컨테이너 야드에서 반출되는 시점을 해당 날짜의 오전, 오후 두 가지 시간대로 분류하여 모델링한 가상 컨테이너 야드에 입고되는 컨테이너는 총 8종의 예정 출고일을 가질 수 있도록 하였다.

위의 두 조건에 따라 75개 컨테이너를 8종의 예정 출고일을 가지도록 무작위로 생성하여 차례대로 적재하도록 환경을 구성하였다.

컨테이너 장치율 100%일 때 125개의 컨테이너를 적재할 수 있으며 모든 장치장에 1단 적재만 하였을 경우 25개 컨테이너를 적재할 수 있다. 적재할 75개 컨테이너 중 1단 이상의 컨테이너는 자유롭게 적재하게 하도록 50 스텝의 재취급 기회를 포함한 125 스텝을 최대 스텝으로 정의하였다.

3.2 제안 강화학습 모델의 행위자 정의

최적 적재 모델의 행위자는 컨테이너 야드 전체 상황을 확인 후 동작을 결정하여야 한다. 따라서 행위자는 가상 컨테이너 야드인 5*5*5, 3차원 배열을 감시하도록 설정하였다. 배열은 행, 열, 높이에 따른 해당 위치에 적재된 컨테이너의 출고일 기준 정보를 가지고 있다.

3.3 제안 강화학습 모델의 동작 정의

컨테이너 장치장에서 컨테이너에 대한 동작은 컨테이너를 장치장에 적재하는 동작과 적재된 컨테이너를 반출하는 동작 2가지로 나누어 볼 수 있다.

컨테이너의 동작에 따른 가상 컨테이너 야드의 반영은 그림 2와 같이 이루어지도록 하였다.

컨테이너 입고의 경우 컨테이너를 적재하는 동작 한가지로 볼 수 있으며, 컨테이너의 재취급 작업의 경우 적재된 컨테이너를 반출한 뒤 다른 장치장에 다시 적재하는 2개 동작의 연속으로 볼 수 있다.

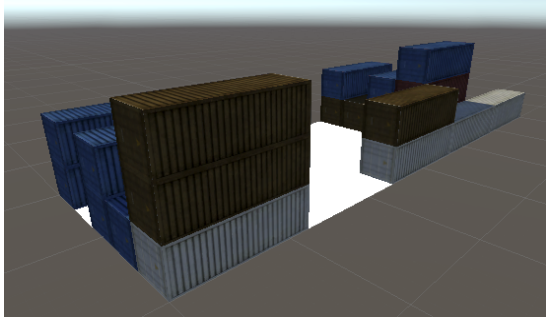


그림 2. 컨테이너의 장치
Fig. 2. Stacking container

3.4 제안 강화학습 모델의 금지 동작 정의

컨테이너의 재취급 과정은 컨테이너의 반출 후 해당 컨테이너를 다시 적재하는 2가지의 연속된 동작으로 정의하였다. 실제 리치 스테커의 업무 순서에 따라 재취급을 위하여 컨테이너를 취급했을 경우 다음 동작은 컨테이너를 적재하는 동작이어야 한다. 만일 적재 동작이 아닌 취급 동작을 다시 시행한다면 이를 금지 동작으로 정의하여 해당 에피소드를 중지 후 환경을 초기화하도록 정의하였다. 이를 통하여 강화학습 모델이 컨테이너 취급 이후 적재 동작을 연속으로 선택하여 동작하도록 하였다. 또한 컨테이너를 적재할 때 상단에서 하단이 내림차순으로 적재 되지 않으면 출고되어야 하는 컨테이너의 상단에 이후 출고될 컨테이너가 적재된 상태가 되어 재취급 행동이 소요되므로 이를 금지 동작으로 정의하였다. 이를 통하여 강화학습 모델이 재취급의 소모를 발생시키지 않도록 하였다.

3.5 제안 강화학습 모델의 보상 정의

컨테이너 야드에서는 가장 적은 횟수의 컨테이너 취급으로 최대한 많은 컨테이너를 적재하고 반

출하도록 해야 하므로 적재 시 반출일 순서로 컨테이너를 적재하여 재취급 작업이 최소화 되도록 해야 한다. 따라서 강화학습의 환경에서 반출일 순서로 컨테이너를 적재할 때 보상을 1씩 부여하도록 설정하였다. 컨테이너의 재취급을 최소화하기 위하여 컨테이너를 반출 하는 동작의 경우 보상을 부여하지 않았다. 따라서 보상은 식 (1), (2)와 같이 나타낼 수 있다.

$$R_{S_{loading}} = 1 \quad (1)$$

$$R_{S_{unloading}} = 0 \quad (2)$$

금지 동작을 하지 않고 정상적으로 모든 컨테이너를 적재하는 경우 에피소드 종료 시점의 스택과 설정한 최대 스택 125와의 차이를 보상으로 부여하였다. 지정된 60% 장치율에 도달하는 경우 추가 보상은 식 (3)과 같이 나타낼 수 있다.

$$R_{Additional} = S_{max} - S_{now} \quad (3)$$

이를 통하여 강화학습 모델이 최대한 많은 보상을 획득하기 위하여 최대한 많은 컨테이너를 적재하는 방향으로 동작을 선택하도록 하였다. 모델은 매 스택에서 동작을 취하며 컨테이너를 쌓을 때 보상을 획득한다. 하지만 보상을 얻지 못하는 컨테이너 반출 동작이 포함된 재취급 작업을 할 때는 보상을 얻지 못하므로 모델의 학습 중 75개의 컨테이너를 재취급 작업 없이 모두 적재할 때 획득 가능한 보상 75와 에피소드 종료 시 부여되는 해당 스택과 최대 스택의 차이 50으로 총 125가 환경에서 획득 가능한 최대 보상으로 식 (4)와 같다. 이는 강화학습 모델에서 임의로 정의한 최대 스택인 125와 같고 에피소드에서 획득하는 보상은 식 (5)와 같이 나타낼 수 있다.

$$S_{max} = 125 \quad (4)$$

$$R = \sum_{step=0}^{125} [R_{S_{loading}} + R_{S_{unloading}}] \quad (5)$$

IV. 제안 강화학습 모델 학습 및 결과

최적 적재 모델의 학습을 위하여 [8]-[11]을 활용하여 알고리즘을 결정하였으며 학습 알고리즘으로 proximal policy optimization 알고리즘을 사용하였다.

3억 스텝 동안 학습을 진행 후 생성된 모델을 분석하였으며 학습 시간은 약 3일이 소요되었다. 학습에 사용된 서버의 사양은 표 1과 같다.

표 1. 러닝 서버 사양

Table 1. Learning server specs

CPU	i7-11700 KF
GPU	CUDA Cores 8960 RTX 3080·12GB*2
RAM	64GB

강화학습 모델은 Unity ML을 활용하여 작성되었으며 함께 제공되는 ML-Agent Python 패키지가 사용되었다. 강화학습의 알고리즘인 Proximal Policy Optimization 알고리즘에 적용한 하이퍼 파라미터는 표 2와 같다.

표 2. Proximal policy optimization 하이퍼 파라미터

Table 2. Proximal policy optimization hyper parameter

Gamma	0.99
Lambda	0.95
Buffer size	204,800
Batch size	256
Number of epochs	7
Learning rate	1e-4
Time_horizon	125
Max steps	1e-6
Beta	5.0e-3
Epsilon	0.1

그림 3을 통하여 각 스텝 구간에서 종료된 에피소드들이 획득한 보상의 빈도를 확인할 수 있다. 학습 과정 중 무작위 생성된 컨테이너 75개를 모두 적재하기 시작한 에피소드는 약 1억1,347만 회 스텝에서 나타났다.

그림 4 모델의 누적 보상 평균을 통하여 학습이 진행되면서 에피소드들이 높은 보상을 획득해 가는 것을 확인할 수 있다.

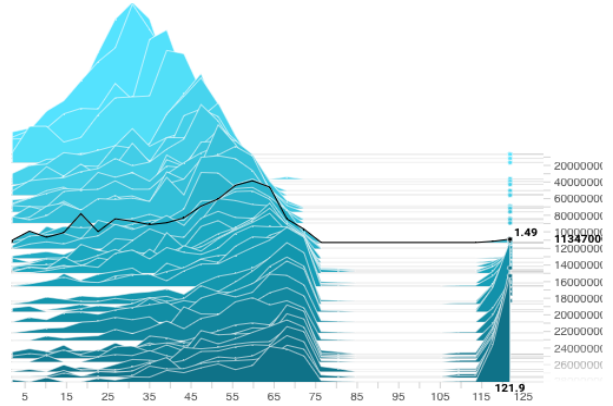


그림 3. 누적 보상 히스토그램

Fig. 3. Cumulative reward histogram

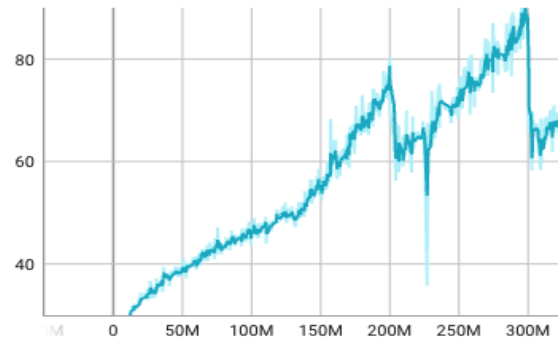


그림 4. 누적 보상 평균

Fig. 4. Cumulative reward average

학습이 완료된 모델을 활용하여 학습 시와 같은 조건에서 모델에게 추론만 실행하도록 하여 모든 컨테이너를 성공적으로 적재하는 500개 에피소드의 결과는 표 3과 같다.

모델이 컨테이너를 모두 성공적으로 적재하였을 때 최소 112, 최대 125의 보상을 획득하였다. 평균은 123.466이며 표준편차는 1.918로 나타났다.

표 3. 학습된 모델의 성능 테스트 결과

Table 3. Performance results of trained model

	MAX	MIN	Average	Standard deviation
Reward	125	112	123.466	1.918

재취급은 평균 0.5회 실행하며 가장 많은 재취급을 한 경우 최대 6회 시행하였으며 표준편차는 0.957로 나타났다.

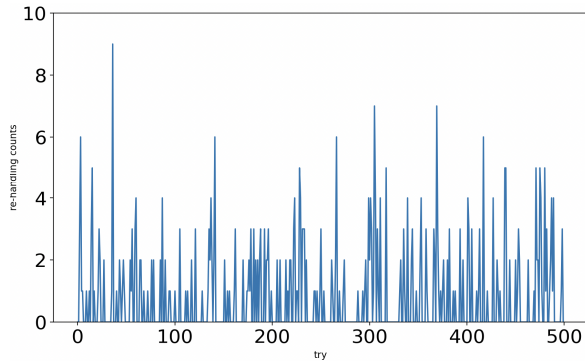


그림 5. 테스트 시행 별 재취급 횟수
Fig. 5. Number of re-handling per test run

모델은 허용된 컨테이너 재취급 최대 횟수인 50회의 재취급을 모두 시행하지 않고 적재 시점에서 환경상의 제약을 피하여 적재할 수 없는 경우 재취급을 시행하도록 학습되어 500회의 테스트 수행 간 평균 0.832회 재취급을 시행하였으며 최대 9회 최소 0회 실시하고 있다.

제안된 모델은 무작위 컨테이너가 순차 입고되며 입고된 컨테이너를 취급하여 다단 적재하고 있으므로 Hong[1]이 제안한 모델의 단점인 입고 컨테이너에 대한 상세를 알 수 없어 적하계획을 수립하지 못하는 상황에서 효율적인 컨테이너 장치가 가능하다. 또한 컨테이너의 입고와 컨테이너 야드의 장치 상황에 따라 재취급을 진행하고 있으므로 Kim[2]이 제안한 모델의 가정사항인 컨테이너 장치 후 재취급을 하지 않음을 개선하고 있다.

Kim[3]이 제안한 장치장 크레인의 작업 할당 전략 최적화의 경우 60% 장치율에서 평균 63회 재취급 작업을 시행하고 있다. 제안 모델 또한 동일한 60% 장치율 조건에서 평균 0.832회 핸들링을 시행하고 있으므로 제안 모델은 재취급 작업을 최소화하고 있음을 알 수 있다.

V. 결론 및 향후 과제

제안된 모델링 방법은 컨테이너 야드를 모형화하고 실제 컨테이너 야드에서 시행하는 업무 조건을 반영하여 컨테이너를 적재하는 강화학습 모형의 설계 방법이다. 모형의 설계는 효율적인 컨테이너 적재를 통하여 재취급의 최소화를 목적으로 하고 있

다. 제시된 방법을 통하여 실험 환경인 5 X 5 X 5 크기의 테스트 컨테이너 야드를 구현하고 최대 장치율 60% 조건에서 컨테이너를 적재하도록 학습된 모델을 테스트한 결과 최대한 많은 컨테이너를 적재하기 위하여 부분적으로 재취급을 시행하고 있으며 적재된 컨테이너는 출고일 순으로 적재되어 있으므로 출고 시에 별도의 재취급이 필요하지 않은 결과를 확인했다. 컨테이너가 임의의 시점에 입고되는 컨테이너 야드를 고려하여 모델링 되었으므로 민간 컨테이너 야드에 적합하며 컨테이너 재취급을 최소화할 수 있다.

Acknowledgement

이 논문은 2022년도 한국정보기술학회 추계 종합 학술대회에서 발표한 논문(항만 물류 환경에서 강화학습 기반 컨테이너 다단 적재 모델링 방법)[12]을 확장한 논문입니다.

References

- [1] D. Hong, "The Method of Container Loading Scheduling through Hierarchical Clustering", *Journal of The Korea Society of Computer and Information*, Vol. 10, No. 1, pp. 201-208, Mar. 2005.
- [2] K. T. Kim, K. M. Kim, T. Y. Lee, and D. H. Kim, "An Optimization Model for Minimizing the Number of Rehandles at Container Yard", *Proceedings of the 2011 Autumn Conference of The Korean Society for Railroad*, Oct. 2011.
- [3] T. Kim, Y. Yang, A. Bae, and K. R. Ryu, "Optimization of Dispatching Strategies for Stacking Cranes Including Remarshaling Jobs", *Journal of Navigation and Port Research*, Vol. 38, No. 2, pp. 155-162, Apr. 2014. <https://doi.org/10.5394/KINPR.2014.38.2.155>
- [4] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal Policy Optimization Algorithms", arXiv: 1707.06347, Jul. 2017. <https://>

doi.org/10.48550/arXiv.1707.06347.

- [5] J. Schulman, S. Levine, P. Abbeel, M. I. Jordan, and P. Moritz, "Trust Region Policy Optimization", arXiv: 1502.05477, Feb. 2015. <https://doi.org/10.48550/arXiv.1502.05477>.
- [6] M. Farsang and L. Szegetes "Decaying Clipping Range in Proximal Policy Optimization", arXiv, 2102.10456, Timisoara, Romania, May 2021. <https://doi.org/10.1109/SACI51354.2021.9465602>.
- [7] S. Jung, J. Lee, M. Sim, and Y. Kim "Eploratory Study of the Factors Affecting Yard Occupancy in Container Terminal", Journal of Shipping and Logistics, Vol. 114, pp. 137-156, Sep. 2022.
- [8] A. Slivkins, "Introduction to Multi-Armed Bandits", arXiv: 1904.07272, Apr. 2019. <https://doi.org/10.48550/arXiv.1904.07272>.
- [9] Y. Li, "Deep Reinforcement Learning: An Overview", arXiv: 1701.07274, Jan. 2017. <https://doi.org/10.48550/arXiv.1701.07274>.
- [10] Y. Li, "Reinforcement Learning in Practive: Opportunities and Challenges", arXiv: 2202.11296, Feb. 2022. <https://doi.org/10.48550/arXiv.2202.11296>.
- [11] V. Mnih, K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, and D. Wiersta, "Playing Atari with Deep Reinforcement Learning", arXiv: 1312.5602, Dec. 2013. <https://doi.org/10.48550/arXiv.1312.5602>.
- [12] W. Jang, H. Lee, M Seo, S Lee, and D. Kim, "Reinforcemnet learning-based container multi-stage loading modeling method in port logistics", Proceedings of KIIT Conference, Jeju, Korea, Vol. 27, pp. 131-134, Dec. 2022.

저자소개

장 우 석 (Woo-Seok Jang)



2023년 2월 : 금오공과대학교
컴퓨터공학과(공학사)
2022년 9월 ~ 현재 :
(주)컨테이너스 개발자
관심분야 : 백엔드 아키텍처, 딥러닝, 항만 물류, 신호처리

이 효 준 (Hyo-June Lee)



2014년 2월 : 부산대학교
IT응용공학과(공학사)
2021년 10월 ~ 현재 :
(주)컨테이너스 개발자
관심분야 : 사물인터넷, 인공지능, 항만 물류

서 민 성 (Min-Seong Seo)



소프트웨어 개발

2021년 2월 : 부산대학교
IT응용공학과(학사)
2021년 10월 ~ 2022년 03월 :
(주)NGL 빅데이터팀 개발자
2022년 4월 ~ 현재 :
(주)컨테이너스 개발자
관심분야 : 사물인터넷, 빅데이터,

이 성 진 (Seong-Jin Lee)



(주)컨테이너스 대표이사
관심분야 : 클라우드, 블록체인, 사물인터넷

2018년 9월 : 부산대학교
IT응용공학과(학사)
2018년 6월 ~ 2020년 3월 :
(주)스마트엠투엠 연구원
2020년 7월 ~ 2020년 11월 :
엔지엘(주) 팀장
2020년 12월 ~ 현재 :

김 동 규 (Dong-Gyu Kim)



2011년 2월 : 부산대학교
전자전기컴퓨터공학부(공학사)
2018년 2월 : 부산대학교
전자전기컴퓨터공학과(박사)
2018년 3월 ~ 2018년 9월 :
부산대학교
컴퓨터정보통신연구소 박사 후

과정

2018년 10월 ~ 2022년 1월 : (주)스마트엠투엠
책임연구원/연구소장

2022년 03월 ~ 현재 : (주)컨테이너스 연구소장
관심분야 : 블록체인, 항만 물류, 인공지능, 신호처리