3D BinPacking with RL

Learning Efficient Online 3D Binpacking on Packing Configuration Trees

2022.09.23 정규헌

이전 내용...

Paper Review
세미나의 목적

- 강화 학습을 공부했는데
- 산업AI에 어떻게 적용하지?
  - 조합최적화
- survey논문을 읽어서 동향을 파악해보자

RL combinatorial optimization

검색결과 약 191,000개 (**0.10**초)

[HTML] Reinforcement learning for **combinatorial optimization**: A survey
N Mazyavkina, S Sviridov, S Ivanov… - Computers & Operations …, 2021 - Elsevier
… of the **RL** field to solve CO problems. Although many practical **combinatorial optimization** … operations research community, we will focus on **RL** approaches for CO problems. This survey …
☆ 저장 인용 159회 인용 관련 학술자료 전체 6개의 버전

ELSEVIER

Computers & Operations Research
Volume 134, October 2021, 105400

Reinforcement learning for combinatorial optimization: A survey ☆

Nina Mazyavkina [a], Sergey Sviridov [b], Sergei Ivanov [c, a], Evgeny Burnaev [a]

RL for CO

Combinatorial Optimization ?

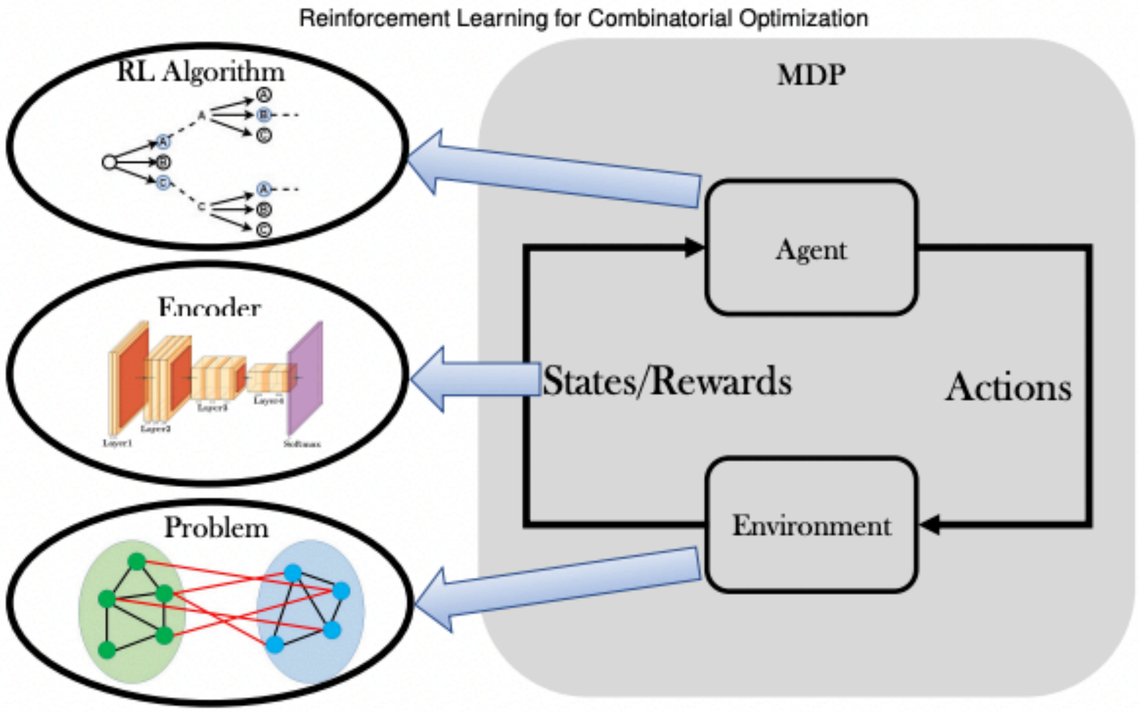
- **Combinatorial Optimization(CO) problem** : 최적화의 한 분야
  - 탐색공간에서 최적의 해를 찾는 문제
    - discrete space의 최적화 문제
    - continuous space의 최적화 문제와는 다른 유형의 솔루션을 가짐
  - **다양한 산업분야**에서 다루고 있는 문제
    - 기존 최적화 알고리즘에는 문제마다 다른 휴리스틱을 사용하는 것이 포함
      - 휴리스틱 : 도메인 전문가에 의해 설계됨
    - RL은 agent를 훈련하여 이러한 휴리스틱 방법들을 자동화 가능

RL for CO

Pipeline for solving CO with RL

우리가 풀어야 할 문제

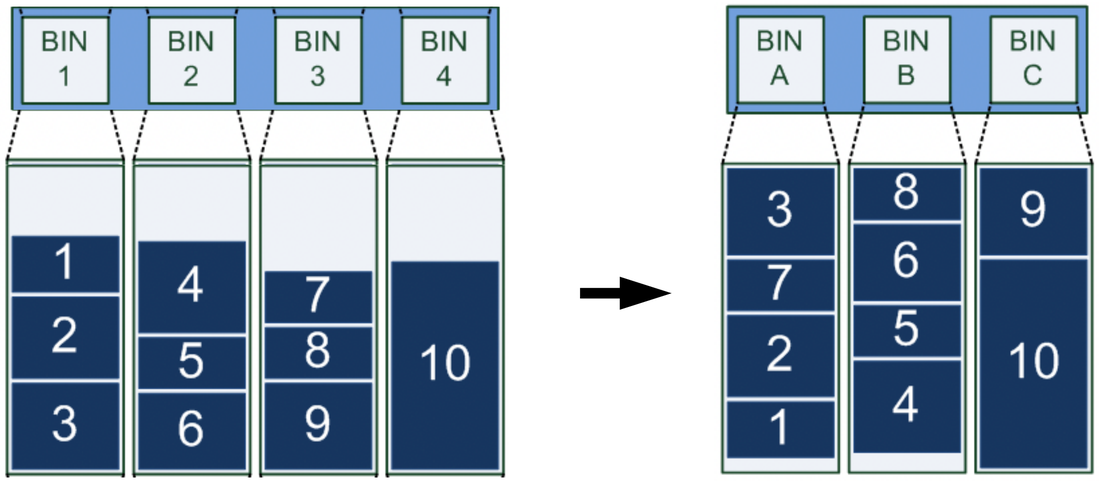
- Environment : MDP
  - State / Reward : Environment에 Action을 주고 얻음
- RL algorithm
  - State, Reward등을 통해 Action, state에 대한 가치를 구함
  - optimal policy를 선택함 - 어떤 Action을 할지
- Encoder
  - 고차원문제의 input space를 encoding하는데 활용



RL for CO

Canonical example of CO

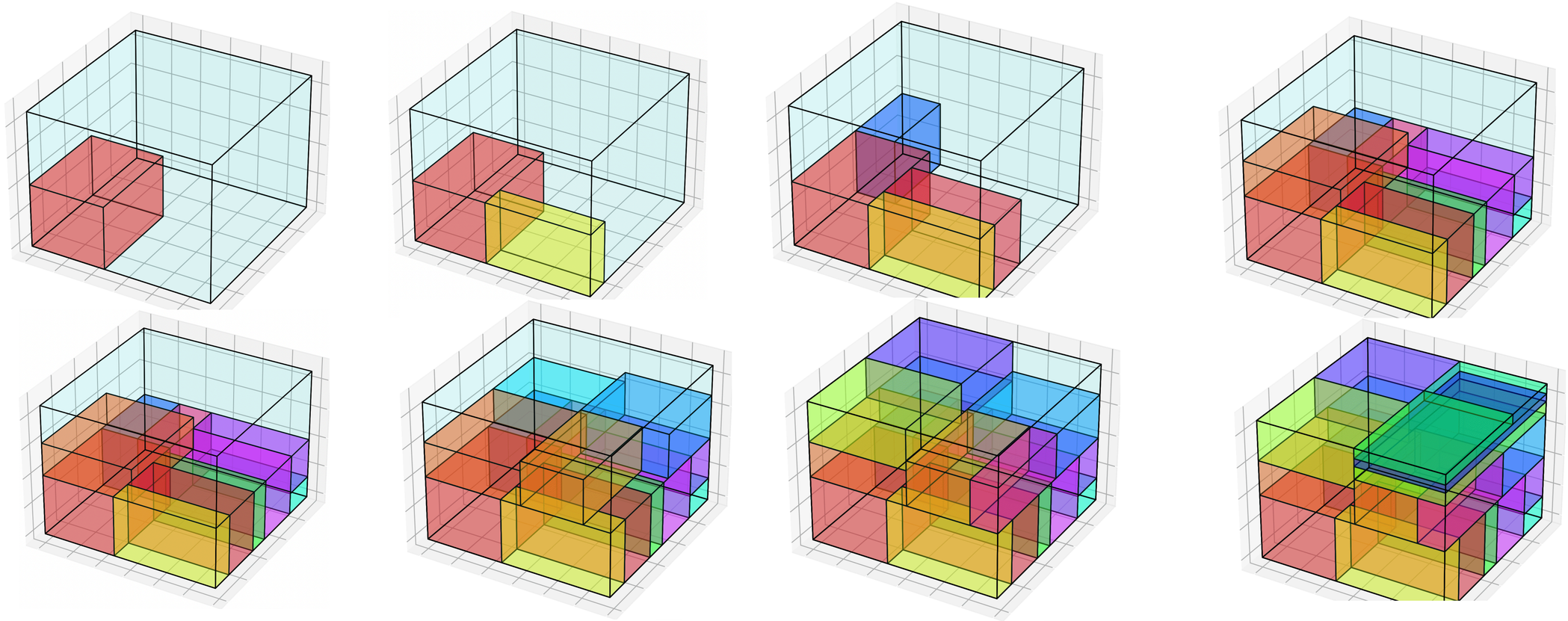
- Bin Packing Problem(BPP)
  - 상자 채우기 문제
  - 최소한의 공간에 최대한의 아이템을 채워넣는 문제
- 활용 분야
  - 1차원 : 메모리 할당
  - 2차원 : 그래픽 리소스 최적화
  - 3차원 : 화물 적재



RL for CO

Canonical example of CO

- Bin Packing Problem(BPP)
- 3D binpacking example



RL for CO

Conclusion

- 적절한 강화학습 알고리즘 선택
  - 현실 문제를 **MDP로 만들었을 때**, state의 sequence가 명확하게 markovian을 따르는지 안따르는지
  - 전체 Trajectory의 performance를 **하나의 error가 영향을 많이 끼치는지**?
    - **CO 문제 : policy based가 좀 더 적합**
- 선행 연구를 참고 하는 것이 중요
  - 강화학습은 적절한 하이퍼파라미터에 따라 성패여부가 갈림
  - 논문을 꼼꼼히 확인해야 함
  - 관련된 논문에서 **MDP parameter, hyperparameter, trick들을 참고**해야함

??? : 3D binpacking 저걸로 논문 쓰면 되겠네?

- 선행 연구를 참고 하는 것이 중요
  - 강화학습은 적절한 하이퍼파라미터에 따라 성패여부가 갈림
  - 논문을 꼼꼼히 확인해야 함
  - 관련된 논문에서 **MDP parameter, hyperparameter, trick들을 참고**해야함

3D-BinPacking을 RL으로 해결한 논문?

한줄기 빛...

Published as a conference paper at ICLR 2022

LEARNING EFFICIENT ONLINE 3D BIN PACKIN ON PACKING CONFIGURATION TREES

Hang Zhao[1,2*], **Yang Yu**[2], **Kai Xu**[1†]
[1]School of Computer Science, National University of Defense Technology, China
[2]National Key Lab for Novel Software Technology, Nanjing University, China
{alex.hang.zhao, kevin.kai.xu}@gmail.com, yuy@nju.edu.cn

이게 최신 업그레이드 버전이라고함

설명이 옆에 논문에 비해 친절함. 그래서 이걸 리뷰하는걸로

The Thirty-Fifth AAAI Conference on Artificial Intelligence (AAAI-21)

Online 3D Bin Packing with Constrained Deep Reinforcement Learning

Hang Zhao[1], **Qijin She**[1], **Chenyang Zhu**[1], **Yin Yang**[2], **Kai Xu**[1*]
[1]National University of Defense Technology
[2]Clemson University

인용수 27 가장 많음

README.md

Github에 코드도 공개되어 있고

Learning Efficient Online 3D Bin Packing on Packing Configuration Trees

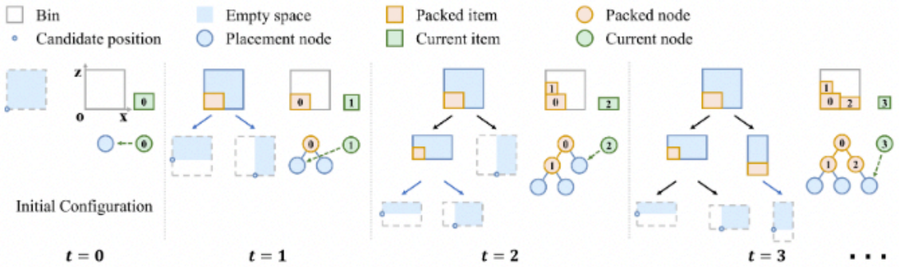
This repository is fully published now, all kinds of feedback and any contribution is welcome!

We propose to enhance the practical applicability of online 3D bin packing problem (BPP) via learning on a hierarchical packing configuration tree which makes the deep reinforcement learning (DRL) model easy to deal with practical constraints and well-performing even with continuous solution space. Compared to our previous work, the advantages of this repo are:

- Container (bin) size and item sizes can be set arbitrarily.
- Continuous online 3D-BPP is allowed and the continuous environment is provided.
- Algorithms to approximate stability are provided (see our other work).
- Better performance and the ability to account for more complex constraints.
- More adequate heuristic baselines for domain development.
- More stable training.

See these links for video demonstration: YouTube, bilibili

If you are interested, please star this repo!



Paper

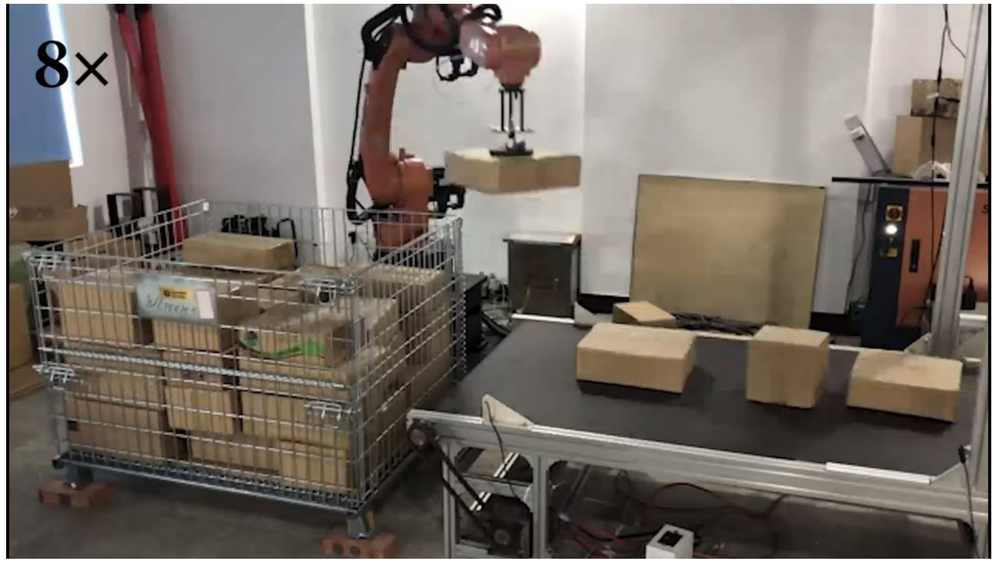
For more details, please see our paper Learning Efficient Online 3D Bin Packing on Packing Configuration Trees which has been accepted at ICLR 2022. If this code is useful for your work, please cite our paper:

```
@inproceedings{
zhao2022learning,
title={Learning Efficient Online 3D Bin Packing on Packing Configuration Trees},
author={Hang Zhao and Yang Yu and Kai Xu},
booktitle={International Conference on Learning Representations},
year={2022},
url={https://openreview.net/forum?id=bfuGjlCwAq}
}
```

논문도 appendix에 정말 상세한 설명이 추가 되어있음

In this appendix, we provide more details and statistical results of our PCT method. Our real-world packing demo is also submitted with the supplemental material.

- Section A gives more descriptions about training methods, which include the implementation of the deep reinforcement learning method, specific GAT network designs for extracting problem features from PCT, and recommendations for finding suitable PCT length.
- Section B elaborates the concept of the leaf node expansion schemes adopted for finding candidate placements in our method. Learning curves and the computational complexity analysis of these schemes are also provided in this section.
- Section C reports more statistical results of our method, including further discussions on generalization ability, running costs, and scalability. The understanding of model behaviors, the visualized results, and details of our real-world packing system are also provided.



친절한데 난이도는 안친절함

논문 리뷰(이론편) 시작합니다..!

Published as a conference paper at ICLR 2022

LEARNING EFFICIENT ONLINE 3D BIN PACKING ON PACKING CONFIGURATION TREES

Hang Zhao[1,2*], **Yang Yu**[2], **Kai Xu**[1†]
[1]School of Computer Science, National University of Defense Technology, China
[2]National Key Lab for Novel Software Technology, Nanjing University, China
{alex.hang.zhao, kevin.kai.xu}@gmail.com, yuy@nju.edu.cn

Learning Efficient online 3D-BPP on PCT

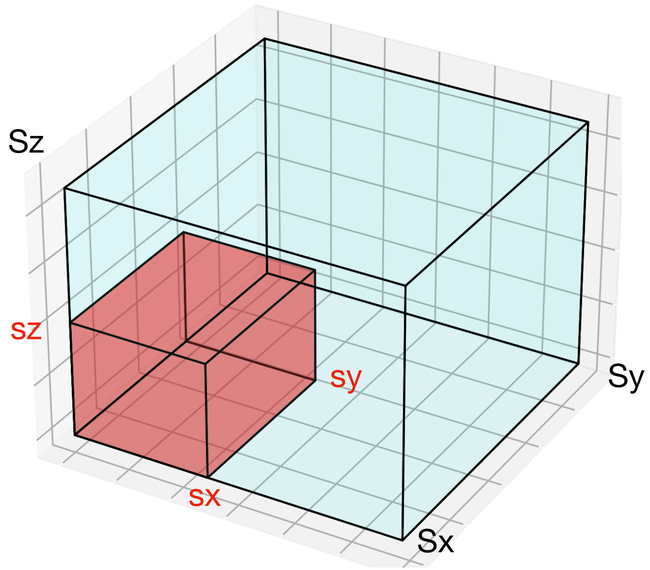
abstract

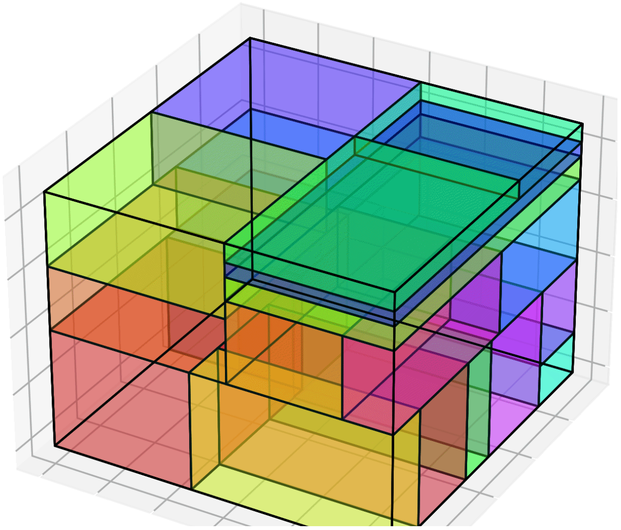
- **3D-BPP**(3D binpacking problem)은 산업 자동화분야에서 널리 응용되고 있음
- 기존 방법의 한계
  - Spartial discretization 문제
    - limited resolution(제한된 해상도)로 문제를 해결
    - Continuous solution space에서 잘 작동하지 않음
  - 복잡한 실제 constraints를 잘 처리하지 못함
- 연구의 주요 내용
  - 새로운 hierarchical representation을 제안 - packing configuration tree(PCT)
    - binpacking의 state, action space에 대한 full-fledged description.
  - PCT는 DRL(Deep reinforcement learning) 기반 policy learning을 지원 가능
    - 강화학습 모델 학습에 용이하고 성능이 좋다

Learning Efficient online 3D-BPP on PCT

Introduction

- **3D-Bin Packing Problem**
  - 가장 고전적인 CO문제중 하나 - 우리만 처음 들음..
  - x,y,z 축을 따라 Item i의 set I를 최소한의 Bin의 수로 packing 하는 문제
  - Item size : sx sy sz
  - Bin size : Sx Sy Sz





Learning Efficient online 3D-BPP on PCT

Introduction

- **Offline 3D BPP (tradditional 3D-BPP)**
  - 모든 item들이 선험적으로 알려져 있다고 가정
    - 향후에 들어올 item을 모두 알아야 실제 산업에 적용이 가능
  - 임의의 순서로 item을 배치할 수 있음
- **Online 3D BPP**
  - 현재 item만 관찰하고 배치를 진행
  - item의 순서에 따라 항목을 그때그때 배치해야함
    - 추가 constraints가 필요

Learning Efficient online 3D-BPP on PCT

Introduction

- **Online 3D BPP의 solution**
  - **Heuristic methods**
    - 일반적으로 action space의 size에 제한을 받지 않음
    - Packing stability, specific packing preferences 같은 복잡한 실제 제약 처리 힘듬
  - **Learning based methods**
    - 일반적으로 다양하고 복잡한 제약조건 상에서 휴리스틱보다 우세
    - Large action space에서는 학습 수렴이 힘들어 제한
      - Ex, the limited resolution of spatial discretization(Zhao et al.,2021)

Learning Efficient online 3D-BPP on PCT

Introduction

- **PCT**(Packing Configuration Tree)
  - Dynamically growing tree, hierarchical representation
    - Internal nodes : packed item’s space configuration
    - Leaf nodes : current item’s packable placements
  - DRL기반의 packing policy learning을 지원
    - binpacking state, action space에 대한 full-fledged description
- PCT로 부터 state features 추출 가능
  - Graph attention networks를 활용, spatial relations를 encoding
  - State features : DRL model의 actor-critic networks의 input으로 사용

Learning Efficient online 3D-BPP on PCT

Introduction

- PCT는 state, action space에 대한 정보를 담고 있음
  - Graph attention networks를 활용, spatial relations를 encoding, state feature를 추출
  - DRL model의 actor-critic networks의 input으로 사용
    - Actor network : pointer mechanism 기반
    - output : Action(final placement)을 출력
  - **Continuous한 solution space에서**
  - **online 3D BPP를 성공적으로 해결하기 위한**
  - **Learning-based method를 배포한 첫번째 연구**

*pointer mechanism

*Actor-Critic network

Learning Efficient online 3D-BPP on PCT

Reference : Offline 3D-BPP

- 모든 item을 알고있고, 임의의 순서로 배치가능
- Extract branch-and-bound approach를 활용해 처음 해결 (Martello et al. 2000)
- 제한된 상황에서 대략적인 solution을 빠르게 얻기 위한 heuristic, meta-heuristic 알고리즘들이 연구됨
- Offline 3D-BPP를 Packing order decision, online 3D-BPP로 나누어서 해결 (Hu et al., 2017)
  - step 1 : Packing order - RL agent로 optimize
  - step 2 : Online placement policy - hand designed heuristic
  - 해당 방법이 널리 받아들여짐 - Duan et al(2019), Hu et al(2020), Zhang et al(2021)

Learning Efficient online 3D-BPP on PCT

Reference : Heuristics for Online 3D-BPP

- Offline 3D-BPP가 꾸준히 연구되었지만, search based를 online setting으로 바로 이전하기 힘듬
- Online에 맞는 다른 heuristic methods들이 제안됨
  - **DBL(the deep-bottom-left) heuristic** - (Karabulut & Inceoglu, 2004) 오랫동안 선호
    - DBL 순서로 빈공간을 정렬, current item을 첫번째 항목에 배치 (Ha et al.,2017)
  - Heightmap-Minimization method - Wang&Hauser(2019b)
    - 적재 방향에서 packing된 item의 volume increase를 최소화
  - Maximize-Accessible-Convex Space method - Hu et al(2020)
    - Packing future에 사용할 수 있는 empty space를 optimize

Learning Efficient online 3D-BPP on PCT

Reference : DRL for Online 3D-BPP

- 휴리스틱 방법의 구현은 직관적, 다양한 시나리오에 쉽게 적용이 가능
  - The price of good flexibility : Online 3D-BPP with specific constraints에서 그리 좋지 않음
  - 3D-bpp의 specific한 classes를 위한 new heuristic을 고안하는 것도 힘듬
- Online 3D-bpp에서 잘 작동하는 policy를 자동으로 생성하려면?

NP-hard solution space, 많은 상황들이 시행착오를 통해 수동으로 미리 계획되어야 함
safety, reliablity를 보장하려면 상당한 도메인 지식도 필수적

  - Verma et al(2020); Zhao et al(2021) - DRL 활용, small discrete coordinate spaces로 한정
  - Zhang et al(2021) - Hu et al(2017) 참조, offline packing needs와 유사한 online placement policy 채택
    - learning-based methods 오직 limited discretization accuracy를 가진 grid world에서만 작동

Learning Efficient online 3D-BPP on PCT

Reference : Practical Constraints

3D-BPP에서 권위적인 연구

- Two Constraints - (Martello et al.,2000)

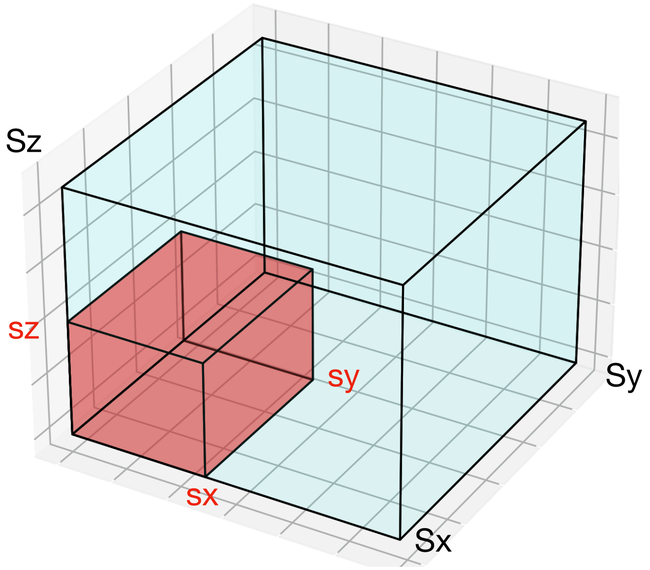
$$p_i^d + s_i^d \leq p_j^d + S^d(1 - e_{ij}^d) \quad i \neq j, i, j \in \mathcal{I}, d \in \{x, y, z\}$$ Basic Non-overlapping constraints 1

$$0 \leq p_i^d \leq S^d - s_i^d \quad i \in \mathcal{I}, d \in \{x, y, z\}$$ containment constraints 2

p_i^d : item i의 front-left-bottom coordinate

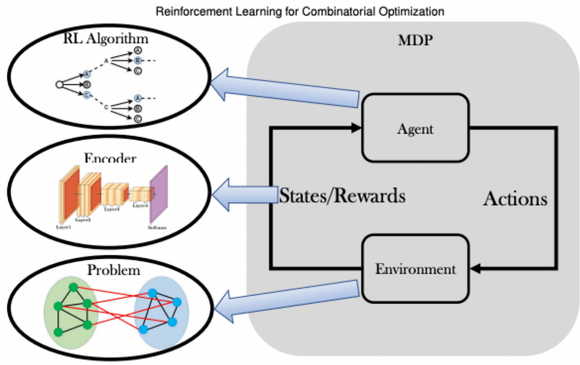
s_i^d : item i의 size

e_{ij}^d : item i가 item j보다 선행하는 경우 0, 아니면 1



Learning Efficient online 3D-BPP on

Method - Packing Configuration Tree



MDP

Agent

Encoder

RL Algorithm

Action

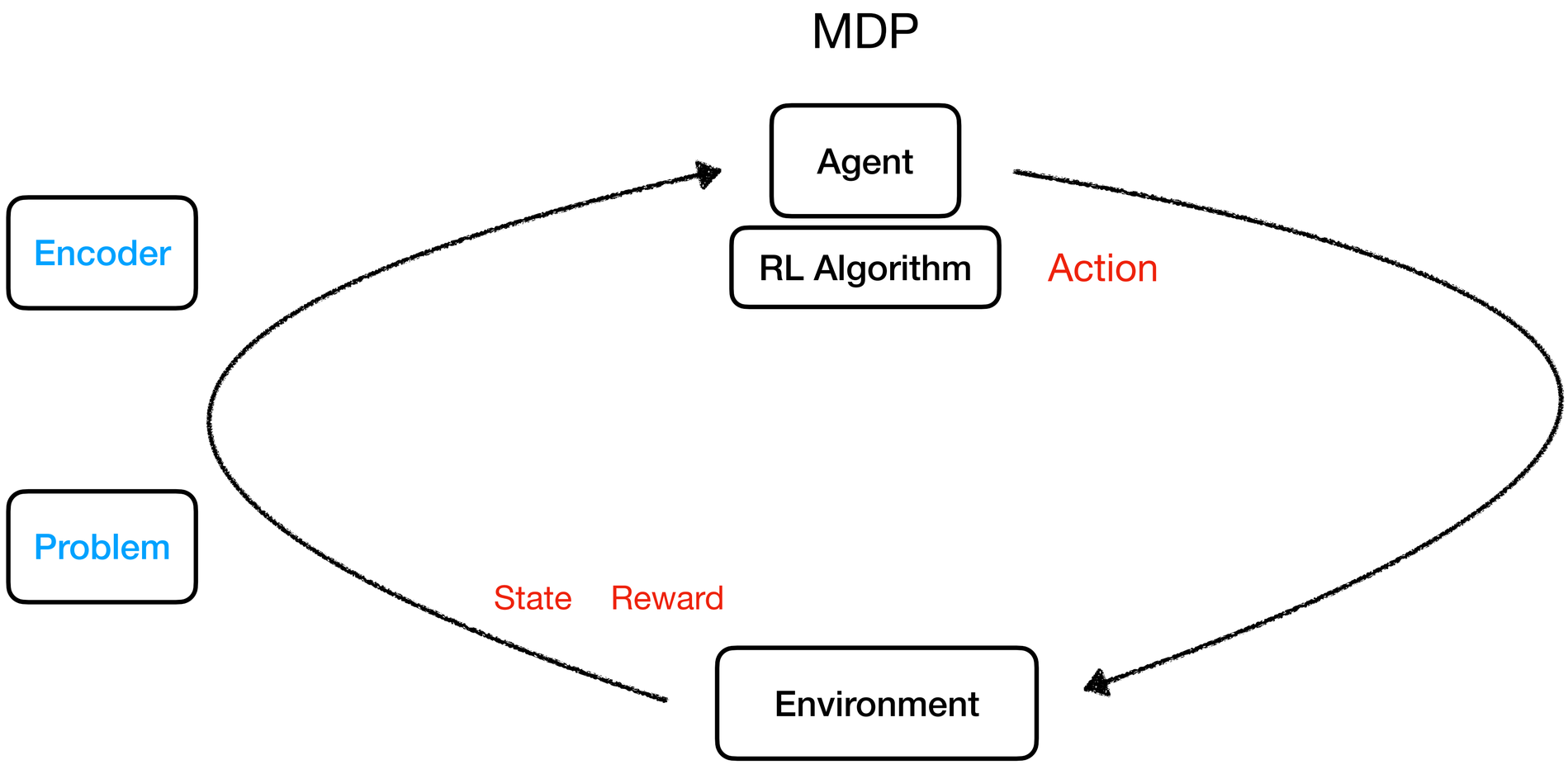
Problem

State Reward

Environment

Learning Efficient online 3D-BPP on PCT

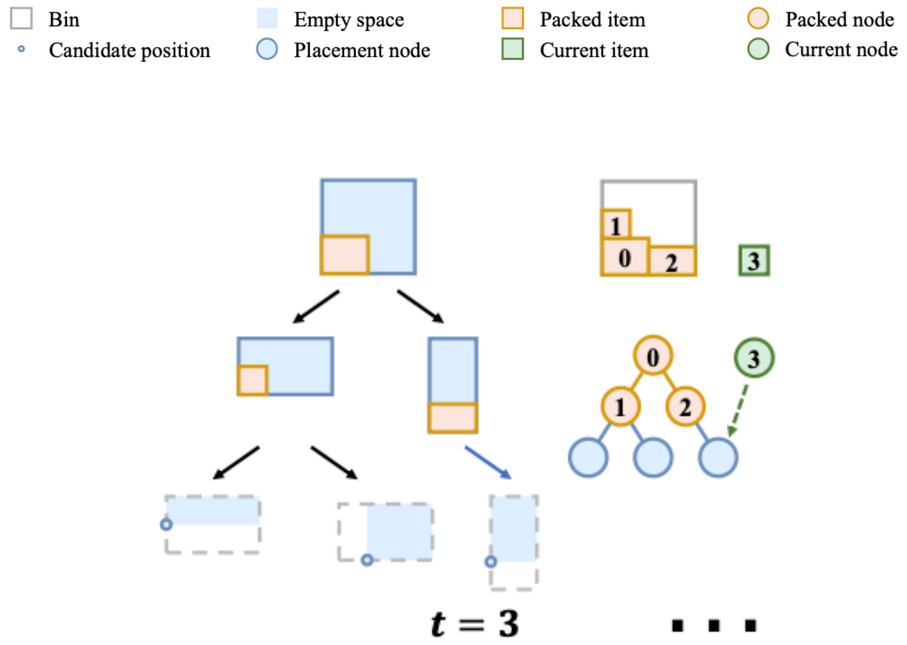
Method - Packing Configuration Tree



Learning Efficient online 3D-BPP on PCT

Method - Packing Configuration Tree

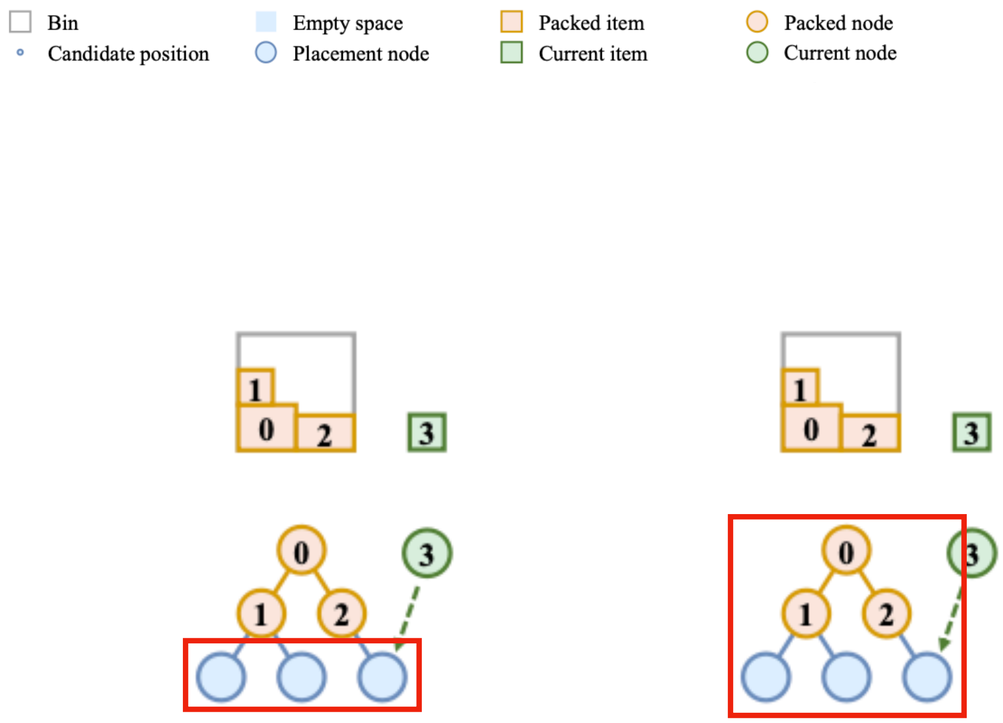
- Packing Time step t
- Packing configuration Tree : T_t
  - Bin configuration
- current item : n_t
- packing position : (p_n^x, p_n^y, p_n^z)
- Vertical top-down packing *(Wang & Hauser, 2019b)*
- Internal node set : B_t
  - Space configuration을 표현
- Leaf node set L_t
  - packing가능한 candidate placements



Learning Efficient online 3D-BPP on PCT

Method - Packing Configuration Tree

- Traditional 3D-BPP
  - Current item nt를 수용하기 위한 remaining placements에만 관심을 두고 있음
  - Policy = π(Lt|Lt,nt)
    - 기존 leaf node와 nt가 주어졌을 때 Lt를 선택할 확률
- PCT 3D-BPP
  - Policy = π(Lt|Tt,nt)
    - Tree와 nt가 주어졌을 때 Lt를 선택할 확률
  - 추가적인 공간정보를 포함시킴
  - Packing 안정성이 강화됨



Learning Efficient online 3D-BPP on PCT

Method - Packing Configuration Tree

- Leaf Node Expansion Scheme
  - item nt 배치 후 새로운 후보 placements(Lt)를 계산하는 방식
  - 최적의 expansion scheme
    - feasible한 packing을 놓치지 않으면서
    - exploring할 solution의 수를 줄여야 함
- 기존 연구들로 입증된 scheme을 PCT expansion으로 확장
  - Corner Point(Martello et al., 2000)
  - Extreme Point(Crainic et al., 2008)
  - Empty Maximal Space(Ha et al., 2017)

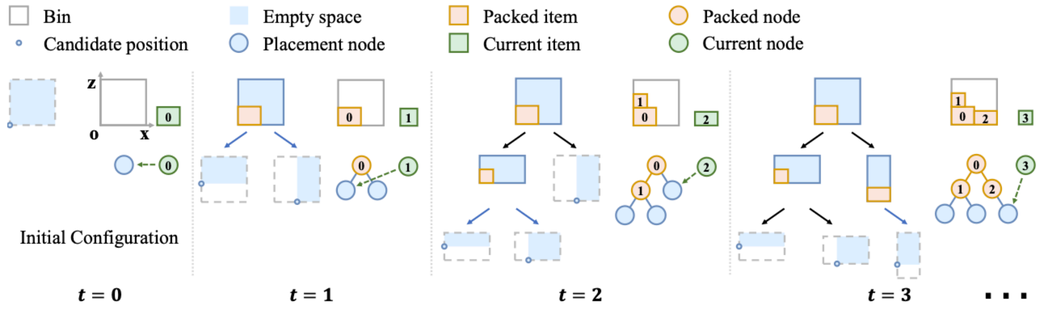
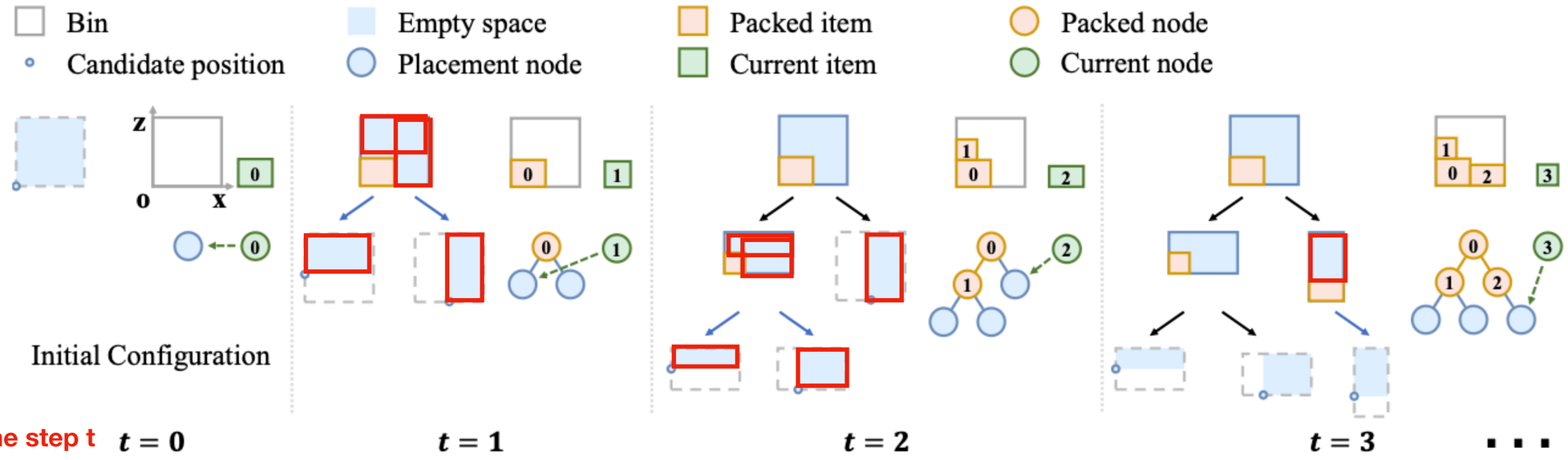


Figure 1: PCT expansion illustrated using a 2D example (in xoz plane) for simplicity and the number of allowed orientations $|\mathbf{O}|$ is 1 (see Appendix B for the 3D version). A newly added item introduces a series of empty spaces and new candidate placements are generated, e.g., the left-bottom corner of the empty space.

Learning Efficient online 3D-BPP on PCT

Method - Packing Configuration Tree

Dynamic Growing Tree, PCT(Packing Configuration Tree)



더 이상 packable한 leaf node가 없으면 episode 종료

Figure 1: PCT expansion illustrated using a 2D example (in xoz plane) for simplicity and the number of allowed orientations $|\mathbf{O}|$ is 1 (see Appendix B for the 3D version). A newly added item introduces a series of empty spaces and new candidate placements are generated, e.g., the left-bottom corner of the empty space.

Learning Efficient online 3D-BPP on PCT

Method - Tree Representation

- Packing policy : π(Lt|Tt, nt)
- Input tuple : (Tt,nt) - Graph
- Encoder : GNN
  - PCT는 time step t에 따라 계속 증가하므로 그래프 구조가 동적 (Non-spectral)
  - GAT(non-spectral Graph Attention Network; Velickovic et al.,2018)를 채택
    - Graph structure에 대한 사전 정보가 필요 없음
- Bt,Lt,nt를 homogeneous node feature에 투영
  - node-wise Multi-Layer Perceptron(MLP) blocks

$$\hat{\mathbf{h}} = \{\phi_{\theta_B}(\mathbf{B}_t), \phi_{\theta_L}(\mathbf{L}_t), \phi_{\theta_n}(n_t)\}$$

MLP MLP MLP

GAT로 Node i를 embedding

scaled Dot-product attention

$$\mathrm{GAT}(\hat{h}_i) = W^O \sum_{j=1}^{N} softmax\left(\frac{(W^Q\hat{h}_i)^T W^K \hat{h}_j}{\sqrt{d_k}}\right) W^V \hat{h}_j$$

W : projection matrices N : feature Number(|Bt|+|Lt|+1)

$W^Q \in \mathbb{R}^{d_k \times d_h}$, $W^K \in \mathbb{R}^{d_k \times d_h}$, $W^V \in \mathbb{R}^{d_v \times d_h}$, and $W^O \in \mathbb{R}^{d_h \times d_v}$

$$\mathbf{h}' = \hat{\mathbf{h}} + \mathrm{GAT}(\hat{\mathbf{h}})$$ 중간변수

$$\mathbf{h} = \mathbf{h}' + \phi_{FF}(\mathbf{h}')$$ Final output

Node-wise feed-forward mlp

Learning Efficient online 3D-BPP on PCT

Method - Leaf Node Selection

- Input : Node features h
  - current item nt를 배치할 leaf node index를 결정
- Time step t에 따라 Tree가 계속 변화
  - Pointer mechanism 사용
  - Variable inputs에 대해 Context-based attention

$$\mathbf{h}: \bar{h} = \frac{1}{N} \sum_{i=1}^{N} h_i.$$

$$\pi_\theta(\mathbf{L}_t | \mathcal{T}_t, n_t) = softmax\left(c_{clip} \cdot \tanh\left(u_{\mathbf{L}}\right)\right)$$

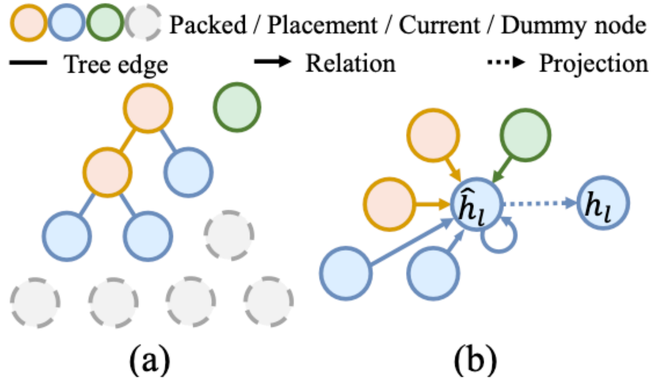


Figure 2: Batch calculation for PCT.

Learning Efficient online 3D-BPP on PCT

Method - Markov Decision Process Formulation

- 이제 time step t에서 tuple (Tt,nt)에만 의존하는 MDP로 공식화
- State : st = (Tt,nt)
  - Tt - Bt(internal nodes) + Lt(leaf nodes)
    - Bt : packed item size, coordinates
    - Lt : packable place size, coordinates
  - Nt : current item size
- Action : select된 leaf node의 index
  - Action space A = Lt size
- Reward : 성공적으로 packing 시 r = c x w, 아니면 0
  - C : 상수, w : weight(ex. Volume occupancy(부피))
- Training method : discounted reward 최대화하는 policy π(at|st)를 찾음
  - Model : ACKTR (actor-critic using Kronecker-factored trust region)

Learning Efficient online 3D-BPP on PCT

Experiments

- 여러 기존의 연구들을 구현해서 직접 비교
- 모든 알고리즘이 합리적인 기간내에 실행 되어야함
  - Discrete dataset (Zhao et al.,2021)
  - Continuous data set
- 지나치게 단순화된 시나리오를 피하기 위해 size setting
  - bin size = 10
  - item size < 5

Learning Efficient online 3D-BPP on PCT

Experiments - Performance

- leaf node expansion schems에 따른 성능
  - Corner Point(CP)
  - Extreme Point(EP)
  - Empty Maximal Space(EMS)
  - Event Point(EV)

- PCT + EMS/EV 가 가장 우수한 성능

Table 1: Performance comparisons. Uti. and Num. mean the average space utilization and the average number of packed items separately. Var. ($\times 10^{-3}$) is the variance of Uti. and Gap is w.r.t. the best Uti. across all methods. *Random* and *Random* & EV randomly pick placements from full coordinates and full EV nodes respectively. DBL, LSAH, MACS, and BR are heuristic methods proposed by Karabulut & Inceoglu (2004), Hu et al. (2017), Hu et al. (2020), and Zhao et al. (2021). Ha et al. (2017), LSAH, and BR are heuristics based on EMS.

	Method	*Setting* 1 Uti.	Var.	Num.	Gap	*Setting* 2 Uti.	Var.	Num.	Gap	*Setting* 3 Uti.	Var.	Num.	Gap
Heuristic	*Random*	36.7%	10.3	14.9	51.7%	38.6%	8.3	15.7	55.1%	36.8%	10.6	14.9	51.4%
	BR	49.0%	10.8	19.6	35.5%	56.7%	6.6	22.6	34.1%	48.9%	10.7	19.5	35.4%
	Ha et al.	52.1%	20.1	20.6	31.4%	59.9%	10.4	23.8	30.3%	51.9%	20.2	20.6	31.4%
	LSAH	52.5%	12.2	20.8	30.9%	65.0%	6.1	25.6	24.4%	52.4%	12.2	20.7	30.8%
	Wang & Hauser	57.6%	11.5	24.1	24.2%	66.1%	8.4	25.9	23.1%	56.5%	11.2	22.3	25.4%
	MACS	57.7%	10.5	22.6	24.1%	50.8%	8.8	20.1	40.9%	57.7%	10.6	22.6	23.8%
	DBL	60.5%	8.8	23.8	20.4%	70.6%	7.9	27.8	17.9%	60.5%	8.9	23.8	20.1%
Learning-based	Zhao et al.	70.9%	6.2	27.5	6.7%	70.3%	4.3	27.4	18.3%	59.6%	5.4	23.1	21.3%
	PCT & CP	69.4%	5.4	26.7	8.7%	81.8%	2.0	31.3	4.9%	69.5%	5.4	26.7	8.2%
	PCT & EP	71.9%	6.6	27.8	5.4%	78.1%	3.8	30.3	9.2%	72.2%	5.8	27.9	4.6%
	PCT & FC	72.4%	4.7	28.0	4.7%	76.9%	3.3	29.7	10.6%	69.8%	5.3	27.1	7.8%
	PCT & EMS	75.8%	4.4	29.3	0.3%	**86.0%**	**1.9**	**33.0**	**0.0%**	75.5%	4.7	29.2	0.3%
	PCT & EV	**76.0%**	**4.2**	**29.4**	**0.0%**	85.3%	2.1	32.8	0.8%	**75.7%**	**4.6**	**29.2**	**0.0%**
	PCT & EVF	75.7%	4.8	29.2	0.4%	80.5%	2.9	31.0	6.4%	73.5%	4.6	28.4	2.9%
	PCT & EV/GS	75.8%	4.7	29.2	0.3%	84.8%	2.1	32.6	1.4%	75.5%	4.8	29.1	0.3%
	Random & EV	45.7%	13.5	18.4	39.9%	51.0%	8.3	20.4	40.7%	45.1%	12.5	18.1	40.4%

Learning Efficient online 3D-BPP on PCT

Experiments - Performance

- PCT representation이 3D-BPP에 도움이 되는지 확인
- PointNet : space configuration node를 독립적으로 embedding
- Ptr-Net : tree 구조를 node sequence로 분해, serialized input으로 embedding
- Setting 2의 경우 internal node를 고려하지 않아도 성능이 충분

Table 2: A graph embedding for complete PCT helps the final performance.

Presentation	*Setting* 1 Uti.	Var.	Num.	Gap	*Setting* 2 Uti.	Var.	Num.	Gap	*Setting* 3 Uti.	Var.	Num.	Gap
PointNet	69.2%	6.7	26.9	8.9%	78.9%	3.2	30.5	7.5%	71.5%	5.3	27.7	5.5%
Ptr-Net	64.1%	10.0	25.1	15.7%	77.5%	4.1	30.1	9.1%	63.5%	7.9	24.8	16.1%
PCT ($\mathcal{T}/\mathbf{B}$)	70.9%	5.9	27.5	6.7%	84.1%	2.6	32.3	1.4%	70.6%	5.3	27.4	6.7%
PCT ($\mathcal{T}$)	**76.0%**	**4.2**	**29.4**	**0.0%**	**85.3%**	**2.1**	**32.8**	**0.0%**	**75.7%**	**4.6**	**29.2**	**0.0%**

Learning Efficient online 3D-BPP on PCT

Experiments - Performance

- Continuous Solution space에서도 잘 작동하는지?
- item들의 높이에 대한 다양성을 제어
- DRL, Heuristic과 비교

Table 3: Online 3D-BPP with continuous solution space.

	Method	*Setting* 1 Uti.	Var.	Num.	Gap	*Setting* 2 Uti.	Var.	Num.	Gap	*Setting* 3 Uti.	Var.	Num.	Gap
Heu.	BR	40.9%	7.4	16.1	37.5%	45.3%	5.2	17.8	31.7%	40.9%	7.3	16.1	38.6%
	Ha et al.	43.9%	14.2	17.2	32.9%	46.1%	6.8	18.1	30.5%	43.9%	14.2	17.2	34.1%
	LSAH	48.3%	12.1	18.7	26.1%	58.7%	4.6	22.8	11.5%	48.4%	12.2	18.8	27.3%
DRL	GD	5.6%	–	2.2	91.4%	7.5%	–	2.9	88.7%	5.2%	–	2.1	92.2%
	PCT & EMS	65.3%	4.4	24.9	0.2%	**66.3%**	**2.3**	**27.0**	**0.0%**	**66.6%**	**3.3**	**25.3**	**0.0%**
	PCT & EV	**65.4%**	**3.3**	**25.0**	**0.0%**	65.0%	2.6	26.4	2.0%	65.8%	3.6	25.1	2.7%

Learning Efficient online 3D-BPP on PCT

Experiments - Performance

- Learning based method의 고질적인 문제 : Generalization ability
- sampling하는 item의 size distribution에 따른 성능 test

Table 4: Generalization performance on different kinds of item sampling distributions.

Test Distribution	Method	*Setting* 1 Uti.	*Setting* 1 Var.	*Setting* 1 Num.	*Setting* 2 Uti.	*Setting* 2 Var.	*Setting* 2 Num.	*Setting* 3 Uti.	*Setting* 3 Var.	*Setting* 3 Num.
$s^d \sim U(0.1, 0.5)$	LSAH	48.3%	12.1	18.7	58.7%	4.6	22.8	48.4%	12.2	18.8
	PCT & EMS	65.3%	4.4	24.9	**66.3%**	**2.3**	**27.0**	**66.6%**	**3.3**	**25.3**
	PCT & EV	**65.4%**	**3.3**	**25.0**	65.0%	2.6	26.4	65.8%	3.6	25.1
$s^d \sim N(0.3, 0.1^2)$	LSAH	49.2%	11.1	18.9	60.0%	4.1	22.9	49.2%	11.0	18.9
	PCT & EMS	**66.1%**	3.6	**25.1**	**64.3%**	3.5	**25.6**	**66.4%**	3.0	**25.2**
	PCT & EV	65.1%	**2.8**	24.7	63.7%	**2.6**	25.3	66.2%	**2.9**	25.1
$s^d \sim N(0.1, 0.2^2)$	LSAH	52.4%	8.9	30.3	62.9%	**2.4**	44.3	52.3%	8.9	30.2
	PCT & EMS	**68.5%**	**2.5**	**39.0**	**66.4%**	3.0	**49.7**	**69.2%**	2.5	**39.4**
	PCT & EV	66.5%	2.7	38.0	64.9%	2.7	48.3	67.4%	**2.4**	38.5
$s^d \sim N(0.5, 0.2^2)$	LSAH	47.3%	12.6	13.0	56.0%	5.5	12.9	47.3%	12.6	13.0
	PCT & EMS	63.5%	5.0	17.3	**64.5%**	**2.8**	**15.4**	**65.2%**	3.8	**17.7**
	PCT & EV	**65.1%**	**3.3**	**17.7**	64.5%	2.8	15.3	65.1%	**3.7**	17.7

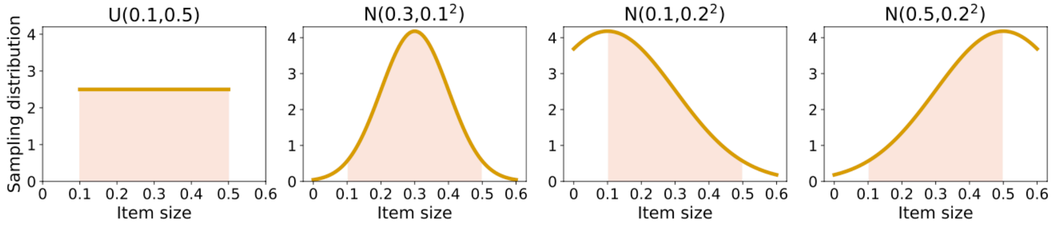


Figure 3: The probability distribution for sampling item sizes. The area of the colored zone is normalized to 1.

Learning Efficient online 3D-BPP on PCT

Experiments - Performance

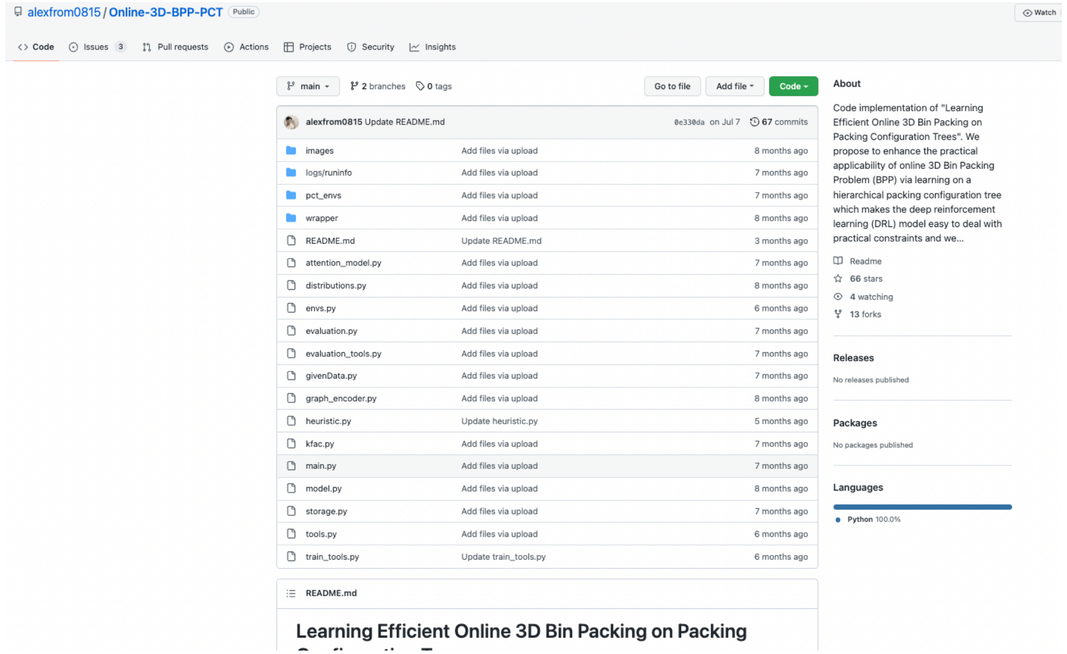
- 복잡한 제약조건이 있는 3D-BPP에서 잘 활용가능하다는 것을 증명
- 추가로 practical Constraints가 있는 online 3D-BPP 적용 2가지 데모
  - 3D-BPP with Isle Friendliness : 같은 범주에 속하는 item을 최대한 조밀하게 포장
  - 3D-BPP with Load Balancing : 포장된 품목이 빈 내에서 균일한 질량 분포를 가져야 함을 지시

	Method	*Setting* 1			*Setting* 2			*Setting* 3		
		Uti.	Num.	Obj.	Uti.	Num.	Obj.	Uti.	Num.	Obj.
Isle Friendliness	Zhao et al.	58.3%	22.5	4.58	64.2%	24.8	4.67	59.0%	22.8	4.60
	PCT & EV	72.1%	29.0	4.44	85.2%	32.8	2.69	74.6%	28.8	4.45
Load Balancing	Zhao et al.	60.3%	23.3	88.0		–		61.1%	23.7	30.1
	PCT & EV	73.7%	28.6	40.5		–		74.0%	28.7	20.9

Learning Efficient online 3D-BPP on PCT

Discussion

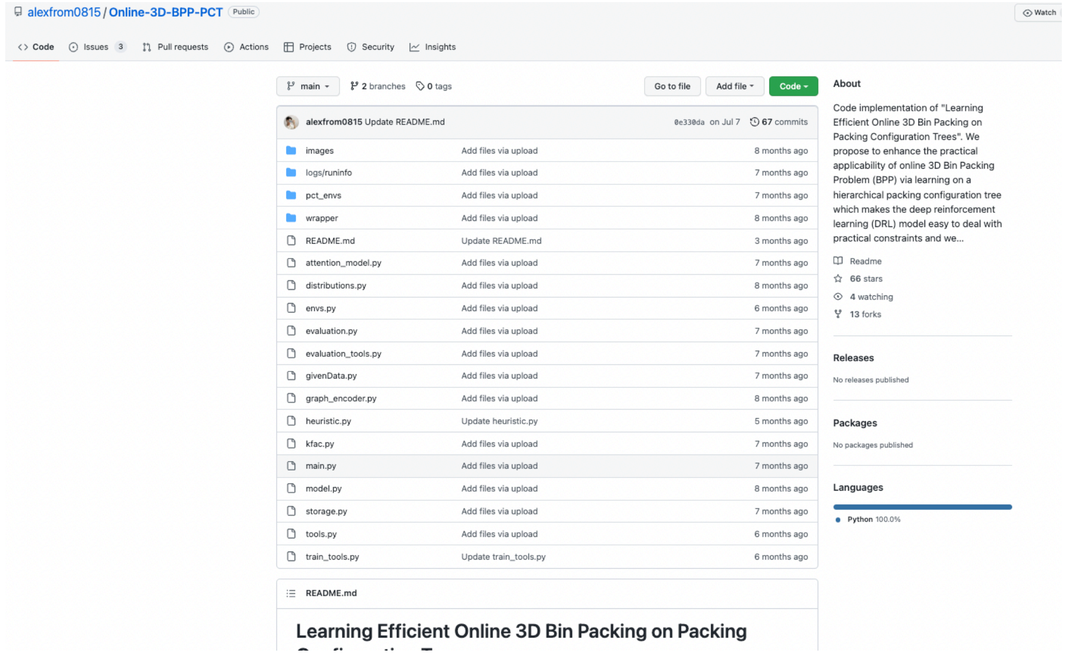
- 논문리뷰 실전편 with Code
  - Implementaiton Details
    - DRL
    - PCT Length
  - Leaf node Expansion Schemes
  - More Results
    - Running costs
    - Visualized result
    - Scalability



Learning Efficient online 3D-BPP on PCT

Discussion

- 논문리뷰 실전편 with Code
  - Implementaiton Details
    - DRL
    - PCT Length
  - Leaf node Expansion Schemes
  - More Results
    - Running costs
    - Visualized result
    - Scalability



감사합니다.