"How can we efficiently leverage heuristics in reinforcement learning (RL)?"

H2RL : Transferring Heuristic Knowledge to a Reinforcement Learning Agent for Solving the Online 3D Bin Packing Problem

Minji Kim[1†], Ganghun Lee[2†], Minsu Lee[2*], and Byoung-Tak Zhang[1,2*]

Dept. of Computer Science and Engineering[1], Seoul National University
Artificial Intelligence Institute (AIIS)[2], Seoul National University
{mjkim, khlee, mslee, btzhang}@bi.snu.ac.kr

Background

- Reinforcement learning (RL) has become a prominent approach for training intelligent agents to acquire optimal behaviors.
- RL-based methods encounter various challenges, including slow convergence, sample inefficiency, and difficulties in generalization.

Research Questions

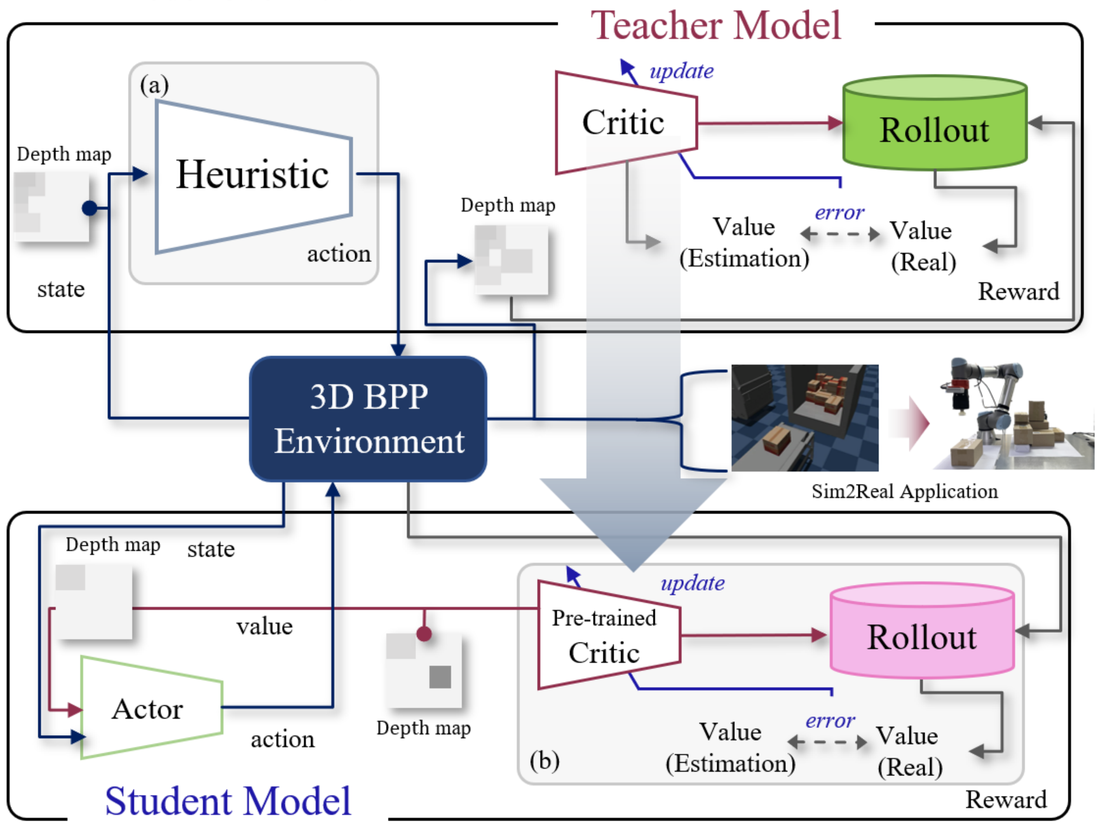
- How can we effectively utilize heuristics in RL for complex environments?
- Can a proposed method surpass the performance of heuristics and traditional RL?
- Will the proposed method perform significantly in real-world robot experiments?

Key Ideas in This Work

- Using knowledge distillation, the teacher critic of reinforcement learning is trained by heuristics(H2RL).
- We compare the effectiveness of our approach in solving a highly uncertain 3D Bin Packing Problem (3D-BPP) with random sequences to heuristics and conventional reinforcement learning methods.

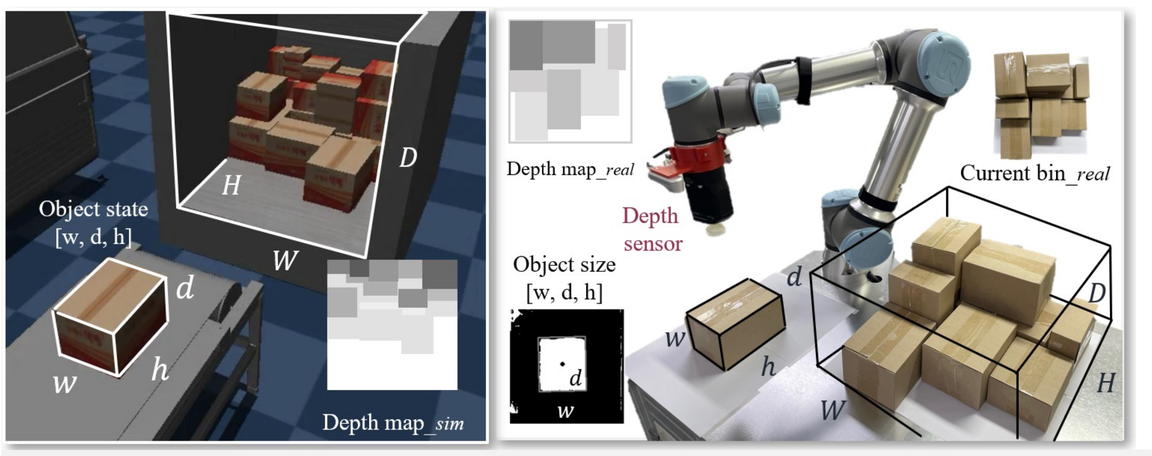
Methods and Materials

- **H2RL Model Overview**



- H2RL is based on knowledge distillation, a two-stage learning process that consists of a teacher model and a student model to extract knowledge from deeper, more complex models.
- First, we trained the teacher critic to predict state-value (cumulative reward) following the action of the heuristic.
- Second, we trained the student agent to interact with the environment by leveraging the pre-trained teacher critic.

Experiment Details



An implementation of a simulation and real-world experimental setting.

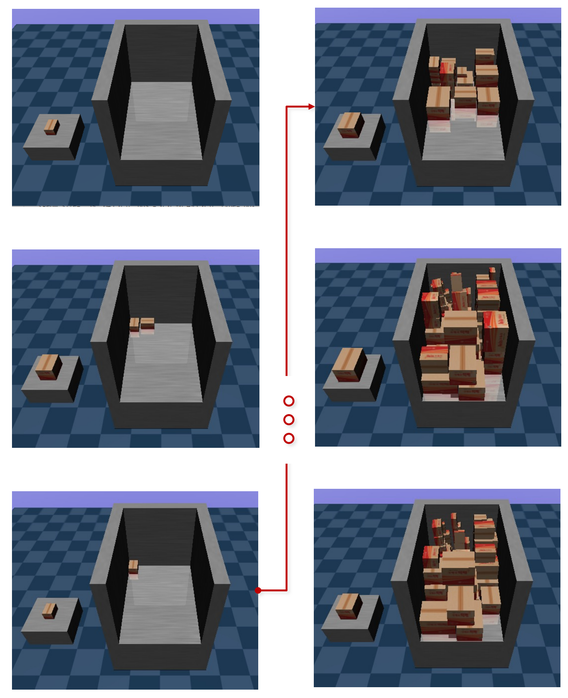
- We designed a simulation for the heuristic by MuJoCo and implemented a robot experiment using UR5e.

Designed Heurisitcs for 3D-BPP

Algorithm 1: Heuristic Algorithm for Dynamic Bin Packing

Input: $W, H, D, \mathbf{0}, \omega, r_g, r_h, z_g, \rho$
generate grid positions $\mathcal{G}$ with window size ω;
$t \leftarrow 0$;
while *not full* **do**
  capture depth map $\mathcal{M}_t(W, H, \mathbf{0})$;
  get next object of size w, h, d;
  define list of global position candidates $\mathcal{C}$;
  // consider 6 orientations
  for $obj \in \{(w,h,d),(w,d,h),(h,w,d),(h,d,w),(d,w,h),(d,h,w)\}$ **do**
    define list of local position candidates $\mathcal{C}_{obj}$;
    // apply sliding window
    for $\mathbf{p} \in \mathcal{G}$ **do**
      get depth map for target region $\mathcal{M}_t(w, h, \mathbf{p})$;
      // find z limit violence of target region
      $z_{max} \leftarrow D - (min(\mathcal{M}_t(w, h, \mathbf{p})) - r_h)$;
      $overz \leftarrow z_{max} + d > D$;
      // find flatness of target region
      quantize $\mathcal{M}_t(w, h, \mathbf{p})$ by z_g;
      $m \leftarrow mode(\mathcal{M}_t(w, h, \mathbf{p}))$;
      $flat \leftarrow ratio(m, \mathcal{M}_t(w, h, \mathbf{p})) \geq \rho$;
      // find if the mode value expresses highest z
      $highest \leftarrow m$ equal to $min(\mathcal{M}_t(w, h, \mathbf{p}))$;
      // add acceptable position cadidate
      $accept \leftarrow$ not $overz$ and $flat$ and $highest$;
      if *accept* **then**
        simulate next depth map $\mathcal{M}'_{t+1}(W, H, \mathbf{0})$;
        $s_{\mathbf{p}} \leftarrow std(\mathcal{M}'_{t+1}(W, H, \mathbf{0}))$;
        add $(\mathbf{p}, s_{\mathbf{p}})$ to $\mathcal{C}_{obj}$;
      end
    end
    $(\mathbf{p}^*_{obj}, s^*_{\mathbf{p}_{obj}}) \leftarrow (\mathbf{p}, s_{\mathbf{p}}) \in \mathcal{C}_{obj}$ which has minimum $distance(\mathbf{p}, \mathbf{0}))$;
    add $(\mathbf{p}^*_{obj}, s^*_{\mathbf{p}_{obj}}, obj)$ to $\mathcal{C}$;
  end
  $(\mathbf{p}^*, obj^*) \leftarrow (\mathbf{p}, obj)$ in $(\mathbf{p}, s_p, obj) \in \mathcal{C}$ which has minimum $s_{\mathbf{p}}$;
  if $\mathcal{C}$ *is empty* **then**
    $full \leftarrow True$
  else
    place object at position $\mathbf{p}^*$ with orientation obj^*;
    $t \leftarrow t + 1$;
  end
end

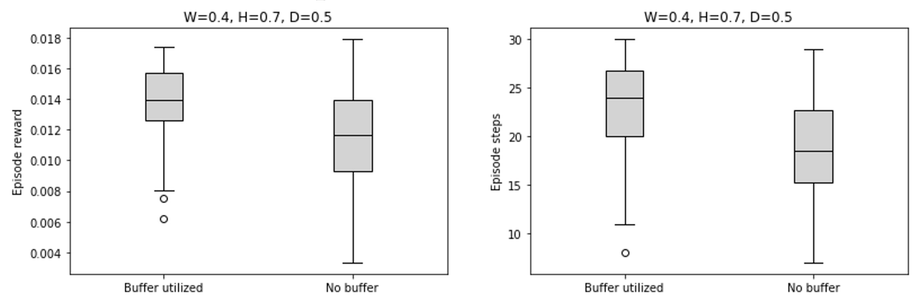
An example of a devised heuristic algorithm applied in a virtual environment for the 3D-BPP



- Novel heuristics were devised to effectively solve the by considering practical conditions and constraints.
- The placement point is chosen from the depth map grid, allowing the agent to adjust the object orientation (0° or 90° around the z-axis), with the z-component determined by the highest point on the support surface.

Experiment Results

- **Performance verification experiments with teacher critic**



- The buffer utilized environment is a scenario where objects in the buffer are selected and loaded based on their predicted values, determined by the size of the objects, based on teacher critic.
- Combining simple heuristics with the teacher critic and utilizing buffers allowed for assessing the effectiveness of value estimation, as seen in the improvement of episode rewards(cumulative volume) and steps(cumulative number of items loaded).

	Heuristic	Heuristic (Noisy)	RL	H2RL (Ours)
Env-1	48.60%	33.20%	50.67%	**54.46%**
Env-2	61.97%	33.89%	43.39%	**65.35%**
Env-2-r	58.93%	47.58%	27.75%	**59.10%**

TABLE I: Comparison of space utilization (SU) in different methods.

- We compared the SU of three methods: the heuristic we designed, the heuristic with added noise used during training, and the RL method(PPO).
- H2RL found a more optimized solution than heuristic in both settings and also outperformed RL.

	Buffer-1	Buffer-2	Buffer-3	Buffer-4	Buffer-5
Env-1	57.58%	59.31%	61.77%	62.57%	**63.69%**
Env-2	68.73%	70.94%	72.55%	73.37%	**73.77%**

TABLE II: Comparison of space utilization (SU) by H2RL in different number of available buffer slots.

- SU increased proportionally to the number of slots.
- This result reveals the essential ability of critics to estimate the impacts of various object sizes, enhancing SU through buffer control.