大阪大学 OSAKA UNIVERSITY

H.U. Group Research Institute

Download a powerpoint version of this material (you can see gif animations) at: https://t.ly/IvPtg

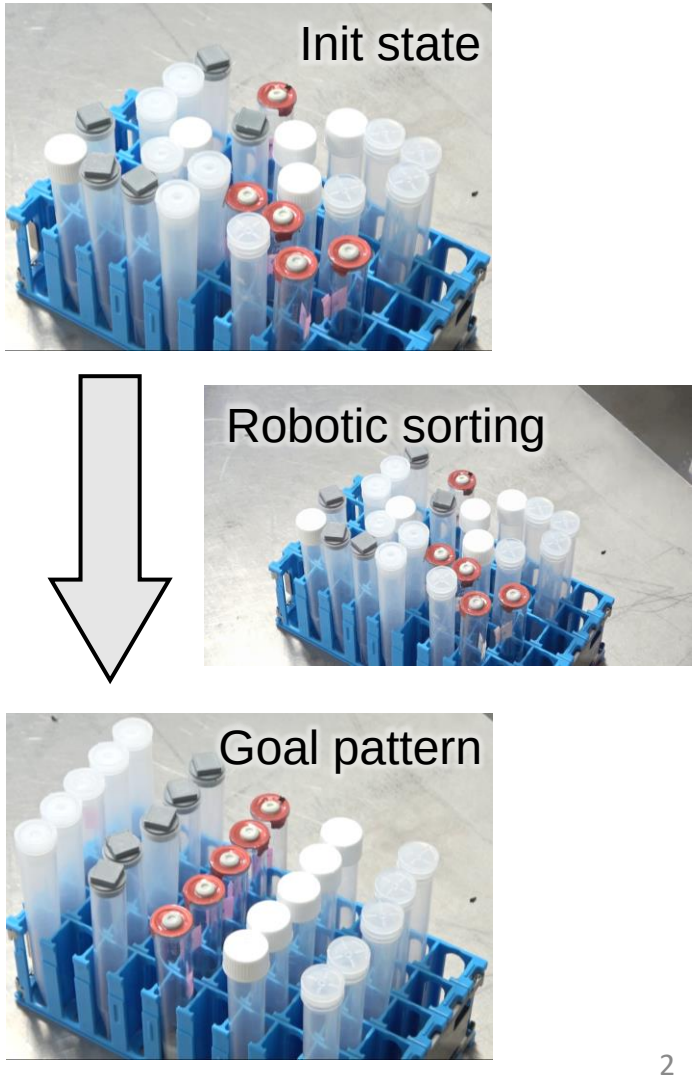
Robotic Test Tube Rearrangement Using Combined Reinforcement Learning and Motion Planning in a Closed Loop

Hao Chen[1], Yu Tang[2], Weiwei Wan[2], Masaki Matsushita[3], Jun Takahashi[3], Takeyuki Kotaka[3], Kensuke Harada[2]

[1] Fujian Agriculture and Forestry University, China
[2] Graduate School of Engineering Science, Osaka University, Japan
[2] H.U. Group Research Inst. G.K., Japan

Brief Introduction

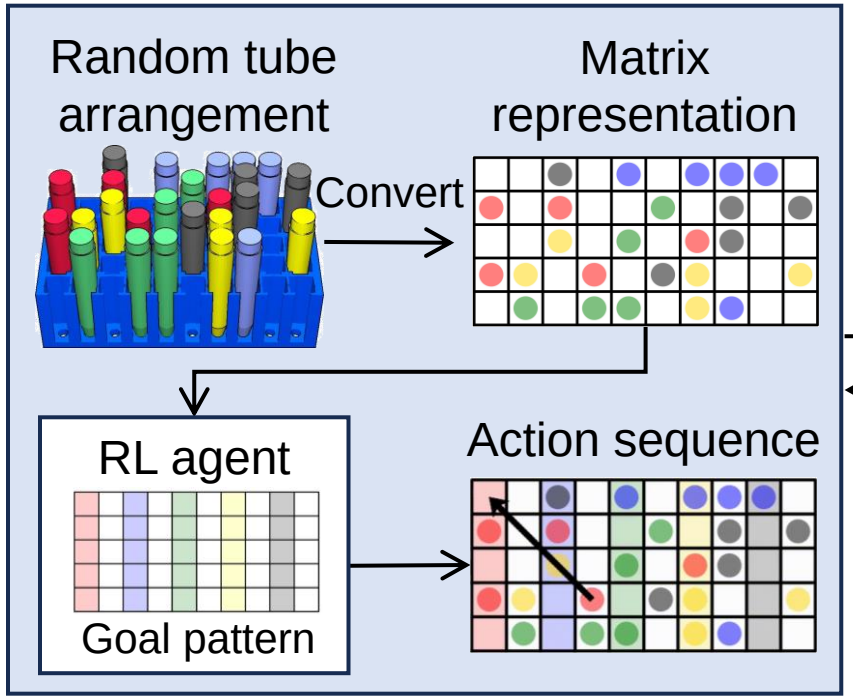
- Purpose:
  - Introduce a combined **task-level Reinforcement Learning** (RL) and motion planning framework to sort in-rack test tubes.
- Contributions:
  - Develop an **A* post-processing** technique to expediate RL training.
  - Close the loop of the task and motion level planner by maintaining a **condition set** for each rack slot.



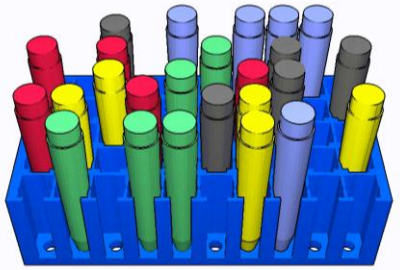
Overview

大阪大学 OSAKA UNIVERSITY
WRS システム
H.U.グループ中央研究所

Task level



Generate action sequences using RL



Action sequence

Replanning
(planning failures)
Invalidate
Condition sets

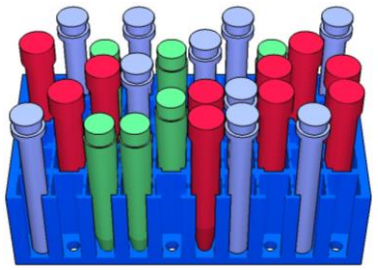
Motion level

Plan pick-and-place motions sequentially

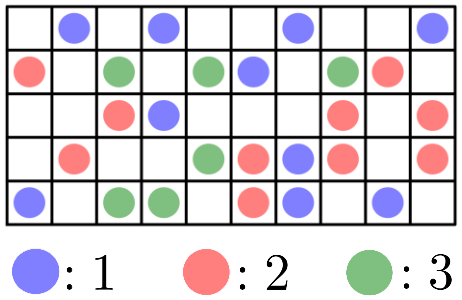
Definitions of the Constrained Markov Decision Process (CMDP)

- **States**:

Test Tube Arrangement

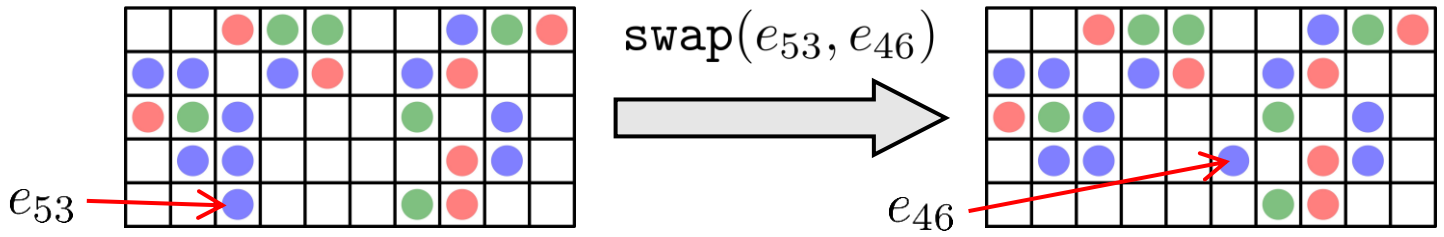


State (Matrix Representation)



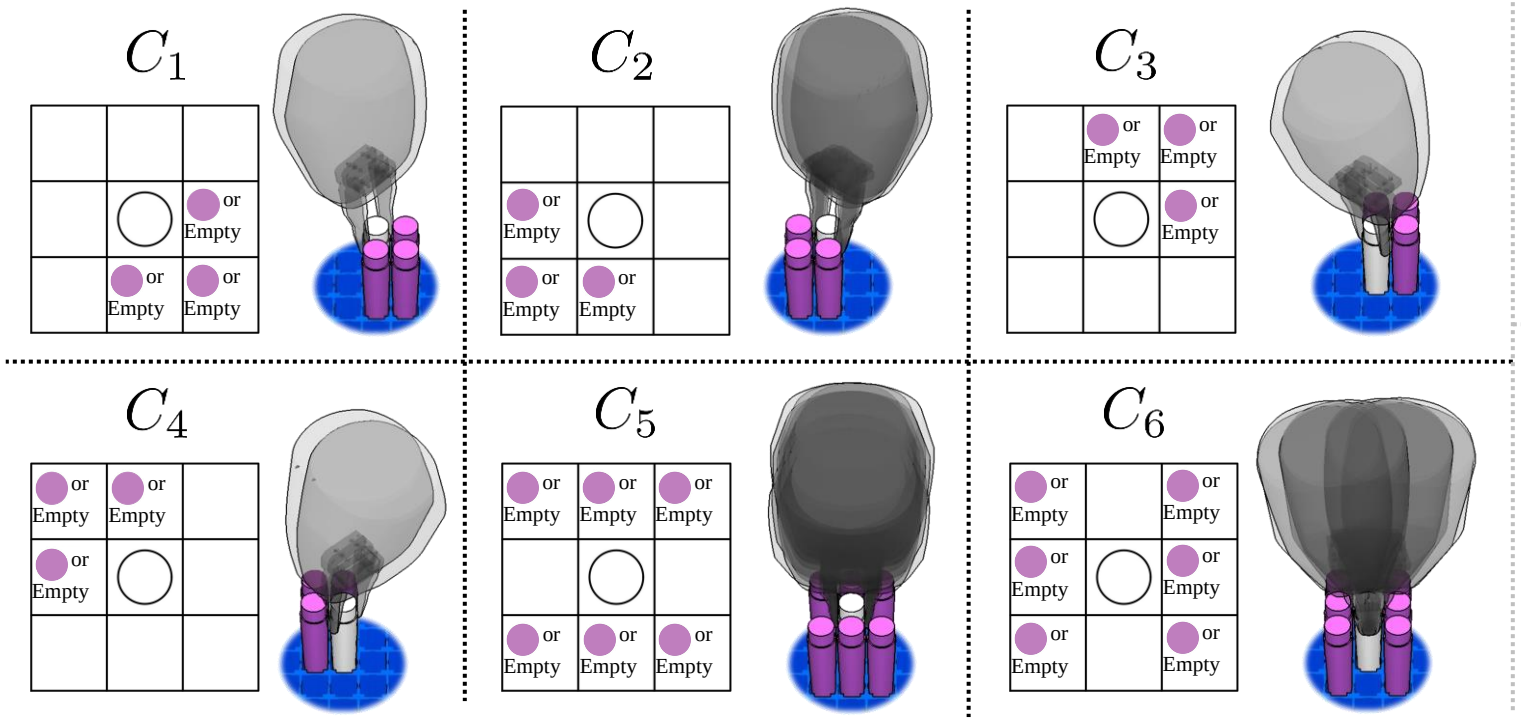
$$=\begin{bmatrix} 0 & 2 & 0 & 2 & 0 & 0 & 2 & 0 & 0 & 2 \\ 1 & 0 & 3 & 0 & 3 & 2 & 0 & 3 & 1 & 0 \\ 0 & 0 & 1 & 2 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 3 & 1 & 2 & 1 & 0 & 1 \\ 2 & 0 & 3 & 3 & 0 & 1 & 2 & 0 & 2 & 0 \end{bmatrix}$$

- **Actions**: "Swap" between two slots
  - Pick slot: The slot with a test tube
  - Place slot: The empty slot

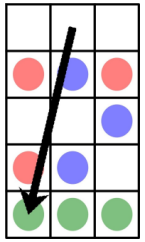


Definitions of the Constrained Markov Decision Process (CMDP)

- **Constraints**: A valid action must satisfy one of the following six conditions to ensure enough space to pose robot fingers

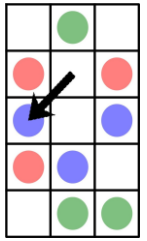


- Actions: $\mathrm{swap}(e_{12}, e_{51})$
- Slot (1,2): Satisfy C_5
- Slot (5,1): Satisfy C_3



➡ Valid Action ✓

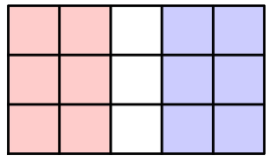
- Actions: $\mathrm{swap}(e_{22}, e_{31})$
- Slot (2,2): Satisfy $\emptyset$ ✗
- Slot (3,1): Satisfy C_5

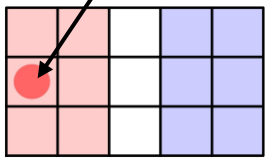


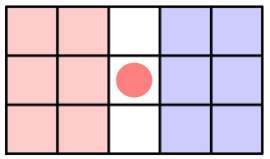
➡ Invalid Action ✗

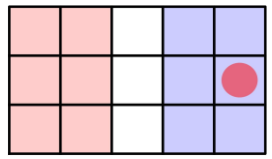
Definitions of the Constrained Markov Decision Process (CMDP)

- **Reward Design**: For the red test tube:

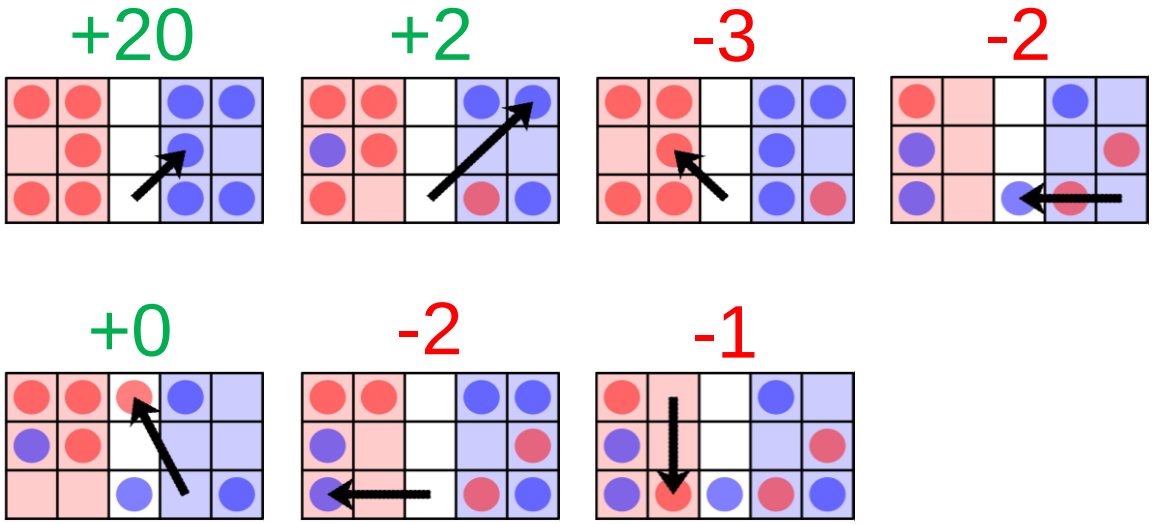

Goal pattern


At *correct-goal slot*


At *non-goal slot*

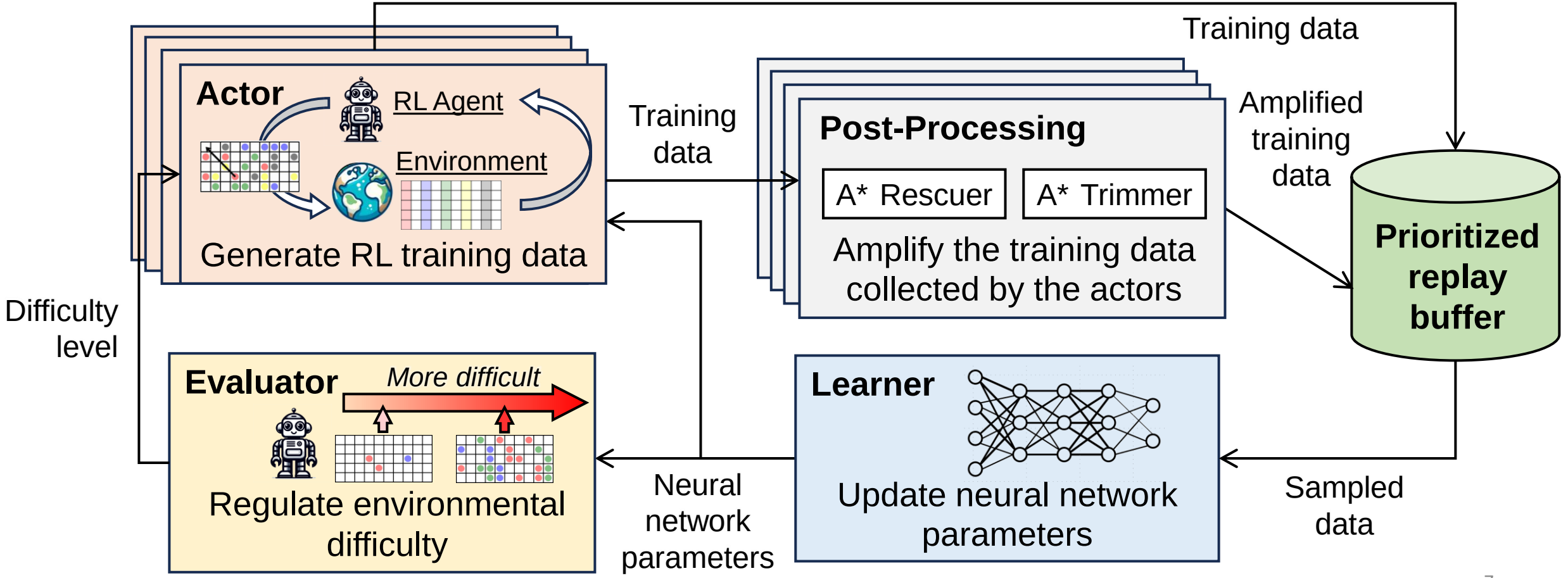

At *incorrect-goal slot*

Case	Rwd.
Goal achievement	+20
Reach deadlock	−20
Incorrect- / Non-Goal Slot → Correct-Goal Slot	
- Not blocking *Correct-Goal Slots*	+2
- Blocking *Correct-Goal Slots*	−3
Correct-Goal Slot → Incorrect- / Non-Goal Slot	−2
Incorrect-Goal Slot → Non-Goal Slot	+0
Non-Goal Slot → Incorrect-Goal Slot	−2
Indifferent movements	−1



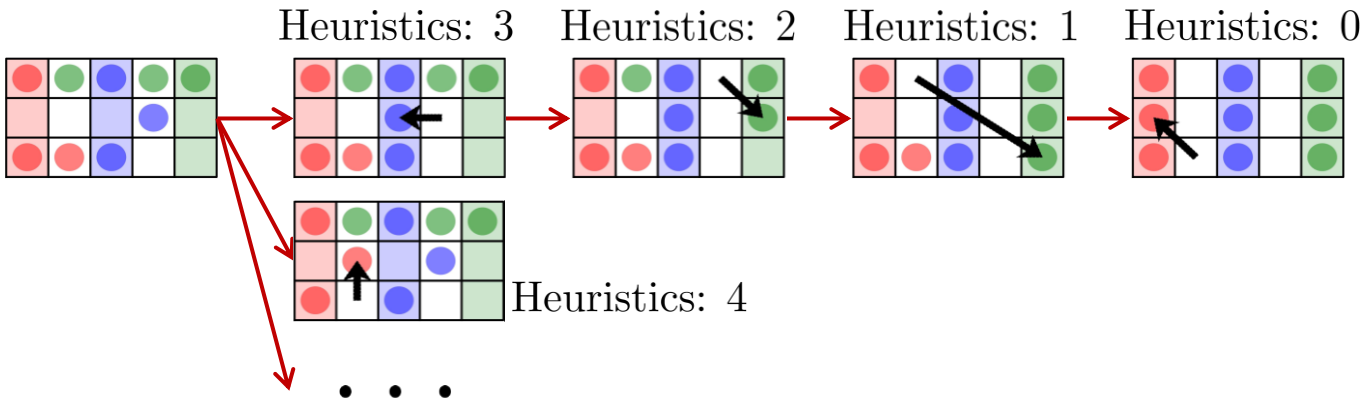
RL Training Process

- Train RL agents with a distributed Q-learning structure



A* Post-Processing

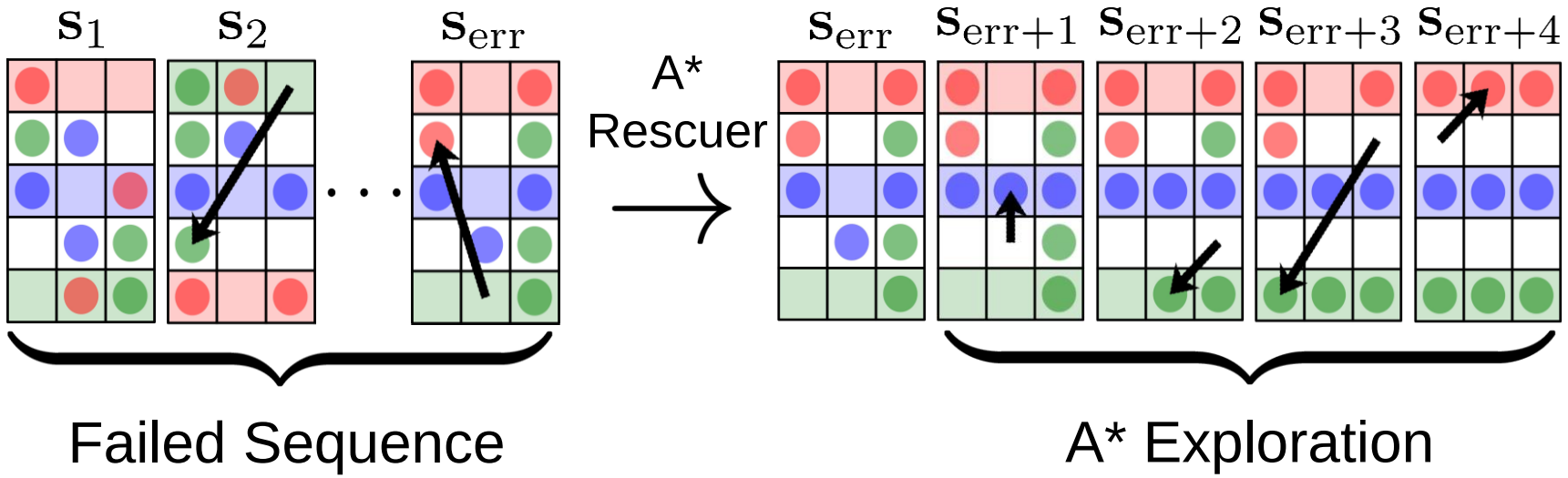
- Improve the efficiency based on A* search (Wan et al. (2022))
  - **Heuristics in A*:** Number of tubes outside their goal pattern



- Two ways to use A* search for RL acceleration:
  - A* Rescuer
  - A* Trimmer

A* Post-Processing

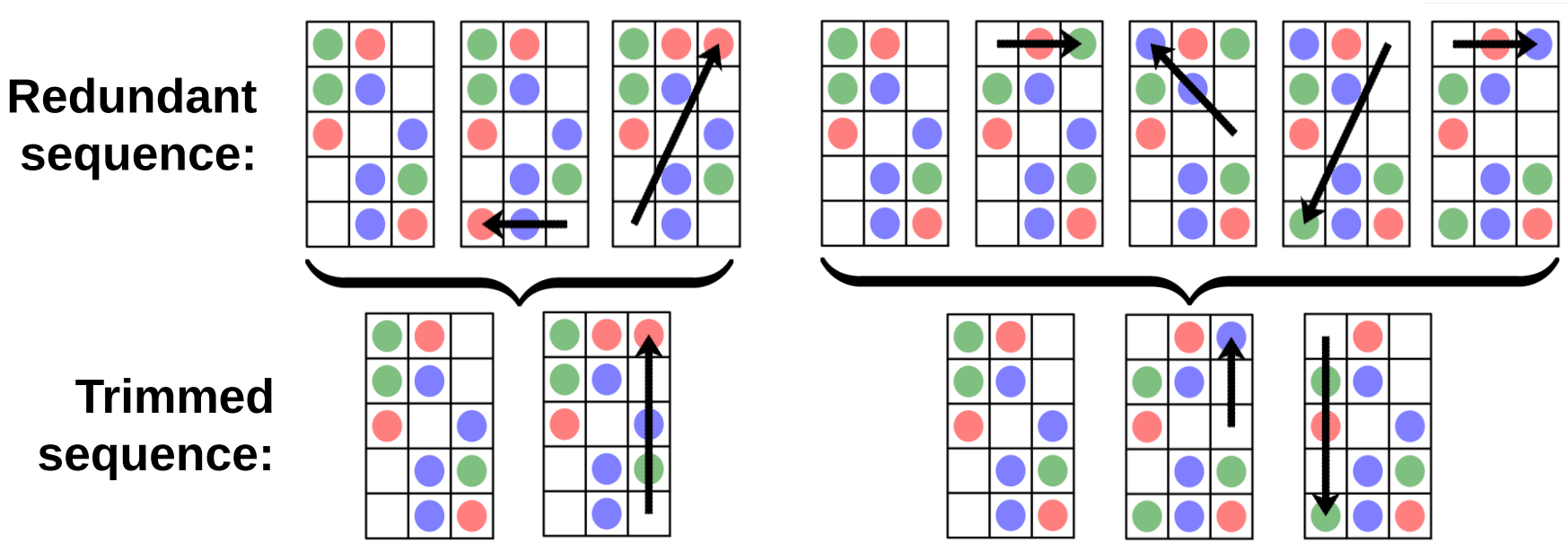
- **A* Rescuer:** Attempts to salvage failed sequences (interrupted by the horizon limit).



Effect: Save incomplete yet valuable data for training

A* Post-Processing

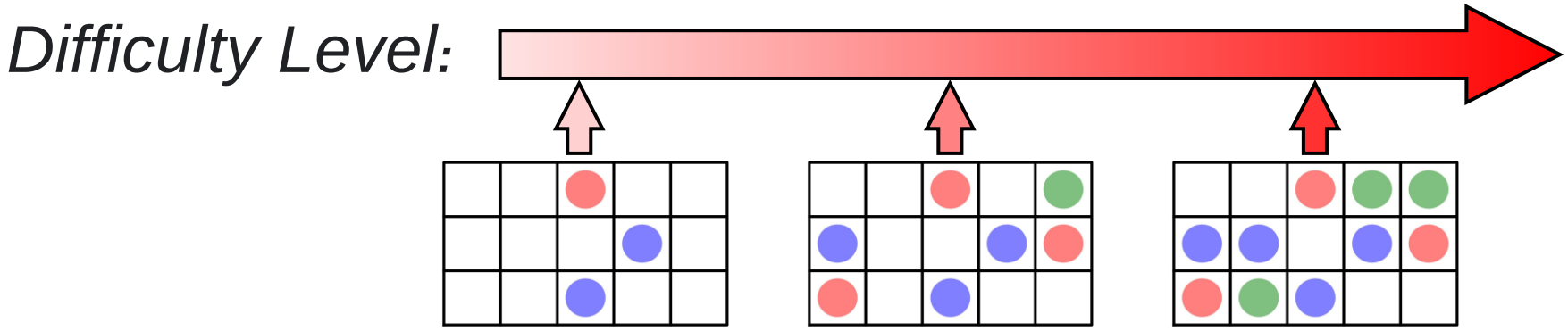
- **A* Trimmer:** Reduce redundancy in action sequences



Effect: Help focus exploration on actions that are more contributory to the goal

Other Training Strategies

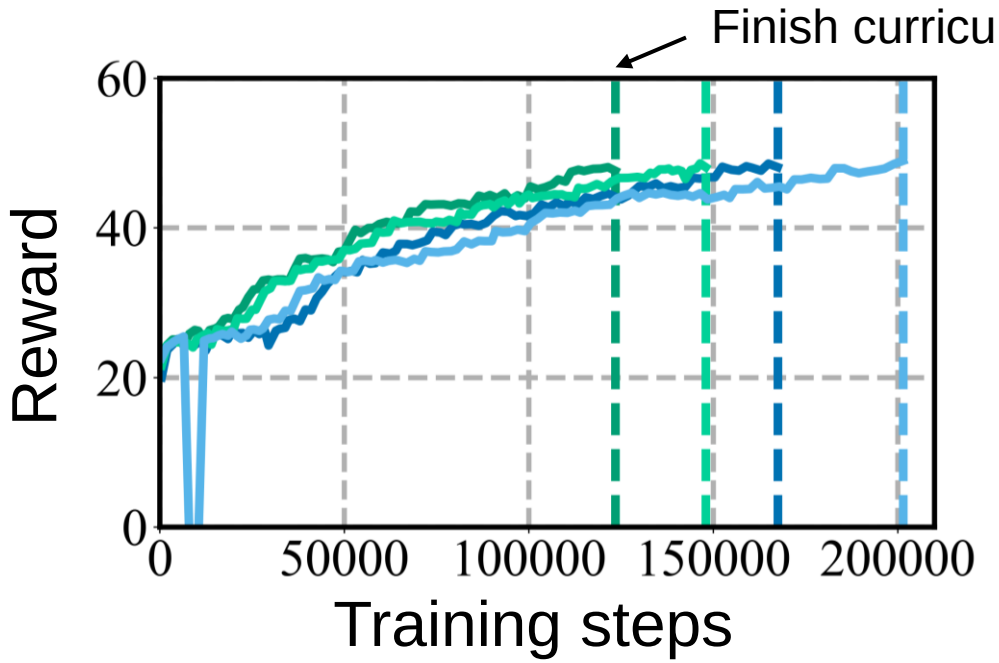
- **Curriculum Learning:** Progressively elevate the difficulty of the environment used for exploration and data collection.



- **Tabu Search Exploration:** Maintain a record of the explored states, avoiding actions that lead to previously visited state.

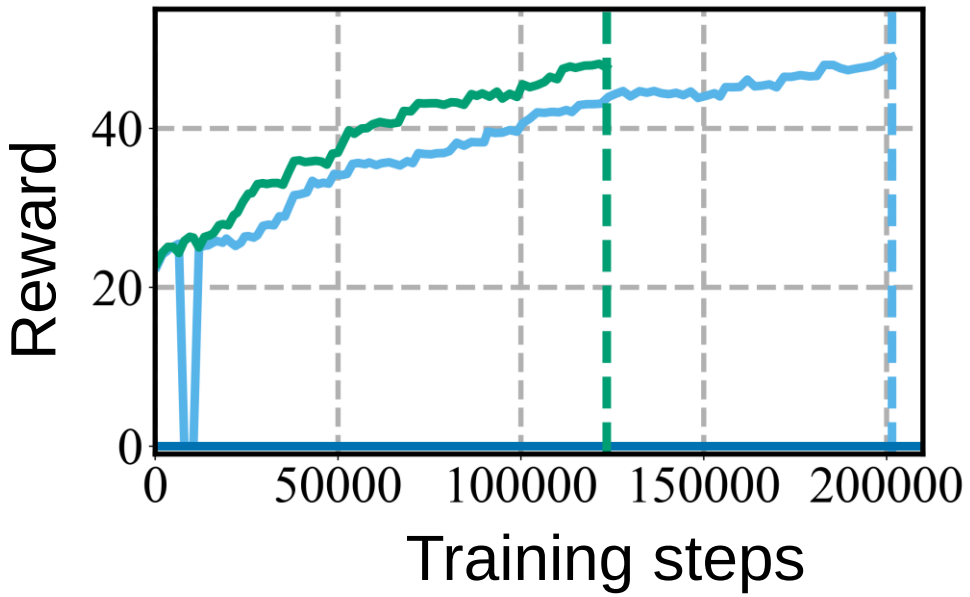
⇒ Avoid self-repeatition

Performance



RL Training
RL Training + A* Rescuer
RL Training + A* Trimmer
RL Training + A* Rescuer + A* Trimmer

A* post-processing can expediate RL training



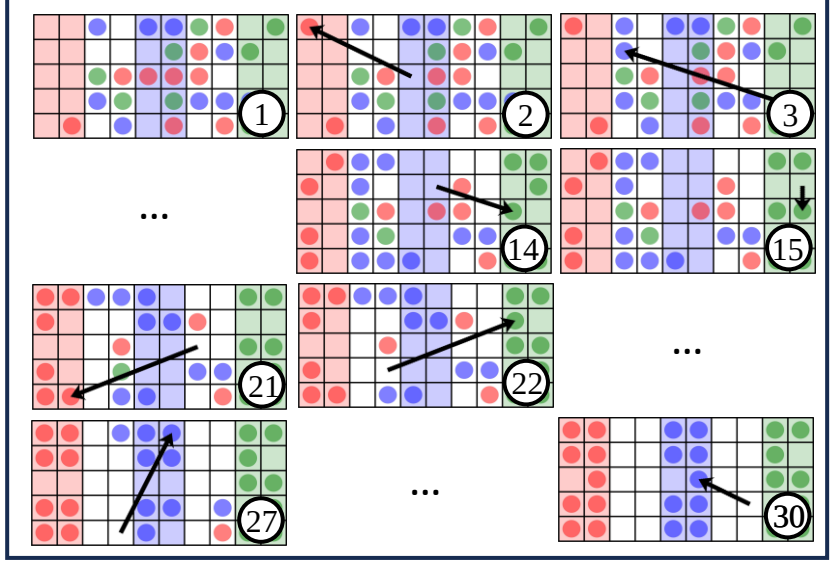
Proposed – Curriculum Learning
Proposed – A* Post-Processing
Proposed

Curriculum learning can stabilize and guide RL training

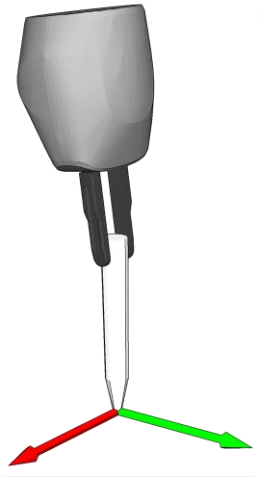
Motion Planning

大阪大学 OSAKA UNIVERSITY
WRS システム
H.U.グループ中央研究所

Task-Level Results



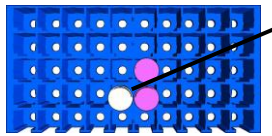
Preparation: Grasp annotation



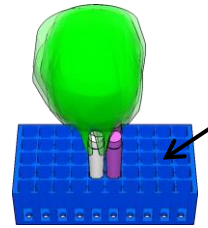
Pre-defined grasp poses

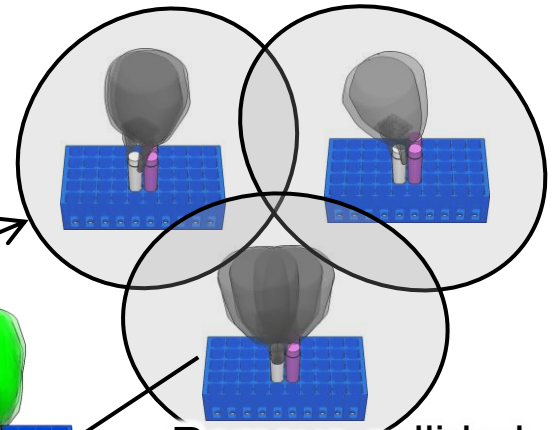
1. Canddiate grasp poses at a slot

Transform the annotated grasps to pick/place slots



Candidate grasp poses

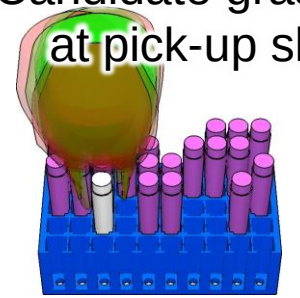




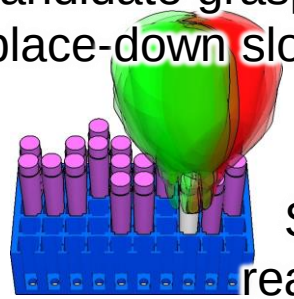
Remove collided grasp poses

2. Shared Grasp Poses

Candidate grasp at pick-up slot

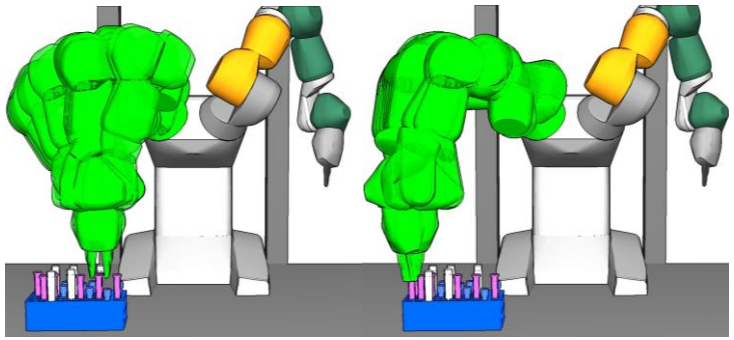


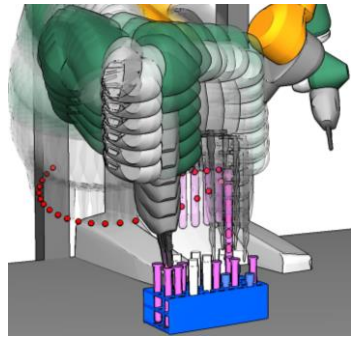
Candidate grasp at place-down slot



Shared grasp poses reachable by the robot

3. Generate pick-and-place motion

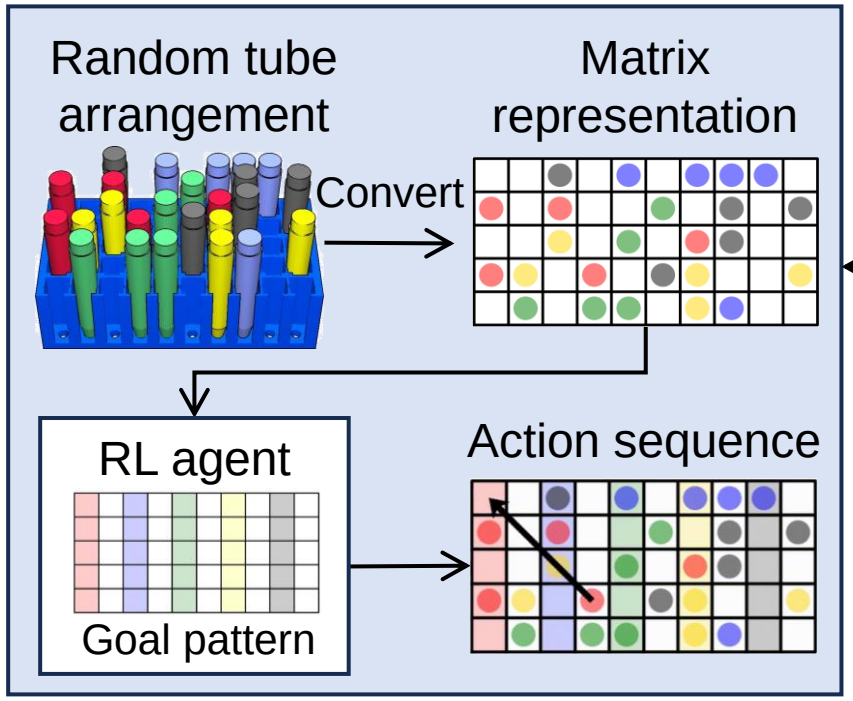




Close the Task and Motion Planning Loop by maintaining individual Condition Sets

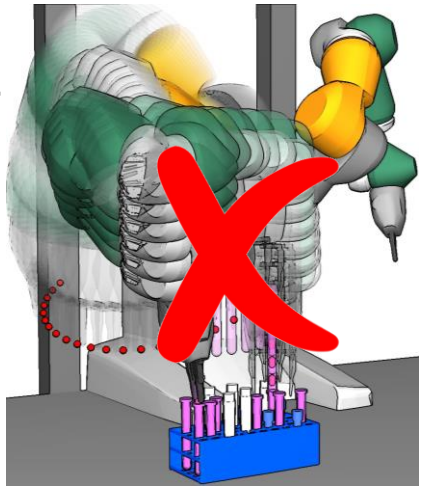
大阪大学 OSAKA UNIVERSITY
WRS システム
H.U.グループ中央研究所

Task level



Trigger Replanning

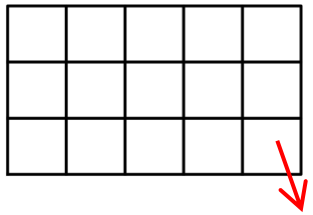
Motion planning may fail to find successful solutions



- Unreachable grasp poses
- Obstruction

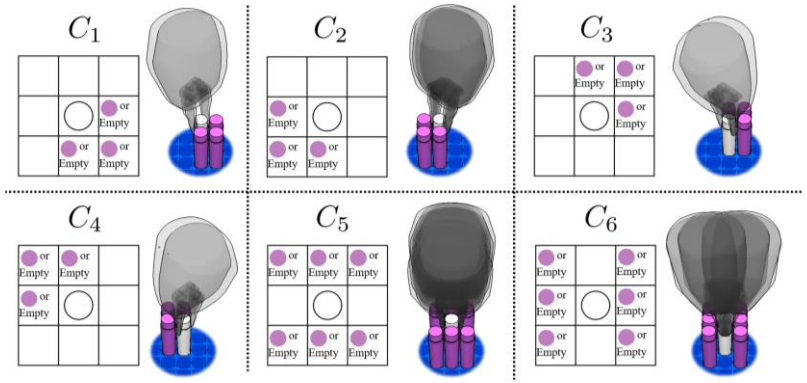
Close the Task and Motion Planning Loop by maintaining individual Condition Sets

- **Condition Set**

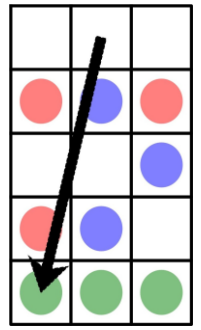


Each slot has a condition set
(Possible conditions to satisfy)

$$\zeta_{35} = \{C_1, C_2, C_3, C_4, C_5, C_6\}$$



- Check valid actions based on the condition sets
  - Actions: $\mathrm{swap}(e_{12}, e_{51})$
  - Slot (1,2): Satisfy C_5
  - Slot (5,1): Satisfy C_3



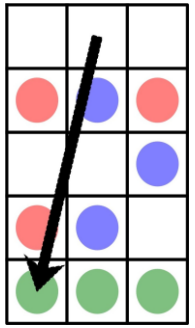
$(C_5 \in \zeta_{12}) \wedge (C_3 \in \zeta_{51})$ ⇒ Valid Action ✓

$(C_5 \notin \zeta_{12}) \vee (C_3 \notin \zeta_{51})$ ⇒ Invalid Action ✗

Close the Task and Motion Planning Loop by maintaining individual Condition Sets

- When motion planning fails ….

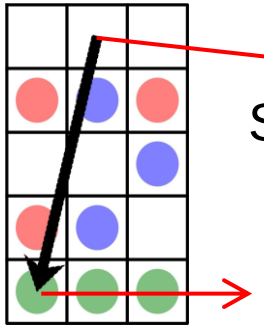
  - If the motion was obstructed by external obstacles:



Obstruction at a pick-up slot ➡ $\zeta_{12} = \emptyset$

Obstruction at a place-down slot ➡ $\zeta_{51} = \emptyset$

  - If the planner failed to find reachable grasp poses:



Unreachable at a pick-up slot:

$$\zeta_{12} = \zeta_{12} \setminus \{C_5\}$$

Unreachable at a place-down slot:

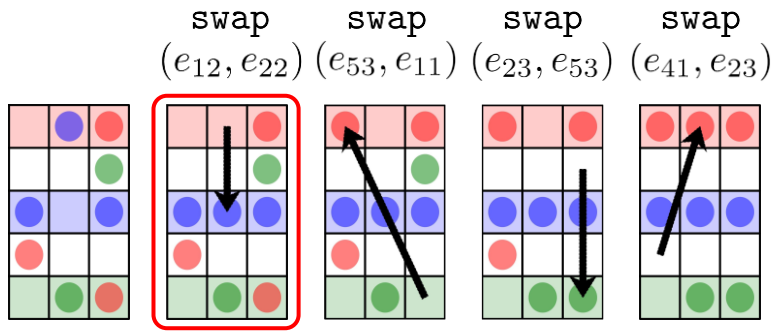
$$\zeta_{51} = \zeta_{51} \setminus \{C_3\}$$

Close the Task and Motion Planning Loop by maintaining individual Condition Sets

大阪大学 OSAKA UNIVERSITY　WRS システム　H.U.グループ中央研究所

Failure happened when performing motion planning based on task-level sequence

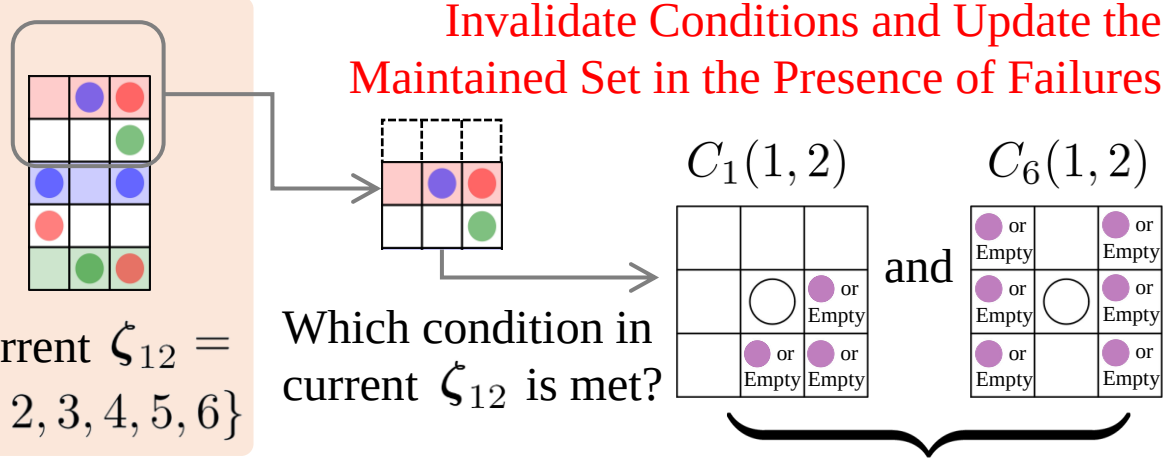
① Invalidate maintained conditions



Motion planning failure for $\texttt{swap}(e_{12}, e_{22})$

Close the loop by invalidating conditions

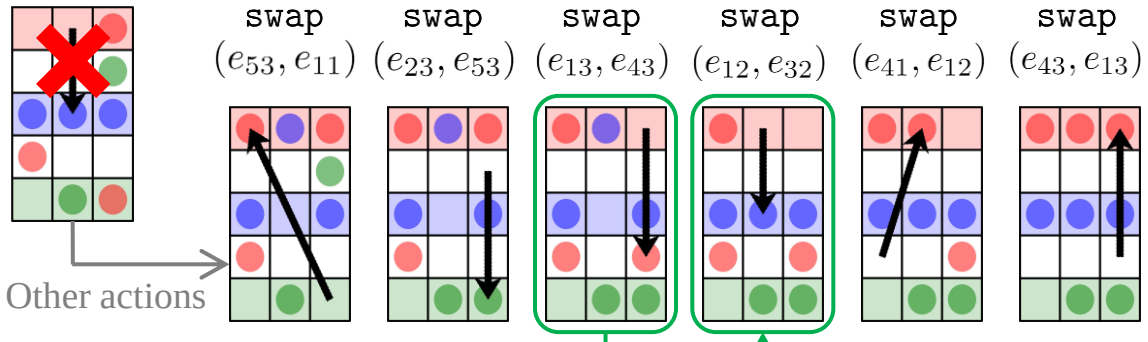
Invalidate Conditions and Update the Maintained Set in the Presence of Failures



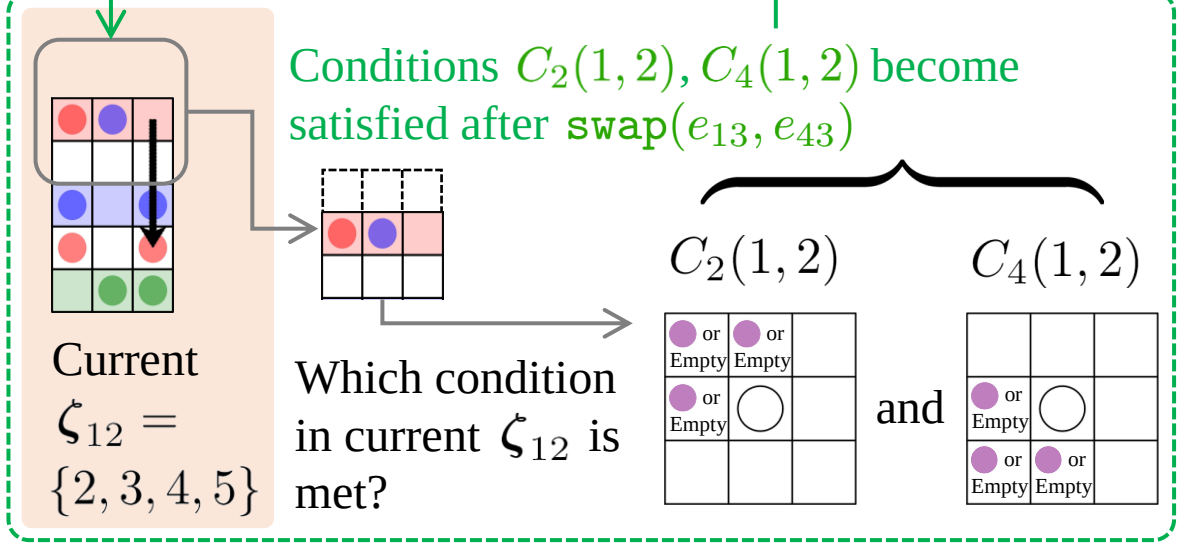
Update: $\zeta_{12} = \{1, 2, 3, 4, 5, 6\} \setminus \{1, 6\} = \{2, 3, 4, 5\}$

② New sequence

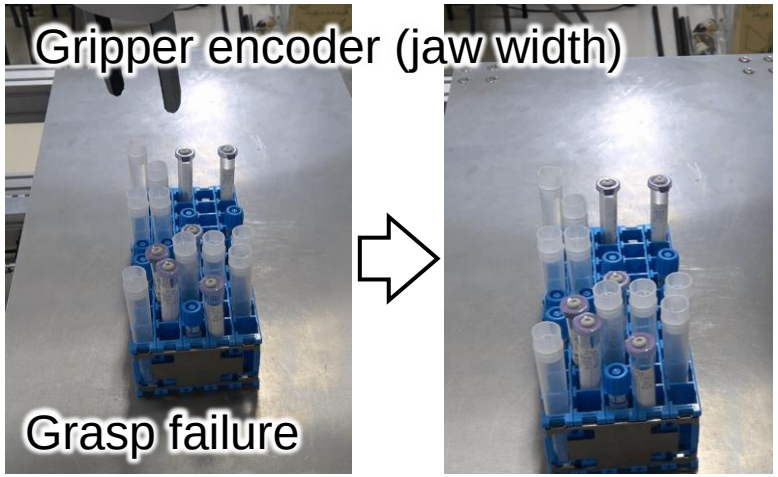
None of the conditions in the updated ζ_{12} are met

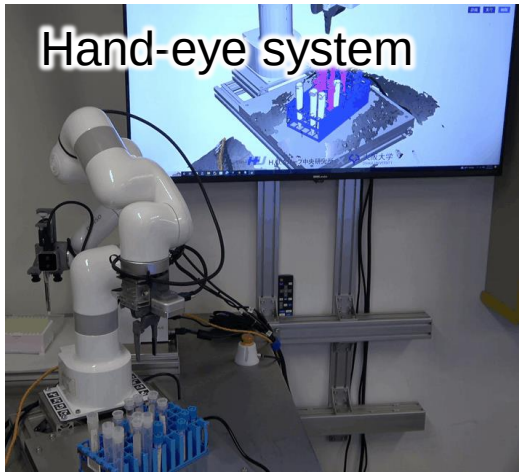


Conditions $C_2(1, 2), C_4(1, 2)$ become satisfied after $\texttt{swap}(e_{13}, e_{43})$



Close the Planning-Execution Loop





Experiments and Analysis

- See the mainmanuscript for experiments and analysis
- See the supplementary video for detailed execution results

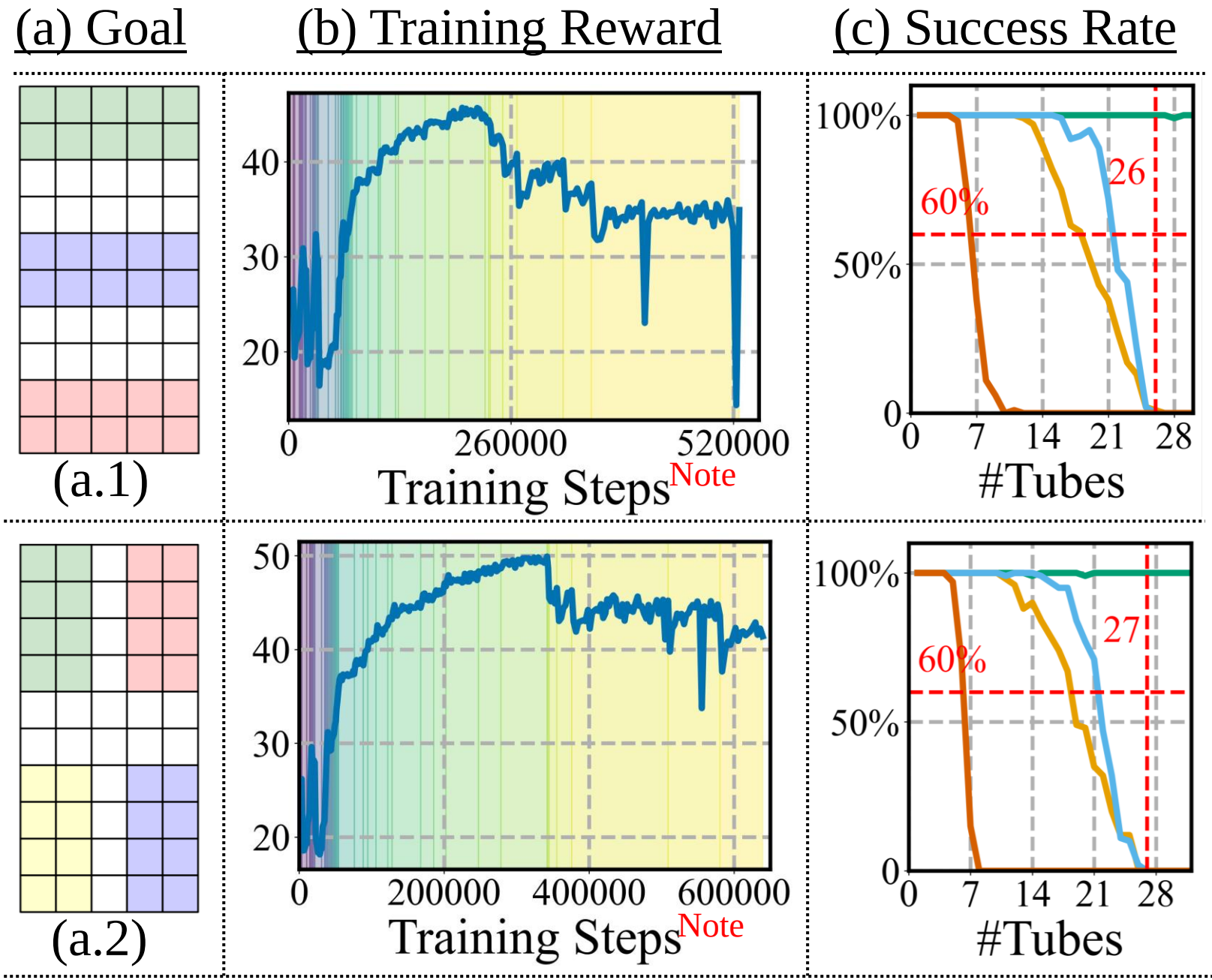
Conclusions

- Propose a combined **task-level RL** and motion planning framework.
- Develop an **A* post-processing** technique to expediate RL training.
- Close the loop of the task and motion level planner by maintaining a **condition set** for each rack slot.

大阪大学
OSAKA UNIVERSITY

H.U. Group Research Institute

Supplementary Materials: Enlarged Experimental Result Figures



RL-based task planner

A *-based task planner

MCTS-based task planner

PDDL-based task planner

Figure 18: *Please refer to the next three pages for the other parts of the image*

(d) Average Inference Time

(e) Average Solution Length

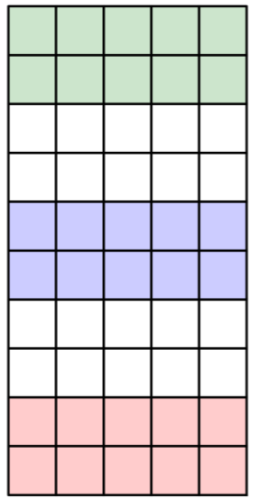
(a) Goal

(d.1) Difficulty I

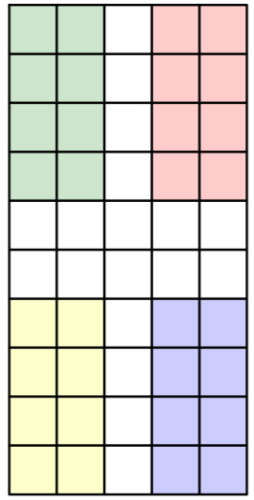
(d.2) Difficulty II

(e.1) Difficulty I

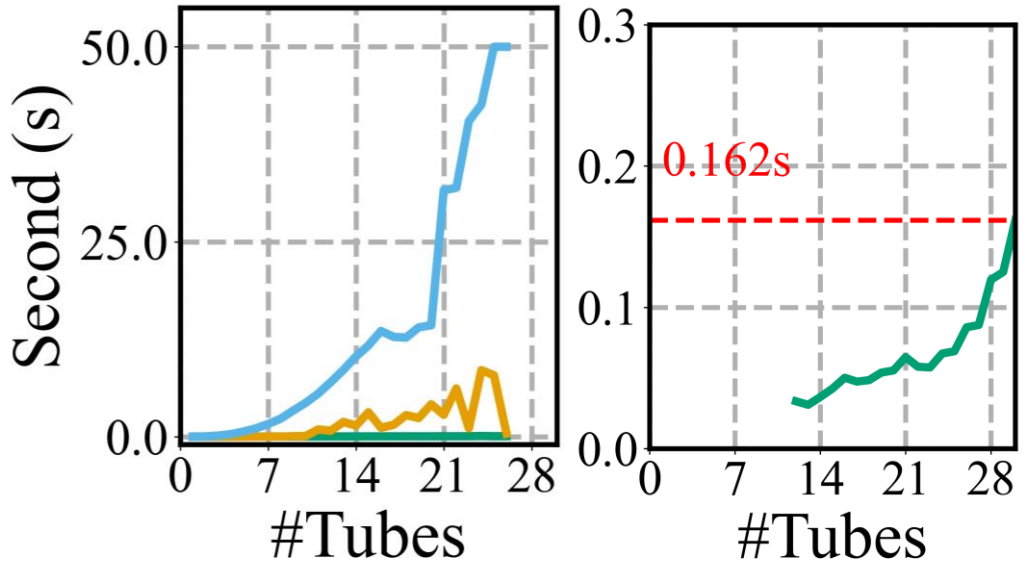
(e.2) Difficulty II

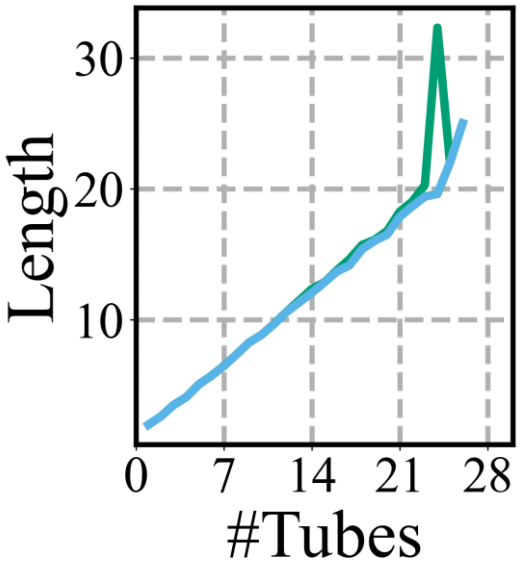


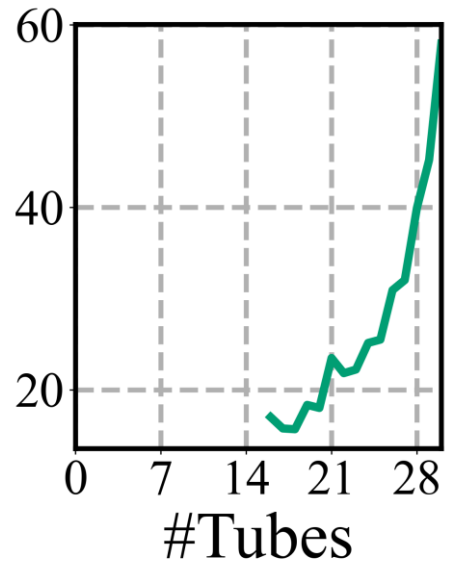
(a.1)

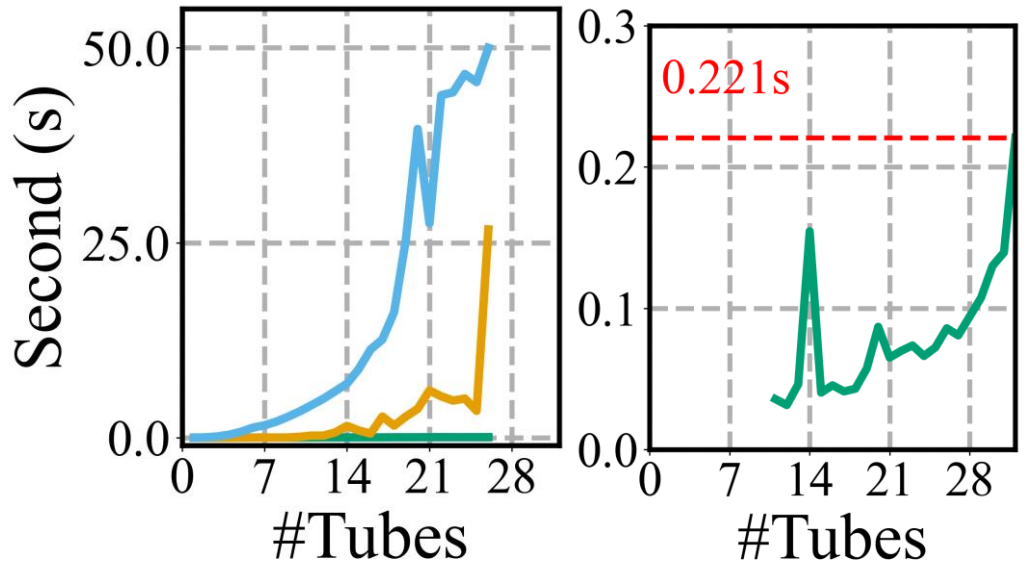


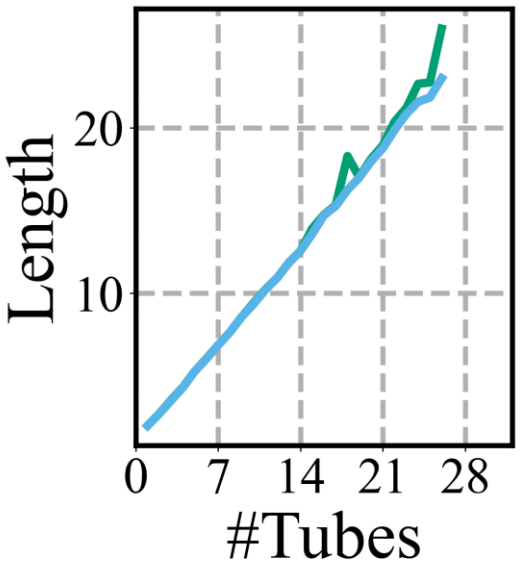
(a.2)

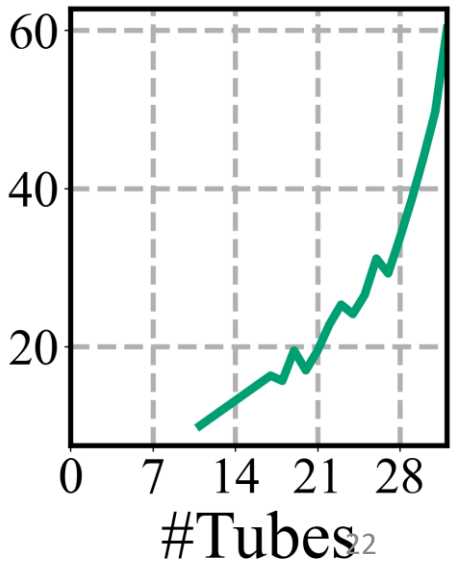




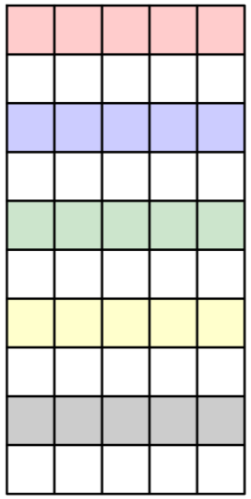






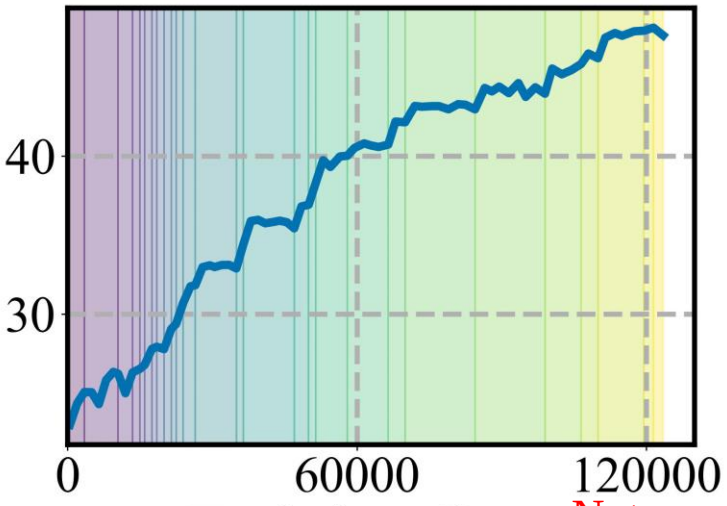


(a) Goal

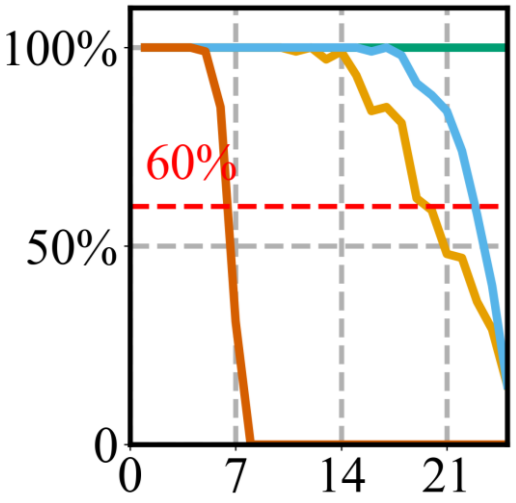


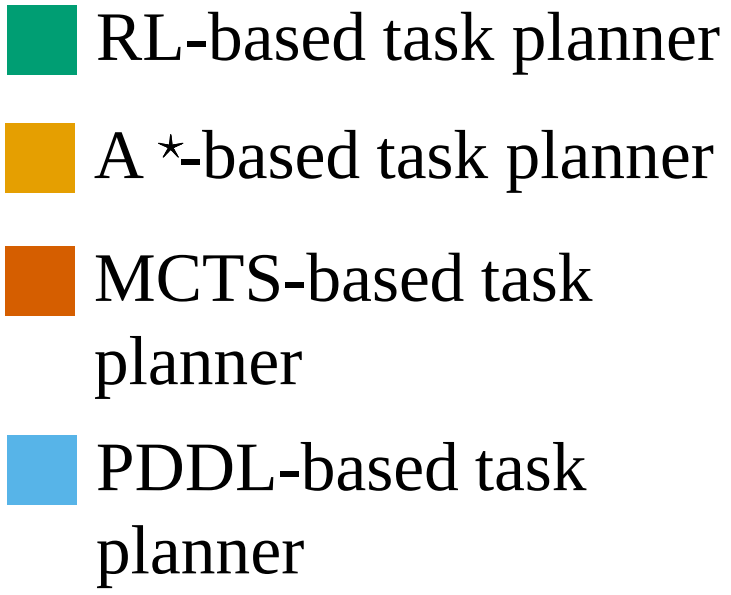
(a.3)

(b) Training Reward



(c) Success Rate





Note Each update of the learner's primary Q network is counted as a training step

Figure 18: *Please refer to the previous 2 pages for goal pattern (a.1) and (a.2). Please refer to the next pages for left part of goal pattern (a.3)*

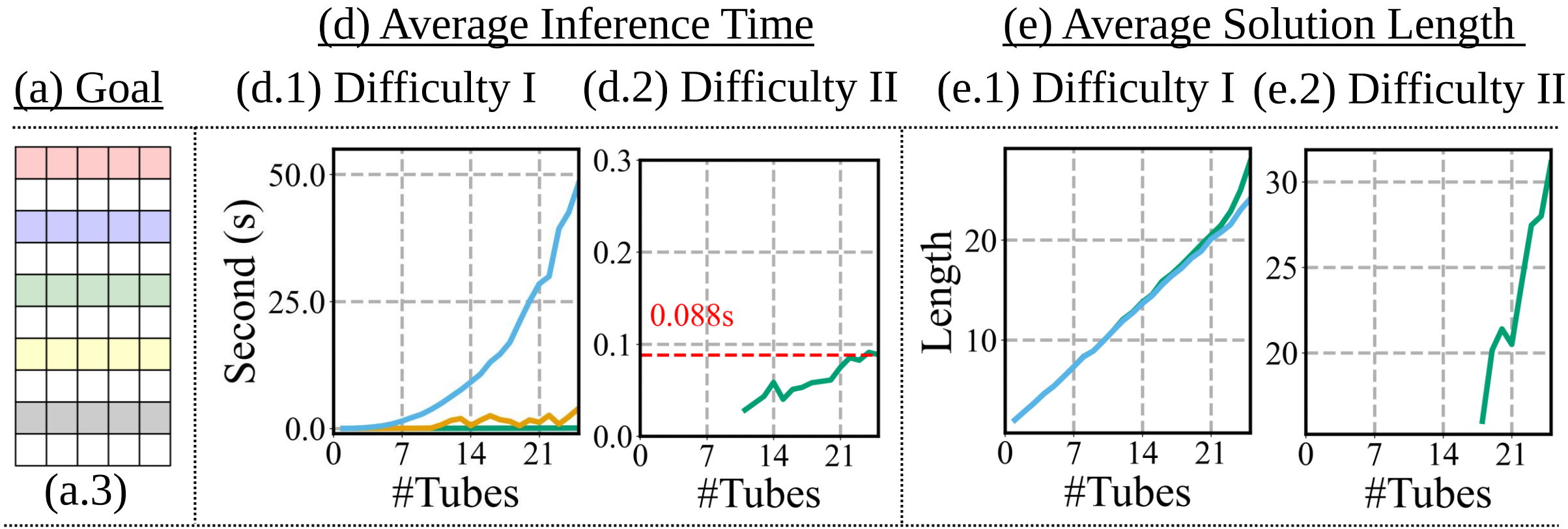


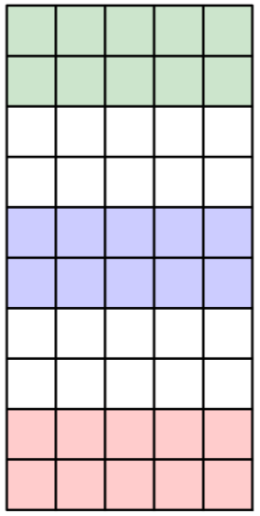
Figure 18. Experimental results for 3 different goal patterns shown in (a). (b) Training rewards of the RL-based planner. (c) Success rates of all planners. (d, e) Average inference time and average solution length of successful solutions.

Please refer to the previous three pages for the other parts of the image

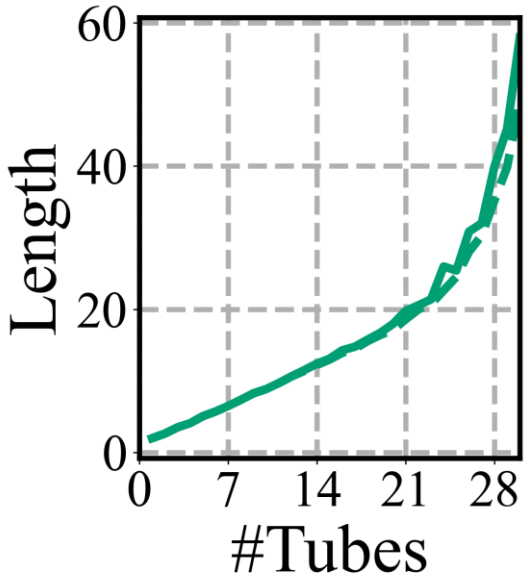
(a) Goal

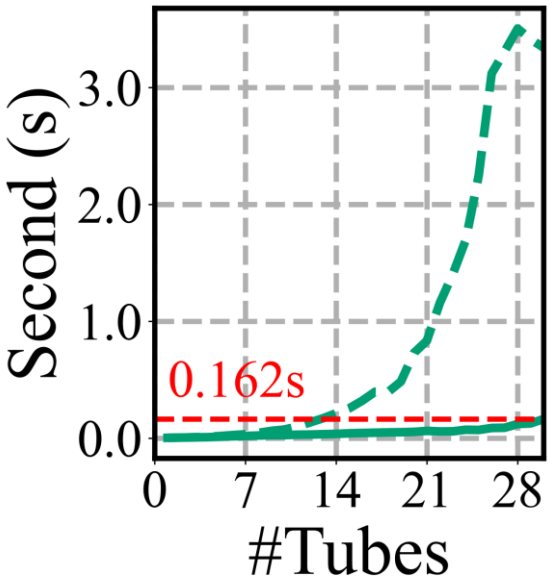
(b) Average Solution Length

(c) Average Inference Time



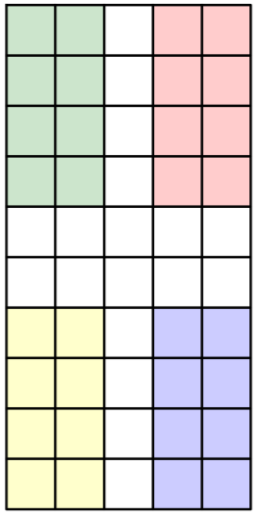
(a.1)



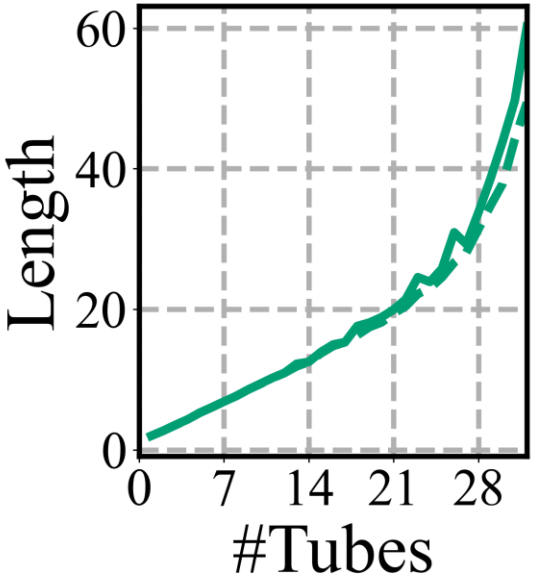


RL-based task planner

RL-based task planner with fluctuation suppression



(a.2)



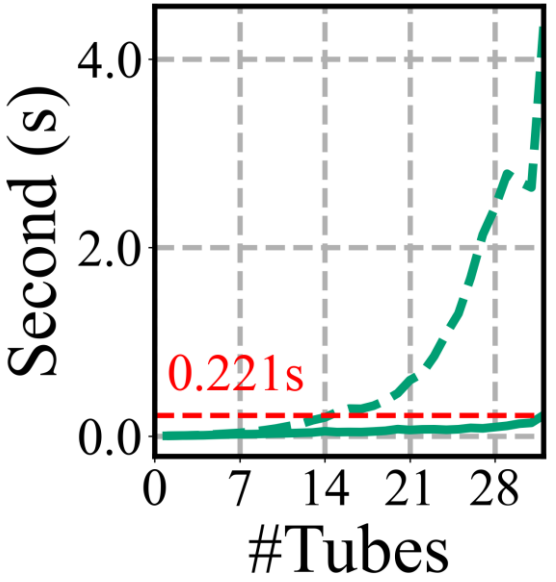


Figure 19: *Please refer to the next page for the lower part of the image*

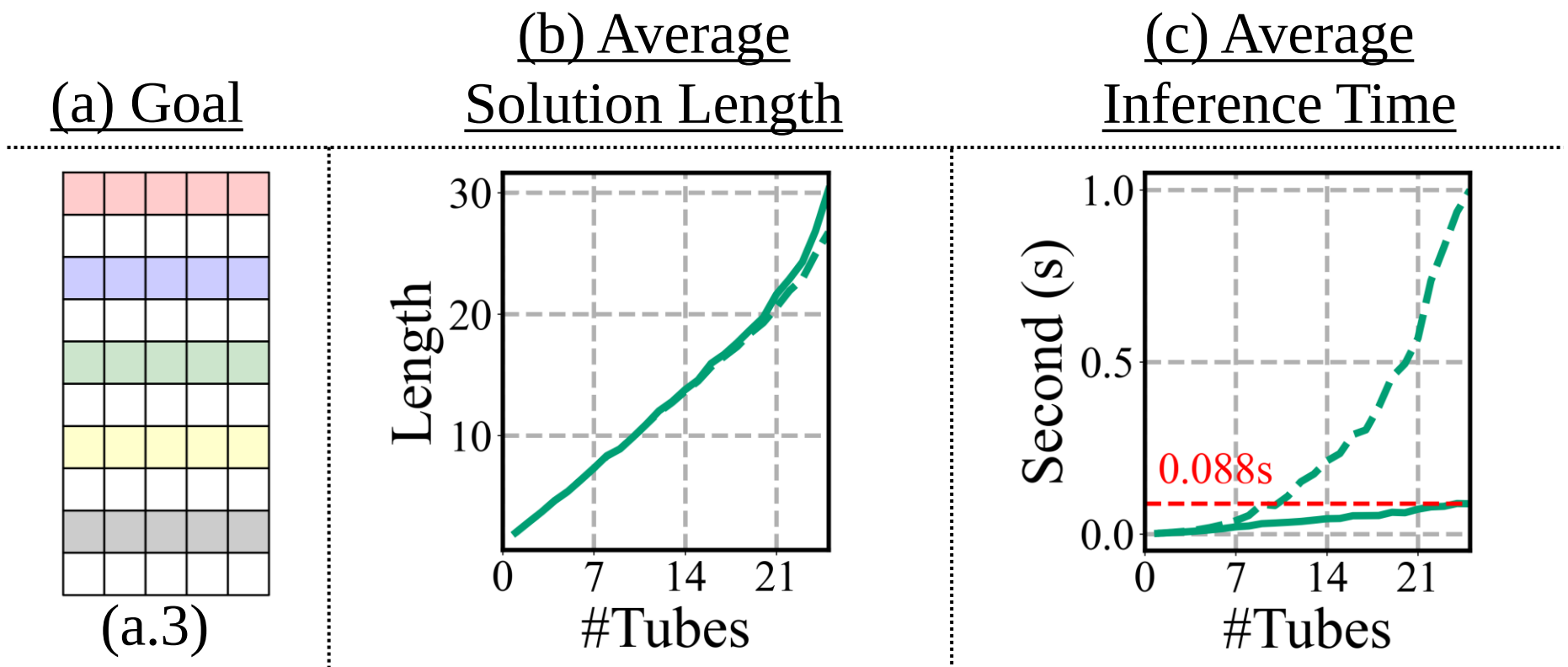


Figure 19. Comparison of the original RL-based task planner and the RL-based planner with fluctuation suppression.

Please refer to the previous page for the upper part of the image

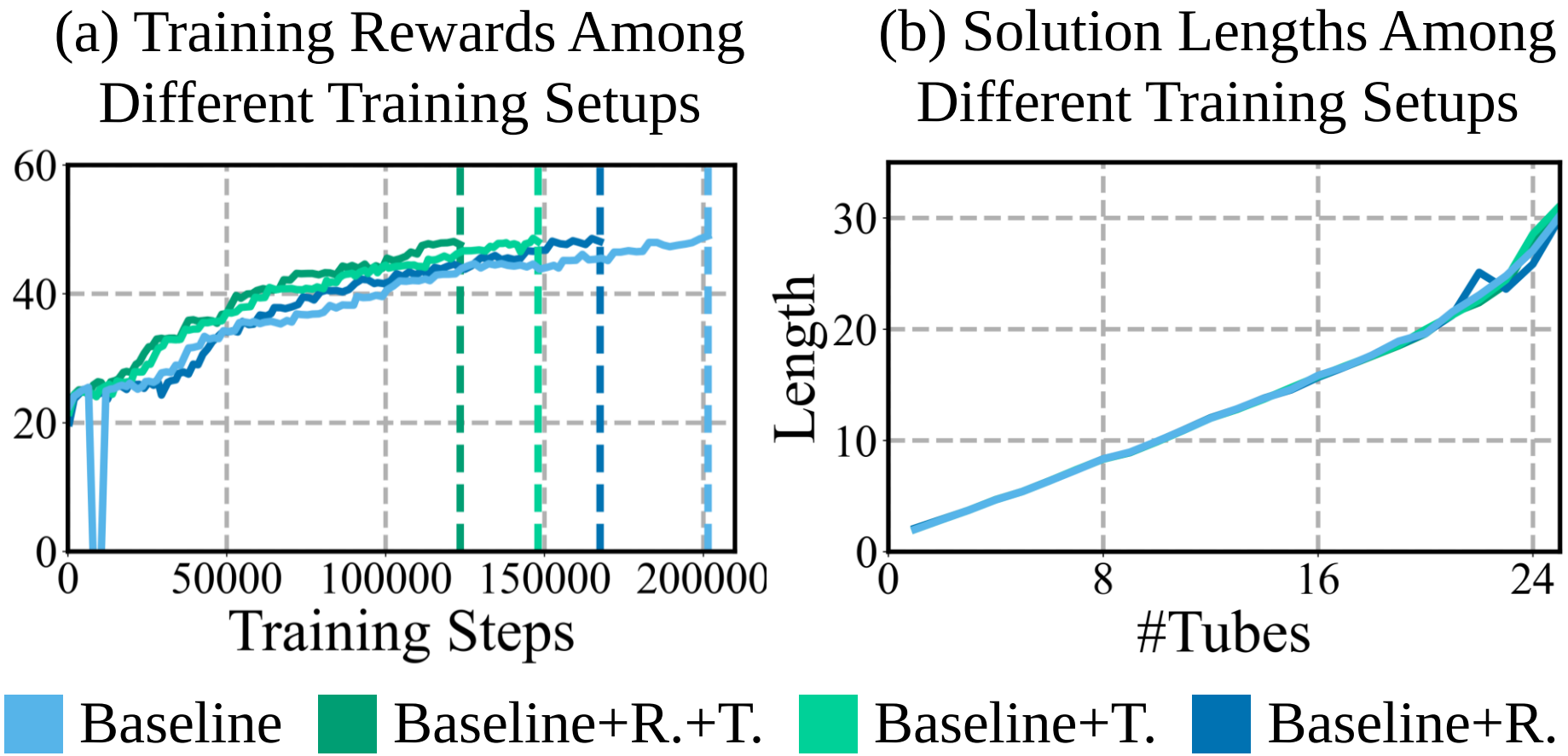


Figure 20. (a) Average rewards obtained in the evaluator of distributed Q learning. The dashed vertical lines indicate the training steps at which the agents finished learning. (b) The average solution lengths for the agents trained by different RL configurations. For each number of test tubes, 100 trials were evaluated with random initial states. All agents achieved a 100% success rate in solving these trials.

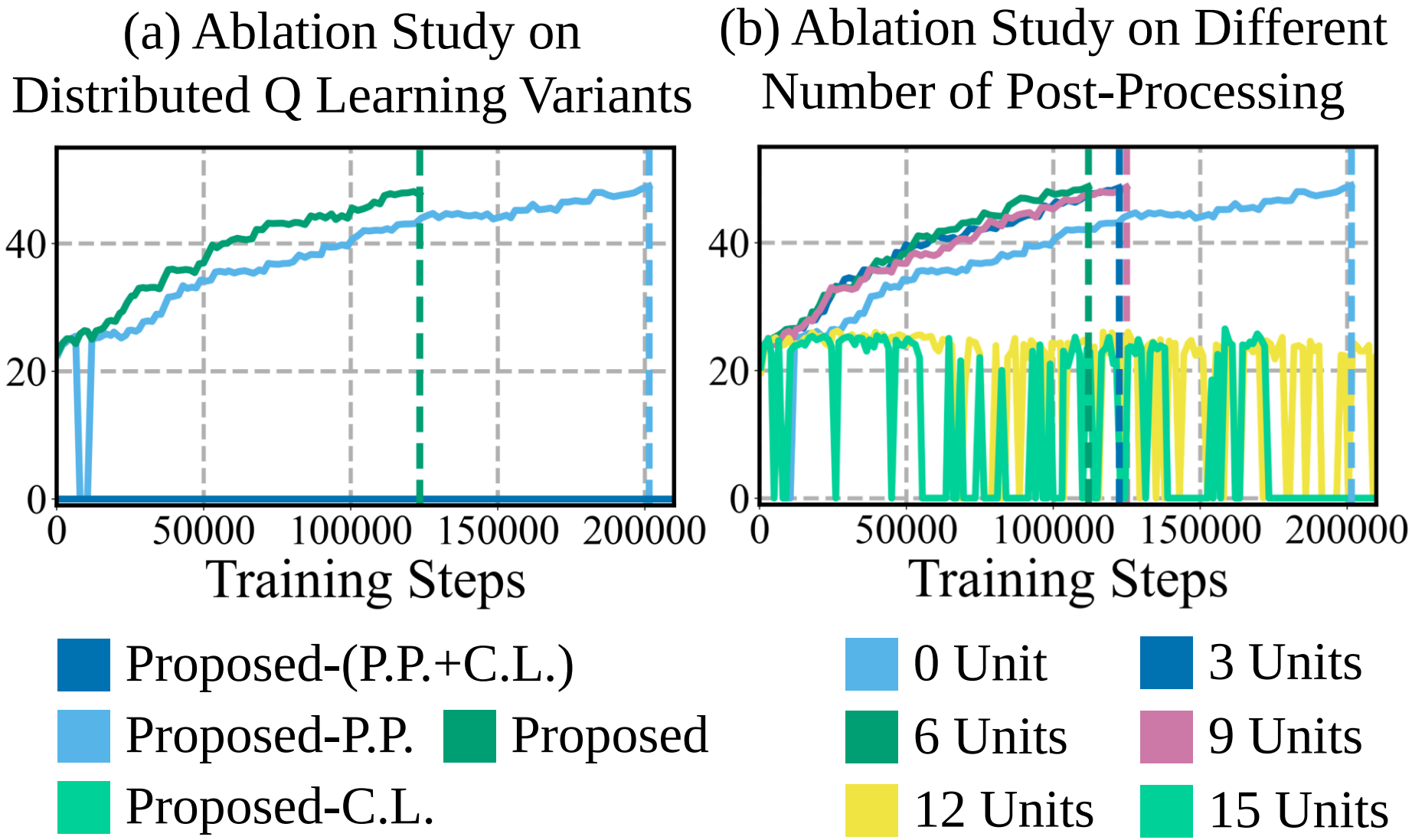


Figure 21. (a) Results of different distributed Q learning component combinations. (b) Influence of the number of units in the post-processing component.

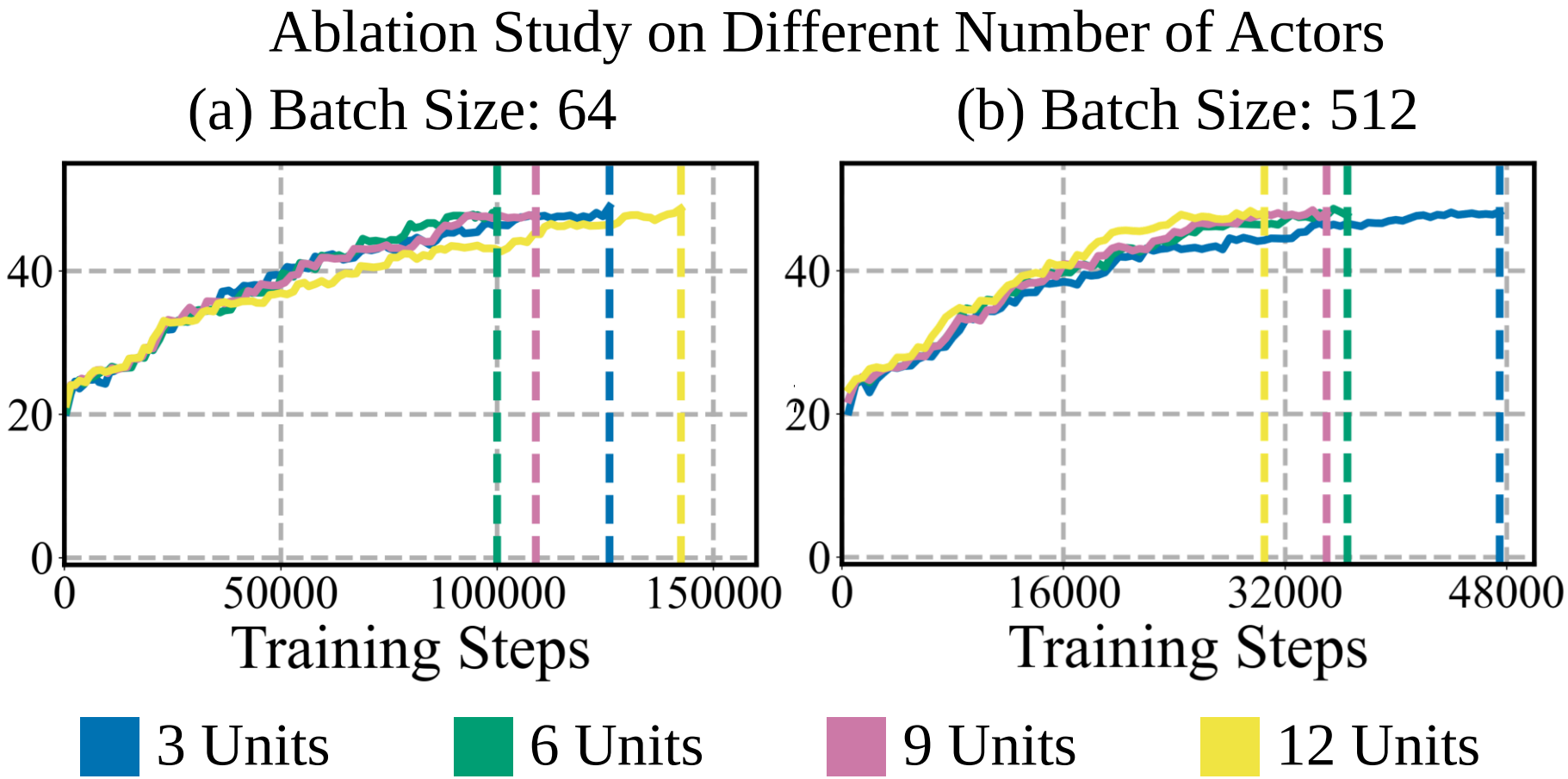


Figure 22. Influence of different numbers of parallel actors using (a) batch size of 64, and (b) batch size of 512, respectively.

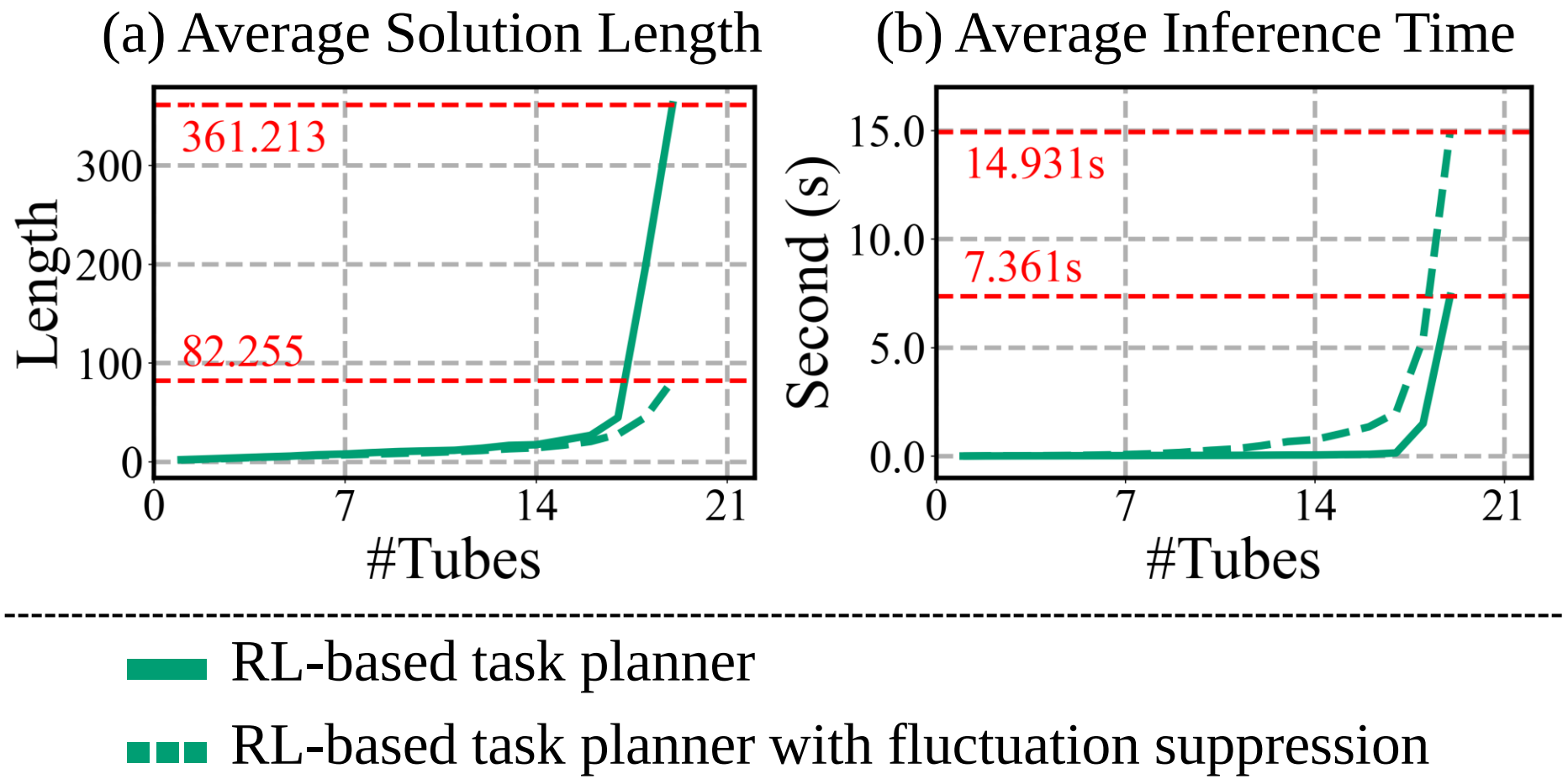
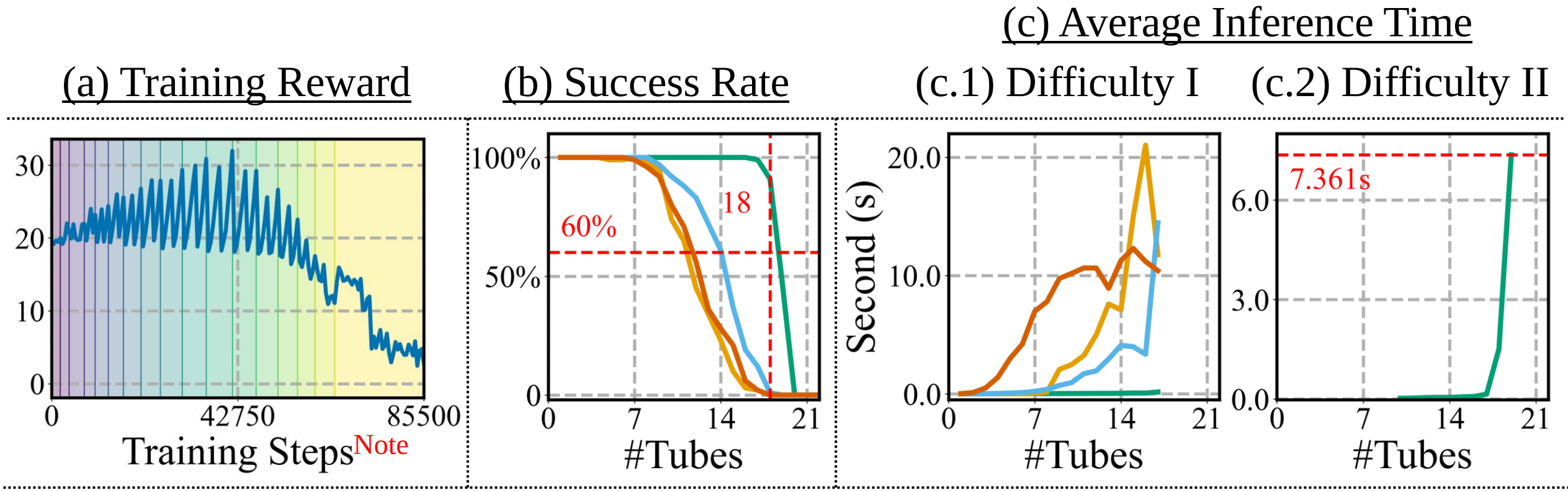


Figure 24. Comparison of the original RL-based task planner and the RL-based planner with fluctuation suppression for arbitrary goal patterns.



Note Each update of the learner's primary Q network is counted as a training step.

Figure 25: *Please refer to the next page for the right part of the image.*

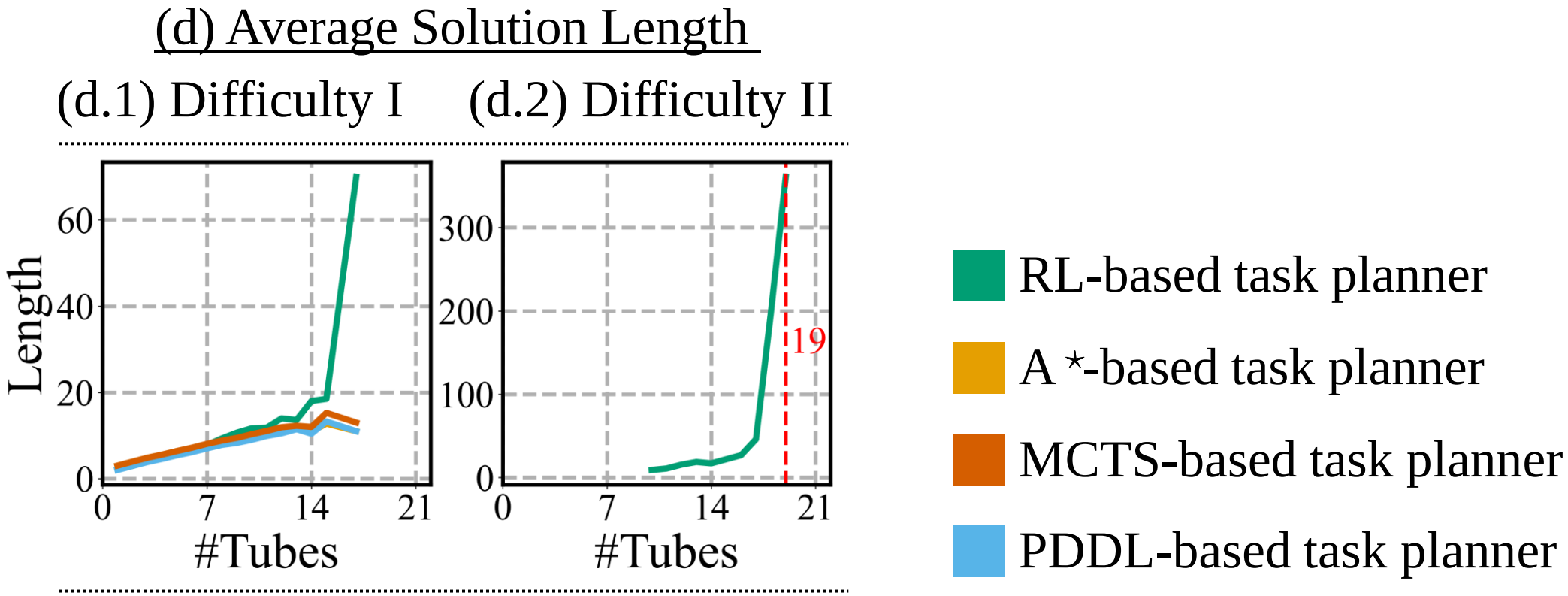


Figure 25. Experimental results for arbitrary goal patterns. (a) Training reward of the RL-based planner. (c) Success rates of all planners. (d, e) Average inference time and average solution length of successful solutions.

Please refer to the previous page for the left part of the image.

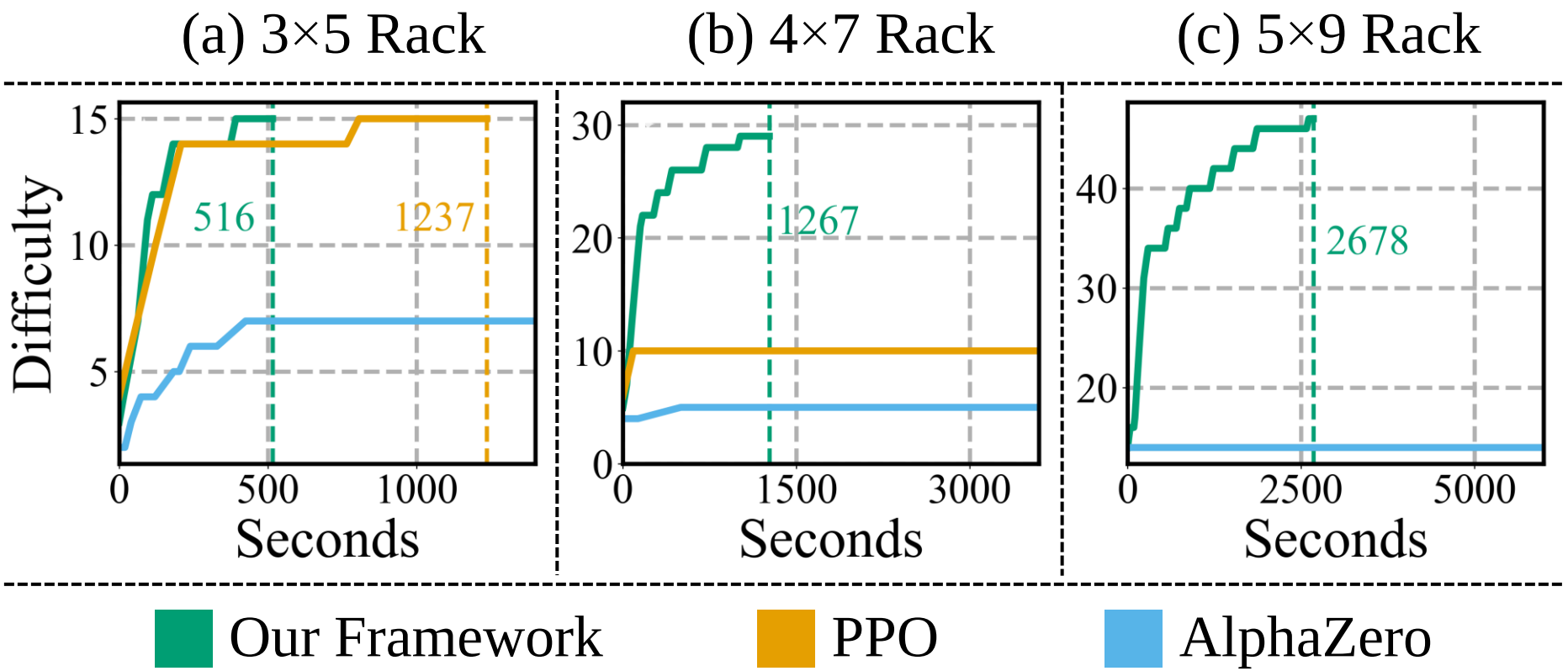


Figure 33. Number of difficulty levels completed in curriculum learning with respect to learning time. The achievable difficulty level for each RL method stops increase beyond the shown x-axis limit. Especially for (c), both PPO an AlphaZero achieved difficulty level 4. Their results overlap and only the PPO curve is visible.

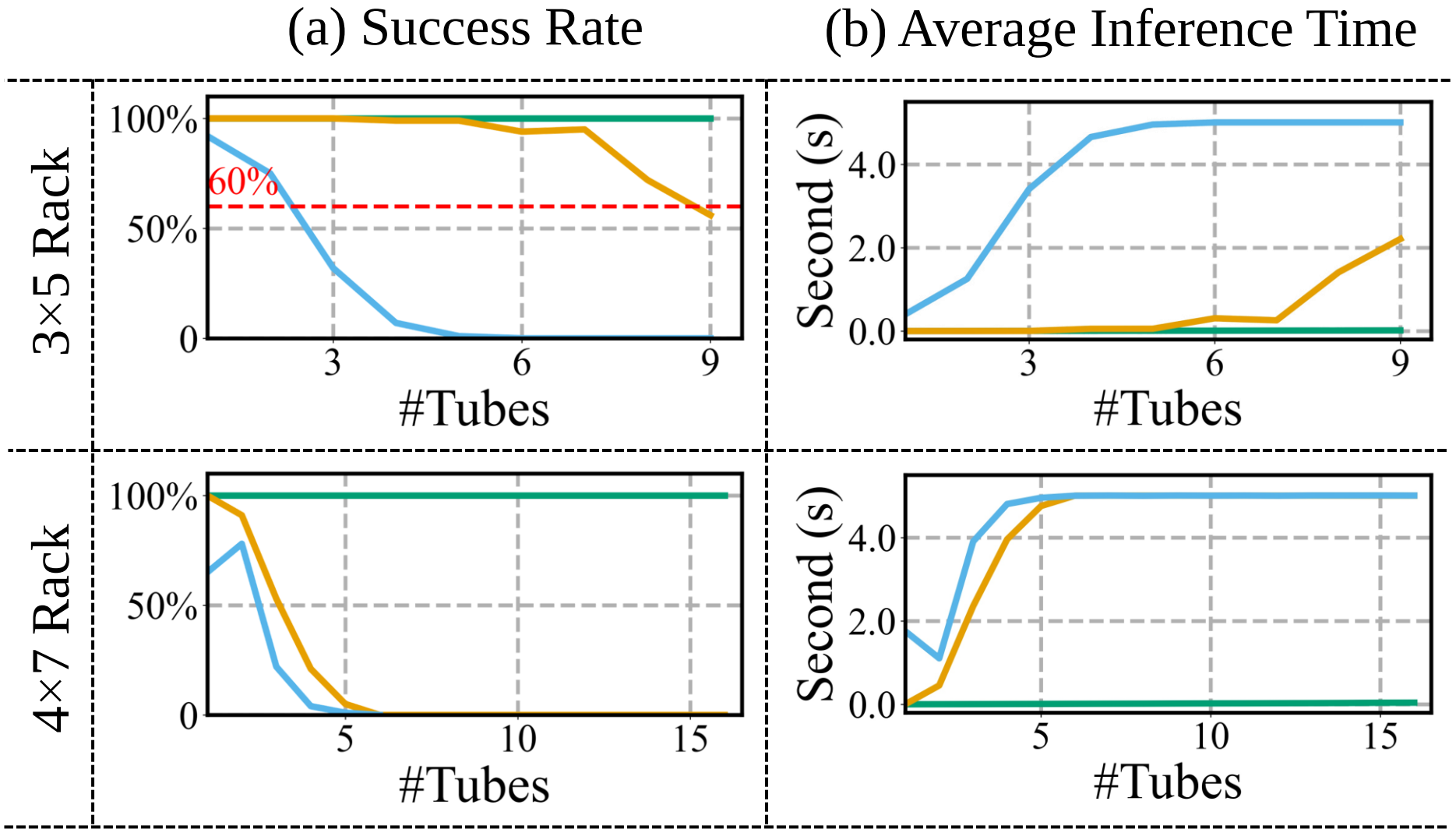


Figure 34: *Please refer to the next page for the lower part of the image*

Please refer to the previous page for the top part of the image

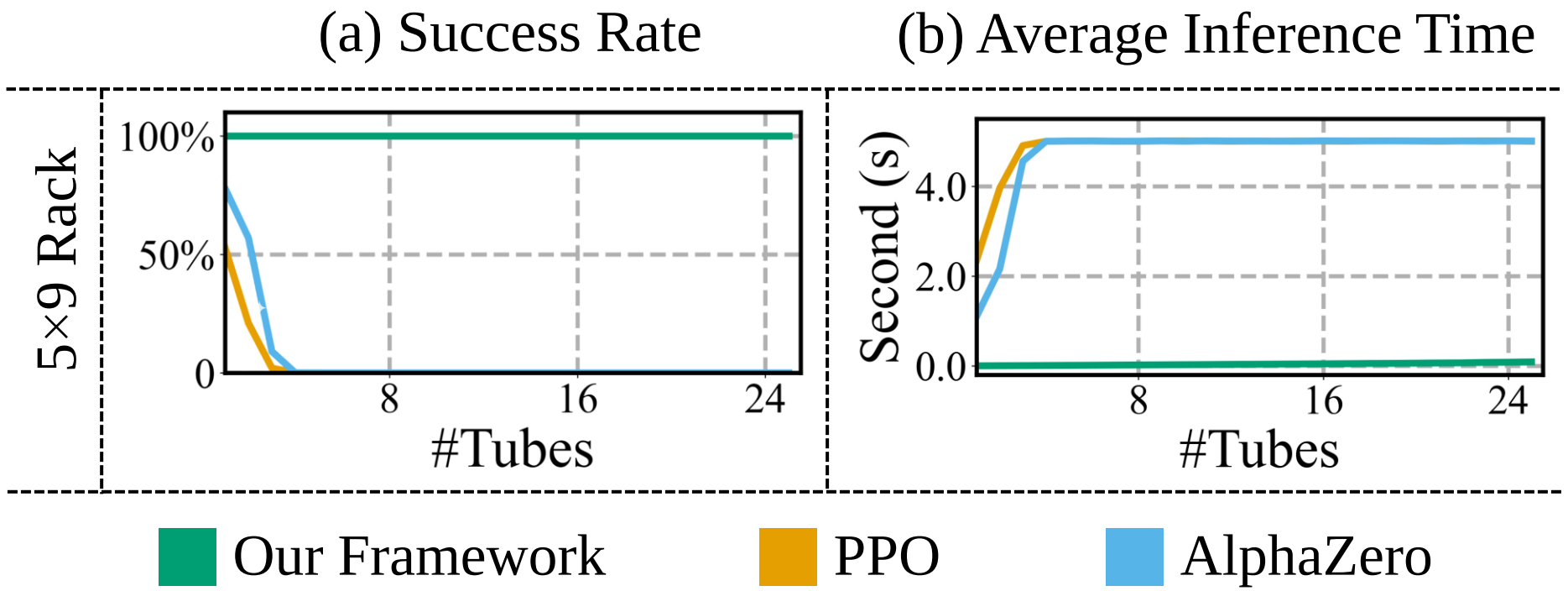


Figure 34. Success rate and average inference time of the proposed RL method, PPO, and AlphaZero on the testing problem.

Goal Pattern:

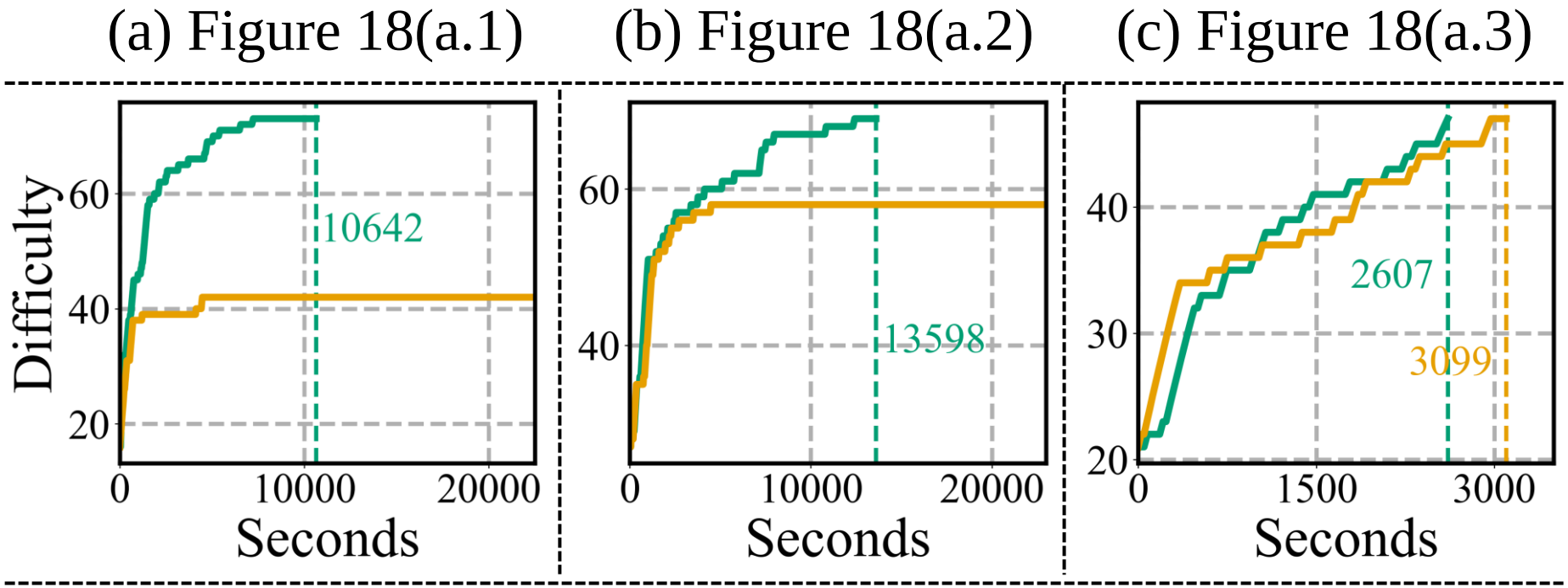


Figure 36. Number of difficulty levels achieved in curriculum learning with respect to learning time.

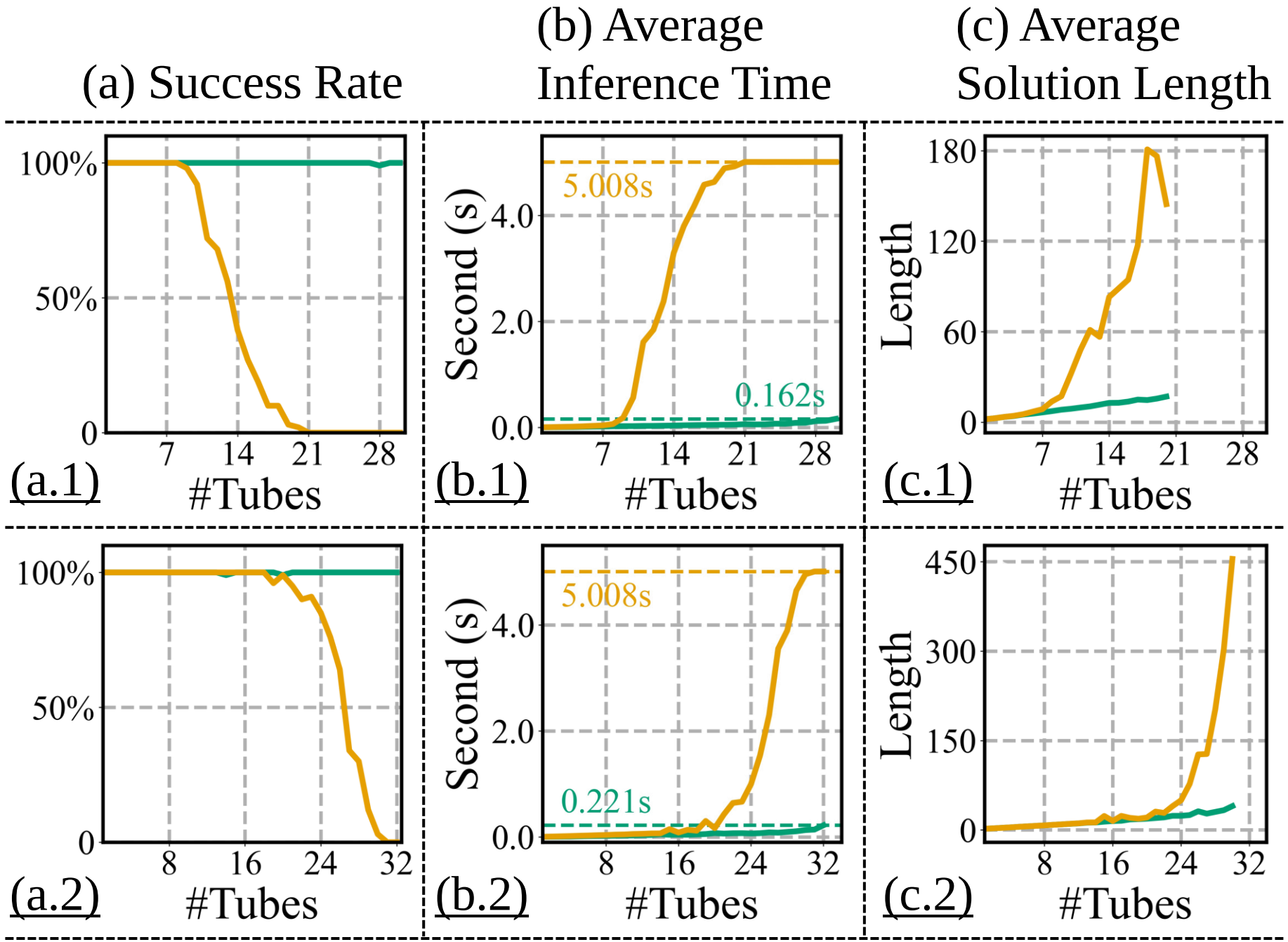


Figure 37: *Please refer to the next pages for the lower parts of the image*

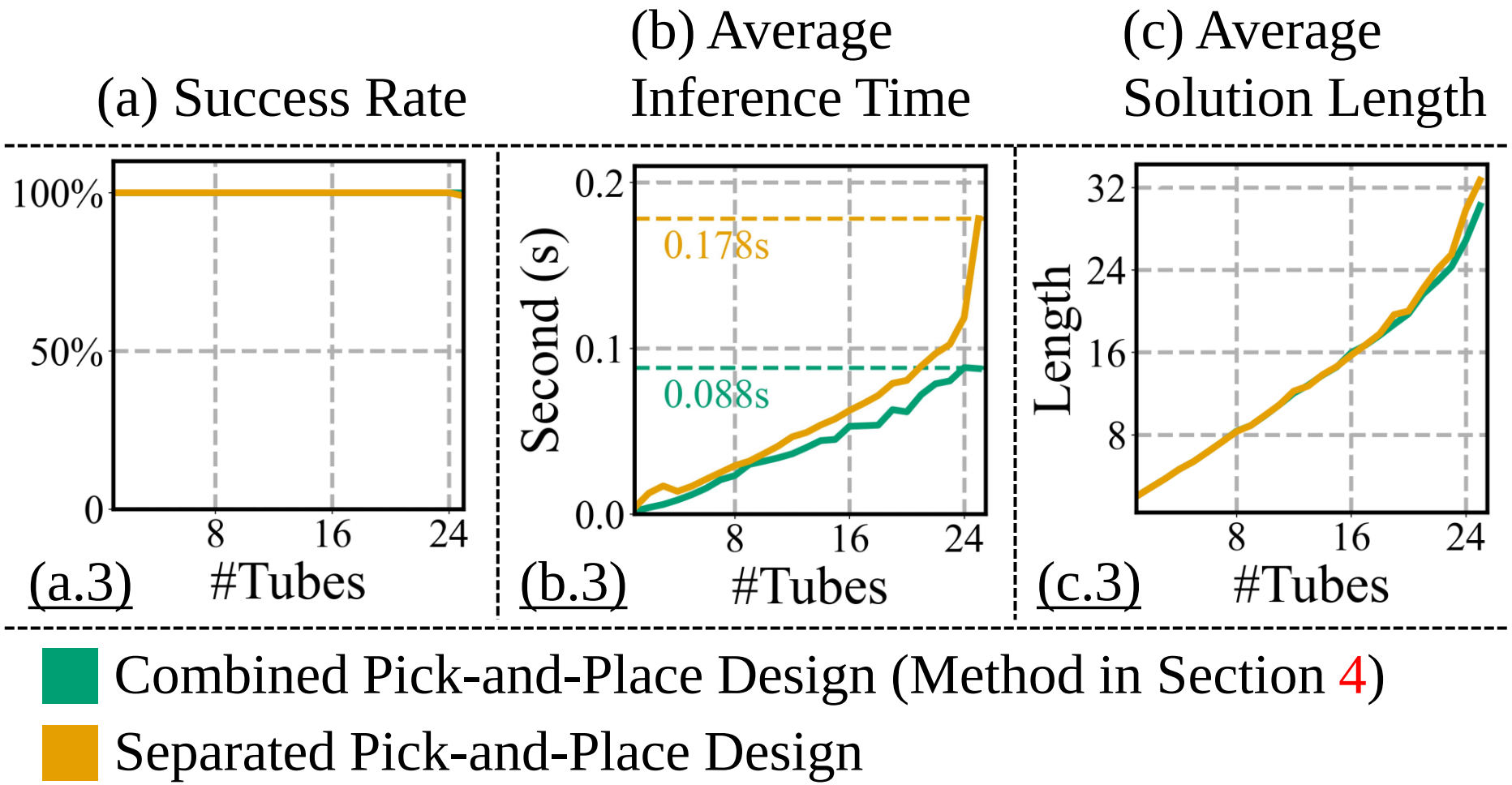


Figure 37. Success rate, average inference time, and average solution length for the combined and separated pick-and-place designs..