CO-PILOT : COllaborative Planning and reInforcement Learning On sub-Task curriculum

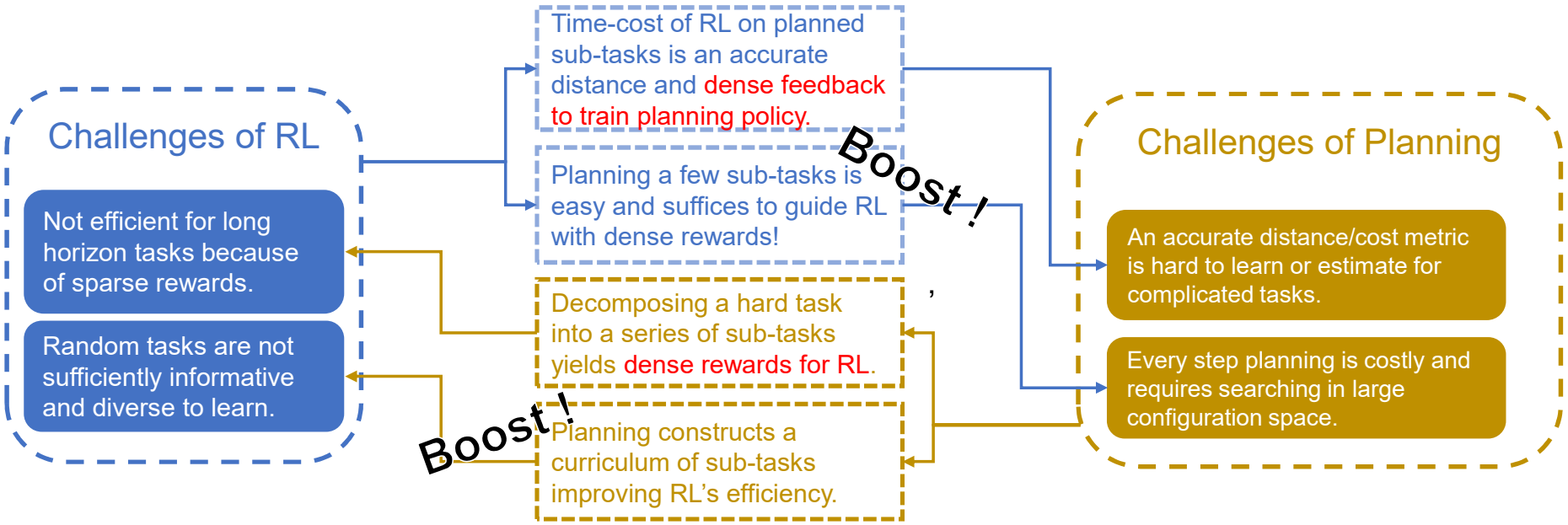
Shuang Ao[1], Tianyi Zhou[2,3], Guodong Long[1], Qinghua Lu[4], Liming Zhu[4] and Jing Jiang[1]

Australian Artificial Intelligence Institute, University of Technology Sydney[1]
University of Washington, Seattle[2]
University of Maryland, College Park[3]
CSIRO's Data61, Australia[4]

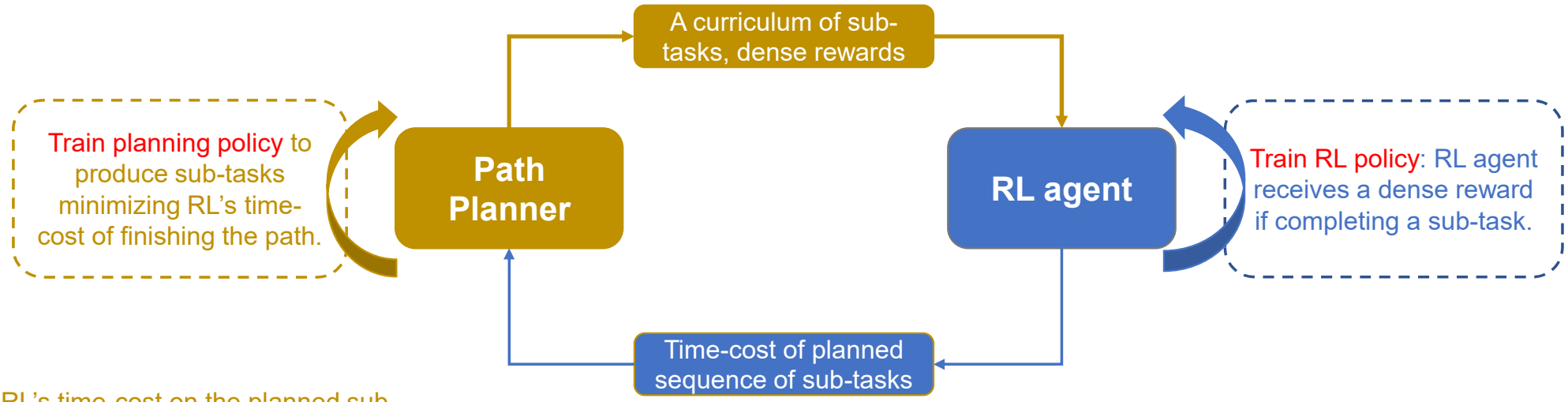
NeurIPS 2021

code: https://github.com/Shuang-AO/CO-PILOT

Collaborative Learning between RL and Planning ("Learn to Plan")



Mutual Training of Path-Planning Policy and RL Agent



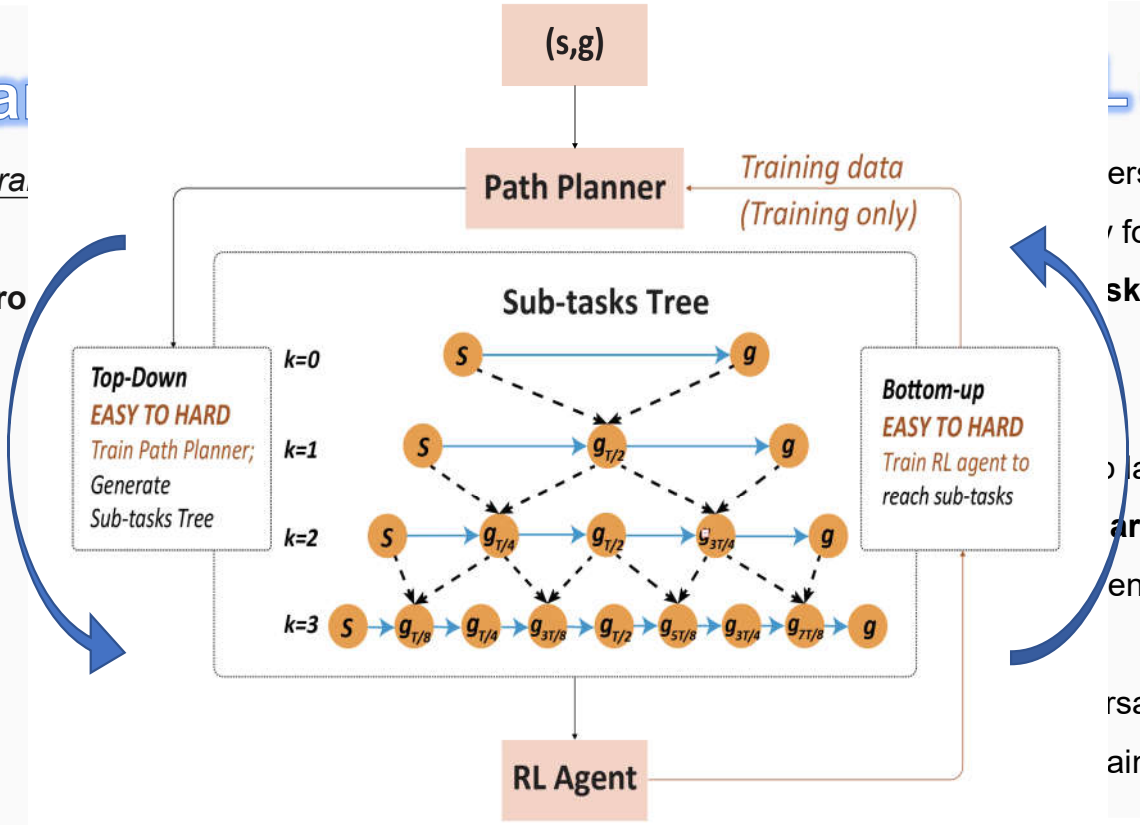
- RL's time-cost on the planned sub-tasks can be the training data to train the policy of path planner;
- Hence, neither prior knowledge nor external feedback is needed to train the path planner.

- Sub-tasks produced by the path-planner provide dense rewards and detailed guidance to RL along long-horizon tasks;
- Hence, the path-planner improves RL's sample efficiency.

CO-PILOT: EASY-TO-HARD sub-task curriculum for both RL and Planning

Path Pla

- Sub-task tree is not *a hiera*
  sub-goal space.
- Path planner **learns to pro**



L agent

ers, the RL agent first learns how
following **a more detailed path**
sks, which is **easy** and provides

layers, the curriculum gradually
ardness for RL by reducing the
en (s, g).

rsal forms an **EASY-TO-HARD**
ain the RL policy.

Path-Planner recursively generates Coarse-to-Fine min-cost Path

Apply planning policy π_p to interpolate a sub-task between a consecutive sub-goal pair (g, g') from the upper layer: (Recursive)

① $\Pr_{\pi_p}(g_{0:T}|g_0 = s_0, g_T = g) \triangleq \Pr_{\pi_p}\left(g_{0:\frac{T}{2}} \middle| g_0, g_{\frac{T}{2}}\right) \Pr_{\pi_p}\left(g_{\frac{T}{2}:T} \middle| g_{\frac{T}{2}}, g\right) \pi_p\left(g_{\frac{T}{2}} \middle| s_0, g\right)$

Planning policy can be applied to **any feasible pair** (s, g).

Define the cost of the sub-task sequence in layer$-k$:

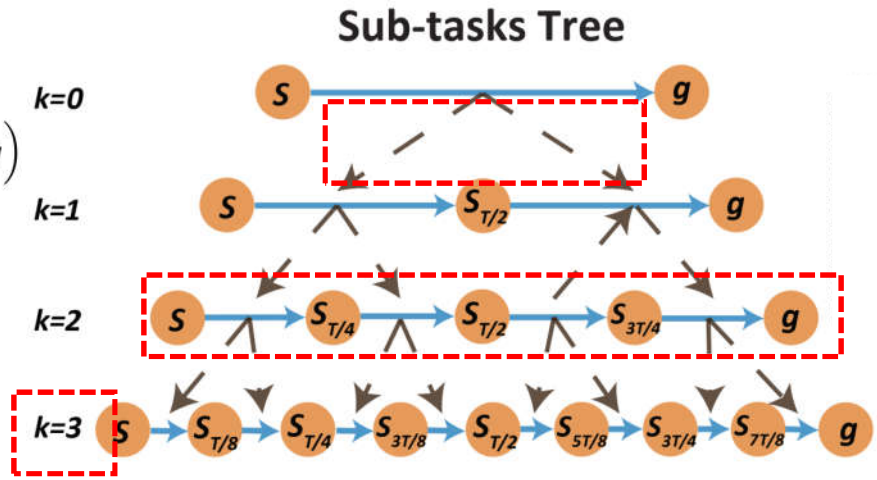
② $c\left(g^k_{0:2^k}\right) \triangleq \sum_{t=0}^{2^k-1} c\left(g^k_{tT/2^k}, g^k_{(t+1)T/2^k}\right)$

Path planner is **adaptive to the RL learning progress.**

Update planning policy π_p to minimize the cost of sub-tasks up to layer$-k$

③ $\nabla J_{\pi_p} = \mathbb{E}_{g_{0:T}\sim\pi_p}\left[c(g_{0:T}) \cdot \nabla \log \Pr_{\pi_p}(g_{0:T}|s_0, g)\right]$

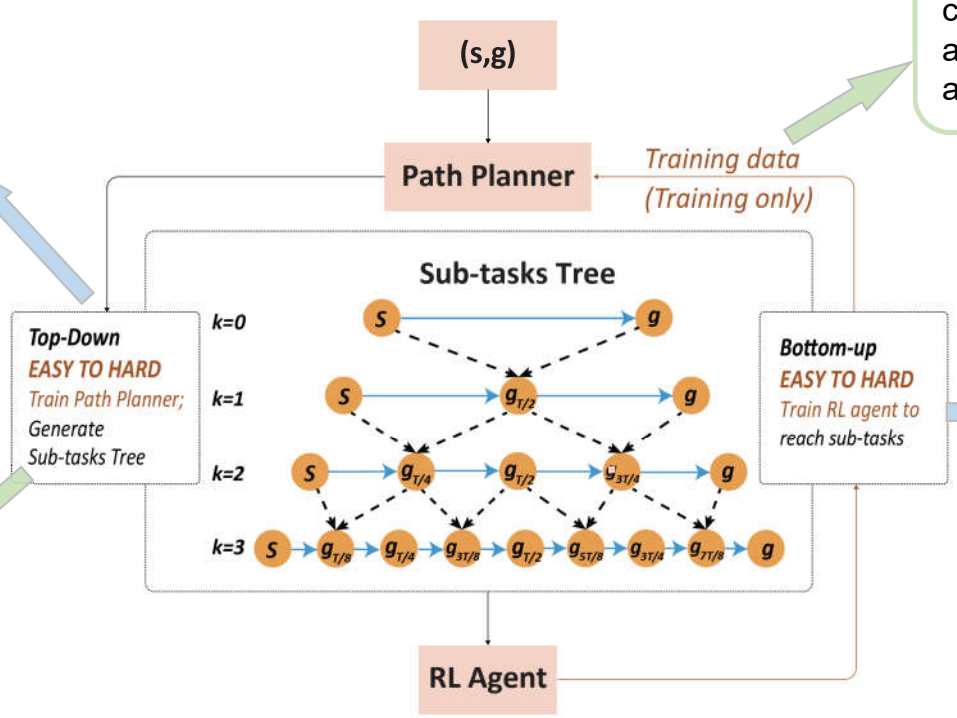
Easy-to-hard train path planner from top to bottom layers.



Main Ideas of CO-PILOT

Top-down construction eases and accelerates the training of path-planner.

Time-cost of RL on sub-tasks can provide dense and accurate distance to evaluate and train the path-planner.

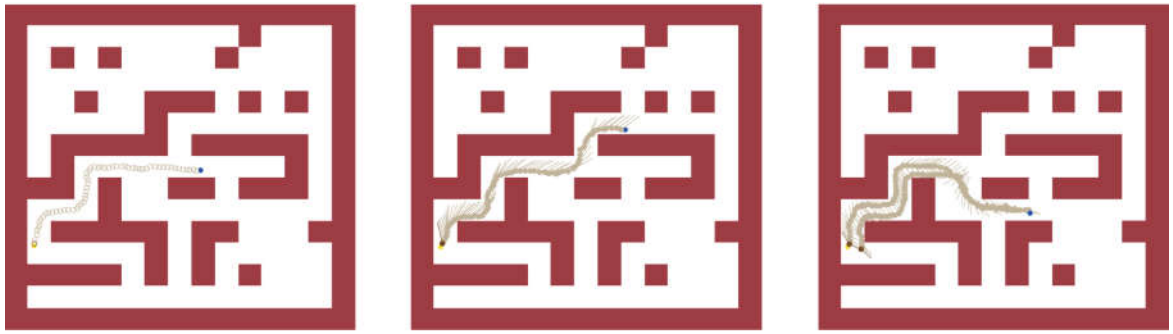


- Dense rewards for RL;
- Time-cost efficient path of sub-tasks for RL;
- Sub-tasks adaptive to RL learning progress.

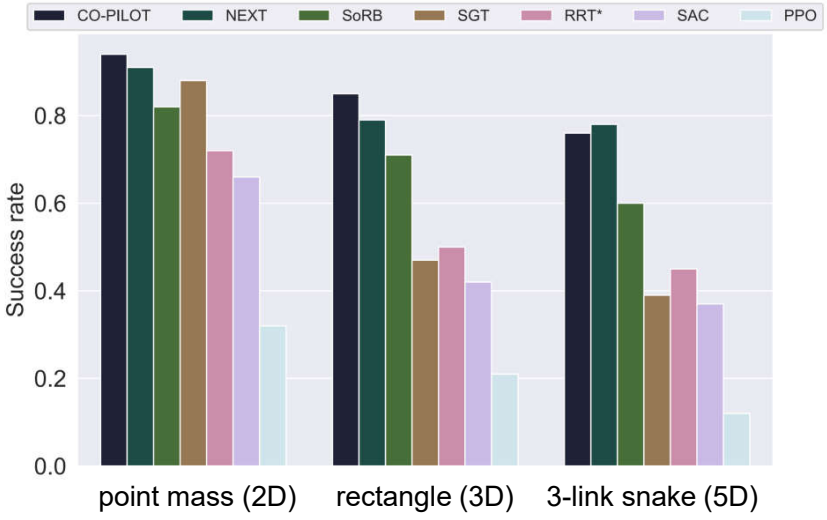
Bottom-up traversal accelerates the training of RL agent as an easy-to-hard sub-task curriculum.

- **Planning serves the RL agent as a co-pilot:** Planner predicts **sub-tasks** to provide **dense rewards** for the RL agent
- RL's time cost is used to train an adaptive planning policy: more accurate distance metric and training objective.
- **Top-down and Bottom-up traversal of the sub-task tree** form an easy-to-hard training curriculum for each of them.

Experiment: 2D maze



Task: In a maze, the agent starts from an initial state s to a goal state g. (s, g) are randomly sampled from a uniform distribution. The agent only receives a reward when it comes close to g.

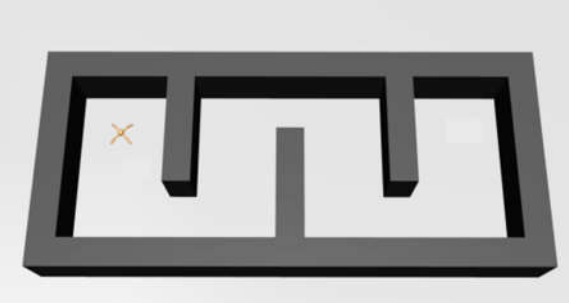


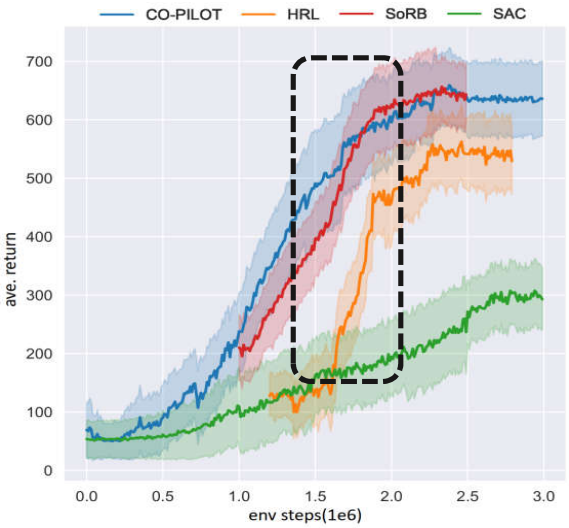
Combine RL with planning: CO-PILOT (our method), SoRB (2019)

Planning: NEXT (2020), SGT (2020), RRT* (2011)

RL: SAC (2018), PPO (2017)

Experiment: Mujoco Ant



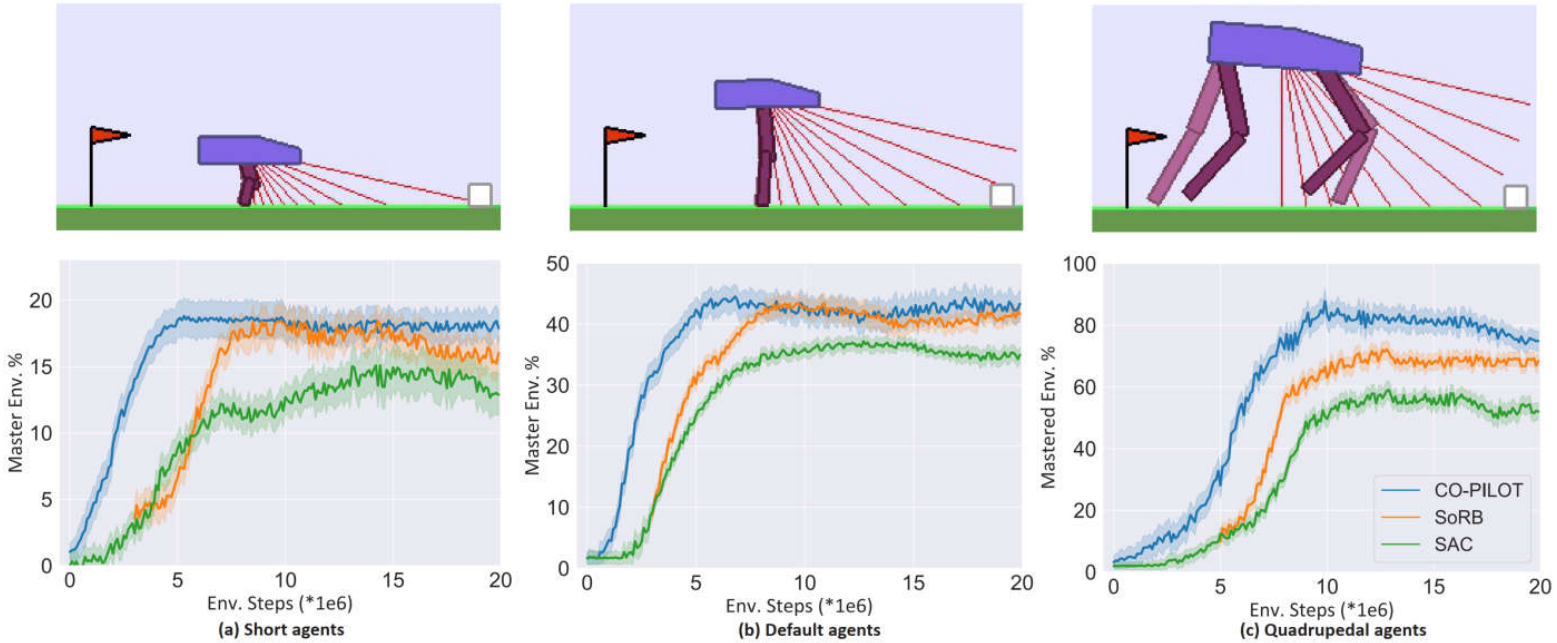


Task: A continuous space task, in which the agent needs to navigate between initial and goal state. When the agent reaches the goal state, it will receive a large reward.

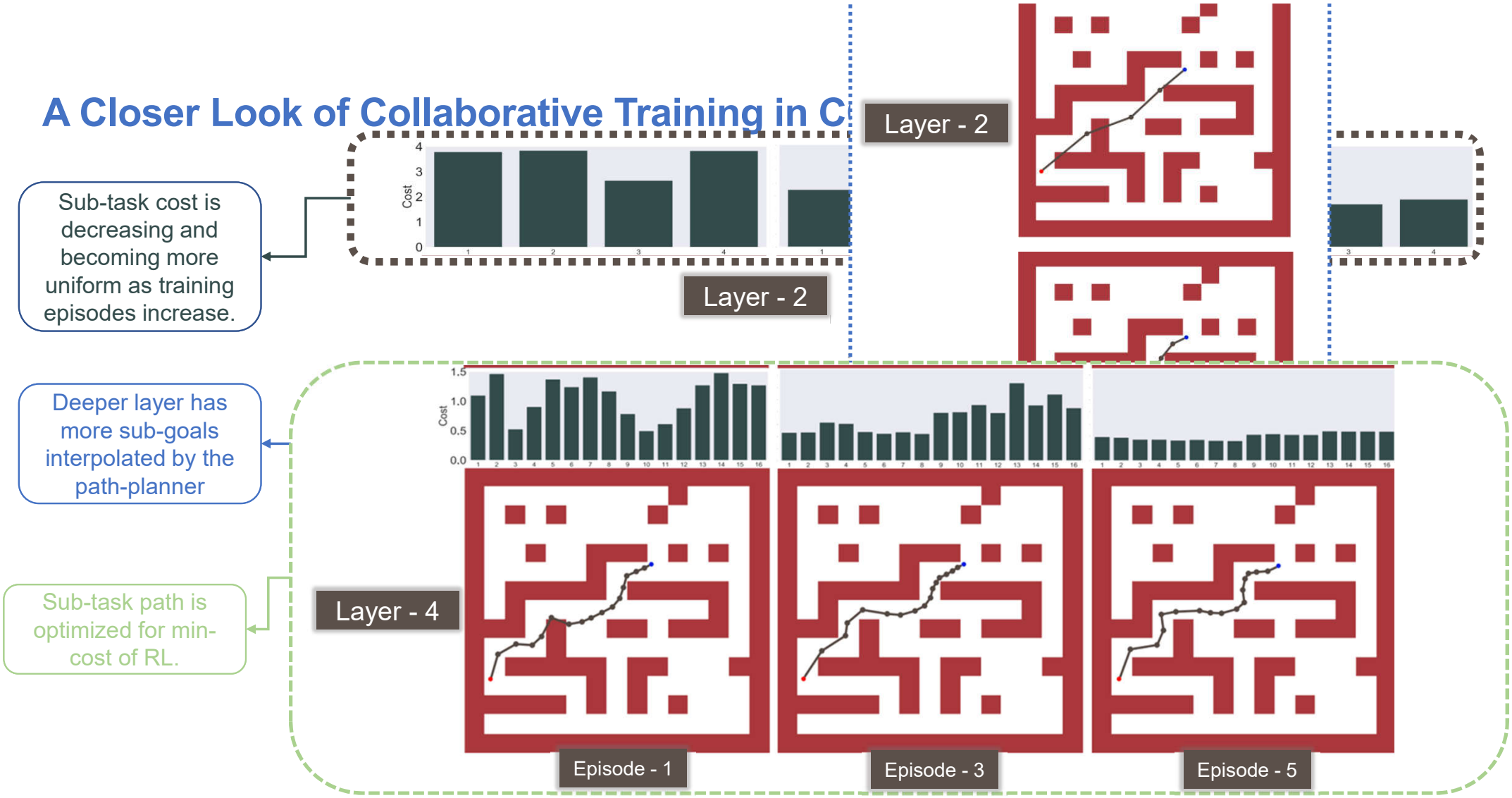
Combine RL with planning: CO-PILOT, SoRB (2019)

RL: HRL (2020), SAC (2018)

Experiment: BipedalWalker



A Closer Look of Collaborative Training in C



Please visit our poster at NeurIPS 2021 for Q/A and Discussion

Poster: 4038

CO-PILOT code: https://github.com/Shuang-AO/CO-PILOT

Shuang Ao
shuang.ao@student.uts.edu.au
Australian Artificial Intelligence Institute,
University of Technology Sydney