Hierarchical Deep Reinforcement Learning: Integrating Temporal Abstraction and Intrinsic Motivation

Tejas D. Kulkarni*
DeepMind, London
tejasdkulkarni@gmail.com

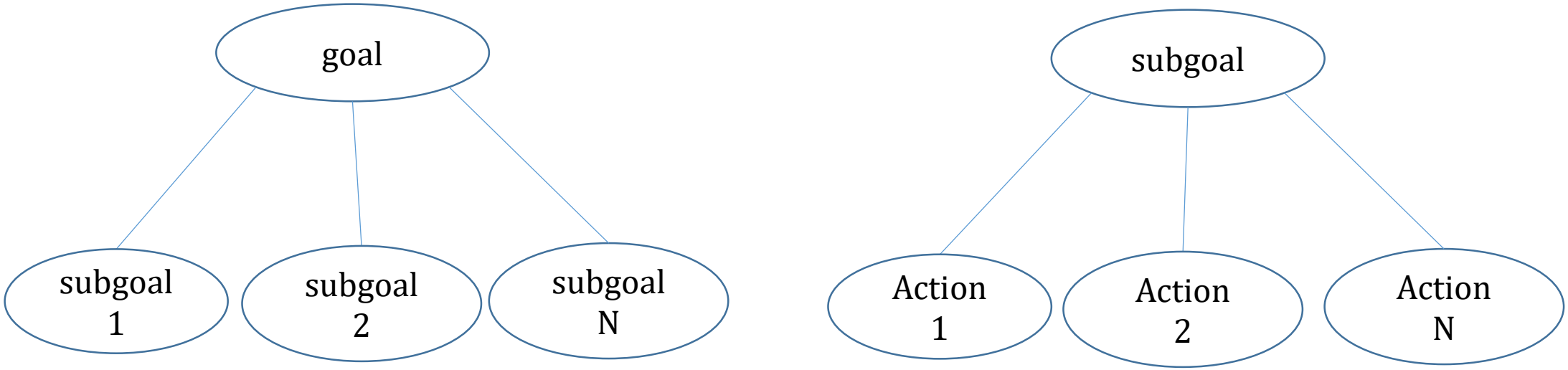
Karthik R. Narasimhan*
CSAIL, MIT
karthikn@mit.edu

Ardavan Saeedi
CSAIL, MIT
ardavans@mit.edu

Joshua B. Tenenbaum
BCS, MIT
jbt@mit.edu

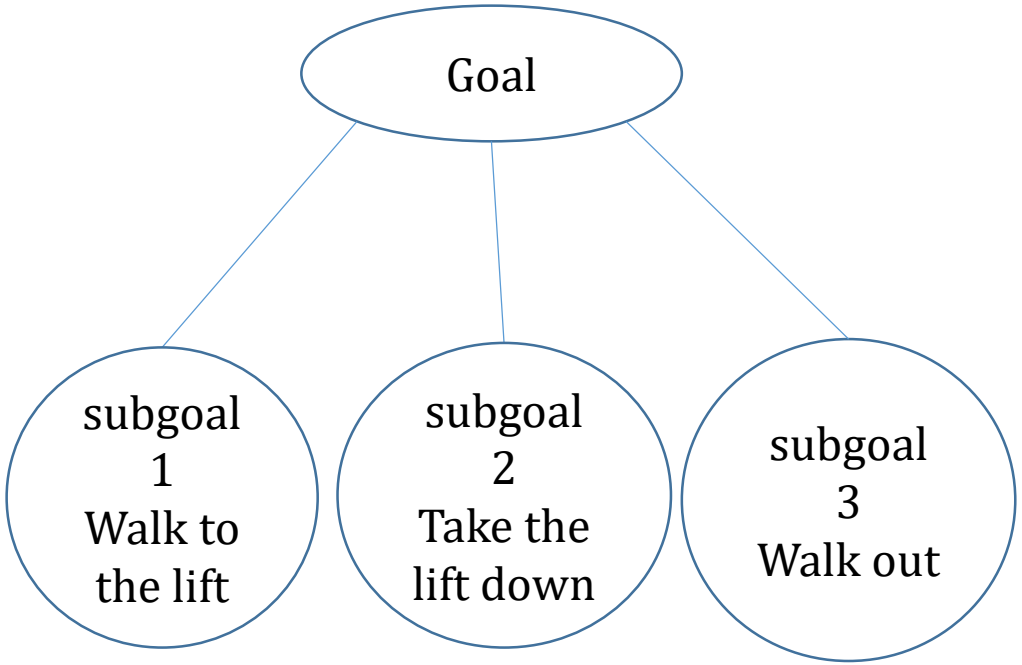
NIPS-2016

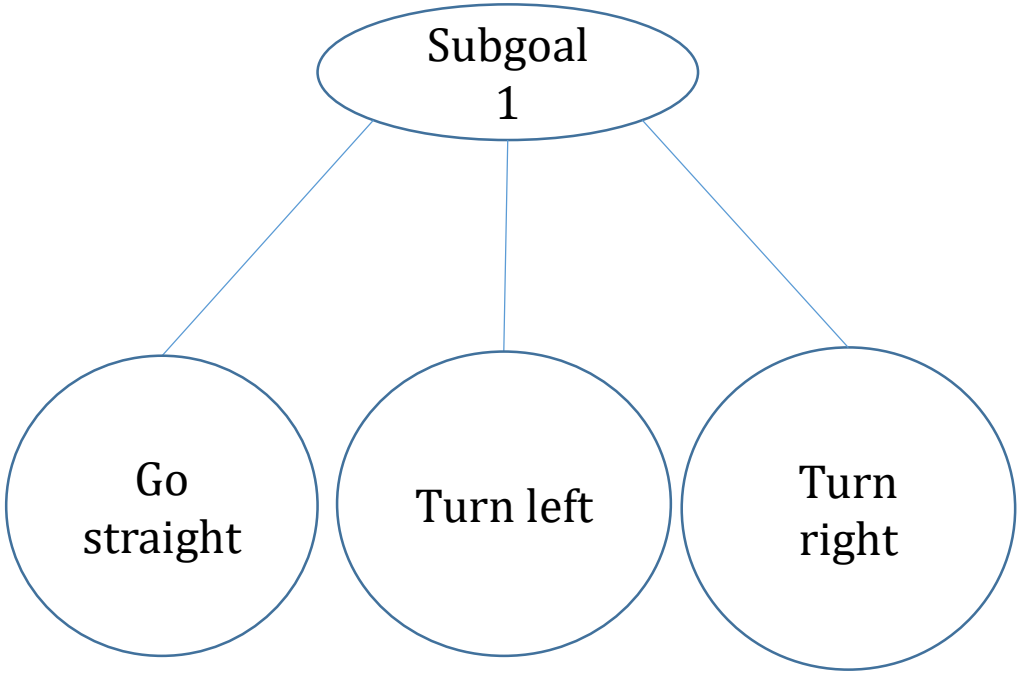
Hierarchical RL



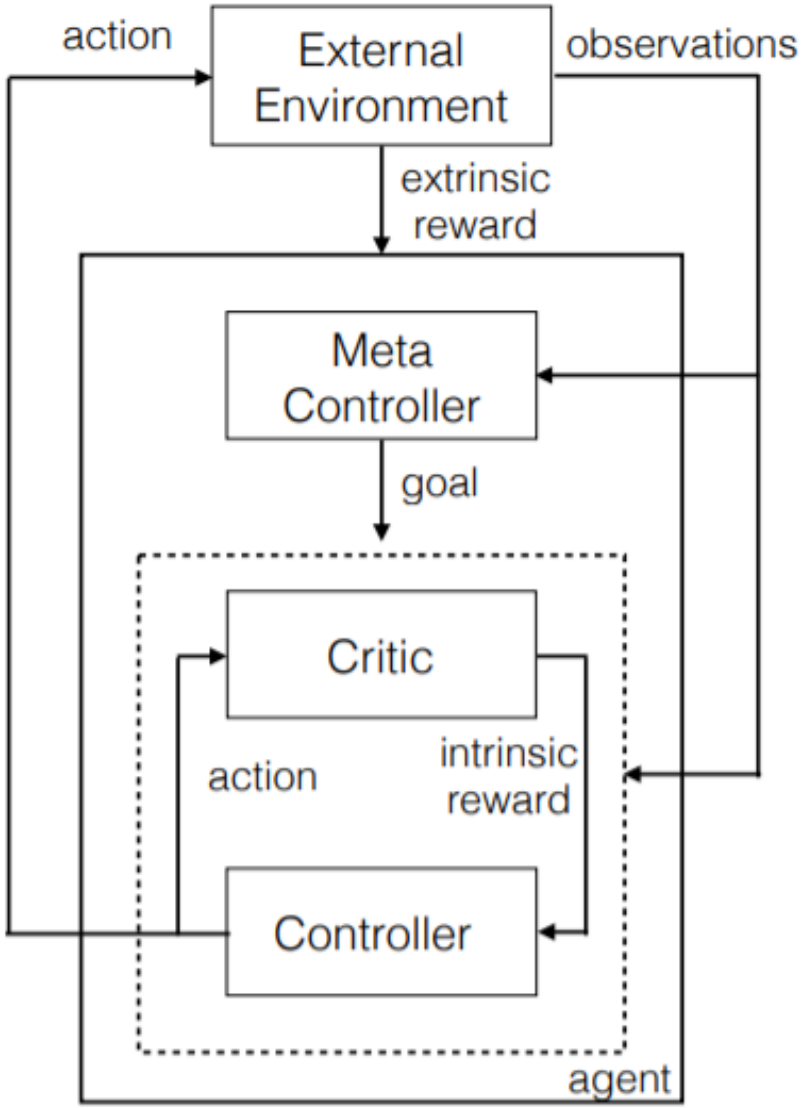
Example

- Goal: walk out of a building





Framework



- Meta controller: choose subgoals
- Controller: achieve subgoals

Maximize two total reward:

$$R_t(g) = \sum_t^{\infty} \gamma^{t'-t} r_{t'}(g),$$

$r_{t'}(g) = 1$ if goal is reached and 0 otherwise

$$F_t = \sum_{t'}^{\infty} \gamma^{t'-t} f_{t'}$$

Compute $f_{t'}$ using external reward

Algorithm 1 Learning algorithm for h-DQN

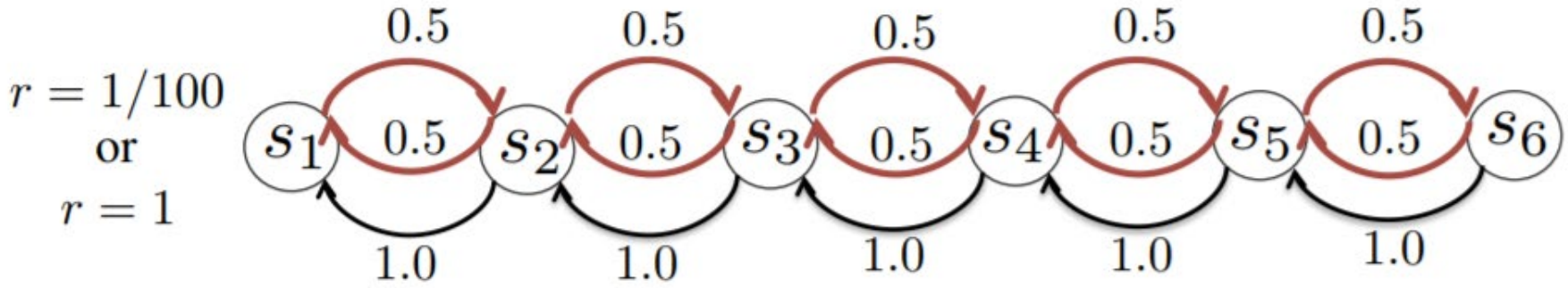
Initialize experience replay memories $\{\mathcal{D}_1, \mathcal{D}_2\}$ and parameters $\{\theta_1, \theta_2\}$ for the controller and meta-controller respectively.
Initialize exploration probability $\epsilon_{1,g} = 1$ for the controller for all goals g and $\epsilon_2 = 1$ for the meta-controller.
for $i = 1, num\_episodes$ **do**
    Initialize game and get start state description s
    $g \leftarrow \text{EPSGREEDY}(s, \mathcal{G}, \epsilon_2, Q_2)$
    while s is **not** terminal **do**
        $F \leftarrow 0$
        $s_0 \leftarrow s$
        while not (s is terminal **or** goal g reached) **do**
            $a \leftarrow \text{EPSGREEDY}(\{s, g\}, \mathcal{A}, \epsilon_{1,g}, Q_1)$
            Execute a and obtain next state s' and extrinsic reward f from environment
            Obtain intrinsic reward $r(s, a, s')$ from internal critic
            Store transition $(\{s, g\}, a, r, \{s', g\})$ in $\mathcal{D}_1$
            $\text{UPDATEPARAMS}(\mathcal{L}_1(\theta_{1,i}), \mathcal{D}_1)$
            $\text{UPDATEPARAMS}(\mathcal{L}_2(\theta_{2,i}), \mathcal{D}_2)$
            $F \leftarrow F + f$
            $s \leftarrow s'$
        end while
        Store transition (s_0, g, F, s') in $\mathcal{D}_2$
        if s is **not** terminal **then**
            $g \leftarrow \text{EPSGREEDY}(s, \mathcal{G}, \epsilon_2, Q_2)$
        end if
    end while
    Anneal ϵ_2 and ϵ_1.
end for

Choose gubgoal

Finish subgoal

update network

Experiment



Initial state: s_2, terminal state: s_1
Action: left, right
Reward: 1 if s_6 is visited or $1/100$ if s_6 not visited.
Subgoal: s_4, s_5, s_6

Experiment

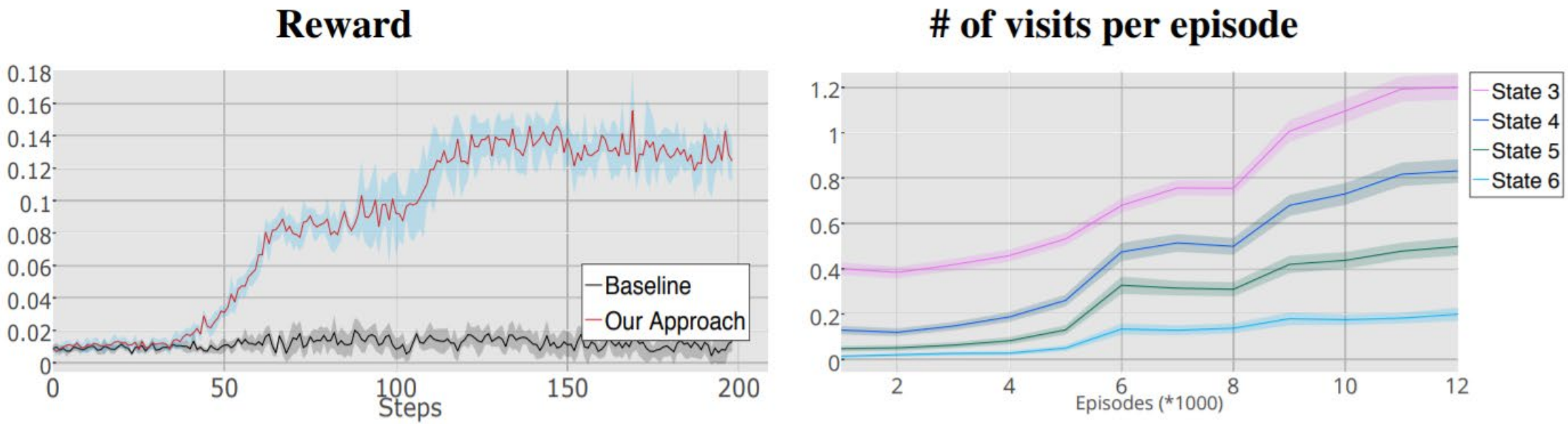
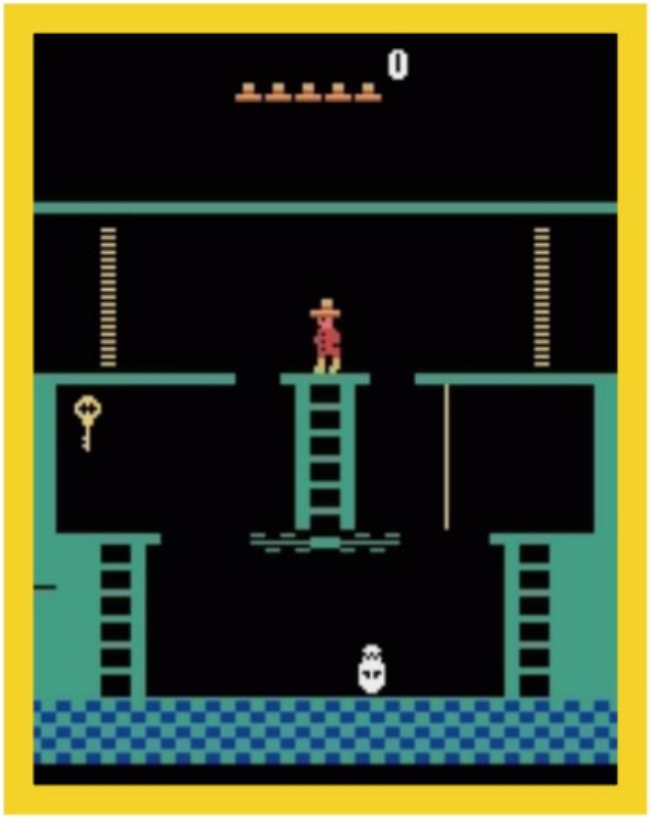


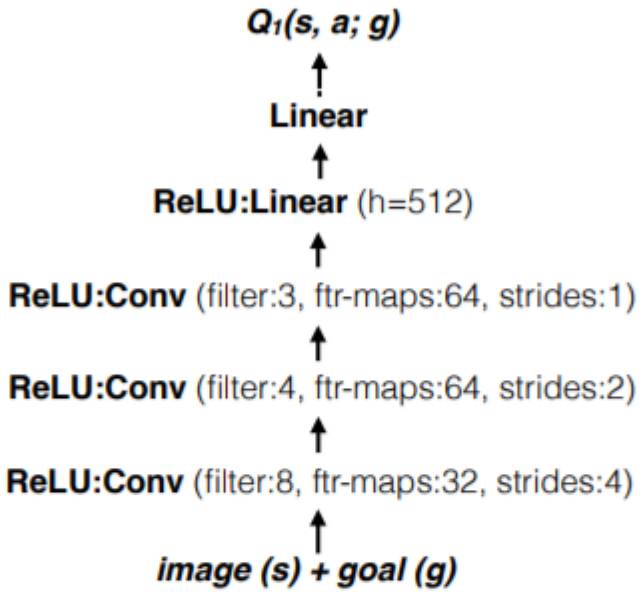
Figure 3: **(left)** Average reward (over 10 runs) of our approach compared to Q-learning. **(right)** #visits of our approach to states s_3-s_6 (over 1000 episodes). Initial state: s_2, Terminal state: s_1.

Experiment

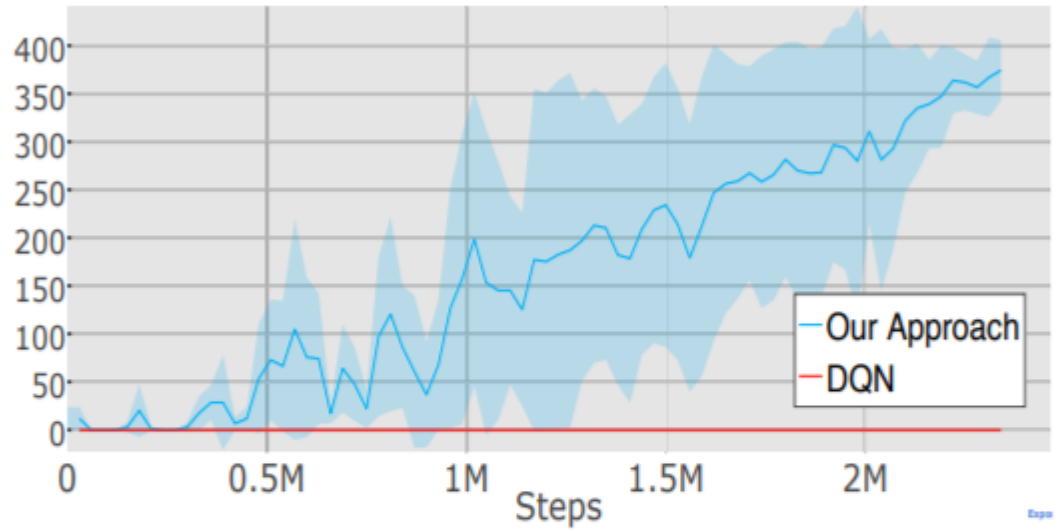


Goal: find the key and use it to open the door
Reward: 100 if key is found and another 300 if door is opened
Subgoals: top-left door; top-right door; middle-ladder; bottom-left ladder; bottom-right ladder; key

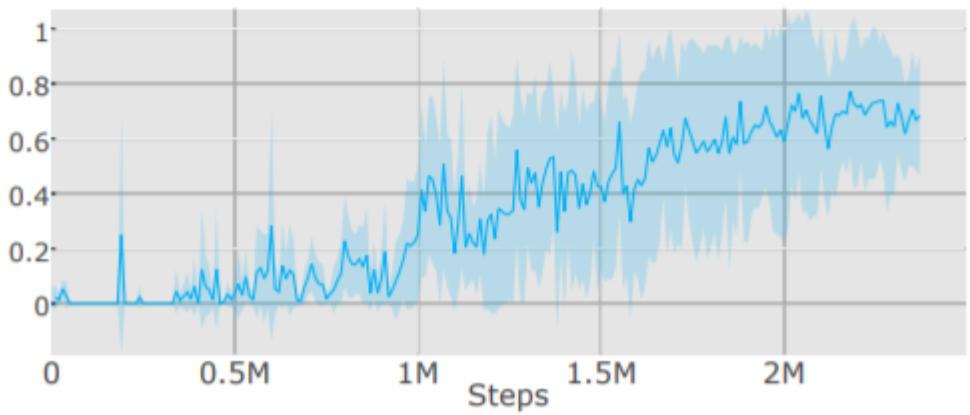
Architecture



Total extrinsic reward



Success ratio for reaching the goal 'key'



Success % of different goals over time