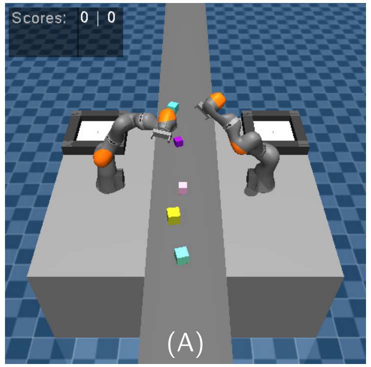
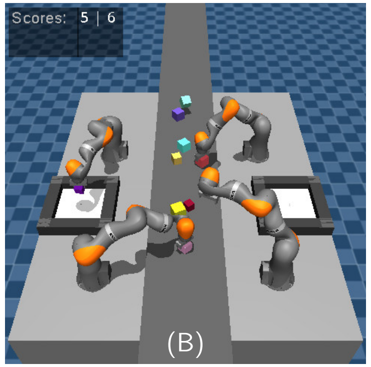
# Factory Manipulation with Cooperative Multi-agent RL

**Nikolas Kirschstein** & **Kassian Köck** (Team 7)

Technical University of Munich




## Big Picture

**goal:** multiple robot arms *cooperating* to *maximise efficiency* in factory manipulation task like pick-and-place
**issue:** conventional pre-programming too *inflexible* and *tedious*
**conjecture:** *multi-agent RL* may find near-optimal behaviour

## Given: The Environment

- even number of robot arms (8 DOF each)
- basket in reach for each arm
- conveyor belt with increasing speed transporting cubes
- score = number of cubes in baskets
- episode termination if either:
  - arm hits the environment (incl. other arms)
  - cube is missed by all arms

## Given: IK Base Policy

- state machine working on *one target object at a time*
- different base policies *ignore each others'* target objects
- control calculated via *inverse kinematics (IK)*

## (A) Full Joint Control: Setting

**action space:** $[-1, 1]$ per joint and learnt arm to control joint state
**observation space:** joint and cube states

learning choices per arm:

- *from-scratch:* learn entire joint control just with RL (hard)
- *delta:* learn only deviation from IK base policy (simplification)
- *base*: execute IK base policy (baseline)

## (A) Full Joint Control: Reward Shaping

- problem: learning hard due to discrete, *highly sparse* reward
  $\Longrightarrow$ denser reward function needed
- desirable incentives:
  $I_0$: reward *new cubes* put into bucket
  $I_1$: reward approach of *gripper* to closest cube
  $I_2$: reward approach of closest cube to *bucket*
- reward function: $r = r_0 + \omega_0 I_0 + \omega_1 I_1 + \omega_2 I_2$
  (base reward $r_0$ prevents learning to terminate episode)

## (A) Full Joint Control: Results

for 2 robot arms (simplest case):

- pure sparse reward: absolutely no learning, *random behaviour*
- progress-based reward ($r_0 = 0.4, \omega_0 = 1, \omega_1 = 0.2, \omega_2 = 0.4$)
  - successful *collision avoidance* due to implicit survival reward
  - BUT: *no gripping* at all (see screenshot (A) and videos)

test results averaged over 100 episodes

| learning choice | avg episode score | avg episode length |
|---|---|---|
| *from-scratch* | $(0, 0)$ | $\mathbf{219.9}$ |
| *delta* | $(0, 0)$ | $\mathbf{220.5}$ |
| *base* | $(\mathbf{1.65}, \mathbf{1.17})$ | $208.8$ |

## (B) IK Toggling Control: Setting

**action space:** $\{\text{ON}, \text{OFF}\}$ per arm to toggle use of IK base policy
**observation space:** joint and cube states + *proposed IK control*

behaviour choices for case OFF:

- *pause:* freeze at the current position
- *retreat:* return to a safe default position
- *base:* continue executing IK policy

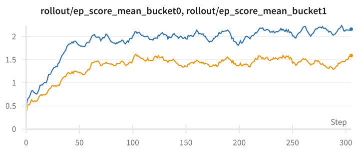
## (B) IK Toggling Control: Results

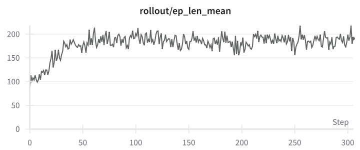
for case of 4 robot arms:

- *pure sparse score-based reward* suffices for effective learning!
- successful *dodging and gripping*! (see screenshot (B) and videos)
- pausing strategy manages to *improve upon baseline*!

test results averaged over 100 episodes

| behaviour choice | avg episode score | avg episode length |
|---|---|---|
| *pause* | $(\mathbf{2.01}, \mathbf{1.5})$ | $173.04$ |
| *retreat* | $(0, 2.12)$ | $\mathbf{216.74}$ |
| *base* | $(1.18, 1.16)$ | $119.74$ |




## Future Work

- relax coordination-only setting to deciding *which cube to grab*
- introduce *auxiliary tasks* (grip, carry, release) & *curriculum learning*
  $\Longrightarrow$ requires careful design and much more computing resources