Documentation Part 1 and 2

NOTE:
We changed in the environment file named: \sim_ur5\mujoco_env\sim_env.py
We added line 238:

```
@property
def object_manager(self):
    return self.object_manager
```

Such that we can access a private method of the object_manager In line 49 in relay_race_sim.py:
start_position = env.object_manager.get_all_block_positions()[0]

This is the only file we change for part 1 and 2 in the file we get from the staff on gitHub

PART 1: Simulation

Stack Function (sim)

The create_cube_stack function organizes a set of cubes into a stable vertical stack at a specified target location within the workspace of a UR5e robotic arm in a Mujoco simulation.

Parameters

- cube_positions (list): A list of [x, y, z] coordinates representing the initial positions of the cubes.
- target_location (list): [x, y, z] coordinates where the stack should be built.

Process

1. **Validation of the Target Location**
   Ensures target_location is within the predefined workspace limits.
2. **Stacking Process**
   - Iterates through each cube position and sequentially picks them up.
   - Moves the cube to the target location, stacking each one at an incremented height.
3. **Pickup and Placement**
   - The robotic arm (ur5e_2) moves above each cube, picks it up, and moves it to the target location.
   - The cube is placed precisely to ensure stability.
   - A brief wait period allows for stabilization before placing the next cube.

Relay Race(sim)

The relay_race_transfer function manages the transfer of a cube between two UR5e robotic arms in a Mujoco simulation environment. The robot ur5e_2 picks up the cube from a starting position and hands it over to ur5e_1 at a designated transfer position while ensuring workspace constraints are respected.

Parameters

- start_position (list): Coordinates [x, y, z] of the cube's initial pickup location.
- handover_position (list): Coordinates [x, y, z] where the handover occurs between the two robots.

Process

1. **Validation of the Handover Position**
   Ensures handover_position is within the defined workspace limits.
2. **Pickup by ur5e_2**
   The ur5e_2 robot moves above the cube and picks it up.
3. **Transport to the Handover Position**
   ur5e_2 carries the cube to handover_position.
4. **Preparation of ur5e_1 for Reception**
   ur5e_1 moves to handover_position and adjusts its orientation for grasping the cube.
5. **Handover and Adjustments**
   ur5e_2 slightly lowers the cube to ensure a smooth transfer.

PART 2 : Lab implementation

Stack Function (lab)

The create_cube_stack function organizes a set of cubes into a stable vertical stack at a specified target location using a real UR5e robotic arm in a laboratory setting. Unlike the simulation-based version, this function directly interacts with real hardware, making precise execution crucial for stability and safety.

Parameters

- cube_positions (list): A list of [x, y, z] coordinates representing the initial positions of the cubes.
- target_location (list): [x, y, z] coordinates where the stack should be built.

Process

1. **Validation of the Target Location**
   Ensures target_location is within the predefined workspace limits.
2. **Homing the Robot**
   Moves the robotic arm to a home position before starting the stacking process.
3. **Stacking Process**
   - Iterates through each cube, picking it up from its initial position.
   - Moves the cube to target_location, stacking each one at an incremented height.
4. **Pickup and Placement**
   - The robotic arm picks up the cube at a predefined safe height.
   - Moves it to the target stack position while ensuring precise alignment.
   - Places the cube carefully to maintain stack stability.

Key Differences from the Simulation Version

The key differences between the simulation version and the real-world lab version of the robotic system are as follows: In terms of environment, the simulation version operates within the Mujoco simulation, whereas the lab version functions in a real-world robotic system.
Regarding motion execution, the simulation version relies on the plan_and_move_to_xyz_facing_down function, while the real robot version uses plan_and_move_to_xyzrz, allowing for more precise control.
For robot control, the simulation version utilizes a simulated motion executor, whereas the lab version employs the ManipulationController for direct hardware control.
When it comes to error handling, the simulation version manages errors through retry logic, while the real-world system requires highly precise execution to prevent instability.
Lastly, in terms of speed and acceleration, the simulation version uses default values, whereas the real-world setup explicitly sets the speed to 2.0 and acceleration to 0.5 for optimal performance

Relay Race (Lab)

The relay_race_transfer function manages the transfer of a cube between two UR5e robotic arms in a real-world laboratory setup. The first robot (ur5e_2) picks up the cube from the starting position, hands it over to the second robot (ur5e_1), which then places it at the final destination (end_position). This function ensures smooth and precise coordination between the two robots, leveraging real-world transformations and movement constraints.

Parameters

- start_position (list): [x, y, z] coordinates of the cube's initial pickup location.
- handover_position (list): [x, y, z] coordinates where the handover occurs between the two robots.
- end_position (list): [x, y, z] coordinates where the cube is finally placed after the transfer.

Process

1. **Homing the Robots**
   Both ur5e_1 and ur5e_2 move to their home positions to ensure a standardized starting point.
2. **Pickup by ur5e_2**
   - ur5e_2 picks up the cube from start_position.
   - Moves to an intermediate position before initiating the handover.
3. **Handover to ur5e_1**
   - Converts handover_position from world coordinates to ur5e_1's frame using point_world_to_robot().
   - Moves ur5e_1 to the handover position.
   - Adjusts ur5e_1's orientation by rotating its end-effector.
4. **Cube Release and Re-Grasping**
   - ur5e_1 moves slightly before grasping to ensure alignment.
   - ur5e_2 releases the cube, and ur5e_1 grasps it.
5. **Final Placement**
   - ur5e_1 moves to end_position and places the cube down.
6. **Returning to Home Position**
   - Both robots return to their home positions after a successful transfer.

Key Differences from the Simulation Version

The key differences between the simulation version and the real-world lab version of the robotic system are as follows:
In terms of environment, the simulation version operates within the Mujoco simulation, while the lab version functions in a real-world robotic system.
For motion execution, the simulation version utilizes the plan_and_move_to_xyz_facing_down function, whereas the real-world version relies on plan_and_move_to_xyzrz for more precise movement.
Regarding robot control, the simulation version uses a simulated motion executor, while the real-world implementation employs the ManipulationController for direct hardware execution.
In coordinate handling, the simulation version directly uses Mujoco coordinates, whereas the real-world system converts them using point_world_to_robot() to account for real-world transformations.
For error handling, the simulation version relies on retry logic, whereas the real-world implementation requires precise execution to account for physical constraints.
Finally, in terms of final placement, the simulation version does not include this step, whereas in the real-world version, the robot places the cube at the specified end_position.