Search

INTRODUCTION

Installation

Multi-goal API

ENVIRONMENTS

Fetch

Shadow Dexterous Hand

Maze

Adroit Hand

Franka Kitchen

Franka Kitchen

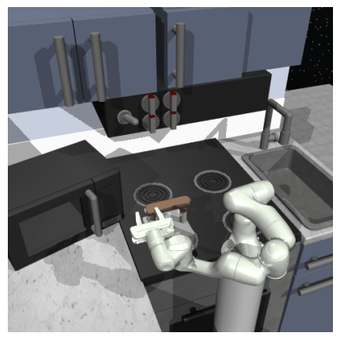
MaMuJoCo (Multi-Agent MuJoCo)

DEVELOPMENT

Github

Release Notes

Franka Kitchen



Description

This environment was introduced in “Relay policy learning: Solving long-horizon tasks via imitation and reinforcement learning” by Abhishek Gupta, Vikash Kumar, Corey Lynch, Sergey Levine, Karol Hausman.

The environment is based on the 9 degrees of freedom Franka robot. The Franka robot is placed in a kitchen environment containing several common household items: a microwave, a kettle, an overhead light, cabinets, and an oven. The environment is a `multitask` goal in which the robot has to interact with the previously mentioned items in order to reach a desired goal configuration. For example, one such state is to have the microwave and sliding cabinet door open with the kettle on the top burner and the overhead light on. The goal tasks can be configured when the environment is created.

Goal

The goal has a multitask configuration. The multiple tasks to be completed in an episode can be set by passing a list of tasks to the argument `tasks_to_complete`. For example, to open the microwave door and move the kettle create the environment as follows:

```
import gymnasium as gym
env = gym.make('FrankaKitchen-v1', tasks_to_complete=['microwave', 'kettle'])
```

The following is a table with all the possible tasks and their respective joint goal values:

Task	Description	Joint Type	Goal
“bottom burner”	Turn the oven knob that activates the bottom burner	slide	[-0.88, -0.01]
“top burner”	Turn the oven knob that activates the top burner	slide	[-0.92, -0.01]
“light switch”	Turn on the light switch	slide	[-0.69, -0.05]
“slide cabinet”	Open the slide cabinet	slide	0.37
“hinge cabinet”	Open the left hinge cabinet	hinge	[0.0, 1.45]
“microwave”	Open the microwave door	hinge	0.37
“kettle”	Move the kettle to the top left burner	free	[-0.23, 0.75, 1.62, 0.99, 0., 0., -0.06]

Action Space

The default joint actuators in the Franka MuJoCo model are position controlled. However, the action space of the environment are joint velocities clipped between -1 and 1 rad/s. The space is a `Box(-1.0, 1.0, (9,), float32)`. The desired joint position control input is estimated in each time step with the current joint position values and the desired velocity action:

Num	Action	Action Min	Action Max	Joint	Unit
0	`robot:panda0_joint1` angular velocity	-1	1	hinge	rad/s
1	`robot:panda0_joint2` angular velocity	-1	1	hinge	rad/s
2	`robot:panda0_joint3` angular velocity	-1	1	hinge	rad/s
3	`robot:panda0_joint4` angular velocity	-1	1	hinge	rad/s
4	`robot:panda0_joint5` angular velocity	-1	1	hinge	rad/s
5	`robot:panda0_joint6` angular velocity	-1	1	hinge	rad/s
6	`robot:panda0_joint7` angular velocity	-1	1	hinge	rad/s
7	`robot:r_gripper_finger_joint` linear velocity	-1	1	slide	m/s
8	`robot:l_gripper_finger_joint` linear velocity	-1	1	slide	m/s

Observation Space

The observation is a `goal-aware` observation space. The observation space contains the following keys:

- `observation`: this is a `Box(-inf, inf, shape=(59,), dtype="float64")` space and it is formed by the robot’s joint positions and velocities, as well as the pose and velocities of the kitchen items. An additional uniform noise of range `[-1,1]` is added to the observations. The noise is also scaled by a factor of `robot_noise_ratio` and `object_noise_ratio` given in the environment arguments. The elements of the `observation` array are the following:

Num	Observation	Min	Max	Joint Name (in corresponding XML file)	Joint Type	Unit
0	`robot:panda0_joint1` hinge joint angle value	-Inf	Inf	robot:panda0_joint1	hinge	angle (rad
1	`robot:panda0_joint2` hinge joint angle value	-Inf	Inf	robot:panda0_joint2	hinge	angle (rad
2	`robot:panda0_joint3` hinge joint angle value	-Inf	Inf	robot:panda0_joint3	hinge	angle (rad
3	`robot:panda0_joint4` hinge joint angle value	-Inf	Inf	robot:panda0_joint4	hinge	angle (rad
4	`robot:panda0_joint5` hinge joint angle value	-Inf	Inf	robot:panda0_joint5	hinge	angle (rad
5	`robot:panda0_joint6` hinge joint angle value	-Inf	Inf	robot:panda0_joint6	hinge	angle (rad
6	`robot:panda0_joint7` hinge joint angle value	-Inf	Inf	robot:panda0_joint7	hinge	angle (rad
7	`robot:r_gripper_finger_joint` slide joint translation value	-Inf	Inf	robot:r_gripper_finger_joint	slide	position (r
8	`robot:l_gripper_finger_joint` slide joint translation value	-Inf	Inf	robot:l_gripper_finger_joint	slide	position (r
9	`robot:panda0_joint1` hinge joint angular velocity	-Inf	Inf	robot:panda0_joint1	hinge	angular velocity (rad/s)
10	`robot:panda0_joint2` hinge joint angular velocity	-Inf	Inf	robot:panda0_joint2	hinge	angular velocity (rad/s)
11	`robot:panda0_joint3` hinge joint angular velocity	-Inf	Inf	robot:panda0_joint3	hinge	angular velocity (rad/s)
12	`robot:panda0_joint4` hinge joint angular velocity	-Inf	Inf	robot:panda0_joint4	hinge	angular velocity (rad/s)
13	`robot:panda0_joint5` hinge joint angular velocity	-Inf	Inf	robot:panda0_joint5	hinge	angular velocity (rad/s)
14	`robot:panda0_joint6` hinge joint angular velocity	-Inf	Inf	robot:panda0_joint6	hinge	angular velocity (rad/s)
15	`robot:panda0_joint7` hinge joint angular velocity	-Inf	Inf	robot:panda0_joint7	hinge	angle (rad
16	`robot:r_gripper_finger_joint` slide joint linear velocity	-Inf	Inf	robot:r_gripper_finger_joint	slide	linear velo (m/s)
17	`robot:l_gripper_finger_joint` slide joint linear velocity	-Inf	Inf	robot:l_gripper_finger_joint	slide	linear velo (m/s)
18	Rotation of the knob for the bottom right burner	-Inf	Inf	knob_Joint_1	hinge	angle (rad
19	Joint opening of the bottom right burner	-Inf	Inf	bottom_right_burner	slide	position (r
20	Rotation of the knob for the bottom left burner	-Inf	Inf	knob_Joint_2	hinge	angle (rad
21	Joint opening of the bottom left burner	-Inf	Inf	bottom_left_burner	slide	position (r
22	Rotation of the knob for the top right burner	-Inf	Inf	knob_Joint_3	hinge	angle (rad
23	Joint opening of the top right burner	-Inf	Inf	top_right_burner	slide	position (r
24	Rotation of the knob for the top left burner	-Inf	Inf	knob_Joint_4	hinge	angle (rad
25	Joint opening of the top left burner	-Inf	Inf	top_left_burner	slide	position (r
26	Joint angle value of the overhead light switch	-Inf	Inf	light_switch	slide	position (r
27	Opening of the overhead light joint	-Inf	Inf	light_joint	hinge	angle (rad
28	Translation of the slide cabinet joint	-Inf	Inf	slide_cabinet	slide	position (r
29	Rotation of the joint in the left hinge cabinet	-Inf	Inf	left_hinge_cabinet	hinge	angle (rad
30	Rotation of the joint in the right hinge cabinet	-Inf	Inf	right_hinge_cabinet	hinge	angle (rad
31	Rotation of the joint in the microwave door	-Inf	Inf	microwave	hinge	angle (rad
32	Kettle’s x coordinate	-Inf	Inf	kettle	free	position (r
33	Kettle’s y coordinate	-Inf	Inf	kettle	free	position (r
34	Kettle’s z coordinate	-Inf	Inf	kettle	free	position (r
35	Kettle’s x quaternion rotation	-Inf	Inf	kettle	free	-

Num	Observation	Min	Max	Joint Name (in corresponding XML file)	Joint Type	Unit
36	Kettle’s y quaternion rotation	-Inf	Inf	kettle	free	-
37	Kettle’s z quaternion rotation	-Inf	Inf	kettle	free	-
38	Kettle’s w quaternion rotation	-Inf	Inf	kettle	free	-
39	Bottom right burner knob angular velocity	-Inf	Inf	knob_Joint_1	hinge	angular velocity (rad/s)
40	Opening linear velocity of the bottom right burner	-Inf	Inf	bottom_right_burner	slide	velocity (n
41	Bottom left burner knob angular velocity	-Inf	Inf	knob_Joint_2	hinge	angular velocity (rad/s)
42	Opening linear velocity of the bottom left burner	-Inf	Inf	bottom_left_burner	slide	velocity (n
43	Top right burner knob angular velocity	-Inf	Inf	knob_Joint_3	hinge	angular velocity (rad/s)
44	Opening linear velocity of the top right burner	-Inf	Inf	top_right_burner	slide	velocity (n
45	Top left burner knob angular velocity	-Inf	Inf	knob_Joint_4	hinge	angular velocity (rad/s)
46	Opening linear velocity of the top left burner	-Inf	Inf	top_left_burner	slide	velocity (n
47	Angular velocity of the overhead light switch	-Inf	Inf	light_switch	slide	velocity (n
48	Opening linear velocity of the overhead light	-Inf	Inf	light_joint	hinge	angular velocity (rad/s)
49	Linear velocity of the slide cabinet joint	-Inf	Inf	slide_cabinet	slide	velocity (n
50	Angular velocity of the left hinge cabinet joint	-Inf	Inf	left_hinge_cabinet	hinge	angular velocity (rad/s)
51	Angular velocity of the right hinge cabinet joint	-Inf	Inf	right_hinge_cabinet	hinge	angular velocity (rad/s)
52	Anular velocity of the microwave door joint	-Inf	Inf	microwave	hinge	angular velocity (rad/s)
53	Kettle’s x linear velocity	-Inf	Inf	kettle	free	linear velo (m/s)
54	Kettle’s y linear velocity	-Inf	Inf	kettle	free	linear velo (m/s)
55	Kettle’s z linear velocity	-Inf	Inf	kettle	free	linear velo (m/s)
56	Kettle’s x axis angular rotation	-Inf	Inf	kettle	free	angular velocity(ra
57	Kettle’s y axis angular rotation	-Inf	Inf	kettle	free	angular velocity(ra
58	Kettle’s z axis angular rotation	-Inf	Inf	kettle	free	angular velocity(ra

- `desired_goal`: this key represents the final goal to be achieved. The value is another `Dict` space with keys the tasks to be completed in the episode and values the joint goal configuration of each joint in the task as specified in the `Goal` section.
- `achieved_goal`: this key represents the current state of the tasks. The value is another `Dict` space with keys the tasks to be completed in the episode and values the current joint configuration of each joint in the task.

Info

The environment also returns an `info` dictionary in each Gymnasium step. The keys are:

- `tasks_to_complete` (list[str]): list of tasks that haven’t yet been completed in the current episode.
- `step_task_completions` (list[str]): list of tasks completed in the step taken.
- `episode_task_completions` (list[str]): list of tasks completed during the episode uptil the current step.

Rewards

The environment’s reward is `sparse`. The reward in each Gymnasium step is equal to the number of task completed in the given step. If no task is completed the returned reward will be zero. The tasks are considered completed when their joint configuration is within a norm threshold of `0.3` with respect to the goal configuration specified in the `Goal` section.

Starting State

The simulation starts with all of the joint position actuators of the Franka robot set to zero. The doors of the microwave and cabinets are closed, the burners turned off, and the light switch also off. The kettle

will be placed in the bottom left burner.

Episode End

The episode will be `truncated` when the duration reaches a total of `max_episode_steps` which by default is set to 280 timesteps. The episode is `terminated` when all the tasks have been completed unless the `terminate_on_tasks_completed` argument is set to `False`.

Arguments

The following arguments can be passed when initializing the environment with `gymnasium.make` kwargs:

Parameter	Type	Default	Description
`tasks_to_complete`	**list[str]**	All possible goal tasks. Go to Goal section	The goal tasks to reach in each episode
`terminate_on_tasks_completed`	**bool**	`True`	Terminate episode if no more tasks to complete (episodic multitask)
`remove_task_when_completed`	**bool**	`True`	Remove the completed tasks from the info dictionary returned after each step
`object_noise_ratio`	**float**	`0.0005`	Scaling factor applied to the uniform noise added to the kitchen object observations
`robot_noise_ratio`	**float**	`0.01`	Scaling factor applied to the uniform noise added to the robot joint observations
`max_episode_steps`	**integer**	`280`	Maximum number of steps per episode

Version History

- v1: updated version with most recent python MuJoCo bindings.
- v0: legacy versions in the D4RL.

Copyright © 2023 Farama Foundation