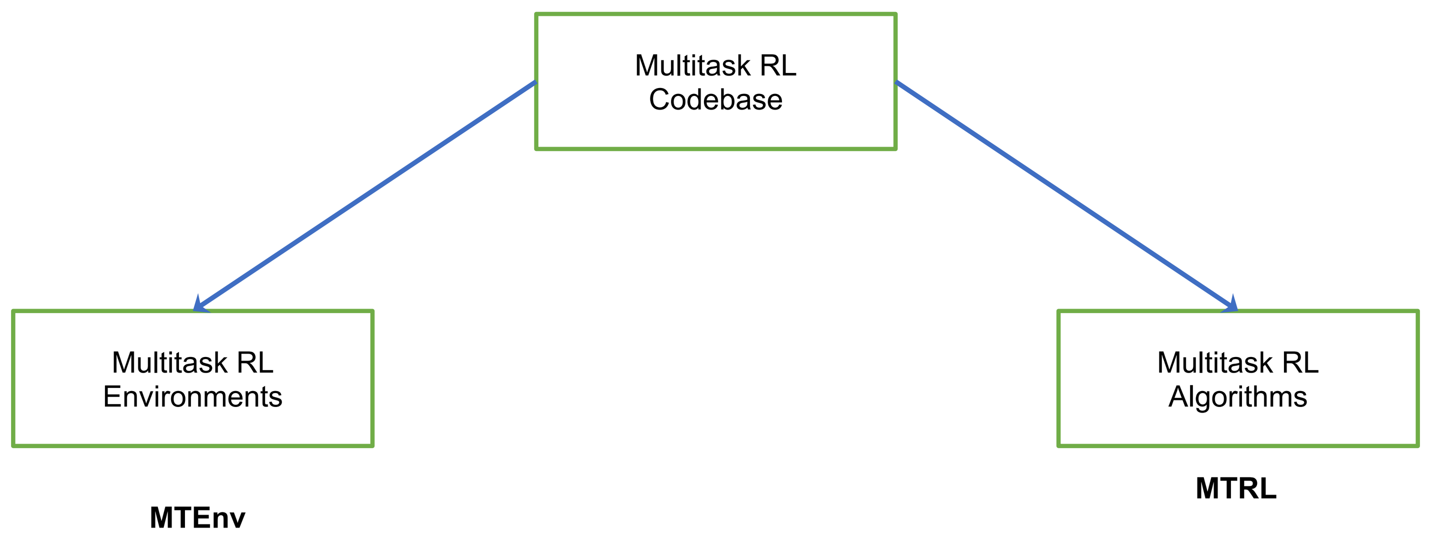
FACEBOOK AI

Environments and Baseline for Multitask Reinforcement Learning

Shagun Sodhani, Amy Zhang, Ludovic Denoyer, Pierre-Alexandra Kamienny, Olivier Delalleau

Facebook AI Research

Motivation: Facilitate research in multitask RL



MTEnv: Standardize multitask RL environments and provide better benchmarks

Extend the OpenAI Gym[1] interface with first-class support for multi-task RL.

```
obs = env.reset()
print(obs)
# {'env_obs': array([-0.03265039,  0.51487777,  0.2368754 , -0.06968209,  0.6235982 ,
#     0.01492813,  0.        ,  0.        ,  0.        ,  0.03933976,
#     0.89743189,  0.01492813]), 'task_obs': 1}
action = env.action_space.sample()
print(action)
# array([-0.76422    , -0.15384133,  0.74575615, -0.11724994], dtype=float32)
obs, reward, done, info = env.step(action)
```

Collection of multitask RL environments

```
from mtenv import make
env = make("MT-HiPBMDP-Finger-Spin-vary-size-v0")
env.reset()
```

Wrappers to extend single-task environments for multi-task setup

```
from mtenv.wrappers import (
    NTasks, NTasksId, SampleRandomTask, EnvToMTEnv)
```

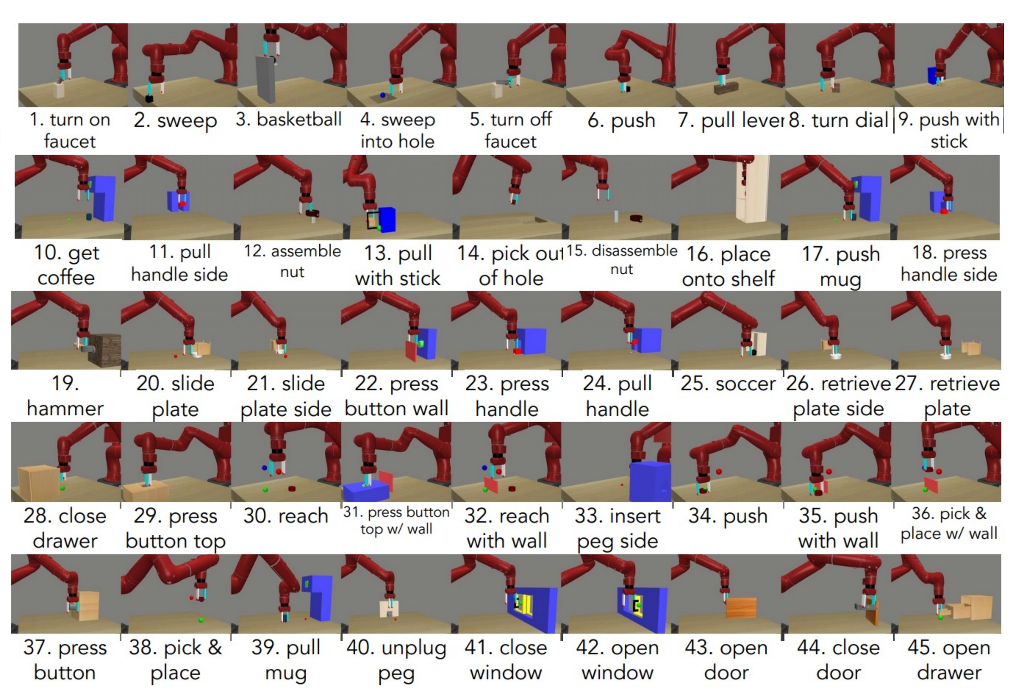
Supported Environments

Environment	Description
Control Tasks	Cartpole, Acrobat environments with varying physical values
HiPBMDP[3]	Environments from DeepMind Control Suite[4] with varying physical values
MetaWorld[2]	50 distinct robotics manipulation tasks
Pixel Mazes[5]	2-D and 3-D mazes

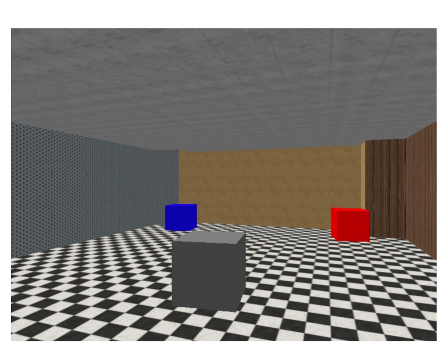
Builds on OpenAI Gym

1. OpenAI Gym[1] offers a standard environment interface for single-task RL and the overhead of switching across environments is lowered. However, Gym is not designed to control the task state and the standardization benefits are lost in the case of Multitask RL.
2. MTEnv extends the OpenAI Gym interface to support multiple task environments.
3. MTEnv has two guiding principles: (i) Make minimal changes to the Gym Interface (which the community is very familiar with) and (ii) Make it easy to port existing environments to MTEnv.

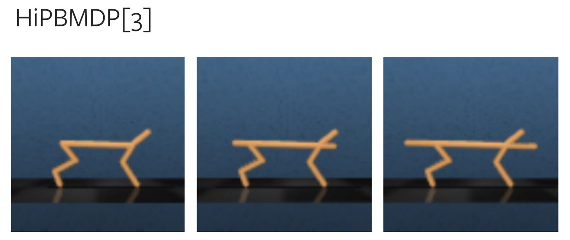
MetaWorld [2]



Pixel Mazes[5]

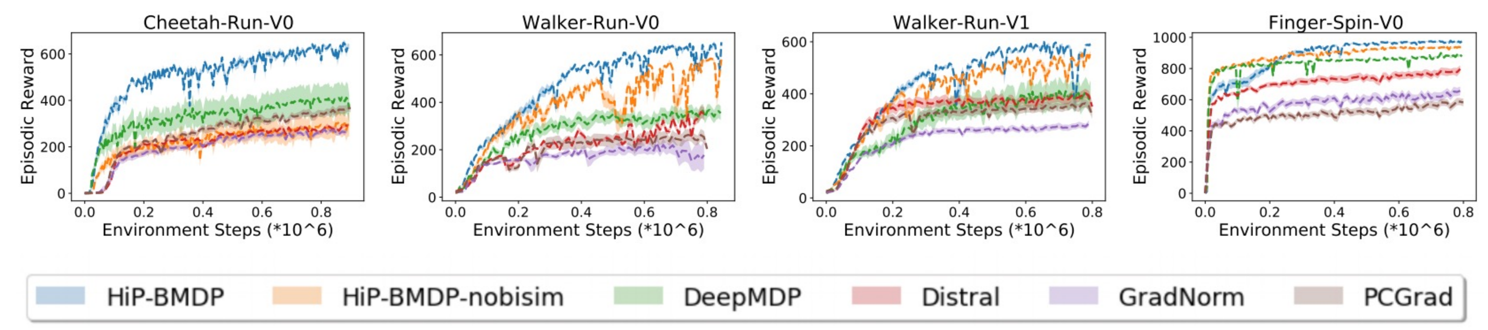


HiPBMDP[3]



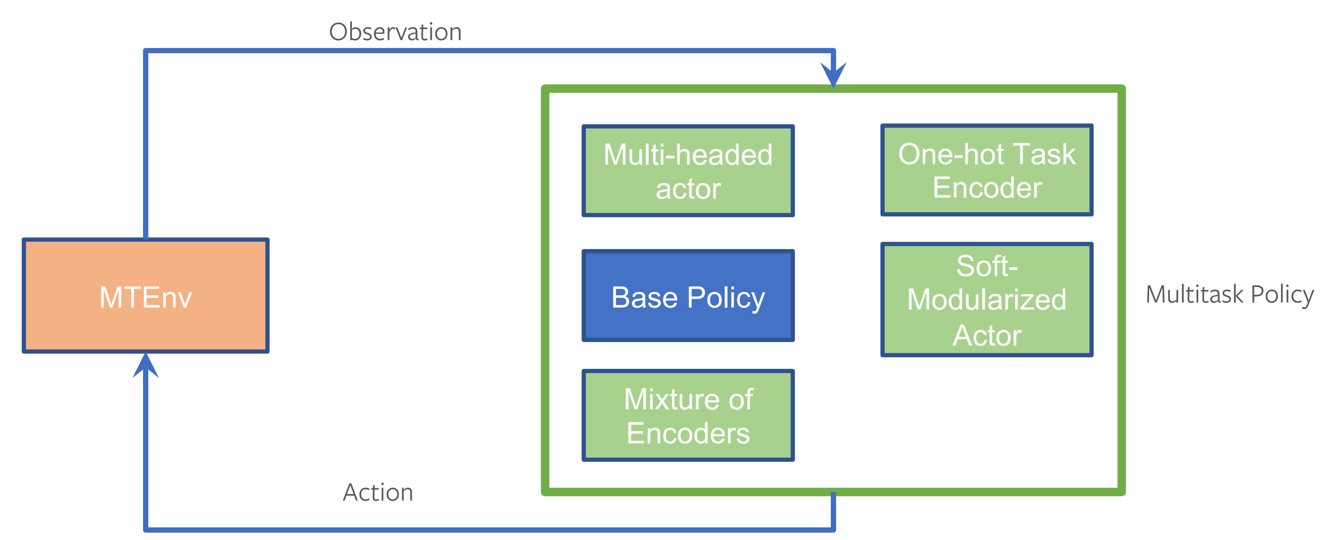
HiPBMDP[3]

MTRL: Baselines for Multitask RL



MTRL Design

1. MTRL has two building blocks: (i) Base (single task) policy and (ii) Components to augment the base policy for multi-task setup.
2. The ideal workflow is to start with a base policy and add multi-task components as they seem fit.
3. The components are *plug and play*, thus giving a lot of freedom and flexibility to the end user.



Supported Components and Algorithms

Multitask RL Components

Components	Description
Multi-headed actor, critic, value function etc	Actor, critic, value functions etc. with task specific heads (output layers)
Task Encoder	One-hot task encoders, context encoders etc
State/observation Encoders	Attention weighted Mixture of Encoders, Gated Mixture of Encoders, Ensemble of Encoders
Modularized actor, critic, value-fuction etc	Actor, critic, value functions etc. which are composed on the fly, based on the task

Single Task Policies

Components	Description
SAC[11]	Soft Actor Critic
SAC-AE[12]	Soft Actor Critic with Auto Encoder
DeepMDP[13]	Continuous Latent Space Models for Representation Learning

Multitask RL Algorithms

Algorithm	Description
Multi-task SAC	SAC with task specific exploration bonus
Multi-task SAC with Task Encoder	Multitask SAC, conditioned on the task representation
HiPBMDP[3]	Learns state abstractions for Hidden-Parameter Block MDPs
Distral[6]	Distill task specific policies into a single, centroid policy
SoftModulaization[7]	Learns a routing network over the RL policy
CARE[8]	Learns contextual, attention-based representations for multitask RL
PCGrad[9]	Gradient Manipulation for multitask learning
GradNorm[10]	Learning weights for different tasks

MTRL in action

Train multi-task SAC on MetaWorld MT10

```
(base) ➜  ~ python main.py setup=metaworld agent=state_sac env=metaworld-mt10 \
            agent.multitask.should_use_disentangled_alpha=True
```

Train multi-head multi-task SAC on MetaWorld MT10

```
(base) ➜  ~ python main.py setup=metaworld agent=state_sac env=metaworld-mt10 \
            agent.multitask.should_use_disentangled_alpha=True \
            agent.multitask.should_use_multi_head_policy=True
```

Directions of Development

Supporting more environments

1. RoboSuite (Simulation Framework and Benchmark for Robot Learning)
2. Issac Gym (Physics simulation environment for RL)
3. MiniTrackmania (GODOT-based racing simulator)
4. Vendee-Globe (racing sailboat simulator)
5. Mvfst-rl (Network simulator for congestion control algorithms)

Supporting more algorithms

1. More base policies: PPO, Impala etc.
2. Context Aware Dynamics Model
3. HyperNetworks
4. Trajectory based context encoders
5. Mixture of Expert based models

Supporting more setups

1. Continual/Lifelong Reinforcement Learning
2. State/action spaces could change across tasks
3. Environments with action conditioned dynamics
4. Language conditioned multi-task RL

Scaling and ease of use

1. Memory-efficient replay buffers
2. Scaling policy components
3. Add examples of complex training pipelines
4. Pre-trained models and weights

Links

MTEnv website: https://github.com/facebookresearch/mtenv

MTRL website: https://github.com/facebookresearch/mtrl

Chat: https://mtenv.zulipchat.com/

References

[1] Brockman, Greg, et al. "Openai gym." arXiv preprint arXiv:1606.01540 (2016).

[2] Yu, Tianhe, et al. "Meta-world: A benchmark and evaluation for multi-task and meta reinforcement learning." Conference on Robot Learning. PMLR, 2020.

[3] Zhang, Amy, et al. "Learning Robust State Abstractions for Hidden-Parameter Block MDPs." International Conference on Learning Representations. 2020.

[4] Tassa, Yuval, et al. "Deepmind control suite." arXiv preprint arXiv:1801.00690 (2018).

[5] Kamienny, Pierre-Alexandre, et al. "Learning adaptive exploration strategies in dynamic environments through informed policy regularization." arXiv preprint arXiv:2005.02934 (2020).

[6] Teh, Yee Whye, et al. "Distral: Robust multitask reinforcement learning." arXiv preprint arXiv:1707.04175 (2017).

[7] Yang, Ruihan, et al. "Multi-task reinforcement learning with soft modularization."Advances in Neural Information Processing Systems 33 (2020).

[8] Sodhani, Shagun, et al. "Multi-Task Reinforcement Learning with Context-based Representations." arXiv preprint arXiv:2102.06177 (2021).

[9] Yu, Tianhe, et al. "Gradient Surgery for Multi-Task Learning." Advances in Neural Information Processing Systems 33 (2020).

[10] Chen, Zhao, et al. "Gradnorm: Gradient normalization for adaptive loss balancing in deep multitask networks." International Conference on Machine Learning. PMLR, 2018.

[11] Haarnoja, Tuomas, et al. "Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor." International Conference on Machine Learning. PMLR, 2018.

[12] Yarats, Denis, et al. "Improving Sample Efficiency in Model-Free Reinforcement Learning from Images." arXiv e-prints (2019): arXiv-1910.

[13] Gelada, Carles, et al. "Deepmdp: Learning continuous latent space models for representation learning." International Conference on Machine Learning. PMLR, 2019.

Take a photo to learn more:

Acknowledgements

We thank Adam Lerer, Amanpreet Singh, Denis Yarats, Jakob Foerster, Joelle Pineau and Omry Yadan for useful discussions and suggestions.