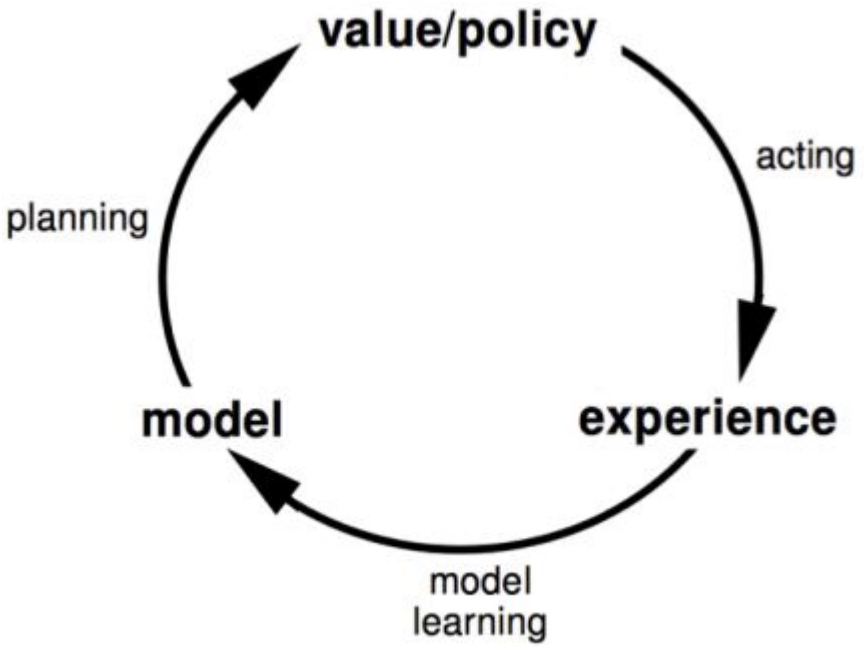
Model Based Reinforcement Learning

B. Ravindran

Model-based RL

- What if we can learn the dynamics of the environment?
- Learn a model of the environment dynamics
- Generate samples using the model.
- Learn and/or plan using those samples.



Model-based RL

Models can be used to improve:

- Sample Efficiency (previous lecture)
- Exploration
- Asymptotic Performance
- Transfer
- Safety and Explainability

Exploration: Challenges

- Redecide on their exploratory decision at every timestep.
- May make an exploratory decision in a state, but decide to undo it at the next timestep. eg. Epsilon-Greedy.
- Model-based RL can be used for potentially better exploration strategies that do not cause such “jitter”.

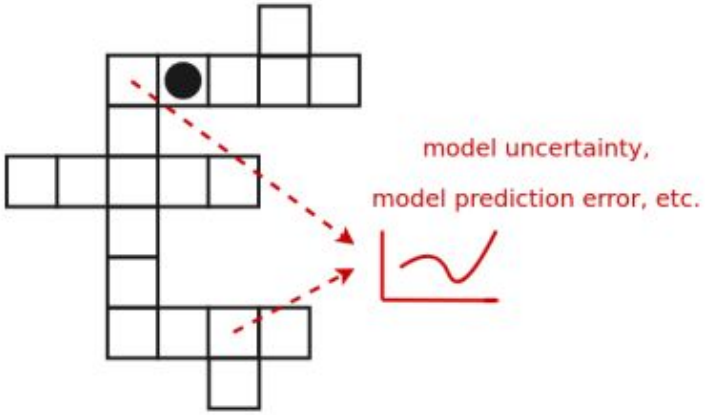
Exploration

Intrinsic motivation: Model-based exploration directed at novel or highly uncertain states.

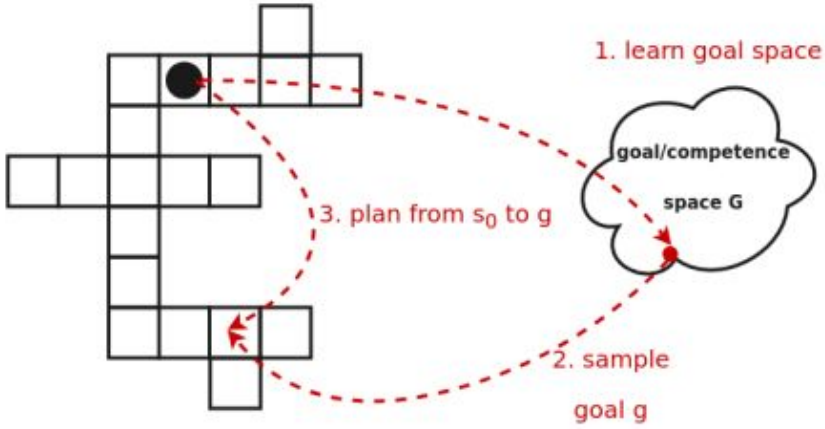
a. Knowledge-based: Every state gets associated with an intrinsic reward based on local characteristics.
b. Competence-based: Identify goal states that capture directions of variation in the domain and try to reach them.

Exploration

Knowledge-based intrinsic motivation



Competence-based intrinsic motivation



Exploration

Knowledge-Based IM:

- Prioritizes states for exploration when they provide new information about the MDP.
- Commonly uses a specific intrinsic reward, which is then propagated together with the extrinsic reward.

$$r_t(s, a, s') = r^e(s, a, s') + \eta \cdot r^i(s, a, s')$$

Exploration

Choice of intrinsic reward:

- Novelty eg. Bayesian Exploration Bonus (Kolter and Ng, 2009).

$$r^i(s, a, s') \propto 1/(1 + n(s, a)),$$

where $n(s, a)$ denotes the number of visits to state-action pair (s, a)

- Recency eg. Dyna-Q+

$$r^i(s, a, s') = \sqrt{l(s, a)},$$

where $l(s, a)$ denotes the number of timesteps since the last trial at (s, a)

- Intrinsic rewards may also help overcome non-stationarity
- A combination of multiple intrinsic rewards may also be used.

Exploration

Competence-based IM:

- Performs exploration based on “learning progress”.
- We may have visited a state often, which would make it uninteresting for knowledge-based IM.
- If we still get better/faster at actually reaching this state, i.e., we still make learning progress.
- Competence-based IM aims to improve learning progress.
- Generates an automatic “curriculum” of tasks, guided by learning progress.

Exploration

- A competence-based IM strategy could be as follows:
  - Learn how to select a set of states with high potential learning progress.
  - Sample from the set eg. sample a state with highest potential for learning progress.
  - Attempt to reach the sampled state .eg. Goal-conditioned Value Functions

- Apart from IM, hierarchical methods can also be used for model-based exploration.

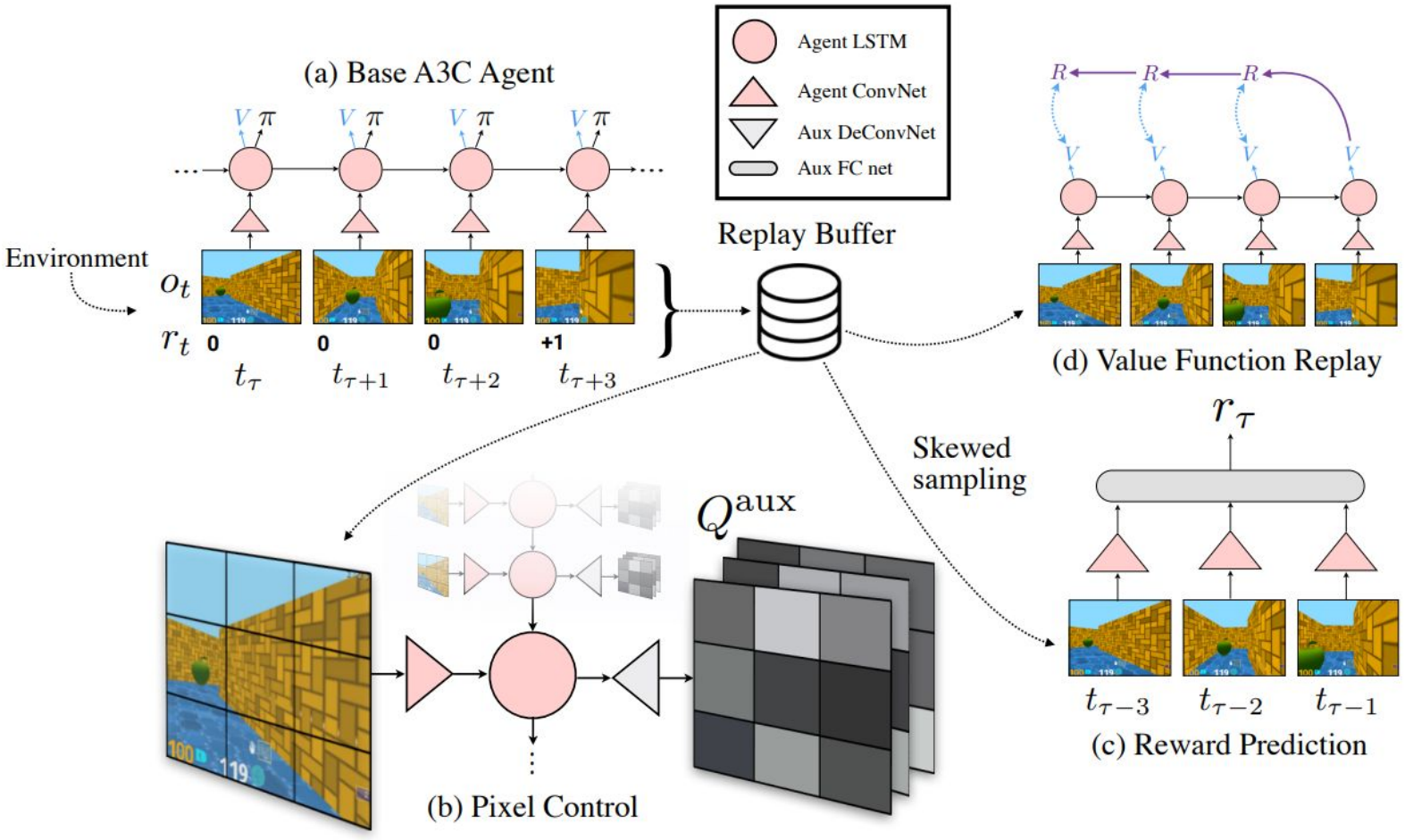
Performance

- It maybe be argued that model-based RL leads to worse asymptotic performance than model-free methods.
- However, empirical success of AlphaZero, MuZero etc. suggests otherwise.
- With a perfect (or good) model, model-based RL may actually lead to better (empirical) asymptotic performance.

Performance

- Can we use models to learn better representations for better performance?
- **UNREAL**: Reinforcement Learning with Unsupervised Auxiliary Tasks:
  - Model prediction used for computing auxiliary objectives.

Performance: UNREAL



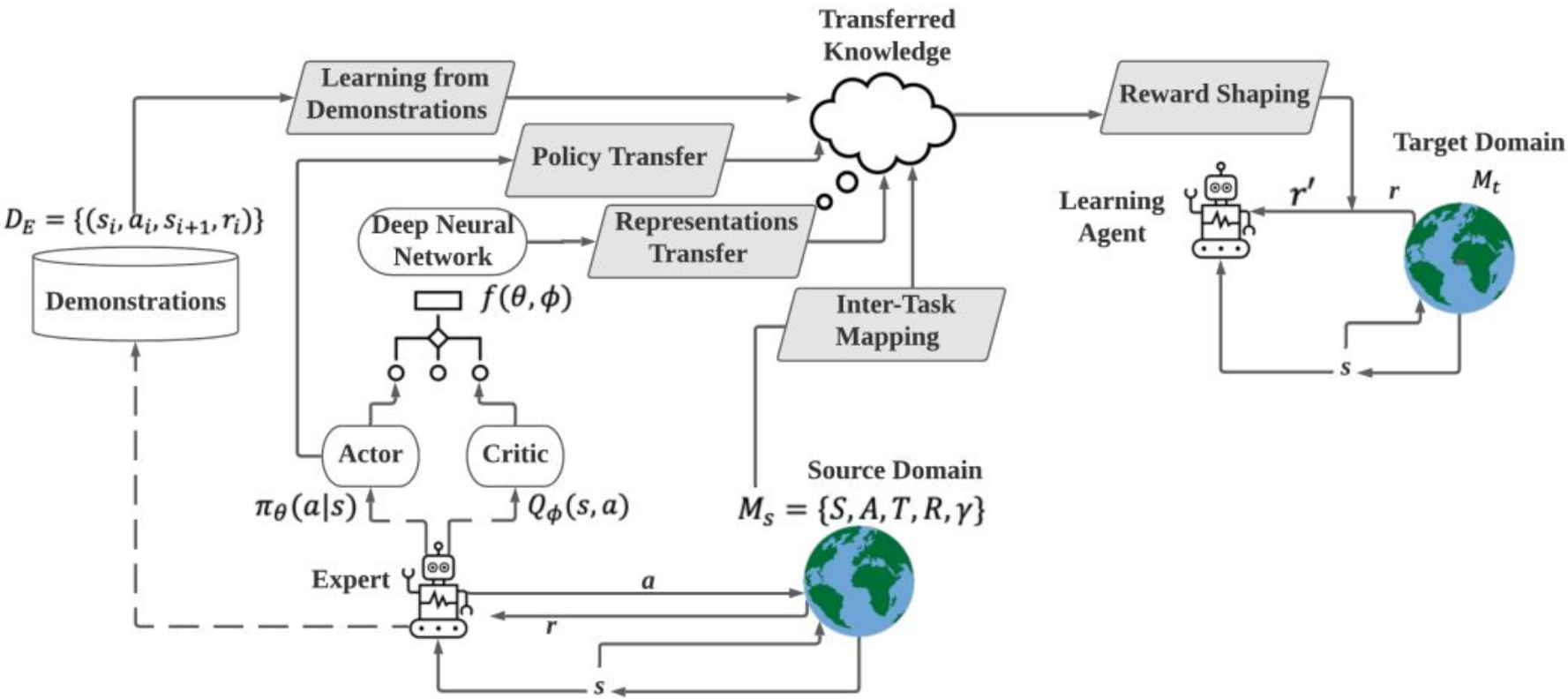
Performance: UNREAL

Components:

- Base A3C Agent: The component that uses A3C's on-policy training.
- Auxiliary Control Tasks: Additional tasks preset by the user giving pseudo-rewards to the agent for specific behavior.
- Auxiliary Reward Tasks: Additional reward prediction tasks that help extract relevant features from the environment.
- Value Function Replay: Additional off-policy training for the value function

Transfer

Transfer learning in RL: Re-use information from a source task to speed-up learning on a new task.



Transfer

Transfer of a dynamics model:

1. Similar dynamics function but different reward function. eg. new level in video game.
2. Slightly changed transition dynamics, eg. transfer from simulation to real-world tasks.

Transfer

Similar dynamics with different reward:

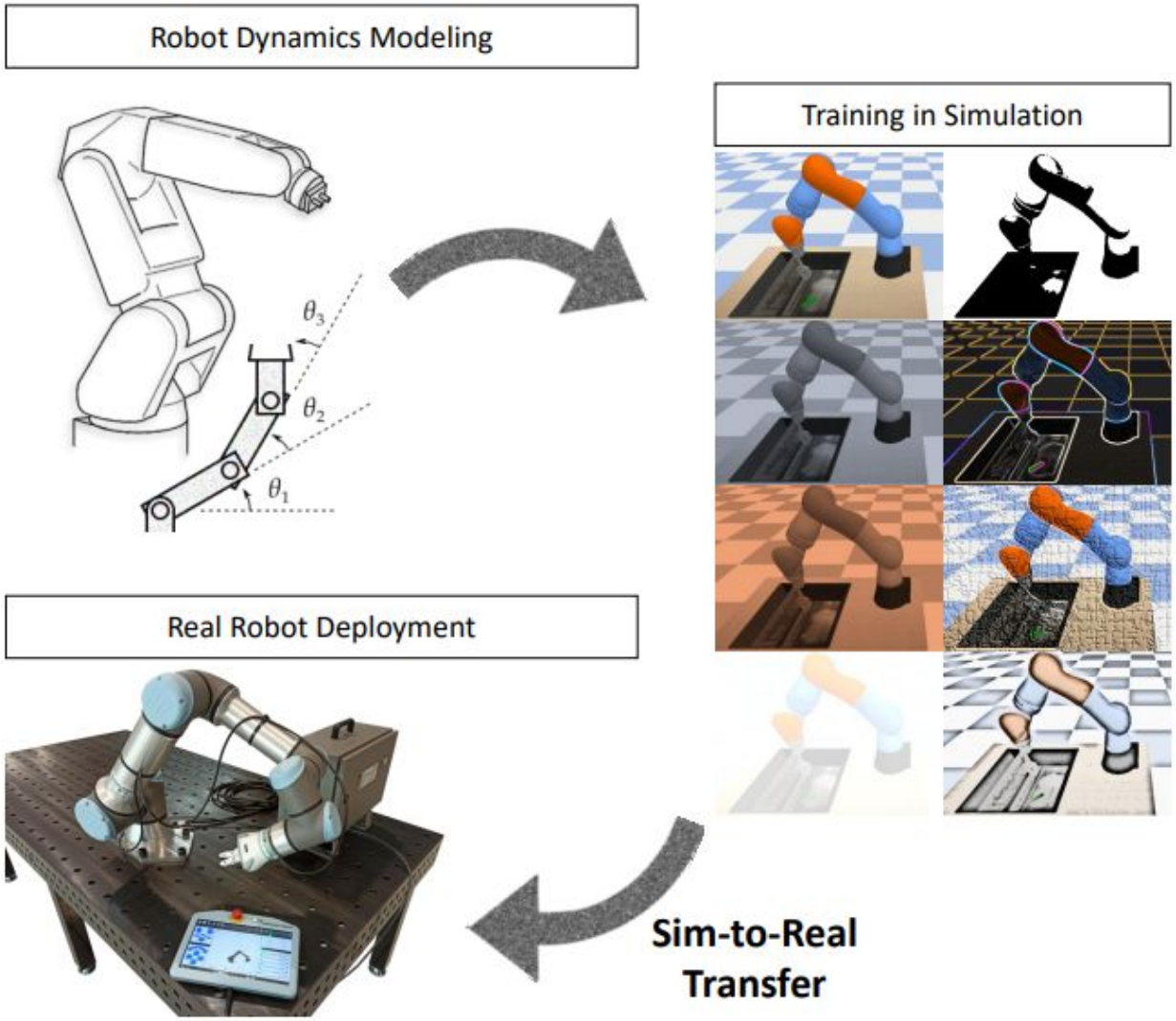
- Can be formulated as a multi-objective RL problem.
- A multi-objective MDP has a single dynamics function but multiple reward functions.
- These rewards can be combined in different ways, each of which lead to a new task specification.
- Successor representations: Another method for changing reward functions.
- Summarize the model in the form of future state occupancy statistics

Transfer

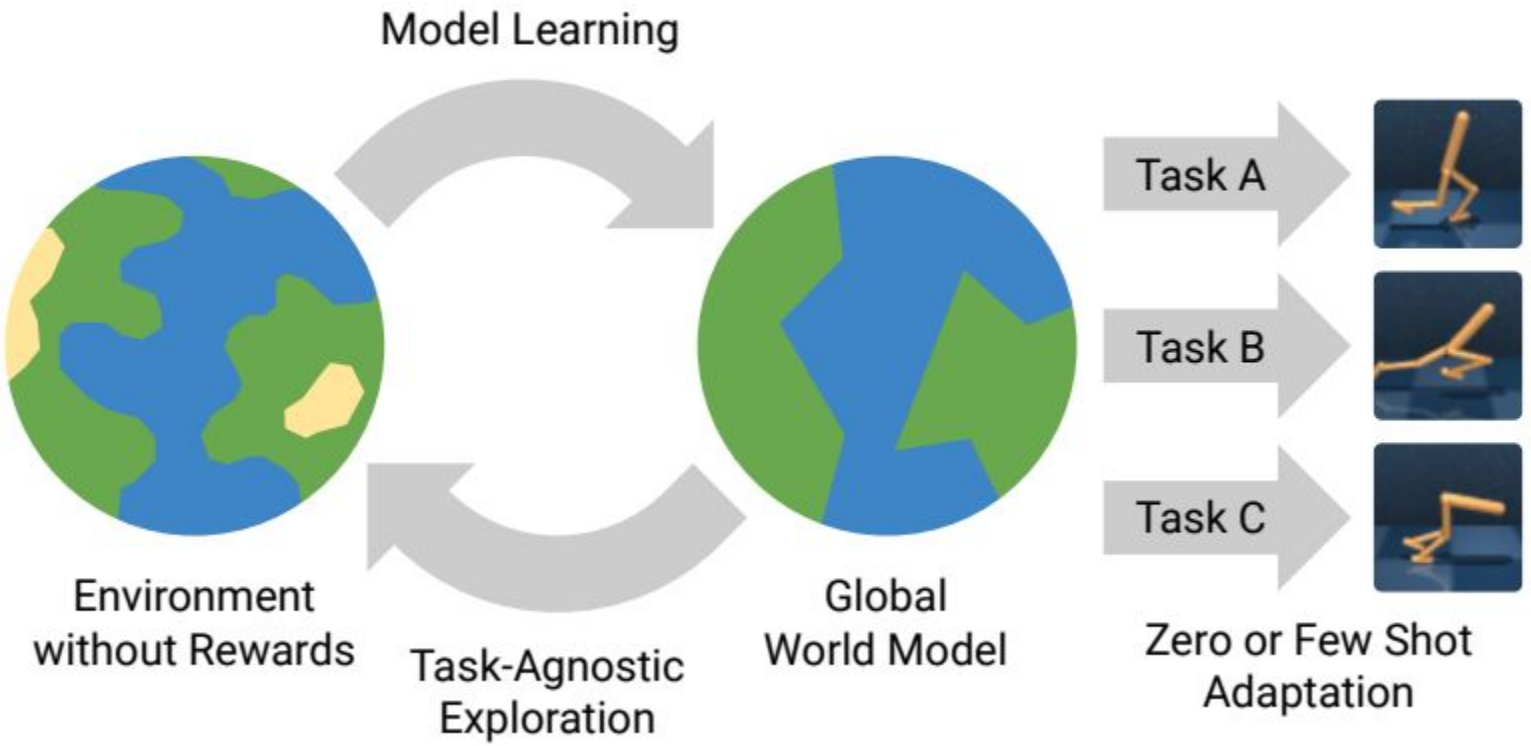
Slightly changed transition dynamics:

- Simulation-to-real transfer is popular in robotics.
- Learn a global neural network initialization that can quickly adapt to new tasks eg. Plan2Explore
- Multi-task learning: Learn a distribution over the task space. When a new task comes in, we may quickly identify in which cluster of known tasks (dynamics models) it belongs.

Transfer: Sim2Real



Transfer: Plan2Explore



Safety

- Safety is an important issue, especially when learning on real-world systems.
- For e.g., with Epsilon greedy exploration it is easy to break a robot before any learning takes place.
- Model-based learning has been used for safety:
  - Given a ‘safe region’ of the current policy, explores while ensuring return to safe region if necessary (Berkenkamp et al.).
  - Maintain two policies using two models:
    - Use the first model for exploration.
    - The second model has uncertainty bounds and is used for verification of the safety of the policy of first model.

Explainability

- Relatively recent sub-field in RL and builds on model-based methods:
  - Model reconciliation for explicable and legible robot behavior.
- Explainability is now widely regarded as a crucial prerequisite for AI to enter society.