Model-based RL

- Have or learn a reward function ("look like the observed behavior").
- Learn a task model (locally weighted regression).
- Use model-based reinforcement learning to find a successful policy.
- Refine model based on experience, and replan.
- Accommodate imperfect models and improve policy using online policy search, or manipulation of optimization criterion.

Reinforcement Learning

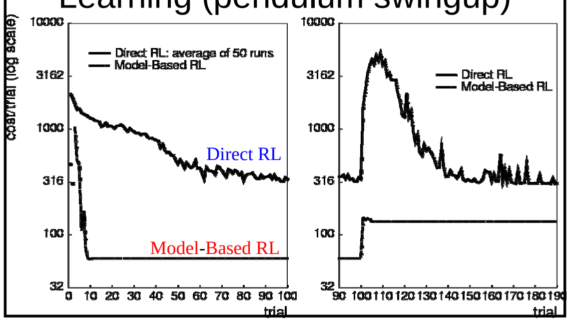
Direct RL

- Adjust parameterized policy.
- An example of a parameterized policy is a Q function, which stores the value of taking each possible action in each

Model-Based RL

- Learn model, plan.
- We use locally weighted learning to learn models.
- We use forms of dynamic programming to plan.

Comparing Direct and Model-Based Reinforcement Learning (pendulum swingup)



Planning With Imperfect Models

$$C = \sum |\mathbf{x}(k) - \mathbf{x}_d(k)|^2 + \sum |\mathbf{u}(k)|^2 + \sum \lambda(\text{confidence})|\mathbf{x}(k+1) - f(\mathbf{x}(k), \mathbf{u}(k))|^2$$

- Optimistic Planning: Include a factor, λ, that depends on how confident the planner is in the model, and controls how much the plan has to obey the model.
- Plan without integrating dynamics: Parameterize x(k) and u(k) and use function optimization to minimize C.

Model-Based Policy Refinement

u(t) ➡ Robot ➡ x(t)

Δu(t) ⬅ Inverse Model ⬅ Δx(t)

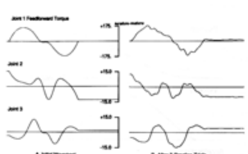


Figure 7.5: Feedforward Torques.

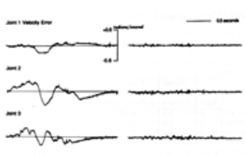


Figure 7.6: Velocity Errors.

Task Level Direct RL

- Adjust reward function given to planner to compensate for steady state execution error.
- Closely related to model-based policy refinement and previous work on task level learning.
- Need to be able to measure steady state error in the presence of disturbances and transient behavior due to learning.

When Should One Use Direct RL/Policy Refinement?

- When small number of actions relevant to task can be abstracted. Q(x,u) is largely independent of state.
- When policy can be parameterized with small number of task parameters.
- On complex tasks with few choices, such as voting.

What did we learn?

- Learning models tremendously accelerates robot learning, allowing small numbers of trials/demonstrations, with little apriori knowledge required.
- Robots can select relevant task variables, and build models.
- Robots can acquire task paradigms from observation and from reasoning.
- Policy refinement can improve model-based learning.
- Need modeling techniques that produce confidence estimates, fast predictions, and fast training.