Shaping

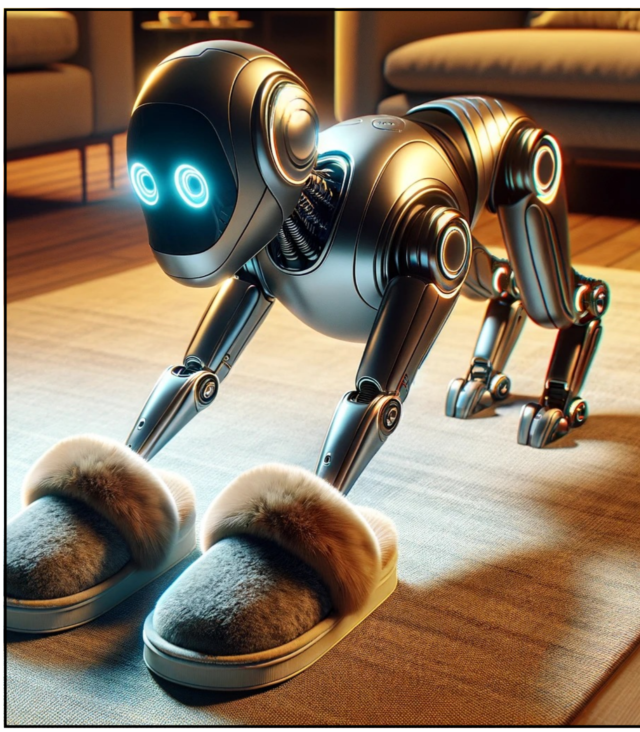
Ron Parr
CSCI 2951-F
Brown University

What is shaping: Psychological perspective

- Rewarding an animal or a child (only) when it achieves a complicated task may results in never giving rewards
  - Not clear how to communicate complicated behavior required to achieve task
  - Random behavior by the learner may never achieve the task
- Shaping:
  - Give small rewards for small tasks on path to desired behavior
  - Gradually change the reward structure to guide the learner

Shaping example:
Train dog to get your slippers

- Reward dog for going into the closet when you say "slippers"
- Reward dog for going into the closet and going near the slippers
- Reward dog for going into the closet, and picking up the slippers
- Reward dog for going into the closet, picking up the slippers, and bringing them to you



Creatures vs. Robots

- Constantly tweaking a reward function while interaction with a pet or child may be practical or even satisfying
- Not clear it's practical/desirable to do this with robots/algorithms

Shaping Fails: 1

- Goal: Balance a bicycle and ride it to a distance goal
- Natural reward structure: Reward for reaching the goal
- Proposed shaping: Add additional reward for balancing
- Pitfalls:
  - Accumulated balancing rewards may eclipse reward for going to the goal
  - Agent may learn an optimal policy that just goes in circles if turning towards the goal involves a risk
  - Could potentially be addressed by carefully balancing the scale of each reward, but tricky in practice

Shaping Fails 2

- Goal: Robot soccer player that scores goals
- Natural reward structure: Reward for scoring goals
- Proposed shaping: Add reward for getting the ball
- Pitfalls:
  - Accumulated reward for touching the ball eclipse reward for scoring
  - Agent “vibrates” continually touching the ball, but never tries to score
  - Hard to balance these rewards

Assumptions

- Original MDP: M
- Original reward function: R(s,a,s')
- Shaping reward: F(s,a,s')
- New MDP M' same as M except:
- New reward function: R'(s,a,s')= R(s,a,s')+ F(s,a,s')

- Desiderata:
  - Optimal policy for M' same as optimal policy for M ← Policy invariance
  - Solving M' is somehow easier than solving M

Intuition (undiscounted case)

- One way to avoid undesirable behaviors from "Fails" is to avoid cycles
- Make sure that shaping function F does not reward cycles
- For any (s_1,a_1,s_2), $(s_2,a_2,s_3)\ldots(s_n,a_n,s_1)$
- $F(s_1,a_1,s_2)+F(s_2,a_2,s_3)+\ldots+(s_n,a_n,s_1) = 0$

- Turns out a generalization of this is both a *sufficient* condition to achieve policy invariance and also a *necessary* one

Potential based shaping functions

- Let $\Phi(s)$ be any real valued function of state
- F is a potential-based shaping function if: $F(s,a,s') = \gamma\Phi(s')-\Phi(s)$
- For γ=1, this satisfies our condition for zero reward cycles

Potential based shaping functions preserve the optimal policy

- Suppose:

$$Q_M^*(s,a) = R(s,a) + \gamma \sum_{s'} P(s'|s,a) \max_{a'} Q_M^*(s',a')$$

- Claim for M' with added shaping:

$$Q_{M'}^*(s,a) = Q_M^*(s,a) - \Phi(s)$$
$$V_{M'}^*(s) = V_M^* - \Phi(s)$$

$$Q^*_{M'}(s,a) = R'(s,a) + \gamma \sum_{s'} P(s'|s,a) \max_{a'} Q^*_{M'}(s',a')$$

$$Q^*_{M'}(s,a) = R(s,a) - \Phi(s) + \gamma \sum_{s'} P(s'|s,a) \left[\max_{a'} Q^*_{M'}(s',a') + \Phi(s')\right]$$

$$Q^*_{M'}(s,a) + \Phi(s) = R(s,a) + \gamma \sum_{s'} P(s'|s,a) \left[\max_{a'} Q^*_{M'}(s',a') + \Phi(s')\right]$$

Satisfied when: $Q^*_{M'}(s,a) = Q^*_{M}(s,a) - \Phi(s)$

Policy Invariance

- Suppose: $Q^*_{M'}(s,a) = Q^*_{M}(s,a) - \Phi(s)$
- Then $\pi^*_{M'} = \pi^*_{M}$
- Why? Because Φ does not depend on a

How does this help?

- Convergence
- Normally start with V=0, do value iteration
- Suppose we start with $V_{M'}$=0, $\Phi = V_M$*
- Then value iteration converges in one iteration b/c

$$V^*_{M'}(s) = V^*_M - \Phi(s) = 0$$

- Picking a shaping reward that is close to V* is good

How does this help?

- Exploration
- Suppose we do ε greedy exploration
- Shaping rewards that give high rewards for good states will focus exploration on good states earlier

Shaping potential example: cart pole

- Suppose we just penalize crashing
- States other than crashing are equivalent until value of crashing propagates

- Suppose Φ=-abs(radians away from upright)
- Doesn't change optimal policy, but learning quickly gets samples suggesting that tipping over is bad

How can this hurt?

- Suppose you pick a terrible shaping function
- Can slow down convergence
- Can cause exploration to waste effort

- But: Damage is limited because optimal policy remains unchanged

Necessity

- What if the shaping reward is not potential based?
- For non-potential based shaping reward, there will exist an MDP that exploits this in a way that changes the optimal policy

Use in practice

- Nice example where theory informs practice
- After this paper, everybody changed how they do shaping
- Still used today
- Sometimes the discount is skipped
- Suppose $s^*=(x,y,z)$ is desired configuration of robot: $F=(|s-s^*|)$

Weiwora’s Observation

- Potential based shaping and value initialization are equivalent
- Adding a shaping function and initializing the value function estimate with the shaping function, i.e., V0=F have equivalent effects