20. Reinforcement Learning 2

Toxonomies of RL

- Methods based on value functions
  - $V^\pi(x)$ is expected long term reward of starting at x and executing policy π
  - Policy Search methods; Actor-critic Methods
  - Policy-Gradient method $\subset$ Policy Search
  - PPO, is Actor-critic method
  - Policy gradient as optimizer
- Topic 2
- Topic 3

Today's Topic outline

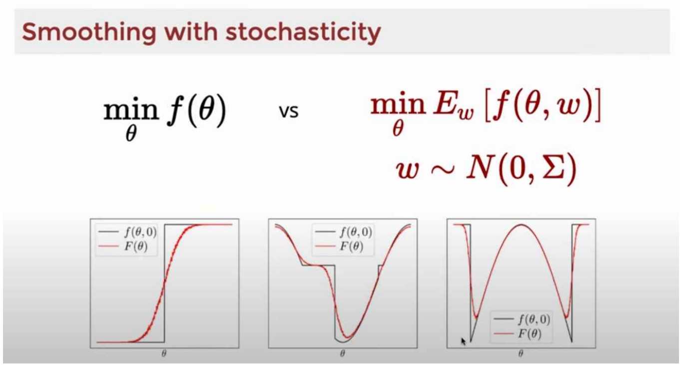
- Reinforce $\subset$ Policy Gradient
  - $min_\alpha E[f(x)], x \propto p_\alpha(x)$, distribution of x, $N(\alpha, \sigma(fixed))$
  - write a distribution over possible x's and minimize the expected value
  - delta-like funciton is the goal, highest possibility at lowest value
  - optimize of gradient descent: $\frac{\partial}{\partial\alpha}E[f(x)] = E[f(x)\frac{\partial}{\partial\alpha}lnP_\alpha(x)]$, log-likelihood method (or policy gradient "trick")
- Reinforce with additive Gaussian noise
  - $x \sim P_\alpha \sim N(\alpha, \sigma^2)$
  - $x = \alpha + \beta$, $\beta \sim N(0, \sigma^2)$
  - $P_\alpha(x) = Ce^{\frac{-(x-\alpha)^T(x-\alpha)}{2\sigma^2}}$, probability density funciton of gaussian
  - $lnP_\alpha(x) = \frac{-(x-\alpha)^T(x-\alpha)}{2\sigma^2} + ...$ terms that do not depend on α
  - $\frac{\partial}{\partial\alpha}lnP_\alpha(x) = \frac{1}{\sigma^2}(\alpha - x)^T = \frac{1}{\sigma^2}\beta^T$
  - $f(x)\frac{\partial}{\partial\alpha}lnP_\alpha(x) = \frac{1}{\sigma^2}f(\alpha+\beta)\beta^T$
  - $\Delta\alpha = -\eta\frac{1}{\sigma^2}f(\alpha+\beta)\beta^T$, η : learning rate

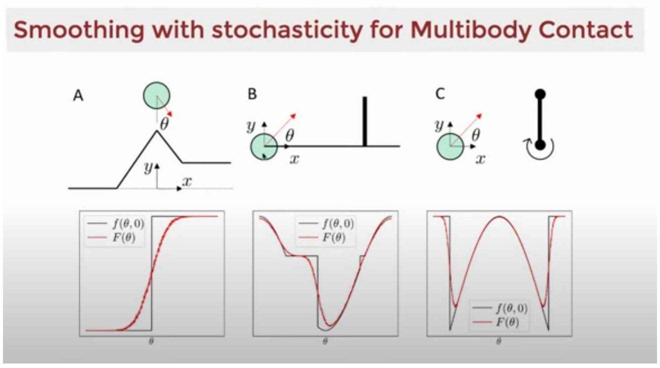
- given a small perturbation:
- $\Delta\alpha = -\eta\frac{1}{\sigma^2}[f(\alpha+\beta) - f(\alpha)]\beta^T$
  - if $f(\alpha+\beta) > f(\alpha)$, move $-\beta$ direction
  - if $f(\alpha+\beta) < f(\alpha)$, move $+\beta$ direction
- If you have gradients, why not use them? (from AutoDiff)
  - the answer is subtle!
  - scienrio: a wsg gripper try to grip a brick, z_{height}
    - controller: descend until z_{close} ,close gripper,raise hand
    - rewards:height of brick at time = 5 sec
    - plot reward in Y vs z_{close} in X, very discontinuous loss landscapes
    - gradient descent on discontinuous landscapes in general doesn't work very well
    - **but adding probability density function, the smoothing effect works well**
- The idea of Non-smooth optimization
  - is "randomized smoothing"
  - new interpretation that, policy gradient in RL is sort of 1 to 1 mapping Randomized Smoothing
  - example, $min\ |x|_1$, l1 norm
- In RL the randomriztion comes from
  - $P_\alpha(x)$, exploration
  - Random initial conditions
  - Domain randomrization
  - the Smoothing effect helps convergence and optimization

1. Good papers on RL

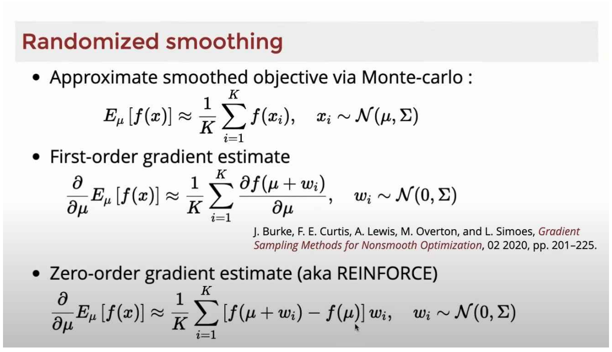
- Do Differentiable Simulators Give Better Policy Gradients?

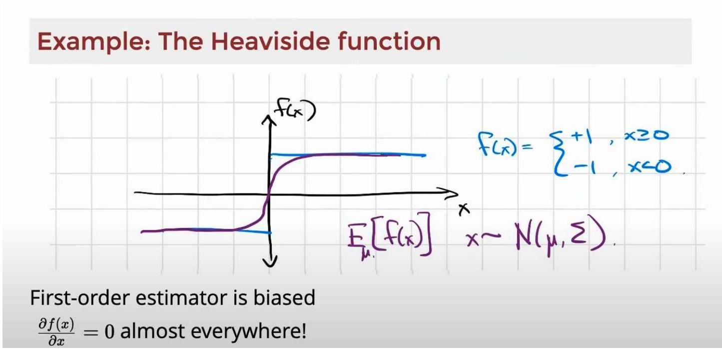
Do Differentiable Simulators Give Better Policy Gradients.pdf





- Do Differentiable Simulators give better policy gradients? the answer is subtle





- Is
-

2. What does policy gradient look like for control? (open question)

- linear Gaussian dynamics + quadratic cost

3. Examples