Is Bang-Bang Control All You Need?

Solving the Continuous Mountain Car problem

Fredson Aguiar

Is Bang-Bang Control All You Need? Solving Continuous Control with Bernoulli Policies

Tim Seyde[1]
MIT CSAIL

Igor Gilitschenski
University of Toronto

Wilko Schwarting
MIT CSAIL

Bartolomeo Stellato
Princeton University

Martin Riedmiller
DeepMind

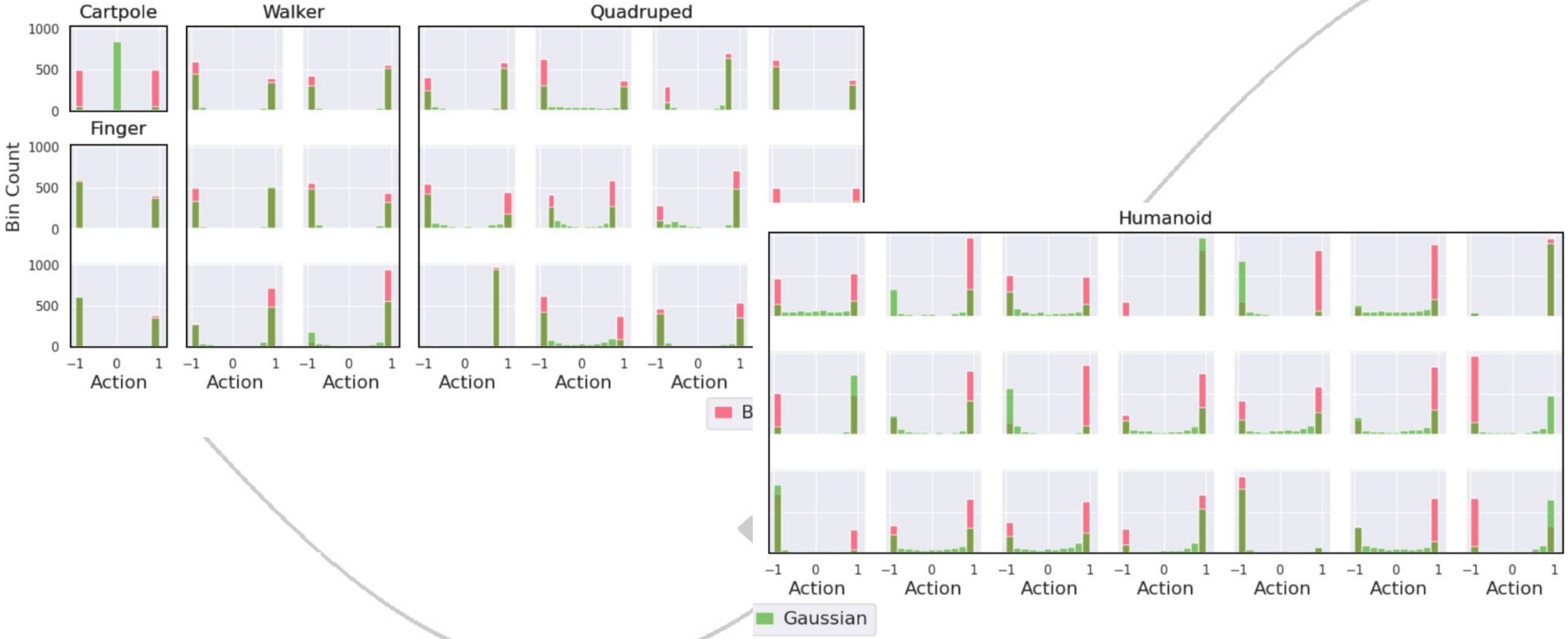
Markus Wulfmeier[2]
DeepMind

Daniela Rus[2]
MIT CSAIL

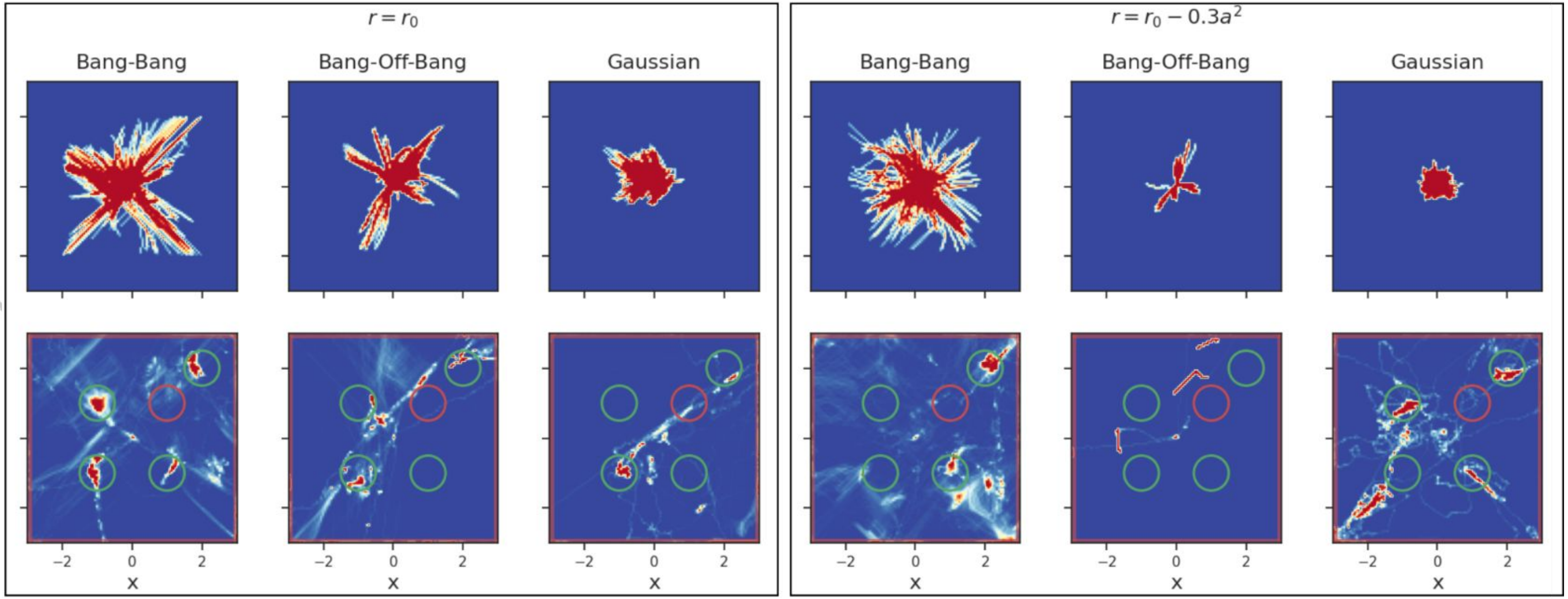
Results from Seyde et. All

- Bang-Bang and Bang-of-Bang controls emerge as optimal solutions
  - Some Specific Problems, such as Maximum State or Minimal Fuel
  - Presents the discussion of Singular Arcs and Chattering Controls
- Shows Competitive Results and Trade-offs of using Bernoulli Policies
  - More efficient algorithms as the explored space is smaller
  - Intrinsic property of space exploration due to extreme actions
  - Problems when dealing with fine controls
- Interesting Experiments
  - Compares the actions taken for various algorithms and problems

Results from Seyde et. All



Results from Seyde et. All



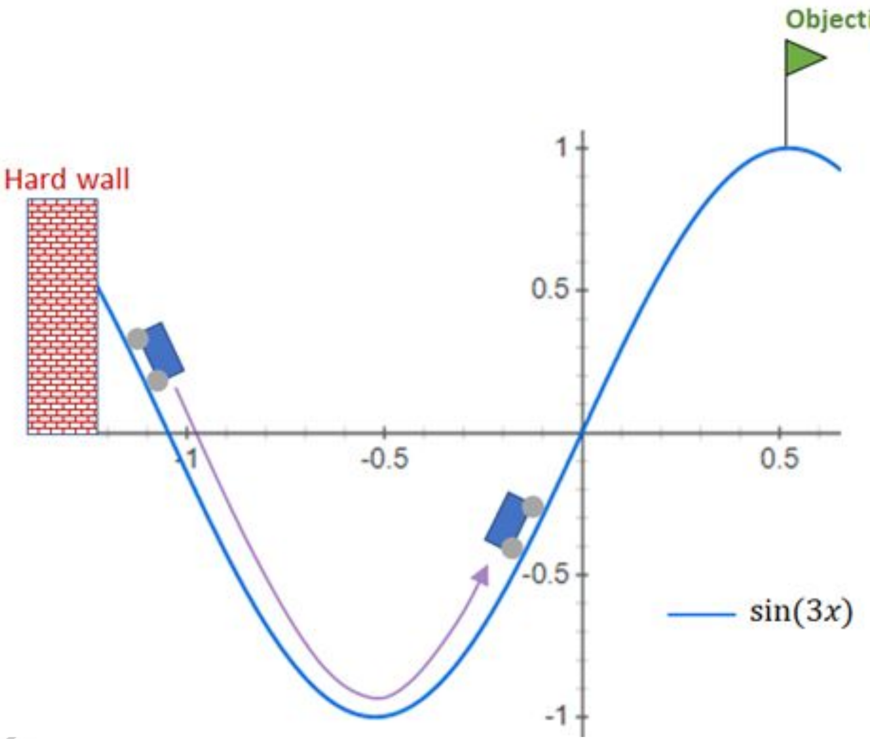
Objectives

- Reproduce The Results
  - The Classical **Mountain Car Continuous** control problem
  - Using increasing complexity models
  - Compare with the previous results from Seyde et. All

- Learn Various New Models
  - Tabular Q-Learning
  - Discretized Tabular Q-Learning
  - REINFORCE with normal parametrization
  - Deep Deterministic Policy Gradient (DDPG)

The Problem

- Classical Control Problem
- Discrete and Continuous
- Long-term planning
- Present in Gymnasium
- Simple for our examples

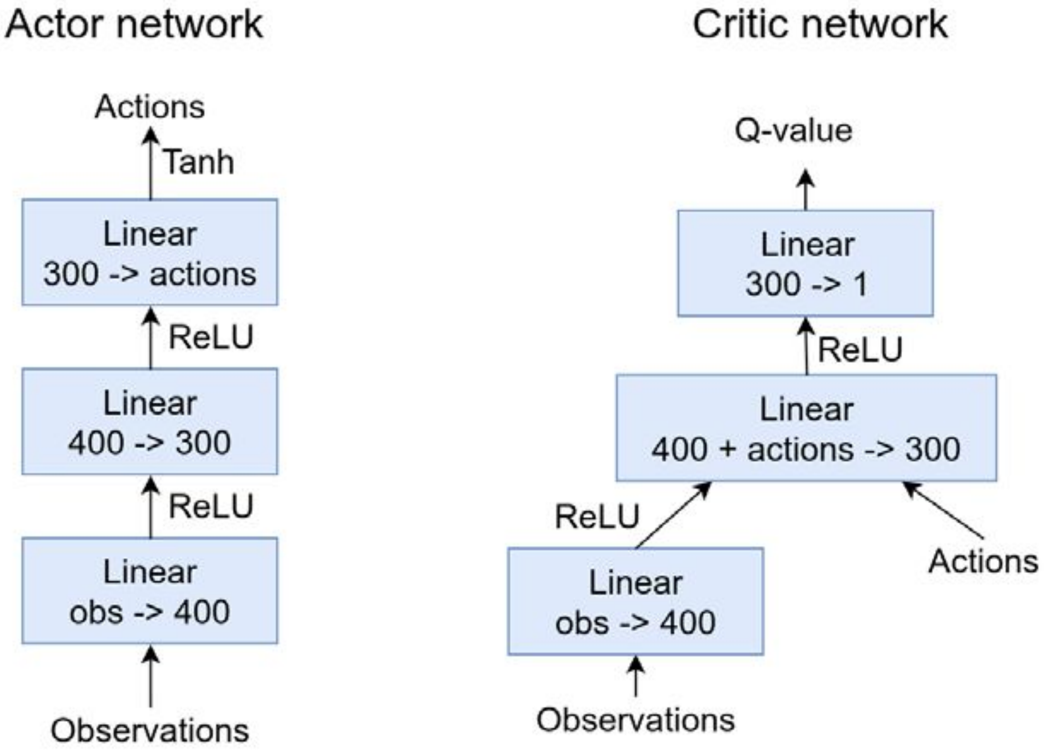


Continuous Action Space

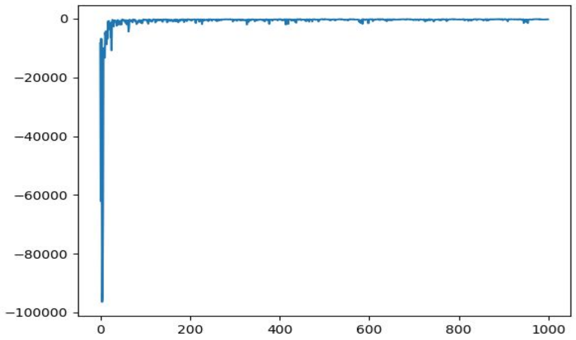
- A parametric policy to continuous action
  - Action space is too large for probabilities array
  - Learns how to generate parameters from states
  - Generates the actions from sampling

$$\pi(a|s,\boldsymbol{\theta}) \doteq \frac{1}{\sigma(s,\boldsymbol{\theta})\sqrt{2\pi}} \exp\left(-\frac{(a-\mu(s,\boldsymbol{\theta}))^2}{2\sigma(s,\boldsymbol{\theta})^2}\right)$$

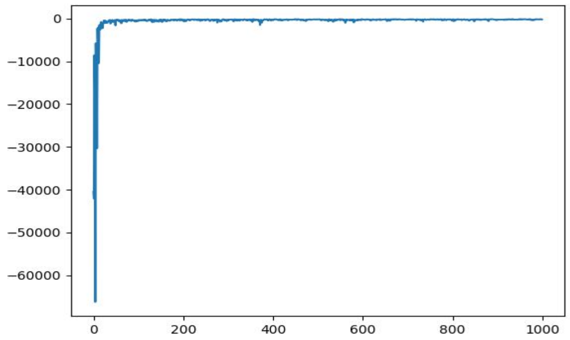
Deep Deterministic Policy Gradient (DDPG)



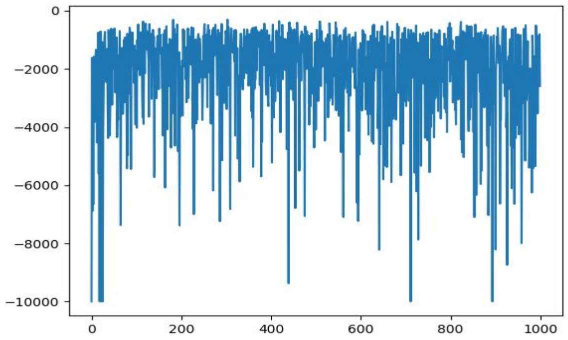
The Results (learning)



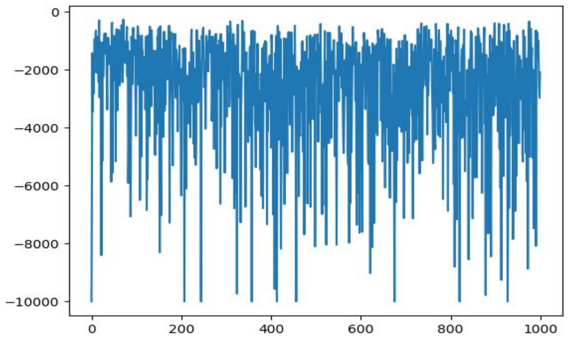
Discrete Tabular Q-Learning



Discretized Tabular Q-Learning

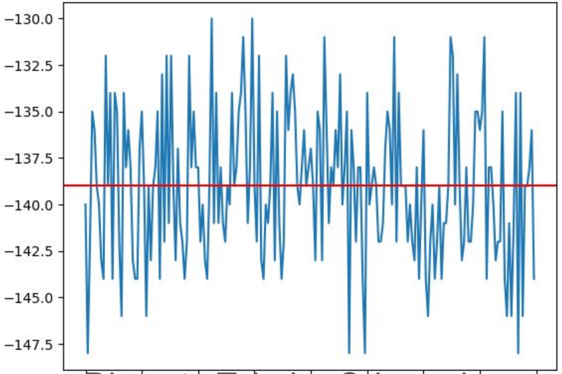


REINFORCE cont normal

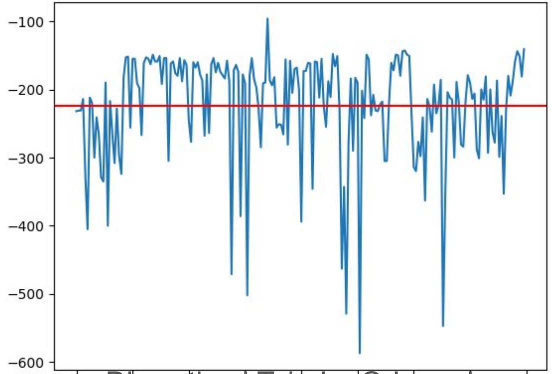


DDPG cont normal

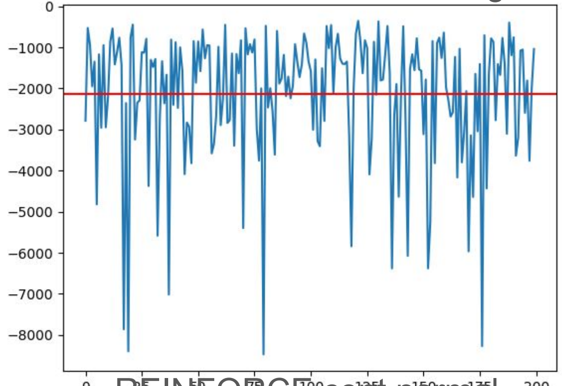
The Results (rewards)



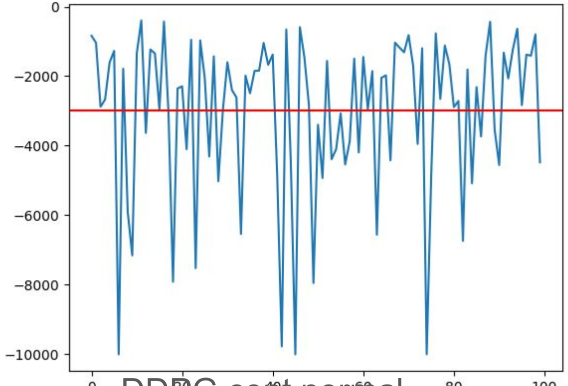
Discrete Tabular Q-Learning



Discretized Tabular Q-Learning



REINFORCE cont normal



DDPG cont normal

The Results (actions)

Discussion

- Results
  - Discretized solutions more efficient and reliable
  - Nearly reproduce the results from Seyde et. All
  - Lack interpretability of the neural nets
  - Issues with fine adjustments in the neural nets

- Difficulties and Learnings
  - Learning the real world side of using Neural Nets
  - Technical issues while implementing the models
  - Different classes of models for different problems
  - Choosing simpler models from knowing the problem