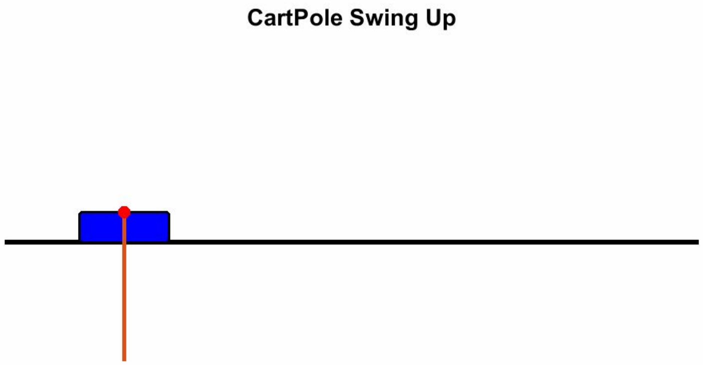
RL methods for cart pole swing up and stabilization

Bogdan Alexandrov, Vadim Shirokinskiy

Problem statement

We have an cart pole, the pendulum is looking down. The task is to lift the pendulum up and hold it.



CartPole Swing Up

Parameters of the system

pendulum mass = 0.1

pendulum length = 0.5

area length = inf, 10, 5 (optionally)

steps per episode = 1500 - 3000

initial state: (pi,)

cart mass = 1

g = 9.81

dt = 0.003

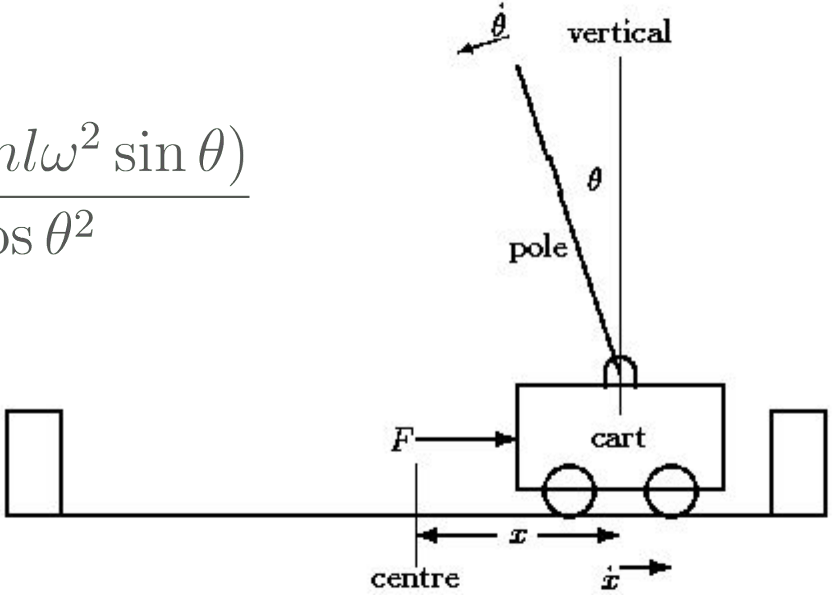
System dynamic

$$\dot{\theta} = \omega$$

$$\dot{\omega} = \frac{(m+m_c)g\sin\theta - \cos\theta(u + ml\omega^2\sin\theta)}{(4/3)(m+m_c)l - ml\cos\theta^2}$$

$$\dot{h} = dh$$

$$\dot{dh} = \frac{u + ml(\omega^2\sin\theta - \dot{\omega}\cos\theta)}{m+m_c}$$



State -> observation transition

$$\theta, \omega \rightarrow \cos\theta, \sin\theta, \omega$$

Since we don’t care about absolute value of angle(10001 * pi or pi are equal for our system).

We need only sine and cosine of angle

Used approaches

RL methods:

1) Reinforce
2) Actor-Critic

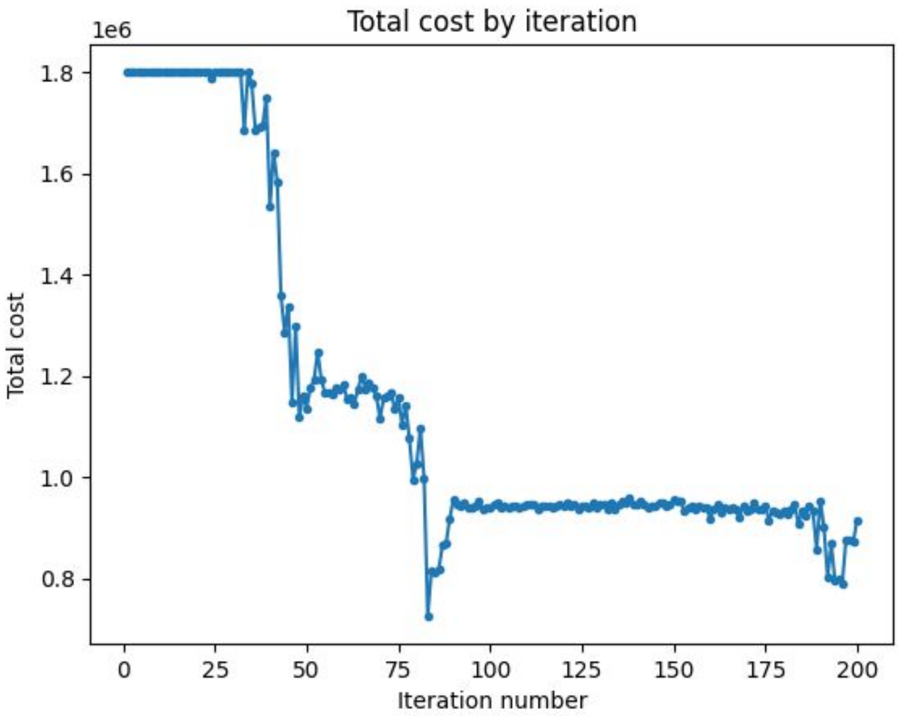
Ways to influence the environment:

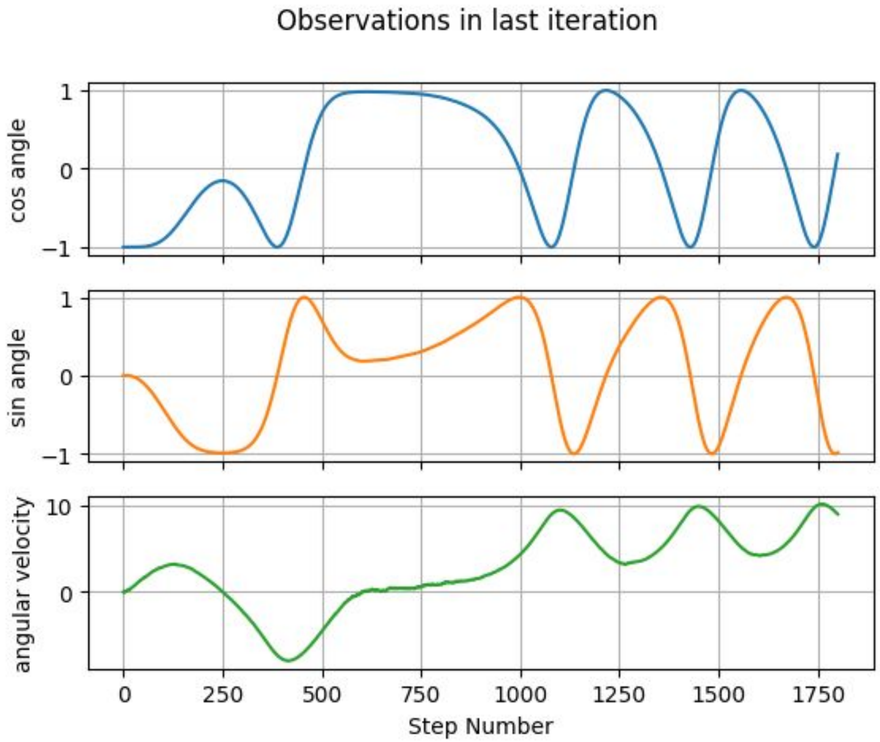
1) Apply continuous bounded action
2) Predict direction of predetermined force (used 10 in our experiments)

Setup1

Reinforce with fixed force. Cost is also fixed. Available surface is unlimited.

```
if $\cos\theta$ < -0.5: return 1000
elif $\cos\theta$ < 0.8:return 500
elif $\cos\theta$ < 0.95: return 200
else: return 0
```

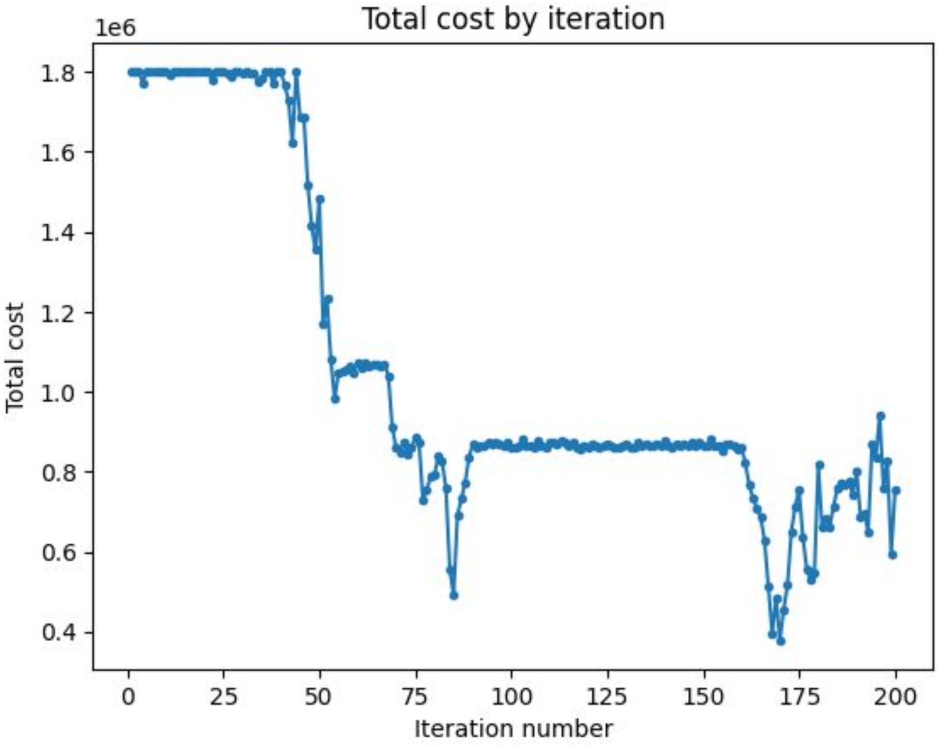


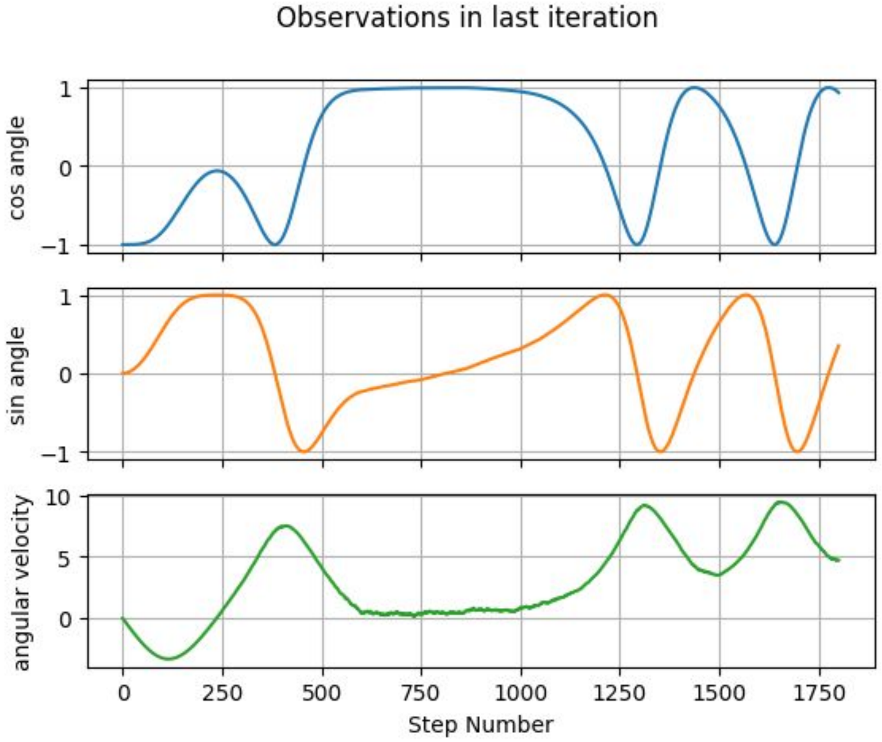


Hidden dim: 32
Hidden layers: 1

lr: 0.005
steps: 1800

N_episode: 2
N_iters: 200

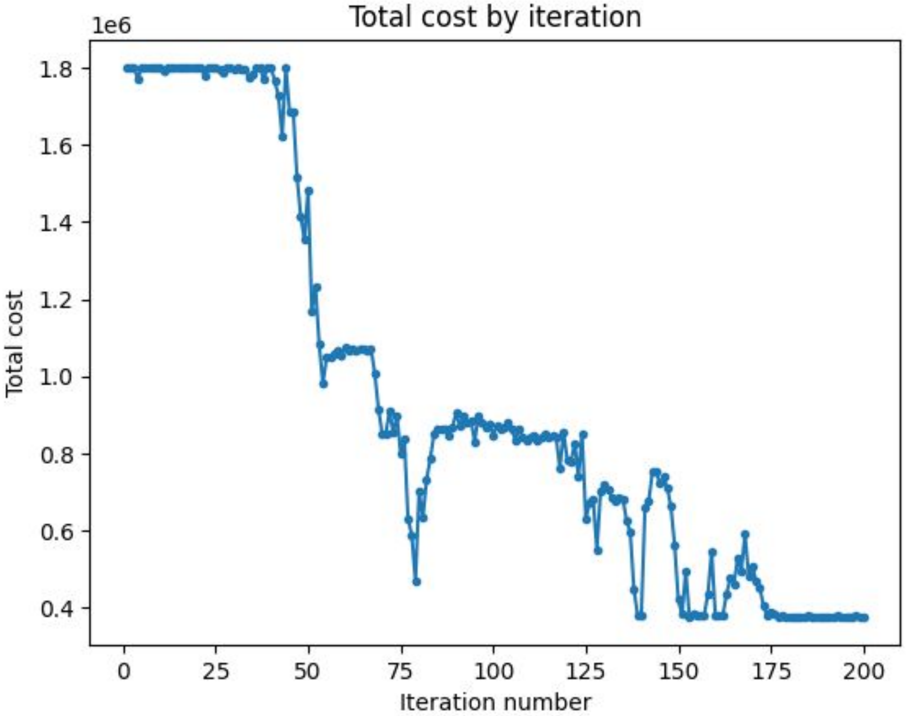


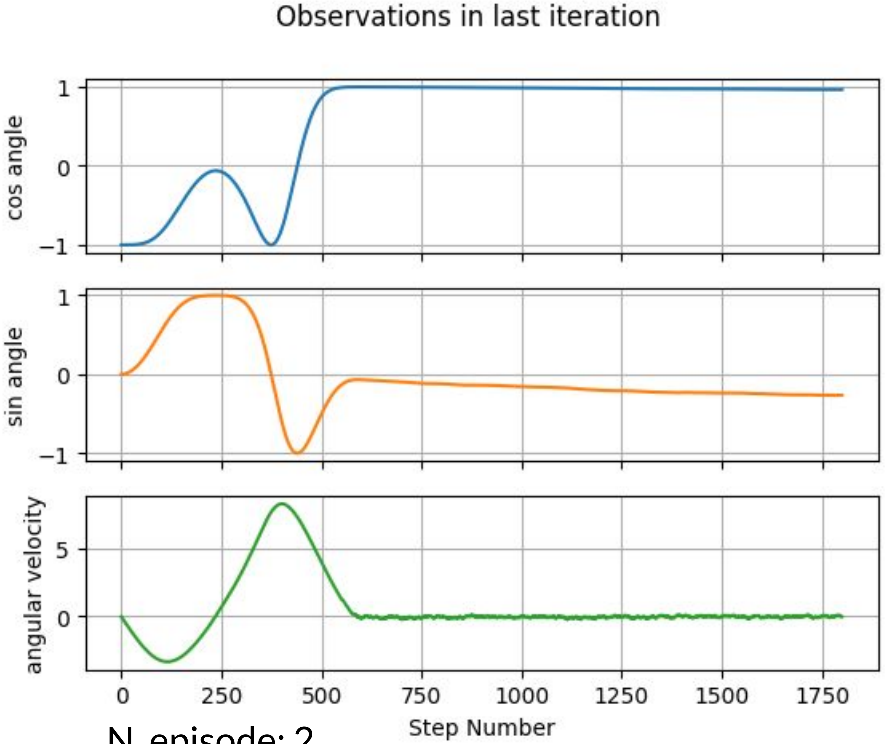


Hidden dim: 32
Hidden layers: 2

lr: 0.005
steps: 1800

N_episode: 2
N_iters: 200





Hidden dim: 32	lr: 0.005	N_episode: 2
Hidden layers: 2	steps: 1800	N_iters: 200

Here added additional punishment on high velocity:

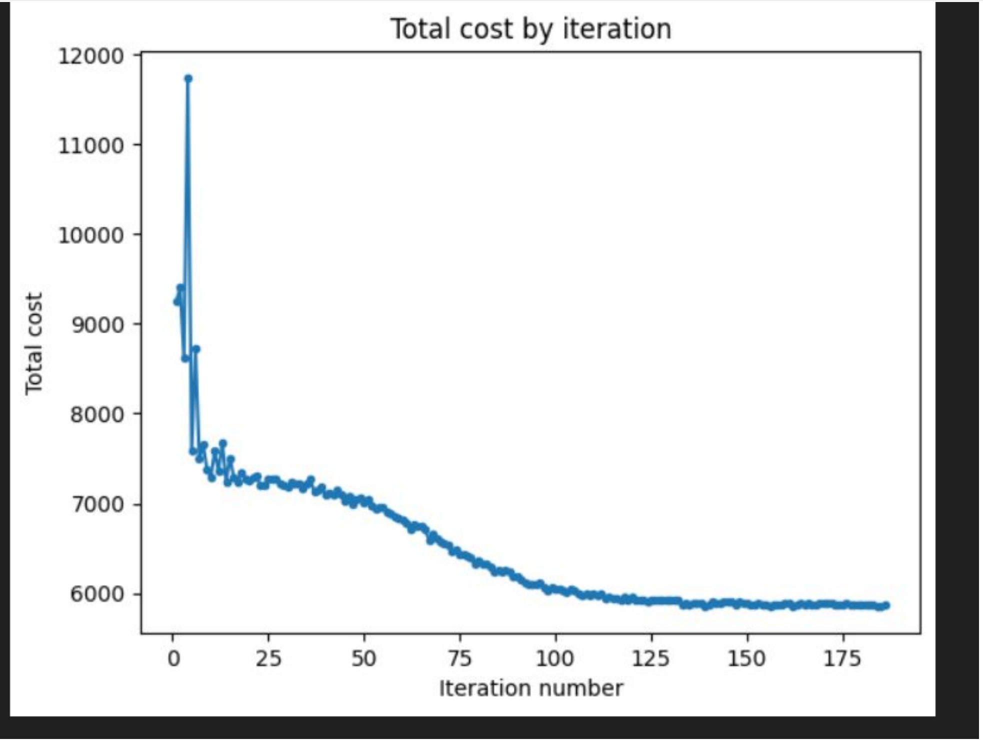
if $\cos\theta > 0.95 :\ \omega^2$

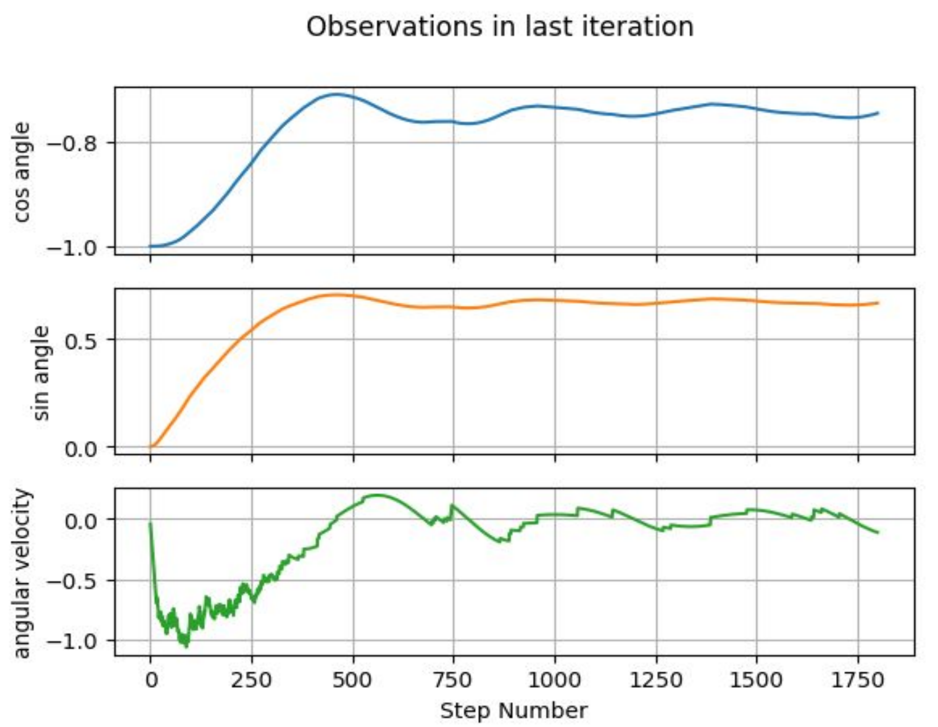
Setup 2

Same as previous, but changed cost to smooth one.

$$a * (1 - \cos\theta)^2 + b * \omega^2$$

where a, b are positive weights



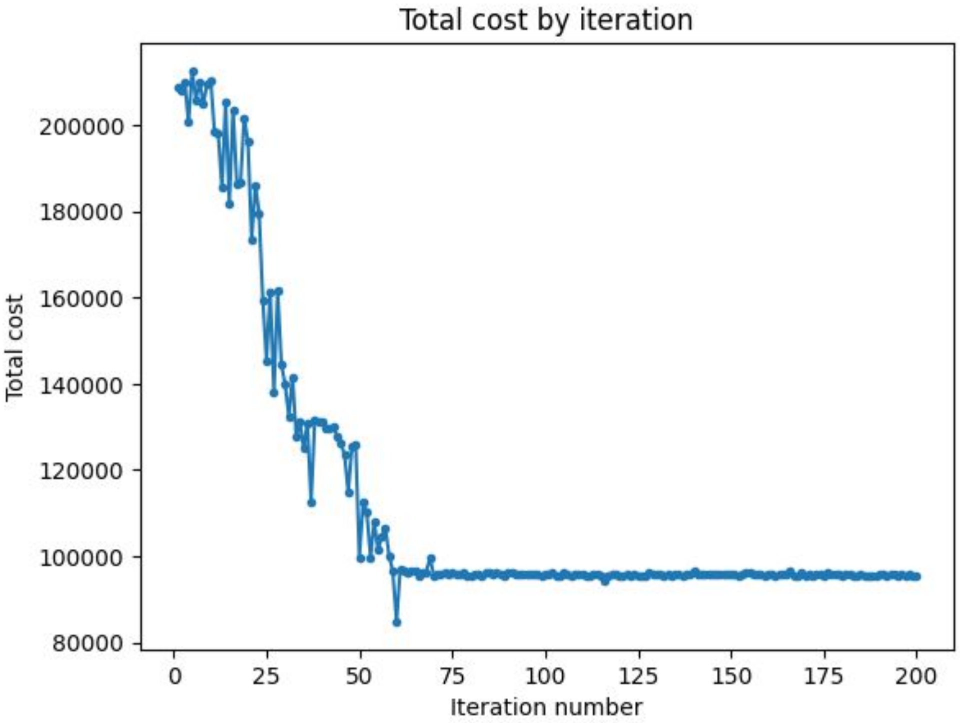


Hidden dim: 32
Hidden layers: 2

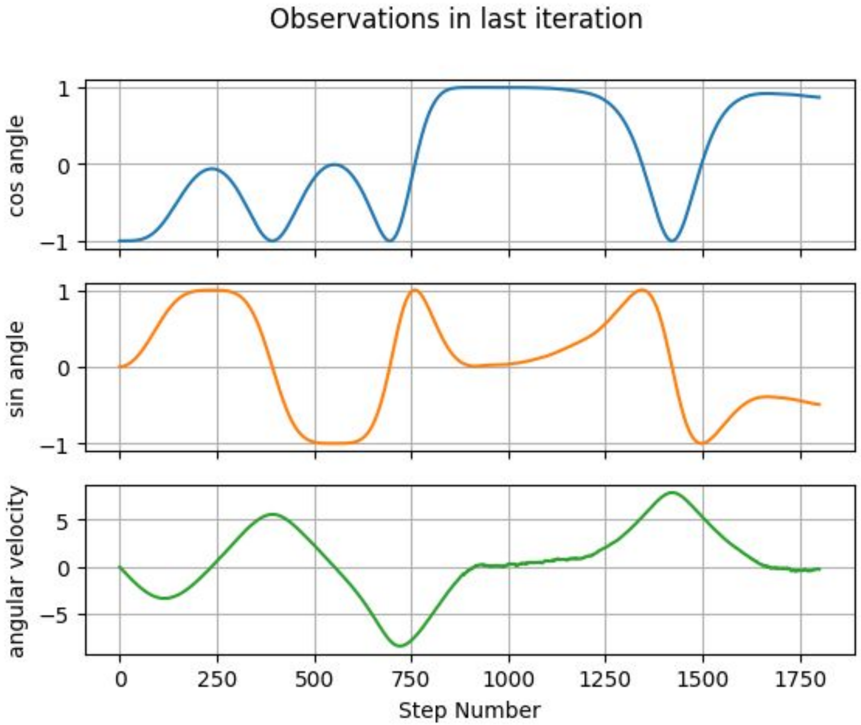
lr: 0.005
steps: 1800

N_episode: 2
N_iters: 200

cost coefs a=b=1.



Total cost by iteration



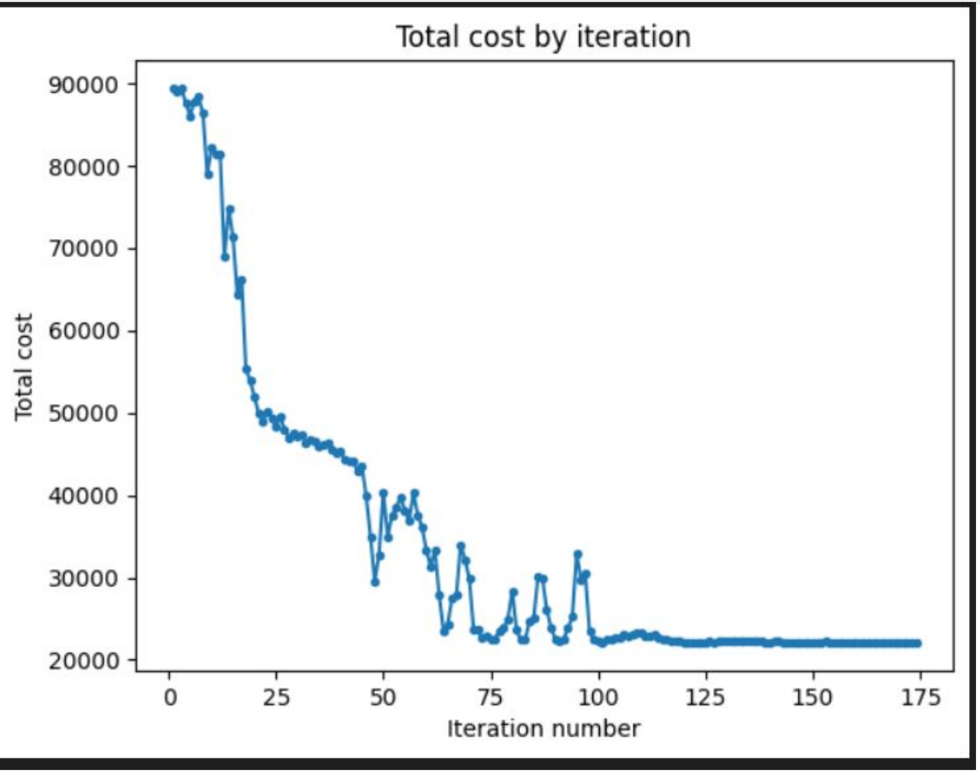
Observations in last iteration

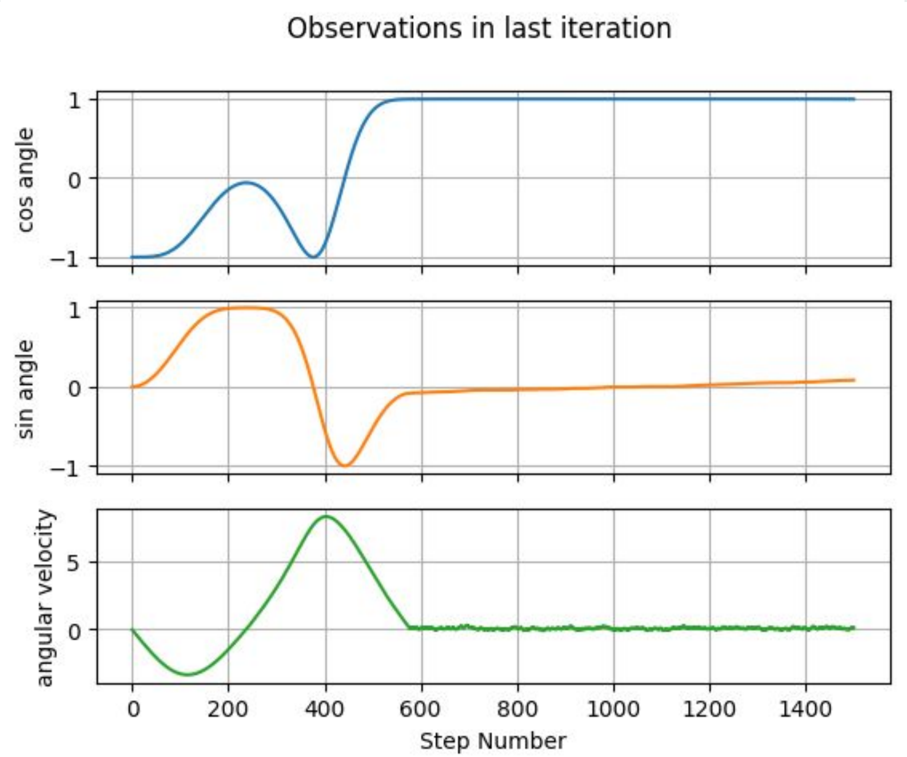
Hidden dim: 32
Hidden layers: 2

lr: 0.005
steps: 1800

N_episode: 2
N_iters: 200

cost coefs a=30, b=1.





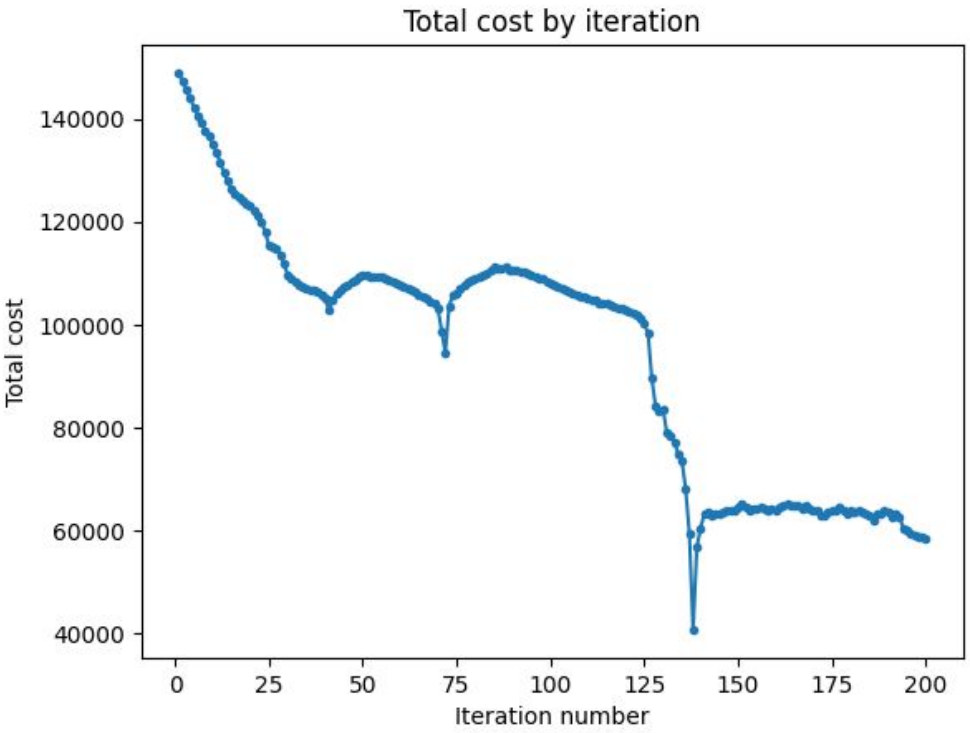
Hidden dim: 32	lr: 0.005	N_episode: 2
Hidden layers: 2	steps: 1500	N_iters: 200

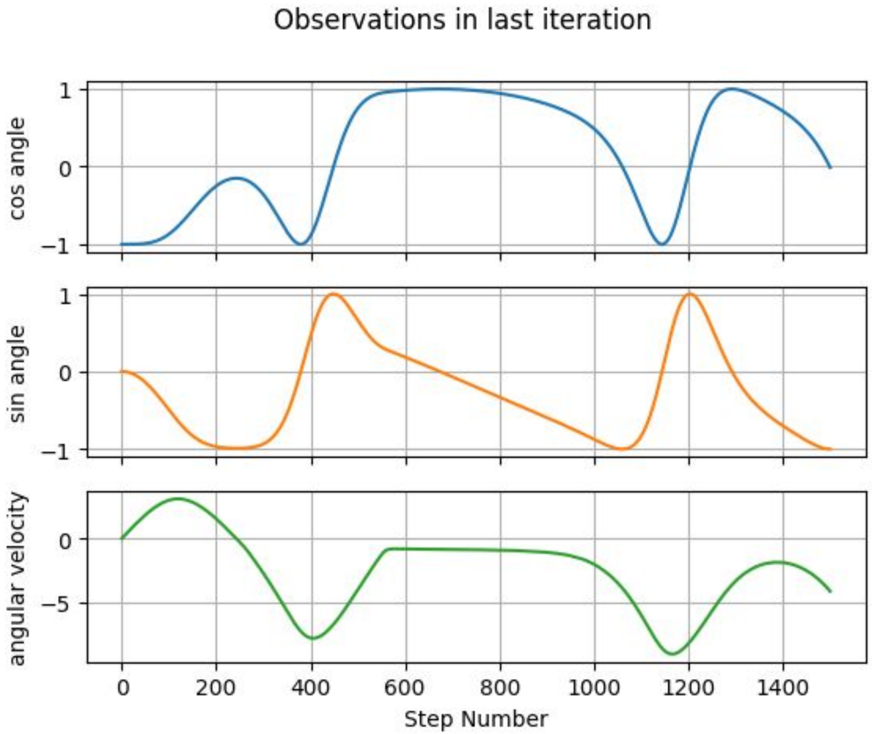
cost coefs a=30, b=1.

Setup 3.

Here we changed model with fixed force to unfixed. Model predicts force itself within [-10, 10].

P.S. worked not good



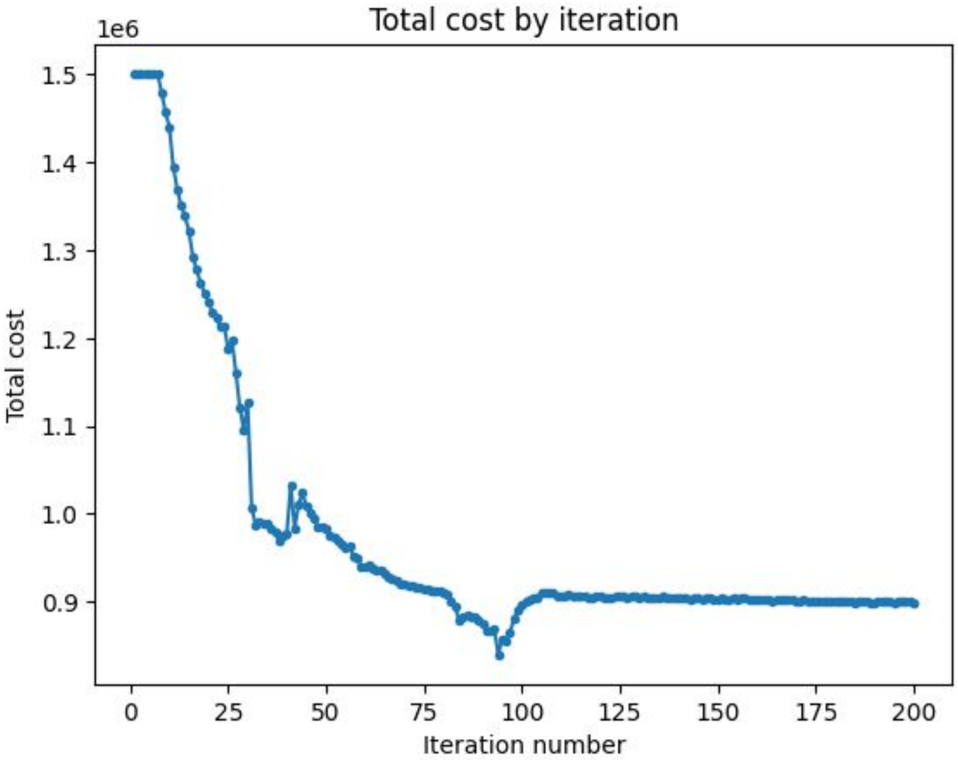


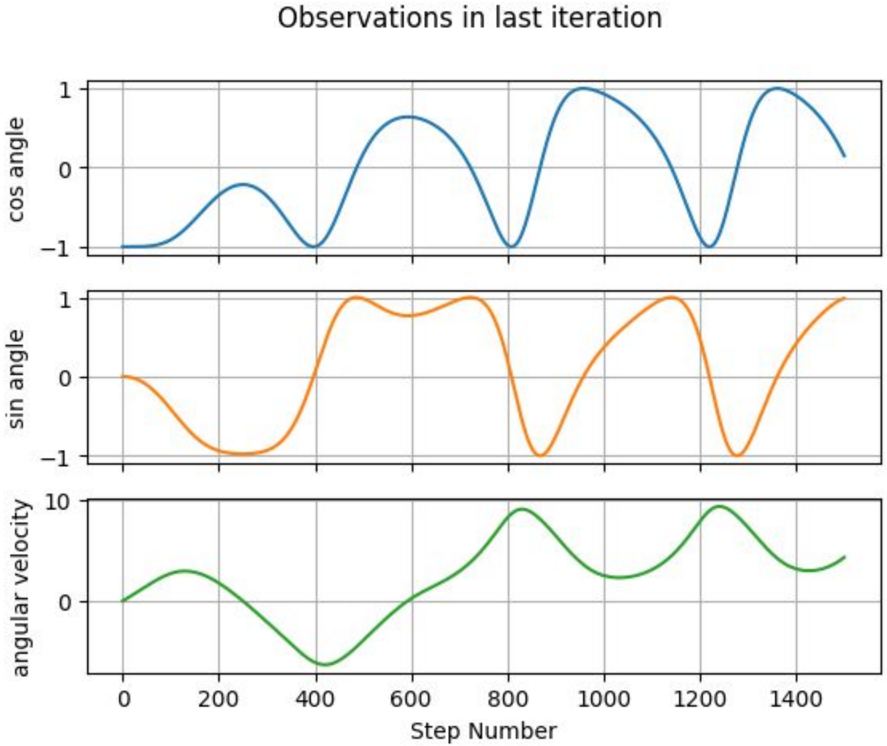
Hidden dim: 32
Hidden layers: 2

lr: 0.005
steps: 1500

N_episode: 2
N_iters: 200

cost function: $50 * (1 - \cos\theta)$



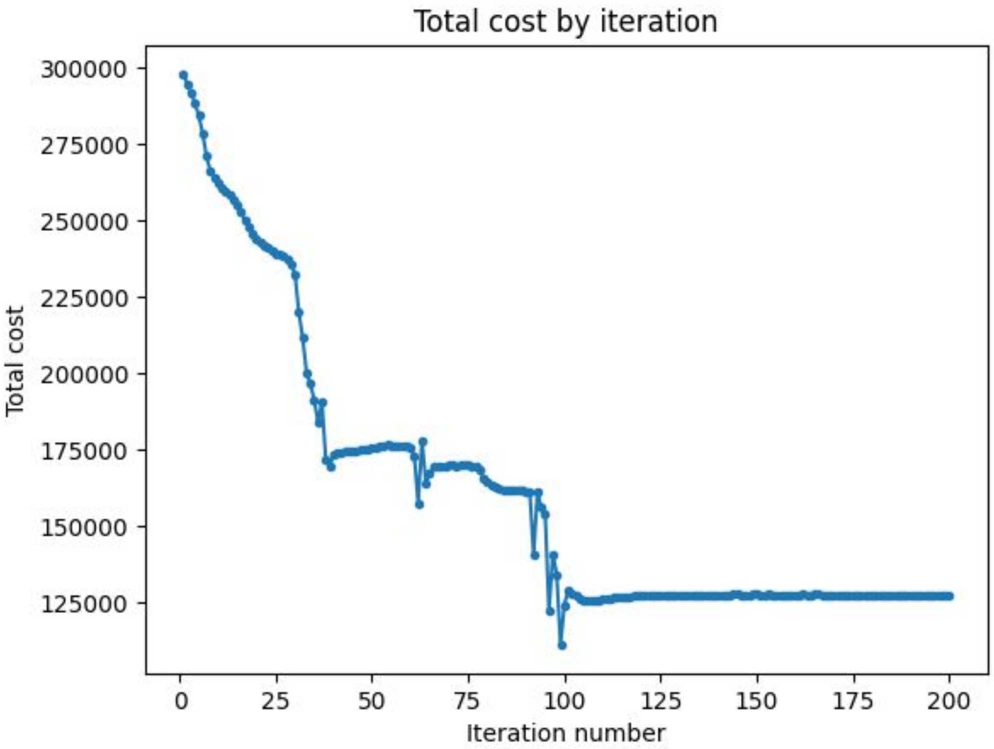


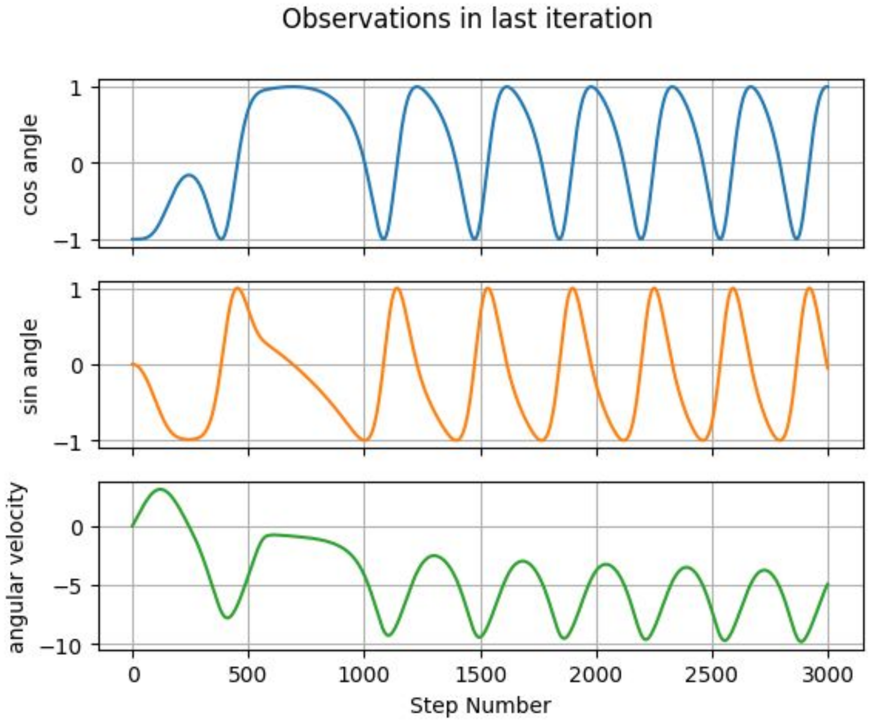
Hidden dim: 32
Hidden layers: 2

lr: 0.005
steps: 1500

N_episode: 2
N_iters: 200

cost function is fixed with velocity cost if angle is small (cosine > 0.95)





Hidden dim: 32
Hidden layers: 2

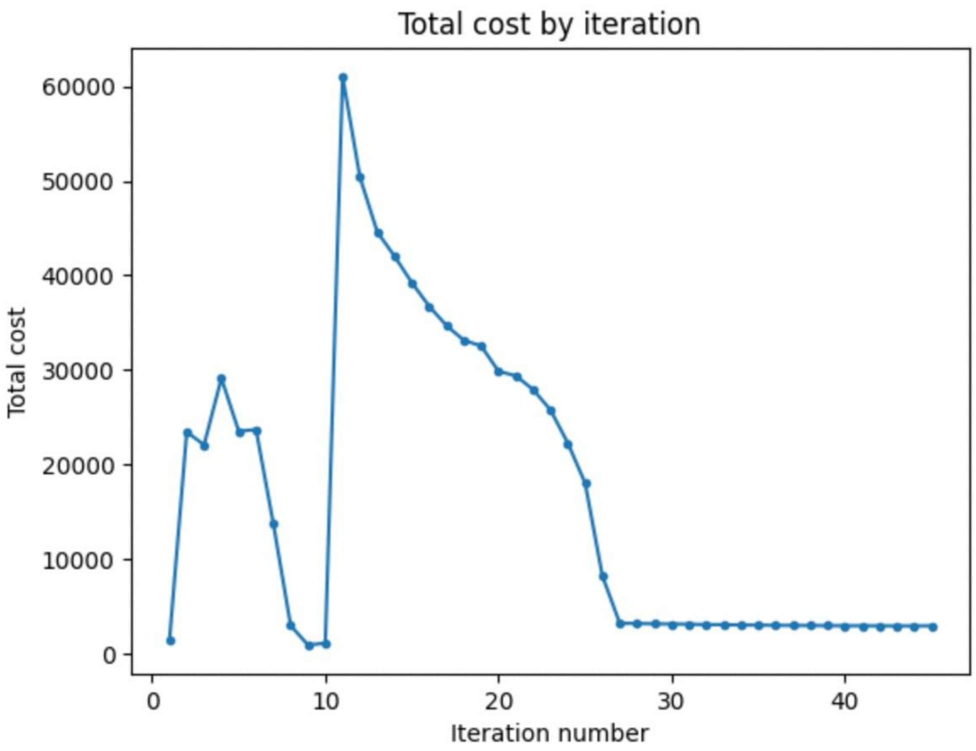
lr: 0.005
steps: 1500

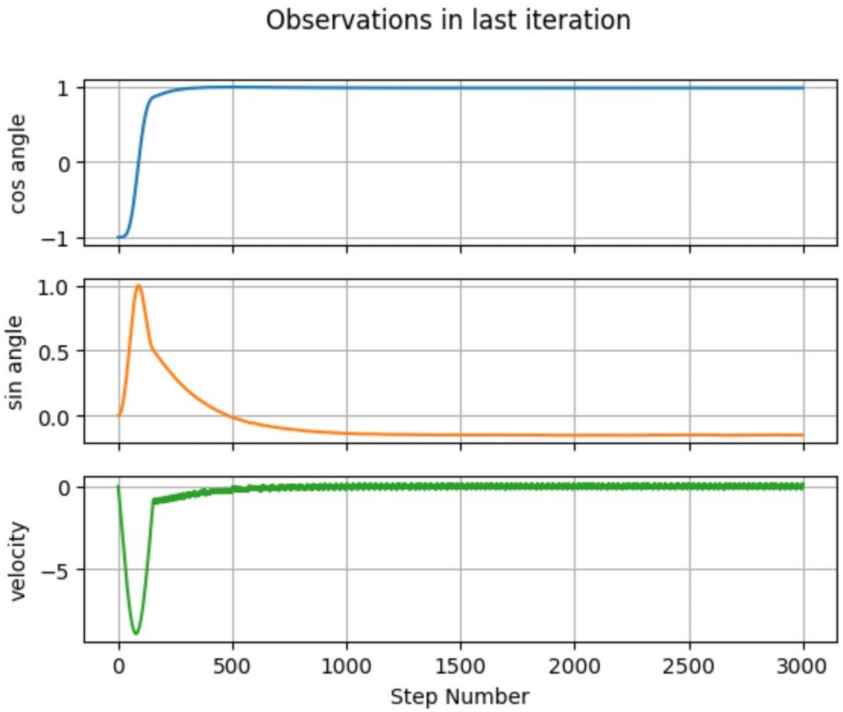
N_episode: 2
N_iters: 200

cost function is $50 * (1 - \cos\theta)$ with velocity cost if angle is small (cosine > 0.95)

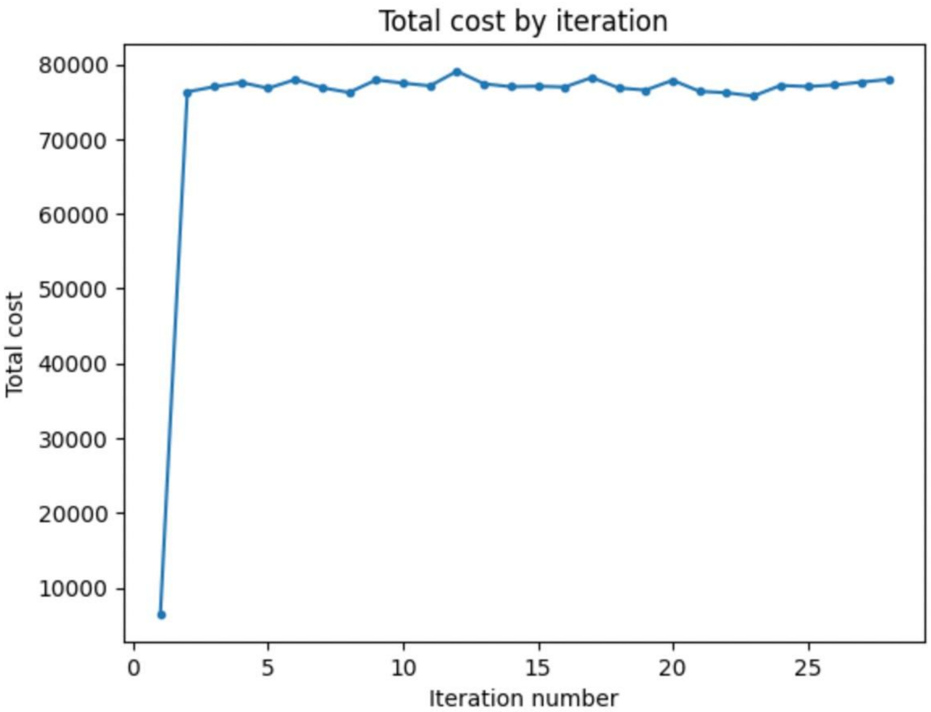
Setup 4

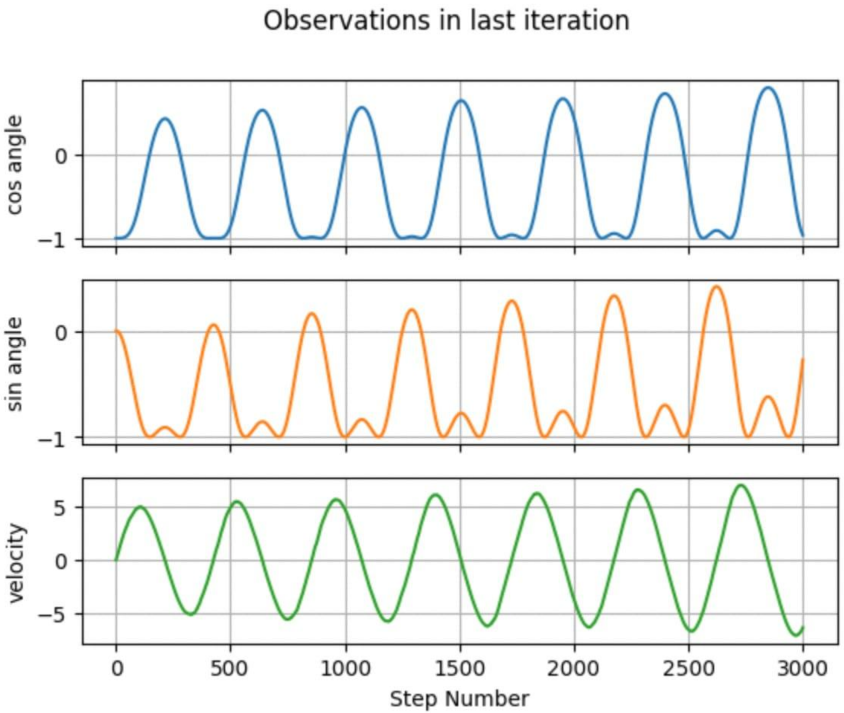
Actor Critic approach. Unlimited surface of cart.





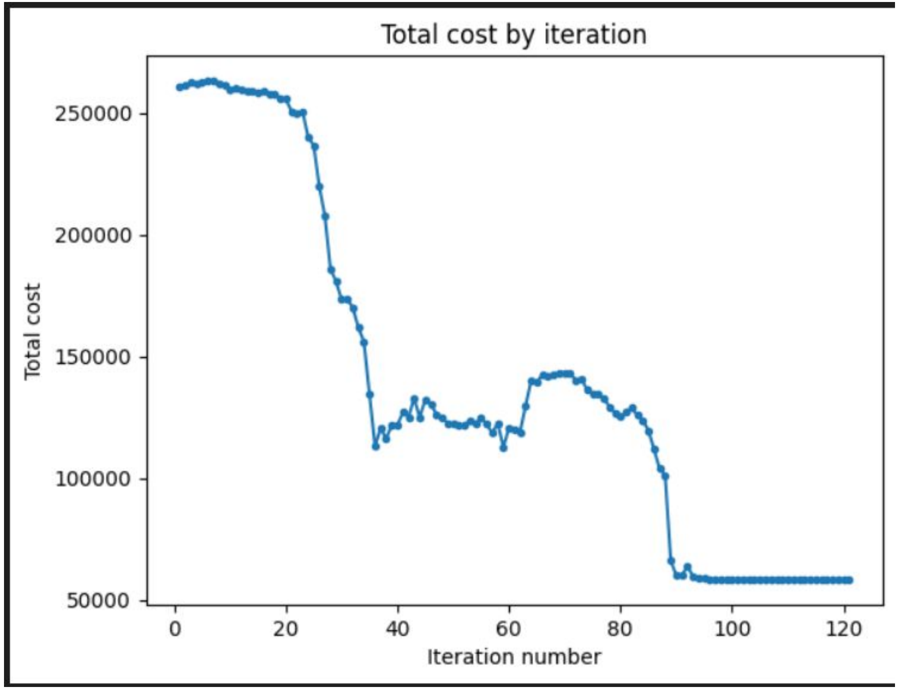
Hidden dim: 64 (both A-C) lr: 0.005 N_episode: 2
Hidden layers: 8 (both A-C) steps: 1500 N_iters: 200
bounds: [-40, 40]
cost function is $(1-\cos\theta)+\omega^2$

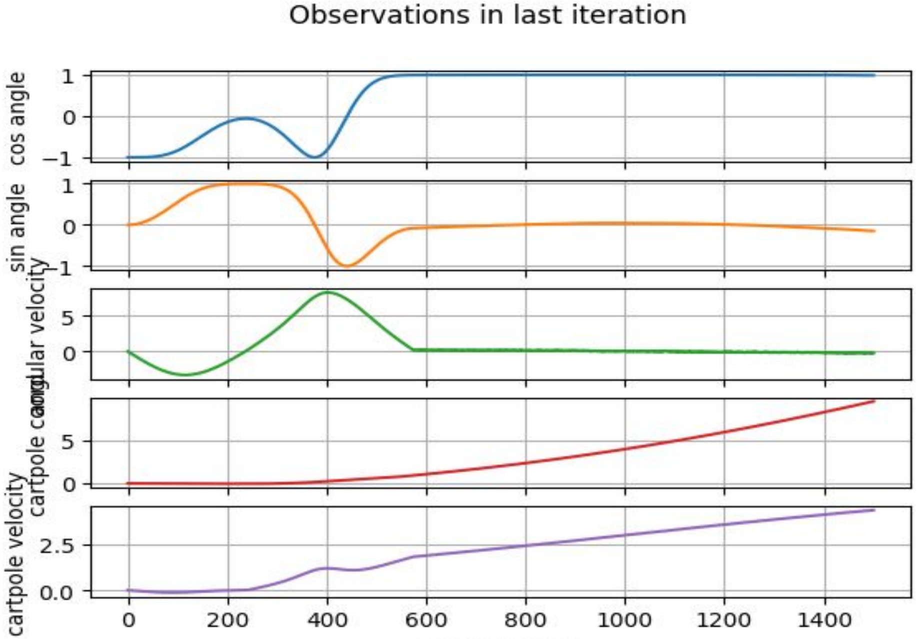




Hidden dim: 64 (both A-C)
lr: 0.005
N_episode: 2
Hidden layers: 8 (both A-C)
steps: 1500
N_iters: 200
bounds: [-40, 40]
alpha: 0.5
cost function is $alpha * (1 - \cos\theta) + (1 - alpha) * \omega^2 + 0.25 * u^2$

Limited Cart Pole (-10, 10)





Hidden dim: 64
Hidden layers: 3

lr: 0.005
steps: 1500

N_episode: 2
N_iters: 200

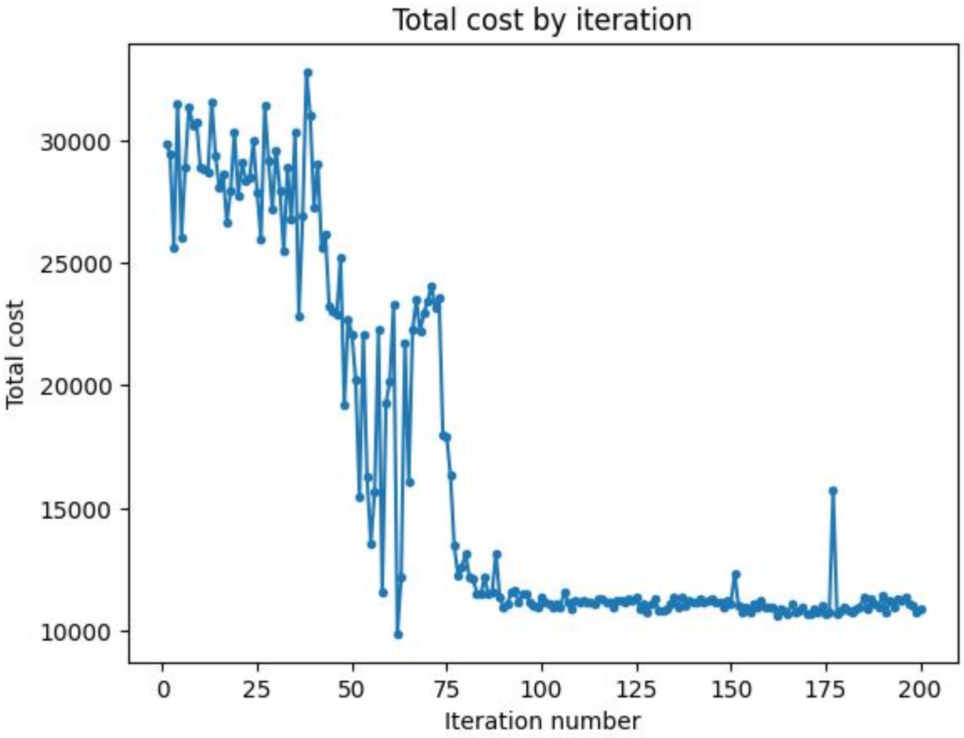
if $\cos\theta$ < 0.9: return $80*(1-\cos\theta)+\omega^2+x^2$
else: return $40*(1-\cos\theta)+\omega^2$

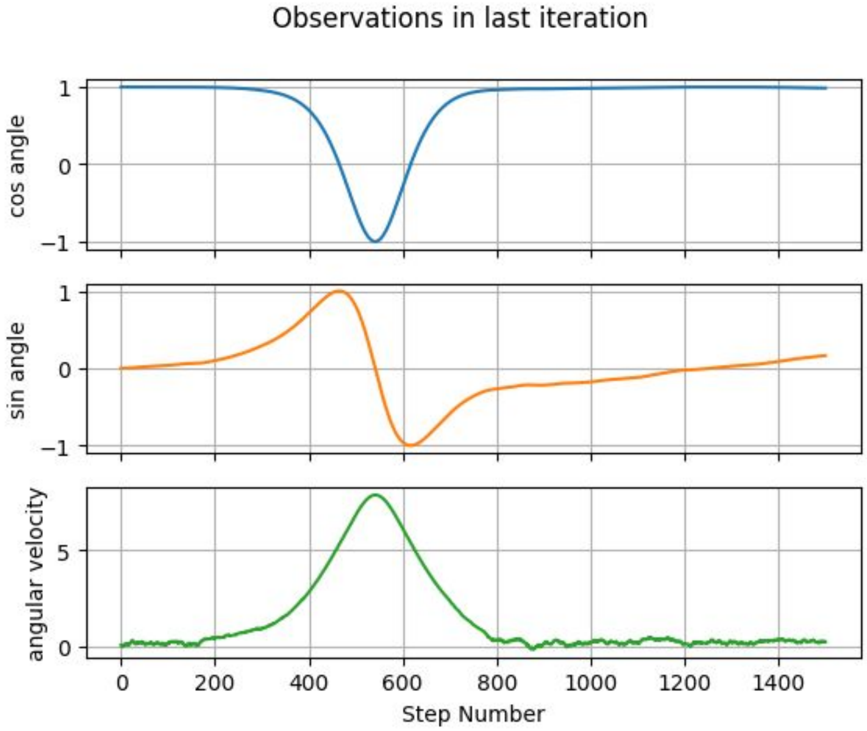
As you may observe, RL suffers with stabilizing system at desired position (angle == 0).

Let’s check, if initial state is already at (0, 0) for unlimited cart pole. And compare with LQR.

LQR formula

$$\dot{x} = \begin{bmatrix} 0 & 1 \\ \frac{(M+m)g}{LM} & 0 \end{bmatrix} x + \begin{bmatrix} 0 \\ -\frac{1}{LM} \end{bmatrix} u$$

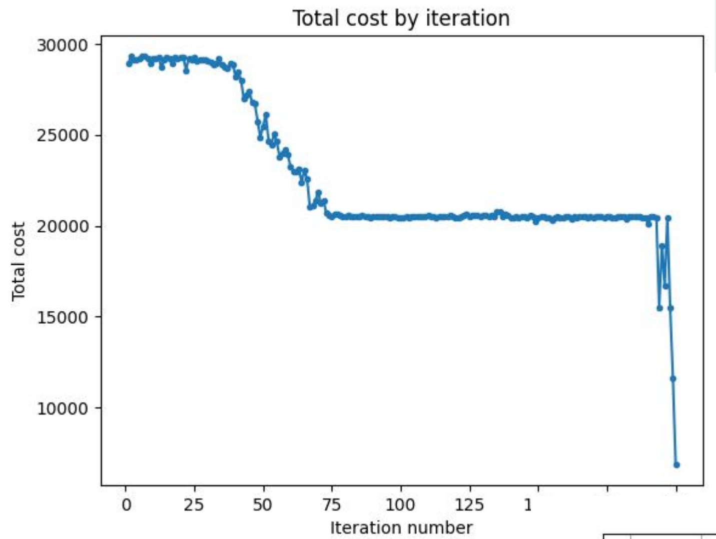




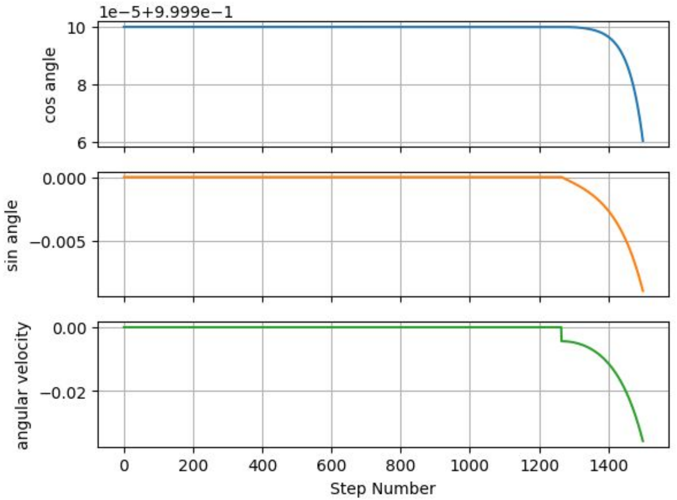
Hidden dim: 32 lr: 0.005 N_episode: 2
Hidden layers: 2 steps: 1800 N_iters: 200

cost fixed with velocity punishment on high cosine.

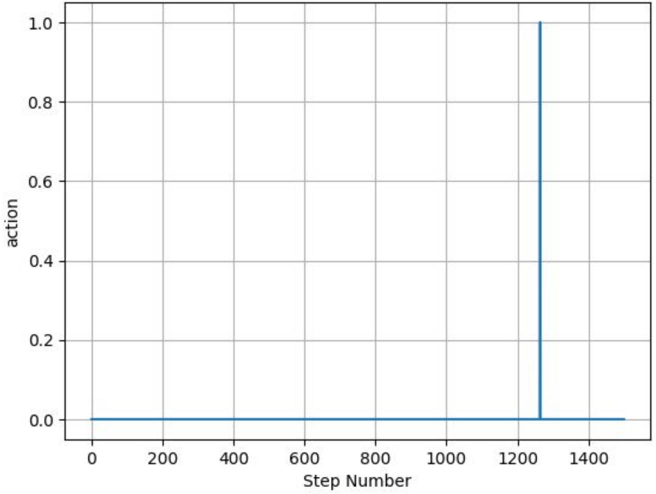
Total cost by iteration



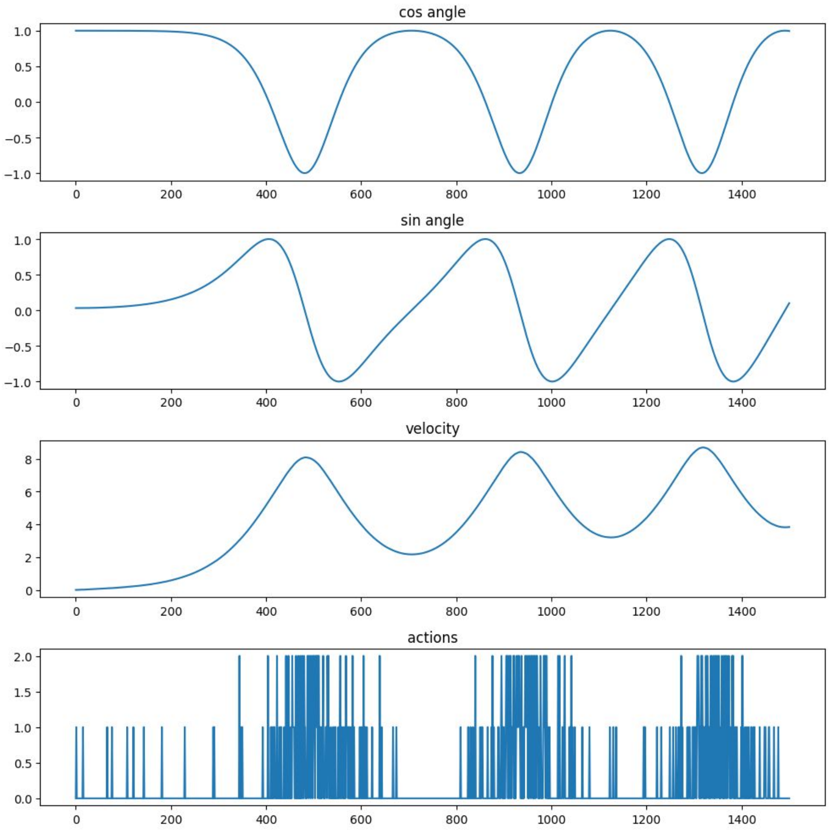
Observations in last iteration

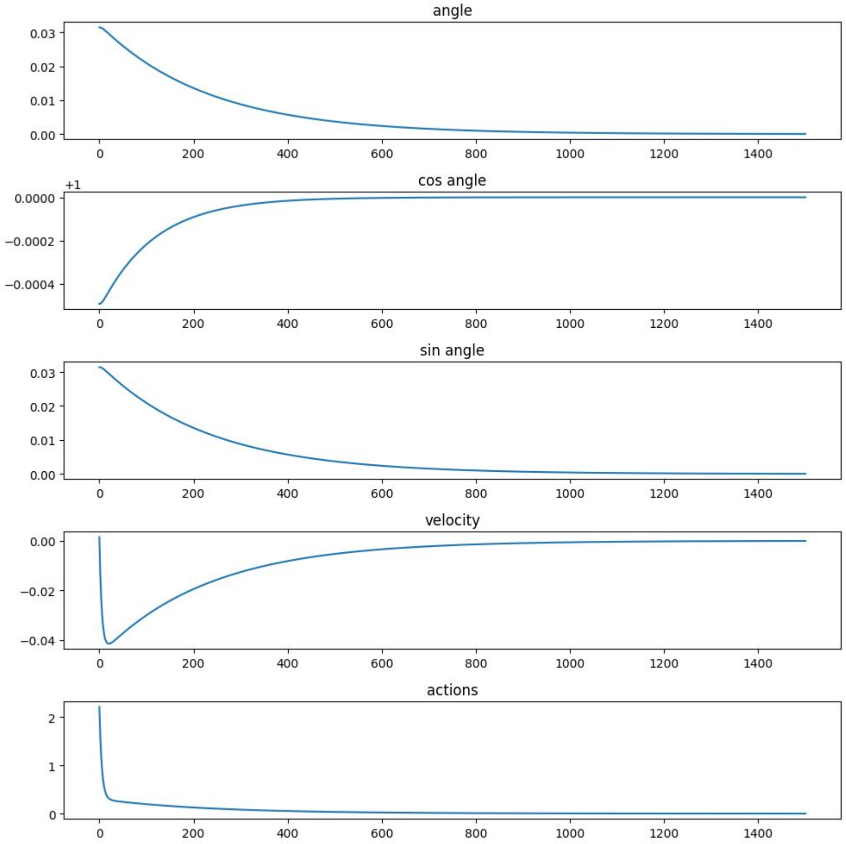


Actions in last iteration



Init state (np.pi / 100, 0)

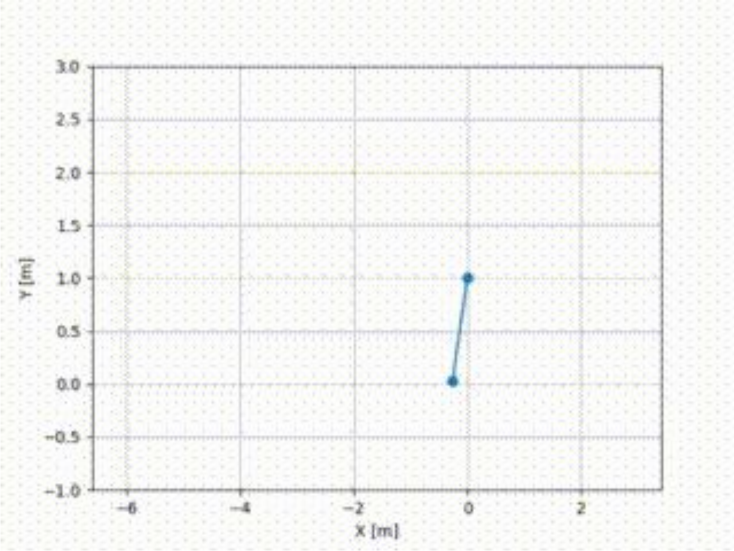




Thank you for your attention!

Ready to answer your questions.

https://github.com/BogChamp/rl_project/tree/master


3.0
2.5
2.0
1.5
1.0
0.5
0.0
−0.5
−1.0
Y [m]
−6
−4
−2
0
2
X [m]