**With the environment uploaded:**

Basic scenario similar to mountainCar optimal one but updated to this new environment configuration:

#It must be multiple of 10

MAX\_RUNS=1500

MAXIMUM\_STEPS=700

#This constant is used for showing one q\_values matrix each INTERPOLATION runs

INTERPOLATION=MAX\_RUNS/10

GAMMA = 0.95

LEARNING\_RATE = 0.25

EXPLORATION\_MAX = 1.0

EXPLORATION\_MIN = 0.05

EXPLORATION\_DECAY = 0.9995

EXPLORATION\_RUNS=1

DISCRETIZATION\_POS=1/50

DISCRETIZATION\_SPEED=1/100

def get\_reward(state, step, done):

if done:

print("Car has reached the goal")

return 500

elif state[0]<4000:

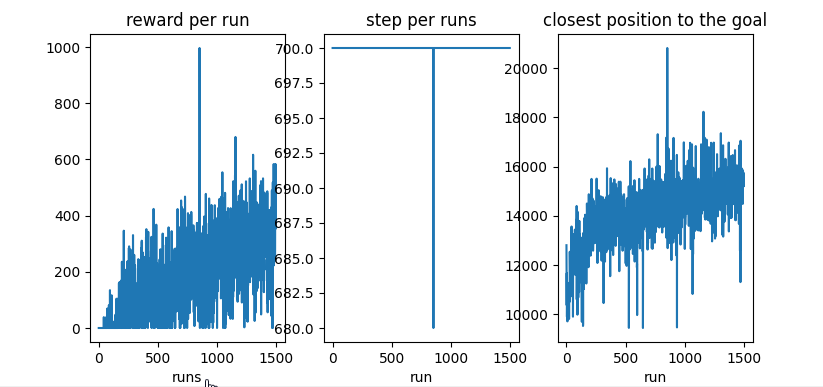
return (abs(state[0]))/10000

elif state[0]>=4000 and state[0]<13000:

return 0

elif state[0]>=13000:

return (state[0]/10000)\*\*3

****

Softer learning rate and reward function with less rewarded area (higher rewarder locations)

#It must be multiple of 10

MAX\_RUNS=700

MAXIMUM\_STEPS=1000

#This constant is used for showing one q\_values matrix each INTERPOLATION runs

INTERPOLATION=MAX\_RUNS/10

GAMMA = 0.95

LEARNING\_RATE = 0.2

EXPLORATION\_MAX = 1.0

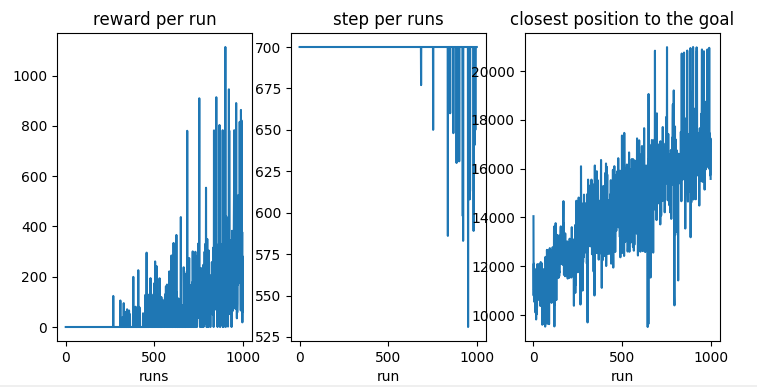
EXPLORATION\_MIN = 0.05

EXPLORATION\_DECAY = 0.9995

EXPLORATION\_RUNS=1

DISCRETIZATION\_POS=1/50

DISCRETIZATION\_SPEED=1/100



Same but more steps per run:

MAX\_RUNS=700

MAXIMUM\_STEPS=1000

