THE RIGHT-TURN RACER

A foray into Monte Carlo Methods

PROBLEM STATEMENT

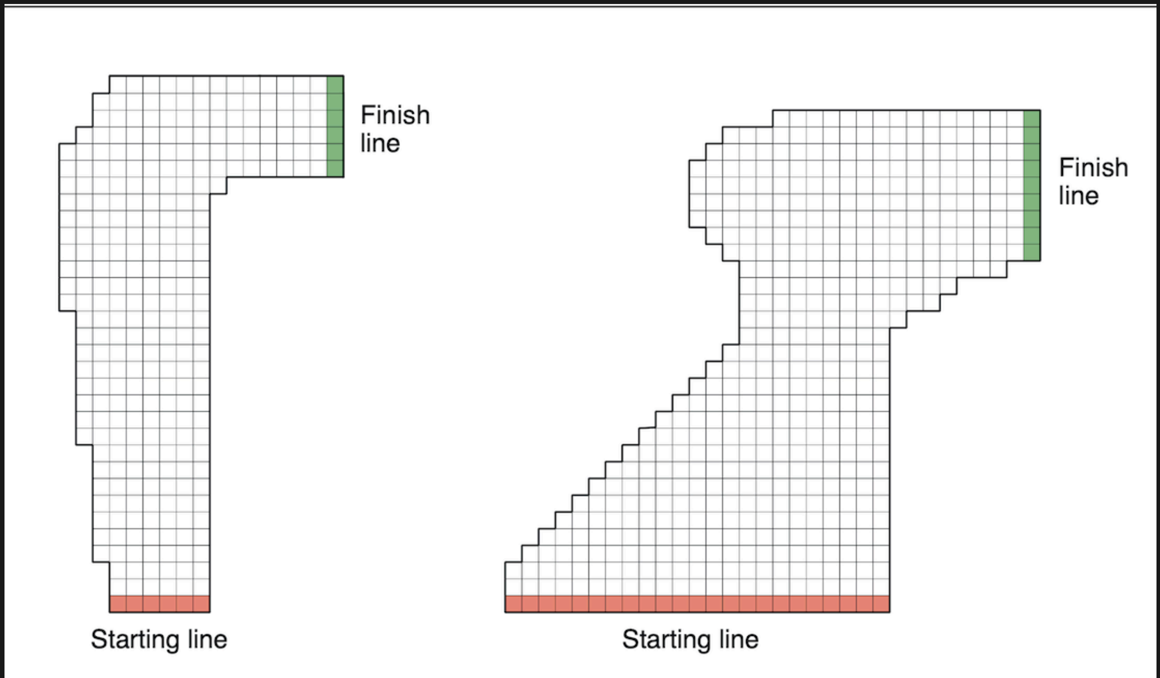


Figure 5.5: A couple of right turns for the racetrack task.

- Car has velocity range of 0 to 4, but can only change by -1 to 1
- Car can't stall (velocity vector goes to 0,0)

SOLUTION

Implement *Monte Carlo method* using an *off-policy* strategy with *weighted importance* sampling.

MONTE CARLO METHOD

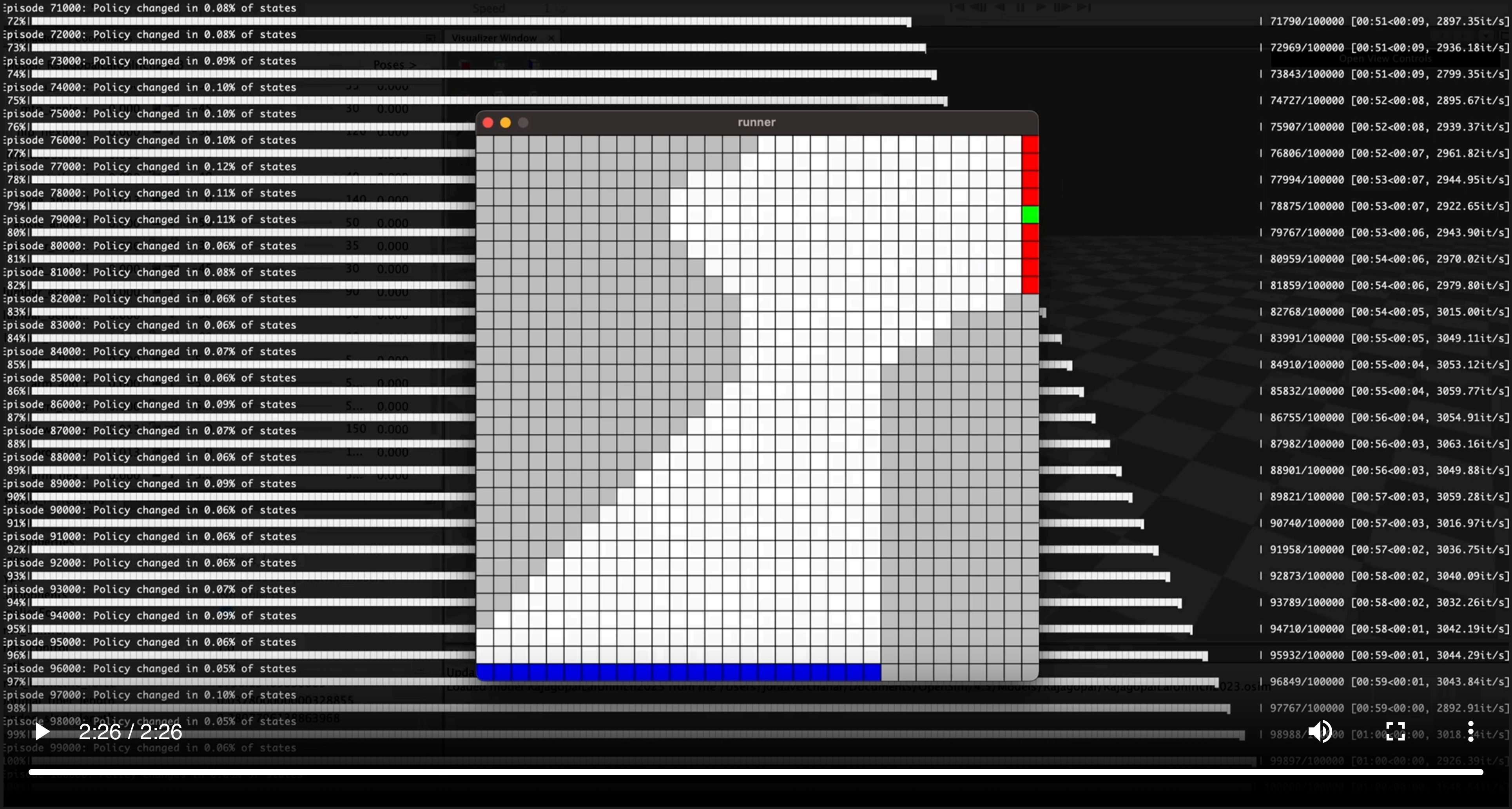
- having a model of the world, but no understanding of its *dynamics*
- Must *sample* it rather than raw dog it with math

OFF-POLICY STRATEGY

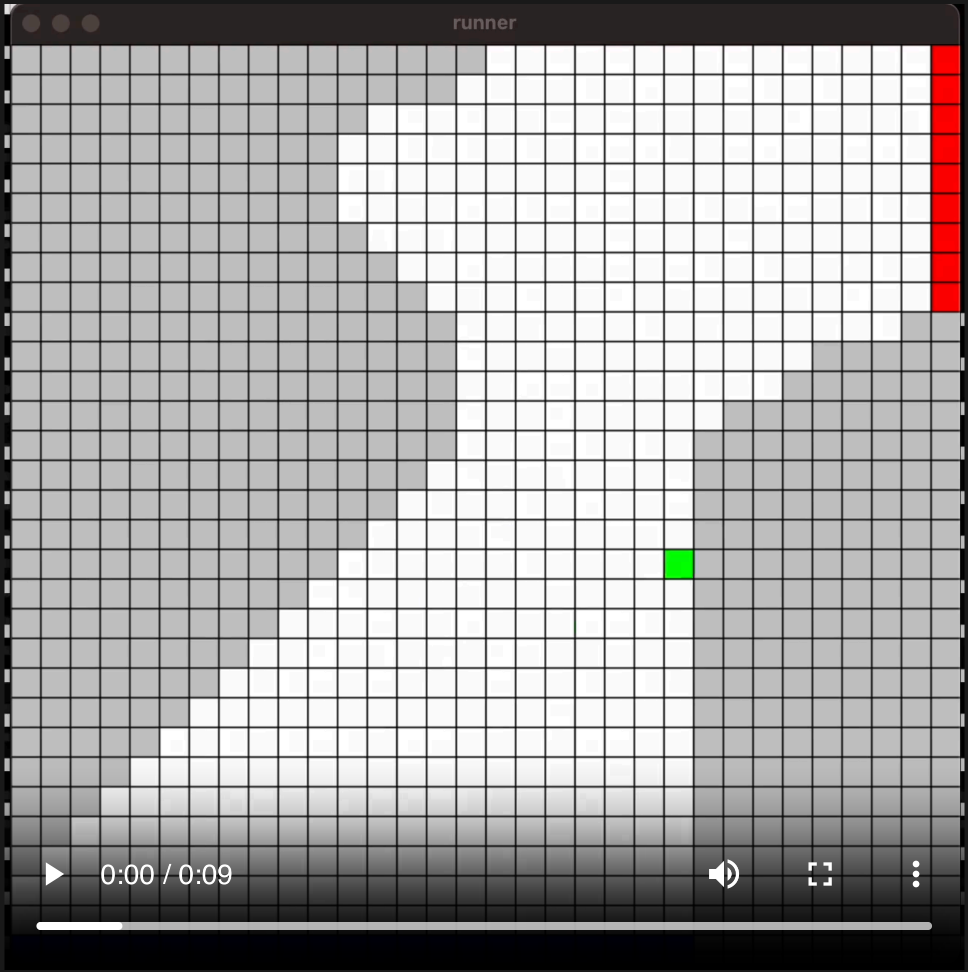
- the way I choose to *explore* the world will be different than how I choose to *exploit* it
- explore: *behavior policy*
- exploit: *target policy*

WEIGHTED IMPORTANCE SAMPLING

- my likelihood of seeing the return from this state while exploring is *different* than with my target policy by some *sampling ratio*


Episode 71000: Policy changed in 0.08% of states
72%| | 71790/100000 [00:51<00:09, 2897.35it/s]
Episode 72000: Policy changed in 0.08% of states
73%| | 72969/100000 [00:51<00:09, 2936.18it/s]
Episode 73000: Policy changed in 0.09% of states
74%| | 73843/100000 [00:51<00:09, 2799.35it/s]
Episode 74000: Policy changed in 0.10% of states
75%| | 74727/100000 [00:52<00:08, 2895.67it/s]
Episode 75000: Policy changed in 0.10% of states
76%| | 75907/100000 [00:52<00:08, 2939.37it/s]
Episode 76000: Policy changed in 0.10% of states
77%| | 76806/100000 [00:52<00:07, 2961.82it/s]
Episode 77000: Policy changed in 0.12% of states
78%| | 77994/100000 [00:53<00:07, 2944.95it/s]
Episode 78000: Policy changed in 0.11% of states
79%| | 78875/100000 [00:53<00:07, 2922.65it/s]
Episode 79000: Policy changed in 0.11% of states
80%| | 79767/100000 [00:53<00:06, 2943.90it/s]
Episode 80000: Policy changed in 0.06% of states
81%| | 80959/100000 [00:54<00:06, 2970.02it/s]
Episode 81000: Policy changed in 0.08% of states
82%| | 81859/100000 [00:54<00:06, 2979.80it/s]
Episode 82000: Policy changed in 0.06% of states
83%| | 82768/100000 [00:54<00:05, 3015.00it/s]
Episode 83000: Policy changed in 0.06% of states
84%| | 83991/100000 [00:55<00:05, 3049.11it/s]
Episode 84000: Policy changed in 0.07% of states
85%| | 84910/100000 [00:55<00:04, 3053.12it/s]
Episode 85000: Policy changed in 0.06% of states
86%| | 85832/100000 [00:55<00:04, 3059.77it/s]
Episode 86000: Policy changed in 0.09% of states
87%| | 86755/100000 [00:56<00:04, 3054.91it/s]
Episode 87000: Policy changed in 0.07% of states
88%| | 87982/100000 [00:56<00:03, 3063.16it/s]
Episode 88000: Policy changed in 0.06% of states
89%| | 88901/100000 [00:56<00:03, 3049.88it/s]
Episode 89000: Policy changed in 0.09% of states
90%| | 89821/100000 [00:57<00:03, 3059.28it/s]
Episode 90000: Policy changed in 0.06% of states
91%| | 90740/100000 [00:57<00:03, 3016.97it/s]
Episode 91000: Policy changed in 0.06% of states
92%| | 91958/100000 [00:57<00:02, 3036.75it/s]
Episode 92000: Policy changed in 0.06% of states
93%| | 92873/100000 [00:58<00:02, 3040.09it/s]
Episode 93000: Policy changed in 0.07% of states
94%| | 93789/100000 [00:58<00:02, 3032.26it/s]
Episode 94000: Policy changed in 0.09% of states
95%| | 94710/100000 [00:58<00:01, 3042.19it/s]
Episode 95000: Policy changed in 0.06% of states
96%| | 95932/100000 [00:59<00:01, 3044.29it/s]
Episode 96000: Policy changed in 0.05% of states
97%| | 96849/100000 [00:59<00:01, 3043.84it/s]
Episode 97000: Policy changed in 0.10% of states
98%| | 97767/100000 [00:59<00:00, 2892.91it/s]
Episode 98000: Policy changed in 0.05% of states
Episode 99000: Policy changed in 0.06% of states
99897/100000 [01:00<00:00, 2926.39it/s]
runner
2:26 / 2:26

But what does this look like while training?


runner
0:00 / 0:09

WHAT PROBLEMS WERE SOLVED FOR ME?

Reward Design

- Decision was to reward -1 per timestep until finish was reached
- Banging into side of track means random restart on start line, NOT END OF EPISODE

WHAT PROBLEMS DID I HAVE?

State Design

Using a simple array to represent the grid made the state space SPARSE, confusing me about the effectiveness of the algorithm

```
(venv) (base) joraaverchahal@Joraavers-MacBook-Pro ReinforcementLearningSuttonBarto % python racetrack_agent.py
  1%|█
pisode 1000: Policy changed in 6.79% of states
  2%|██
pisode 2000: Policy changed in 3.21% of states
  3%|███
pisode 3000: Policy changed in 2.08% of states
  4%|████
pisode 4000: Policy changed in 1.84% of states
  5%|█████
pisode 5000: Policy changed in 1.34% of states
  6%|██████
pisode 6000: Policy changed in 1.46% of states
  7%|███████
pisode 7000: Policy changed in 1.26% of states
  8%|████████
pisode 8000: Policy changed in 1.00% of states
  9%|█████████
pisode 9000: Policy changed in 1.14% of states
 10%|██████████
pisode 10000: Policy changed in 1.09% of states
 11%|███████████
pisode 11000: Policy changed in 0.95% of states
 12%|████████████
pisode 12000: Policy changed in 0.82% of states
 13%|█████████████
pisode 13000: Policy changed in 0.86% of states
 14%|██████████████
pisode 14000: Policy changed in 0.78% of states
 15%|███████████████
pisode 15000: Policy changed in 0.73% of states
 16%|████████████████
pisode 16000: Policy changed in 0.64% of states
 17%|█████████████████
pisode 17000: Policy changed in 0.75% of states
 18%|██████████████████
pisode 18000: Policy changed in 0.61% of states
 19%|███████████████████
pisode 19000: Policy changed in 0.61% of states
 20%|████████████████████
pisode 20000: Policy changed in 0.63% of states
 21%|█████████████████████
pisode 21000: Policy changed in 0.74% of states
 22%|██████████████████████
pisode 22000: Policy changed in 0.64% of states
 23%|███████████████████████
pisode 23000: Policy changed in 0.67% of states
 24%|████████████████████████
pisode 24000: Policy changed in 0.62% of states
 25%|█████████████████████████
pisode 25000: Policy changed in 0.52% of states
 26%|██████████████████████████
pisode 26000: Policy changed in 0.55% of states
 26%|██████████████████████████
```

```
 85%|████████████████████████████████████████████████
Episode 85000: Policy changed in 19.78% of states
 86%|████████████████████████████████████████████████
Episode 86000: Policy changed in 19.78% of states
 87%|████████████████████████████████████████████████
Episode 87000: Policy changed in 19.78% of states
 88%|████████████████████████████████████████████████
Episode 88000: Policy changed in 19.78% of states
 89%|████████████████████████████████████████████████
Episode 89000: Policy changed in 19.79% of states
 90%|████████████████████████████████████████████████
Episode 90000: Policy changed in 19.79% of states
 91%|████████████████████████████████████████████████
Episode 91000: Policy changed in 19.80% of states
 92%|████████████████████████████████████████████████
Episode 92000: Policy changed in 19.80% of states
 93%|████████████████████████████████████████████████
Episode 93000: Policy changed in 19.80% of states
 94%|████████████████████████████████████████████████
Episode 94000: Policy changed in 19.81% of states
 95%|████████████████████████████████████████████████
Episode 95000: Policy changed in 19.81% of states
 96%|████████████████████████████████████████████████
Episode 96000: Policy changed in 19.81% of states
 97%|████████████████████████████████████████████████
Episode 97000: Policy changed in 19.83% of states
 98%|████████████████████████████████████████████████
Episode 98000: Policy changed in 19.83% of states
 99%|████████████████████████████████████████████████
Episode 99000: Policy changed in 19.83% of states
100%|████████████████████████████████████████████████
Episode 100000: Policy changed in 19.84% of states
100%|████████████████████████████████████████████████
```

WHAT PROBLEMS DID I HAVE?

Q-value Initialization

Off-policy MC control, for estimating $\pi \approx \pi_*$

Initialize, for all $s \in \mathcal{S}$, $a \in \mathcal{A}(s)$:
    $Q(s, a) \in \mathbb{R}$ (arbitrarily)
    $C(s, a) \leftarrow 0$
    $\pi(s) \leftarrow \arg\max_a Q(s, a)$ (with ties broken consistently)

Loop forever (for each episode):
    $b \leftarrow$ any soft policy
    Generate an episode using b: $S_0, A_0, R_1, \ldots, S_{T-1}, A_{T-1}, R_T$
    $G \leftarrow 0$
    $W \leftarrow 1$
    Loop for each step of episode, $t = T-1, T-2, \ldots, 0$:
        $G \leftarrow \gamma G + R_{t+1}$
        $C(S_t, A_t) \leftarrow C(S_t, A_t) + W$
        $Q(S_t, A_t) \leftarrow Q(S_t, A_t) + \frac{W}{C(S_t, A_t)} [G - Q(S_t, A_t)]$
        $\pi(S_t) \leftarrow \arg\max_a Q(S_t, a)$ (with ties broken consistently)
        If $A_t \neq \pi(S_t)$ then exit inner Loop (proceed to next episode)
        $W \leftarrow W \frac{1}{b(A_t|S_t)}$

WHAT PROBLEMS DID I HAVE?

Q-value Initialization

- The value estimator for the best next action is tied to the reward
- Initializing only positive values made episodes learning useless with a negative only reward

Q-value Initialization

- Q = [30, 10, 60] at start
  - max is 60, action = 2
- Q = [30, -1, 60] after evaluating final step
  - max is 60, action = 2
    - best action != action taken, ignore rest of episode
- continue until you play russian roulette

FIN

Code available here

Presentation courtesy of reveal.js