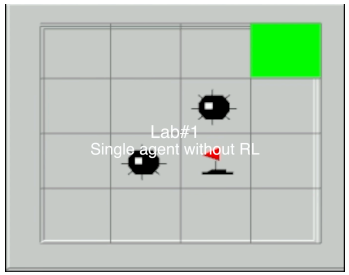
[ECE277_Final_Project]

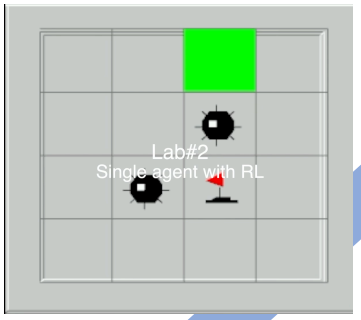
Finding multiple flags
Q-learning vs. Monte Carlo (MC) learning

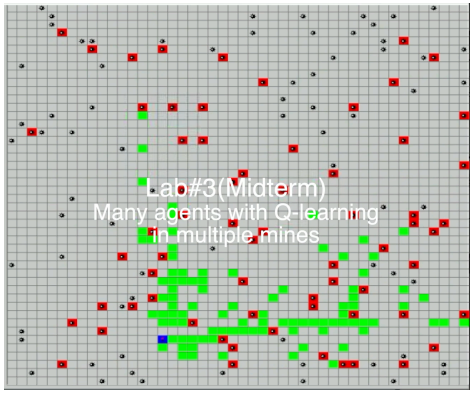
Haenara SHIN
Ph.D. student/M.S. student
MATS/ECE
University of California, San Diego

Motivation

Q-learning in Lab1, 2, 3(Midterm)
: Finding ONE flag from ONE flag environment with single agent/many agents







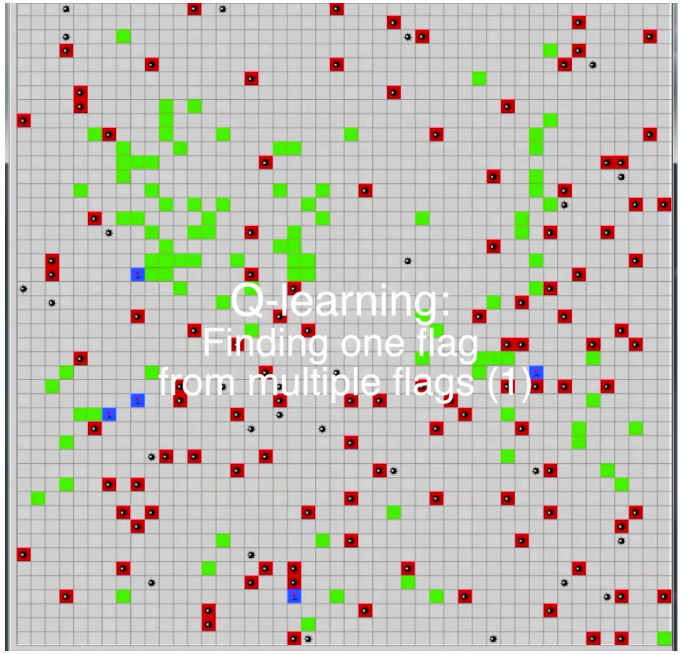
Q-learning in Finding ONE flag from MANY flags environment

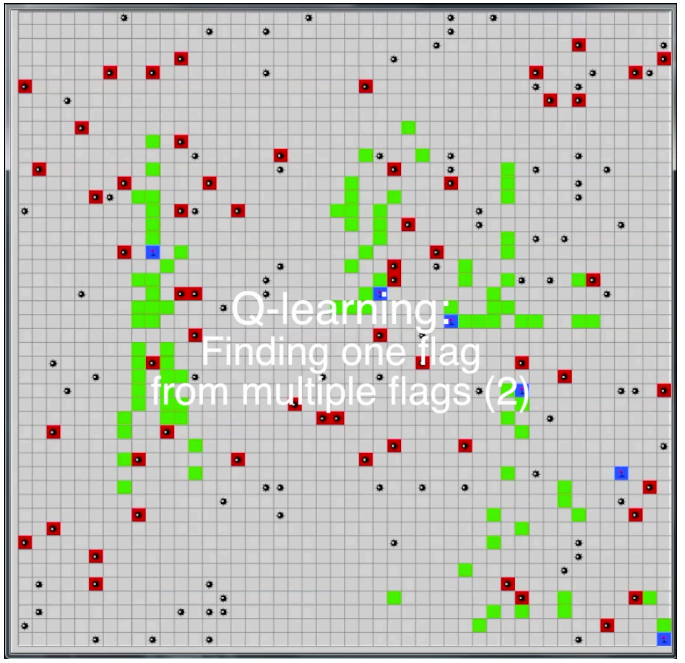
Q-learning in Finding MANY flags from MANY flags environment

Another RL algorithm in Finding MANY flags from MANY flags environment

Limitation of Q-learning in finding multiple flags environment

Q-learning in Finding **ONE** flag from **MANY** flags environment

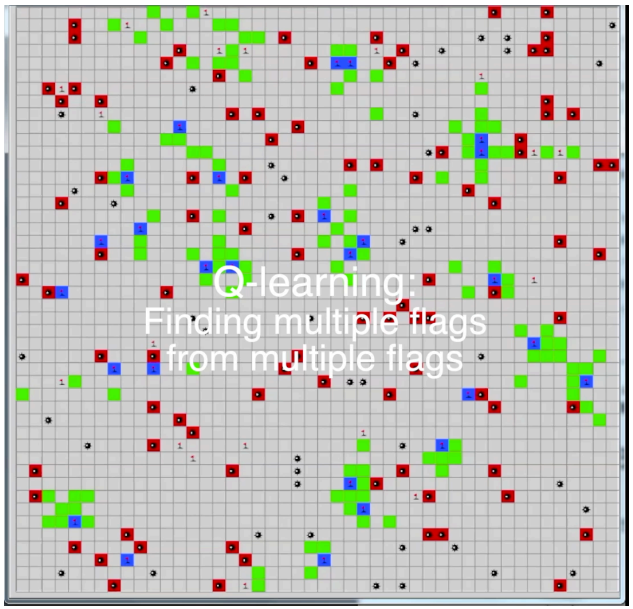




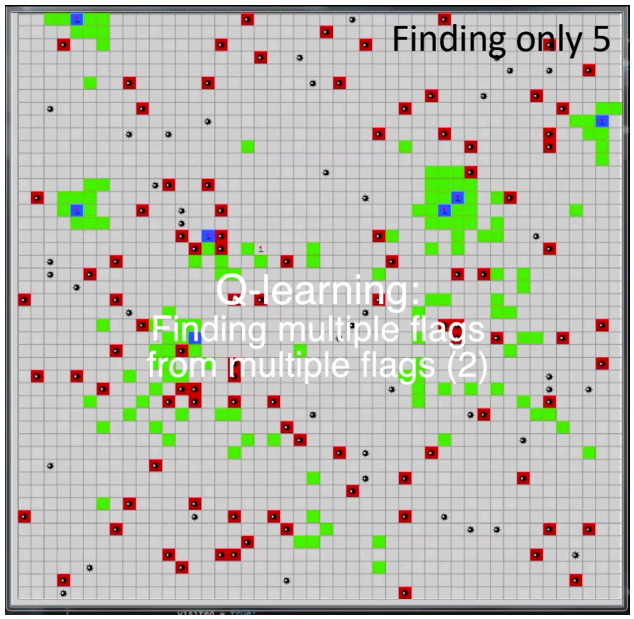
- Many agents are moving to their neighbor flags.
  - More flags → Short convergence routes or higher chance to find any flag

Limitation of Q-learning in finding multiple flags environment

Q-learning in Finding MANY flags from MANY flags environment



Lessen flags



- Finding multiple flags from multiple flags environment
  - Agents seems to just move (or, oscillate) around the flags.
  - They have short-sights, which is opposite what we thought that agents should ‘ explore‘ longer distance to pursue the higher rewards.

MC learning (Algorithm and approaching method to the problem) [4]

Initialize, for all $s \in \mathcal{S}$, $a \in \mathcal{A}(s)$:
$Q(s,a) \leftarrow$ arbitrary
$Returns(s,a) \leftarrow$ empty list
$\pi \leftarrow$ an arbitrary ε-soft policy

Repeat forever:
(a) Generate an episode using π
(b) For each pair s, a appearing in the episode:
$R \leftarrow$ return following the first occurrence of s, a
Append R to $Returns(s,a)$
$Q(s,a) \leftarrow$ average$(Returns(s,a))$
(c) For each s in the episode:
$a^* \leftarrow \arg\max_a Q(s,a)$
For all $a \in \mathcal{A}(s)$:

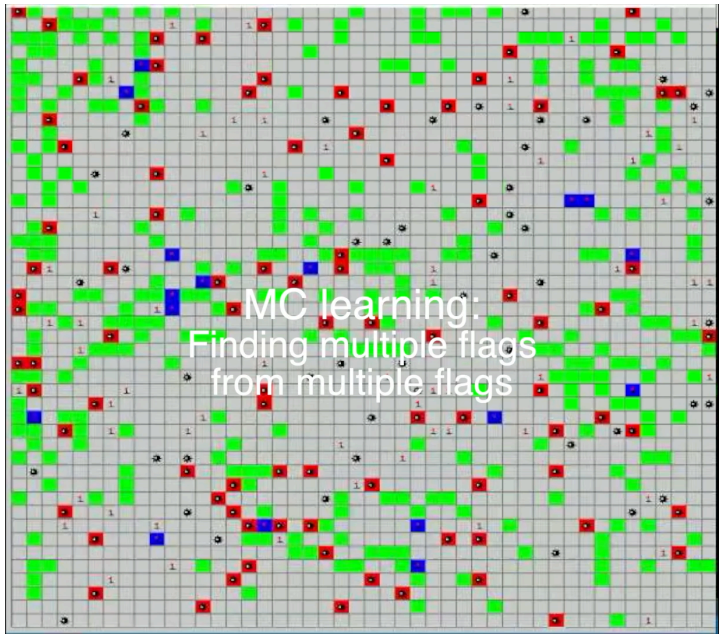
$$\pi(a,s) = \begin{cases} max(A^*(s,a)), & \text{if random} > \epsilon \\ \text{random}(A(s)), & \text{otherwise} \end{cases}$$

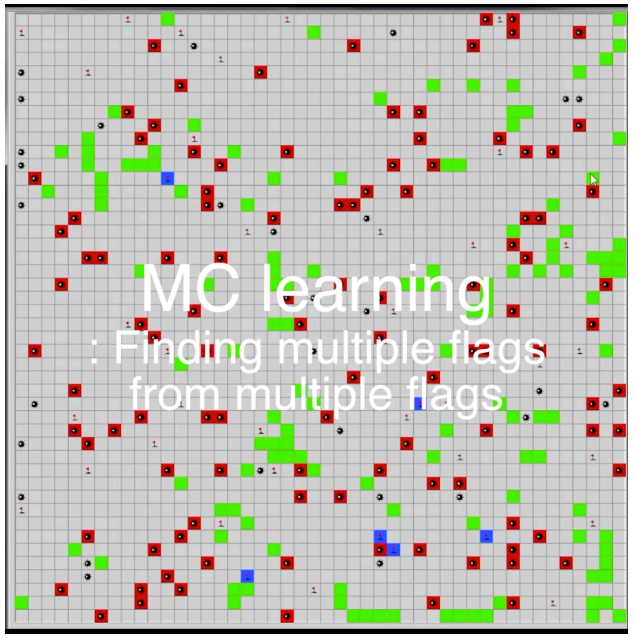
$$N(S,A) \leftarrow N(S,A) + 1$$

$$Q(S,A) \leftarrow Q(S,A) + \frac{1}{N(S,A)}(Reward\ Until\ the\ end\ - Q(S,A)$$

Result (MC learning for finding multiple flags)

- Agents are trying to find other flags (less-oscillation/near-sighted).
  - However, they do not find the closet flag as well.
  - The overall performance are lower than our expectation.





AC, A3C, A2C, SAC, PPO…

DQN or other variations.

- This is because every single episode should be ended first, then update will be done. If the case that the episode is not ended, or the case that the episode is long, (ex., gaming like Starcraft), the learning is difficult.

Comparison the Q-learning and Monte Carlo (MC) learning [1-9]

Q-learning (Time(temporal)-Difference learning)		MC learning
▪ Can learn online after every step and does not need to wait until the end of episode ▪ Exploits the Markov property ▪ Off-policy temporal-difference control ▪ Epsilon-greedy policy(action) + max-Q value(learning)	**Good**	▪ Zero bias ▪ Good convergence properties even with function approximation ▪ Not very sensitive to initial value. ▪ Does not exploit the Markov property
▪ Bootstrapping value.	**Bad**	▪ MC must wait until the end of the episode before the return is known. ▪ High variance ▪ Can only learn from complete sequence (slow) ▪ Only works for episodic(terminating) environments (slow)

$$V(S_t) \leftarrow V(S_t) + \alpha\Big(R_{t+1} + \gamma V(S_{t+1}) - V(S_t)\Big)$$

$$V(S_t) \leftarrow V(S_t) + \alpha\Big(G_t - V(S_t)\Big)$$

$$Q(S,A) \leftarrow Q(S,A) + \alpha\Big(R + \gamma \max_{a'} Q(S',A') - Q(S,A)\Big)$$

Reference.

- [1] https://www.oreilly.com/library/view/statistics-for-machine/9781788295758/e8f0bbd1-b60a-47d2-9cd4-0d0661fc9212.xhtml
- [2] https://medium.com/ai³-theory-practice-business/reinforcement-learning-part-5-monte-carlo-and-temporal-difference-learning-889053aba07d
- [3] https://towardsdatascience.com/monte-carlo-learning-b83f75233f92
- [4] https://adityajain.me/blogs/monte-carlo-and-temporal-difference.html
- [5] https://github.com/wonseokjung/ReinforcementLearning_byWonseok
- [6] https://www.slideshare.net/LeejinJeong/mdp-montecarlo-timedifference-sarsa-qlearning-1
- [7] https://www.slideshare.net/DongMinLee32/part-2-91522217
- [8] https://daeson.tistory.com/327
- [9] https://codereview.stackexchange.com/questions/158957/simple-minesweeper-using-opengl-glut
- [10] https://github.com/WoodyWangHou/CUDA/tree/master/lab2/Src/Qlearning

Thank you!