**CS 4641 PROJECT 4**

Name: CHANG Yingshan

GT Account: ychang363

GT ID: 903457645

# Problem Description

1. Reinforcement learning

Reinforcement learning is a type of Machine Learning, which allows agents to automatically determine the ideal behavior within a specific context, in order to maximize its “reward”. It is characterized by a process of agents observing the environment output consisting of a reward and the next state, and then acting upon that.

1. Markov Decision Process

Markov Decision Process formally describe an environment for reinforcement learning, where the environment should be fully observable. In any MDP problem, we assume the Markov Property: the effects of an action taken in a state depend only on the current state and not on the prior history. The following items are given in a typical MDP model:

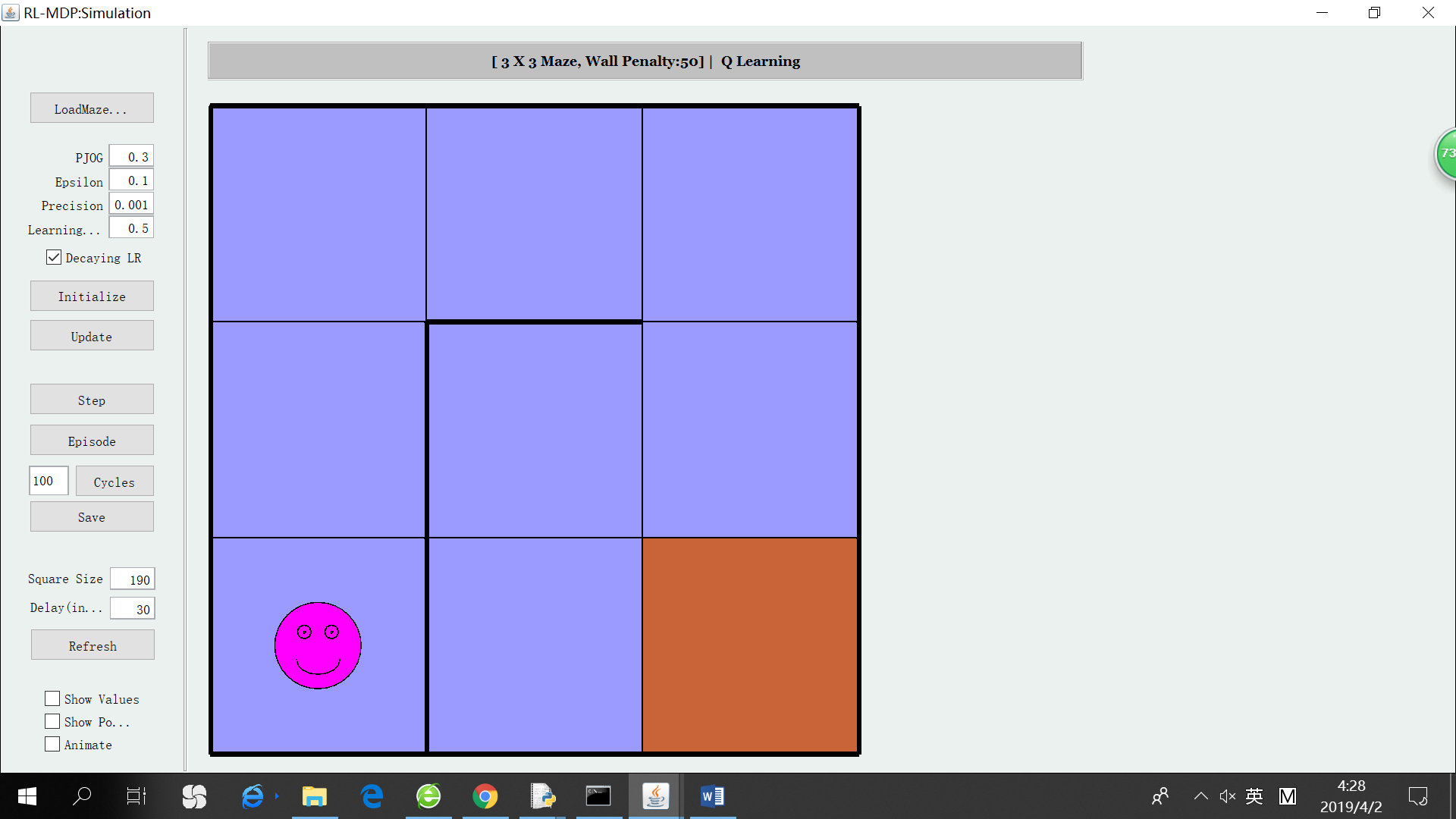
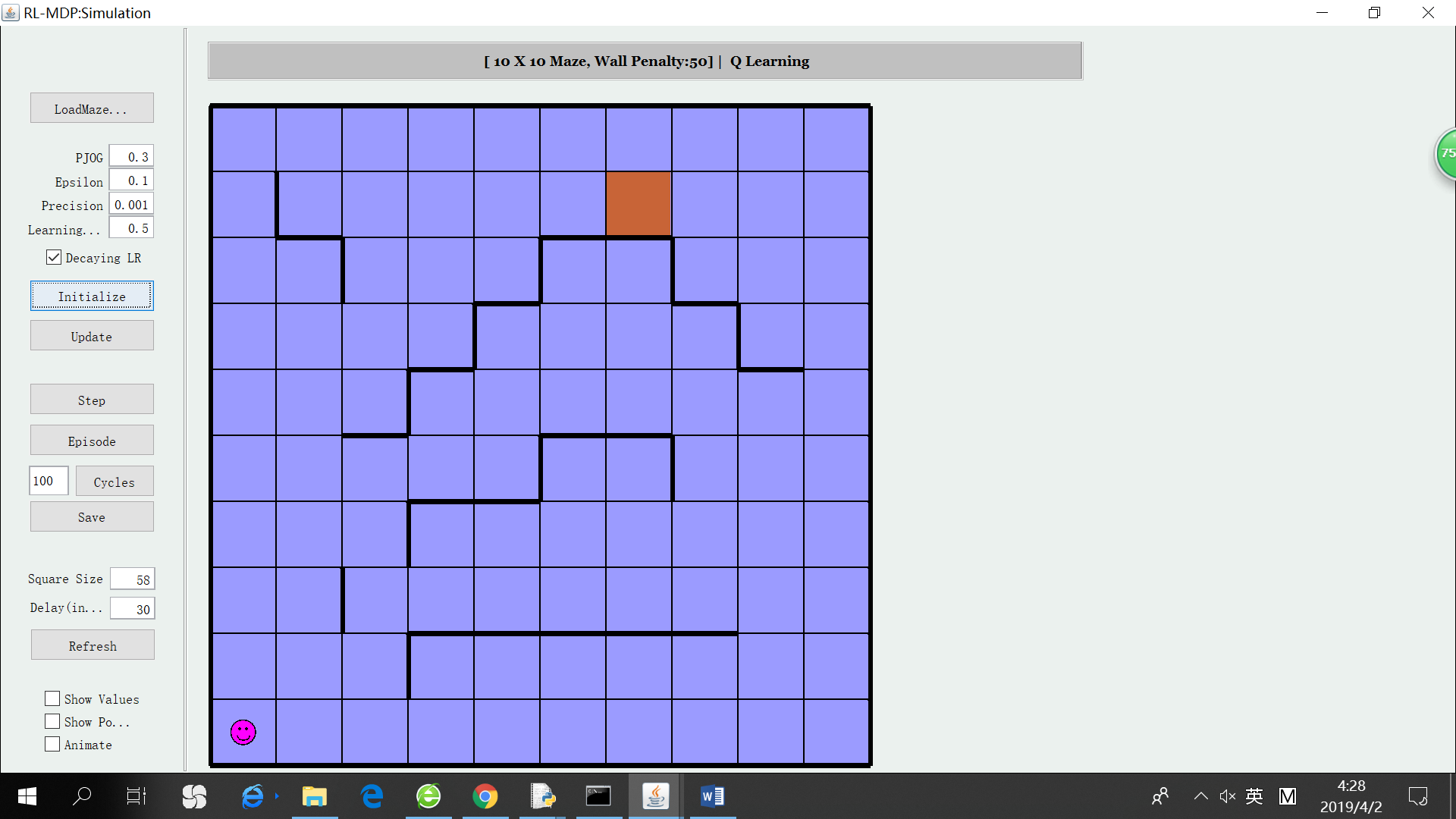
* 1. State Space S: a set of possible world states.
  2. Action Space A: a set of possible actions
  3. A real values reward function R.
  4. Transition Model T: a description of each action’s effects on each state.

1. Grid World Problem

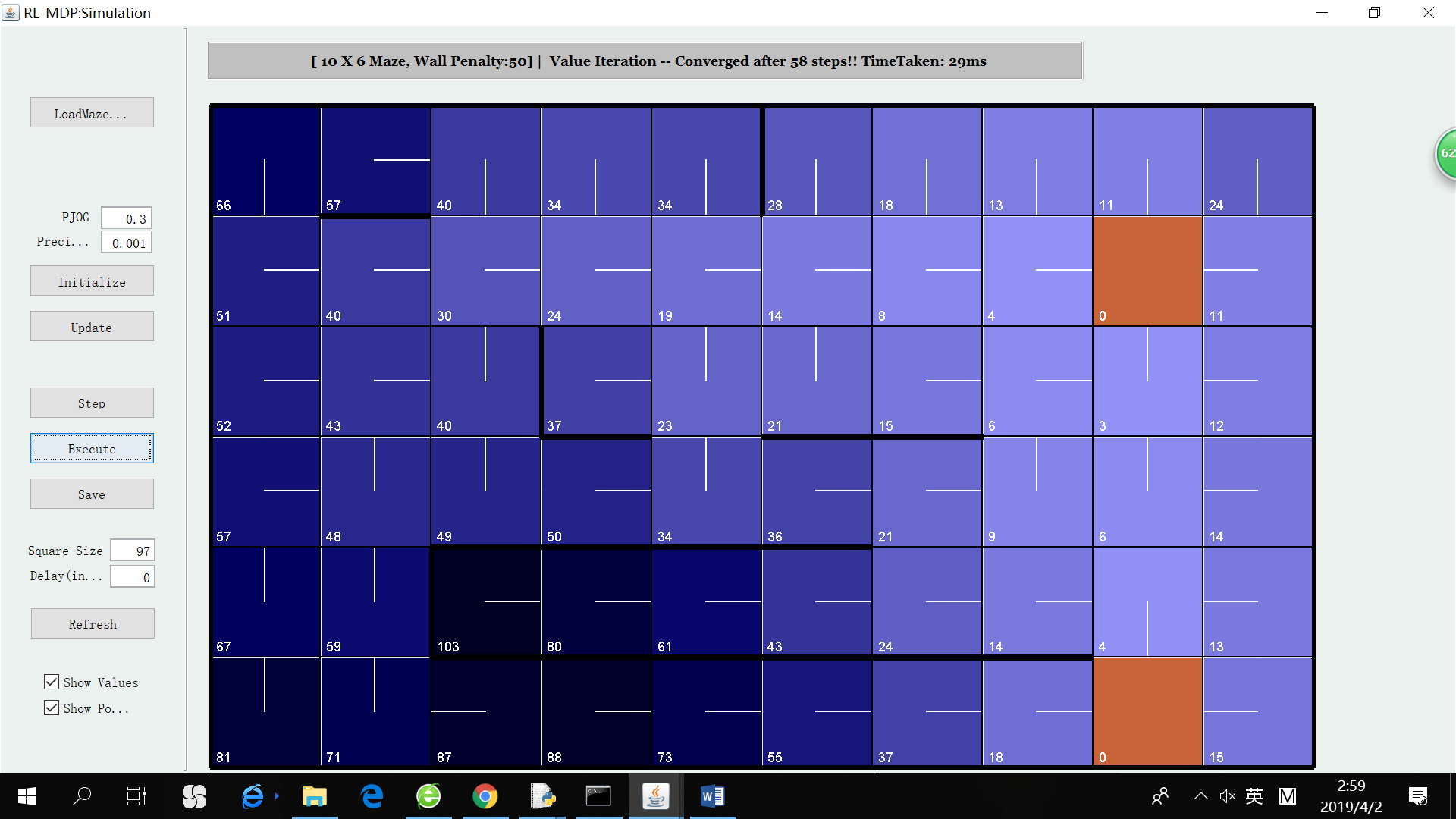
For this project, I have explored the Grid World Problem by multiple approaches. The Grid World problem is a classic problem for MDP because it models the exploration process of practical problems. In the grid world, the agent sets out from a starting point and must traverse a grid-like square to reach a goal. The State Space is all the possible squares that an agent can stay in. In any state, an agent can choose an action from going up, going down, going left and going right. Each action taken causes a constant “cost”, which urges the agent to find the goal as soon as possible instead of moving around in the grid world for unnecessarily long times. There is no reward given for reaching the goal state, nor is there a “cost” for it.

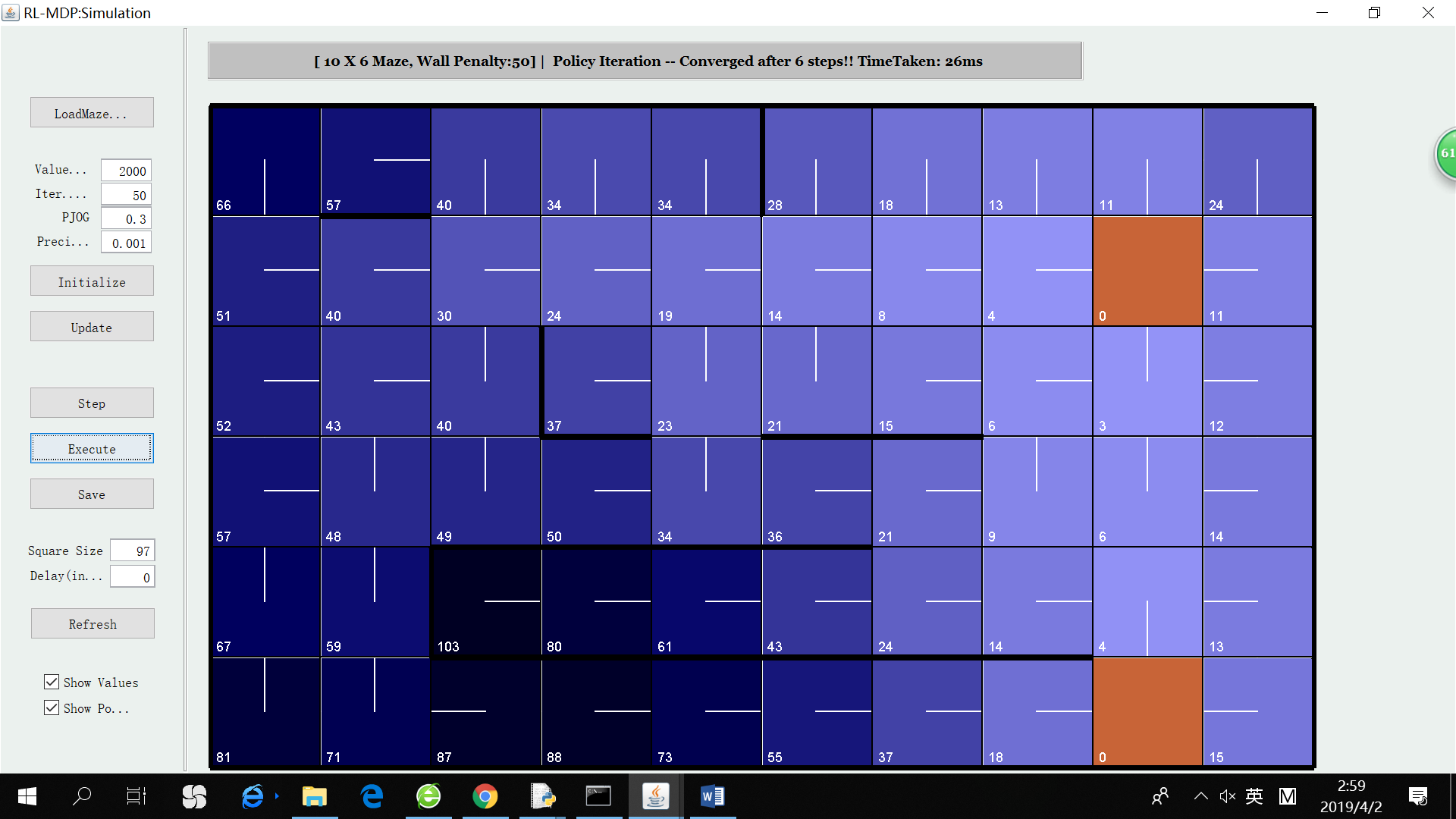
Moreover, there are “walls” on the four sides of the maze, as well as some particular places in the middle of the maze. Any action taken towards a wall will result in the agent staying at its original state. Additionally, “bumping into a wall” will result in a negative penalty of 50.

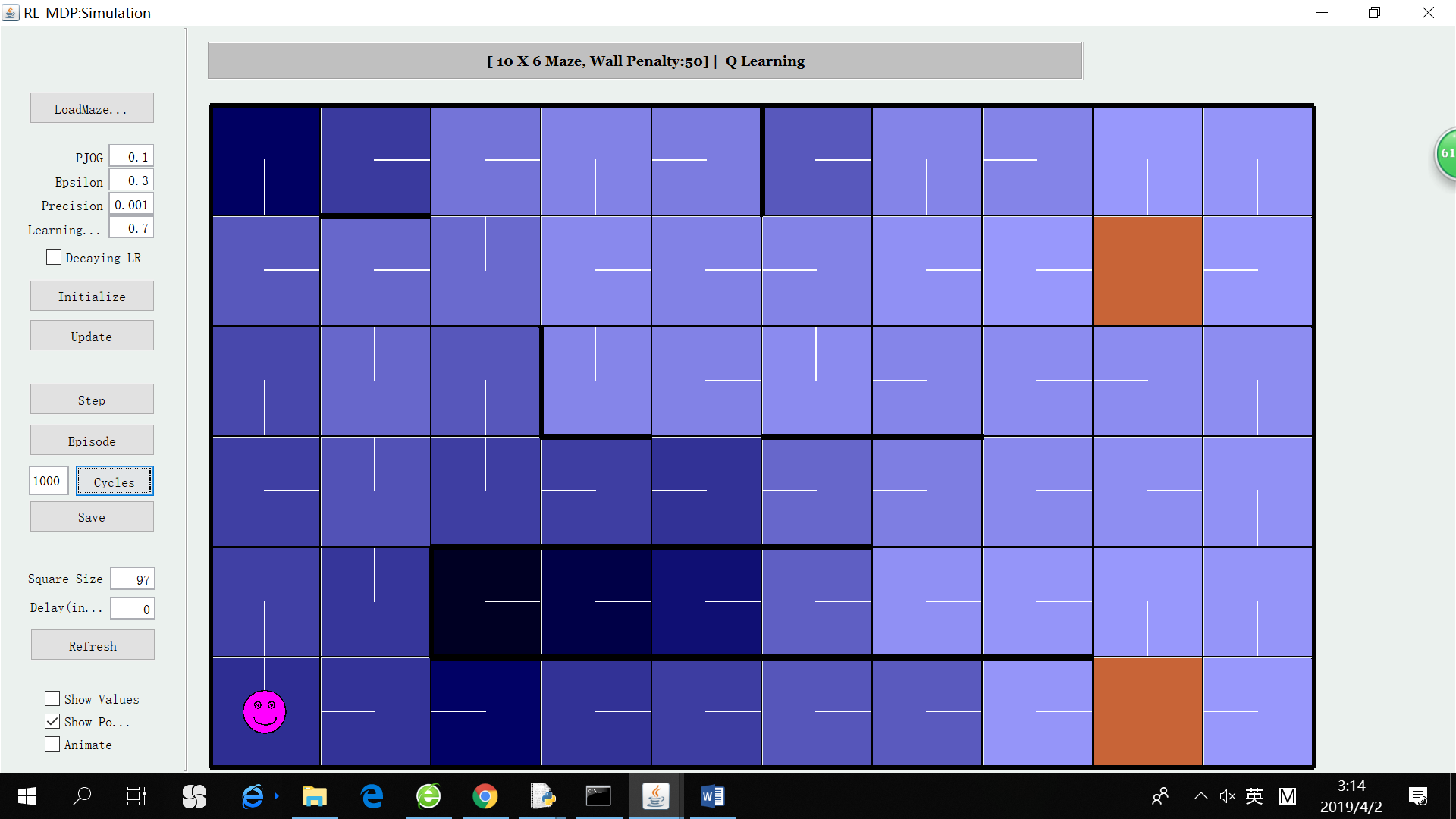
The transition model for this problem is a non-deterministic one. With probability 1-PJOG, (0<PJOG<1), the agent will move towards the target direction, while with probability PJOG/3, the agent will move towards each of the remaining directions. I defined two Gird World Problems using a simple maze with only 9 states and a complex maze with much more states and walls.

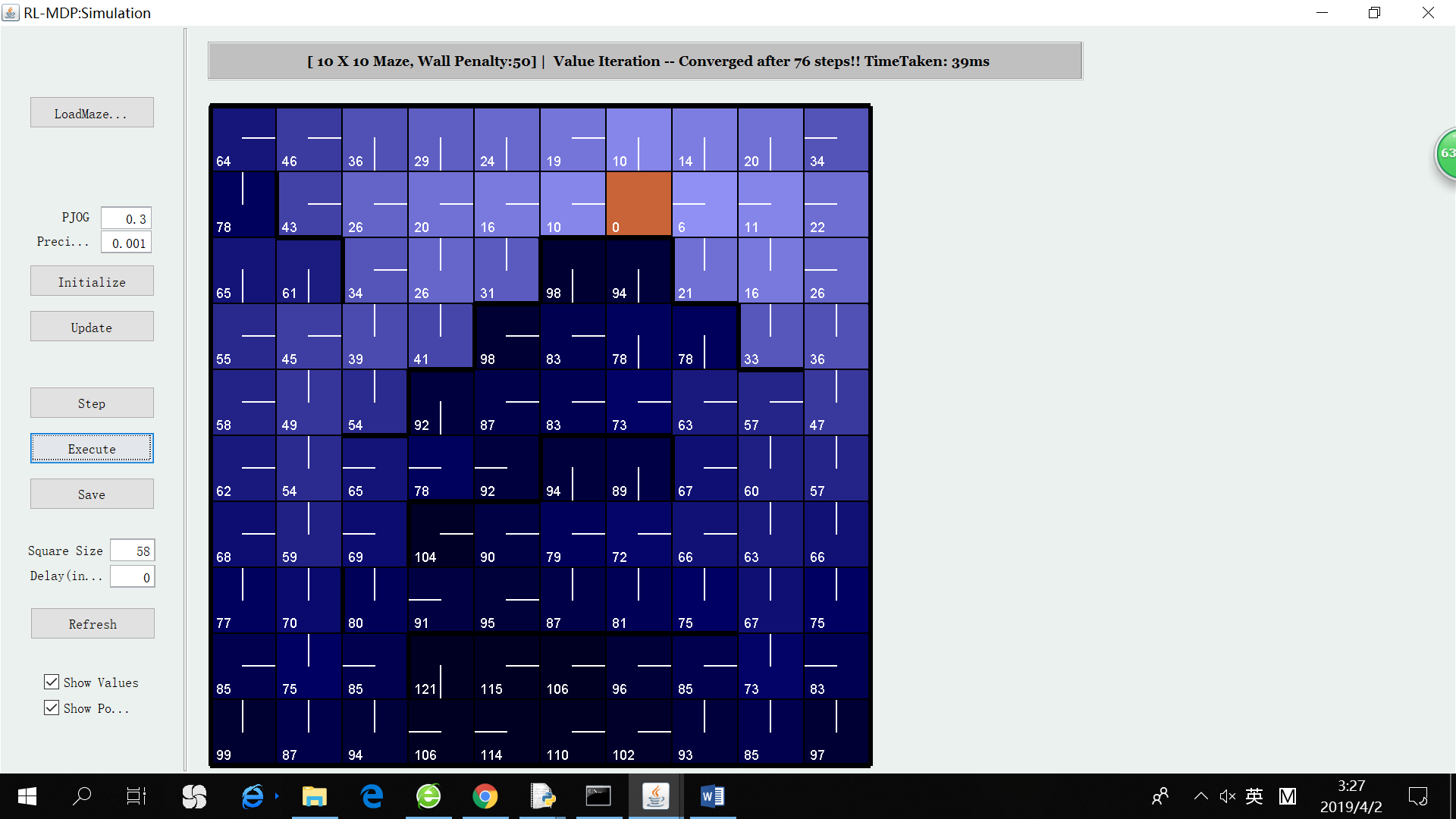
 

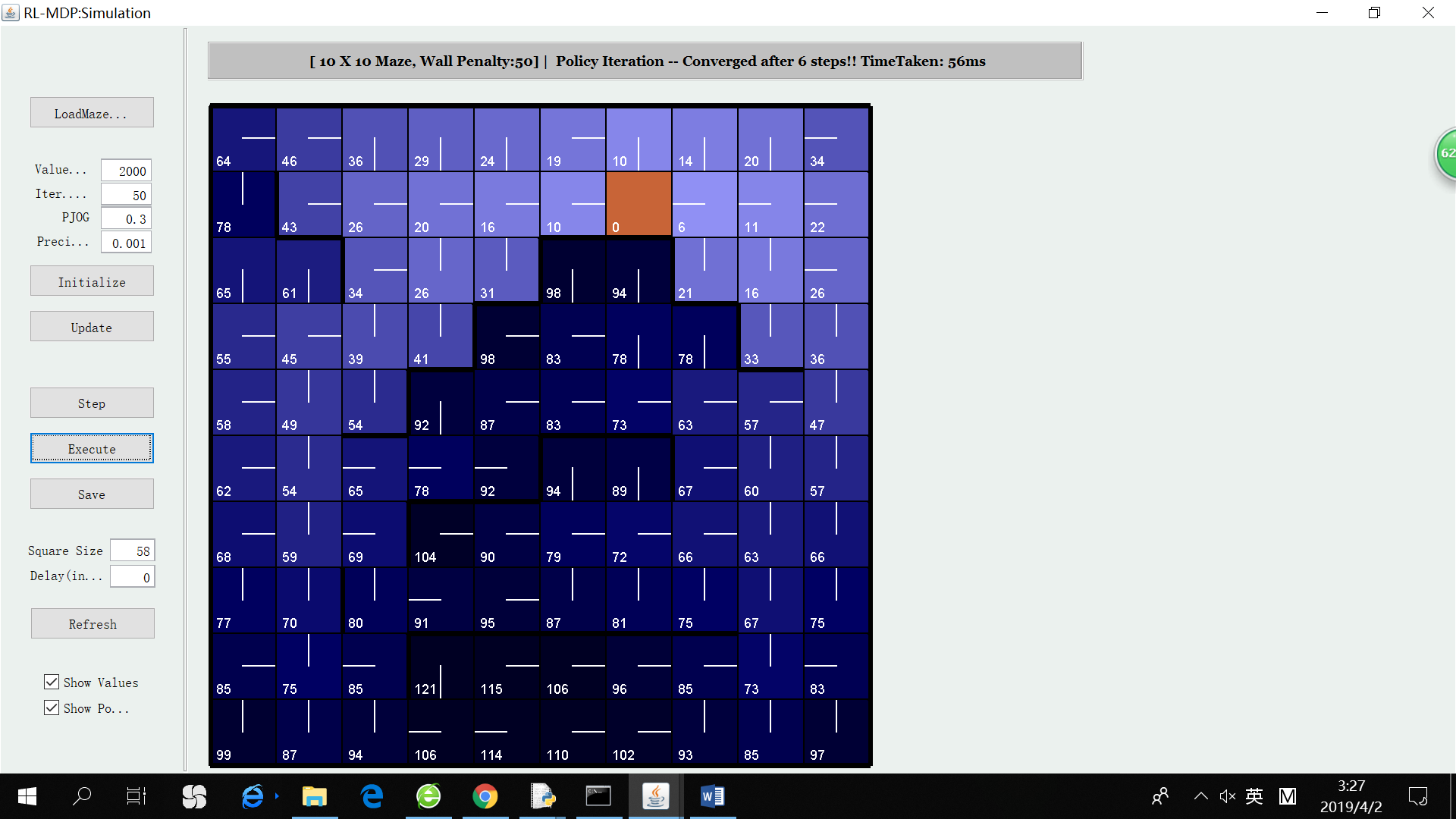
The starting point for both mazes is the bottom-left corner and the goal is highlighted in orange color. The walls are highlighted by bold black lines. The target of this paper is to solve the two Grid World Problems using different reinforcement learning techniques, and compare the algorithms’ performances.

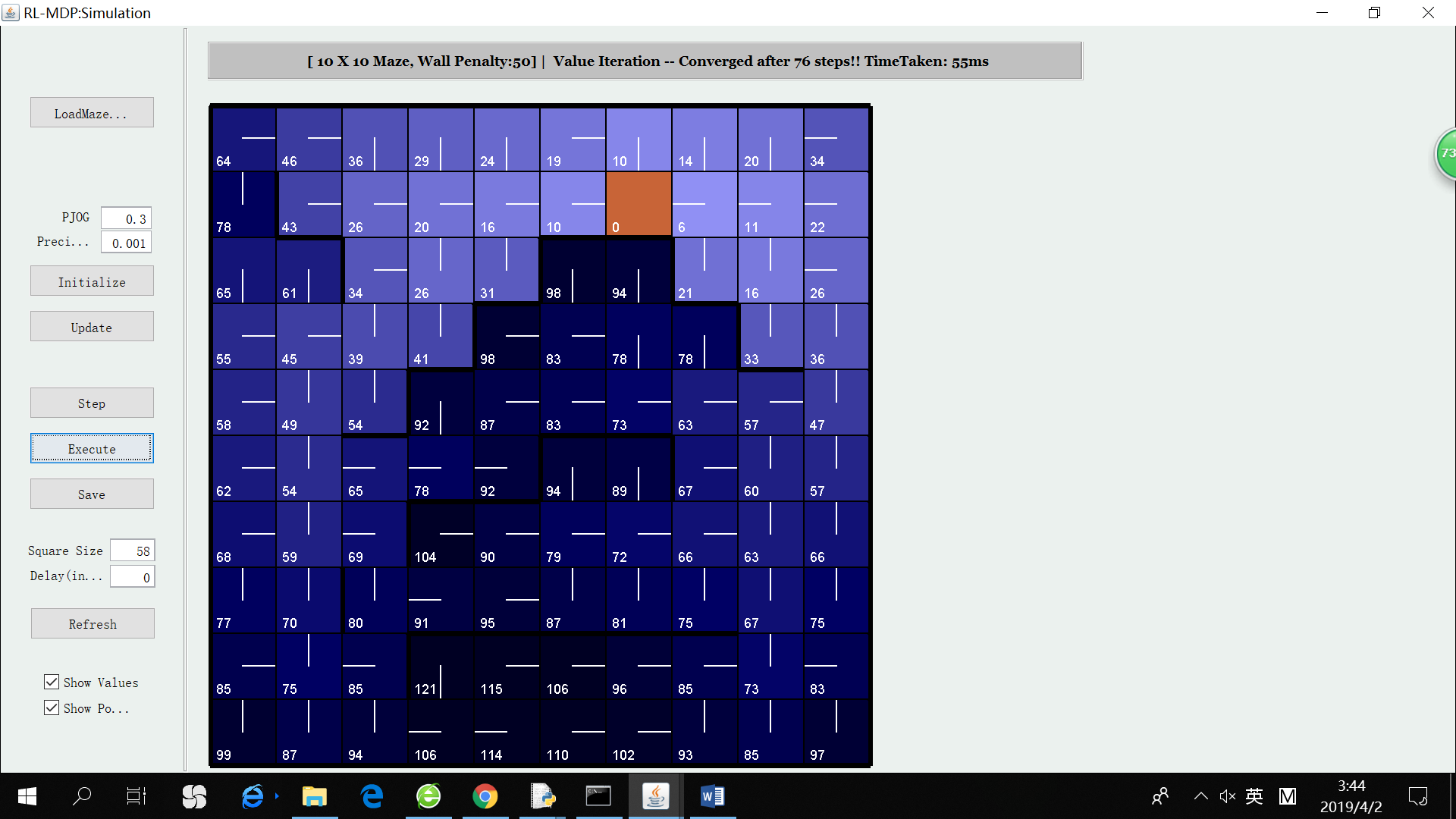


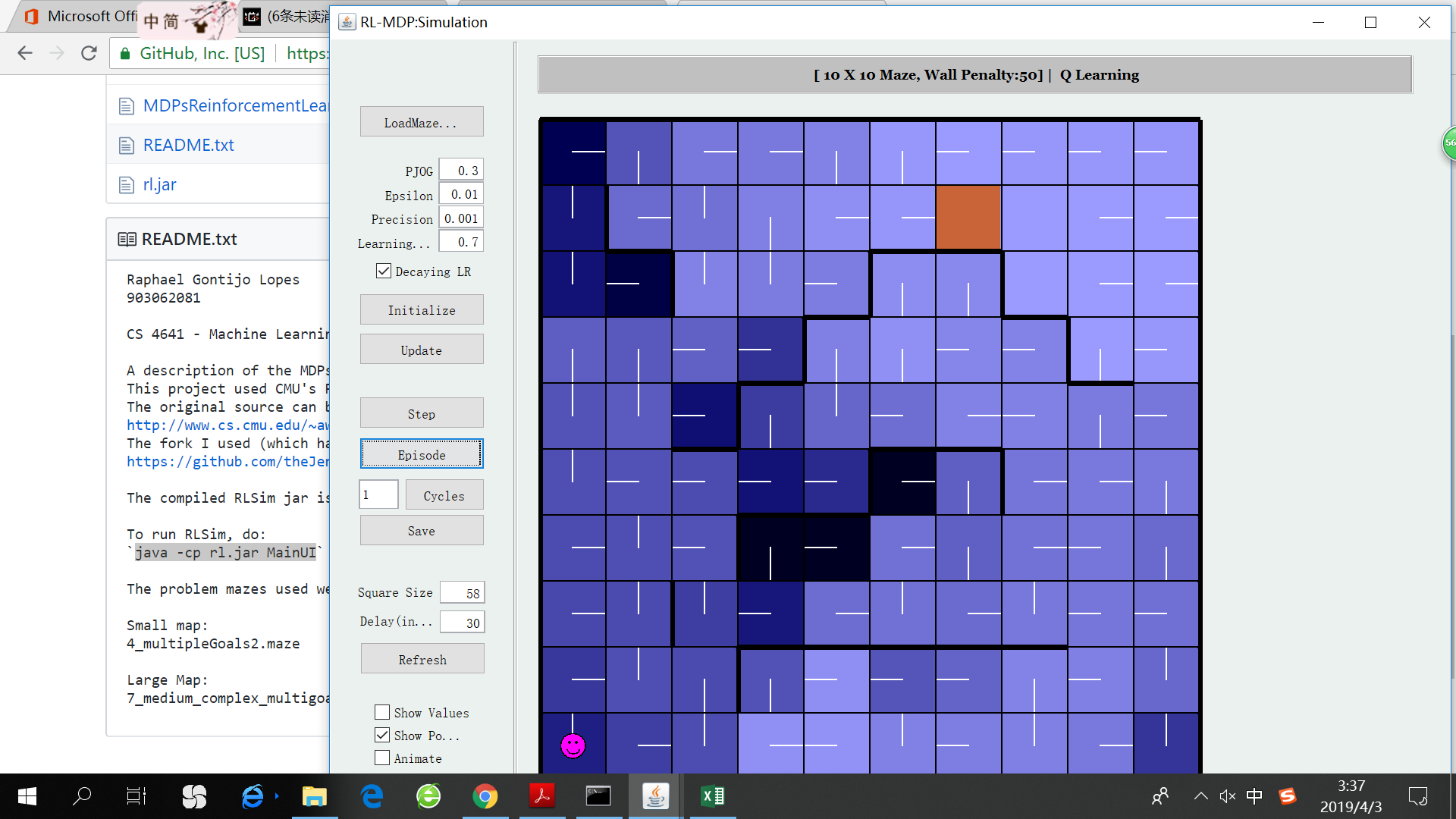




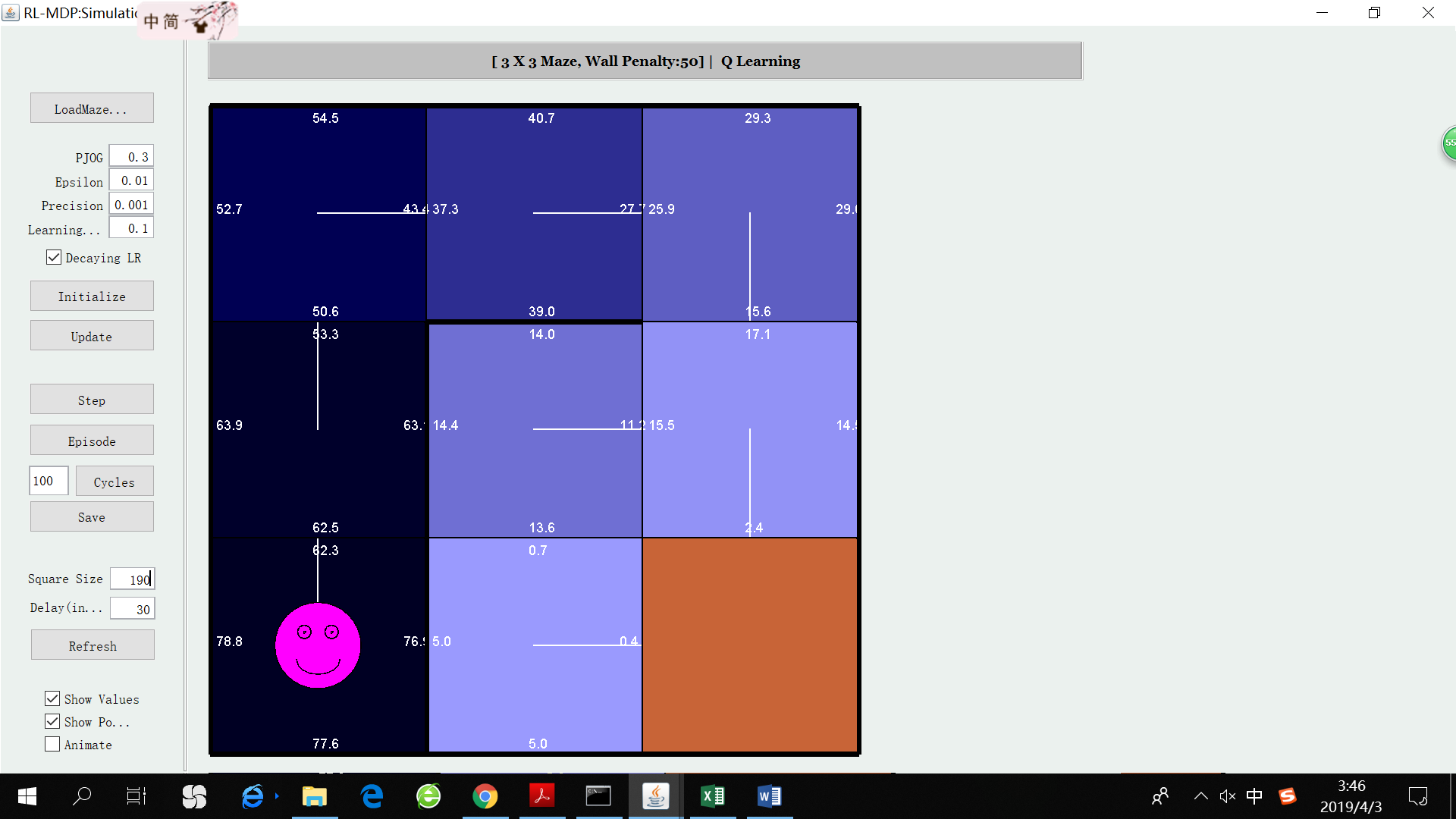




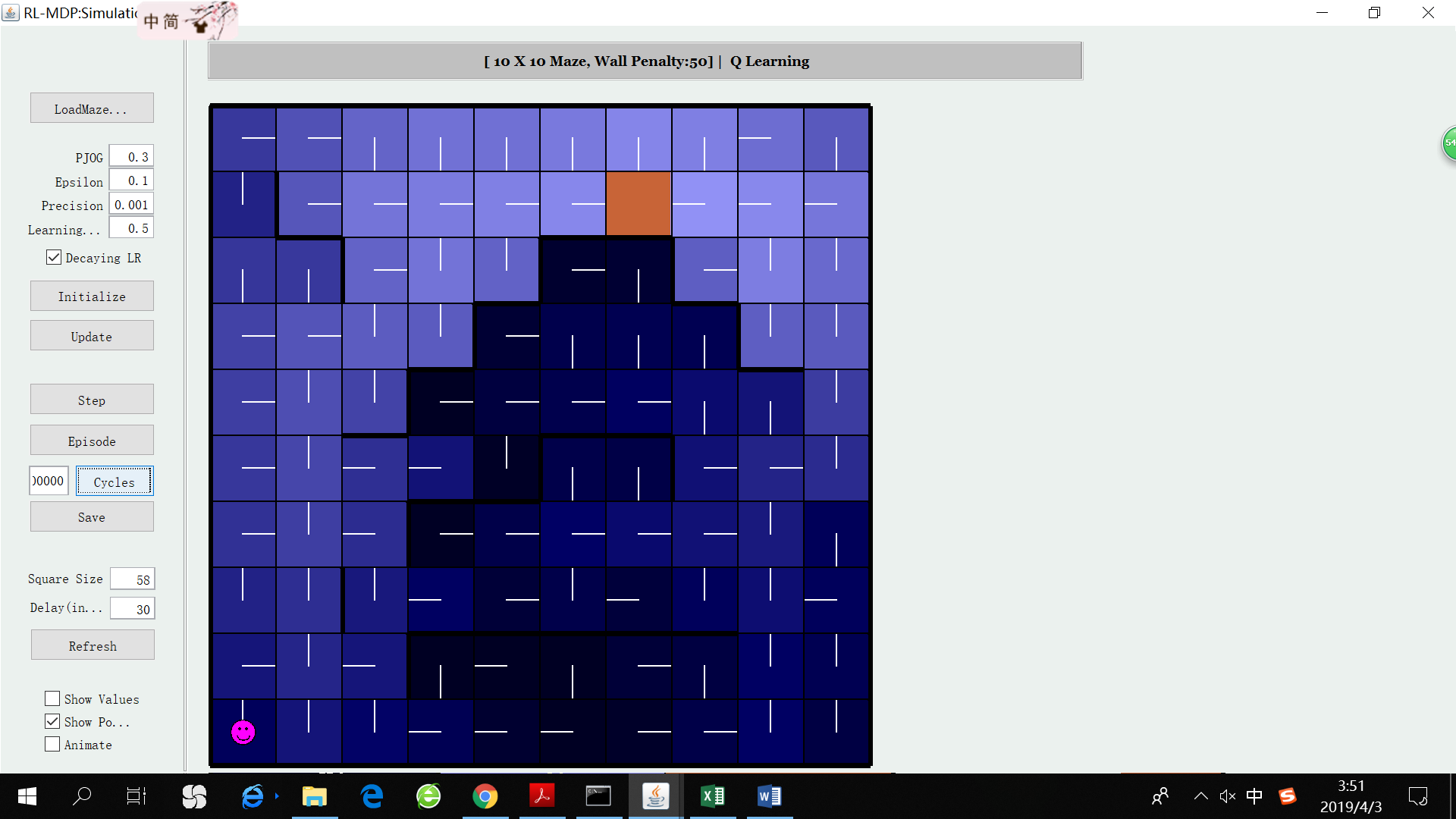




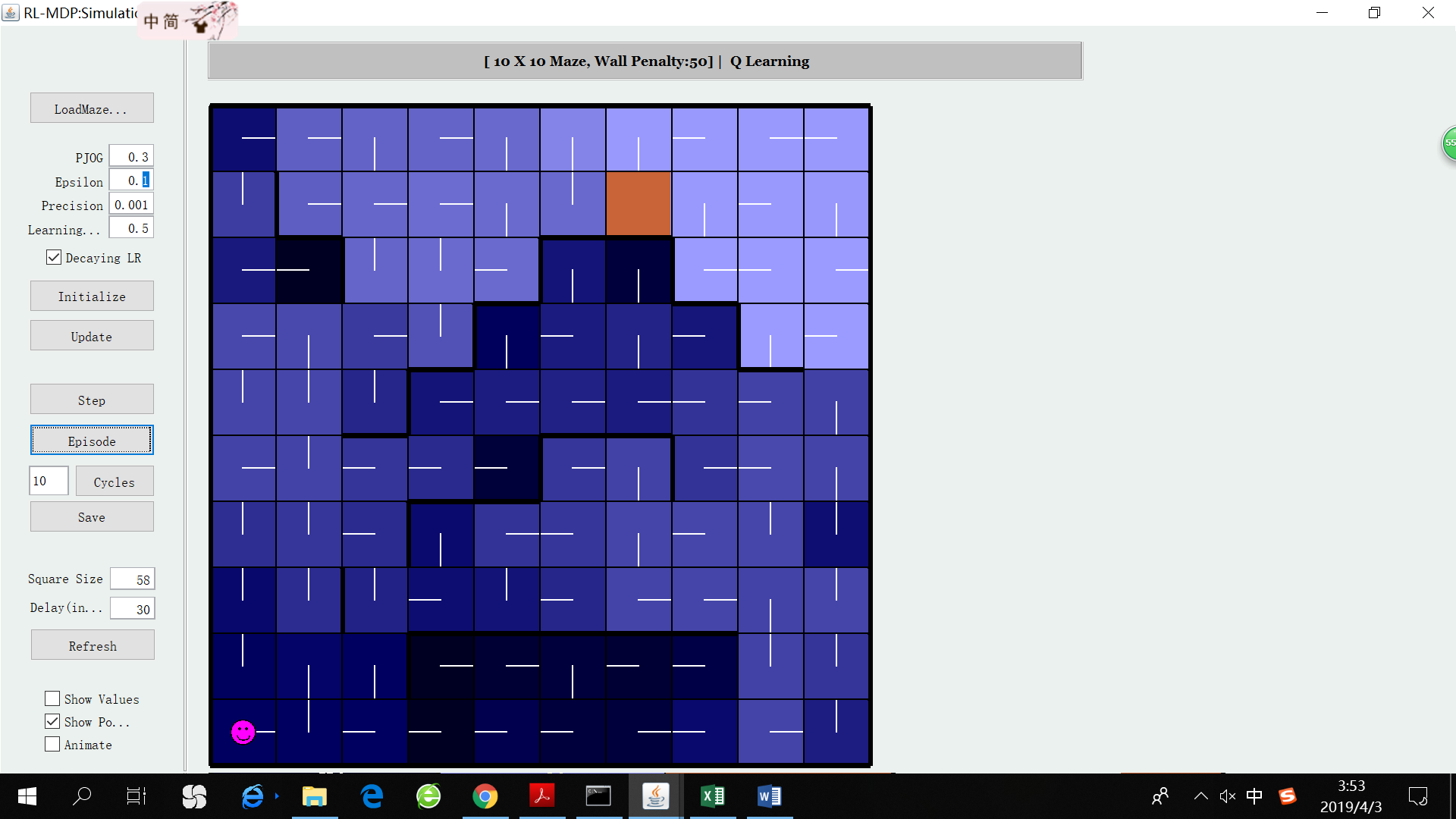
Q learning after 100

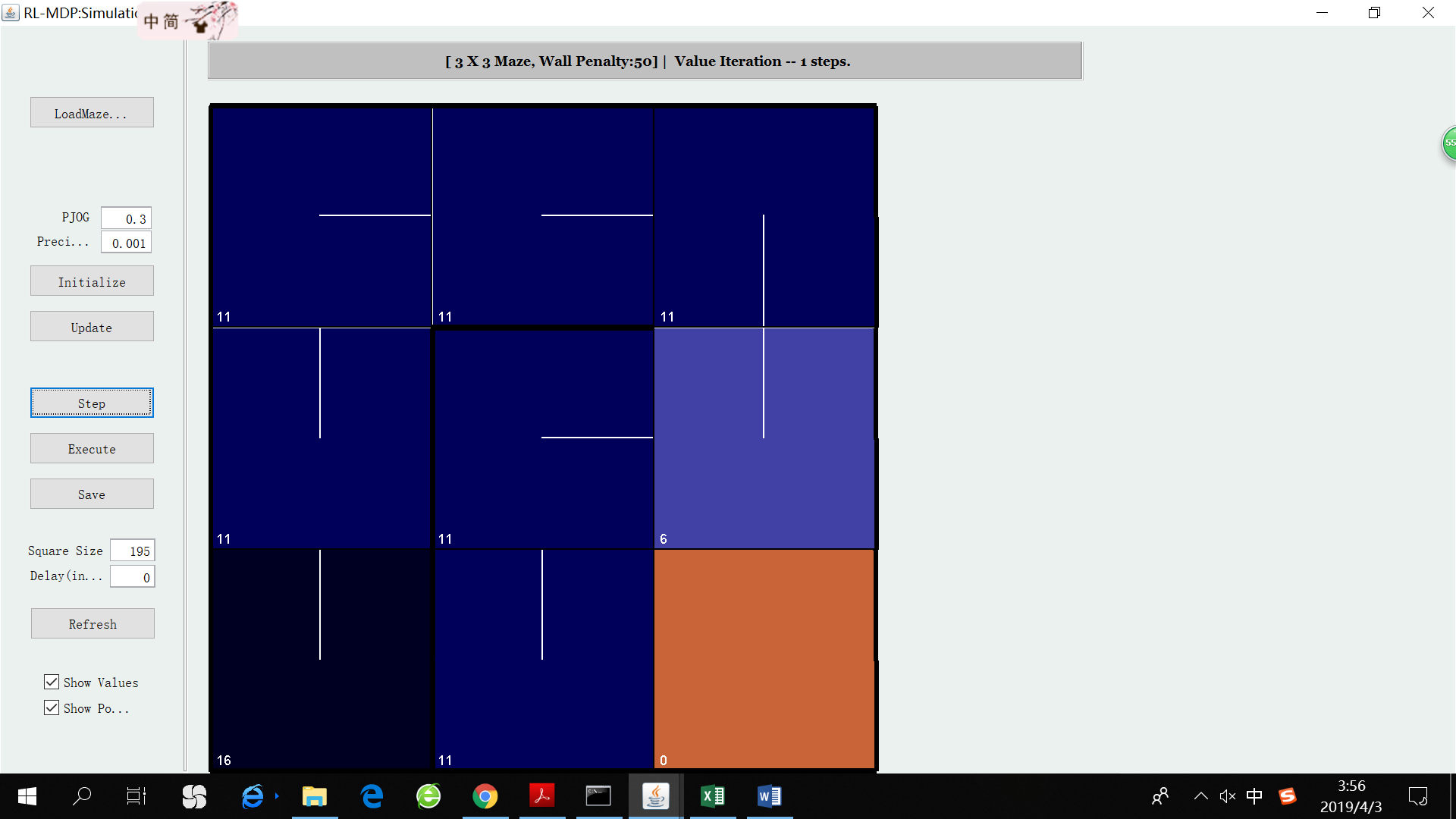


Q learning after 10000

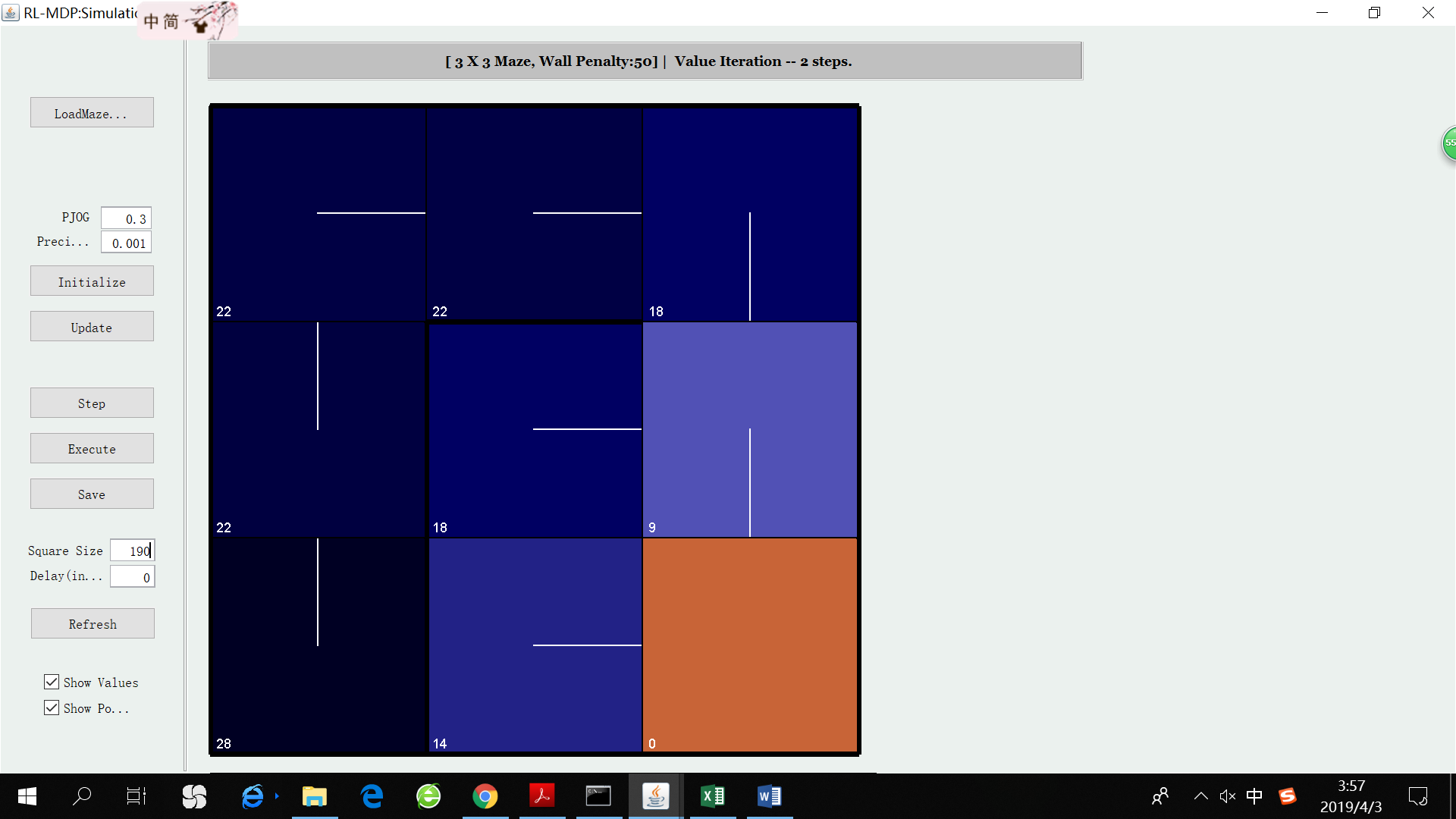


Q learning after 20: already learn useful trick, q-value near wall are larger





VI 1 step



VI 2steps

Maze6: VI’s policy converges within 10 steps.