SWEN90004 Assignment 2
Proposal

Saier Ding()
Lin Fan (945510)
Jinxin Hu ()

School of Engineering
University of Melbourne
Australia
2020

1 Overview

The model of Rebellion is a classic social science problem adapted from Joshua Epstein's civil violence model proposed by 2002. The main idea is analysing the behaviours of a group of subjugated populations against the central authority and generating patterns based on the experiments.

There are two types of people in this model, cops and agents. The trigger for agents to openly rebel depends on the value of grievance and the cost in jail, when the grievance is large enough and the cost of being jailed is small enough, the rebellion happens. Simultaneously, the cops represent the central government who works on quelling the rebellion effectively, they wander around randomly and arrest people who are actively rebelling.

2 Design of the Netlogo Model

In this model, there are two separate threads respectively representing 'cops' and 'agents'. At the set-up stage, they are both assigned to each patch of the map according to the 'initial-density' randomly.

- **Agents**
  - 'Move' action for agents limits 'agents' can only enter an empty patch where the candidate is empty or contains only jailed agents in vision.
  - 'Grievance' is influenced by 'perceived-hardship' (whether 'agents' accept the current government) and 'government-legitimacy'.
  - 'The cost of being jailed' is influenced by 'risk-aversion' (the degree that 'agent' are reluctant to rebel because fearing of being arrested) and 'estimated-arrest-probability' (determined by the number of 'cops' and the number of active 'agents' in vision).
  - 'Active' action means the 'Agents' are rebelling against the government and the prerequisite for rebellion is the difference between 'grievance' and 'the cost of being jailed' less than a minimum threshold.
- **Cops**
  - 'Move' action for cops is almost similar to 'agents'. While when the global variable 'movement' is set to 'off', 'agents' cannot move but 'cops' still can do it.

  - ‘Enforce’ action for cops means when there exists an active agent, ‘cops’ will arrest it and send it into jail with a random jail-term which is less than max-jail-term.

For the move states, the agent and cop are assigned with the random location initially and move in the next state to the other available patch within the visible range. In each move, the field of the patch is limited to one, and only unoccupied patches are valid for moving.

Apart from move state, the cops can also change to arrest state, once the active agents appear in the vision, the arrest state for cops will be triggered and one active agent will be arrested randomly to the jail.

The states of the agent’s behaviours can be divided into three variables:

- **active:** The “active” agents in the group of active rebels which has not been arrested yet.
  - The state will be changed to “jailed” once s/he appears on the cops’ vision.
  - The state will be changed to ‘quiet” if the agents start to calm down and the balance between grievance and cost of being jailed becomes stable.

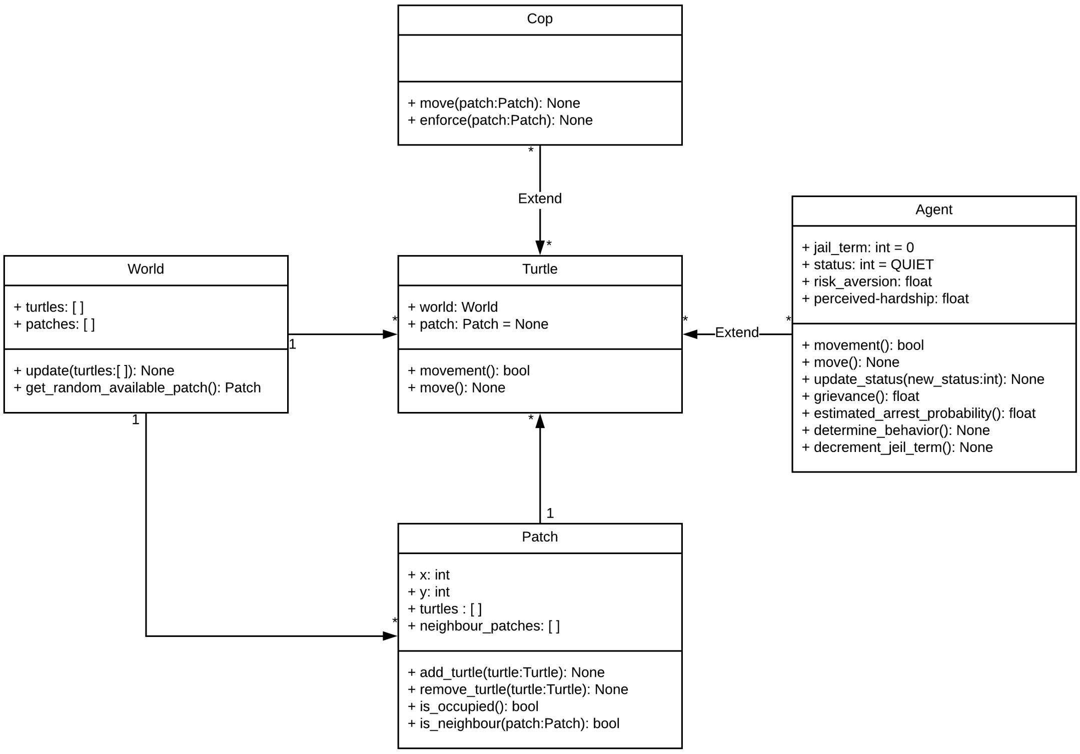
  $$grievance - riskAversion \times estimatedArrestProbability \leq threshold \tag{1}$$

- **quiet:** The “quiet” agents represent the stable people who remain quiet without “active” on the rebellion. The “quiet” state changes to “active” once the balance between grievance and cost of being jailed has been broken and exceeds a certain threshold.

  $$grievance - riskAversion \times estimatedArrestProbability > threshold \tag{2}$$
  $$(grievance = perceivedHardship \times (1 - governmentLegitimacy)) \tag{3}$$

- **jailed:** The “jailed” state grouped the agents who had been already arrested, and the “jailed” state will be finally changed to “quiet” after passing through the maximum jailed period.

3 Design of the Python Model



- **World:** The world-class is the world of the whole map. For each time tick, the world updates the action of turtles.
- **Patch:** Patch class corresponding to the single patch in the map, a turtle can move a patch and signal as an occupied patch by modifying the method in the patch class. Turtles can also detect the neighbour target by accessing through the turtles' list nearby stored in the Patch class.
- **Turtle:** Turtle is a super-class for all movable objects in the map, its subclass includes Agent and Cop.
- **Agent:** Agent is one of the child classes for Turtle, the "jail_term" has been initialised as 0 and the status of the agent begins with "quiet". The agents can only change actions when the "jail_term" is zero.

- **Cop:** The other child class Cop only contains two methods, "move" and "enforce", the cop can randomly arrest one active agent after moving to the surrounding area of agents.

4 Experiments

4.1 Hypothesis 1

If the 'initial-cop-density' is high enough, the number of active agents will remain almost zero.

- **Parameter setting:** 'initial-cop-density' from 0.04 to 0.2
- **Result:** The number of 'agents' in each class. ('quiet', 'jailed', 'active')

4.2 Hypothesis 2

Which of the number of cops and government legitimacy has a greater impact on the rebellion?

- **Parameter setting:** control variable and compare the influence of 'initial-cop-density' and 'government-legitimacy'
- **Result:** The number of 'agents' in each class. ('quiet', 'jailed', 'active')

5 Task Plan

5.1 Timeline

11th May	Proposal Submission.
18th May	Model Implementation.
22th May	Experiments.
24th May	Model Extension.
27th May	Report Writing.
29th May	Report Submission.

5.2 Task Breakdown

Task no.	Task Breakdown	Team Member
1	World Class, Patch Class	Saier Ding
2	Turtle Class, Cop Class	Lin Fan
3	Agent Class	Jinxin Hu
4	Experiments	All members
5	Model Extension	All members
6	Report Writing (Background, Design of Model)	Saier Ding
7	Report Writing (Extension, Result of experiments)	Lin Fan
8	Report Writing (Discussion, Appendix)	Jinxin Hu

References

[1] Wilensky, U. (2004). NetLogo Rebellion model. http://ccl.northwestern.edu/netlogo/models/Rebellion. Center for Connected Learning and Computer-Based Modeling, Northwestern University, Evanston, IL.

[2] Wilensky, U. (1999). NetLogo. http://ccl.northwestern.edu/netlogo/. Center for Connected Learning and Computer-Based Modeling, Northwestern University, Evanston, IL.