REBELLION MODEL DESCRIPTION

GRACE I. LIN - GSOC 2011

Contents

1. Model Description

The rebellion model (Figure 1) simulates how a group of people rebels against the authority (cops). Each person is in one of the three states:

- **active**: red color; person decides to rebel (see calculation below for becoming an active person)
- **jailed**: black color; active person arrested by cops; remain in jail until jail term expires
- **quiet**: green color; neither active nor jailed

The cops (blue color) will randomly arrest active people within the visibility area. To become an active (rebel), the person calculates his/her level of grievance and arrest probability:

$$grievance = perceived\_hardship \times (1 - government\_legitimacy)$$
$$arrest\_probability = 1 - \exp\left(- k \left\lfloor \frac{num\_of\_cops}{num\_of\_active\_people} \right\rfloor \right)$$
$$rebel\ if\ (grievance - risk\_aversion \times arrest\_probability) > threshold$$

where

$perceived\_hardship$ = 1.0 (fixed for lifetime)
$government\_legitimacy$ = between 0 and 1; user specified
k = constant for one cop and one person within vision = 2.3
$risk\_aversion$ = 1.0 (fixed for lifetime)
$threshold$ = 0.1

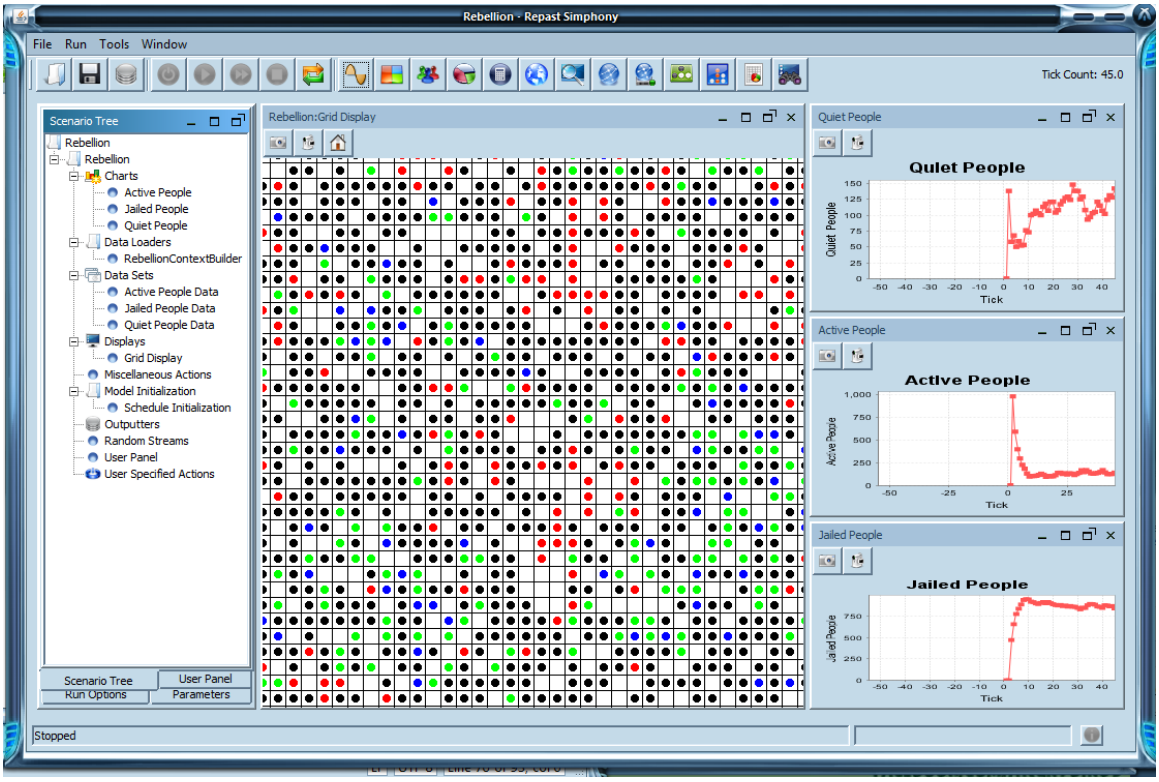

FIGURE 1. Rebellion Model

2. Files and GUI

Source files and description:

src/rebellion/**Constants.java** : list of constants
src/rebellion/**Cop.java** : class for cop
src/rebellion/**CoverageCounter.java** : counts number of people in different states; used for data sets and charts in GUI Scenario Tree
src/rebellion/**Person.java** : class for person
src/rebellion/**RebellionContextBuilder.java** : set up environment initially
src/rebellion/**SMUtils.java** : miscellaneous functions
src/rebellion.observer/**PersonStyleOLG2D.java** : color assignment for cop/person

Remember to setup the grid and space projections in the resource folder "Rebellion.rs": in context node, add projection for type=continuous space and id=space, and add projection for type=grid and id=grid. The id's can be found in the file Constants.java. Once we

run the Rebellion model, set up the Scenario Tree as follow:

- **Data Loaders** : Use RebellionContextBuilder
- **Display**: Projection with grid, agents are Person and Cop, and agent style is rebellion.observer.PersonStyleOLG2D for both Person and Cop
- **Data Sets**: One for each active, quiet, and jailed people count from CoverageCounter. Include a Title column for each data set so the chart will treat the counts in each data set as one series. For example for active people, the Agent Class is CoverageCounter, agent mappings have column name Tick, ActivePeopleCount (source is getActivePeopleCount()), and Title (use Add → Add Using Wizards, and set value="Number of Active People")
- **Charts**: One for each active, quiet, and jailed people data sets. For example for active people count, X-Axis is Tick, Y-Axis Value is ActivePeopleCount with Series Name as Title.

3. Model Parameters

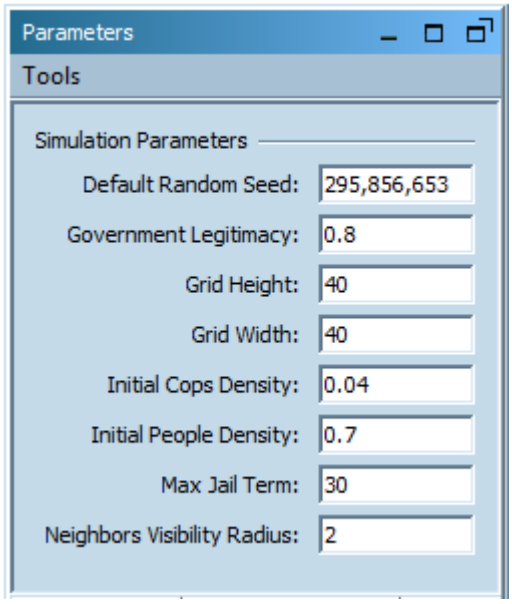

Figure 2. Rebellion Model - Parameters

User-specified parameters are shown in Figure 2 with descriptions as follow:

- **Government Legitimacy**: Number of rows.
- **Grid Height**: Number of columns.
- **Grid Width**: Number of rows.

- **Initial Cops Density**: Percentage of grid size populated by cops.
- **Initial People Density**: Percentage of grid size populated by people.
- **Max Jail Term**: Maximum turns for people who are jailed.
- **Neighbors Visibility Radius**: Radius of area that the agent can see.

4. References

- NetLogo's termites model: *http://ccl.northwestern.edu/netlogo/models/Rebellion*
- Modeling civil violence: An agent-based computational approach (*http://www.pnas.org/content/99*