

## NETLOGO CODE

:: SOCIAL MEDIA IN UK RIOTS

:: Original model by Uri Wilensky, modified by Antonio A. Casilli and Paola Tubaro on 10 August 2011.

:: Authorised by copyright notice as reported below.

:: Original model available in the Netlogo model library, downloadable at: <http://ccl.northwestern.edu/netlogo/>

```
breed [agents agent]
```

```
breed [cops cop]
```

```
globals [
```

```
  k ; factor for determining arrest probability
```

```
  threshold ; by how much must G > N to make someone rebel?
```

```
]
```

```
agents-own [
```

```
  risk-aversion ; R, fixed for the agent's lifetime, ranging from 0-1 (inclusive)
```

```
  perceived-hardship ; H, also ranging from 0-1 (inclusive)
```

```
  active? ; if true, then the agent is actively rebelling
```

```
  jail-term ; how many turns in jail remain? (if 0, the agent is not in jail)
```

```
]
```

```
patches-own [
```

```
  neighborhood ; surrounding patches within the vision radius
```

```
]
```

```
to setup
```

```
  clear-all
```

```
  :: set globals
```

```
  set k 2.3
```

```
  set threshold 0.1
```

```
  ask patches [
```

```
    ;; make background a slightly dark gray
```

```
    set pcolor gray - 1
```

```
    ;; cache patch neighborhoods
```

```
    set neighborhood patches in-radius vision
```

```
  ]
```

```
  :: create cops
```

```
  create-cops round (initial-cop-density * .01 * count patches) [
```

```
    move-to one-of patches with [not any? turtles-here]
```

```
    display-cop
```

```
  ]
```

```
  :: create agents
```

```
  create-agents round (initial-agent-density * .01 * count patches) [
```

```
    move-to one-of patches with [not any? turtles-here]
```

```
    set heading 0
```

```
    set risk-aversion random-float 1.0
```

```
    set perceived-hardship random-float 1.0
```

```
    set active? false
```

```
    set jail-term 0
```

```
    display-agent
```

```
  ]
```

```
;; plot initial state of system
update-plots
end
```

```
to go
ask turtles [
; Rule M: Move to a random site within your vision
if (breed = agents and jail-term = 0) or breed = cops
[ move ]
; Rule A: Determine if each agent should be active or quiet
if breed = agents and jail-term = 0 [ determine-behavior ]
; Rule C: Cops arrest a random active agent within their radius
if breed = cops [ enforce ]
]
; Jailed agents get their term reduced at the end of each clock tick
ask agents
[ if jail-term > 0 [ set jail-term jail-term - 1 ] ]
; update agent display
ask agents [ display-agent ]
ask cops [ display-cop ]
; advance clock and update plots
tick
update-plots
if ticks = 1000 [stop]
end
```

```
;; AGENT AND COP BEHAVIOR
```

```
;; This was the original model by U. Wilensky (now inactive)
;; move to an empty patch
;;to move ;; turtle procedure
;;if movement? or (breed = cops) [
;; move to a patch in vision; candidate patches are
;; empty or contain only jailed agents
;; let targets neighborhood with
;; [not any? cops-here and all? agents-here [jail-term > 0]]
;; if any? targets [ move-to one-of targets ]
;; ]
;; end
```

```
;; These are the modifications introduced by A. Casilli and P. Tubaro on 10 August 2011
```

```
to move ;; turtle procedure
;; for cops only: it is just the same as before
ifelse breed = cops [
;; move to a patch in vision; candidate patches are
;; empty or contain only jailed agents
let targets neighborhood with
[not any? cops-here and all? agents-here [jail-term > 0]]
if any? targets [ move-to one-of targets ]
]
;; for agents: this has been changed
;; this introduces an asymmetry that provides an additional advantage for agents
[if movement? [
;; move to a patch in vision
;; candidate patches are empty or contain only jailed agents and
;; they choose among them those with highest number of active agents around
let targets neighborhood with
[not any? cops-here and all? agents-here [jail-term > 0]]
if any? targets [
move-to max-one-of targets [count (agents-on neighborhood) with [active?]]
]
]
]
]
end
```

```
]]]
```

```
end
```

```
:: AGENT BEHAVIOR
```

```
to determine-behavior
```

```
  set active? (grievance - risk-aversion * estimated-arrest-probability > threshold)  
end
```

```
to-report grievance
```

```
  report perceived-hardship * (1 - government-legitimacy)  
end
```

```
to-report estimated-arrest-probability
```

```
  let C count cops-on neighborhood  
  let A 1 + count (agents-on neighborhood) with [active?]  
  ;; See Information tab for a discussion of the following formula  
  report 1 - exp (- k * floor (C / A))  
end
```

```
:: COP BEHAVIOR
```

```
to enforce
```

```
  if any? (agents-on neighborhood) with [active?] [  
    ;; arrest suspect  
    let suspect one-of (agents-on neighborhood) with [active?]  
    ask suspect [  
      set active? false  
      set jail-term random max-jail-term  
    ]  
    move-to suspect ;; move to patch of the jailed agent  
  ]  
end
```

```
:: VISUALIZATION OF AGENTS AND COPS
```

```
to display-agent ;; agent procedure
```

```
  ifelse visualization = "2D"  
    [ display-agent-2D ]  
    [ display-agent-3D ]  
end
```

```
to display-agent-2D ;; agent procedure
```

```
  set shape "circle"  
  ifelse active?  
    [ set color red ]  
    [ ifelse jail-term > 0  
      [ set color black + 3 ]  
      [ set color scale-color green grievance 1.5 -0.5 ] ]  
end
```

```
to display-agent-3D ;; agent procedure
```

```
  set color scale-color green grievance 1.5 -0.5  
  ifelse active?  
    [ set shape "person active" ]  
    [ ifelse jail-term > 0
```

```
    [ set shape "person jailed" ]
    [ set shape "person quiet" ] ]
end
```

```
to display-cop
  set color cyan
  ifelse visualization = "2D"
    [ set shape "triangle" ]
    [ set shape "person soldier" ]
end
```

```
:: PLOTTING
```

```
to update-plots
  let active-count count agents with [active?]
  let jailed-count count agents with [jail-term > 0]
```

```
  set-current-plot "Active agents"
  plot active-count
```

```
  set-current-plot "All agent types"
  set-current-plot-pen "active"
  plot active-count
  set-current-plot-pen "jailed"
  plot jailed-count
  set-current-plot-pen "quiet"
  plot count agents - active-count - jailed-count
end
```

```
:: Original model by Uri Wilensky, modified by Antonio Casilli and Paola Tubaro on 10 August 2011.
:: Authorised by copyright notice as reported below.
```

```
:: Copyright 2004 Uri Wilensky. All rights reserved.
:: Permission to use, modify or redistribute this model is hereby granted, provided that both of the following
requirements are followed:
:: a) this copyright notice is included.
:: b) this model will not be redistributed for profit without permission from Uri Wilensky.
:: Contact Uri Wilensky for appropriate licenses for redistribution for profit.
:: This model was created as part of the projects:
:: PARTICIPATORY SIMULATIONS: NETWORK-BASED DESIGN FOR SYSTEMS LEARNING IN
CLASSROOMS and/or INTEGRATED SIMULATION AND MODELING ENVIRONMENT.
:: The project gratefully acknowledges the support of the National Science Foundation (REPP & ROLE programs)
-- grant numbers REC #9814682 and REC-0126227.
```