JÜLICH
FORSCHUNGSZENTRUM

Introduction to NetLogo

Multi-agent programmable modeling environment

Simulation of road traffic flow

29.04.2015 | Antoine Tordeux[3] and Mohcine Chraibi[4] | Forschungszentrum Jülich

Wilhelm Dörpfeld Gymnasium

[3] a.tordeux@fz-juelich.de
[4] m.chraibi@fz-juelich.de

NetLogo simulation module

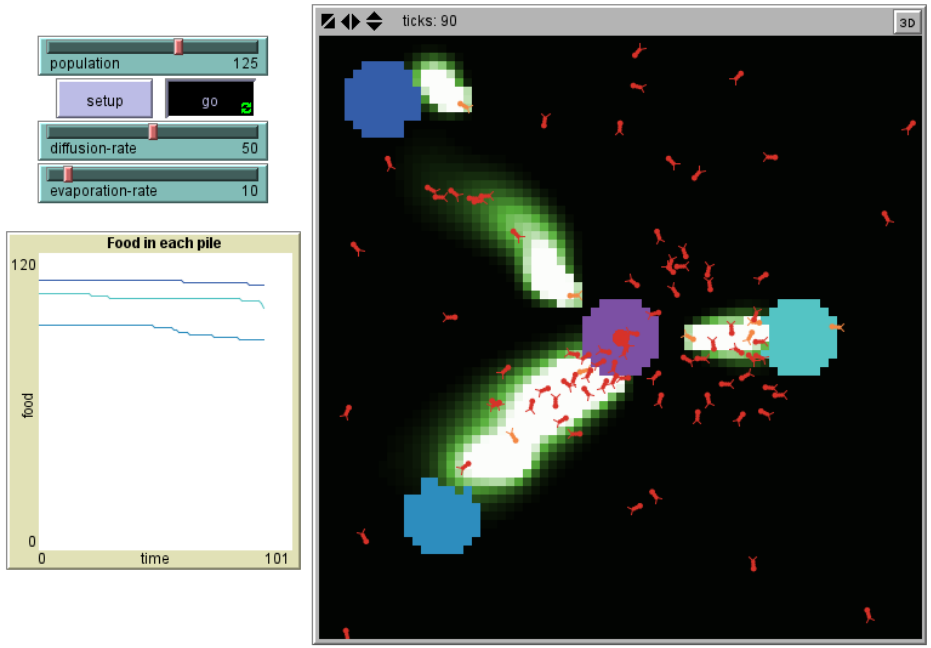
- **NetLogo : Agent-based integrated modeling environment**
  - Authored by **Uri Wilensky** and developed at CCL[5]
  - **Free and open source software**[6]
  - Logo **functional and agent-based programming language**
  - **Assisted** programming and interface graphic
- **Modelling of agents evolving in a plan**
  - **Complex collective systems** with many **agents in interaction**
  - Connection between the **micro-level behavior of individuals** and **macro-level patterns** emerging from their interaction

[5] Northwestern Institute on Complex Systems, Northwestern University

[6] `http://ccl.northwestern.edu/netlogo/download`

Example of NetLogo model[7]

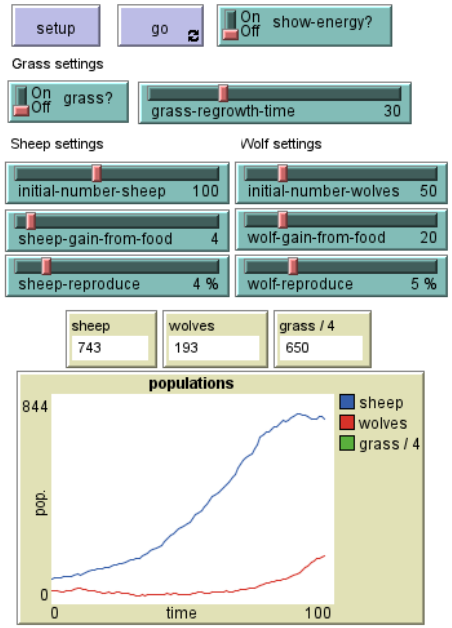
Ants

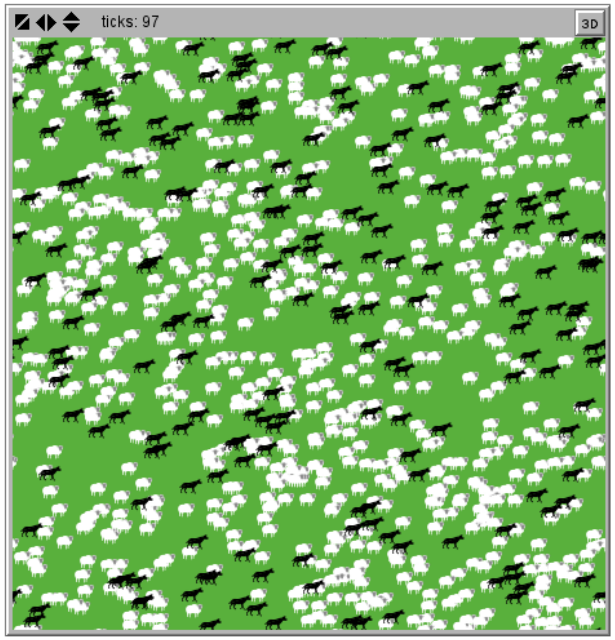


[7] see NetLogo Models Library

Example of NetLogo model[8]

Wolf Sheep Predation

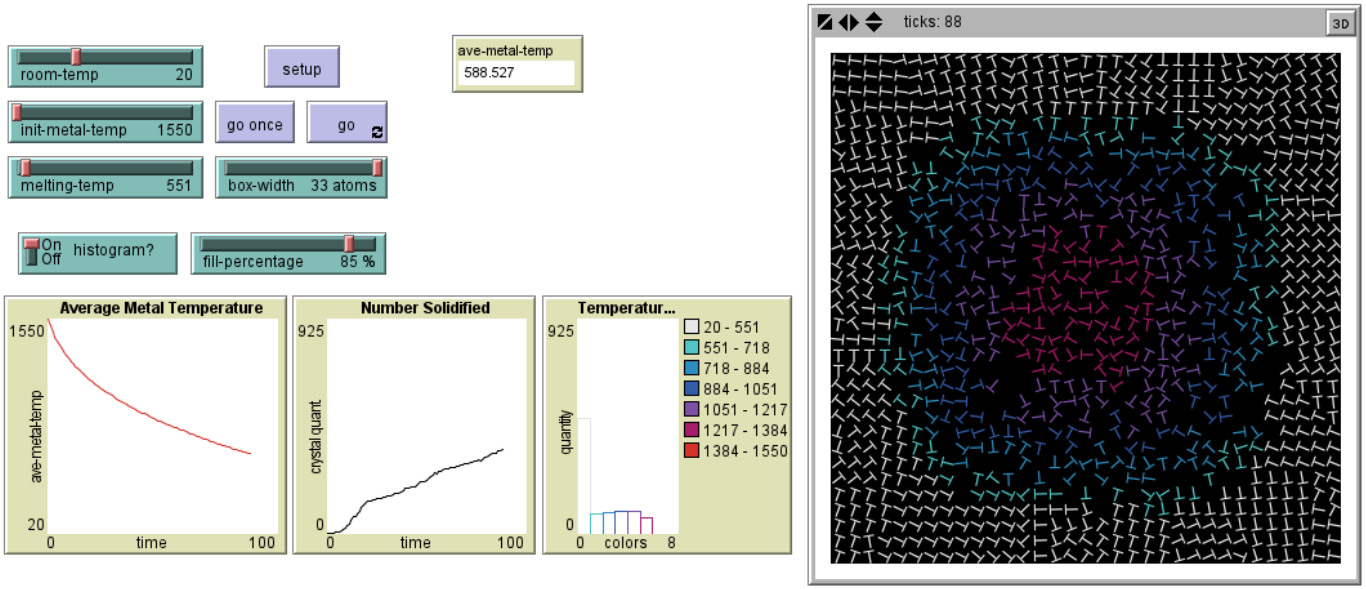




[8] see NetLogo Models Library

Example of NetLogo model[9]

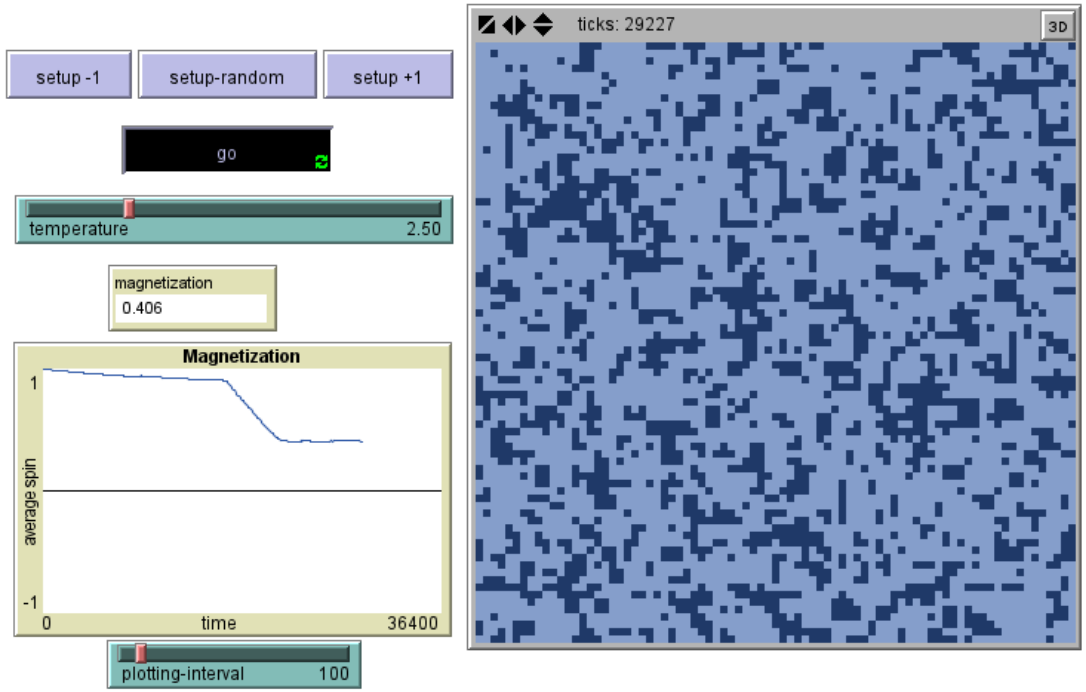
Crystallisation



[9] see NetLogo Models Library

Example of NetLogo model[10]

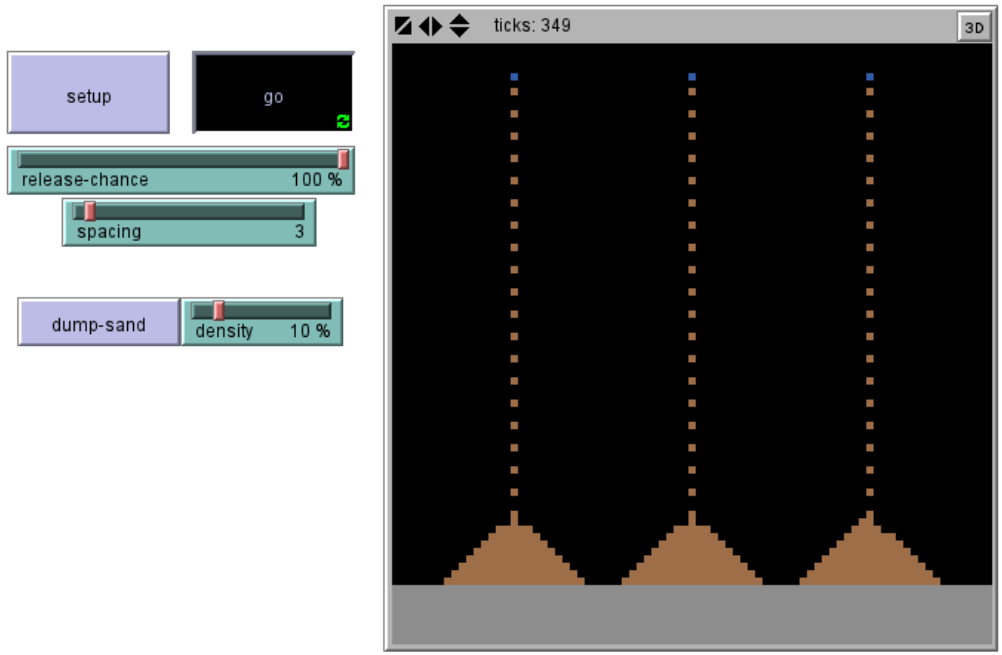
Ising



[10] see NetLogo Models Library

Example of NetLogo model[11]

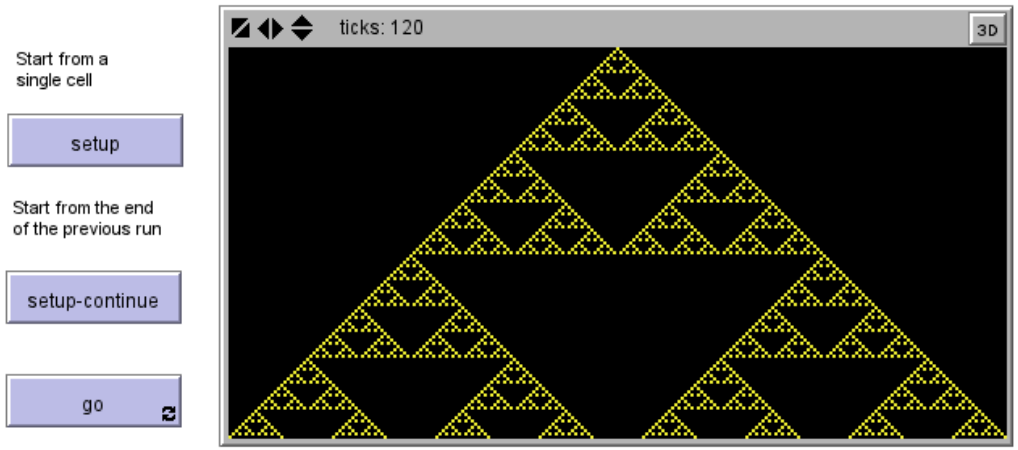
Sand



[11] see NetLogo Models Library

Example of NetLogo model[12]

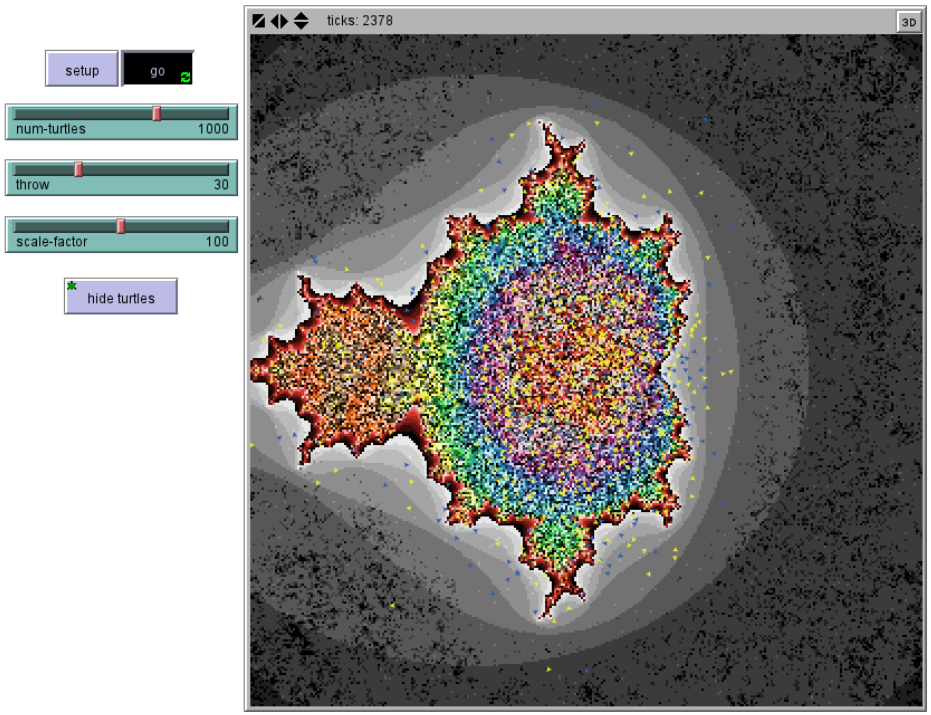
Cellular Automata Rule 90



[12] see NetLogo Models Library

Example of NetLogo model[13]

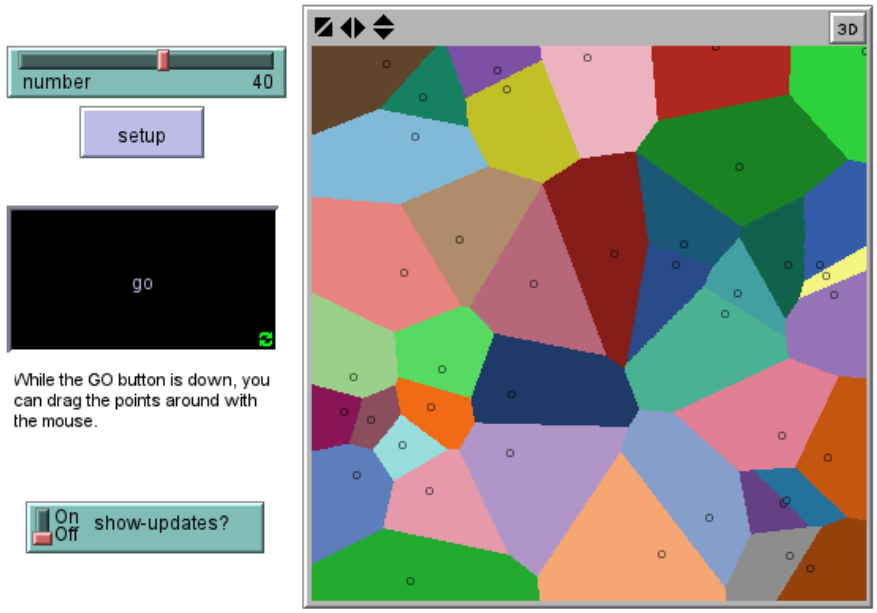
Mandelbrot set



[13] see NetLogo Models Library

Example of NetLogo model[14]

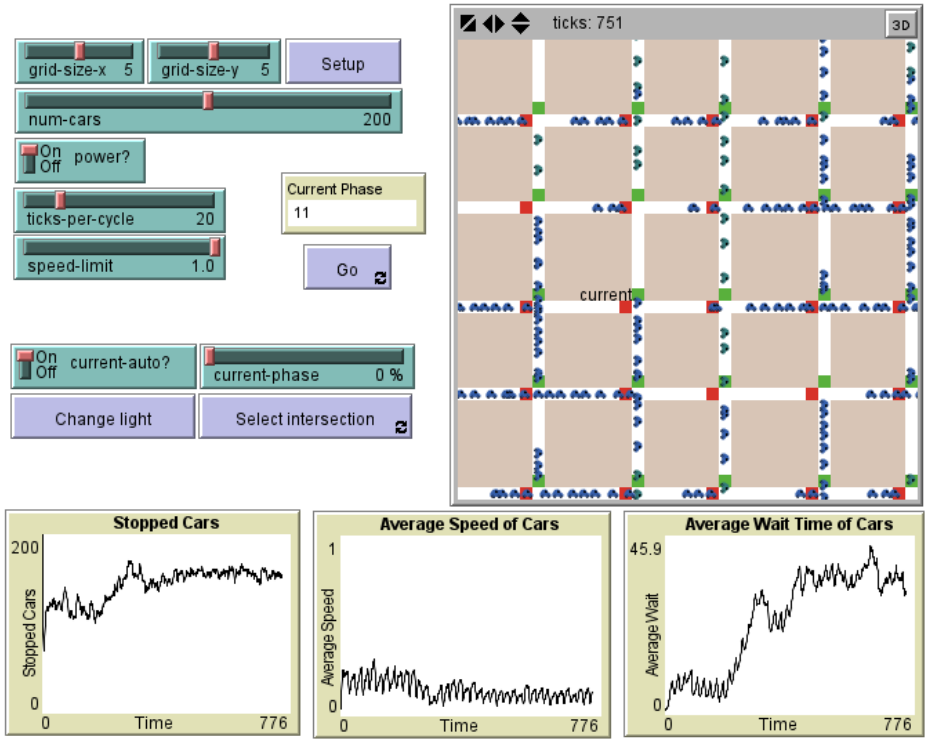
Voronoi set



[14] see NetLogo Models Library

Example of NetLogo model[15]

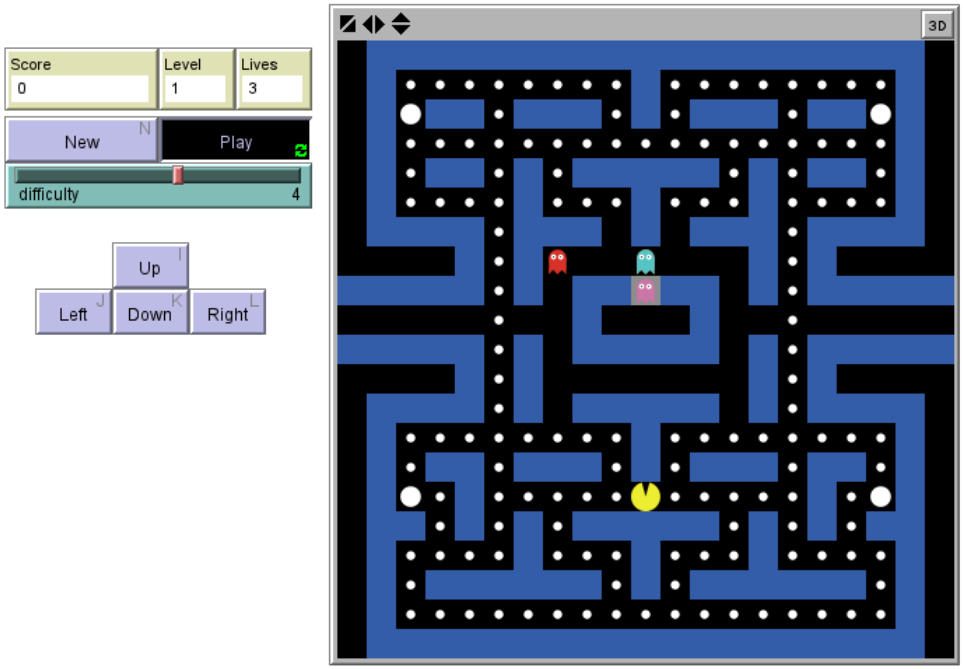
Traffic grid



[15] see NetLogo Models Library

Example of NetLogo model[16]

Game Pac-Man



[16] see NetLogo Models Library

Simulation of a model with NetLogo

- **Two main windows in NetLogo** :

  The **interface graphic** / The **code**

  - Variables can be created from the interface or from the code
  - The code is executed from the interface

- Two mains **objects** with own features and functions :

  - The **Agents** (initially turtles)
  - The **Patches** (the space is an orthogonal lattice)

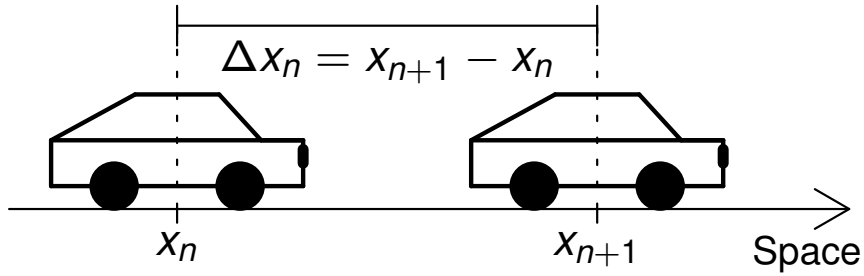
Example: Simulation of road traffic flow

- **Microscopic uni-directional flow models** can be
  - **Speed** function (first order)
  - **Acceleration** function (second order)
- Simulation of the continuous **acceleration model** (differential equation)

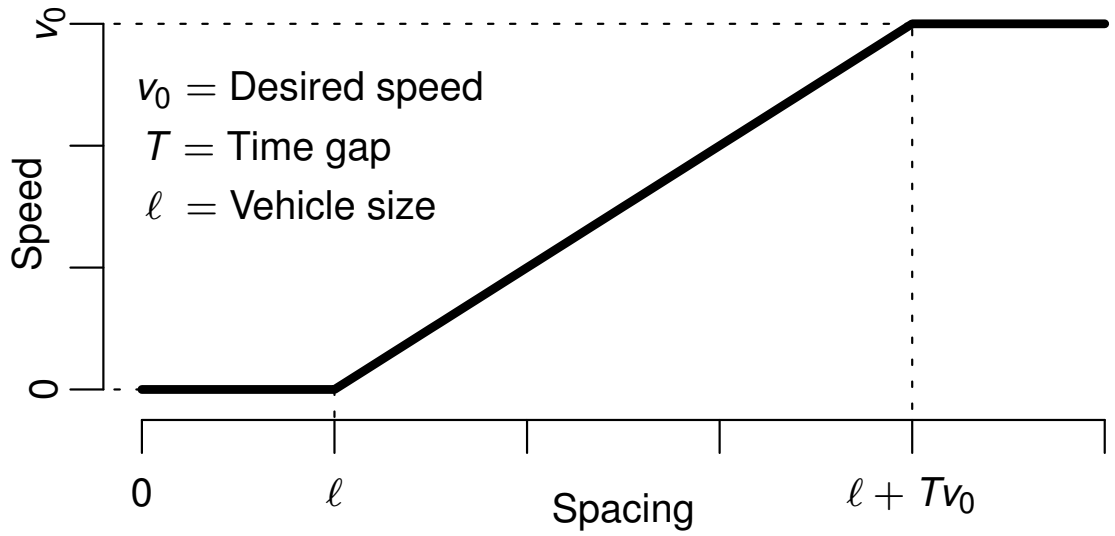
$$\ddot{x}_n(t) = \frac{1}{\tau}[V(\Delta x_n(t)) - \dot{x}_n(t)]$$

  - ⇝ Targeted speed $V(\cdot)$ function of the spacing delayed by reaction time τ

x	position
$\dot{x}$	speed
$\ddot{x}$	acceleration
Δx	Spacing



Example of targeted speed function



- **Free state** if the spacings are higher than $\ell + Tv_0$: **Speed equal to desired one** v_0
- **Congested (or interactive) state** if the spacings are smaller than $\ell + Tv_0$: Speed proportional to the spacing to keep **constant time gap** T with the predecessor

NetLogo Traffic model

Initialisation

- **Interface graphic :**
  - In Settings : `max-pxcor = 20 max-pycor = 1` (1*D* lane)
  - Creation of a button entitled `set-up`
- **Code :**

```
breed [agents agent]                          Creation of object 'agent'
globals [time]                                Global variables
agents-own [speed]                            Agents-own variables

to set-up                                     Function set-up
  clear-all
  create-agents 20 [set heading 90 set xcor random-xcor]
end
```

NetLogo Traffic model

Distance spacing and speed function

- **Interface graphic**
  - Creation of sliders `Desired-speed` and `Time-gap`
- **Code :**

```
to-report dist-gap                                   Function calculating the distance spacing
  let p one-of agents with [xcor > [xcor] of myself] with-min [xcor]
  ifelse p = nobody
    [report world-width + min [xcor] of agents - xcor]
    [report [xcor] of p - xcor]
end

to-report V [d]                                      Speed function of the distance gap
  report max(list 0 min(list Desired-speed (d / Time-gap)))
end
```

NetLogo Traffic model

Programming the model

- **Interface graphic :**
  - Creation of a button `motion`
  - Creation of sliders `dt` and `Reaction-time`
- **Code :**

```
to motion                                   Function for the agent movements
  ask agents [set speed
    speed + dt * (V(dist-gap) - speed) / Reaction-time]
  ask agents [forward(dt * speed)]
  set time precision (time + dt) 5
end
```

→ Repetition of callings to the function `motion` through box `forever`

NetLogo Traffic model

Plotting the trajectories

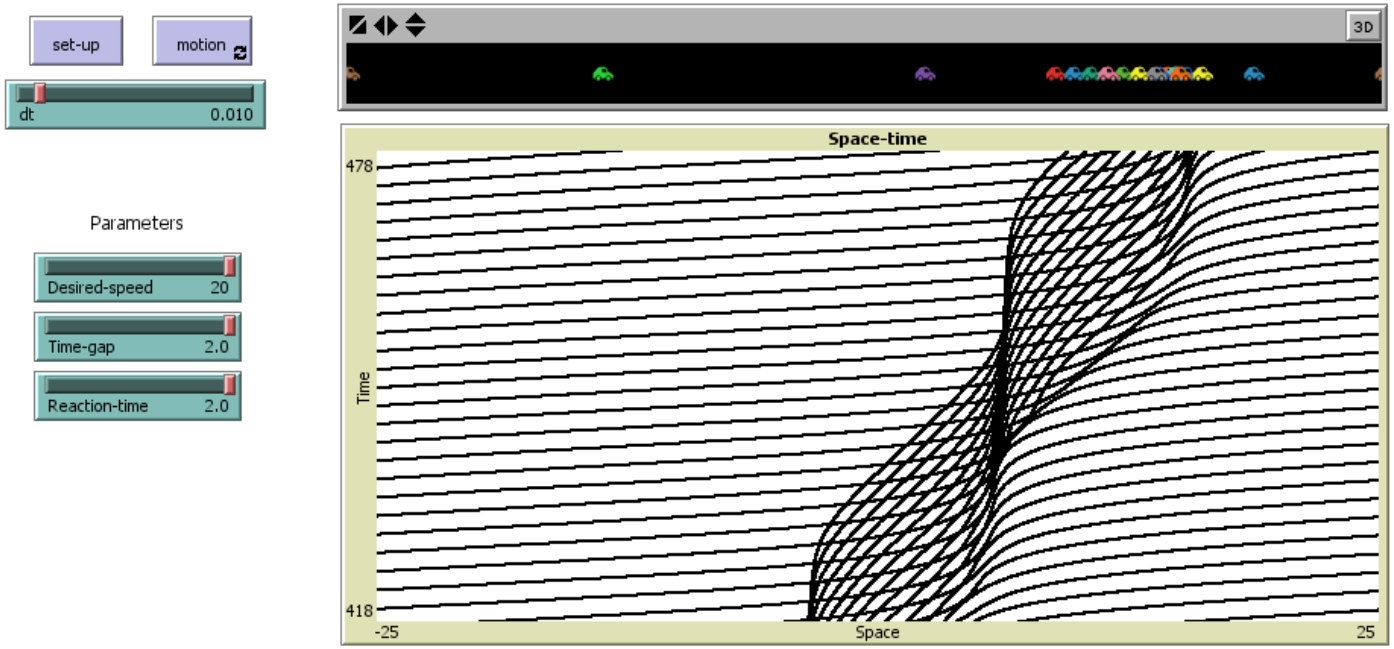
- **Interface graphic :**
  - Creation of a plot entitled `space-time`
  - `Pen` set to `point`
- **Code :**

```
to plot!                    Function plotting agent positions over the time
  ask agents [plotxy xcor time]
  set-plot-y-range round(time - 60) round(time)
end
```

  → Function having to be called with the `motion` function

NetLogo Traffic model

Interface graphic



NetLogo Traffic model

Global code

```
breed [agents agent]
globals [time]
agents-own [speed]

to set-up
  clear-all
  create-agents 20 [set heading 90 set xcor random-xcor]
end

to-report dist-gap
  let pred one-of agents with [xcor > [xcor] of myself] with-min [xcor]
  ifelse pred = nobody
    [report world-width + min [xcor] of agents - xcor]
    [report [xcor] of pred - xcor]
end

to-report V [d]
  report max(list 0 min(list Desired-speed (d / Time-gap)))
end

to motion
  ask agents [set speed speed + dt * (V(dist-gap) - speed)/ Reaction-time]
  ask agents [forward(dt * speed)]
  set time precision (time + dt) 5
end

to plot!
  ask agents [plotxy xcor time]
  set-plot-y-range round(time - 60) round(time)
end
```

Simulation results

- **Two types of dynamics** according to **reaction time τ and time gap T values**
  - **Stop-and-go phenomena** with collisions for **high reaction times :**

  - **Homogeneous flows** with constant speeds for **low reaction times :**

- **Critical relation for the phase separation :** $\tau < 2T$