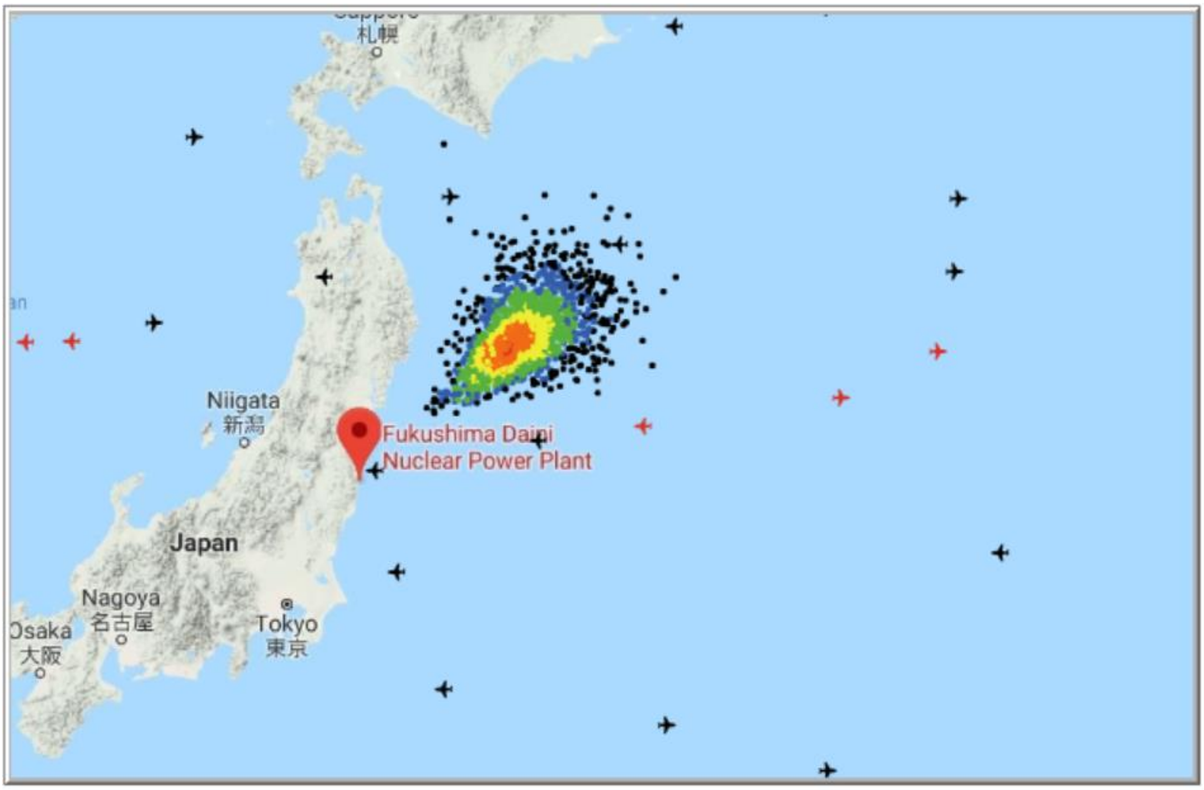
2019

SOFTWARE SIMULATION of a NUCLEAR DISASTER, IMPACT on AIR TRAFFIC and AIR POLLUTION by IODINE-131 with NETLOGO by Alexandre Genette – DSTI Autumn 2018

Alexandre Genette
DSTI Cohort Autumn 2018
20/04/2019

Student Name: Alexandre Genette
Cohort: DSTI Automn 2018

SOFTWARE SIMULATION of a NUCLEAR DISASTER, IMPACT on AIR TRAFFIC and AIR POLLUTION by IODINE-131 with NETLOGO



Introduction:

For this project I have chosen to code my own NetLogo algorithm. All information below can also be found in the NetLogo program in the info menu.

WHAT IS IT?

The model tries to simulate the nuclear cloud (here composed of Iodine 131 particles) after the first days of Fukushima Daiichi nuclear disaster that occurs the 11th of March 2011 in Japan and its potential impact on the air traffic.

HOW IT WORKS

1. AGENT called “PARTICLES” in NetLogo:

It represents the IODINE-131 radioactive isotope. It is one of the most abundant particles released in the atmosphere after a nuclear disaster (like in Tchernobyl).

First, for the model, the released of Iodine-131 in the air is considered as a PUFF at Tick 0 (t0). The spread of the particles follows the Gaussian-Plume modelization.

For the Gaussian-Plume model you need to know the direction of the wind, the instability of the air (empirical value), the distance between the point at t0 and at the measured time. As distance increases so does the dispersion.

The spread follows a normal distribution and there exists an empirical formula where the sigma of the spread of particles is equal to:
- (factor of instability) * (distance from t0) ^ factor (0.92 or 0.89 depends of the factor)

This table summarizes this equation:

TABLE 4.7 Instantaneous Values for σ_y and σ_z in meters [11]

Parameter	Stability Condition	Equation*
σ_y	Unstable	$\sigma_y = 0.14\,(x)^{0.92}$
	Neutral	$\sigma_y = 0.06\,(x)^{0.92}$
	Very Stable	$\sigma_y = 0.02\,(x)^{0.89}$
σ_z	Unstable	$\sigma_z = 0.53\,(x)^{0.73}$
	Neutral	$\sigma_z = 0.15\,(x)^{0.70}$
	Very Stable	$\sigma_z = 0.05\,(x)^{0.61}$

*x is the distance downwind in meters.

source : “Dispersion for point sources - CE 524 - February 2011”

To represent the spread in width in NetLogo particles will turn following the normal distribution N(0, sigma), for 0 the particle follows the wind direction.
For the distribution in length (of the plume) same equation is applied but it’s multiplied by the wind-force (the more the wind is strong the more the spread in length is too)

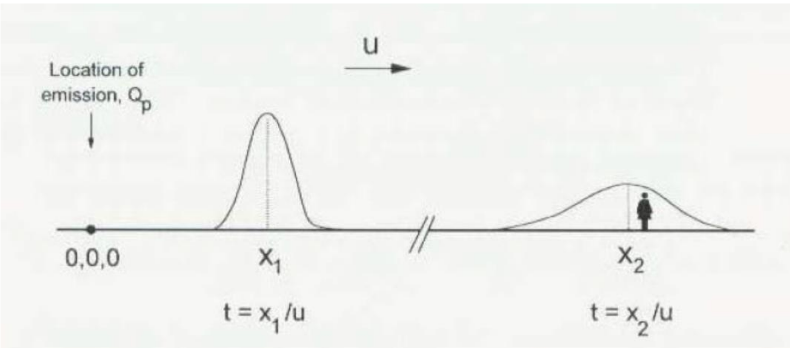
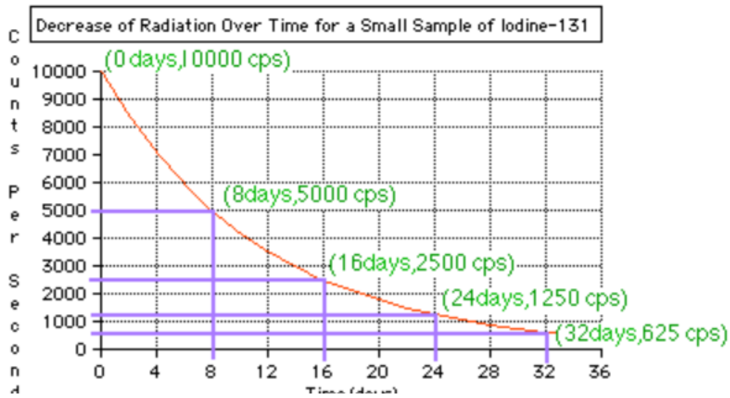


FIGURE 4-9 An instantaneous puff traveling downwind at windspeed, u.

source: “Dispersion for point sources - CE 524 - February 2011”

Half-Life of IODINE-131:

As a radioelement, IODINE-131 has a half-life that is 8.025 days. It means that every 8.025 days the concentration in IODINE-131 is divided by two.



ln(concentration at T0) / ln(concentration at T) = k * T, with k the decay constant

For IODINE-131 that constant is 9.9967×10^-7 per second

We will monitor the decrease of IODINE-131 in percentage with a plot in the interface window.

1. AGENT called "PLANES" in NetLogo:

Virtual airplanes are created (some heading to east and some heading to west).

When an airplane "sees" particles in front of it, it changes his course doing a U-turn and changing its color to red.

If an airplane goes through a patches containing a particle it is considered as contaminated. It gets the label "irradiated" and changes its color to green.

This modelization is pretty simple, but a more accurate model could be done using real data of flights in the area. It could possibly anticipate what an airplane would do in case of nuclear cloud in the area (changing destination airport, giving iodine pills to passengers…).

HOW TO USE IT

- You choose the stability condition with a Slider:
  0.14 (unstable)
  0.06 (neutral)
  0.02 (stable)
- The direction of the wind with a slider, knowing that 0 is North and 90 is east.
  The 11th March in Fukushima the wind blows to north-east (~50°)
- IODINE-131 concentration plots: Quantity of IODINE in the atmosphere in percentage
- Airplane Status plot:
  - in Black : the number of airplanes not affected by the radioactive cloud
  - in Red : the number of airplanes doing a U-turn
  - in Green : the number of airplanes affected by radioactivity

THINGS TO NOTICE

We consider that 1 tick is equal to 1 hour so:
For the half-life equation I had to calculate it manually for 1 hour and the result is 99.6407655962586
The probability that a particle disappears is:
exponential ((T * 24 * 3600 * (- 9.9967 * 10 ^ -7) + ln 100))
For T = 8.025 days, the result is 50 % of course (for 1 hour it is 99.64)
Over 2000 particles the computation time starts to be pretty slow
The model was fit to be as close as possible from the real event after 48H (position and shape of the radioactive cloud)

THINGS TO TRY

Moving the instability factor

EXTENDING THE MODEL

This model is a prototype and it has its own limitation:
- The gaussian-plume is not the most complicated model existing
- Once the wind direction is chosen you cannot alter it
- airplanes follow a really “dumb” path (just heading to east or west)
- because of limitation of particles (computation time) it happens that an airplane can’t see it is heading directly to the radioactive cloud and goes through it.
- other explosion occurred during the next hours and days and more radioactive elements were released in the atmosphere. This model only creates the first Puff.

You could predict the impact if another nuclear plant disaster would occur on a different place in the world by changing the map and the initial parameters.

NETLOGO FEATURES

N/A

RELATED MODELS

I coded everything from scratch couldn’t find a model close to this one, especially for the Gaussian-Plume modelization.
For the code I followed the NetLogo Help and got some inspiration with the code studied during the courses with “sheep and wolves” .

CREDITS AND REFERENCES

Gaussian-Plume theory: “Dispersion for point sources - CE 524 - February 2011” (pdf file)
http://home.engineering.iastate.edu/~leeuwen/CE%20524/Notes/Dispersion_Handout.pdf

Wikipedia: https://en.wikipedia.org/wiki/Fukushima_Daiichi_nuclear_disaster

IRSN official document : “Summary of the Fukushima accident’s impact on the environment in Japan, one year after the accident” http://www.irsn.fr/en/publications/thematic/fukushima/documents/irsn_fukushima-environment-consequences_28022012.pdf

CODE:

```
globals [
          halflife ;half-life of iodine 131, it 8.025 days
          dist ;distance from the origin of the puff
          concentration ;concentration of iodine particles
          factor ;empirical factor for gaussian-plume for atmosphere instability
0.89 for low variability 0.92 for moderate to strong
        ]

breed [ particles particle ] ;simulate particles of iodine 131
breed [ planes plane ] ;simulate planes in the sky


to setup ;initialization

  set halflife 100 ;set to 100%

  resize-world -310 310 -200 200 ;resize the world to the size of the .jpeg
  clear-all ;clear everything
  ask patches [ set pcolor black ]
  set-patch-size 1 ;change the size of patches to 1
        import-drawing "fukushima_ocean.jpg" ;import the map was copied from
      google map must be in the same directory than the code

  ;CREATION of PARTICLES
  ;---------------------
  create-particles 2000 [ ;create iodine particles
    ; then initialize their variables
    set color blue
    set size 8  ; easier to see
    setxy random-normal -127 0.1 random-normal -40 0.1 ;coordinates of fukushima
in the map
    set heading wind-direction ;choose the direction of the wind with the slider
    set shape "dot" ;shape of the particles
  ]

  ;CREATION OF PLANES
  ;------------------
  create-planes 10 [ ;creation of randomly positioned planes heading to west
    set color black
    set size 10
    set shape "airplane"
    setxy random-xcor random-ycor
    set heading 270  ;to west
  ]
  create-planes 10 [ ;creation of randomly positioned planes heading to east
    set color black
    set size 10
    set shape "airplane"
    setxy random-xcor random-ycor
    set heading 90 ;to east
```

```
  ]

  reset-ticks ;set tick counter to 0
end ;end of initialization

to go

  ifelse stability-condition = 0.02 [set factor 0.89 ] [ set factor 0.92 ] ;change factor according to the chooser button
  if ticks = 78 [ stop ] ;end the run after 78 cycles/ticks

  ask particles [
    set dist distancexy -127 -40 ;calculate distance between fukushima (-127, 40) and the position of the particle (distancexy)
    rt random-normal 0 ( (stability-condition * (dist ^ factor)) ) ;empirical Gaussian-plume spreading sigma in width
    fd random-normal (wind-force) ( wind-force * ( dist ^ factor ) ) ;empirical Gaussia-plume spreading sigma in length

    move ;call of the function "move" to move the particles (see below)

    set halflife 99.6407655962586 ; for 1 hour probability --> halflife e ^ (T * 24 * 3600 * (- 9.9967 * 10 ^ -7) + ln 100) with T in day, for 1 hour the result is ~99.641%
    if random-float 100 > halflife [  die ] ;calculate probability and if the random is higher the particle dies

    ;CLOUD VISUALIZATION OF PARTICLES
    ;---------------------------------
    ;ARBITRARY CHOICE OF VARIABLES JUST TO LOOK NICE
    set concentration count particles in-radius 5 ;create a gradient of color to draw the cloud in the interface
      if concentration > 100 [set color red ]
      if concentration < 80 [ set color orange ]
      if concentration < 60 [ set color yellow ]
      if concentration < 40 [ set color green ]
      if concentration < 20 [ set color blue ]
      if concentration < 10 [ set color black ]
  ]

  ask planes [
    if any? particles-on patch-ahead 100 ;the plane checks if there are particles ahead of it
      [ lt 180 set color red ] ;if particle ahead of the plane U-Turn and change its color to red
      fd 20 ;speed of the plane
    if any? particles-on neighbors [ set color green set label "IRRADIATED" set label-color red]
    ;if there are particles in the 8 patches surrounding the plane it is irradiated and change its color to green with a label "irradiated"
  ]
  tick
end

to move  ;turtle procedure to move the particle
 fd random-normal wind-force stability-condition * ticks
end
```