

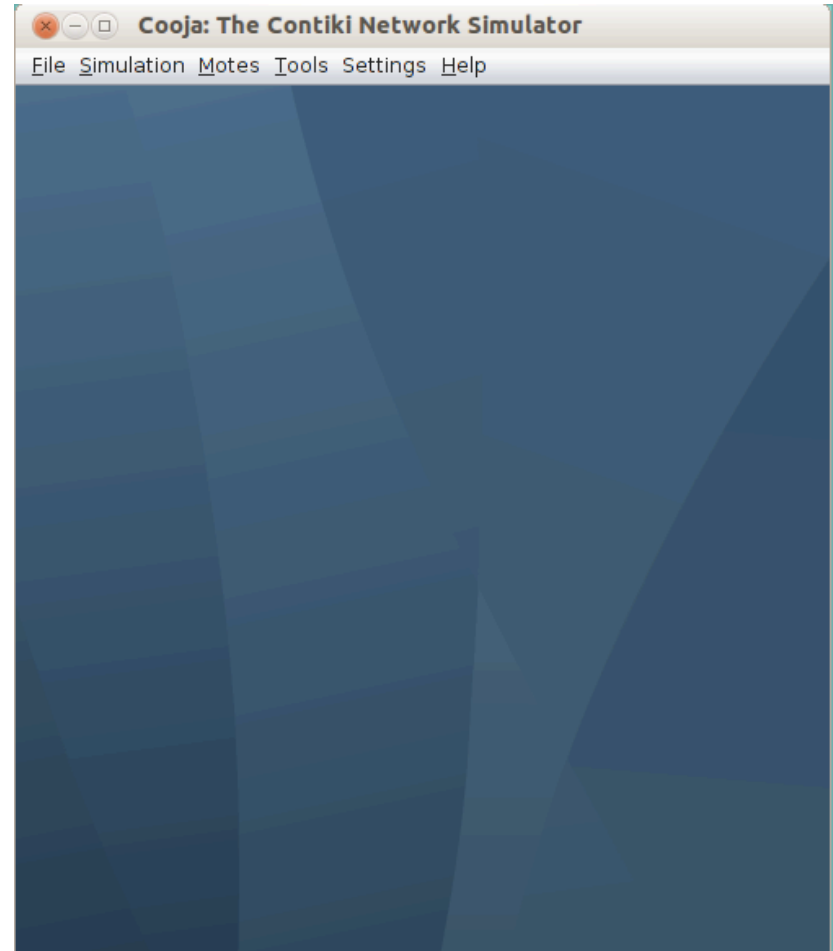
# Cooja Tutorial

# What is Cooja

- Cooja is a cross-layer Java-based wireless sensor network simulator distributed with Contiki
  - It allows the simulation of different levels from physical to application layer
  - Also allows the emulation of the hardware of a set of sensor nodes
- By using Cooja it becomes possible to simulate Wireless Sensor Networks (WSNs) behavior *before* deploying them in the real world
  - Cooja uses for the simulated devices the same source code that will run on the real ones, making possible its validation before deployment on hardware

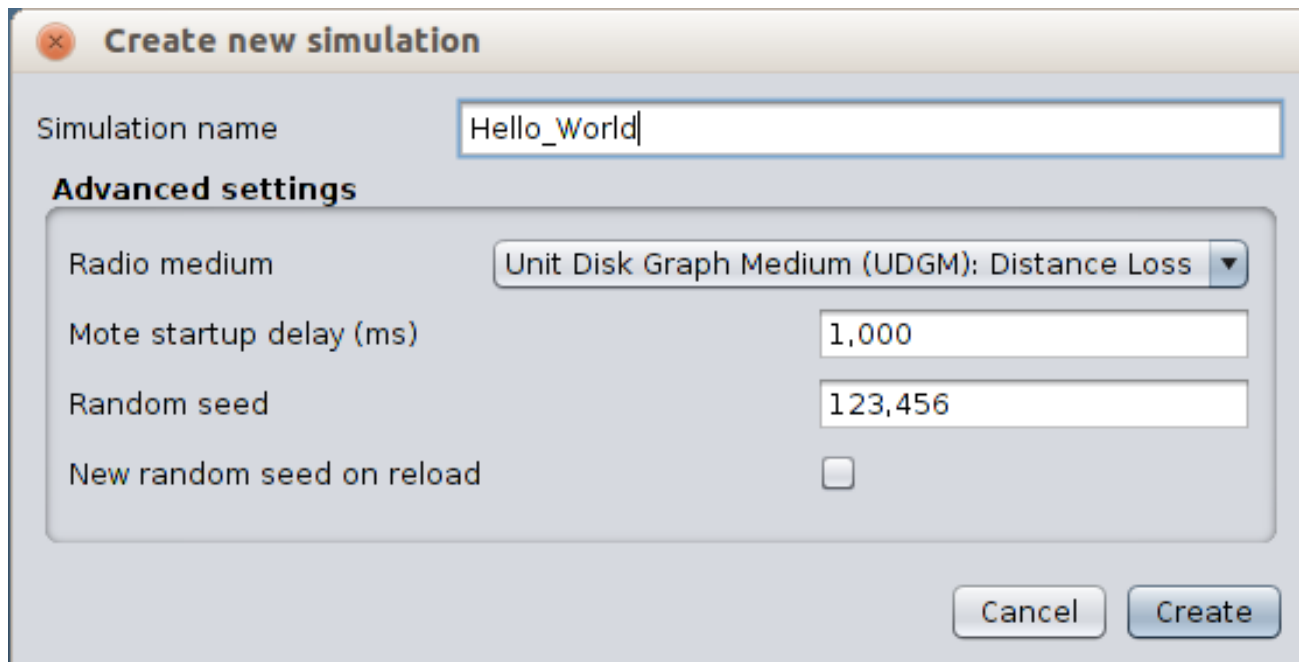
# Running Cooja

- To start Cooja manually
  1. Open a terminal and change the directory to “contiki/tools/cooja/”
  2. Run the command “ant run”
- Once Cooja starts, a window will open as illustrated in the screenshot on the right



# Creating a Simulation

- Click on File > New Simulation
- Give a name to the simulation, then click “Create”



The screenshot shows a dialog box titled "Create new simulation". It contains a text field for "Simulation name" with the value "Hello\_World". Below this is a section titled "Advanced settings" which includes a dropdown menu for "Radio medium" set to "Unit Disk Graph Medium (UDGM): Distance Loss", a text field for "Mote startup delay (ms)" with the value "1,000", a text field for "Random seed" with the value "123,456", and a checkbox for "New random seed on reload" which is currently unchecked. At the bottom right of the dialog are "Cancel" and "Create" buttons.

Simulation name: Hello\_World

**Advanced settings**

Radio medium: Unit Disk Graph Medium (UDGM): Distance Loss

Mote startup delay (ms): 1,000

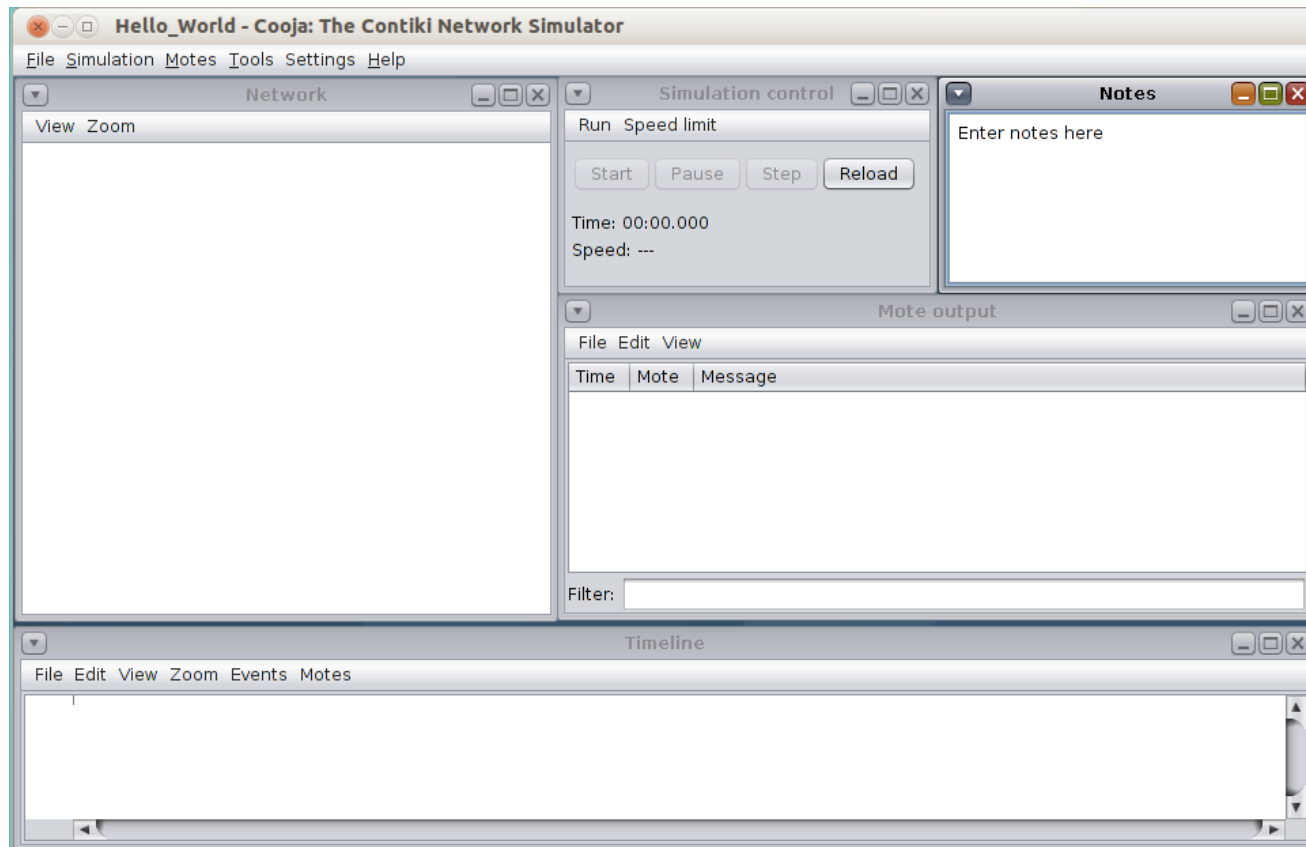
Random seed: 123,456

New random seed on reload: ☐

Buttons: Cancel, Create

# Simulation Interface

- The Cooja graphical interface consists of five areas

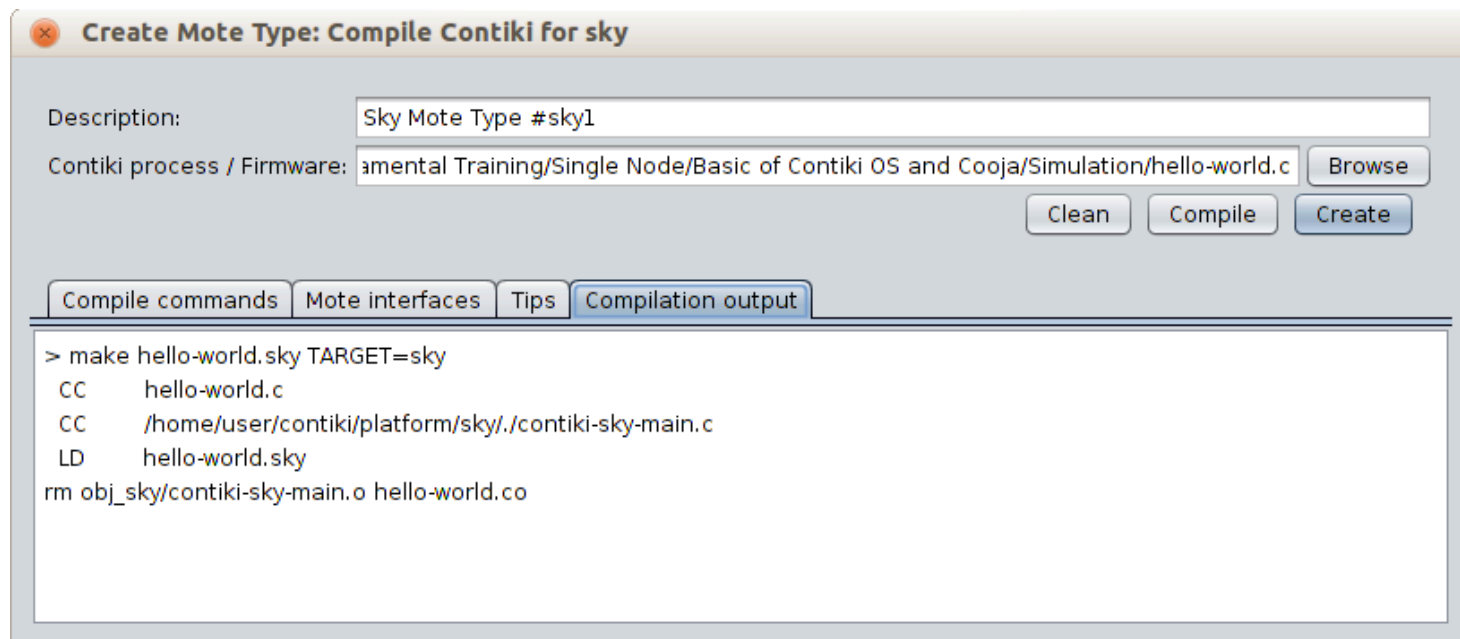


# Simulation Interface (cont.)

- Network window
  - Shows the physical layout of the network
  - You can place motes here and move them around, as needed, in order to form the intended network topology
- Simulation control window
  - Start, stop, reload, and control the simulation rate of the simulation
- Mote output window
  - Shows output generated by all the motes, e.g., the output from the `printf()` function
  - Output can be filtered via input in the “Filter” field (e.g., to only show output from mote 2, then you can enter “ID:2” in this field)
- Note window
  - Allows taking notes during the simulation
- Timeline window
  - Shows the events that occur on each mote during the simulation

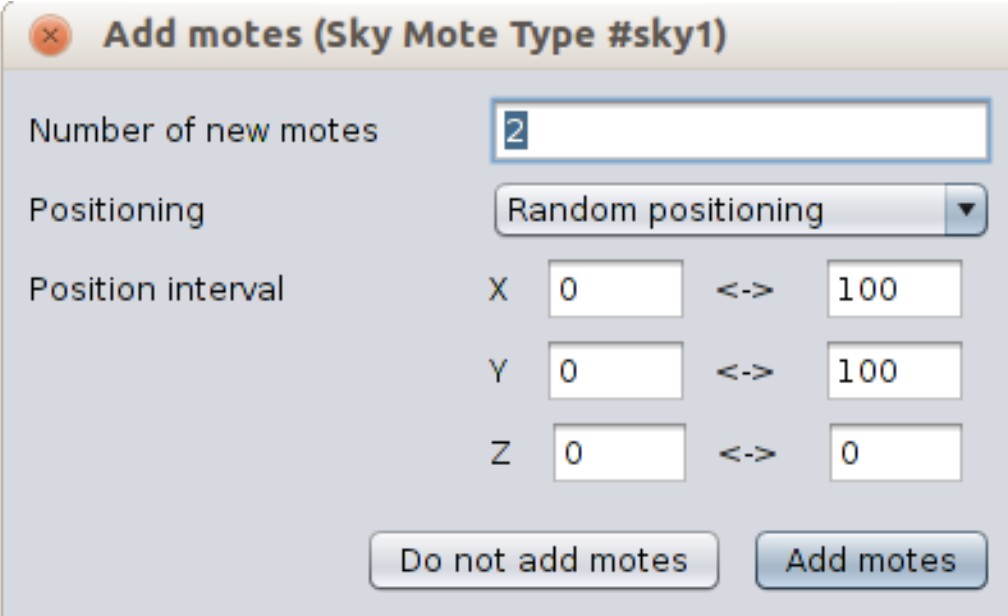
# Adding Motes

- Click on the menu Motes > Add Motes
- In Create New Mote Type, select Sky Mote (also called Tmote Sky)
- In the window that appears, click “Browse” and navigate to the simulation directory “iotrain-sim/database/fundamental\_training/single\_node/basics\_Contiki\_cooja/simulation”
- Select the file “hello-world.c”, click “Compile”, then “Create”



# Adding Motes (cont.)

- After pressing “Create”, a window to configure the number of motes and their positions appears
- Add the number of motes you desire (the example below shows 2 motes being added)



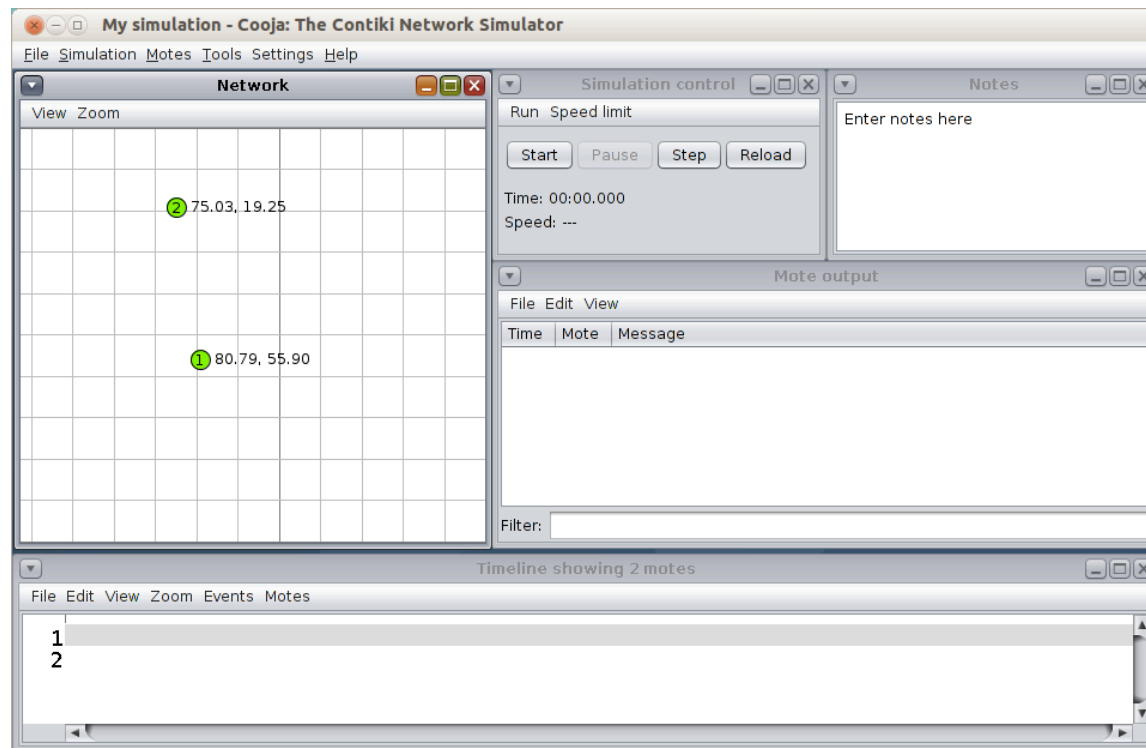
The screenshot shows a dialog box titled "Add motes (Sky Mote Type #sky1)". It contains the following fields and controls:

- Number of new motes:** A text input field containing the value "2".
- Positioning:** A dropdown menu currently set to "Random positioning".
- Position interval:** Three rows of input fields for X, Y, and Z coordinates, each with a range indicator "<->".
  - X: 0 to 100
  - Y: 0 to 100
  - Z: 0 to 0
- Buttons:** Two buttons at the bottom: "Do not add motes" and "Add motes".



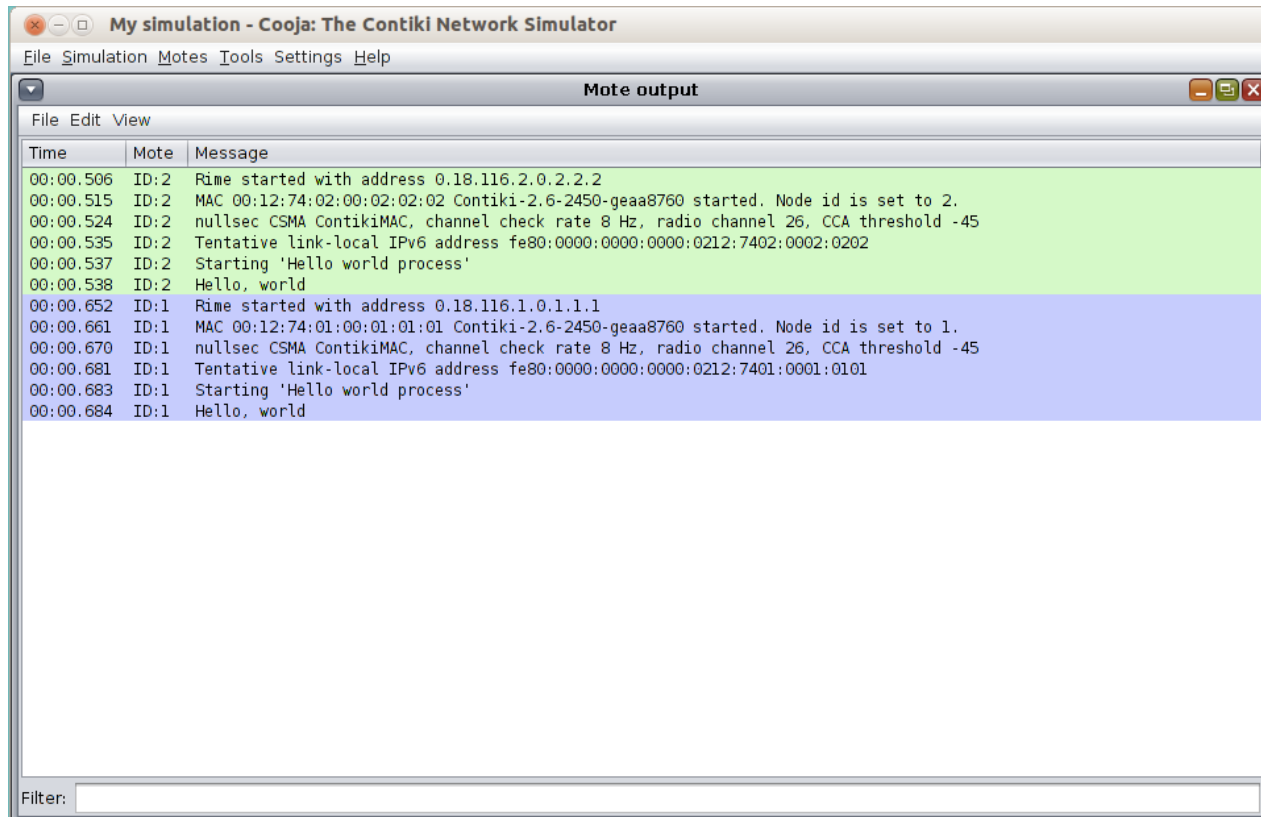
# Visualizing the Scenario

- By default, motes are placed at random positions in the Network window
- Use the View menu in the Network window for view options
- Press the “Start” button to run the simulation



# Mote Output

- The Mote output window displays information about the simulated motes, e.g., MAC address details, messages, etc.



# Saving and Loading Simulations

- Cooja allows to save/load simulation configurations, including all the active plugins
  - Note that the state of a simulation is not saved, and all nodes are reset when the simulation is loaded again
- To save the current simulation, click the menu item File > Save simulation as... and choose the target directory
  - Simulations are stored as files with the extension “.csc”, which means ‘Cooja Simulation Configuration’
- To later open an existing simulation, click the menu item File > Open simulation > Browse... and select the desired simulation configuration file (.csc)