

Network Environments in AnyLogic

Nathaniel Osgood

MIT 15.879

March 16, 2012



Hands on Model Use Ahead



Load your recently created SI model
(provided alternative: `MinimalistSIRNetworkABM`)

The *Environment* defines both Spatial & Network (Topological) Context

The screenshot displays the AnyLogic Advanced software interface. The main workspace shows a simulation environment with various components and variables. A red arrow points to the 'environment' component, which is highlighted in a pink box. The text 'Environment' is written in red over the workspace.

The 'Properties' window for the 'environment - Environment' component is open, showing the following settings:

- Space type: Continuous Discrete GIS
- Width: 500
- Height: 500
- Columns: 100
- Rows: 100
- Neighborhood type: Moore
- Layout type: User-defined Apply on startup
- Network type: User-defined Apply on startup
- Connections per agent: 2
- Capacity per agent: 50

The 'Advanced' section of the Properties window is highlighted in green, and the text 'Spatial Characteristics' is written in green over it. A green arrow points from the text to the 'Advanced' section.

The 'Network Characteristics' section is highlighted in blue, and the text 'Network Characteristics' is written in blue over it. A blue arrow points from the text to the 'Network type' dropdown menu.

The 'Model' palette on the right side of the interface shows various components and variables, including:

- Parameter
- Flow Aux Variable
- Stock Variable
- Event
- Dynamic Event
- Plain Variable
- Collection Variable
- Function
- Table Function
- Port
- Connector
- Entry Point
- State
- Transition
- Initial State Pointer
- Branch
- History State
- Final State
- Environment

The 'Action' palette at the bottom right shows:

- Action
- Analysis
- Presentation
- Connectivity
- Enterprise Library
- More Libraries...

Network Specification in AnyLogic

- When considering networks in AnyLogic, we specify two somewhat distinct (but coupled) things
 - Network topologies
 - Spatial (and visual) Layouts

Networks & Spatial Layouts

- Distinct node attributes: Location & connections
 - Spatial layouts determine where nodes appear in space (and on the screen!)
 - Network type determines who is connected to who
 - For the most part, these characteristics are determined independently
- Network topologies (connectedness) can be defined either *alternative to* or *in addition to spatial layouts*
 - Agents will have spatial locations in either case

Network Types

The screenshot displays the AnyLogic Advanced software interface. The main window shows a project tree on the left with the 'environment' object selected under 'Spatial SEIR with Waning Immunity > Main > Environments'. The 'Advanced' properties panel is open, showing the following settings:

- Space type: Continuous Discrete GIS
- Width: 500
- Height: 500
- Columns: 500
- Rows: 500
- Neighborhood type: Euclidean
- Layout type: User-defined Apply on startup
- Network: Scale free Apply on startup

The 'Network' dropdown menu is open, showing the following options:

- Random
- Ring lattice
- Small world
- Scale free (selected)
- Distance based

The 'Problems' panel at the bottom left shows several error messages related to undefined methods and unresolved types.

Select environment

Layout Types

The screenshot displays the AnyLogic Advanced software interface, titled "AnyLogic Advanced [EDUCATIONAL USE ONLY]". The interface is divided into several panels:

- Project Panel (Left):** Shows a hierarchical tree structure of the simulation project, including "Parameters", "Functions" (AddNewAgent, AddNewAgentA, AddNewAgentB, AddNewAgentC), "Environments", "Embedded Objects", "Presentation", "AgentFactory", "Simulation: Main", and "Spatial SEIR with Waning Immunity" (Main, Parameters, Plain Variables, Environments, Embedded Objects, Presentation, Person, Plain Variables, Statecharts).
- Simulation Environment (Center):** A grid-based workspace showing a simulation environment. The "Person" environment is selected, and its properties are visible on the right: ImmunityDuration, TotalPopulation, AveragellnessDuration, ContactRatePerNetworkK, PerContactInfectionProb, and nSusceptible.
- Properties Panel (Bottom Center):** Shows the "environment - Environment" properties. The "Layout type" dropdown menu is open, displaying options: "User-defined" (highlighted), "Random", "Arranged", "Ring", and "Spring mass". Other properties include "Space type" (Continuous, Discrete, GIS), "Width" (500), "Height" (500), "Columns" (500), "Rows" (500), "Neighborhood type" (Euclidean), "Network", "Connections per node", "Connection range", "Neighbor link fraction" (0.95), and "M" (5).
- Problems Panel (Bottom Left):** Lists several error messages, such as "Cannot make a static reference to the non-static method getState() of the type Person", "The method getCurrentState() is undefined for the type Person", and "Type_statechart cannot be resolved".
- Library Panel (Right):** A vertical toolbar containing various UI components and shapes, including "Model", "Action", "Analysis", "Presentation", "Line", "Polyline", "Curve", "Rectan...", "Round...", "Oval", "Arc", "Pixel", "Text", "Image", "Group", "Button", "Check...", "Edit Box", "Radio...", "Slider", "Comb...", "List Box", "File Ch...", "Progre...", "CAD D...", "GIS Map", "Connectiv...", and "Enterpris...".

Layout Type

- **Random:** Uniformly distribute X and Y position of nodes
- **Arranged:** Set node locations in a regular fashion (normally in a 2D grid)
- **Ring:** Set node locations in periodically spaced intervals around a ring shape
- **Spring Mass:** Adjust node locations such that node locations that are most tightly connected tend to be closer together
 - (Sets location based on network!)
- **User-Defined** User can set location (e.g. in initialization code)

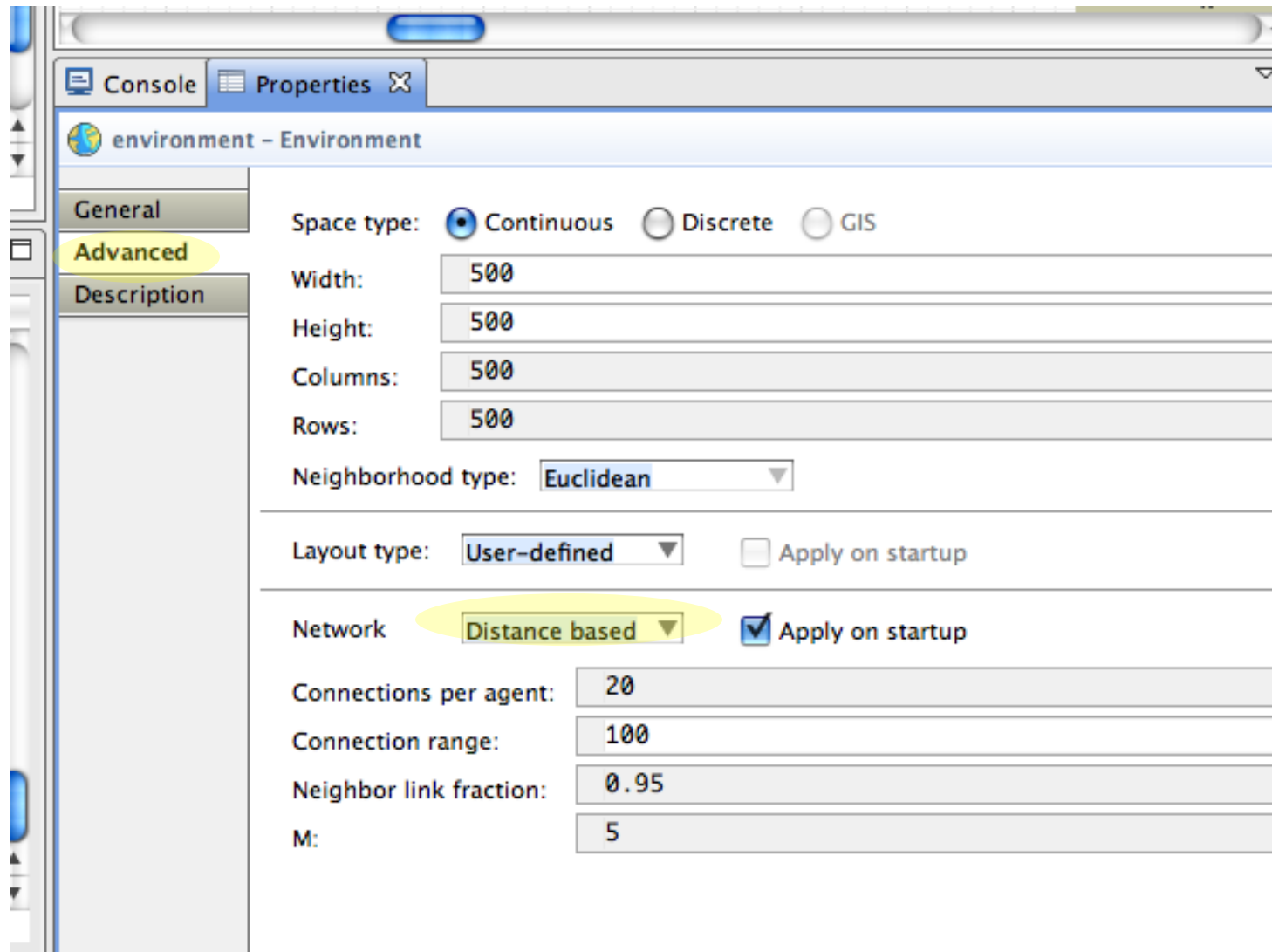
Distance Based Networks

- Function: Capturing geographic locality in networks
- Networks may be *discontinuous* (divided into disjoint *components*) when
 - The threshold is small
 - The density of the spaces (nodes per unit area) is too small

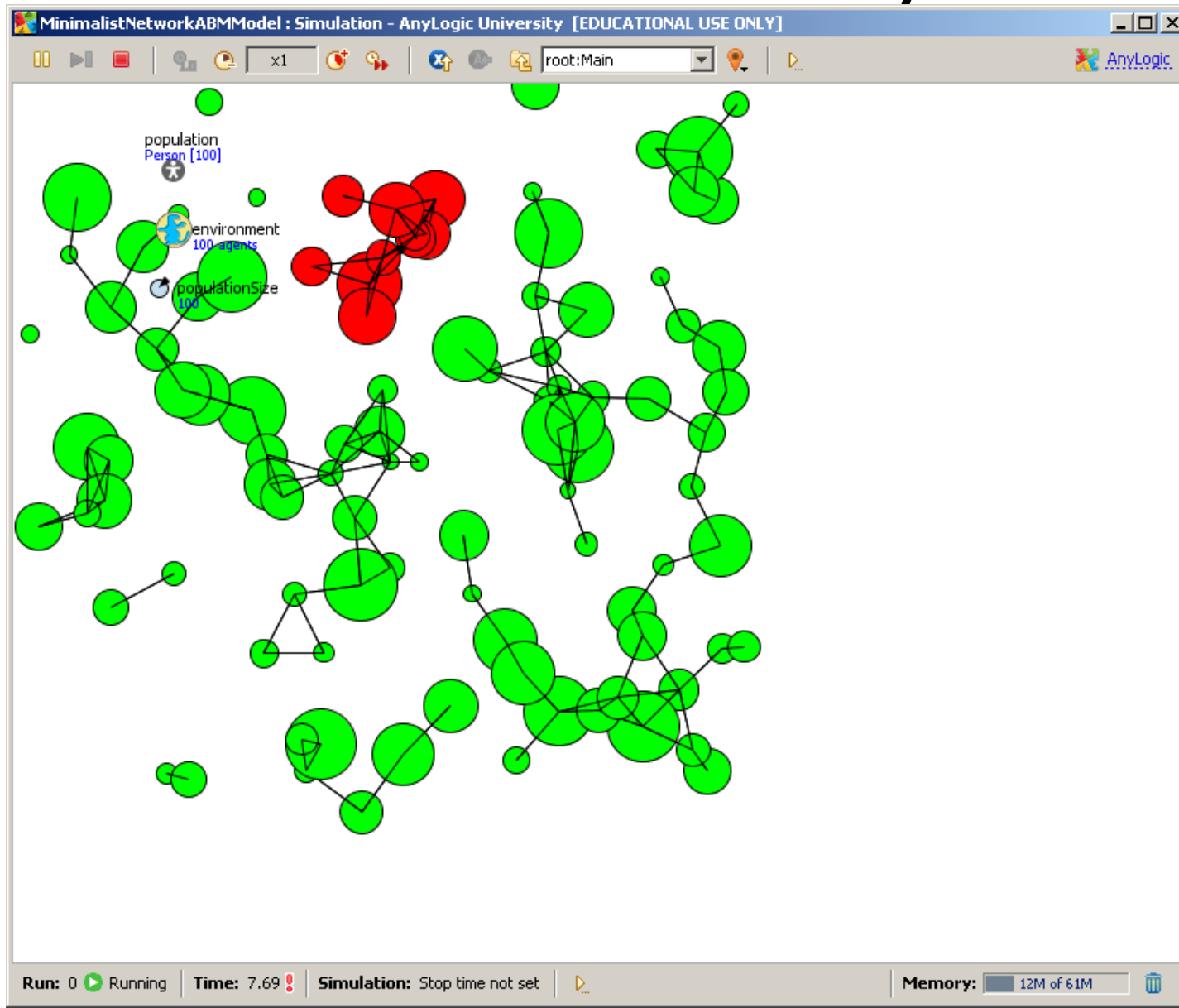
Interaction Between Network&Location 1

- For one type of networks (Distanced Based), whether there is a connection between A and B depends on the *distance* between A & B
 - This sets connectivity based on location considerations!

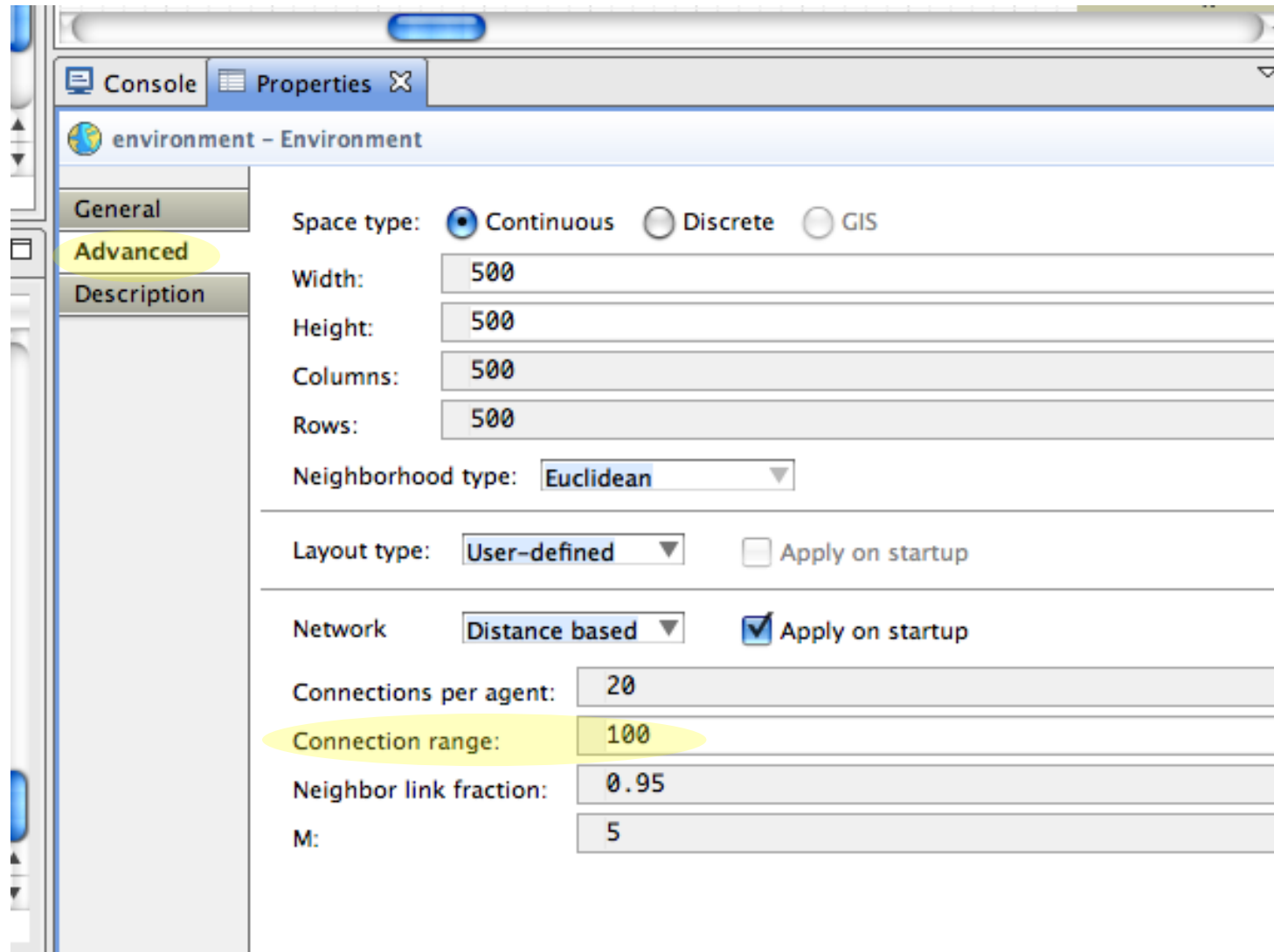
Property for Distance-Based Layout: Distance Threshold



Distance-Based Layout



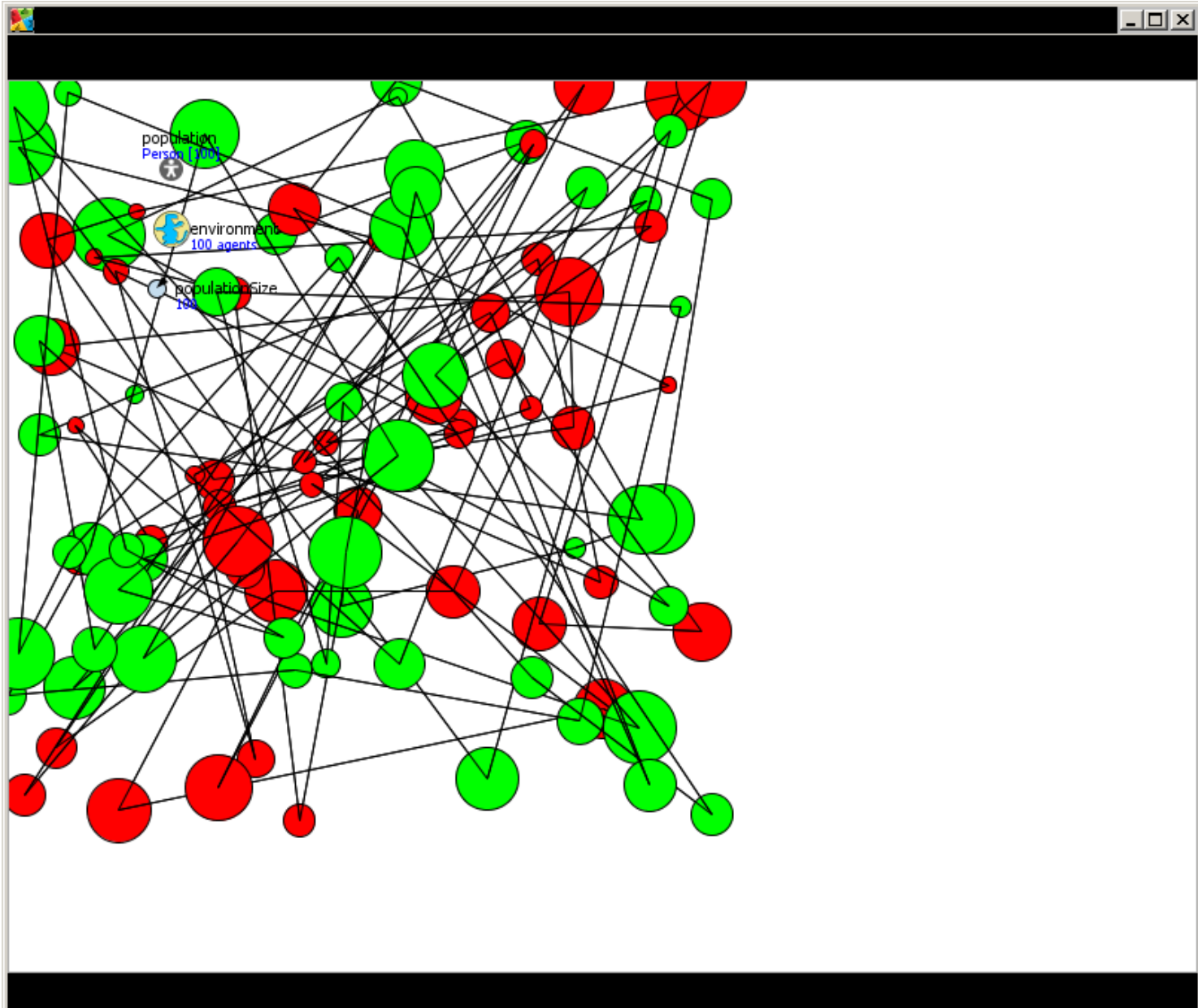
Property for Distance-Based Layout: Distance Threshold



Purely Local Connections: Ring Lattice

- Purely **local** connectivity
 - Agents arranged in a ring
 - Connections by a given agent to some number of agents on either side of itself in the ring
- Slow propagation of infection (no “short cuts” from one region to other regions)
- NB: Most naturally displayed with “Ring” “Layout type”

Ring Lattice – No Ring Layout



Ring Lattice – Choosing Ring Layout

The screenshot displays the AnyLogic software interface for configuring an environment. The main workspace shows a grid with a central circle and a vertical line, representing the environment layout. The 'environment' object is highlighted in the workspace, and its properties are shown in the 'Properties' panel below.

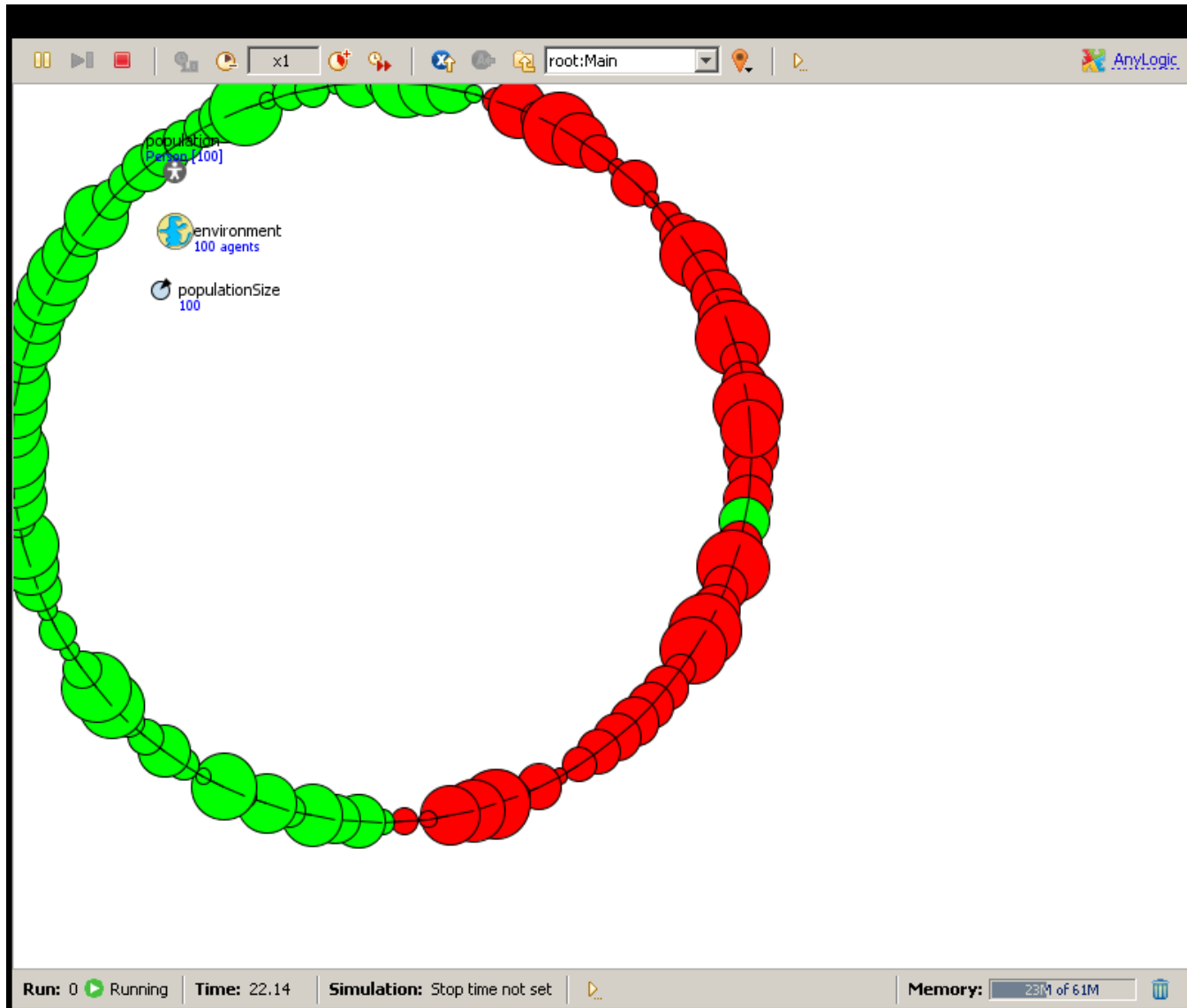
The 'Properties' panel for 'environment - Environment' is open, showing the following settings:

- Space type: Continuous2D Continuous3D Discrete2D GIS
- Width: 500
- Height: 500
- Z-Height: 0
- Columns: 100
- Rows: 100
- Neighborhood type: Moore
- Layout type: Ring (highlighted in a yellow box)
- Apply on startup:
- Network type: Ring (highlighted in a yellow box)
- Apply on startup:
- Connections per...: [empty]
- Connection range: 50
- Neighbor link fraction: 0.95
- M: 10

The 'Layout type' dropdown menu is open, showing the following options: Ring, User-defined, Random, Arranged, Ring, and Spring mass. The 'Ring' option is selected and highlighted in a yellow box.

The 'Network type' dropdown menu is also open, showing the following options: Random, Arranged, Ring, and Spring mass. The 'Ring' option is selected and highlighted in a yellow box.

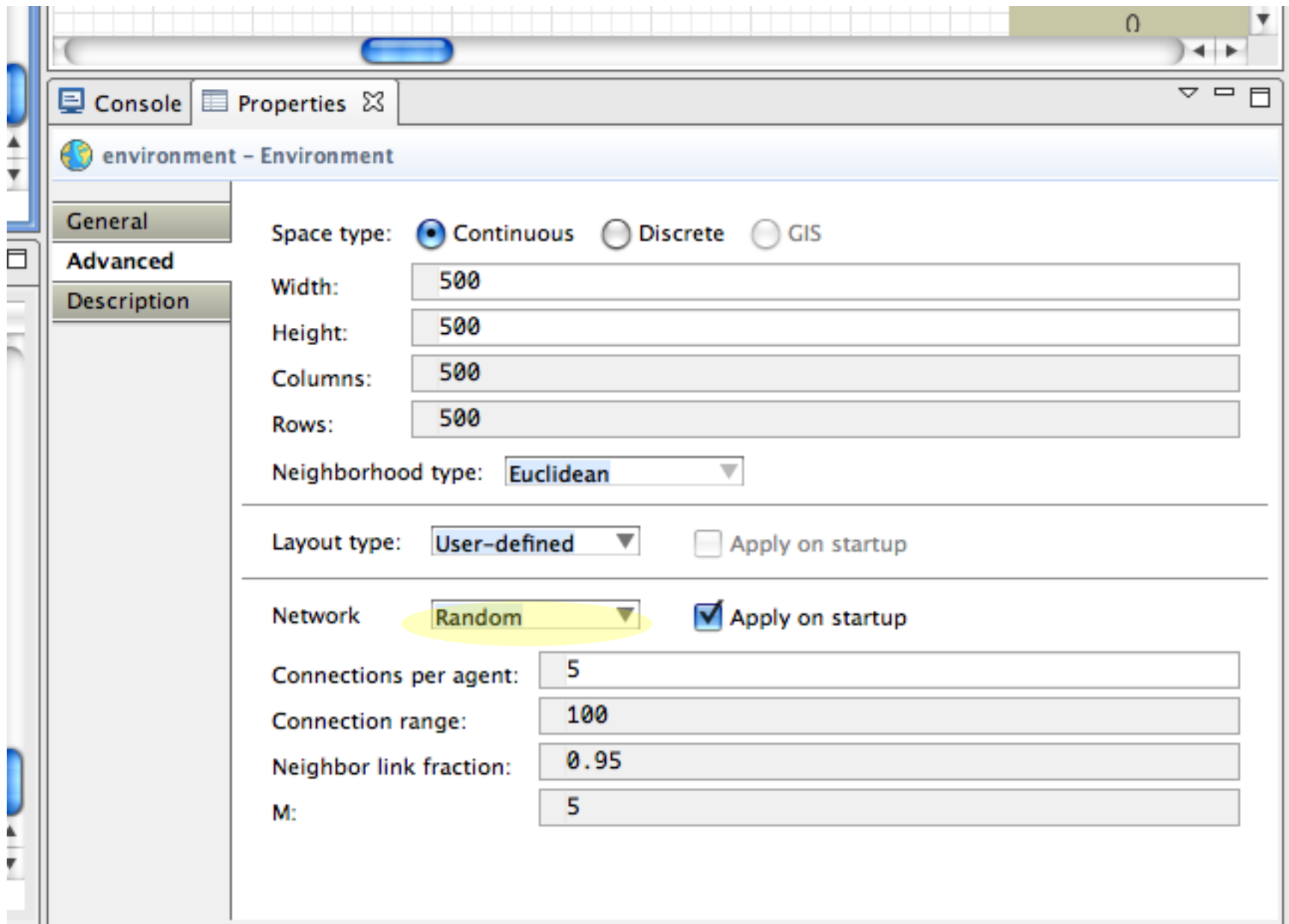
Ring Lattice Topology – With Ring Layout



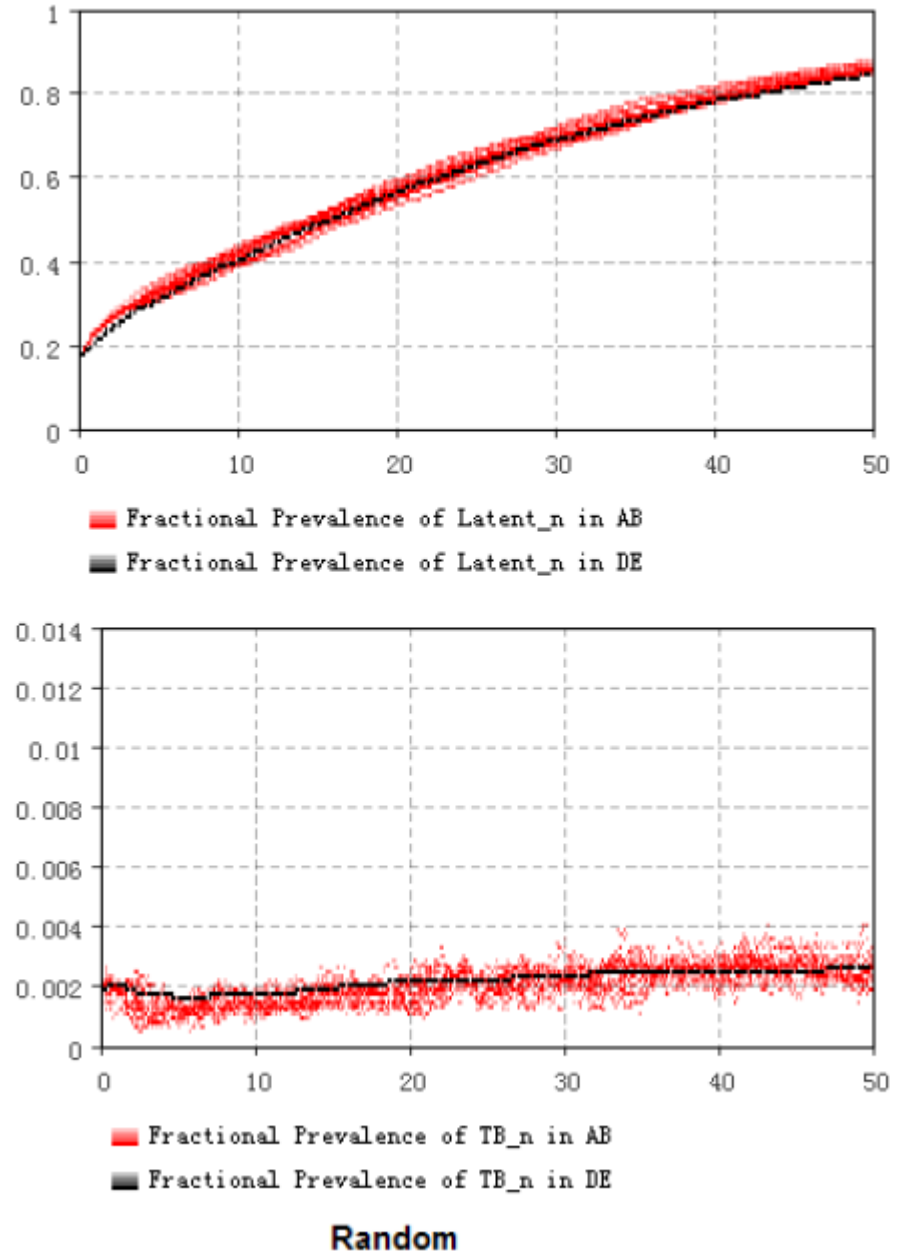
Global Connectivity: Poisson Random Networks

- In Poisson random networks (also called “random networks” or “Bernoulli networks”), any pair of nodes (A,B) exhibits the same chance of connection as any other pair of nodes
- This network type has no preference for any sort of “locality” (topological or spatial)
 - There is no more overlap in the connections of two neighbors than among two arbitrary nodes in the population

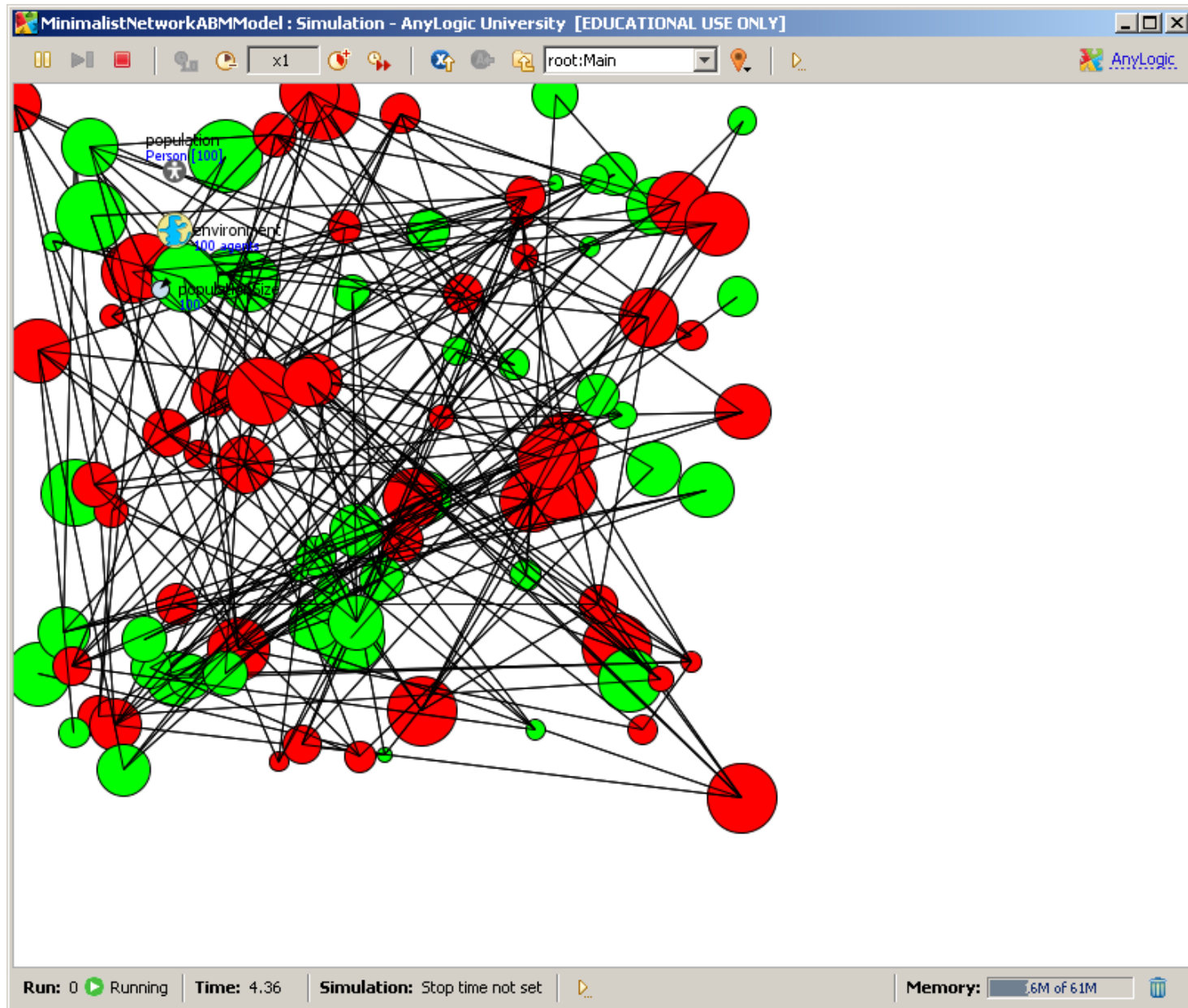
Global Random Mixing: Random Connections



Connections over static Random networks can yield results very similar to what results from continuous, dynamic random mixing in an aggregate model



With Random Connections



Scale-Free Network

The screenshot shows the 'Properties' dialog box for an environment in NetLogo. The 'Network' section is highlighted, showing the 'Scale free' network type selected. The 'Apply on startup' checkbox is checked. Other settings include a width and height of 500, 500 columns and rows, and a Euclidean neighborhood type.

environment - Environment

General

Advanced

Description

Space type: Continuous Discrete GIS

Width: 500

Height: 500

Columns: 500

Rows: 500

Neighborhood type: Euclidean

Layout type: User-defined Apply on startup

Network: Scale free Apply on startup

Connections per agent: 5

Connection range: 100

Neighbor link fraction: 0.95

M: 5

Intuitive Plausibility of Importance of Heterogeneity

- Someone with high # of partners is both
 - More likely to be infected by a partners
 - More likely to pass on the infection to another person
- Via targeted interventions on high contact people, may be able to achieve great “bang for the buck”
- We may see very different infection rates in high contact-rate individuals
- **How to modify classic equations to account for heterogeneity? How affects infection spread?**

Scale-Free Networks

- A node's number of connections (a person's # of contacts) is denoted k
- The chance of having k partners is proportional to $k^{-\gamma}$.
- For human sexual networks, γ is between 2 and 3.5
 - E.g. if $\gamma=2$, likelihood having 2 partner is proportional to $\frac{1}{4}$, of having 3 is proportional to $\frac{1}{9}$, etc.

Power Law Scaling

- This frequency distribution is a “power law” that exhibits invariance to scale
- Suppose we change our scale (“zoom out”) in terms of number of connections (k) by a factor of α

Cf: $p(k) = ck^{-\gamma}$

$$p(\alpha k) = c(\alpha k)^{-\gamma} = c\alpha^{-\gamma}k^{-\gamma} = \alpha^{-\gamma}ck^{-\gamma} = dp(k)$$

In other words, the function $p(k)$ “looks the same” at any scale – it “zooming out” on the scale of # of connections by factor α just leads it to be multiplied by a different constant

- We can get power law scaling from many sources; a key source is dimensional structure
- Power law probability distributions have “long tails” compared to e.g. an exponential or normal

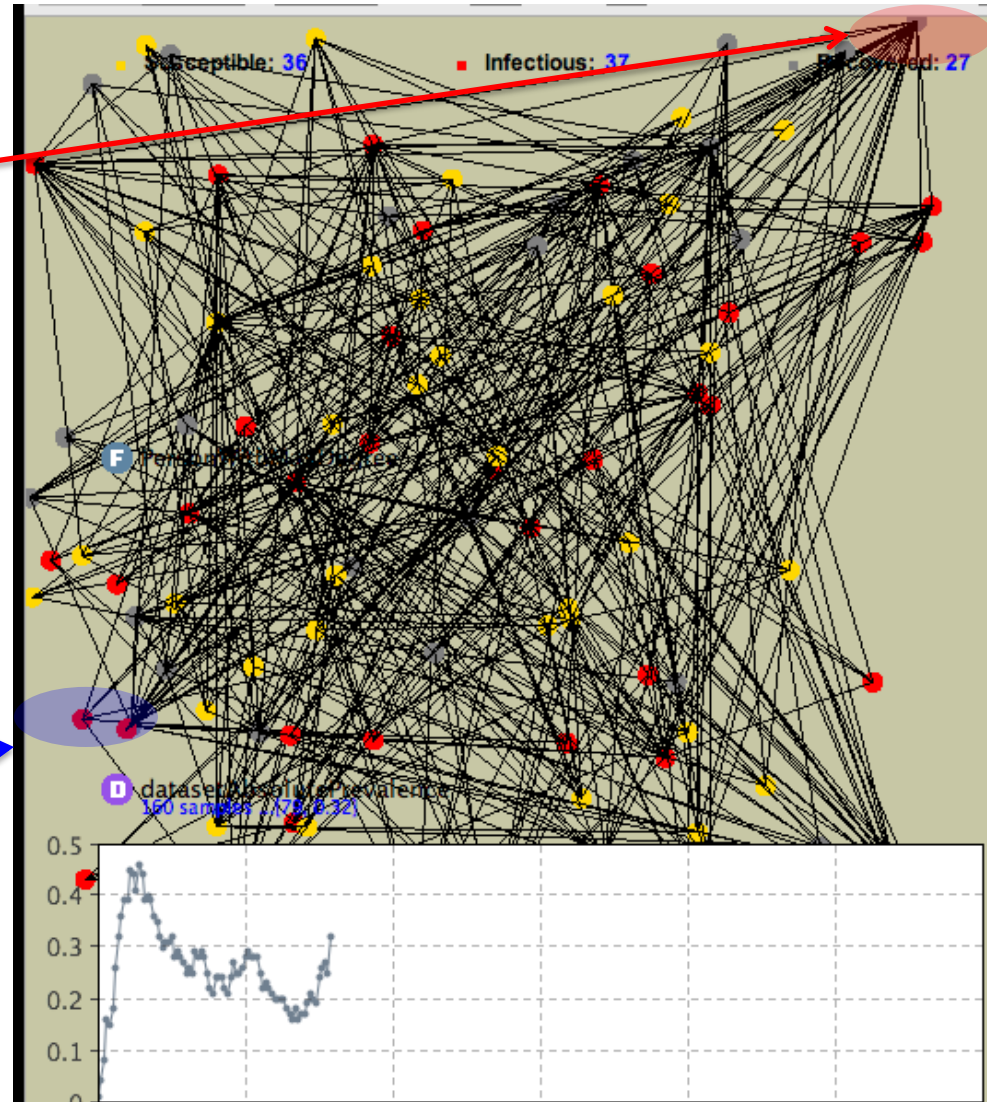
The Signature of a Power Law

- Plotting a power law function on a log-log plot will yield a straight line
 - This reflects fact that $p(k)=ck^{-\gamma} \Rightarrow \log[p(k)]=c-\gamma \log [k]$
 - So if our axes are $v=\log[p(k)]$ and $h=\log[k]$, $v=c-\gamma h$
- This relates to the fact that the impact of scaling (scaling) is always the identical (divides the function by the same quantity)
 - e.g. if $\gamma=2$, doubling k always divides $p(k)$ by 4 (no matter what k is!)
 - In other words, no matter how many connections we may have, the fraction of people with this many connections is 4x the fraction with 2x this many connections!
 - e.g. if $\gamma=3$, doubling k always divides $p(k)$ by 8

Slides Adapted from External Source
Redacted from Public PDF for Copyright
Reasons

Scale-Free Network

High degree node



Low degree node

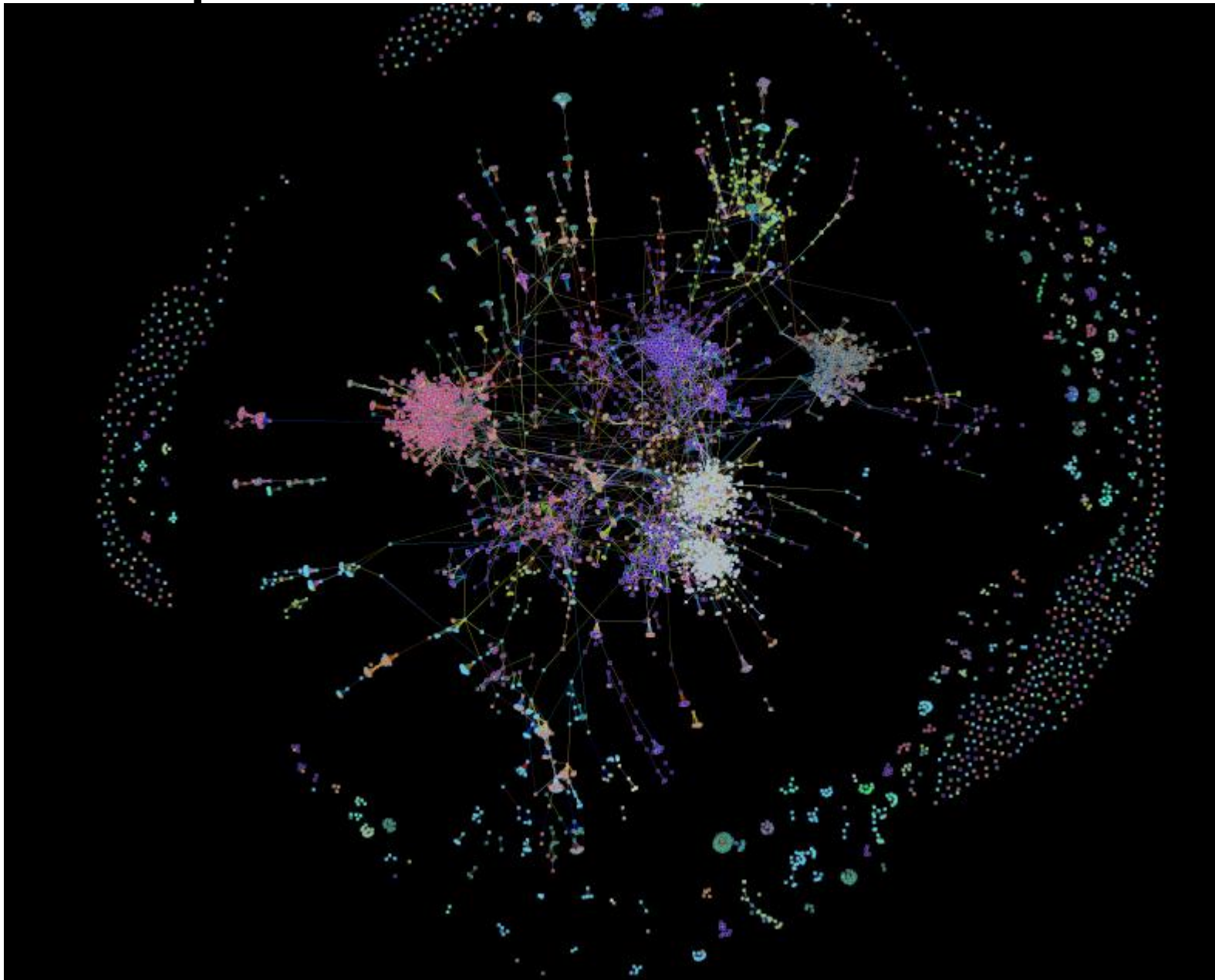
Small World

- Small world networks represent a sort of “weighted combination” of
 - Ring lattice network (purely local connections)
 - Random network (mostly global connections)
 - The “Neighbor link fraction” in AnyLogic dictates what fractions of the connections are to the local neighbors (per ring lattice)
- Beware of the inconsistency in the definition of “Connections per agent” for small world networks
 - Off by a factor of two!

Interaction Between Network & Location 2

- In a Spring-mass layout, the nodes that are highly connected will tend to be clustered
- Here, we are determining the location based on the connectivity!

Example Network Substructure



General ABM Network Caveats

- In thinking about the effects of & tradeoffs between interventions, need to recognize that networks are emergent phenomena, driven by
 - Mobility patterns
 - Relationship formulation & dissolution
- Many networks are dynamic, but traditional measures rarely yield dynamic high temporal resolution data
- We typically have only partial information on network structure
- Often collected via a non-random sampling process
- Networks specific to definition of “contact”

Example: Contact Tracing Networks

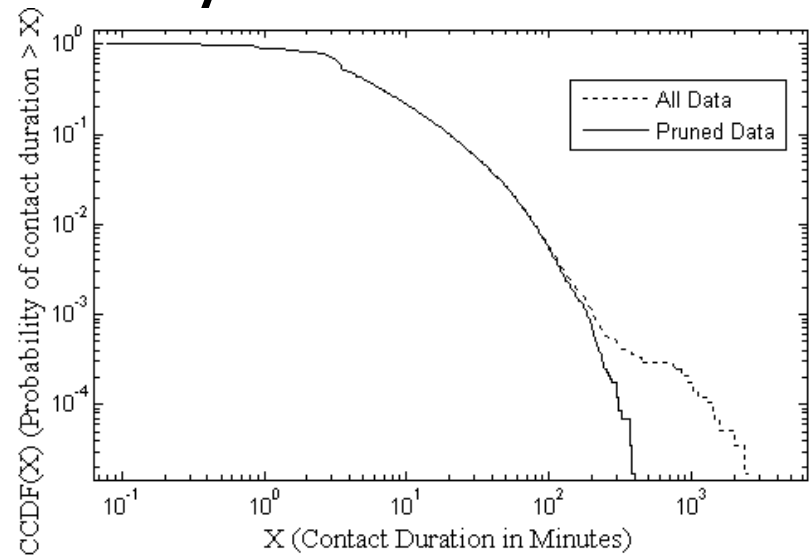
- These are produced by an asymmetric or biased contact tracing protocol
 - Uses definition of contact (e.g. needle-sharing incident, spending >8 hours in past 30 days, past or ongoing sexual relationship)
 - Perform tracing only under certain conditions
- Data at hand is likely collected over a substantial amount of time
 - The network may have changed during this time
- Unclear what this says about the network of the general population

AnyLogic Network Caveats

- Built-in networks are handy for routine tasks, but do not offer much flexibility e.g. preferential attachment, post-construction additions, etc.
- Constructing built-in networks can be computationally expensive
- The “M” parameter in a Scale-Free network would not appear to be either classic parameters γ nor m (from Barabasi paper)
 - **Mean # of connections/person is approximately twice this value**
 - Number of connections per individual are often in discrete categories?
- NB: The “Small world” network uses a definition of connections/person inconsistent with those for other networks
 - Off by a factor of 2!

Network Dynamics in AnyLogic

- Observed networks are often dynamic over a wide range of timescales
- These dynamics can be very important to overall system dynamics.
- We can represent switching connections using
 - Removing a connection
 - Adding a new connection



Hashemian, M., Stanley, K., and Osgood, N. 2010. Flunet: Automated tracking of contacts during flu season. Proceedings of the 6th International workshop on Wireless Network Measurements (WiNMee 2010), 557-562, 6pp.

Automatically Wired Connections

- Predefined built-in (i.e. non-user-defined) AnyLogic network types can take care of “wiring in” a new node into an existing network
 - Just call *environment.applyNetwork()* to get the environment to “recalculate” the network – and thereby include the new node.

AnyLogic methods for Adding & Deleting Connections

- *agentA.connectTo(agentB)*
 - Connects *agentA* to *agentB*
 - NB: Connections are assumed to be undirected and symmetric (i.e. if *agentA* is considered to be connected to *agentB*, then *agentB* is considered to be connected to *agentA*)
- *agentA.disconnectFrom(agentB)*
 - Disconnects *agentA* and *agentB* from each other
- For more details and additional methods, see the slides for the *Networks* lecture

Useful Methods for Dealing with Networks

- *agentA.getConnectionsNumber()*
 - Gets count of connections associated with agentA
- *agentA.isConnectedTo(agentB)*
 - Return true if agentA and agentB are connected; false otherwise
- *agent.getConnectionedAgent(int index)*
 - Returns the *index*th agent connected to agentA. Note: The first person is at index 0 (not index 1!)
- *agent.getConnections()*
 - Returns list (LinkedList<Agent>) of all connections of Agent *agent*. Can loop over this with e.g. a *for* loop