

Axelrod's dissemination of culture model in Repast J

1. Requirements


The model was developed and tested in the following environment.

- Windows XP PC
- J2SE Development Kit 5.0 Update 10
- J2SE Runtime Environment 5.0 Update 10
- Repast J version 3.1

Running the model in other environments: e.g. with earlier versions of the Java Runtime Environment (JRE), may require recompilation of the Java model classes. The Java sources are provided in file `AxelrodModelSources.zip` for this purpose. Note that the Java code uses the generics feature introduced in J2SE 5.0, but it would be easy to adapt the code for earlier versions of Java by simply deleting code of the form `<Type>`.

2. Installing and loading the model

Copy file `AxelrodModel.jar` to your Repast J **models** folder (usually `C:\Program Files\Repast 3\Repast J\models`).

1. Start the Repast J GUI.
2. Click on the first button in the toolbar – the **Load Model** button .
3. Expand the **Models** folder in the **Load Model** dialogue.
4. You should see the model name: **Axelrod model**. (If not, check that `AxelrodModel.jar` is in your Repast J **models** folder. This problem may also be caused by using an incompatible JRE.)
5. Click on the model name and click the **Load** button.

3 Axelrod's model

This Repast model implements the modelling of the dissemination of culture, as developed by Robert Axelrod. It covers the simple model described in Axelrod (1997), plus several of the model extensions proposed in that article and in Axelrod's "Complexity of Cooperation Web Site". For more information on the model, its extensions, original Pascal code and online access to his paper, see Axelrod's website at <http://www-personal.umich.edu/~axe/>.

3.1 Parameters

The parameters of this model are as follows (as displayed in Repast's **Settings window** in logical order). The default values (in brackets) give the simple model described in Axelrod (1997, pp.207-209).

GridWidth: (10) The width of the lattice – the number of agents horizontally.

GridHeight: (10) The height of the lattice – the number of agents vertically.

Torus: (false) Specifies whether the lattice has boundaries (false), or wraps around the northern and southern edges, and the eastern and western edges (true). See Axelrod (1997, p.215).

NeighbourhoodType: (Von Neumann) Specifies the type of neighbourhood of possible interacting agents around the chosen active agent. All sites in the neighbourhood have equal probability of being selected. There are three options: Global uniform, Moore and Von Neumann. If Global uniform, the neighbourhood consists of all other agents (Axelrod's "long distance interaction – complete mixing"). If Moore, the neighbourhood has a square shape centred on the active agent. If Von Neumann, the neighbourhood extends equally in the North, South, West, East directions from the active agent.

NeighbourhoodExtent: (1) The size of the Moore or Von Neumann neighbourhood – its "radius" – the number of agents from the active agent to the furthest edge of the neighbourhood.

FeatureCount: (5) The number of cultural features possessed by each agent.

TraitCount: (10) The number of traits in each feature.

MutationProbability: (0.0) The probability that, at each tick of the simulation, a randomly chosen agent has one of its trait values randomly changed. (This is independent of, and in addition to, any other interactions that may have taken place at the same tick.) This relates to a model extension suggested by Axelrod that simulates "cultural drift", similar to spontaneous mutation in genetics.

BroadcastProbability: (0.0) The probability that, at each tick of the simulation, a randomly chosen agent is made to interact with the **Broadcast culture** agent in the same way that it would normally interact with a neighbouring agent. (This is independent of, and in addition to, any other interactions that may have taken place at the same tick.) This relates to a model extension suggested by Axelrod that simulates the effects of advertising, education, mass communication, printed books, radio, television etc. The **Broadcast culture** currently consists of all zero trait values.

DisplayInterval: (1000) Number of ticks between updates of Repast's **Display window**. Having large intervals allows the simulation to run more quickly. Change this to 1 if you want to see the display change at every step of the simulation.

OutputInterval: (1000) Number of ticks between output of data to Repast's **Output window**, and possibly also to files and charts. The data require computing the number of regions and zones. Having large intervals allows the simulation to run more quickly.

LoadGui: (true) Is set to false when running simulations in **Multi-Run** mode.

3.2 Agent properties and representation

In this model agents represent static cultural sites arranged in a rectangular lattice.

An agent's culture comprises a set of traits represented by a vector of **FeatureCount** integers each in the range [0, **TraitCount** – 1].

The graphical representation of an agent in the **Display window** is a coloured rectangle. The colour is determined by mapping the agent's set of trait values to the sRGB colour space.

3.3 Model dynamics

Ignoring mutation and broadcast events, the core dynamics of the model are as follows.

An active agent **A** is chosen at random.

An agent **N** is chosen at random from the neighbourhood of **A**.

With probability equal to the proportion of features that are the same in **A** and **N**, an interaction occurs between **A** and **N**.

Select at random a feature of **A** and **N** that differ (if there is one).

Change the value of the selected feature of **A** to that of **N**.

The simulation stops automatically when the number of regions equals the number of zones.

3.4 Output data

Region and zone counts are calculated at intervals of **OutputInterval** ticks and written to the **Output window**.

The final number of regions and zones are written to a file named **Axelrod.txt** (or **Axelrod1.txt** or **Axelrod2.txt** etc.), in folder **C:\Program Files\Repast 3\Repast J\models**. If **LoadGui** is true, then each agent's final trait values are also written to file – a separate line for each agent in column-wise order of grid coordinates: (0,0), (0,1)...(1,0), (1,1)...etc.

3.5 Charts

A time series chart of numbers of regions and zones plotted against ticks is created automatically.

Reference

Axelrod, R. (1997). The dissemination of culture: A model with local convergence and global polarization. *Journal of Conflict Resolution*, Vol. 41, 203 – 226.