
Introduction to agent-based modelling for social scientists

Dr. J. Kasmire
UK Data Service



16 January 2020

UK Data Service



ABM for social scientists – webinar series

- **ABM: An Intro**
 - – **Jan 16, 2020**
- **ABM: Adding Data**
 - – Jan 30, 2020
- **ABM: Experiments and Output**
 - – Feb 13, 2020



Table of Contents

- Top-down vs. Bottom-up for problems and problem-solving
- Agents and agent-based models
- Example – Telephone Game
- Pros and cons of ABM, platforms/ languages, resources, etc.
- Q&A



- **Top-down**

- A problem can be understood as “top-down” if it occurs within a system that is ‘whole’, ‘well-understood’ and with central control or structure.
- The ‘cause’ can be isolated to constituent parts, defined interactions, or specific problem areas.
- Top-down problems suit classic scientific method, which focusses on isolation, prediction, repetition, etc.



- Top-down examples

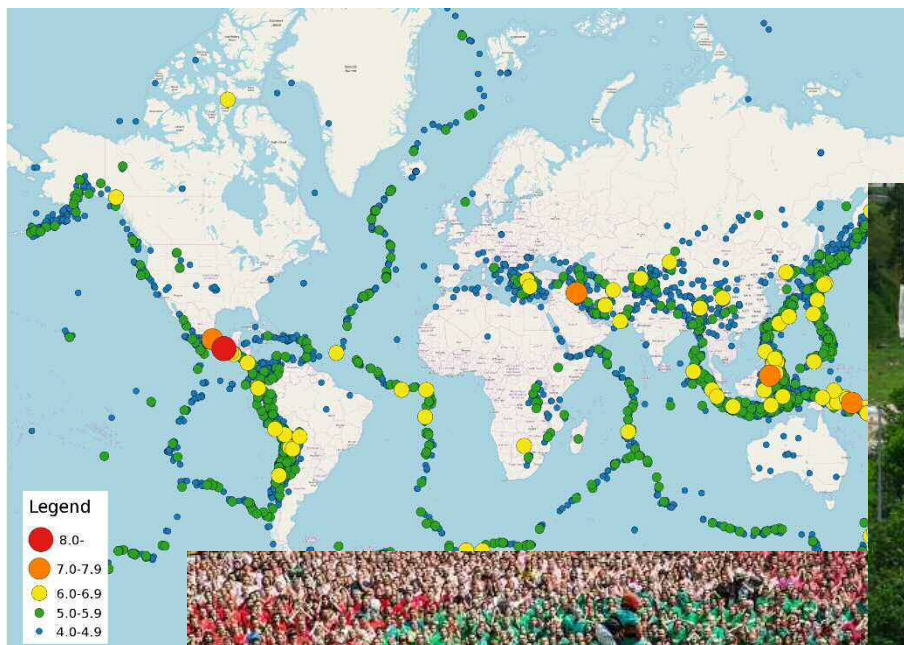


- **Bottom-up**

- A problem is “bottom-up” if it happens within a poorly-understood, open system with vague parts and interactions and with partial or decentralised control.
- Not reasonable to assume reducibility, linearity, time/context-independence, or closed/simplicity.
- Understand the problem by recreating it though:
 - Deterministic low-level rules,
 - Finite parameters,
 - Varying (including extreme or counter-factual) conditions.
- Not easy to apply classic scientific method.



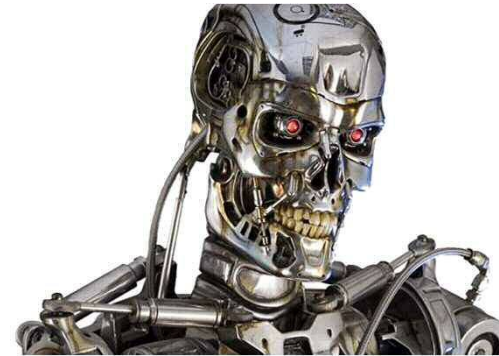
- Bottom-up examples



- Agents – Three common views

- Artificial Intelligence –

- Autonomous
- Individual
- Learners and/or problem solvers



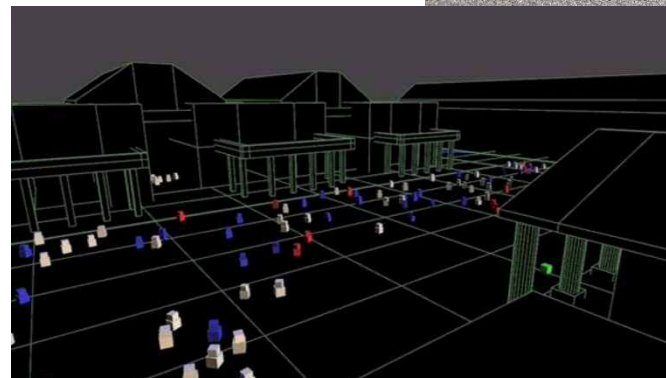
- Multi-agent systems –

- Distributed control
- Specific problems



- Agent-based models –

- Multiple
- Interacting
- Rule-based



- Agent-based models have:

- A simulated world of varying richness, including time
- Objects and/or agents that have:
 - States
 - Rules
- Example – a game of ‘Telephone’



- **Agent-based models: The World**

- Representing most anything (stock market, city, room, etc.)
- Proceeds through time in discrete ‘time steps’
- Has representations of potentially interesting factors (current selling price, weather, capacity, etc.)
- Is unique for each agent as it contains everything *else* (including the other agents)
- Has states composed of their features and contents, which may affect/are affected by agents, objects, itself, random factors, etc. according to the rules



• The World - Telephone Game

Telephone - NetLogo (C:\Users\mzyssjkc\Dropbox (The University of Manchester)\CMI\Slides\ABM)

File Edit Tools Zoom Tabs Help

Interface Info Code

Edit Delete Add

normal speed view updates
ticks: 0 on ticks Settings...

setup go once go

Current noise level: 27 Total prize count: 0

Prize distribution

Prize count: 95 0
Players: 0 95

layout-style: random Number_Players: 95
choose-starter: randomly game-length: 4
choose-next: randomly apply-noise-distortion?: Off
Acceptable-noise-level: 70
Garrulous?: Off

Command Center
Initial set up is now complete



- **Agent-based models: The Agents**

- Representing most anything (people, cars, ocean waves, etc.)
- Have representations of potentially interesting factors (money, age, colour, speed, etc.) that can be dynamic
- Are unique and behave uniquely
- Have states that affect/are affected by their unique world according to the rules
- Make decisions based on world, states, rules, other agents, etc.



• The Agents – Telephone Game

The screenshot shows the NetLogo 'Telephone' game interface. The main window displays a black field with many small human figures (agents) scattered across it. A 'turtle 4' window is open, showing a red human figure and its properties. The interface includes a menu bar (File, Edit, Tools, Zoom, Tabs, Help), a toolbar with buttons for Edit, Delete, Add, and a slider for speed (set to 'faster', ticks: 560). There are also checkboxes for 'view updates' and 'on ticks', and a 'Settings...' button. On the left, there are buttons for 'setup', 'go once', and 'go'. Below these are two monitors: 'Current noise level' (293) and 'Total prize count' (796). A 'Prize distribution' histogram is also visible, showing a distribution of prize counts for 95 players. On the right, there are several monitors: 'Number_Players' (95), 'game-length' (4), a checkbox for 'On apply-noise-distortion?' (OFF), and 'Acceptable-noise-level' (70). The 'Properties' window for 'turtle 4' lists various attributes: who (4), color (13), heading (292), xcor (29), ycor (11), shape ('person'), label (''), label-color (9.9), breed (turtles), hidden? (false), size (3), pen-size (1), pen-mode ('up'), prize-count (13), temp-word (["2" "2" "2"]), current-word (["2" "2" "2"]), and currently-playing (0). The Command Center at the bottom shows the message: 'Player 3 can hear easily at 46 decibels.'



- **Agent-based models: States**

- Sum of current features
- Can be dynamic or static
- Can be inputs, outputs, etc.
- Used to make decisions



• States – Telephone Game

* Telephone - NetLogo (C:\Users\mzyssjkc\Dropbox (The University of Manchester)\CMI\Slides\ABM)

File Edit Tools Zoom Tabs Help

Interface Info Code

Edit Delete Add abc Button faster ticks: 560 view updates on ticks Settings...

setup go once go

Current noise level: 293 Total prize count: 796

Prize distribution

Prize count: 95 0 0 Players: 95

Command Center

Player 3 can hear easily at 46 decibels.

turtle 4 View

Watch

Properties

| | |
|-------------------|---------------|
| who | 4 |
| color | 13 |
| heading | 292 |
| xcor | 29 |
| ycor | 11 |
| shape | "person" |
| label | "" |
| label-color | 9.9 |
| breed | turtles |
| hidden? | false |
| size | 3 |
| pen-size | 1 |
| pen-mode | "up" |
| prize-count | 13 |
| temp-word | ["2" "2" "2"] |
| current-word | ["2" "2" "2"] |
| currently-playing | 0 |

Number_Players: 95

game-length: 4

On apply-noise-distortion? OFF

Acceptable-noise-level: 70



• Agent-based models: Rules

- Govern how states change or decisions are made
- Use model generated info, modeller choice, randomness
- Can be simple or complex (if-then, while, for each, etc.)

```
to go
  set noise-level noise-level + random 10 - random 10
  if noise-level < 0 [set noise-level 0]
  ifelse length current-turtle = 0
  [start-new-game]
  [ifelse length current-turtle = game-length
  [wait .5
  end-game]
  [play-game]]
  tick
end

to start-new-game
  ask turtles [set currently-playing 0]
  if choose-starter = "randomly" [ask-random]
  if choose-starter = "most prizes" [ask-most]
  if choose-starter = "least prizes" [ask-least]
end

to ask-random
  ask one-of turtles with [currently-playing = 0 ]
  [ if Garrulous? [print "Random player asked to play..."]
  participate]
end

to ask-most
  ifelse mean [prize-count] of turtles != 0 and any? turtles with [currently-playing = 0 and prize-count > mean [prize-count] of turtles]
  [ask one-of turtles with [currently-playing = 0 and prize-count > mean [prize-count] of turtles] [participate]]
  [ask-random]
end
```



• Telephone Game – The interface tab

The screenshot shows the NetLogo interface for the Telephone Game simulation. The title bar reads "Telephone - NetLogo (C:\Users\mzyssjk\Dropbox (The University of Manchester)\CMI\Slides\ABM)". The menu bar includes "File", "Edit", "Tools", "Zoom", "Tabs", and "Help". The interface is divided into several sections:

- Control Panel:** Includes buttons for "Edit", "Delete", "Add", and a "Button" dropdown. A "normal speed" slider is set to "ticks". A checkbox for "view updates on ticks" is checked. A "Settings..." button is also present.
- Game State:** Three buttons: "setup", "go once", and "go".
- Statistics:** Two monitors: "Current noise level" (0) and "Total prize count" (0).
- Prize Distribution:** A graph titled "Prize distribution" with "Prize count" on the y-axis (0 to 150) and "Players" on the x-axis (0 to 10).
- Parameters:** A list of sliders and monitors: "layout-style" (random), "Number_players" (158), "choose-starter" (most prizes), "game-length" (6), "choose-next" (most prizes), "apply-noise-distortion?" (On/Off), "Acceptable-noise-level" (70), and "Garrulous?" (On/Off).
- Command Center:** A purple bar at the bottom of the interface.



• Telephone Game – The info tab

Telephone - NetLogo {C:\Users\mzyssjk\Dropbox (The University of Manchester)\CM1\Slides\ABM}

File Edit Tools Zoom Tabs Help

Interface Info Code

Find... Edit

WHAT IS IT?

This is a model of a game of Telephone (also known as Chinese Whispers in the UK), with agents representing people that can be asked, to play. The first player selects a word from their internal vocabulary and "whispers" it to the next player, who may mishear it depending on the current noise level, who whispers that word to the next player, and so on.

When the game ends, the word chosen by the first player is compared to the word heard by the last player. If they match exactly, all players earn large prize. If the words do not match exactly, a small prize is awarded to all players for each part of the words that do match. Players change color to reflect their current prize-count. A histogram shows the distribution of colors over all the players.

The user can decide on factors like

- * how many players there are,
- * whether they are laid out in a circle or just randomly,
- * how many players participate in a game,
- * whether to apply noise-distortion or not,
- * at what decibel level noise distortion starts interfering with the game,
- * how the first player to participate is chosen,
- * how further players are chosen, and
- * whether or not the games run quickly and silently or slowly and with commentary to explain what is happening.

These factors influence how likely players are to win a game and thus how the color of players will be distributed over time.

HOW IT WORKS

The world has dimensions and also a noise level that moves up and down randomly at each time step, but cannot fall below 0. When the model is initiated, a number of characters are laid out across the dimensions according to a modeller input, which appears in the interface as "layout-style" and gives the options of random or circle. When created, all agents have a vocabulary (set in the code) of several 2 character "words" and an alphabet consisting of all the characters that appear in any position of any of the words in their vocabulary.

The first player is chosen according to a modeller input, which appears on the interface as "choose-starter" which gives the options of randomly, most prizes and least prizes (self-explanatory). That agent randomly selects one of the words in their vocabulary.

The next player is chosen according to a modeller input, which appears on the interface as "choose-next" which gives the options of randomly, most prizes, least prizes, nearest and nearest (self-explanatory?). The first player "whispers" their chosen word to the next player, who will hear it correctly if the noise-level is below the "Appropriate-noise-level" as set by the modeller. If the noise level is above "Appropriate-noise-level" then a small test is performed for each part of the word, with a chance that the listener may mishear some but not all of the sounds. The listener then becomes the whisperer and the process is repeated until the number of players reaches the "game-length" as set by the modeller. At that point, the game ends, the word



• Telephone Game – The code tab

```
Telephone - NetLogo (C:\Users\mzyssjkc\Dropbox (The University of Manchester)\CMI\Slides\ABM)
File Edit Tools Zoom Tabs Help
Interface Info Code
Find... Check | Procedures v |  Indent automatically

globals [ noise-level
  current-turtle
  vocabulary
  alphabet
]

turtles-own [
  prize-count
  temp-word
  current-word
  currently-playing
]

to setup
  clear-all
  ifelse Garrulous?
    [resize-world -100 100 -50 50]
    [resize-world -100 100 -100 100]
  set current-turtle []
  set alphabet ["a" "b" "c" "c" "d" "e" "f" "g" "o" "1" "2" "3" "4" "5"]
  set vocabulary [{"a" "a" "a"} [{"b" "b" "b"} [{"c" "c" "c"} [{"1" "1" "1"} [{"2" "2" "2"} [{"d" "3" "d"} [{"f" "o" "o"} [{"g" "e" "e"} [{"g" "4" "5"} [{"1" "o" "1"}]]]]]]]]
  set-default-shape turtles "person"
  ask patches with [ pxcor < -11 and pxcor > 11 ] [ set pcolor white]
  create-turtles Number_Players [
    set size 3 ;; be easier to see
    set color 2
    __set-line-thickness 2
    set currently-playing 0
    set prize-count 0
    set current-word [""]
  ]
  establish-layout
  set noise-level random 100
  reset-ticks
  if Garrulous? [print "Initial set up is now complete."]
end

to establish-layout
  ifelse layout-style = "random" [ask turtles [move-to one-of patches]]
    [layout-circle turtles 50]
  if Garrulous? [print (word "Layout set to " layout-style ".")]
end

to go
  set noise-level noise-level + random 10 - random 10
  if noise-level < 0 [set noise-level 0]
  ifelse length current-turtle = 0
  [start-new-game]
  [ifelse length current-turtle = game-length
    [wait .5
     end-game]
    [play-game]]
  tick
end

to start-new-game
  ask turtles [set currently-playing 0]
```



• Telephone Game – Modeller choices

The screenshot shows a NetLogo simulation window for a Telephone Game model. The interface includes a menu bar (File, Edit, Tools, Zoom, Tabs, Help), a toolbar with buttons for Edit, Delete, Add, and Tab Button, and a status bar showing 'normal speed' and 'ticks: 305'. The main area is divided into several sections:

- Control Panel:** Buttons for 'setup', 'go once', and 'go'. Below these are two monitors: 'Current noise level' (value: 15) and 'Total prize count' (value: 1172).
- Prize Distribution Graph:** A histogram titled 'Prize distribution' with 'Prize count' on the y-axis (0 to 140) and 'Players' on the x-axis (0 to 140). The graph shows a single bar at prize count 0 with a height of approximately 140.
- Simulation Area:** A large black square containing many small, multi-colored human figures representing players.
- Parameter Panel:** A vertical list of sliders and monitors on the right side:
 - 'layout-style' set to 'random'
 - 'choose-starter' set to 'randomly'
 - 'choose-next' set to 'randomly'
 - 'Number_Players' set to 300
 - 'game-length' set to 10
 - 'apply-noise-distortion?' set to 'Off'
 - 'Acceptable-noise-level' set to 75
 - 'Garrulous?' set to 'Off'
- Command Center:** A large empty text area at the bottom with a 'Clear' button and a 'command center' label.
- Observer Console:** A small text area at the bottom left containing the text 'observer >'.



• Telephone Game – Sliders

File Edit Tools Zoom Tabs Help

Interface Info Code

Edit Delete Add Tab Button normal speed view updates ticks: 305 on ticks Settings...

setup go once go

Current noise level: 15 Total prize count: 1172

Prize distribution

Prize count: 0 to 140

Players: 0 to 140

layout-style layout-circle

choose-starter randomly

choose-next randomly

Number_Players: 100

game-length: 10

apply-noise-distortion? Off

Acceptable-noise-level: 75

Garrulous? Off

Command Center Clear

observer>



• Telephone Game – Setup and Go

The screenshot shows a NetLogo interface for a Telephone Game simulation. The interface includes a menu bar (File, Edit, Tools, Zoom, Tabs, Help), a toolbar with buttons for Edit, Delete, Add, and Tab Button, and a speed control set to 'normal speed'. The main window is divided into several sections:

- Control Panel:** Contains buttons for 'setup', 'go once', and 'go'. Below these are two monitors: 'Current noise level' (value: 37) and 'Total prize count' (value: 0).
- Prize Distribution:** A graph with 'Prize count' on the y-axis (0 to 95) and 'Players' on the x-axis (0 to 95). The graph is currently empty.
- Game Area:** A large black square containing many small grey human figures representing players.
- Settings Panel:** Located on the right, it includes dropdown menus for 'layout-style' (set to 'layout-circle'), 'choose-starter' (set to 'randomly'), and 'choose-next' (set to 'randomly'). It also features sliders for 'Number_Players' (95) and 'game-length' (4), a checkbox for 'apply-noise-distortion?' (set to 'Off'), a text input for 'Acceptable-noise-level' (70), and a checkbox for 'Garrulous?' (set to 'Off').
- Command Center:** A large empty area at the bottom for entering commands, with a 'Clear' button and a prompt 'observer>'.



• Telephone Game – Action

The screenshot shows a NetLogo window titled "Telephone Game". The interface includes a menu bar (File, Edit, Tools, Zoom, Tabs, Help) and a toolbar with buttons for Edit, Delete, Add, and a Tab Button. A speed slider is set to "normal speed" and "ticks: 5". A checkbox for "view updates" is checked and set to "on ticks". A "Settings..." button is also present.

The main display area shows a circular arrangement of 95 players, represented by small grey figures. The control panel on the right includes:

- layout-style: layout-circle
- choose-starter: randomly
- choose-next: randomly
- Number_Players: 95
- game-length: 4
- On apply-noise-distortion?: Off
- Acceptable-noise-level: 70
- On Garrulous?: Off

On the left, there are two monitors: "Current noise level" (61) and "Total prize count" (20). Below them is a "Prize distribution" graph with "Prize count" on the y-axis (0 to 95) and "Players" on the x-axis (0 to 95). The graph shows a single bar at player 0 with a height of 95.

The Command Center at the bottom displays the following text:

```
Layout set to layout-circle.  
Initial set up is now complete.  
Random player asked to play...  
Player 0 has decided to whisper the word [b b b].  
Random player asked to play...  
Player 1 is listening...  
Player 1 can hear easily at 57 decibels.  
Random player asked to play...  
Player 2 is listening...  
Player 2 can hear easily at 57 decibels.  
Random player asked to play...  
Player 3 is listening...  
Player 3 can hear easily at 57 decibels.  
The players this round were [77 13 4 93].  
Everyone gets a big prize! Player 0 whispered [b b b] and Player 4 heard [b b b].  
Layout set to layout-circle.  
observer>
```



• Telephone Game – Go forever

The screenshot shows a NetLogo interface for a Telephone Game simulation. The main window is titled "Interface Info Code" and contains several control elements:

- Buttons: "setup", "go once", "go".
- Current noise level: 65.
- Total prize count: 24.
- Prize distribution graph: A line graph with "Prize count" on the y-axis (0 to 95) and "Players" on the x-axis (0 to 95). The graph shows a single point at (95, 0).
- Central visualization: A black circle with small grey figures representing players arranged around its perimeter.
- Control panel (right):
 - layout-style: layout-circle
 - choose-starter: randomly
 - choose-next: randomly
 - On apply-noise-distortion?: Off
 - Acceptable-noise-level: 10
 - On Garrulous?: Off
- Command Center (bottom):

```
observer>
Ah, it is quite noisy at 64 decibels. Player 1 has misheard the third phoneme.
Player 1 has heard[2 2 5].
Random player asked to play...
Player 2 is listening...
Player 2 is listening closely, even at 60 decibels and has heard the first phoneme correctly.
Ah, it is quite noisy at 60 decibels. Player 2 has misheard the second phoneme.
Ah, it is quite noisy at 60 decibels. Player 2 has misheard the third phoneme.
Player 2 has heard[2 3 b].
Random player asked to play...
Player 3 is listening...
Ah, it is quite noisy at 58 decibels. Player 3 has misheard the first phoneme.
Player 3 is listening closely, even at 58 decibels and has heard the second phoneme correctly.
Player 3 is listening closely, even at 58 decibels and has heard the third phoneme correctly.
Player 3 has heard[f 3 b].
The players this round were [20 39 84 72].
It was just too noisy. Player 0 whispered [2 2 2] but Player 4 heard [f 3 b]. Still, small prizes will be awarded if any of the phonemes match.
Layout set to layout-circle.
```



• Telephone Game – After 1500 time steps

Telephone - NetLogo (C:\Users\mzyssjck\Dropbox (The University of Manchester)\CMI\Slides\ABM)

File Edit Tools Zoom Tabs Help

Interface Info Code

Edit Delete Add Take Button faster view updates on ticks Settings...

ticks: 1534

setup go once go

Current noise level: 0 Total prize count: 1888

Prize distribution

Prize count: 95 0 0 Players: 95

layout-style: layout-circle Number_Players: 95

choose-starter: randomly game-length: 4

choose-next: randomly apply-noise-distortion? On Off

Acceptable-noise-level: 10

Garrulous? On Off

Command Center

Ah, it is quite noisy at 34 decibels. Player 1 has misheard the third phoneme.
Player 1 has heard[2 2 5].
Random player asked to play...
Player 2 is listening...
Player 2 is listening closely, even at 60 decibels and has heard the first phoneme correctly.
Ah, it is quite noisy at 60 decibels. Player 2 has misheard the second phoneme.
Ah, it is quite noisy at 60 decibels. Player 2 has misheard the third phoneme.
Player 2 has heard[2 3 b].
Random player asked to play...
Player 3 is listening...
Ah, it is quite noisy at 58 decibels. Player 3 has misheard the first phoneme.
Player 3 is listening closely, even at 58 decibels and has heard the second phoneme correctly.
Player 3 is listening closely, even at 58 decibels and has heard the third phoneme correctly.
Player 3 has heard[f 3 b].
The players this round were [20 39 84 72].
It was just too noisy. Player 0 whispered [2 2 2] but Player 4 heard [f 3 b]. Still, small prizes will be awarded if any of the phonemes match.
Layout set to layout-circle.



• Telephone Game – After 4000 time steps

Telephone - NetLogo [C:\Users\mzyssjck\Dropbox (The University of Manchester)\CMI\Slides\ABM]

File Edit Tools Zoom Tabs Help

Interface Info Code

Edit Delete Add view updates on ticks

ticks: 4004

Current noise level: 205 Total prize count: 4704

Prize distribution

Prize count: 95 0 0 Players: 0 95

layout-style: layout-circle

layout-circle

Number_Players: 95

choose-starter: randomly

game-length: 4

choose-next: randomly

On apply-noise-distortion? Off

Acceptable-noise-level: 10

On Off Garrulous?

Command Center

Ah, it is quite noisy at 34 decibels. Player 1 has misheard the third phoneme.
Player 1 has heard[2 2 5].
Random player asked to play...
Player 2 is listening...
Player 2 is listening closely, even at 60 decibels and has heard the first phoneme correctly.
Ah, it is quite noisy at 60 decibels. Player 2 has misheard the second phoneme.
Ah, it is quite noisy at 60 decibels. Player 2 has misheard the third phoneme.
Player 2 has heard[2 3 b].
Random player asked to play...
Player 3 is listening...
Ah, it is quite noisy at 58 decibels. Player 3 has misheard the first phoneme.
Player 3 is listening closely, even at 58 decibels and has heard the second phoneme correctly.
Player 3 is listening closely, even at 58 decibels and has heard the third phoneme correctly.
Player 3 has heard[f 3 b].
The players this round were [20 39 84 72].
It was just too noisy. Player 0 whispered [2 2 2] but Player 4 heard [f 3 b]. Still, small prizes will be awarded if any of the phonemes match.
Layout set to layout-circle.



- Ok... But why an ABM?

- Some features are not intuitive.
- Try to imagine the histogram after changing:
 - Choose-starter?
 - Choose-next?
 - Number of players?
 - Layout-style?
 - Vocabulary size?
 - Noise-distortion and/or Acceptable-noise-level?
 - Other possibilities?



- **Pros of ABM**

- No need to rely on intuition for complex behaviours
- Formalise mental models so can be inspected by others
- Force abstract concepts to be represented concretely
- Relatively cheap and easy way to test the 'untestable'
- Potentially very fast
- Can show range of possible futures
- Can run using real data, training data, random data, etc.
- Can be run again and again (with or without changes)
- Can be very intuitive for non-specialists to understand



- **Cons of ABM**

- Mental models (and ABM based on them) are hard to be critical about
- Concrete representations may not adequately capture abstract concepts
- Speed lends itself to being used for ‘prediction’
- Possible futures are hard to turn into risk profiles
- Can give false sense of security and truthiness
- Computational methods may motivate distrust
- Easy to misunderstand



-
- Platforms, languages, etc.
 - Free, open-source, relatively small download size, tutorials and info available on web
 - Mason – focus on easy to learn, not most well recognised
 - Repast – more powerful, steeper learning curve
 - NetLogo – built in visualisation, not most powerful
 - EMLab-Agentspring – modular, not most well recognised
 - Object-oriented software (e.g. Python) – extremely powerful, not targeted so harder to learn/find answers
 - https://en.wikipedia.org/wiki/Comparison_of_agent-based_modeling_software
 - https://en.wikipedia.org/wiki/Agent-based_model
 - <https://www.comses.net/codebases/c32072dd-cd7c-4547-beba-e93a3f508c5f/releases/1.0.0/>



Summary

- Bottom-up: non-intuitive large-scale effects of many small acts
- Agents: heterogeneous actors that observe, decide, act
- ABM: bottom-up simulation with easy testing
- Pros: can be inspected, cheap, easy, fast, can do 'impossible' experiments
- Cons: inevitably simplify complex concepts, often misused for prediction, easy to misunderstand



Questions

Dr. J. Kasmire

julia.kasmire@manchester.ac.uk

 @JKasmireComplex

ukdataservice.ac.uk/help/

Subscribe to UK Data Service news at

<https://www.jiscmail.ac.uk>

 @UKDataService

