

Phase 2: Links with Behavior

Eric Araújo, and Jonathan Hill

Keywords agent-based modeling, social sciences, computational methods

1. LEARNING OBJECTIVES

By the end of Phase 2, students should be able to:

- Model agent interaction through links.
 - Implement influence dynamics.
 - Make links dynamic (removing them when conditions change).
 - Observe and interpret network changes over time.
-

2. CONCEPTUAL FRAME

Now that we can create links, let's use them to model how people influence each other. Links don't just represent connections—they're the channels through which interaction happens. When people are connected, they affect each other. Over time, they become more similar.

3. WHAT HAPPENS IN PHASE 2

We implement the mechanisms of **influence** and **tie decay**:

1. Each time step, connected people influence each other, moving opinions toward similarity.
2. If two people become too different, their link breaks.

No new links form yet (that comes in Phase 3). We're just observing what happens when these two mechanisms operate.

4. KEY INSIGHTS FROM PHASE 2

Question: If people only influence each other to become similar, what happens?

Answer: People in connected clusters converge to similar opinions. BUT, links break between people who are too different. So the network becomes MORE fragmented even though people within groups become more similar.

This demonstrates an important principle: **Local similarity and global fragmentation can coexist.**

5. STARTER CODE FOR PHASE 2

```

breed [people person]
undirected-link-breed [social-bonds social-bond]

people-own [
  opinion
  influenced?
]

social-bonds-own [strength]

to setup
  clear-all

  create-people 20 [
    set opinion random 100
    setxy random-xcor random-ycor
    set size 1
    set influenced? false

    ; Color by opinion: red = extreme, blue = moderate
    set color scale-color red opinion 0 100
  ]

  ask people [
    create-social-bonds-with n-of 3 other people [
      set strength 0.5
      set color grey
      set thickness 0.3
    ]
  ]

  reset-ticks
end

to go
  ask people [ set influenced? false ]

  influence-through-links
  break-weak-ties

  tick
end

to influence-through-links
  ask people [
    let nbors link-neighbors

    if any? neighbors [
      ; Calculate average opinion of nbors
      let avg-neighbor-opinion mean [opinion] of nbors
    ]
  ]

```

```

; Update opinion: 70% your opinion, 30% average neighbor opinion
let new-opinion 0.7 * opinion + 0.3 * avg-neighbor-opinion

; Only update if change is significant
if abs (new-opinion - opinion) > 0.1 [
  set opinion new-opinion
  set influenced? true
]
]

; Update color to reflect new opinion
set color scale-color red opinion 0 100
]
end

to break-weak-ties
; If two linked people differ too much, the link breaks
ask social-bonds [
  let opinion-diff abs ([opinion] of end1 - [opinion] of end2)

  if opinion-diff > 50 [
    die ; Remove the link
  ]
]
end

```

You can find the [full starter code here](#).

6. PHASE 2 MECHANICS EXPLAINED

Now we turn to explaining the two main processes: influence and tie decay.

6.a. The Influence Process

```

to influence-through-links
  ask people [
    let nbors link-neighbors

    if any? nbors [
      let avg-neighbor-opinion mean [opinion] of nbors
      let new-opinion 0.7 * opinion + 0.3 * avg-neighbor-opinion

      if abs (new-opinion - opinion) > 0.1 [
        set opinion new-opinion
      ]
    ]
  ]
end

```

This implements a **weighted average**:

- Your new opinion = 70% of your old opinion + 30% of your nbors' average opinion.
- This means you're influenced but not completely changed.
- The 0.7 and 0.3 can be varied to change how much nbors influence you.

Why weighted average? Because realistic influence is gradual. People don't instantly change their minds; they move incrementally toward others' views.

What if 0.7 is smaller? Nbors have MORE influence, opinions converge faster.

What if 0.7 is larger? You're stubbornness; opinions converge slower.

6.b. *The Tie Decay Process*

```
to break-weak-ties
  ask social-bonds [
    let opinion-diff abs ([opinion] of end1 - [opinion] of end2)

    if opinion-diff > 50 [
      die
    ]
  ]
end
```

This implements a **hard threshold**: if people differ by more than 50 opinion points, the relationship breaks.

Why a threshold? At some point, differences become too fundamental to maintain the relationship. This models the idea that relationships have limits.

What if threshold is lower (e.g., 30)? Links break more easily; faster fragmentation.

What if threshold is higher (e.g., 70)? Links persist longer despite disagreement; slower fragmentation.

7. CLASSROOM ACTIVITIES - PHASE 2

7.a. *Activity 2.1: Observe Influence*

Run the model:

```
setup
repeat 50 [go]
```

Answer:

- “What happened to people’s opinions?”
- “Did everyone end up with the same opinion?”
- “Which links broke and why?”
- “Is the network still fully connected?”

Learning point: Influence makes similar people more similar, but tie decay fragments the network.

7.b. Activity 2.2: Modify Influence Strength

Create a more flexible procedure:

```
to go-with-strength [influence-rate]
  ask people [ set influenced? false ]

  influence-through-links-with-rate influence-rate
  break-weak-ties

  tick
end

to influence-through-links-with-rate [rate]
  ask people [
    let nbors link-neighbors

    if any? nbors [
      let avg-neighbor-opinion mean [opinion] of nbors
      let new-opinion (1 - rate) * opinion + rate * avg-neighbor-opinion

      if abs (new-opinion - opinion) > 0.1 [
        set opinion new-opinion
      ]
    ]

    set color scale-color red opinion 0 100
  ]
end
```

Now run two scenarios:

```
setup
repeat 50 [go-with-strength 0.1] ; Weak influence
```

vs.

```
setup
repeat 50 [go-with-strength 0.5] ; Strong influence
```

Answer:

- “What’s different between weak and strong influence?”
- “Does the final state differ?”
- “Why might this be?”

Learning point: Parameter values matter. Small changes in influence strength produce different outcomes.

The final code for this activity [can be found here](#).

7.c. Activity 2.3: Visualizing Link Dynamics

Add:

```
to visualize-strong-ties
  ask social-bonds [
    let opinion-diff abs ([opinion] of end1 - [opinion] of end2)
    ; Thicker lines for people with similar opinions
    set thickness 0.1 + 0.4 * (1 - opinion-diff / 100)
  ]
end
```

Call this in go:

```
to go
  ask people [ set influenced? false ]
  influence-through-links
  break-weak-ties
  visualize-strong-ties
  tick
end
```

Now run and watch:

```
setup
repeat 100 [go]
```

Answer:

- “What do the thick lines represent?”
- “What do the thin lines represent?”
- “As the model runs, what happens to line thickness?”
- “Can you predict when a link will break?”

Learning point: Visualization helps us see mechanisms. Line thickness reveals relationship strength.

The final code for this activity [can be found here](#).

7.d. Activity 2.4: Count Fragmentation

Add a simple fragmentation counter:

```
to-report estimate-components
  let isolated-people count people with [count link-neighbors = 0]
  report isolated-people
end
```

Answer:

- “How many people have no connections?”
- “Is this number growing or shrinking over time?”
- “What does this tell us about polarization?”

- “What does this tell us about polarization?”

7.e. Real-World Parallel: Echo Chambers

Phase 2 demonstrates the **echo chamber effect** without explicit homophily (new links forming):

- People are influenced by those they interact with (influence)
- Relationships break when views diverge too much (tie decay)
- Result: people end up in clusters of similar opinions

Real-world social media shows similar patterns:

- Algorithms don't require people to preferentially choose similar others
- Just by showing you content similar to what you've seen before (influence-like mechanism), recommendations create clustering
- People who disagree have weaker connections and often stop interacting (tie decay)

7.f. Assessment - Phase 2

Students should be able to:

- Explain why opinions converge within groups but diverge between groups.
 - Predict what happens if influence rate increases or decreases.
 - Identify which links are most likely to break.
 - Write code to count opinions in a certain range: `count people with [opinion > 40 and opinion < 60]`.
-