Interaction and Telemarketers

EES 4760/5760

Agent-Based and Individual-Based Computational Modeling

Jonathan Gilligan

Class #14: Monday, October 07 2024

Getting Started

- Download files for from the download page for “Interaction Models” on the course web site: ees4760.jgilligan.org/downloads/interaction_class_14.

Announcements

- Due date for individual project analysis has been moved to Oct. 16.
- I have rescheduled some classes:
  - In-class discussion of individual project ODDs next Monday (Oct. 14)
  - Scheduling design concept on Wed. Oct. 16
  - Stochasticity design concept on Mon. Oct. 21
  - No class on Wed. Oct. 23.
  - After that, everything returns to the original schedule.

Telemarketer Model

Telemarketer Model

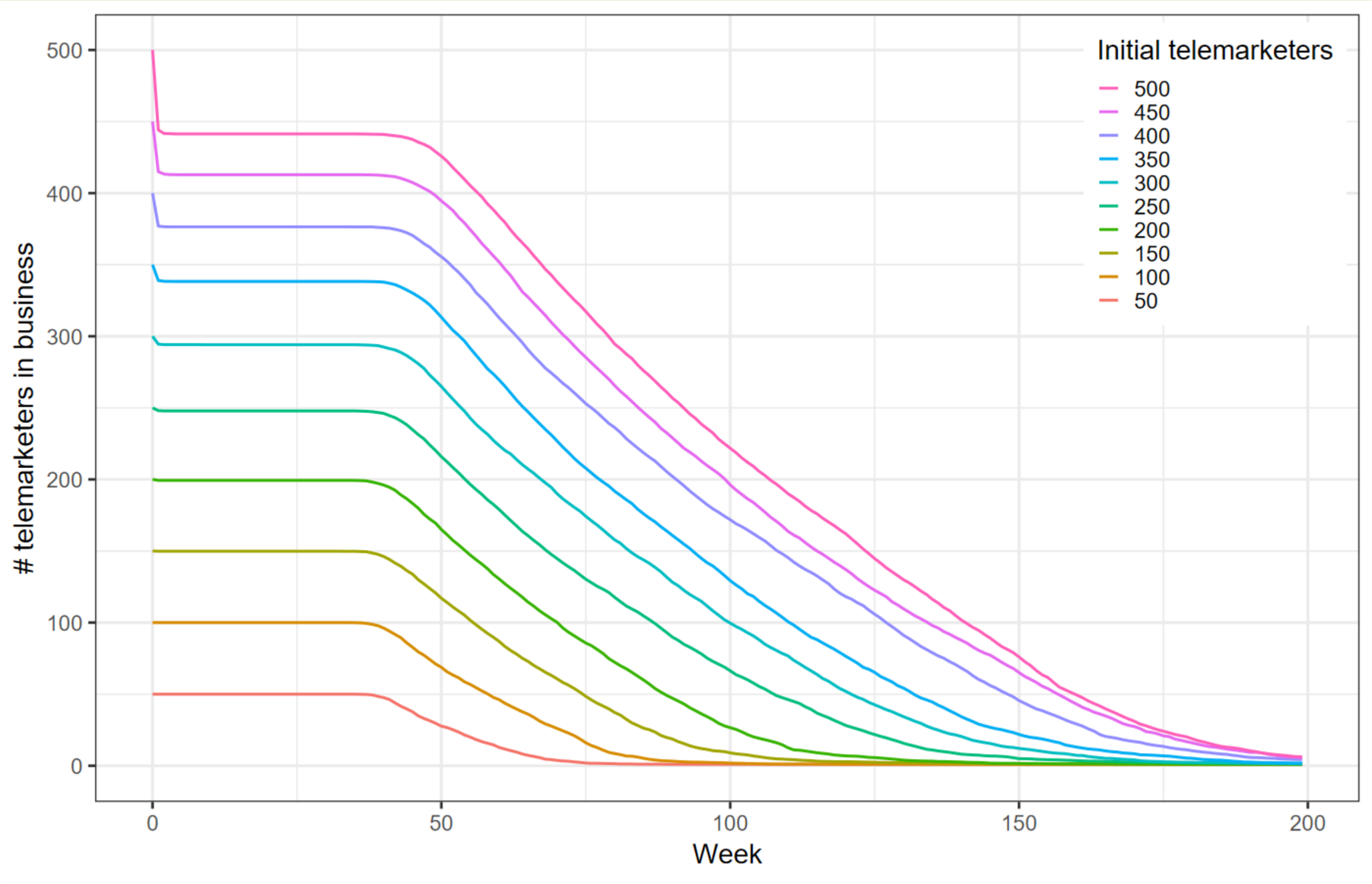
- Telemarketing firms interact
  - Telemarketer calls patches
  - If patch has received a previous call that tick, it hangs up
  - If patch has not received a previous call that tick, it buys something
  - Interaction is indirect, mediated by patches
- Accounting:
  - Net profit = 2 × sales - 50 × size
  - If balance < 0, firm goes bankrupt
- Growth
  - If balance > growth threshold, firm increases size proportional to excess balance

Telemarketer Interactions

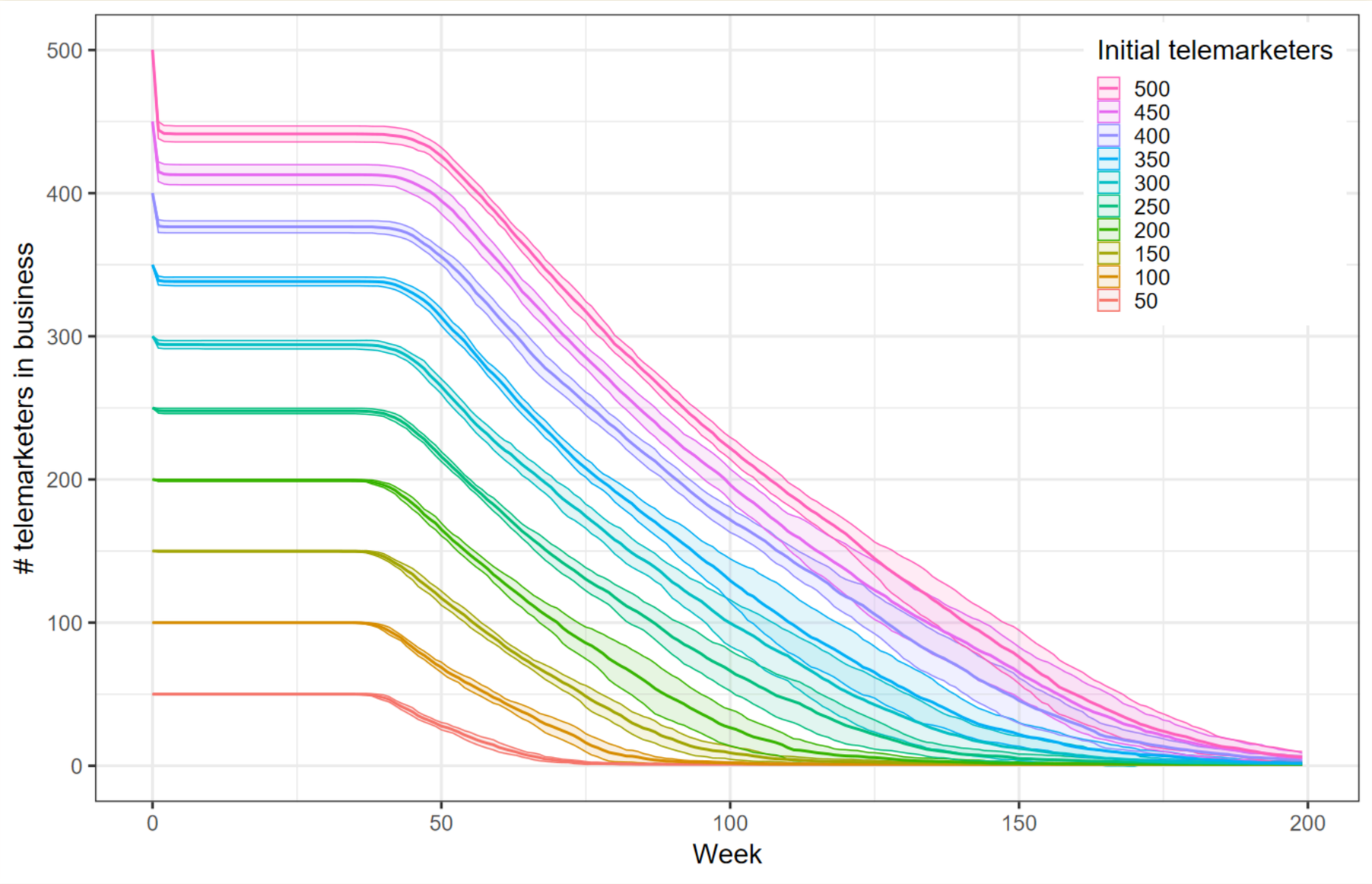
- **Indirect interactions:**
  - Customers (patches) only buy from the first marketer (turtle) to call them that tick.
  - If one marketer calls a customer, that prevents other marketers from selling to that customer.
  - Competition
- **Direct interactions:**
  - When one telemarketing company goes broke, a larger one may buy it.
  - Create persistent owner/subsidiary relationship.
  - Direct transfer of money each tick.

Results

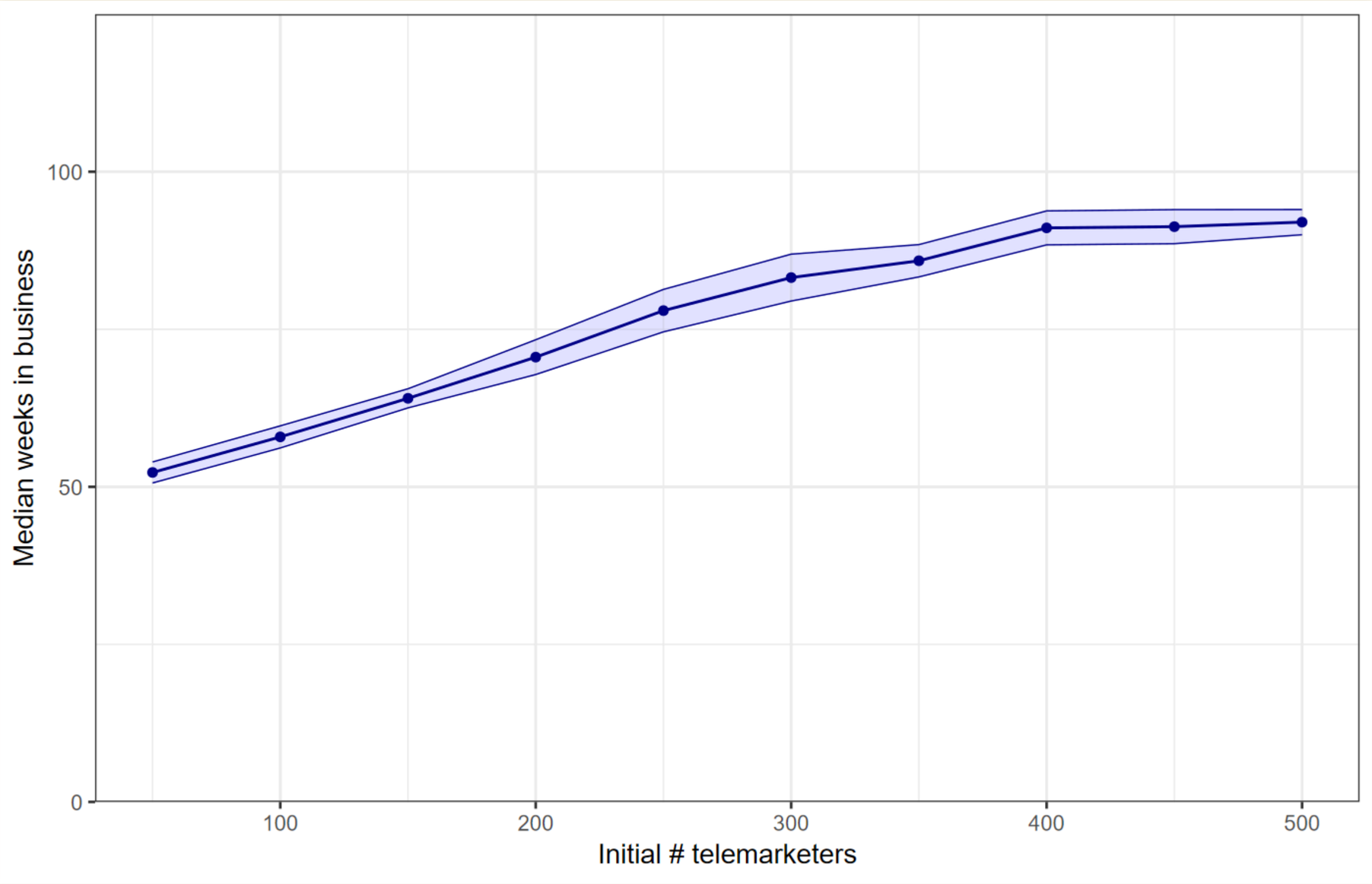
Results


Initial telemarketers
500
450
400
350
300
250
200
150
100
50
telemarketers in business
500
400
300
200
100
0
0
50
100
150
200
Week

Variation


Initial telemarketers
500
450
400
350
300
250
200
150
100
50
telemarketers in business
0
100
200
300
400
500
0
50
100
150
200
Week

Median Weeks in Business



Mergers

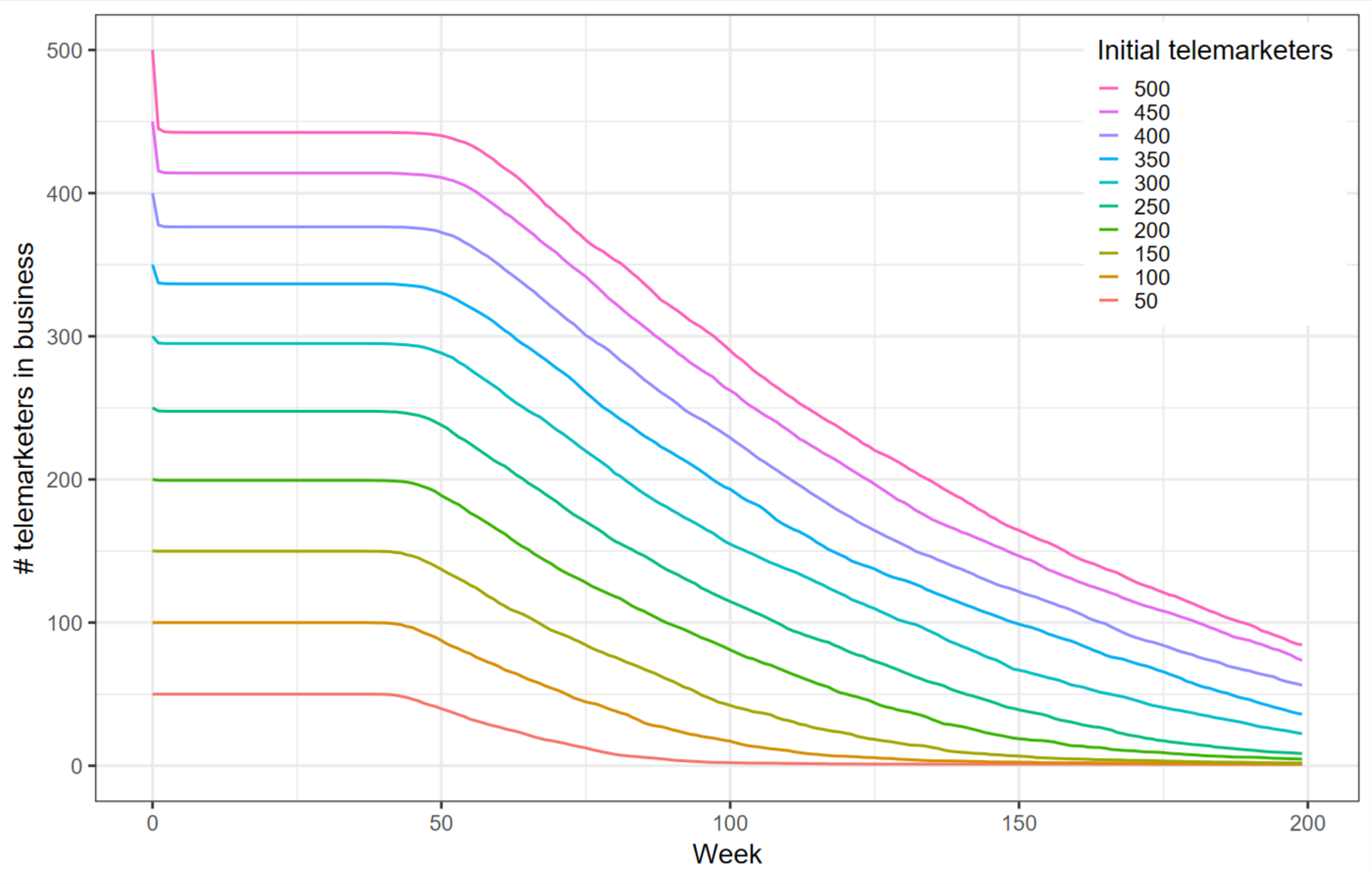
Mergers

- Instead of going bankrupt when the bank balance drops below 0, firms look for acquisition partner
  - Find a company that’s bigger and has enough money to pay off deficit.
  - If it finds a parent, parent pays off deficit
    - child firm ends up with zero balance
  - In future turns, child pays parent 50% of its net profits.
  - In future, if child’s balance becomes negative:
    - If parent has enough money, it pays child’s deficit
    - If parent does not have enough money, child dies.

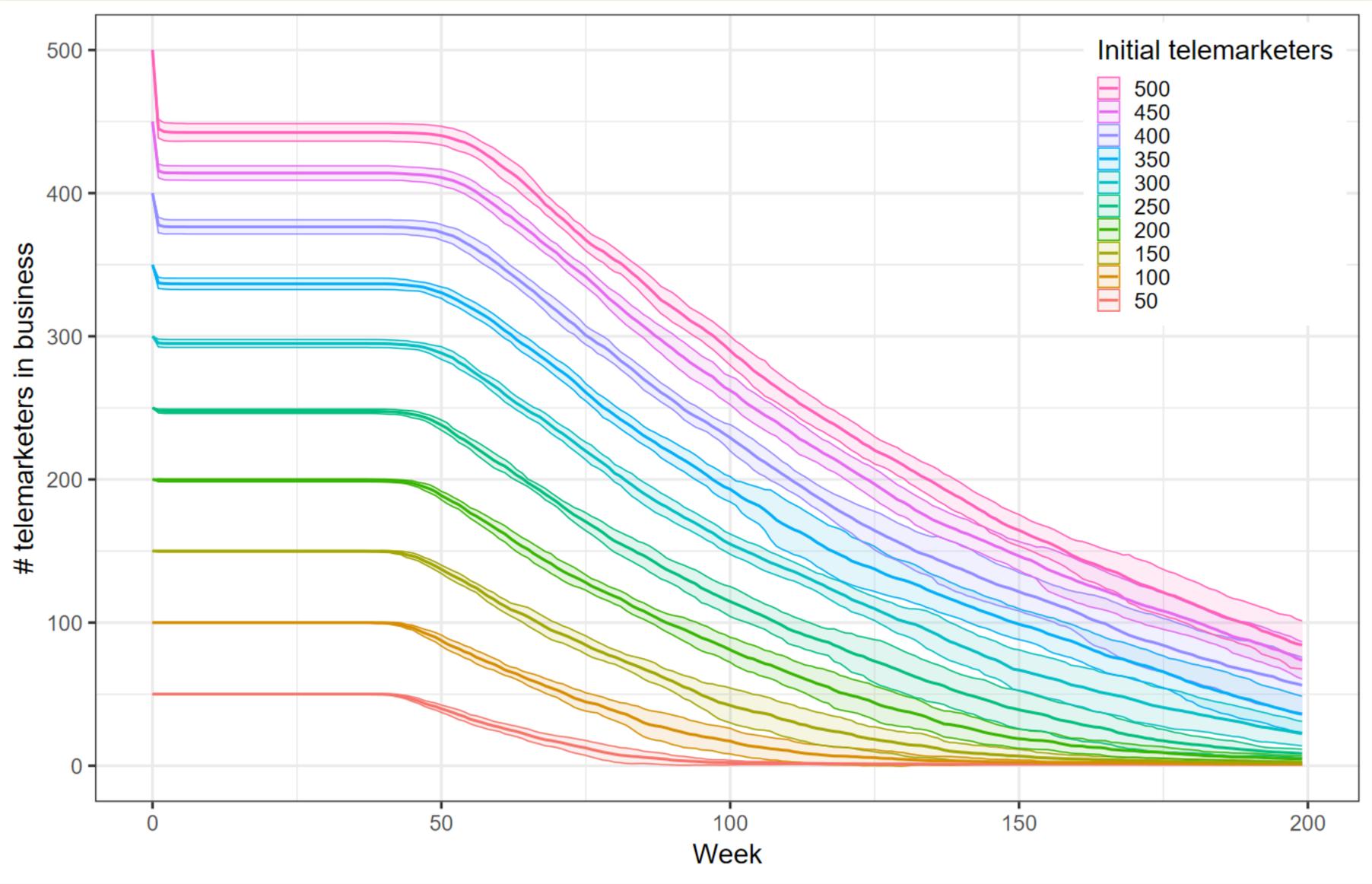
Implementing Mergers

- Turtles-own variable: parent
  - Initialize `set parent nobody` in `to setup`
  - When a merger happens, the broke turtle being acquired sets `parent` to the larger turtle buying it.
    - The owner does not have a record of the turtles it owns.
- Links
  - No special initialization
  - When a merger happens, create a directed link from the owner to the subsidiary.
    - Now turtles can track both their owners and their subsidiaries.
    - Easier to keep track of relationships
    - Relationships can be displayed on the model view.

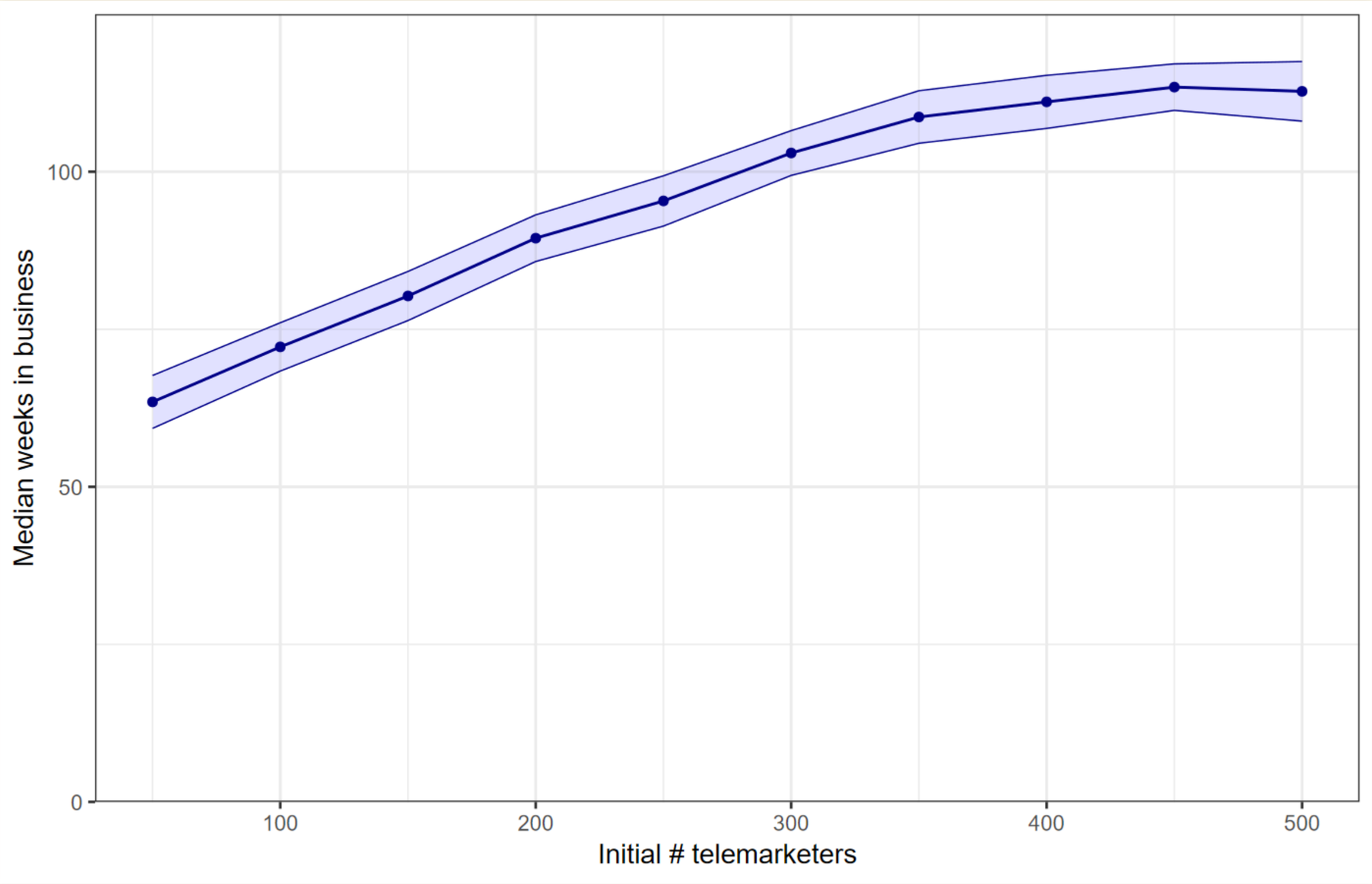
Results


Initial telemarketers
500
450
400
350
300
250
200
150
100
50
telemarketers in business
0
100
200
300
400
500
0
50
100
150
200
Week

Variation


Initial telemarketers
500
450
400
350
300
250
200
150
100
50
telemarketers in business
500
400
300
200
100
0
0
50
100
150
200
Week

Median Weeks in Business



Cooperation and Coordination

Game Theory

- Modern Formal Game Theory originated in the 1940s
  - John von Neumann, *On the Theory of Games of Strategy* (1944)
  - John F. Nash, Jr. (1994 Nobel Prize in Economics), “Nash equilibrium”
- Older history (informal)
  - Many centuries of writings on war and gambling
  - Sun Tzu, *The Art of War* (5th century BCE)

> *Knowing the other and knowing oneself: In one hundred battles, no danger.*
> *Not knowing the other and knowing oneself: One victory for one loss.*
> *Not knowing the other and not knowing oneself: In every battle, certain defeat.*

  - Girolamo Cardano, *Liber de Ludo Alea* (*Book on Games of Chance*) (1564)
- Big question:
  - If you are playing a game with another person, what is the best strategy?

John von Neumann (Photo: Los Alamos National Laboratory)

John Nash (Photo: MIT Museum)

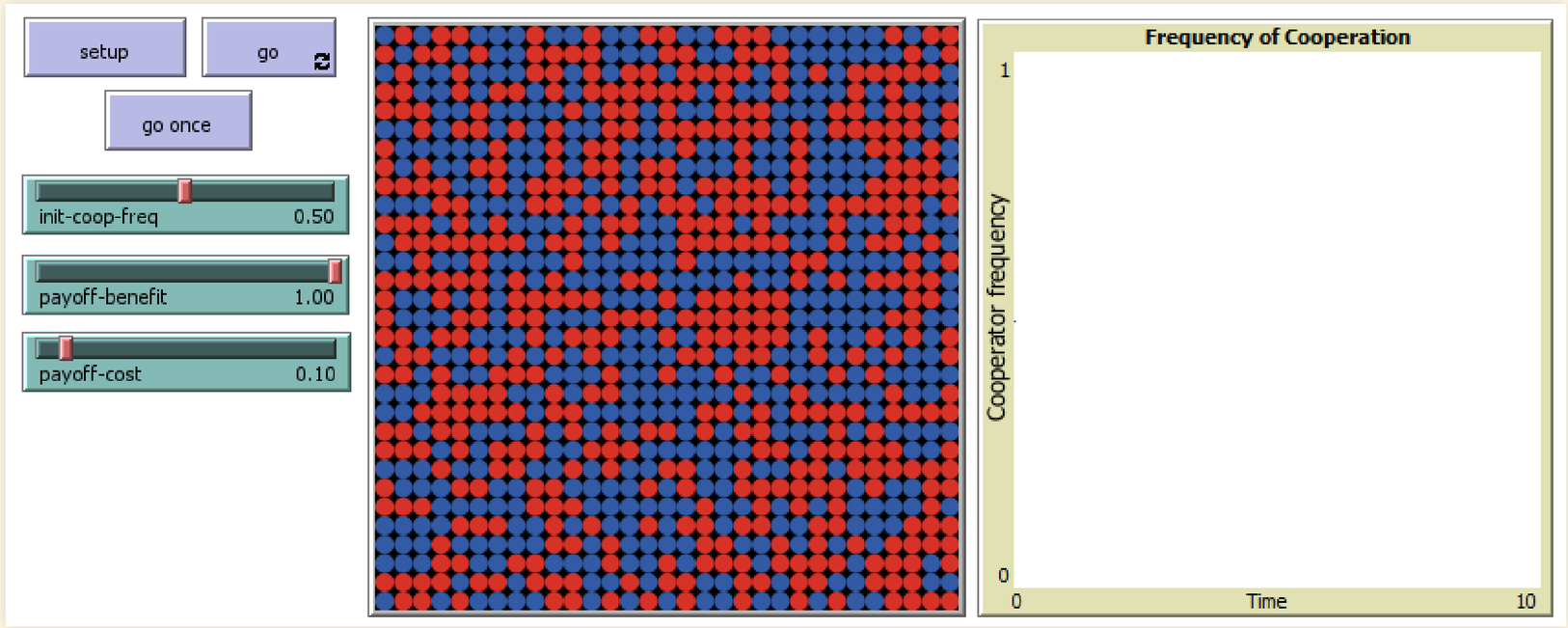
Cooperation vs. Defection (The Prisoner's Dilemma)

- Two people play, and they cannot communicate to discuss their moves.
  - Model:
    - Two people are arrested and accused of a crime
    - If both **cooperate** and remain silent, both are convicted of a minor crime
    - If one **defects** and testifies against the other, they are given a lighter sentence, and the other is convicted of a more serious crime and receives a harsh sentence
    - If **both defect**, they are both convicted of the serious crime, but receive some time off their sentence as a reward for defecting.
  - Mathematical representation:
    - Value for (A, B) of each choice
    - b = benefit of the other person cooperating (lesser crime)
    - c = cost of not defecting (don't get time off the sentence)

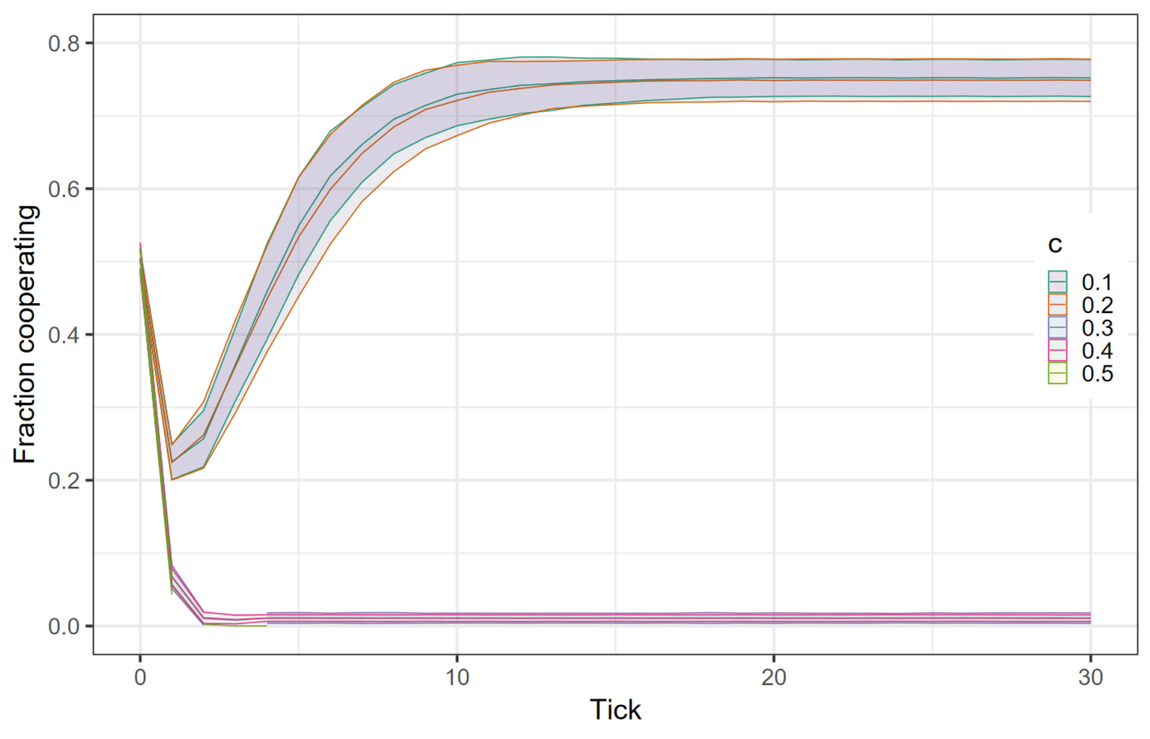
	B cooperates	**B defects**
A cooperates	$b - c, b - c$	$-c, b$
A defects	$b, -c$	0

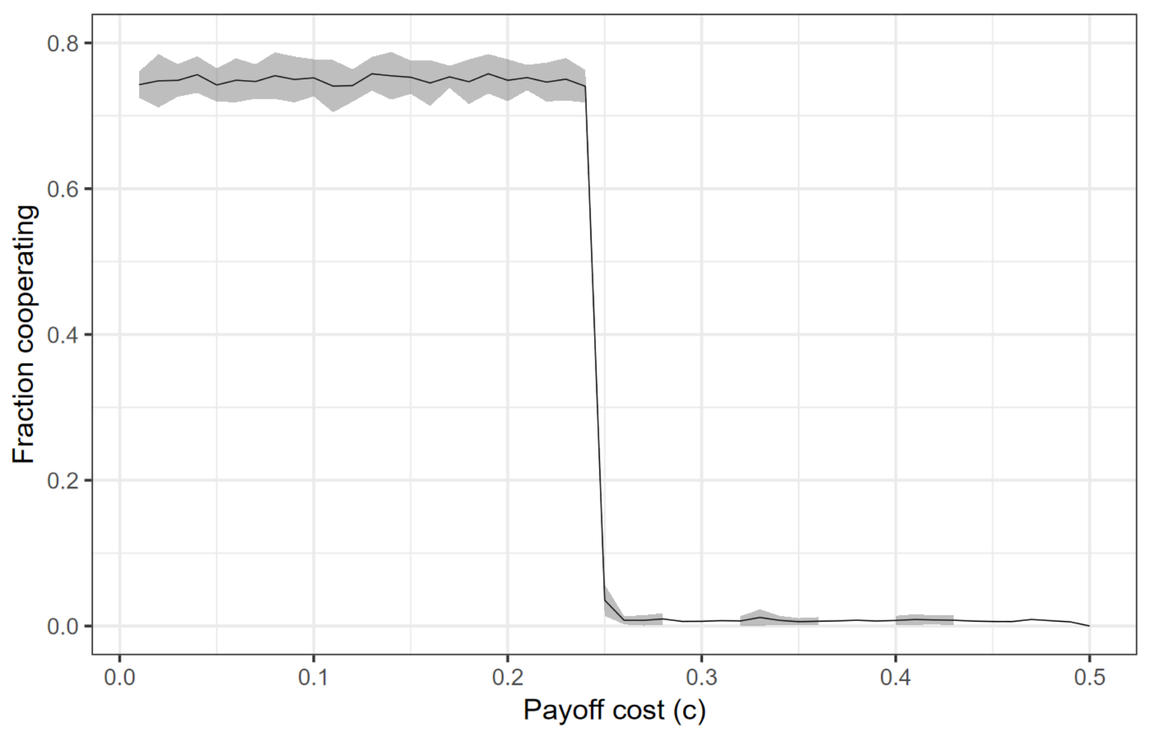
Agent-Based Model: `PD_simple.nlogo`

- Each turtle either always cooperates (blue) or always defects (red)
- Each turtle plays against the four neighbors with which it shares a side
- After each turn the turtle “evolves” by comparing its payoff to its four neighbors and copying the most successful strategy
- What happens when you vary the cost of not defecting (*c*)?



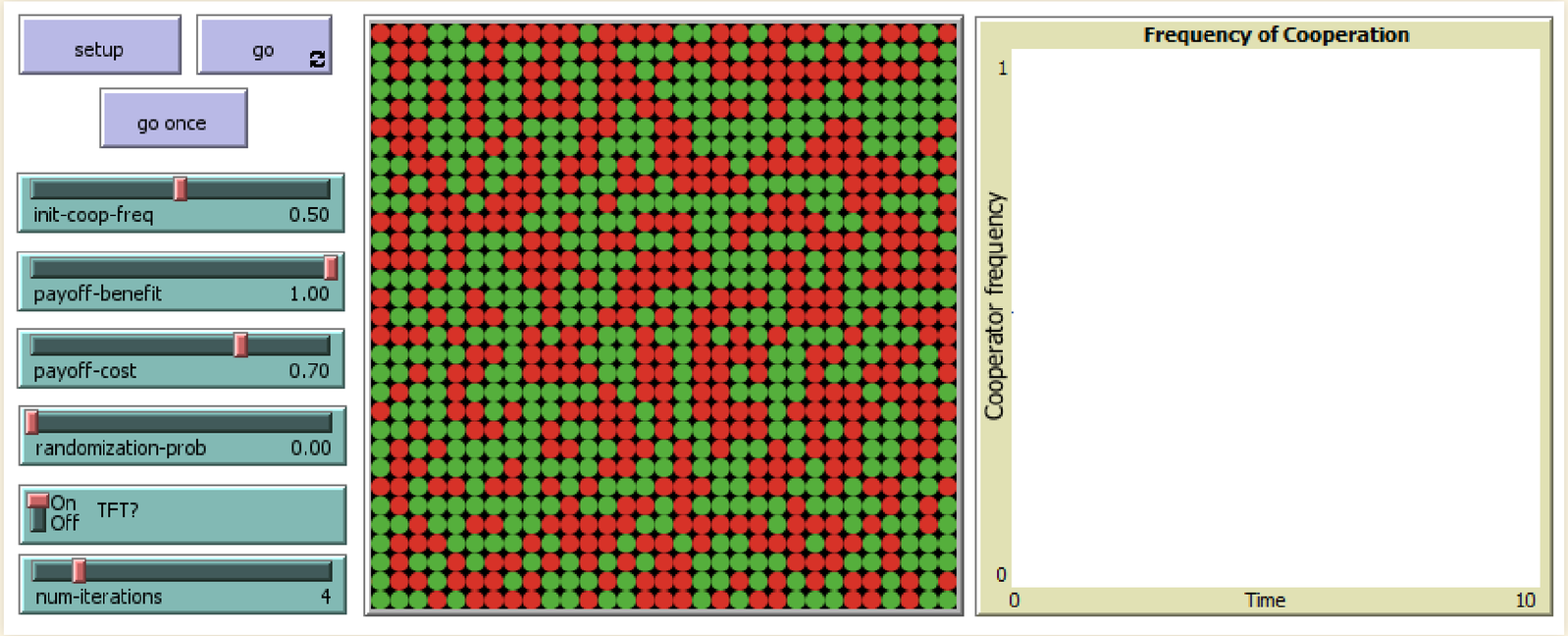
Vary cost (*c*)


Fraction cooperating
0.0
0.2
0.4
0.6
0.8
0
10
20
30
Tick
c
0.1
0.2
0.3
0.4
0.5

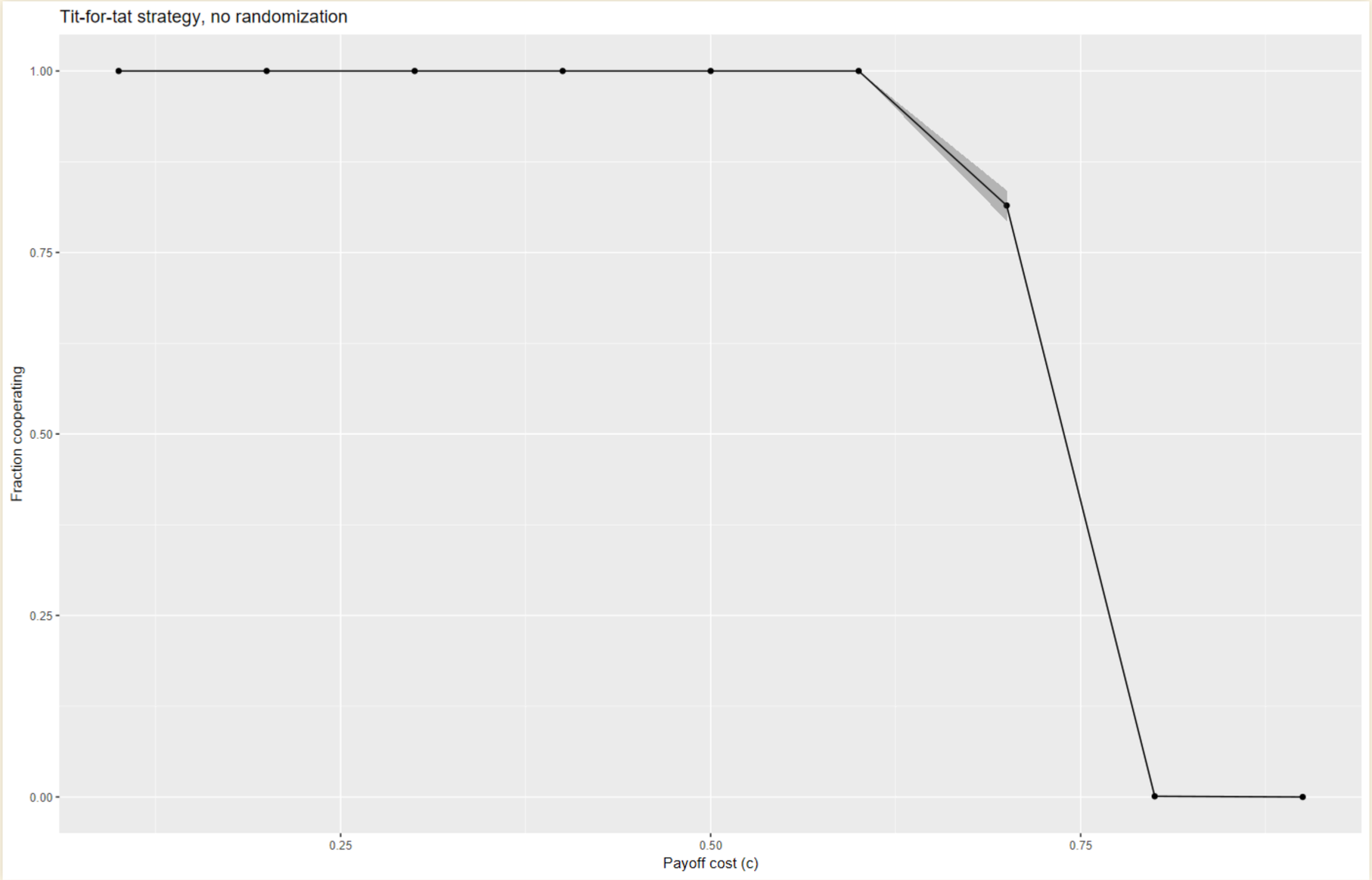

Fraction cooperating
0.0
0.2
0.4
0.6
0.8
0.0
0.1
0.2
0.3
0.4
0.5
Payoff cost (c)

Reciprocity: Tit-for-tat strategy

- PD_reciprocity.nlogo
- New strategy: tit-for-tat.
  - Start by cooperating, then copy whatever the opponent did the last time
  - Repeat 4 or more times per tick with each opponent
  - Randomization randomly swaps turtle positions, with some probability



Vary cost of not defecting (*c*)



Interaction between randomization and # iterations