Title

An Agent-Based Simulation of Preferential Aggregation Based on Past Experience in Fission-Fusion Societies

Author

Blonder, Aviva

Publication Date

2024

An Agent-Based Simulation of Preferential Aggregation Based on Past Experience in Fission-Fusion Societies

By

AVIVA BLONDER
DISSERTATION

Submitted in partial satisfaction of the requirements for the degree of

DOCTOR OF PHILOSOPHY

in

Animal Behavior

in the

OFFICE OF GRADUATE STUDIES

of the

UNIVERSITY OF CALIFORNIA

DAVIS

Approved:

Jeffrey Schank, Chair

Brenda McCowan

Cristina Moya

Committee in Charge

2024

Abstract

Agent-based simulations can be a powerful tool for exploring possible evolutionary trajectories, community structures, or social network processes. Much of their power comes from the ability to run vast quantities of simulations and cover a broad parameter space. Here we present the SimDataCollection package, which automates running simulations over wide swaths of parameter space defined by all possible combinations of user-specified parameter values and gathering data from agent-based simulations built using the MASON Multi-agent Toolkit. The SimDataCollection package was employed in the two following simulations of the formation of aggregation preferences based on repeated interaction, or familiarity, in fission-fusion societies. A variety of species across taxa preferentially associate and cooperate with familiar individuals (those they have interacted with before) over those which they have not interacted with. This tendency to aggregate and cooperate with familiar individuals occurs even in fission-fusion societies, characterized by the frequent splitting and combining of groups, resulting in highly dynamics populations, which are highly prevalent in nature and should make the formation of particular partnerships difficult. In our first simulation, we employed an evolutionary agent-based model to illustrate that a preference for aggregating with familiar individuals could in fact make cooperation evolutionarily stable in such fission-fusion societies. Our second simulation focused on the effect of fission-fusion dynamics and the formation of associations over repeated interaction on the emergence of social networks, which we quantified using a variety of network approaches. These simulations present an initial view into the possible power of fission-fusion dynamics to effect social structure and therefore evolutionary outcomes, however, these

simulation results can only go from conjecture to scientific truth when combined with empirical results.

Introduction

Who interacts with whom, and why? An important, and highly complex question, influenced by spatial proximity, environmental context, the type of interaction, attributes of the individuals themselves, and their past experience (Farine et al., 2015). Furthermore, it is becoming increasingly clear that these individual interactions don't occur in a vacuum; social interactions can not only be affected by who your partner is, but also who else you and your partner interact with, and even the social structure of the whole population—the entire social network (Pinter-Wollman et al., 2014). In turn, who interacts with whom can have wide-spanning implications for the transmission of information or disease, the distribution of resources, reproductive success, and the evolution of populations (Farine et al., 2015). Perhaps unsurprisingly, the complexity of social structures makes them difficult to study. When a sample is an entire group or population, sample sizes are inevitably small, and it is rarely feasible to keep track of all interactions among all individuals at all times. As a result, determining the causes and results of social structure can be particularly difficult. One proposed solution is using agent-based models, which simulate each individual in a population and the interactions between them, to predict the effect of various individual- and group-level factors on social structure, which can then be compared to empirical networks (Farine et al., 2015).

There has already been extensive work simulating various facets of social structure and its effects. There are countless models of the effect of social network structure on transmission and evolutionary processes, and their interaction with various rules for changing the connections between individuals (Szabó & Fáth, 2007). Likewise, there has been remarkably robust empirical and theoretical work done exploring the dynamics of collective movement; how individuals interact in space and who they choose to follow (Couzin et al., 2002). However, the intersection between these domains is more sparse (Bode et al., 2011b). Limited work has been done on the effect of social network structure on collective movement dynamics (Bode et al., 2011a), and even less has been done on the effect of collective movement on social network structure (Bode et al., 2011b) even though movement dynamics are expected to have a sizable effect on social networks in natural populations (Farine et al., 2015).

Collective movement is expected to substantially constrain the possible connections that can form between individuals (Farine et al., 2015), and may have an even more substantial effect on associations based on past interaction. In particular, animals of many species across taxa prefer to associate with familiar individuals over unfamiliar ones (Ward & Webster, 2016; Griffiths & Ward, 2011; Massen et al., 2010; Wilkinson et al., 2019; Gherardi et al., 2012). It is reasonable to expect that preferentially associating with familiar individuals may result in feedback loops where individuals that happen to interact in the course of collective movement become familiar, causing them to preferentially associate, changing the dynamics of collective movement, which in turn further shapes the social structure. These common preferences for familiar individuals are especially noteworthy because animals of many species not only associate with familiar individuals, but also cooperate with them, in some cases even in the absence of kinship or reciprocal relationships (Griffiths & Ward, 2011).

To begin to address the complex interaction between collective movement, social structure, and evolutionary outcomes, I constructed an agent-based model. The model consisted of agents moving in space analogous to flocking birds, schooling fish, or herding ungulates (Joshi et al., 2017). The agents formed associations with their neighbors over repeated interactions and I examined the effect of preferentially aggregating with their familiar associates. In the first chapter, I present a package which I used to automate running and analyzing the model to make the most of agent-based models’ versatility and the ability to explore the vast parameter space of these complex models. In the second chapter, I present the model of collective movement and the formation of familiarity, and examine how a preference for aggregating with familiar individuals influences the evolution of cooperation in a prisoner’s dilemma game. Finally, in the third chapter, I further analyze the model to examine the effects of collective movement and the formation of associations based on familiarity on the resulting social network of familiarity associations. Together, these three chapters demonstrate the complex connection between collective movement and social structure, and the advantages and disadvantages of the agent-based modeling approach.

Chapter 1: SimDataCollection – A Package for Running and Gathering Data from Agent-based Simulations Based on the MASON Multi-agent Toolkit

Introduction

Agent-based models are a powerful tool for simulating complex processes across domains, from robotics, to sociology, to ecology and evolution. Agent-based models are defined by how they represent individuals, whether robots, animals, or molecules, as agents that are programmed to have their own attributes and behaviors which determine how they interact with each other and the environment. This individual-level resolution enables researchers to simulate the emergence of complex collective phenomena based on simple rules (Couzin & Krause, 2003), and to precisely replicate actual populations to simulate possible outcomes. Agent-based models have been especially valuable in studying the likely effects of anthropogenic change on ecological communities across multiple levels of organization (An, 2012). Within populations, agent-based models have revolutionized the study of the effect of network dynamics on processes such as disease transmission (Parker & Epstien, 2011) and social evolution (Szabó & Fáth, 2007). Agent-based models may be especially valuable in studying dynamic networks, as evidenced by their recent use in studying the feedback between collective behavior and network structure (Bode et al., 2011; Sueur et al., 2018). With increases in computational speed and storage capacity, the power of agent-based models has only increased (Bradhurst et al., 2015). A wide variety of software packages have been published to enable researchers across domains to create and run agent-based models without needing to recreate the basic infrastructure (e.g. NetLogo, MASON, MESA, etc…).

Like other modeling approaches, one of the greatest advantages of agent-based models is the capacity to run large numbers of simulations and test many possible combinations of parameter values. Agent-based models also allow researchers to gather a wide variety of information about agents' actual states and the state of the larger system, which would be impossible to measure or even observe in the real-life systems being modeled. Like creating the models themselves, running vast numbers of simulations and collecting data requires non-trivial domain general infrastructure for reading in sets of parameter values, running arbitrary numbers of simulations, and outputting data to files in a convenient format for analysis. Agent-based models can also generate data at a variety of different levels, including simulation-level averages, time course data during a run of a simulation, agent-level variables, and social network structures.

Here I present SimDataCollection – a package capable of running and gathering data at multiple levels from simulations built using the MASON Multi-agent Simulation Environment (Luke et al., 2005). MASON is a fast, versatile simulation toolkit that facilitates building agent-based simulations with large populations of agents embedded in space, networks, or other structures. In addition, MASON is built on the general purpose Java programming language, which gives the modeler nearly infinite flexibility in designing a model suited to their particular application (Luke et al., 2019). However, MASON has no built-in means for running more than one simulation at a time or getting raw data out of a simulation and Java lacks the data analysis tools that have been developed for languages like R and Python. SimDataCollection was created to make it easy for modelers to automate running many simulations with different combinations of parameter values, and to efficiently gather all of the relevant metrics in a format convenient for analysis in R or Python (Lee et al., 2015). Additionally, the SimDataCollection package is open source with the code available on GitHub (https://github.com/ablonder/SimDataCollection), making it easy for users to add additional functionality as the need arises, which can then be made available for the whole community.

The SimDataCollection package enables users to designate parameter values and automates the running of multiple replicates for all combinations of values for an arbitrary number of parameters. SimDataCollection also includes utilities for randomly sampling a parameter space by drawing parameter values from a variety of distributions (Thiele et al., 2014). The user has the option to collect the values of any simulation- or agent-level variables. All results are written to an output file at user-defined intervals throughout each run of the simulation. SimDataCollection also provides the option to output edge lists for any network built using the MASON Network class. I originally built SimDataCollection for use with my own models of the effect of collective movement on the evolution of cooperation and social network structure, but I have adapted it into a domain-independent framework for running and gathering data from agent-based models.

Architecture

SimDataCollection is implemented in Java as a subclass of MASON's SimState class; the superclass on which all MASON models are built. To use SimDataCollection the user defines their own model class which extends SimDataCollection (an example ModelTemplate is provided within the SimDataCollection repository). This allows the user to access all standard MASON utilities while enabling methods in SimDataCollection to handle setting parameter values from a user-created input file, running the simulation, and writing data to output files. Subsequent sections go into each of these steps in greater detail.

The Input File

All but the very simplest and most rigid agent-based models use variables, called parameters, to alter the model's behavior. The essence of running simulation experiments is changing the parameters' values and observing changes in the outcome. In MASON, the only ways to change parameter values is to alter the value of a variable in the code directly, or to use the Graphical User Interface (GUI). SimDataCollection instead uses more versatile input files, which users can modify at will.

A SimDataCollection input file is automatically generated for each model, and consists of a list of variables from the simulation that the user can assign to values or collect from the model as results. The input file also provides several key parameters (Table 1) that enable the user to indicate how many steps to run the simulation for, how frequently to collect output measures at each level, and how many replicates to run for each set of parameter values, or whether to instead run the simulation with a GUI. The user can provide multiple values for as many parameters as they like and the simulation will be run for all possible combinations of parameter values (warning: this may result in a very large number of simulations). The input template also allows the user to indicate whether they would like a parameter to be randomly drawn from a distribution (current options: uniform, normal, or gamma) and to specify the arguments to characterize the distribution. The input file can only be used to set simulation-level parameters, but it can be used to gather data at the simulation-, time course-, agent-, and network-levels.

(Detailed instructions for how to use the input file are provided in the input template automatically generated by running a simulation using SimDataCollection.)

Key Parameter	Use
seed	Random seed for the MersenneTwister for the first replicate for each combination of parameter values (incremented for each additional replicate)
sep	Separator character for the output file (optional: defaults to ',')
steps	Maximum number of timesteps each run of the simulation lasts
iters	Number of sets of randomly drawn parameters (optional: only used when randomly drawing parameter values)
reps	Number of times the simulation is run for each combination of parameter values
fname	Beginning of the name of all output files created from running the model with this input file
testint	How frequently simulation-level data is collected (in timesteps)
agentint	How frequently agent-level data is collected (in timesteps; optional: only used if agent-level data is collected. If not provided, defaults to testint)
netint	How frequently network edge lists are outputted (in timesteps; optional: only used when edge lists are outputted. If not provided, defaults to testint)
teststart	How many steps into each run to start collecting data, allowing for a burn in period (defaults to 0; the beginning of the simulation)
gui	Whether the simulation is run with or without a GUI (optional: defaults to false; if true, the simulation only runs with the initial set of parameter values and does not use any of the other key parameters except for seed)

agentInfo	A list of agent-level variables to be collected (optional)
edgeList	A list of the names of MASON Network fields of the model class whose edge lists are to be outputted to file (optional)

Table 1. The names and uses of all key parameters for running a simulation from an input file using SimDataCollection. All key parameters are required for a simulation to run unless indicated as optional. See a sample input template at https://github.com/ablonder/SimDataCollection/blob/master/inputTemplate.txt.

Running the Simulation

From initializing the model, SimDataCollection handles setting parameter values, running simulations and gathering data according to the provided input file. SimDataCollection parses the input file to read in the necessary information, draws the desired number of iterations of random parameter values, recursively sweeps all possible combinations of provided parameter values, runs the simulation for the desired number of replicates, sets the value of each parameter for each run, and gathers the model, agent, and network results, outputting them to a file (Figure 1).

```
Parse input file
For each iteration:
    Draw random parameter values
    For each combination of provided parameter values:
        For each replicate:
```

```
Set parameter values
For each timestep:
    Run the simulation
    Output variables to file
```

Figure 1. The high-level structure of SimDataCollection.

Randomization

A user can designate which parameters they want to be randomly drawn by indicating the distribution they want to draw from (uniform, Normal, Beta, or Gamma) and the parameters for that distribution (e.g. min and max, or mean and variance). See the SimDataCollection repository on GitHub for all options and notation used (https://github.com/ablonder/SimDataCollection/blob/master/inputTemplate.txt).

If the user has designated any parameters to be randomly drawn from a distribution, a new instance of the built-in MASON random number generator, MersenneTwister, is initialized with the user-provided random seed. This random number generator is used to draw the designated parameter values for each iteration. It uses the methods in the distSampler class to efficiently and stably draw from Beta, Gamma, and Erlang distributions. The same set of randomly drawn values is used for all runs of the simulation (sweeps and replicates) in that iteration (see Figure 1) and all randomly drawn values are included in the output file for analysis.

Sweeps

If multiple values have been provided for any parameter(s), the simulation is run with all possible combinations of the provided parameter values. Recursion enables the user to evaluate all possible combinations of values of an arbitrary number of parameters (this should be used cautiously as the resulting number of runs grows exponentially). The simulation is run multiple times for each combination of parameter values according to the user designated number of replicates (see Table 1). The user-provided random seed is incremented for each replicate and reset to the initial user-provided value for each new combination of parameter values to ensure replicability.

Setting Parameter Values

Each time the simulation is run, the parameter values are reset from the current complete list of parameter values (updated recursively for each combination of parameter values). Public primitive and string fields of the model class are set automatically using Java's Reflection utilities. The modeler can also overwrite the built-in function in the model class to manually interpret more complex parameters.

Outputting to File

Over the course of each simulation, the results are written to output files according to the test (or agent or network) interval. Public primitive, String, and Collection fields of the model and agent classes are gathered automatically using Java's Reflection utilities. Edge lists are automatically generated from MASON Network fields of the model class by looping through all edges in the network and getting their source, target, and info. The modeler can also overwrite the built-in function in the model class to manually calculate more complex output measures.

For each input file, one output file is created for each result category, including time course results, end of simulation results, agent-level results, and network edge lists. Each output file documents the random seed for each run of the simulation and the parameter values used for efficient analysis in R or Python, supported by the combineFiles R function, which is included in the repository.

Applications

SimDataCollection has already been used in models of self-censorship (Zhong et al., in prep) as well as undergraduate projects in the Schank lab, including a project on the evolution of empathic cooperation and an honors thesis on the impact of publication bias on published effect size distributions. Much of the development of SimDataCollection was driven by my own needs as I worked on the following two chapters. It was used to run all of the simulations and gather model- and agent-level data on the effect of a preference for associating with familiar individuals on the evolution of cooperation, and to get the raw network data that I used to examine the effects of movement dynamics on association network structure.

Future directions include the addition of the option to collect lists of data from the simulation, and utilities for importing empirical networks. I may also add more distributions for parameter randomization and additional functionalities as needed by users (contact afblonder@gmail.com to discuss potential new features).

Chapter 2: Preferential Aggregation with Familiar Individuals Can Promote the Evolution of Cooperation in Fission-Fusion Societies

Introduction

Aggregation with conspecifics is essential to countless species across taxa. Animals find safety in numbers, have increased access to potential mates, huddle together for warmth, forage more successfully in groups, and sometimes even cooperate, benefiting others at a cost to oneself, but sociality is not without costs, such as increased competition and disease transmission. The benefits of aggregation often depend on which individuals aggregate together (Farine et al. 2015). Accordingly, a wide variety of species have been found to exhibit preferences for aggregation partners, not only based on phenotype, but also on past experience. Preferences for which individuals to aggregate with are likely to be especially important to dynamic, fission-fusion societies, like schools of fish, flocks of birds, and herds of ungulates, characterized by the splitting, or fissioning, and combining, or fusing, of groups.

In particular, animals of many species, including those in fission-fusion societies, preferentially associate with familiar individuals, or, those they have encountered frequently, over those they have never, or rarely encountered (Ward & Webster, 2016; Griffiths & Ward, 2011; Massen et al., 2010; Wilkinson et al., 2019; Gherardi et al., 2012). It has been proposed that familiar individuals are preferable partners because they are more cooperative and less aggressive, and therefore require less attention to monitor than unfamiliar individuals (Griffiths et al., 2004). Indeed, associating with familiar individuals has been found to convey a variety of cooperative benefits even among non-relatives and in dynamic, fission-fusion societies (Griffiths & Ward, 2011; Massen et al., 2010; Ward & Webster, 2016). Many species of fish are known to prefer to associate with familiar individuals (Griffiths & Ward, 2011), and schools of familiar individuals have been found to exhibit increased cooperation and cohesion in predator defense (Chivers et al., 1995), decreased aggression (Webster & Hart, 2007), and increased foraging success (Ward & Hart, 2005), and repeatedly engage in predator inspection with the same partner, despite frequently low relatedness in schools (Griffiths & Ward, 2011). Many mammals, including humans, cooperate more with familiar individuals (Massen et al., 2010). Female bats of several species roost with familiar individuals and engage in cooperative behaviors, such as greater spear-nosed bats (*P. hastatus*) protecting unrelated roostmates' offspring from infanticide, and vampire bats (*D. rotundus*) sharing blood meals with kin and non-kin (Wilkinson et al., 2019). Similarly, long-term associations with non-relatives afford female wild horses greater protection from harassment by males (Cameron et al., 2009). Ravens are more likely to support familiar individuals in accessing resources (Szipl et al., 2015). Even some invertebrates are less aggressive and more cooperative with familiar individuals (Gherardi et al., 2012).

It is particularly remarkable that individuals of so many species cooperate with familiar individuals because in dynamic fission-fusion populations, there is no reason to expect that the conspecifics that an individual happens to become familiar with should be less aggressive or more cooperative. We propose that preferentially aggregating with familiar individuals may in fact promote the evolution of cooperation in fission-fusion societies, resulting in populations that exhibit cooperation with familiar individuals. We evaluated this hypothesis using an agent-based model where individuals, or agents, moved around a simulated space like schools of fish or flocks of birds, and offspring dispersed widely from their parents, resulting in low relatedness between interacting individuals (Joshi et al., 2017). Agents played a prisoner's dilemma with their spatial neighbors, to represent the most difficult case for the evolution of cooperation. We included a complete model of selection, where selection for cooperation acted on survival as well as fecundity (Rodrigues & Kokko, 2016). This makes the model particularly well-suited to explaining examples of cooperation in nature like predator inspection in fish (Milinski et al., 1997), and blood meal sharing in vampire bats (Wilkinson, 1984; 1990), where there is a clear risk of mortality. Whenever two agents interacted, they accumulated familiarity, and familiarity between a pair of agents gradually decayed when they did not interact. We compared a condition where agents preferentially aggregated with familiar individuals to a condition where agents exhibited no preference.

We demonstrate that a preference for aggregating with familiar individuals can favor cooperation at higher population density and with weaker selection than when individuals have no preference between familiar and unfamiliar conspecifics. In particular, our model suggests that a preference for aggregating with familiar individuals divides up a population, so that each individual has fewer potential partners to interact with. As a result, defectors are more likely to over-exploit their few cooperative neighbors, decreasing their chances of survival. When a defector's cooperative neighbors eventually die out, the defector will then be left isolated and will, in turn, have a lower chance of reproducing and surviving. This further demonstrates how interdependence, an underexplored mechanism for assortment among cooperators (Queller, 2011; Bshary et al., 2016), may emerge in natural populations as a result of association preferences based on familiarity.

Model Overview

The model description follows the ODD (Overview, Design concepts, Details) protocol for describing individual- and agent-based models (Grimm et al., 2006), as updated by Grimm et al. (2020).

We created a model written in the Java programming language using the MASON Multi-agent simulation toolkit (Luke et al., 2005). We simulated populations of individuals, or agents, in a continuous, two-dimensional spatial environment. Agents moved through the space like flocking birds, schooling fish, or herding ungulates, based on the active, self-propelled movement model of Joshi et al. (2017), which was used to study the coevolution of cooperation and aggregative tendency in fission-fusion populations. Agents moved towards and aligned their movement with other agents within their aggregation radius, but were repealed by and moved away from those that got too close.

We built on their movement model to simulate preferential aggregation with familiar individuals. Individuals weighted movement towards and alignment with familiar individuals versus unfamiliar individuals based on a familiarity bias parameter, which ranged between 0 (only moving toward and aligning with unfamiliar individuals) and 1 (only moving toward and aligning with familiar individuals), with no preference between familiar and unfamiliar individuals at .5. We modeled familiarity as a simple learning process over repeated interaction. Agents' degree of acquaintance increased whenever they interacted and decayed over time when they did not interact. A threshold determined whether agents were sufficiently well acquainted to be considered familiar, constrained by limited memory capacity.

Agents interacted by playing a prisoner's dilemma game with their neighbors, to represent the most stringent form of cooperation. Each agent either always defected or always cooperated, and accumulated or lost payoff accordingly, which determined whether the agent was able to survive and reproduce. Agents passed on their cooperative strategy to their offspring with some small chance of mutation, allowing cooperative strategy to evolve over the course of the simulation. All parameters were varied to evaluate the conditions under which cooperation could persist and to analyze its stability.

Entities, State Variables, and Scales

Entities: Our model consists of 1000 agents, which represent individuals of a focal species, analogous to schooling fish, flocking birds, or herding ungulates.

State Variables:

Variable	Meaning	Range	Value
Cooperative strategy	Whether this agent employs the cooperator or defector strategy when it plays another agent in a prisoner's dilemma.	Cooperator or Defector.	Static - based on the agent's parent's strategy or the initial proportion of cooperators.
Familiar bias	This agent's weight in favor of aggregating with familiar agents as opposed to unfamiliar agents.	From 0 (only aggregate with unfamiliar) to 1 (only aggregate with familiar).	Static - drawn from a Beta distribution with the population mode and "sample size".
Interacted agents	All agents this agent has interacted with and its degree of acquaintance with each agent.	Weighted, directed network of agents.	Dynamic - agents accumulate acquaintance as they interact and lose acquaintance when they

			do not interact.
Familiar agents	Which agents this agent considers to be familiar.	Directed network of agents.	Dynamic - agents can be added and removed based on changes in this agent's degree of acquaintance with them.
Memory capacity	The maximum number of agents this agent can consider to be familiar at a time.	From 0 agents to the maximum population size.	Static - drawn from a Gamma distribution with the population mean and variance.
Familiarity threshold	Degree of acquaintance with another agent necessary to consider it familiar.	From 0 (no acquaintance necessary) to an arbitrarily high degree of acquaintance.	Static - drawn from a Gamma distribution with the population mean and variance.
Familiarity decay	How much this agent's acquaintance with other agents decreases on each step in which they do not interact.	From 0 (no decrease) to 1 (lose acquaintance at the same rate as it is accumulated).	Static - drawn from a Beta distribution with the population mode and "sample size".
Accumulated payoff	Net payoff this agent has accumulated over the course of the simulation.	From 0 (no payoff) to 1 (threshold for reproduction).	Dynamic - payoff can be gained or lost from playing a prisoner's dilemma, exogenous payoff, and the cost of reproduction.
Maximum lifespan	Maximum number of timesteps this agent can survive.	From 0 steps to the total length of the simulation.	Static - drawn from a Gamma distribution with the population mean and variance.
Age	Number of timesteps this agent has been alive so far.	From 0 steps to its maximum lifespan.	Dynamic - increases by 1 on each timestep from when an agent is created until it dies.
Location	The agent's x-y coordinates in the simulated space.	A pair of coordinates each of	Dynamic - changes as the agent moves through

		which can be between 0 and the space size.	the space according to the movement algorithm.
Direction of movement	How much this agent moved on the x and y axis in the previous step.	A pair of x-y values whose tangent is equal to the simulation step size.	Dynamic - changes as the agent moves through the space according to the movement algorithm.

Table 1. All agent state variables, their meanings, ranges, and how their values are set.

Scales: The model employs a two-dimensional continuous space with periodic (toroidal) boundaries, representing an infinitely large area without edge-effects. We varied the dimensions of the space to account for different population densities, around a typical space size of 90x90 units. Each unit of space with a radius of one is roughly analogous to the “personal space” of a single agent to allow for schooling or flocking interactions (after Joshi et al., 2017) within a sizable population.

Time is modeled in discrete steps. On each step, all agents are able to move a short distance, allowing for precise simulation of schooling or flocking behavior. Cooperative interactions occur on the same timescale. We varied the average generation time around a typical mean maximum lifespan of 500 timesteps, which provided agents with sufficient time to aggregate as well as accumulate payoffs to reproduce. We ran the simulation for 100 generations, or 50,000 timesteps.

(Note that in subsequent sections, agent’s state variables are in *italics*, while simulation-level parameters, which will be defined with the subroutines, are in *italics and underlined*)

Process Overview and Scheduling

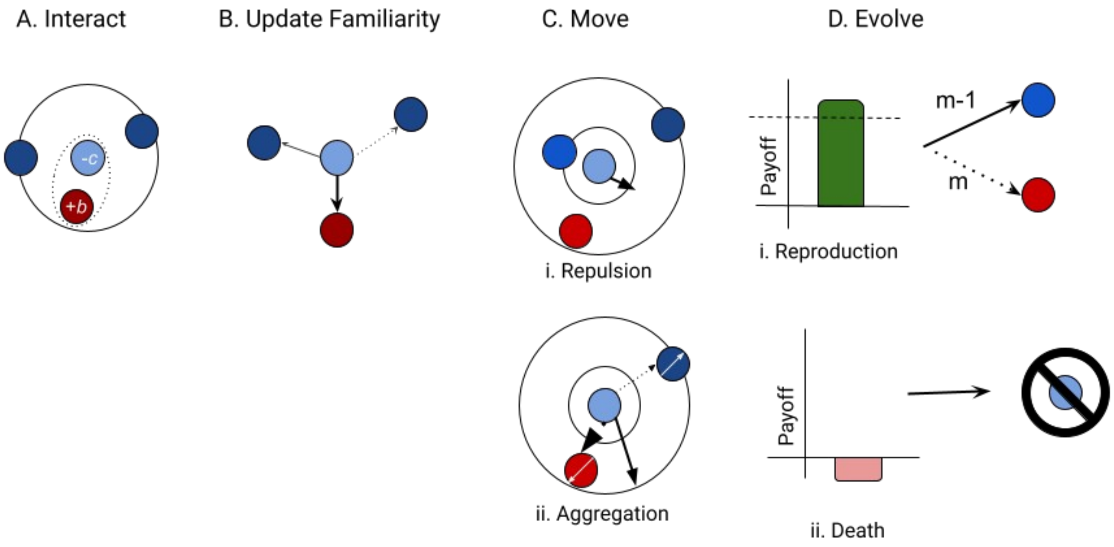


Figure 1. Schematic of the actions each agent takes on each step of the simulation. The focal agent is depicted as the pale blue circle, defectors are red, and cooperators are blue. (A) The focal agent chooses a partner at random from the agents within its interaction radius (large black circle) and plays a prisoner's dilemma (with the other agent in the dotted circle). Cooperators pay a cost of -*c* to give their partner a benefit of *b*, while defectors pay no cost and confer no benefit. (B) Irrespective of the outcome of the interaction, the focal agent increases its degree of acquaintance with its partner and may come to consider it familiar (large, dark arrow). Its acquaintance with all other individuals decays by a small amount (small, light arrows). If its acquaintance with another agent drops too low, it no longer considers that agent to be familiar (small, dotted arrow). (C) Agents move according to a standard self-propelled movement model (actual direction of movement represented by the solid black arrow); i. if there are any agents within the focal agent's repulsion radius (small black circle) it moves away from their average location. ii. If there are no agents within the focal agent's repulsion radius, but there are agents

within its aggregation radius (large black circle), it moves toward them and to align with their direction of movement (small white arrows) with a bias in favor of familiar agents (large dotted arrow) over unfamiliar agents (small dotted arrow). (D) i. If the focal agent's accumulated payoff (green bar) from playing prisoner's dilemmas and the set exogenous payoff on each step exceeds the reproduction threshold (dotted line) and the total number of agents is less than the maximum population size, it reproduces, creating a new agent of the same strategy (blue circle), with a small chance of mutation (*m*; red circle) at a random location in space. ii. If the focal agent's accumulated payoff drops below zero (red bar), the agent dies (black crossed out circle).

On each timestep, all agents are randomized and each carries out the following processes in the following order:

1. If there are any agents within the agent's <u>*interaction radius*</u>, it chooses a random other agent to interact with and:
    a. Update Familiarity: both agents update their *interacted agents* and *familiar agents* to increment their degree of acquaintance with each other, and determine whether they now consider each other to be familiar.
    b. Interact: both agents update their *accumulated payoff* based on the outcome of a prisoner's dilemma game.
2. Move: updates the agent's *location* and *direction of movement* based on its current *direction of movement* and the *location*s, *direction*s *of movement,* and *familiarity* of the other agents within its <u>*aggregation radius*</u>.

3. Prune Network: decrements the degree of acquaintance with the other agents in this agent's *interacted agents* based on its *familiarity decay* and updates its *familiar agents* accordingly.
4. Evolve:
    a. The agent adjusts its *accumulated payoff* based on the *exogenous payoff*.
    b. The agent increments its *age* by one.
    c. If the agent has enough *accumulated payoff* to create a new agent, it adjusts its *accumulated resources* according to the *reproduction cost* and, if the total number of agents is less than the *maximum population size*, a new offspring agent is created.
    d. If the agent's *age* is above its *maximum age* or its *accumulated payoff* is less than 0, the agent is removed from the population.

On each step, the order in which the agents act is randomized to avoid order effects. Typically, all agents move a much smaller distance in the space than their *interaction radius*, so the order in which Move and Interact occurs is largely arbitrary. Prune Network occurs after Move, so that the agents have an opportunity to aggregate with newly familiar individuals before their degree of acquaintance is decremented. All payoff is accumulated before the agent reproduces or dies, enabling it to more accurately assess whether it has sufficient *accumulated payoff* to reproduce on that step or whether its payoff has dropped below 0. The order of reproduction and death is largely arbitrary because agents only reproduce if they have sufficient payoff, and the *reproduction cost* is never higher than the *reproduction threshold* so agents never die immediately after reproduction. The only exception is when an agent has passed its *maximum lifespan*; Reproduction comes before death to give the agent a final chance to reproduce before it dies, but this should make little difference given how long agents' *maximum lifespan*s are.

Algorithm 1. Step. On each step all agents are randomized and each may have the opportunity to interact with a neighboring agent, update their degree of acquaintance and familiarity with that agent, play a prisoner's dilemma and accumulate payoff accordingly, move, prune inactive familiar connections, reproduce, and die.

```
if this agent has any neighbors within interaction radius units of its location then
    Choose a random neighbor
    UPDATE FAMILIARITY between this agent and its neighbor
    INTERACT with its neighbor and update both agent's accumulated payoff
end if
MOVE
PRUNE NETWORK of this agent's interacted agents and familiar agents based on this agent's  familiarity decay
EVOLVE
```

Initialization

The model is initialized with the maximum number of agents (i.e. up to *maximum population size*), each of which is created by the Create Agents submodel, to reflect a stable population at carrying capacity. At initialization, all agents are placed at random x-y *location*s in continuous space and into empty familiarity and interaction networks. Each agent's *age* is randomized by drawing uniformly from 0 to its *maximum age*, and its *accumulated payoff* is randomized by drawing uniformly from agents' *starting payoff* at birth to the *reproduction threshold*, or the maximum amount of payoff required for reproduction, to reflect a typical state of a population with overlapping generations and minimize initialization effects. To further decrease the effects of initial conditions, there is a burn-in period of 1000 steps, or approximately 2 generations, in which there is no mutation, to allow aggregations and familiarity to form before cooperative strategy is given the chance to evolve - or go extinct.

Algorithm 2. Initialize. Create a population of the maximum number of agents with randomly drawn ages and accumulated payoff.

```
for 0 to maximum population size do
    CREATE AGENT a
    if the number of agents created so far < initial proportion of cooperators*maximum population size then
        Set agent a's cooperative strategy to Cooperator
    else
        Set agent a's cooperative strategy to Defector
    end if
    Set agent a's age ~ U(0, a's maximum lifespan)
```

```
    Set agent $a$'s accumulated payoff ~ U(starting payoff, reproduction threshold)
end for
```

Submodels

Parameters

Name	Meaning	Default Value (and Units)	Range Analyzed	Source
Initialization				
Maximum population size	Initial and maximum number of agents.	1000 agents	1000	
Space dimensions	Length and width of the space.	90 spatial units	50-100	
Create Agent				
Starting payoff	Amount of accumulated payoff all new agents are initialized with.	0 payoff units	0-1	
Mean lifespan	Mean for drawing agents' maximum lifespans from a Gamma distribution.	500 steps	100-5000	
Standard deviation in lifespan	Standard deviation for drawing agents' maximum lifespans from a Gamma distribution.	.1	0-.9	
Mode familiar bias	Mode for drawing agents' familiar bias from a Beta distribution.	.9	.5-1	
Mode familiarity decay	Mode for drawing agents' familiarity decay from a Beta distribution.	.1 units of acquaintance	0-.7	
Mean familiarity	Mean for drawing agents' familiarity threshold from a	5 units of acquaintance	1-100	

threshold	Gamma distribution.			
Mean memory capacity	Mean for drawing agents' memory capacities from a Gamma distribution.	5 agents	1-100	
Variation in familiarity parameters	Variance for drawing agents' familiarity threshold and memory capacity from a Gamma distribution, and "sample size" for drawing agents' familiarity bias and familiarity decay from a Beta distribution.	.1	0-1	
Interact				
Interaction radius	Distance at which an agent looks for other agents to interact with.	2 spatial units	1-50	Joshi et al., 2017
Benefit of cooperation	Benefit provided by cooperators in the prisoner's dilemma.	.5 payoff units	.001-1	
Cost:benefit ratio of cooperation	Proportion of the benefit of cooperation paid by cooperators as a cost in the prisoner's dilemma.	.5	.01-1	
Move				
Repulsion radius	Distance at which an agent moves away from its neighbors.	1 spatial unit	1-6	Joshi et al., 2017
Aggregation radius	Distance at which an agent moves toward and aligns with its neighbors.	6 spatial units	2-6	Joshi et al., 2017
Flocking weight	Relative weight of aligning with other agents as opposed to moving toward them.	.5	0-1	Joshi et al., 2017
Persistence weight	Relative weight of continuing in the same direction as opposed to aggregating with others.	.2	0-1	Joshi et al., 2017
Movement error	Standard deviation of random error in movement.	.05	0-1	Joshi et al., 2017

Maximum rotation	Maximum change in direction of movement on a single step.	50 degrees	25-360	Joshi et al., 2017
Step size	Distance moved on each timestep.	.2 spatial units	.05-2	Joshi et al., 2017
Evolve				
External payoff	Additional payoff gained/lost by all agents on each step irrespective of interaction.	-.05 payoff units	-.1-1	Andras et al., 2007
Reproduction threshold	Accumulated payoff required to reproduce.	1 payoff unit	1 (Sets the scale)	
Reproduction cost	Payoff lost when an agent reproduces.	1 payoff unit	0-1	
Mutation rate	Probability of having offspring of the opposite cooperative strategy from their parent.	.01	0-.5	

Table 2. All model parameters, their meanings, baseline values and value ranges used in simulation experiments, and relevant citation (if applicable).

Create Agent

New agents are created at a random *location* in the space, with a pair of x-y coordinates drawn uniformly from 0 to the *space size*, and a random *direction of movement* drawn uniformly from 0 to 360 degrees. They are initialized with no *interacted agents* or *familiar agents*. Their *maximum lifespan* is drawn from a Gamma distribution (Lai, 2010) with the simulation *mean lifespan* and *standard deviation in lifespan*, and a minimum of 1. The agent's *accumulated resources* are initialized at the simulation *starting resources*. The agent's *familiar bias* and *familiarity decay* are drawn from Beta distributions, as they both range from 0 to 1 (making more common unbounded distributions, such as the normal distribution, ill-suited), with the simulation *mode familiar bias* and *mode familiarity decay* respectively and "sample size" (Kruschke, 2010) derived from the Model *variation in familiarity parameters* (Equation 1) to allow for a range of variance from none when the *variation in familiarity parameters* is 0, to a nearly uniform distribution when it is equal to 1. This results in and parameters

$$(1) \qquad \alpha = \frac{mode}{(\frac{3}{4}var)^2} \qquad \beta = \frac{1 - mode}{(\frac{3}{4}var)^2}$$

The agent’s *familiarity threshold* and *memory capacity* are drawn from Gamma distributions with the simulation *mean familiarity threshold* and *mean memory capacity* respectively, and standard deviation equal to the simulation *variation in familiarity parameters*. The agent’s *cooperative strategy* is provided, either in model initialization (see Initialization) or by the agent’s parent (see Reproduce).

Algorithm 3. Create Agent. When a new agent is created, it is given randomly drawn familiarity parameters and lifespan, an age of 0, the simulation starting payoff, and a randomly drawn location and direction of movement.

```
Set lifespan ~ Gamma(mean lifespan, standard deviation in lifespan)
Set familiar bias ~ Beta(mode familiar bias, ¾ * variation in familiarity parameters)
Set memory capacity ~ Gamma(mean memory, variation in familiarity parameters)
Set familiarity threshold ~ Gamma(mean familiarity threshold, variation in familiarity parameters)
Set familiarity decay ~ Beta(mode familiarity decay, ¾ * variation in familiarity parameters)
```

```
Initialize age = 0
Initialize accumulated payoff = starting payoff
Initialize location = x-y coordinates ~ U(0, space size)
Initialize direction of movement ~ U(0, 360)
```

Interact

If any other agents are within *interaction radius* units of this agent's *location*, then one of those agents is chosen at random to play it in a prisoner's dilemma to test the most stringent case for the evolution of cooperation. If an agent is a Cooperator, then its *accumulated payoff* is decremented by the *benefit of cooperation* times the *cost:benefit ratio of cooperation* (*c*) and its partner's *accumulated payoff* is incremented by the *benefit of cooperation* (*b*), while defectors pay no cost and provide their partners with no benefit (Table).

		Player 2	
		Cooperate	*Defect*
Player 1	*Cooperate*	$(b-c,\ b-c)$	$(-c, b)$
	Defect	$(b, -c)$	$(0, 0)$

Algorithm 4. Interact. Agents play a prisoner's dilemma and gain or lose resources accordingly.

```
Given a partner agent randomly chosen from within this agent's interaction radius
if this agent's cooperative strategy is Cooperator then
```

```
    Decrement this agent's accumulated payoff by the cost of cooperation
    Increment its partner's accumulated payoff by the benefit of cooperation
end if
if partner's cooperative strategy is Cooperator then
    Decrement partner's accumulated payoff by the cost of cooperation
    Increment this agent's accumulated payoff by the benefit of cooperation
end if
```

Update Familiarity

$F_{i,j}$ = 1 if agent i considers agent j to be familiar, and 0 otherwise

$I_{i,j}$ = agent i's degree of acquaintance with agent j

h_i = the *familiarity threshold* of agent i

m_i = the *memory capacity* of agent i

Whenever a pair of agents interacts, they each increment their degree of acquaintance with each other by one (see Equation 8 for the net change in degree of acquaintance on each step) and then decide whether to consider the other to be familiar. If an agent's partner is not in its *interacted agents*, then the agent creates an edge to its partner with an acquaintance of 1, otherwise, their acquaintance is just incremented.

In addition, if an agent does not consider its partner to be familiar and their degree of acquaintance ($I_{i,j}$) is greater than the agent's *familiarity threshold* (h_i), then it can become familiar

($F_{i,j} = 1$). If the agent's total number of *familiar agents* ($\sum F_i$) is lower than its *memory capacity* (m_i), it will now recognize its partner as familiar. Alternatively, if the agent's memory is at capacity, but the agent's acquaintance with its partner is higher than its lowest acquaintance with a familiar agent (i.e., $MIN(I_{i,k}F_{i,k})$) the partner will become familiar, replacing the lowest acquaintance familiar agent.

$$(2) \qquad F_{i,j} = \begin{cases} 1, & \textbf{if } \; I_{i,j} > h_i \textbf{ and } (\sum F_i < m_i \textbf{ or } MIN(I_{i,k}F_{i,k}) < I_{i,j}) \\ 0, & \textbf{otherwise} \end{cases}$$

Algorithm 5. Update Familiarity. When two agents interact, their degree of acquaintance increases and they may recognize each other as familiar.

```
Given an interaction partner agent
if partner is not in this agent's interacted agents then
    Add partner to interacted agents
    Set level of acquaintance to partner = 1
else
    Increment level of acquaintance with partner by 1
end if
if partner is not in this agent's familiar agents then
    if level of acquaintance with partner > this agent's familiarity threshold then
        if this agent's number of familiar agents < this agent's memory capacity then
```

```
            Add partner to this agent's familiar agents
        else if level of acquaintance with partner > MIN(familiar acquaintance) then
            Remove MIN(familiar acquaintance) from familiar agents
            Add partner to familiar agents
        end if
    end if
end if
```

Move

d_i = the *direction of movement* of agent i

l_i = the *location* of agent i

R_i = array of agents within the *repulsion radius* of agent i

A_i = array of agents within the *aggregation radius* of agent i

w_f = *flocking weight*

w_p = *persistence weight*

s = *step size*

e = random error in movement ~ N(0, *movement error*)

b_i = the *familiar bias* of agent i

$F_{i,j}$ = 1 if agent i considers agent j to be familiar, and 0 otherwise

On each step, each agent (*i*) updates its *location* (l_i) and *direction of movement* (d_i) according to the active movement model of Joshi et al. (2017), modified to take into account familiarity between agents ($F_{i,j}$). If there are any other agents (R_i) within *repulsion radius* units of the active agent's *location*, then it moves in the opposite direction of their average location (Equation 3). Repulsion takes precedence over all other movement decisions.

$$r_i = \frac{1}{|R_i|} \sum_{j}^{R_i} \frac{l_i - l_j}{\|l_i - l_j\|} \tag{3}$$

Otherwise, if there are any other agents (A_i) within *aggregation radius* units of the active agent's *location*, then the agent moves to align with (Equation 4)

$$f_i = \frac{1}{|A_i|} \sum_{j}^{A_i} d_j b_i^{F_{i,j}} (1 - b_i)^{1-F_{i,j}} \tag{4}$$

and move toward the other agents (Equation 5),

$$o_i = \frac{1}{|A_i|} \sum_{j}^{A_i} \frac{l_j - l_i}{\|l_j - l_i\|} b_i^{F_{i,j}} (1 - b_i)^{1-F_{i,j}} \tag{5}$$

weighted by the *flocking weight* (w_f) and the agent's *familiar bias* (b_i), with some tendency to continue moving in the same direction, according to the *persistence weight* (w_p; Equation 6).

$$a_i = w_f(1 - w_p)f_i + (1 - w_f)(1 - w_p)o_i + w_p d_i \tag{6}$$

Otherwise, if there are no other agents present, then the agent continues in its current *direction*. In all cases, the agent's movement is adjusted by some random error (*e*), drawn from a normal distribution, with a mean of 0 and a standard deviation of *movement error* (Equation 7).

$$d_i = \epsilon + \begin{cases} r_i, & \textbf{if } |R_i| > 0 \\ a_i, & \textbf{if } |A_i| > 0 \\ d_i, & \textbf{otherwise} \end{cases} \quad (7)$$

All movement is constrained by the *maximum rotation*. If the difference between the agent’s current *direction of movement* and the new calculated direction is greater than *maximum rotation*, its *direction of movement* is only incremented by the *maximum rotation* in that direction.

$$(8) \qquad d_{i,t+1} = d_{i,t} + \mathrm{MIN}(|d_{i,t+1} - d_{i,t}|, \mathit{max\ rotation})$$

The agent will then move *step size* units in the calculated direction.

Algorithm 6. Move. Each step, each agent moves according to the active movement model of Joshi et al. (2017). If there are any other agents within its repulsion radius, the agent moves away from their average location. Otherwise, if there are any agents within its aggregation radius, the agent moves toward them and aligns with their direction of movement. Otherwise, the agent persists in moving at its previous trajectory with some random error.

```
if this agent has any neighbors within repulsion radius units of its location then
    Set new direction = average direction away from neighbors
else if this agent has any neighbors within aggregation radius units of its location then
    for each neighbor do
        if the neighbor is in this agent’s familiar agents then
```

```
            Set weight = this agent's familiar bias
        else
            Set weight = 1 - this agent's familiar bias
        end if
        Increment flocking direction by neighbor's direction of movement * weight
        Increment aggregation direction by direction toward neighbor's location * weight
    end for
    Set new direction = persistence weight * this agent's direction of movement + (1-persistence weight)*(flocking weight * flocking direction + (1-flocking weight) * aggregation direction) + random error
else
    Set new direction = current direction of movement + random error
end if
if ABS(new direction - current direction of movement) > maximum rotation then
    Set this agent's direction of movement = current direction of movement + maximum rotation in the desired direction
else
    Set this agent's direction of movement = new direction
end if
```

Prune Network

On each step, each agent (i) goes through its *interacted agents* (I_i) and decrements its degree of acquaintance with each of the other agents (j) by its *familiarity decay*, resulting in a net change on each step as described by Equation 8.

$$\Delta I_{i,j} = \begin{cases} 1 - decay_i, & \textbf{if } i \text{ and } j \text{ interacted on this step} \\ -decay_i, & \textbf{otherwise} \end{cases} \tag{9}$$

If any acquaintance drops below 0, the agent is removed from *interacted agents*, and if the agent was in *familiar agents*, it is now removed as well and no longer considered to be familiar.

Algorithm 7. Prune Connections. On each step, each agent goes through its interacted agents, decrements their degree of acquaintance by its decay rate, and removes all agents whose acquaintance drops below 0 from its interacted agents and familiar agents.

```
for each connection in this agent's interacted agents do
    Decrement the degree of acquaintance of the connection by this agent's decay rate
    if the degree of acquaintance of the connection < 0 then
        Remove the connection from this agent's interacted agents
        if the connection is also in this agent's familiar agents then
            Remove the connection from this agent's familiar agents
        end if
    end if
end for
```

Evolve

On each step, all agents' *accumulated payoff* is incremented or decremented by the *exogenous payoff* (depending on whether it is positive or negative), irrespective of whether the agent played a prisoner's dilemma on that step.

All agents' *age*s are also incremented by 1.

Reproduction: If an agent's total *accumulated payoff* is greater than the *reproduction threshold*, it reproduces and the *reproduction cost* is subtracted from its *accumulated payoff*. If the total number of agents in the population at that timestep is less than the *maximum population size*, then a new agent is created with the same *cooperative strategy* as its parent, with a probability of switching to the other strategy equal to the *mutation rate*. Otherwise, the offspring is assumed to have died due to high juvenile mortality, and no agent is created. This is analogous to a Moran process (Moran, 1962), but allows for the population size to vary below a maximum carrying capacity.

Death: If an agent's *age* is greater than its *maximum lifespan*, then it is removed from the population. Additionally, if the agent's *accumulated payoff* is less 0, then it is removed from the population irrespective of its *age*, representing death from hunger or chance predation. Such a dynamic has been found to promote the evolution of cooperation (Smaldino et al., 2013).

Algorithm 8. Evolve. On every step, each agent's accumulated payoff is adjusted by the exogenous payoff and its age is incremented by 1. If its accumulated payoff reaches the reproduction threshold, it pays the reproduction cost and, if the population is below the maximum population size, an offspring agent is created with the same cooperative strategy with a small chance of mutation. If the agent’s age is greater than its maximum lifespan or its accumulated payoff is less than 0, it dies, and is removed from the simulation.

```
Add the exogenous payoff to this agent’s accumulated payoff
Increment this agent’s age by 1
if this agent’s accumulated payoff > reproduction threshold then
    Decrement this agent’s accumulated payoff by the reproduction cost
    if the total number of agents in the simulation < maximum population size then
        CREATE AGENT
        if a draw ~ U(0,1) > the mutation rate then
            Set the new agent’s cooperative strategy = this agent’s cooperative strategy
        else
            Set the new agent’s cooperative strategy = the other cooperative strategy
        end if
    end if
end if
if this agent’s age > this agent’s maximum lifespan or this agent’s accumulated payoff < 0
then
    Remove this agent from the population
end if
```

Simulation Experiments

To determine the effect of a preference for aggregating with familiar individuals on the evolution of cooperation, we compared a no-preference condition, in which agents exhibited no preference for familiar individuals (mode b_i = .5), to a preference condition, in which agents weighted familiar individuals 9 times more strongly than unfamiliar individuals when deciding which direction to move in (mode b_i = .9), across a variety of selection and movement conditions. For all simulation experiments, we evaluated the number of cooperative agents and the number of defector agents present at the end of 51,000 steps (with a burn-in period of 1,000 steps without any mutation to minimize the effects of initial conditions), or approximately 100 generations, for ten replicate populations under each condition.

First, we performed a broad sweep of the parameter space to find the regions in which a preference for aggregating with familiar individuals had a noticeable effect on the evolution of cooperation. We particularly focused on exogenous payoff, which determined the strength of selection, as well as the population density and aggregation radius, which we expected to be the key parameters in determining the population structure in which familiarity formed (Joshi et al., 2017). We evaluated whether a population of cooperators could resist invasion by defectors when both cooperators and defectors did and did not have a preference for aggregating with familiar individuals (familiar bias of .5 versus .9) across a range of exogenous payoffs (from -0.25 to 0.5), population densities (space size of 50 to 500 units), and aggregation radii (2 to 6).

Once we identified regions of interest in this parameter space, we examined the change in number of cooperators and defectors in the population in each simulation under representative conditions, with and without a preference for aggregating with familiar individuals (.5 versus .9). We also evaluated whether a preference for aggregating with familiar individuals might enable cooperators to invade a population of defectors - a more difficult challenge for the evolution of cooperation - by also evaluating the change in number of cooperators and defectors in populations under representative conditions that started with all defectors.

We additionally performed stability analyses to examine the effect of each of the other parameters on how well a preference for aggregating with familiar individuals was able to maintain the stability of cooperation. Due to the large number of parameters in the model, we were unable to evaluate all combinations of all parameters, so instead we varied each parameter around an ecologically plausible baseline condition. We selected values for movement parameters after Joshi et al. (2017) and chose reasonable intermediate values for the costs and benefits of cooperation and life history parameters (see Table 2 for all baseline values and parameter ranges evaluated).

Results

The Effect of a Bias in Favor of Associating with Familiar Individuals

A preference for associating with familiar individuals enabled cooperation to resist invasion by defectors under some conditions where cooperation would otherwise be overwhelmed by defection (Figure 2).

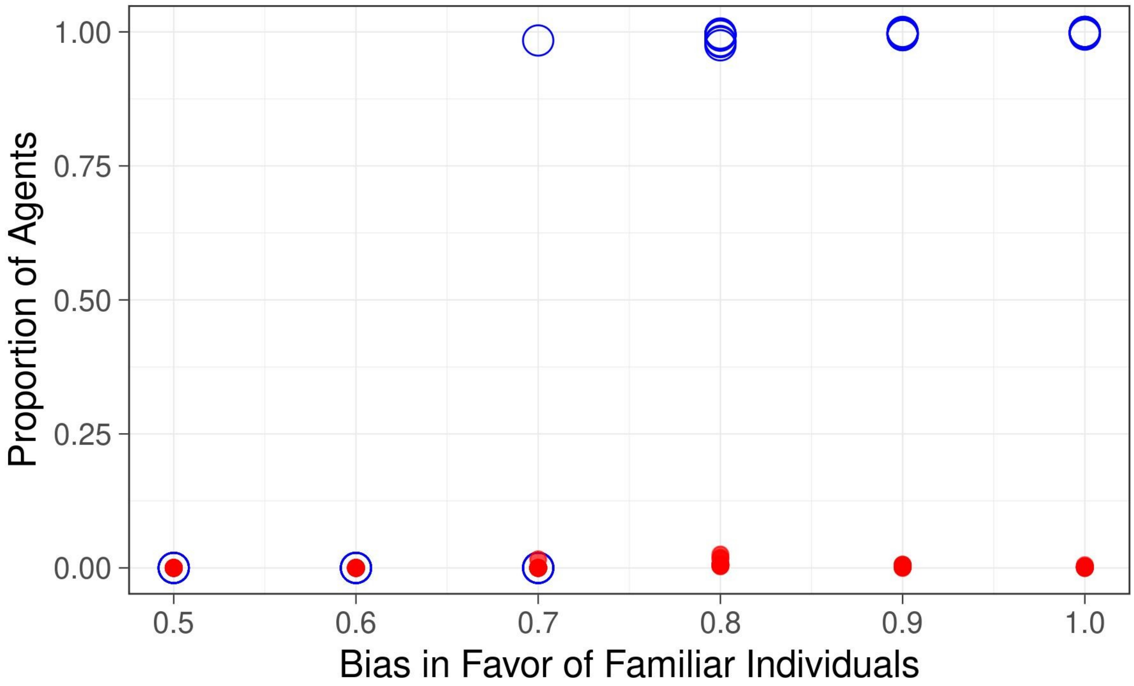


Figure 2. Effect of familiarity bias on proportion of cooperators (blue circles) and defectors (red dots) surviving after 50,000 steps in 10 replicates over a range of degree of preference for familiar individuals, from aggregating with familiar and unfamiliar individuals equally (0.5) to only aggregating with familiar individuals and not unfamiliar individuals (1) under a representative condition where cooperation resisted invasion in populations with a preference, but was invaded in the absence of a preference (density = 0.123; exogenous payoff = -0.05; aggregation radius = 6).

Under this condition, as the strength of agents' preference for associating with familiar individuals increased, the population of cooperators stabilized (Figure 3). Even a slight preference for familiar individuals (0.6) allowed some populations to persist for nearly 50,000 timesteps (Figure 3B), a sizable increase from a maximum of around 30,000 timesteps in the absence of any preference (Figure 3A). With a somewhat stronger preference (0.7), some populations persisted for the entire 50,000 timesteps with only a few major fluctuations (Figure 3C). When the bias in favor of familiar individuals was set to 0.8, all populations persisted for 50,000 timesteps with only occasional fluctuations (Figure 3D). Particularly striking was that when agents had a strong preference for associating with familiar individuals (0.9-1), there were no fluctuations at all (Figure 3E-F); that is, defectors were unable to get any foothold in the population, and any invasion was cut off before it began.

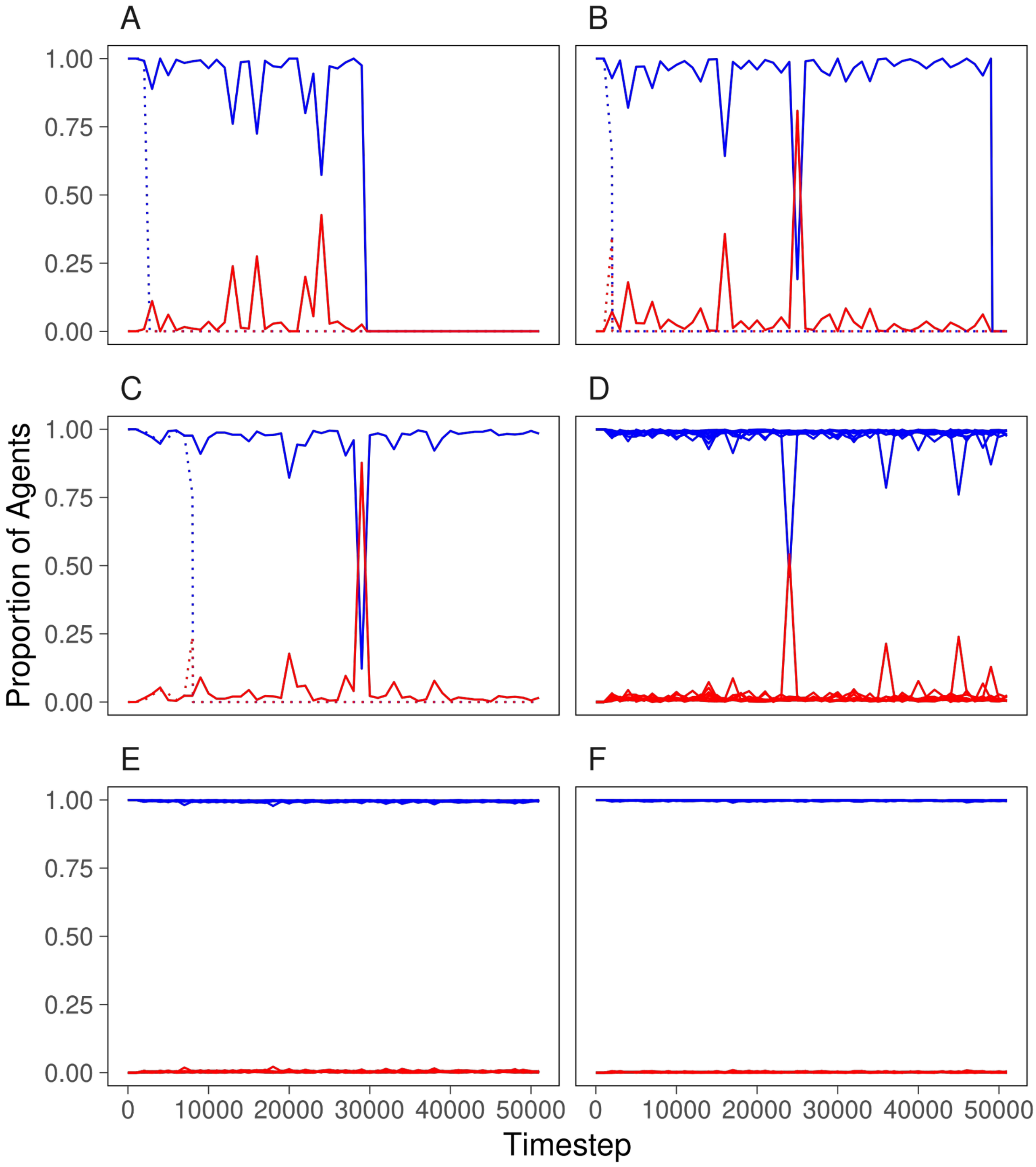


Figure 3. Proportion of cooperators (blue) and defectors (red) over 51,000 timesteps with (*A*) no preference for associating with familiar individuals (0.5), from the replicate that went extinct in

the shortest time (dotted) and the replicate that went extinct in the longest time (solid); (*B*) a weak preference (0.6), from the replicates that went extinct in the shortest (dotted) and the longest (solid) time; (*C*) an intermediate preference (0.7), from the replicate that went extinct in the shortest time (dotted) and the only replicate that lasted all 51,000 timesteps (solid); (*D*) an intermediate preference (0.8), from all 10 replicates; (*E*) a strong preference (0.9), from all 10 replicates; and (*F*) a complete preference (1), from all 10 replicates, under a representative condition where cooperation resisted invasion in populations with a preference, but was invaded in the absence of a preference (density = 0.123; exogenous payoff = -0.05; aggregation radius = 6).

Note that in the condition that has been highlighted thus far, each agent received a negative exogenous payoff (-0.05) on each step, meaning that the only way agents could survive and reproduce was by interacting with cooperators, and that populations of only defectors would inevitably go extinct. However, a preference for aggregating with familiar individuals also stabilized cooperation under some conditions where defectors would otherwise have been able to survive and reach fixation, resulting in highly stable populations almost entirely composed of cooperators (Figure 4).

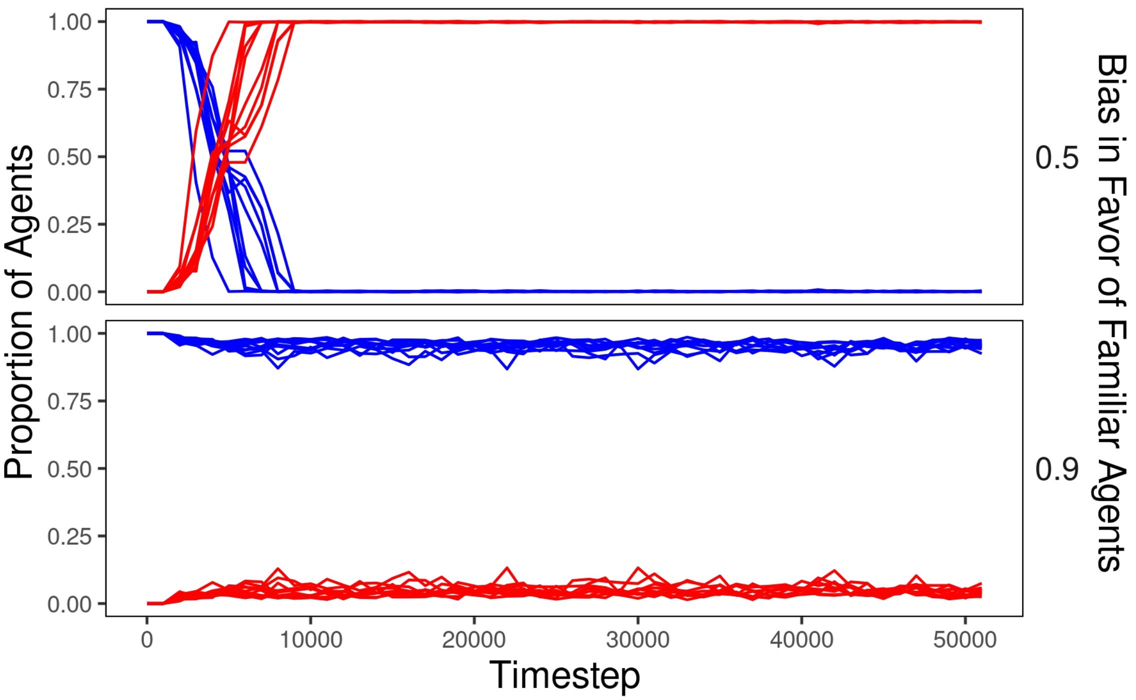


Figure 4. Effect of familiarity bias on proportion of cooperators (blue) and defectors (red) over the course of 50,000 steps in 10 replicates with no preference for familiar individuals (0.5; top) and a strong preference for aggregating with familiar individuals (0.9; bottom) under a representative condition where cooperation resisted invasion in populations with a preference, but was invaded in the absence of a preference (density = 0.025, exogenous payoff = 0.025; aggregation radius = 6).

This pattern is typical of the behavior of the model, however, under a few conditions, evolution favored a stable mix of cooperators and defectors, and a preference for associating with familiar individuals was able to push the mixed equilibrium toward a fully cooperative one (Figure 5).

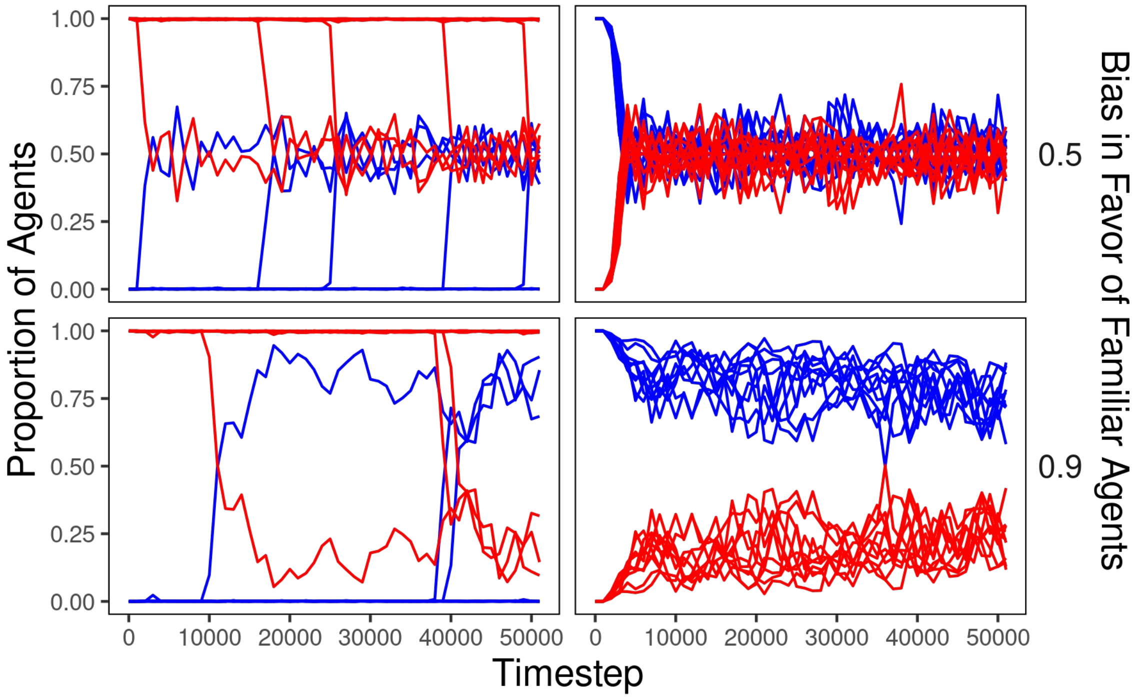


Figure 5. Number of cooperators (blue) and defectors (red) over the course of 50,000 steps in 10 replicates with no preference for familiar individuals (0.5; top) and a strong preference for aggregating with familiar individuals (0.9; bottom); starting with all defectors (left) or all cooperators (right); under a boundary condition where cooperation persisted at intermediate levels (density = 0.044, exogenous payoff = 0.005; aggregation radius = 4).

Interaction with Exogenous Payoff and Population Density

A bias in favor of associating with familiar individuals was not the only factor which determined whether cooperation was stable. We observed that the population density had a particularly dramatic effect, with cooperation persisting at a much higher rate at lower density, particularly as the aggregation radius increased (Figure 8).

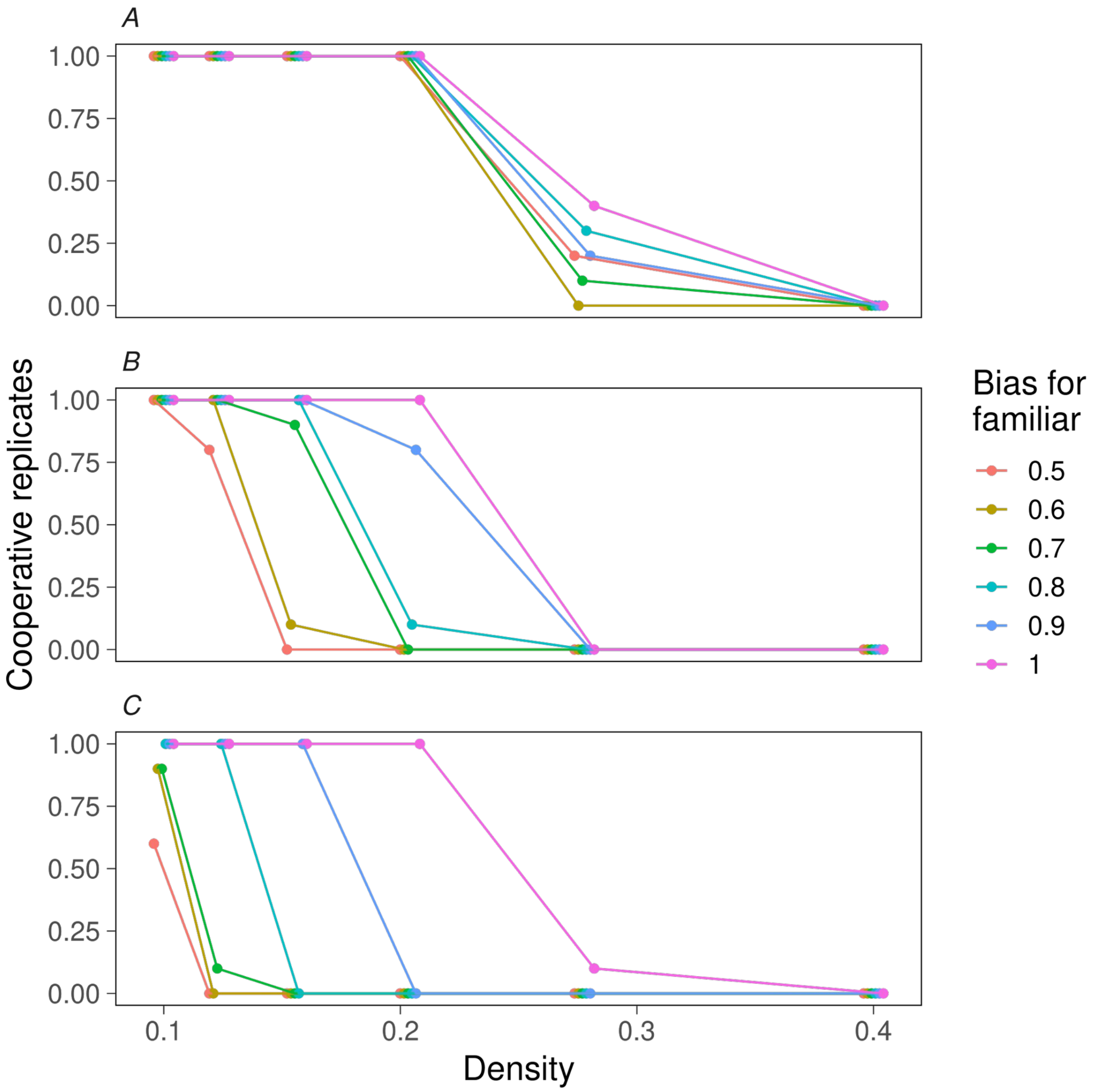


Figure 8. Proportion out of 10 replicates with at least 90% of cooperators remaining at the end of 50,000 timesteps, over a range of population densities (in agents per unit of space) and a range of strength of bias in favor of aggregating with familiar individuals, from no preference for associating with familiar or unfamiliar individuals (.5; red) to a complete preference where

agents only associated with familiar individuals and did not attend to unfamiliar individuals (1; purple), with an aggregation radius of (A) 2, (B) 4, and (C) 6.

Even more striking was the interaction between population density and exogenous payoff. We found that a preference for associating with familiar individuals tipped the scales in favor of cooperation at the boundary between cooperation being evolutionarily stable and defection successfully invading, which occurred at high density with very low (negative) exogenous payoff, and at low density with high (positive) exogenous payoff. The wider the aggregation radius, the more strongly this effect was felt (Figure 9).

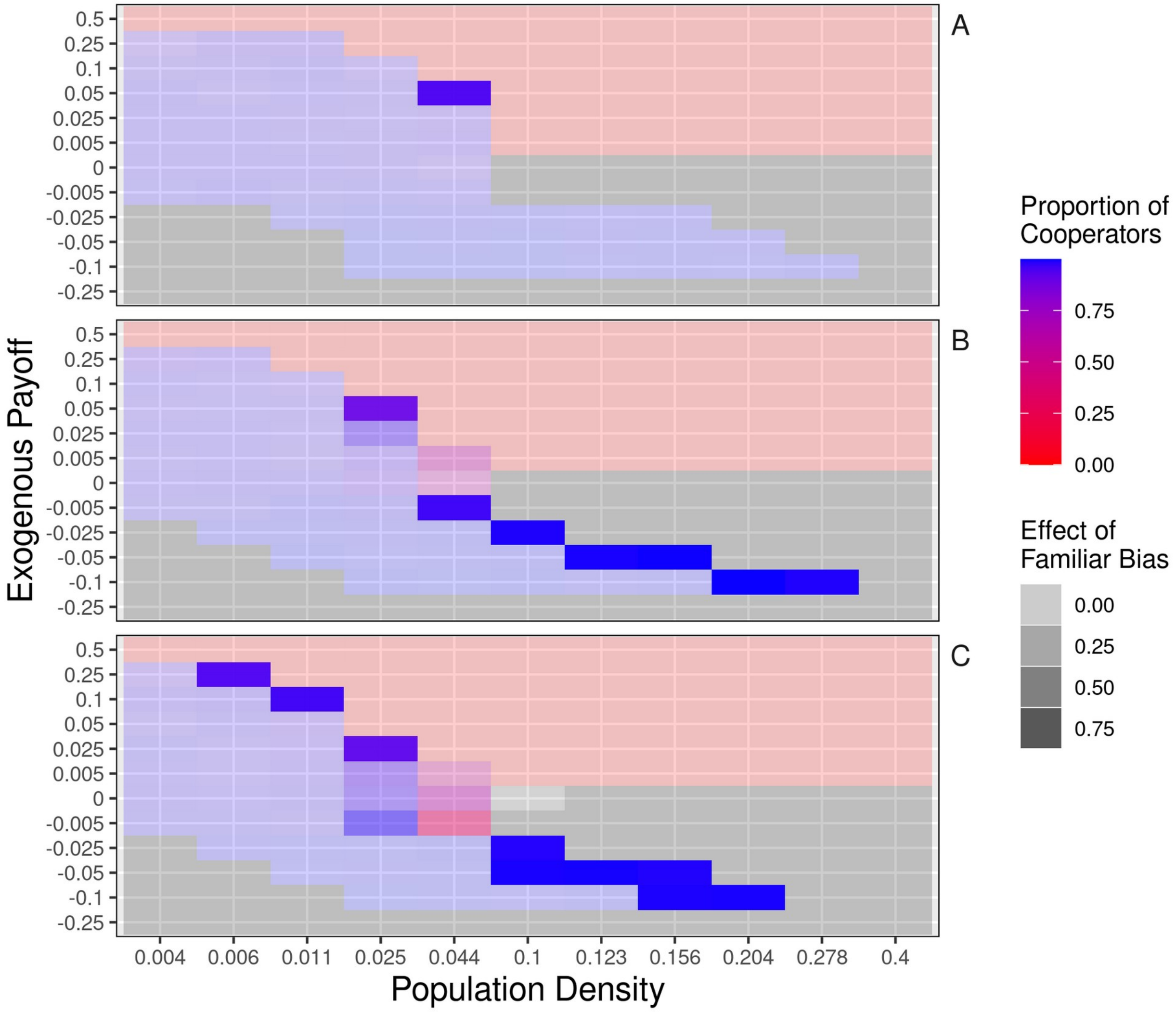


Figure 9. The minimum surviving proportion of cooperators after 50,000 steps over 10 replicates, in populations starting with all cooperators, over a range of population densities, exogenous payoffs with an aggregation radius of (A) 2, (B) 4, and (C) 6. Blue areas represent cooperators completely excluding defectors (100% cooperators), red areas represent populations where defectors completely invaded (0% cooperators), and black areas represent populations that went completely extinct (populations of defectors could not survive with 0 or negative exogenous payoff). The opacity of each square reflects the difference in surviving proportion of cooperators

between conditions with a preference and without a preference for familiar individuals. Bright blue squares represent conditions where cooperators completely excluded defectors in populations with a preference for familiar individuals, but where defection completely invaded in populations with no preference (see Figure 4; the reverse never occurred). Intermediate colors (purple) and degrees of opacity (e.g. population density 0.025 to 0.044 and baseline fitness -0.005 to 0.005) reflect conditions where a stable mix of cooperators and defectors is possible (as depicted in Figure 5).

These results suggest that a preference for associating with familiar individuals was able to favor cooperation by making it harder for defectors to find new cooperators to exploit, effectively isolating defectors. This is particularly evident in conditions with a negative exogenous payoff, where agents could only survive by receiving the benefits of cooperation. Therefore, when defectors invaded, they very quickly drove the cooperators and therefore themselves to extinction, causing the population as a whole to go extinct. Sometimes a small group of cooperators remained to bring the population back from the brink, but, typically, these cycles of population crashes followed by rebounds were unsustainable and eventually drove the population to extinction (Figure 6).

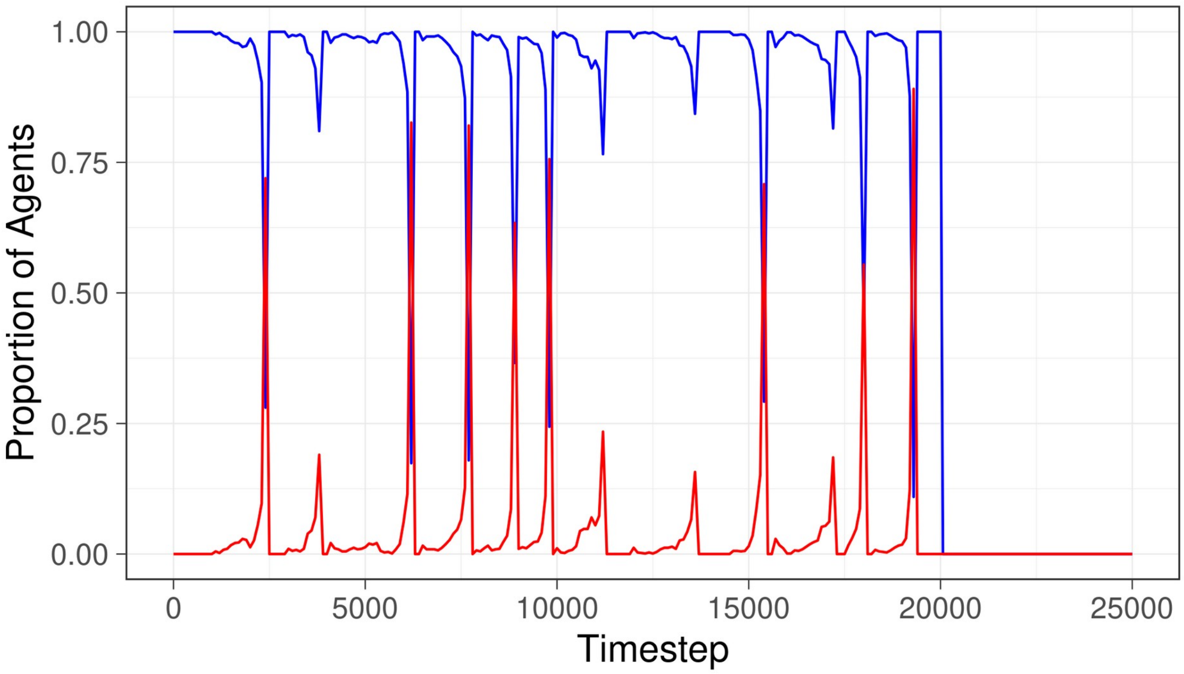


Figure 6. Proportion of cooperators (blue) and defectors (red), over 25,000 timesteps from one representative replicate with no preference for familiar individuals (0.5) where cooperation could not resist invasion in the absence of a preference (density = 0.123; exogenous payoff = -0.05; aggregation radius = 6)

In general, when a defector invaded, it overexploited its cooperative neighbors, eventually driving them to their deaths. A preference for associating with familiar individuals made it harder for the defector to find other cooperators willing to interact with it, leaving the defector with no one to exploit, decreasing its own chances of survival and reproduction (Figure 7). A bias in favor of associating with familiar individuals was also costly for those cooperators who interacted with a defector and were therefore more likely to interact with it again. However, low population densities or highly negative exogenous payoffs shielded the majority of cooperators from exploitation.

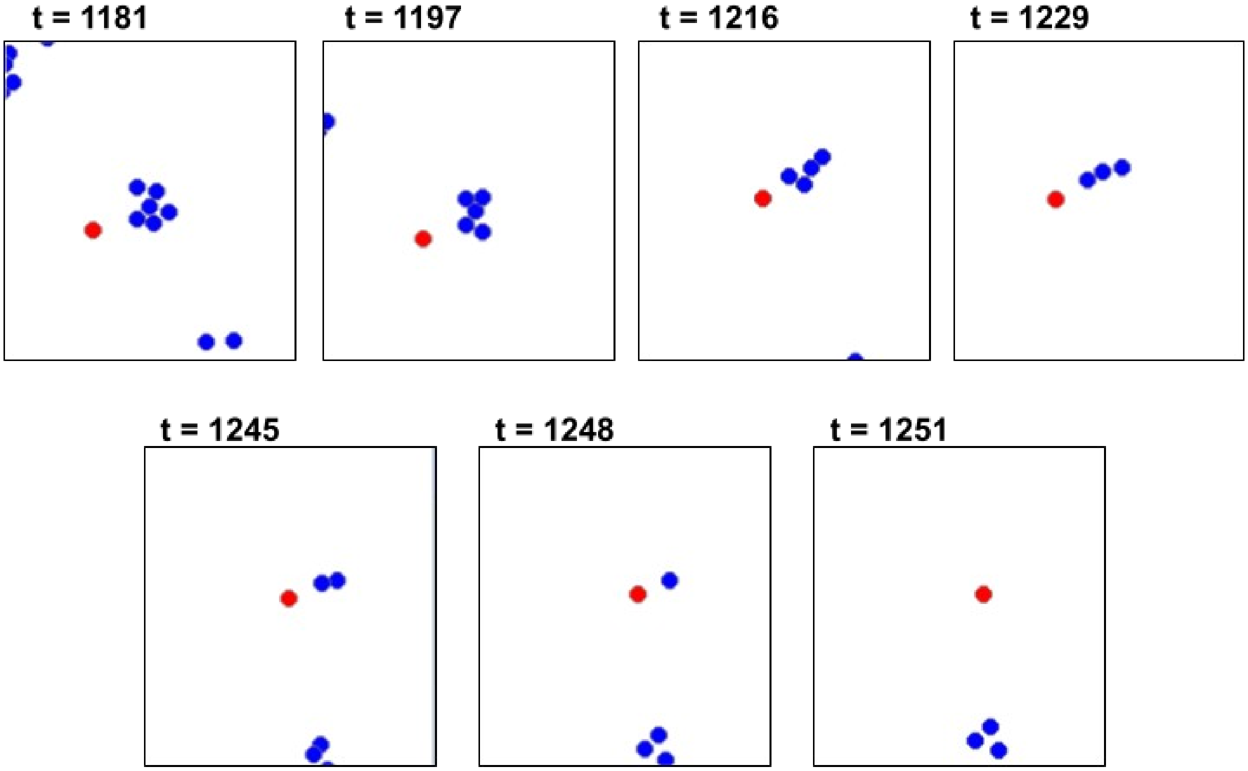


Figure 7. Visualization of agents' locations in a small region of the simulated space from timestep 1181 through timestep 1251, with cooperators in blue and defectors in red, under a representative condition (density = 0.025, baseline fitness = 0.025; aggregation radius = 6, bias for familiar individuals = .9).

Stability Analyses

To evaluate the stability of the results to the other model parameters, we varied the values of each parameter around a baseline condition in which cooperation persists in populations where agents have a preference for aggregating with familiar individuals, but where it is invaded by defection when agents do not have a preference. (See Table 2 for baseline and ranges of parameter values.)

Cooperation Parameters

Surprisingly, our results were highly robust to variation in the cost of cooperation; when agents preferred to associate with familiar individuals, cooperation persisted with a cost:benefit ratio of cooperation as high as 0.8. In contrast, in the absence of a preference for aggregating with familiar individuals, cooperation was only stable with an extremely low cost:benefit ratio (0.01; Figure 5 B). The benefit provided by cooperators had only a moderate effect; as long as the benefit was high enough for agents to accumulate enough payoff for reproduction, a preference for aggregating with familiar individuals enabled cooperation to be evolutionarily stable (Figure 5 A). As intuitively expected, cooperation quickly collapsed as the radius of interaction increased, making the population increasingly well mixed (Figure 5 C).

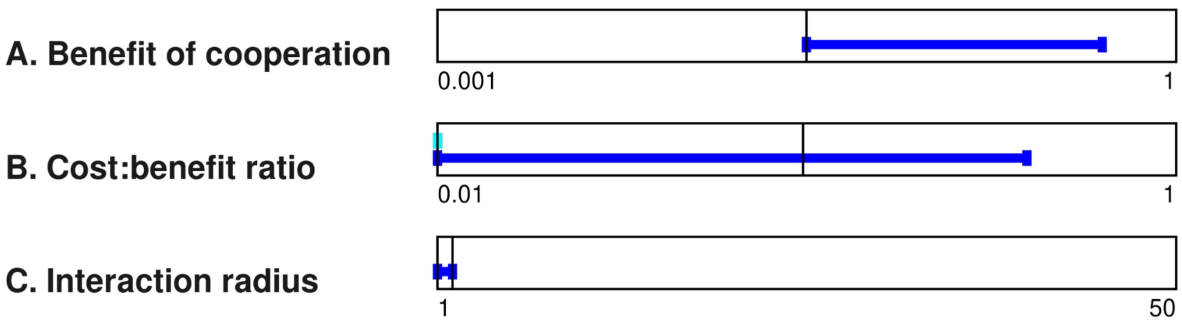


Figure 5. The range of values of each cooperation parameter, in simulations with a strong preference for aggregating with familiar individuals (0.9; dark blue) and no preference (0.5; light blue), under which cooperation is able to persist (i.e. there were at least 900 cooperative agents remaining after 51,000 steps in all 10 replicates starting with all cooperators). Where no light blue line appears, cooperation was not evolutionarily stable in the absence of a preference for aggregating with familiar individuals for any value of that parameter under baseline conditions. The thin vertical line represents the baseline value of each parameter used when varying all other parameters.

Movement Parameters

A preference for aggregating with familiar individuals generally made cooperation more robust to a wider range of movement strategies (Figure 6). In particular, a preference for aggregating with familiar individuals appears to allow cooperation to persist even in denser populations (i.e. with smaller repulsion radii; Figure 6 A) that have a stronger tendency to aggregate as opposed to flocking (Figure 6 C), persisting in the same direction (Figure 6 D), or moving randomly (Figure 6 B). A preference for aggregating with familiar individuals also made cooperation more robust to increased velocity, which typically results in a more well-mixed population (Figure 6 F). However, even with a preference for aggregating with familiar individuals, cooperation was only able to persist for a small maximum angle of rotation (Figure 6 E), which limited agents’ movement, possibly preventing the population from becoming too well-mixed.

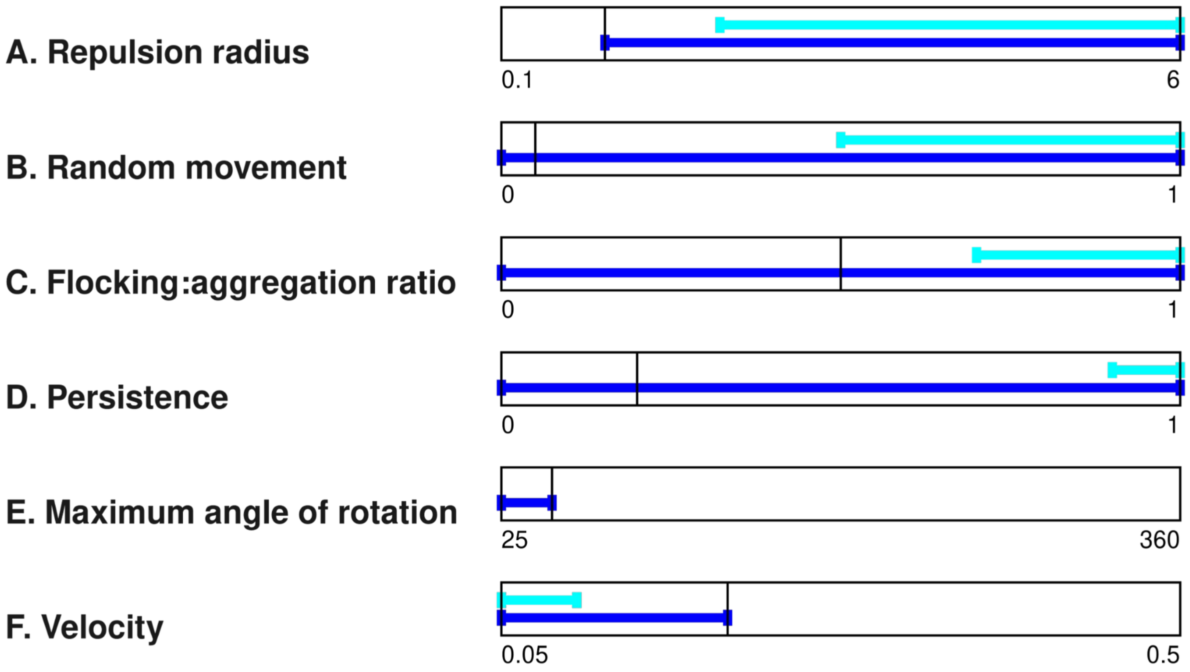


Figure 6. The range of values of each movement parameter, in simulations with a strong preference for aggregating with familiar individuals (0.9; dark blue) and no preference (0.5 light blue), under which cooperation is able to persist (i.e. there were at least 900 cooperative agents remaining after 51,000 steps in all 10 replicates starting with all cooperators). Where no light blue line appears, cooperation was not evolutionarily stable in the absence of a preference for aggregating with familiar individuals for any value of that parameter under baseline conditions.

The thin vertical line represents the baseline value of each parameter used when varying all other parameters.

Life History Parameters

Our results were generally robust to variation in life history traits (Figure 7). As expected, increasing the mutation rate, or the rate at which defectors were introduced, quickly drove the cooperators extinct, however a preference for aggregating with familiar individuals enabled cooperation to persist with higher mutation rates (Figure 7 A). We additionally found that even with a preference for aggregating with familiar individuals, under baseline conditions cooperation was only evolutionary stable when the cost of reproduction was equal to the reproduction threshold (Figure 7 B).

A preference for aggregating with familiar individuals also increased the robustness of cooperation to variation in the starting payoff for new agents (Figure 7 C). The higher the starting payoff, the more of a buffer new agents had against dying immediately if they found themselves isolated, without any cooperators to assist them, up to a starting payoff of 1, where agents were born ready to reproduce (and in doing so paid the cost of reproduction). We believe that a preference for aggregating with familiar individuals was able to buffer agents against early mortality risk because the population was divided into more, smaller groups, which covered the space more uniformly, enabling cooperators to find each other more easily, but groups of cooperators would still quickly die out if invaded by a defector.

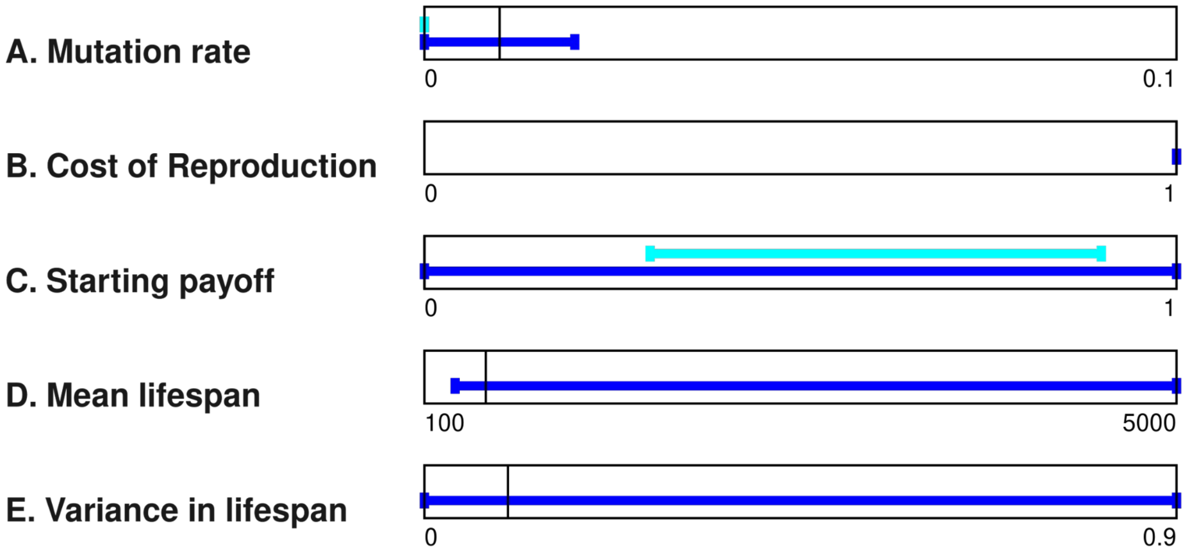


Figure 7. The range of values of each life history parameter, in simulations with a strong preference for aggregating with familiar individuals (0.9; dark blue) and no preference (0.5; light blue), under which cooperation is able to persist (i.e. there were at least 900 cooperative agents remaining after 51,000 steps in all 10 replicates starting with all cooperators). Where no light blue line appears, cooperation was not evolutionarily stable in the absence of a preference for aggregating with familiar individuals for any value of that parameter. The thin vertical line represents the baseline value of each parameter used when varying all other parameters. Note that the baseline value of starting payoff does not appear because it is 0 and the baseline cost of reproduction does not appear because it is 1.

Familiarity Parameters

We also found that our results were highly robust to the values used to parameterize how agents acquired and lost familiarity (Figure 8), with the requirement that the threshold for recognizing another individual as familiar needed to be low enough for agents to actually be able to become familiar based on repeated interaction (Figure 8 A).

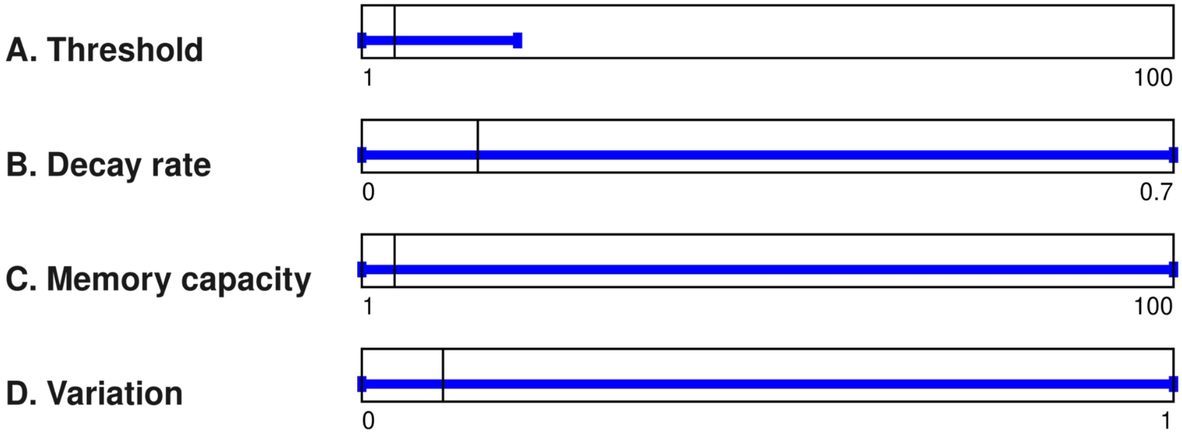


Figure 8. The range of values of each familiarity parameter, in simulations with a strong preference for aggregating with familiar individuals (0.9; dark blue) under which cooperation is able to persist (i.e. there were at least 900 cooperative agents remaining after 51,000 steps in all 10 replicates starting with all cooperators). Note that no light blue line appears because cooperation was not evolutionarily stable in the absence of a preference for aggregating with familiar individuals under baseline conditions for any value of any familiarity parameter. The thin vertical line represents the baseline value of each parameter used when varying all other parameters.

Discussion

Our model demonstrates that a preference for aggregating with familiar individuals can increase the range over which cooperation is an evolutionarily stable strategy. In particular, a preference for aggregating with familiar individuals is important at the boundary between stable cooperation and complete invasion by defectors, tipping the scales in favor of cooperation at high density under harsh environmental conditions, and at low density under more abundant environmental conditions. A preference for aggregating with familiar individuals can also favor the invasion of higher levels of cooperation in fission-fusion populations at boundary conditions with intermediate density and baseline fitness. Our results suggest that aggregating with familiar individuals promotes cooperation by favoring repeated interaction with the same small set of individuals, which breaks up larger aggregations into smaller, more isolated groups (in line with Bode et al. 2011a). In smaller groups, defectors overexploit their cooperative neighbors more intensely, driving them to extinction faster. This isolates the defectors, who then have a lower chance of survival and reproduction without any cooperators to exploit. These results suggest that aggregating with familiar individuals enhances the efficacy of this process.

These results suggest that a preference for aggregating with familiar individuals favors cooperation by increasing the interdependence among neighboring individuals. Interdependence between cooperative partners has previously been proposed as an explanation for the prevalence of cooperation in nature. Bshary and colleagues (2016) suggest that interdependence can change the long-term payoffs of cooperation and defection so that it is in animals' interests to cooperate. Even if an individual pays a cost to help their partner, if that helps their partner survive, and they rely on their partner for other social benefits, such as protection against predators, or increased foraging success, then it may actually be in their own interest to cooperate. In some cases, such as for predator defense in large groups, dependence on any one individual may be so low that interdependence is not relevant, but we would expect that repeated interaction among the same individuals, such as through aggregation with familiar individuals, would enhance its effects. We expect interdependence to be especially important to species such as vampire bats which cannot survive more than two nights of failed foraging attempts without receiving a donation from their roost mates (Wilkinson, 1984; 1990) and indeed have been found to form stable food-sharing relationships with familiar partners within a highly dynamic fission-fusion society (Carter & Wilkinson, 2013). Queller (2011) described a similar process mathematically as a form of expanded inclusive fitness analogous to symbiosis, where an individual's fitness depends on its ability to increase the fitness of its cooperative partner - such as by increasing its partner's chances of survival - thereby enabling its partner to cooperate in return. Our work extends this, demonstrating how interdependence naturally emerges from aggregation with familiar individuals in fission-fusion populations when selection can act on survival. Furthermore, we show that interdependence, which can arise from aggregation with familiar individuals, is enough to make cooperation an evolutionarily stable strategy even in the absence of other mechanisms for the evolution of cooperation, such as kin selection and reciprocity (Nowak, 2006).

Our model additionally provides insight into how population structure more broadly can promote cooperation. We show that non-viscous fission-fusion populations can make cooperation evolutionarily stable. Furthermore, our model employs a social network of familiarity relations between agents, on which cooperative interactions take place. Many models have suggested that social network structure can promote the evolution of cooperation (for a review see Szabó & Fáth, 2007), however, most social network models are static (e.g. Marcoux & Lusseau, 2013; Fotouhi et al., 2019; Battiston et al., 2017; Kun & Scheuring, 2008; Santos & Pacheco, 2005; Ohtsuki et al., 2007; Ifti et al., 2004) or alter connections based on the outcome of cooperative interactions (for a review see Perc & Szolnoki, 2010). To our knowledge, ours is the first model to demonstrate that a dynamic network that emerges from interactions with spatial neighbors in a fission-fusion population can favor the evolution of cooperation. Future work is needed to explore not only the effects of network structure on the evolution of cooperation, but also the rich potential for interaction between social network structure and movement in space (Bode et al., 2011b).

Our model also resembles Wilson's (1975) trait-group model, where individuals gather into groups that are acted on by selection in accordance with their proportion of cooperators and defectors, and then individuals disperse and form new groups. Our model provides a potential mechanism for the formation of trait-groups through the aggregation of familiar individuals in space, in contrast with past models that have relied on aggregation among kin (e.g. Maynard Smith, 1964; Traulsen & Nowak, 2006), making it difficult to disentangle group-level selection and kin-selection (Marshall, 2011). Our model additionally suggests a means of assortment among cooperators, which is necessary for selection at the group level to favor cooperation (Okasha, 2006), as well as a means of selection between trait-groups, based entirely around agents playing a prisoner's dilemma game, as groups that are invaded by defectors are more likely to die out, while those composed of cooperators survive and contribute more offspring to the next generation. Future work is needed to more directly study the process of group-level selection and to gain a deeper understanding of the important role that interdependence and mutualistic processes may play in the evolution of cooperation.

Preferentially aggregating with familiar individuals is particularly promising as a means of promoting the evolution of cooperation because it is widespread across taxa, including those without aggregation among relatives (e.g. fish, Griffiths & Ward, 2011; ravens, Szipl et al., 2015; bats, Carter & Wilkinson, 2016; Tonkean macaques, Sueur et al., 2010; bighorn ewes, Vander Wal, 2016; horses, Cameron et al., 2009; Humboldt penguins, Coffin et al., 2011) and does not necessarily require costly complex cognitive abilities (Carlson et al., 2020). Individuals of many species not only preferentially associate with familiar individuals, but also are more cooperative and cohesive in predator defense (Chivers et al., 1995), have increased foraging success (Ward & Hart, 2005), are less aggressive (Webster & Hart, 2007; Gherardi et al., 2012), support each other in accessing resources (Szipl et al., 2015), share food (Carter & Wilkinson, 2013), and provide protection from harassment (Cameron et al., 2009). It has been proposed that because they are more cooperative and less aggressive, familiar individuals require less attention to monitor than unfamiliar individuals, making familiar individuals preferable associates (Griffiths et al., 2004). However, in well mixed populations, there is no reason to expect that those that an individual happens to become familiar with should be more cooperative and therefore require less attention.

The results of our model imply that in fission-fusion populations, preferentially aggregating with familiar individuals can enable cooperation to remain stable against defection at boundary conditions, potentially resulting in a population of cooperative groups of familiar individuals. Future work is needed to explore the possibility of coevolution between a preference for familiar individuals and cooperation, as well as how preferentially associating with familiar individuals may interact with and possibly support the evolution of other mechanisms for promoting the evolution of cooperation.

Chapter 3: An Agent-based Simulation of the Effect of Collective Movement and the Formation of Associations over Repeated Interaction on Emergent Social Network Structure

Introduction

Population structure can have wide-reaching effects on individuals and populations. Much work has been done on the determiners and effects of group size. More recent work has emphasized the importance of which individuals group together (Farine et al., 2015). Questions about population structure are especially complex for fission-fusion societies. Fission-fusion dynamics are characterized by the splitting (fission) and merging (fusion) of groups, resulting in fluid group composition over time (Couzin, 2006). Fission-fusion dynamics are prevalent across taxa, including fish (Couzin, 2006), dolphins and whales (Connor et al., 1998), elephants (Couzin, 2006), primates (Aureli et al., 2008), bats (Wilkinson et al., 2019), and likely birds (Silk et al., 2014). Fission-fusion dynamics range from structured, multi-level societies based around family units, as found in elephants (Couzin, 2006), toothed whales such as orcas (Connor et al., 1998), and some primates (Aureli et al., 2008), to less structured societies with high rates of mixing between groups, as found in some species of fish (Couzin, 2006), dolphins (Connor et al., 1998), some primates (Aureli et al., 2008), and bats (Wilkinson et al., 2019). Aureli et al. (2008) take this a step further and propose that all species exist on a continuum of fission-fusion dynamics, from high variation in group cohesion, size, and composition, to low variation.

Social network analysis has come to prominence in recent decades as a powerful tool for quantifying and understanding population structure (Pinter-Wollman et al., 2014). However, the complex interplay between social network structure and the collective movement dynamics that drive and are shaped by it is less well studied (Bode et al., 2011a). Advances in automated tracking have enabled researchers to achieve a fine-grained understanding of how individuals make moment-to-moment movement decisions based on the positions and movements of others (Oscar et al., 2023). Agent-based models have been used to understand the effect of more long-term association networks on collective movement (Bode et al., 2011b). However, little work has been done on how social network structures may emerge from collective movement dynamics and the potential for feedback between the two fundamental and potentially divergent social forces.

The interaction between collective movement and social network structure may be especially important for the many species across taxa that are known to form preferential associations with familiar individuals (e.g., Ward & Webster, 2016; fish, Griffiths & Ward, 2011; mammals and some birds, Massen et al., 2010; bats, Wilkinson et al., 2019; some invertebrates, Gherardi et al., 2012). Collective movement patterns can determine which individuals have the opportunity to interact based on mere spatial proximity, which can then lead to the formation of more long-term connections. Furthermore, if individuals preferentially associate with familiar individuals, this can lead to a feedback loop where individuals form connections based on proximity due to the dynamics of collective movement, and then alter their movement to continue to associate with the now familiar individuals.

This question is particularly well suited to investigation using agent-based models, which have been used extensively to simulate collective movement. The simple individual-level rules driving collective movement decisions are well understood, conceptually and mathematically (Katz et al., 2011). Agent-based models are also frequently employed to understand the effects of social network structure on phenomena such as disease transmission, information transfer, and the evolution of cooperation (Szabó & Fáth, 2007), and there is untapped potential for using agent-based models to understand how such social network structures arise in the first place. Agent-based models are particularly valuable for disentangling the actual underlying network structure from coincidental proximity.

Here we use a typical agent-based model of collective movement (Joshi et al., 2017) with an underlying social network based on familiarity to understand the effect of altering the dynamics of collective movement (Couzin et al., 2002) on social network structure and the feedback that results from preferentially associating with familiar individuals. We take advantage of the fine-grained level of control over agent-based models to look at how each of the parameters that define collective movement affects the resulting network structure. We also evaluate how the parameters that define how individuals acquire and lose familiarity modulate these effects. We both directly evaluate the effect of collective movement on familiarity network structure and look at the additional effect of feedback in populations where individuals prefer to associate with familiar conspecifics.

Model Overview

We employed a simplified version of the model in chapter 2 for this chapter, with only collective movement and social network processes.

This model description follows the ODD (Overview, Design concepts, Details) protocol for describing individual- and agent-based models (Grimm et al., 2006), as updated by Grimm et al. (2020).

Entities, State Variables, and Scales

Entities: Our model consists of 100 agents, which represent individuals of a focal species, analogous to schooling fish, flocking birds, or herding ungulates.

State Variables:

Variable	Meaning	Range	Value
Interacted agents	All agents this agent has interacted with and its degree of acquaintance with each agent.	Weighted, directed network of agents.	Dynamic - agents accumulate acquaintance as they interact and lose acquaintance when they do not interact.
Familiar agents	Which agents this agent considers to be familiar.	Directed network of agents.	Dynamic - agents can be added and removed based on changes in this agent's degree of acquaintance with them.
Location	The agent's x-y coordinates in the simulated space.	A pair of coordinates each of which can be between 0 and the space size.	Dynamic - changes as the agent moves through the space according to the movement algorithm.
Direction of movement	How much this agent moved on the x and y axis in the previous step.	A pair of x-y values whose tangent is equal to the simulation step size.	Dynamic - changes as the agent moves through the space according to the movement algorithm.

Table 1. All agent state variables, their meanings, ranges, and how their values are set.

Scales: The model employs a two-dimensional continuous space, standardized to 30x30 units for a fixed population density, with periodic (toroidal) boundaries, representing an infinitely large area without edge effects. Each unit of space with a radius of one is roughly analogous to a single agent's "personal space" to allow for schooling or flocking interactions (after Joshi et al., 2017) within a sizable population.

Time is modeled in discrete steps. On each step, all agents move a short distance, allowing for precise simulation of schooling or flocking behavior, and gradually form and break familiarity associations over the same timescale.

Note that in subsequent sections, agents' state variables are in *italics*, while simulation-level parameters, which will be defined with the subroutines, are in <u>*italics and underlined*</u>.

Process Overview and Scheduling

On each timestep, all agents are randomized and each carries out the following processes in the following order:

5. If there are any agents within the agent's <u>*interaction radius*</u>:
    a. Update Familiarity: choose a random agent and update both agents' *interacted agents* and *familiar agents* to increment their degree of acquaintance with each other, and determine whether they now consider each other to be familiar.
6. Move: updates the agent's *location* and *direction of movement* based on its current *direction of movement* and the *location*s, *direction*s *of movement*, and *familiarity* of the other agents within its <u>*aggregation radius*</u>.

7. Prune Network: decrements the degree of acquaintance with the other agents in this agent's *interacted agents* based on the *familiarity decay rate* and updates its *familiar agents* accordingly.

On each step, the order in which the agents act is randomized to avoid order effects. Typically, all agents move a much smaller distance in the space than their *interaction radius*, so the order in which Move and Interact occur is largely arbitrary. Prune Network occurs after Move, so that the agents have an opportunity to aggregate with newly familiar individuals before their degree of acquaintance is decremented.

Algorithm 1. Step. On each step all agents are randomized and each may have the opportunity to interact with a neighboring agent, update their degree of acquaintance and familiarity with that agent, move, and prune inactive familiar connections.

```
if this agent has any neighbors within interaction radius units of its location then
    Choose a random neighbor
    UPDATE FAMILIARITY between this agent and its neighbor
end if
MOVE
PRUNE NETWORK of this agent's interacted agents and familiar agents based on this
agent's  familiarity decay
```

Initialization

The model is initialized with a stable population of 1000 agents. All agents are placed at random x-y *location*s in the space, with a pair of x-y coordinates drawn uniformly from 0 to the *space size*, and a random *direction of movement* drawn uniformly from 0 to 360 degrees. They are initialized with no *interacted agents* or *familiar agents*.

Algorithm 2. Initialize. Create a population of the total number of agents with randomly drawn locations, and directions of movement.

```
for 0 to 1000 do
    Create new agent a
    Initialize a's location = x-y coordinates ~ U(0, space size)
    Initialize a's direction of movement ~ U(0, 360)
end for
```

Submodels

Parameters

Name (variable)	Meaning	Default Value (and Units)	Range Analyzed	Source
Update Familiarity				
Interaction radius	Distance at which an agent looks	2 spatial units	1-20	Joshi et

(interactrange)	for other agents to interact with.			al., 2017
Familiar bias (famBias)	This agent’s weight in favor of aggregating with familiar agents as opposed to unfamiliar agents.	.5 (no bias), .9 (strong bias)		
Familiarity decay rate (decay)	How much this agent’s acquaintance with other agents decreases on each step in which they do not interact.	.01 units of acquaintance	0-.7	
Familiarity threshold (lrate)	Degree of acquaintance with another agent necessary to consider it familiar.	5 units of acquaintance	1-100	
Maximum memory capacity (memory)	The maximum number of agents this agent can consider to be familiar at a time.	3 agents (limited memory), 100 agents (unlimited)		
Move				
Repulsion radius (repulserange)	Distance at which an agent moves away from its neighbors.	1 spatial unit (sets the scale)		Joshi et al., 2017
Aggregation radius (viewrange)	Distance at which an agent moves toward and aligns with its neighbors.	2 spatial units	1-20	Joshi et al., 2017
Flocking weight (flockstr)	Relative weight of aligning with other agents as opposed to moving toward them.	.5	0-1	Joshi et al., 2017
Persistence weight (contstr)	Relative weight of continuing in the same direction as opposed to aggregating with others.	.2	0-1	Joshi et al., 2017
Movement error (error)	Standard deviation of random error in movement.	.05	0-1	Joshi et al., 2017
Maximum rotation (maxrotate)	Maximum change in direction of movement on a single step.	50 degrees	10-360	Joshi et al., 2017
Step size (stepsize)	Distance moved on each timestep.	.2 spatial units	.1-2	Joshi et al., 2017

Table 2. All model parameters, their meanings, baseline values and value ranges used in simulation experiments, and relevant citation (if applicable).

Update Familiarity

$F_{i,j}$ = 1 if agent i considers agent j to be familiar, and 0 otherwise

$I_{i,j}$ = agent i's degree of acquaintance with agent j

h = the *familiarity threshold*

m = the *maximum memory capacity*

If any other agents are within *interaction radius* units of this agent's *location*, then one of those agents is chosen at random to "interact". Each agent increments its degree of acquaintance with the other by one (see Equation 8 for the net change in degree of acquaintance on each step) and then decides whether to consider the other to be familiar. If an agent's partner is not in its *interacted agents*, then the agent creates an edge to its partner with an acquaintance of 1, otherwise, their acquaintance is just incremented.

In addition, if an agent does not consider its partner to be familiar and their degree of acquaintance ($I_{i,j}$) is greater than the *familiarity threshold* (h), then it can become familiar ($F_{i,j}$ = 1). If the agent's total number of *familiar agents* ($\sum F_i$) is lower than the *maximum memory capacity* (m), it will now recognize its partner as familiar. Alternatively, if the agent's memory is at capacity, but the agent's acquaintance with its partner is higher than its lowest acquaintance with a familiar agent (i.e., $MIN(I_{i,k}F_{i,k})$) the partner will become familiar, replacing the lowest acquaintance familiar agent.

$$F_{i,j} = \begin{cases} 1, & \textbf{if} \;\; I_{i,j} > h_i \textbf{ and } (\sum F_i < m_i \textbf{ or } MIN(I_{i,k}F_{i,k}) < I_{i,j}) \\ 0, & \textbf{otherwise} \end{cases} \tag{2}$$

Algorithm 3. Update Familiarity. When two agents interact, their degree of acquaintance increases and they may recognize each other as familiar.

```
Given a partner agent randomly chosen from within this agent's interaction radius
if partner is not in this agent's interacted agents then
    Add partner to interacted agents
    Set level of acquaintance to partner = 1
else
    Increment level of acquaintance with partner by 1
end if
if partner is not in this agent's familiar agents then
    if acquaintance with partner > familiarity threshold then
        if this agent's number of familiar agents < maximum memory capacity then
            Add partner to this agent's familiar agents
        else if level of acquaintance with partner > MIN(familiar acquaintance) then
```

```
            Remove MIN(familiar acquaintance) from familiar agents
            Add partner to familiar agents
        end if
    end if
end if
```

Move

d_i = the *direction of movement* of agent i

l_i = the *location* of agent i

R_i = array of agents within the *repulsion radius* of agent i

A_i = array of agents within the *aggregation radius* of agent i

w_f = *flocking weight*

w_p = *persistence weight*

s = *step size*

= random error in movement ~ N(0, *movement error*)

b = the *familiar bias*

$F_{i,j}$ = 1 if agent i considers agent j to be familiar, and 0 otherwise

On each step, each agent (*i*) updates its *location* (l_i) and *direction of movement* (d_i) according to the active movement model of Joshi et al. (2017), modified to take into account familiarity between agents ($F_{i,j}$). If there are any other agents (R_i) within *repulsion radius* units of the active agent's *location*, then it moves in the opposite direction of their average location (Equation 3). Repulsion takes precedence over all other movement decisions.

$$r_i = \frac{1}{|R_i|} \sum_{j}^{R_i} \frac{l_i - l_j}{\|l_i - l_j\|} \tag{3}$$

Otherwise, if there are any other agents (A_i) within *aggregation radius* units of the active agent's *location*, then the agent moves to align with (Equation 4)

$$f_i = \frac{1}{|A_i|} \sum_{j}^{A_i} d_j b_i^{F_{i,j}} (1 - b_i)^{1 - F_{i,j}} \tag{4}$$

and move toward the other agents (Equation 5),

$$o_i = \frac{1}{|A_i|} \sum_{j}^{A_i} \frac{l_j - l_i}{\|l_j - l_i\|} b_i^{F_{i,j}} (1 - b_i)^{1 - F_{i,j}} \tag{5}$$

weighted by the *flocking weight* (w_f) and the *familiar bias* (b), with some tendency to continue moving in the same direction, according to the *persistence weight* (w_p; Equation 6).

$$a_i = w_f(1 - w_p) f_i + (1 - w_f)(1 - w_p) o_i + w_p d_i \tag{6}$$

Otherwise, if no other agents are present, then the agent continues in its current *direction*. In all cases, the agent's movement is adjusted by some random error ( ) drawn from a normal distribution, with a mean of 0 and a standard deviation of *movement error* (Equation 7).

$$d_i = \epsilon + \begin{cases} r_i, & \textbf{if } |R_i| > 0 \\ a_i, & \textbf{if } |A_i| > 0 \\ d_i, & \textbf{otherwise} \end{cases} \tag{7}$$

All movement is constrained by the __*maximum rotation*__. If the difference between the agent’s current *direction of movement* and the newly calculated direction is greater than __*maximum rotation*__, its *direction of movement* is only incremented by the __*maximum rotation*__ in that direction.

$$d_{i,t+1} = d_{i,t} + \mathrm{MIN}(|d_{i,t+1} - d_{i,t}|, \textit{max rotation}) \tag{8}$$

The agent will then move __*step size*__ units in the calculated direction.

Algorithm 4. Move. Each step, each agent moves according to the active movement model of Joshi et al. (2017). If there are any other agents within its repulsion radius, the agent moves away from their average location. Otherwise, if there are any agents within its aggregation radius, the agent moves toward them and aligns with their direction of movement. Otherwise, the agent persists in moving at its previous trajectory with some random error.

```
if this agent has any neighbors within repulsion radius units of its location then
    Set new direction = average direction away from neighbors
else if this agent has any neighbors within aggregation radius units of its location then
```

```
    for each neighbor do
        if the neighbor is in this agent’s familiar agents then
            Set weight = the familiar bias
        else
            Set weight = 1 - the familiar bias
        end if
        Increment flocking direction by neighbor’s direction of movement * weight
        Increment aggregation direction by direction toward neighbor’s location * weight
    end for
    Set new direction = persistence weight * this agent’s direction of movement + (1-persistence weight)*(flocking weight * flocking direction + (1-flocking weight) * aggregation direction) + random error
else
    Set new direction = current direction of movement + random error
end if
if ABS(new direction - current direction of movement) > maximum rotation then
    Set this agent’s direction of movement = current direction of movement + maximum rotation in the desired direction
else
    Set this agent’s direction of movement = new direction
end if
```

Prune Network

On each step, each agent (*i*) goes through its *interacted agents* (I_i) and decrements its degree of acquaintance with each of the other agents (*j*) by the *familiarity decay rate*, resulting in a net change on each step as described by Equation 8.

$$\Delta I_{i,j} = \begin{cases} 1 - decay_i, & \textbf{if } i \text{ and } j \text{ interacted on this step} \\ -decay_i, & \textbf{otherwise} \end{cases} \tag{9}$$

If any acquaintance drops below 0, the agent is removed from *interacted agents*, and if the agent was in *familiar agents*, it is now removed as well and no longer considered to be familiar.

Algorithm 5. Prune Connections. On each step, each agent goes through its interacted agents, decrements their degree of acquaintance by the familiarity decay rate, and removes all agents whose acquaintance drops below 0 from its interacted agents and familiar agents.

```
for each connection in this agent's interacted agents do
    Decrement the degree of acquaintance of the connection by the familiarity decay rate
    if the degree of acquaintance of the connection < 0 then
        Remove the connection from this agent's interacted agents
        if the connection is also in this agent's familiar agents then
            Remove the connection from this agent's familiar agents
```

```
        end if
    end if
end for
```

Simulation Experiments

To evaluate the effect of different types of collective movement and dynamics of acquiring familiarity, we analyzed the structure of the familiarity network after allowing the simulation to run for 5,000 steps, with 10 replicates for a range of different movement and familiarity conditions. We used the igraph package (Csardi & Nepusz, 2006) in the R Statistical Software (v4.2.1; R Core Team, 2022) to calculate the edge density (Wasserman & Faust, 1994), mean and standard deviation in degree, diameter, mean distance (West, 1996), number of components, and mean and standard deviation in component size. We additionally used igraph's fast-greedy clustering algorithm to find the modularity as well as mean and standard deviation in cluster size and the proportion of cross-cluster edges (Clauset et al., 2004). All measures were computed for all conditions (see Supplemental Results for the complete results).

We compared the effects of each parameter under a baseline condition, where agents formed connections based on familiarity, but did not preferentially aggregate with familiar individuals ($b = .5$), and under a preference condition, where agents strongly preferred to aggregate with familiar individuals ($b = .9$). We additionally evaluated each parameter's effects with limited memory capacity (3 agents) and unlimited memory capacity (100 agents). Due to time and computational constraints, we could not evaluate all possible combinations of parameter values. So, instead we simulated a range of values of each of the movement parameters, while keeping all other parameters constant (see Table 2 for ranges and default values).

Results

Here we present the effect of each of the movement and familiarity parameters individually on network structure. For all network measures with all combinations of parameter values run, see Supplementary Figures.

Movement Parameters

Aggregation Radius

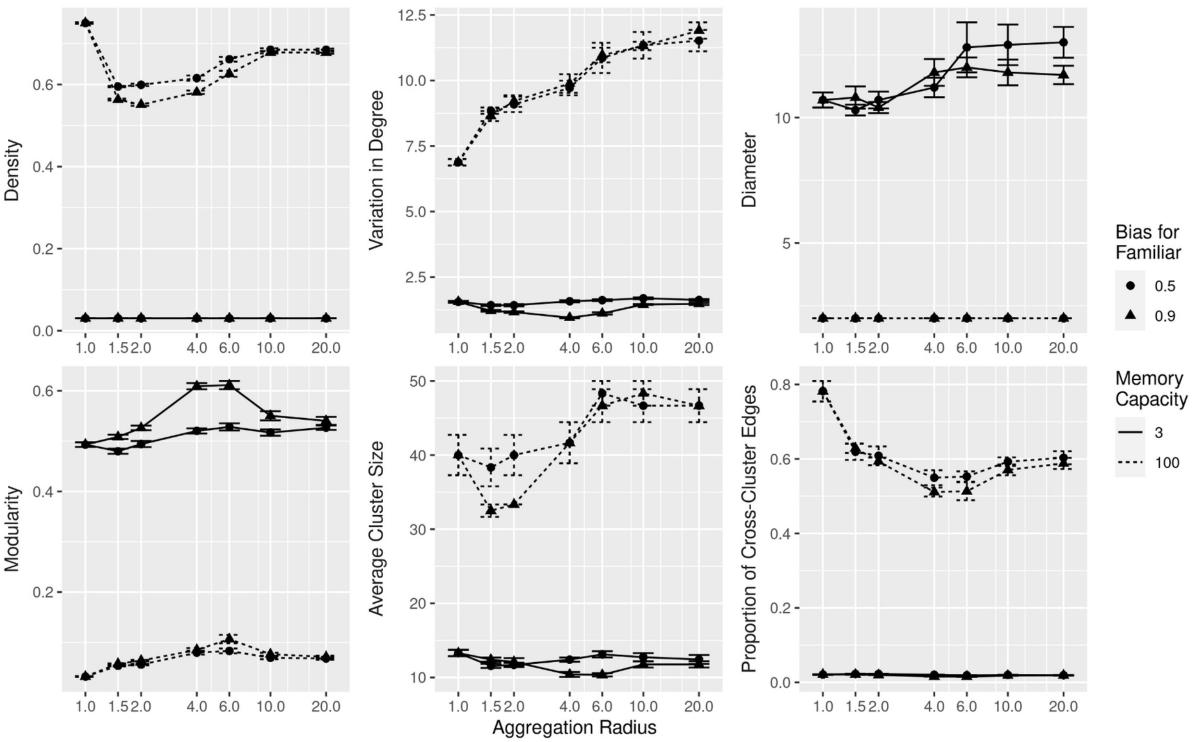


Figure 1. The effect of the radius at which individuals aggregate with their neighbors (from no aggregation at 1 to aggregating with neighbors from across nearly the whole space at 20) on the structure of the emergent network of familiarity relations based on repeated interaction after 5,000 simulation steps, averaged over 10 replicates, with no bias (.5; circles) and strong bias (.9; triangles) for familiar individuals, and limited memory capacity (3; solid line) and unlimited memory (100; dashed). Network structure was analyzed in terms of a) density - the proportion of all possible edges on the network that actually exhibited a familiarity relation, b) standard error in degree - variation in how many familiar associations each individual had, c) network diameter - shortest distance between the two furthest agents on the network, d) modularity - the extent to which the network was divided into discrete clusters, e) average cluster size - the average size of clusters of highly interconnected agents, and f) proportion of cross-cluster edges - the proportion of all possible edges between clusters that actually exhibited a familiarity relation.

Increasing the radius at which agents aggregated with their neighbors resulted in familiarity networks with larger diameters when agents had limited memory capacity (memory = 3; Figure 1C), more variation in degree among agents (Figure 1B), and overall higher density when individuals had unlimited memory capacity (memory = 100; Figure 1A). The exception to this trend was that the density was highest when the aggregation radius was equal to the repulsion radius (aggregation radius = 1; Figure 1A), so that agents just avoided those that got too close and did not aggregate toward them. This suggests that with unlimited memory, under default conditions where it is easy to become and stay familiar (decay rate = .01; familiarity threshold = 5), random movement resulted in a more connected network than intentional aggregation, likely because moving randomly caused agents to interact with more individuals rather than repeatedly interacting with the same ones.

Aggregation radius shows a non-monotonic effect on modularity (Figure 1D), where the highest modularity occurs at intermediate aggregation radii. This effect is especially pronounced when agents had limited memory capacity (memory = 3) and a strong bias in favor of aggregating with familiar individuals (bias = .9). However, the corresponding, opposite trend in the proportion of cross-cluster edges, which should be directly inverse to modularity, is strongly apparent with unlimited memory capacity (100; Figure 1F), suggesting that despite the small peak in modularity due to the high network density, there is still a sizable change in how much the network is divided up into modular clusters. Expectedly, the average cluster size increases as agents have a wider radius of other agents to aggregate with (with a little non-monotonicity around entirely random movement at aggregation radius = 1), though the effect is only felt when agents have unlimited memory capacity (memory = 100), whereas limited memory capacity forces clusters to remain small even as modularity decreases (Figure 1E).

Strength of Flocking vs. Aggregation

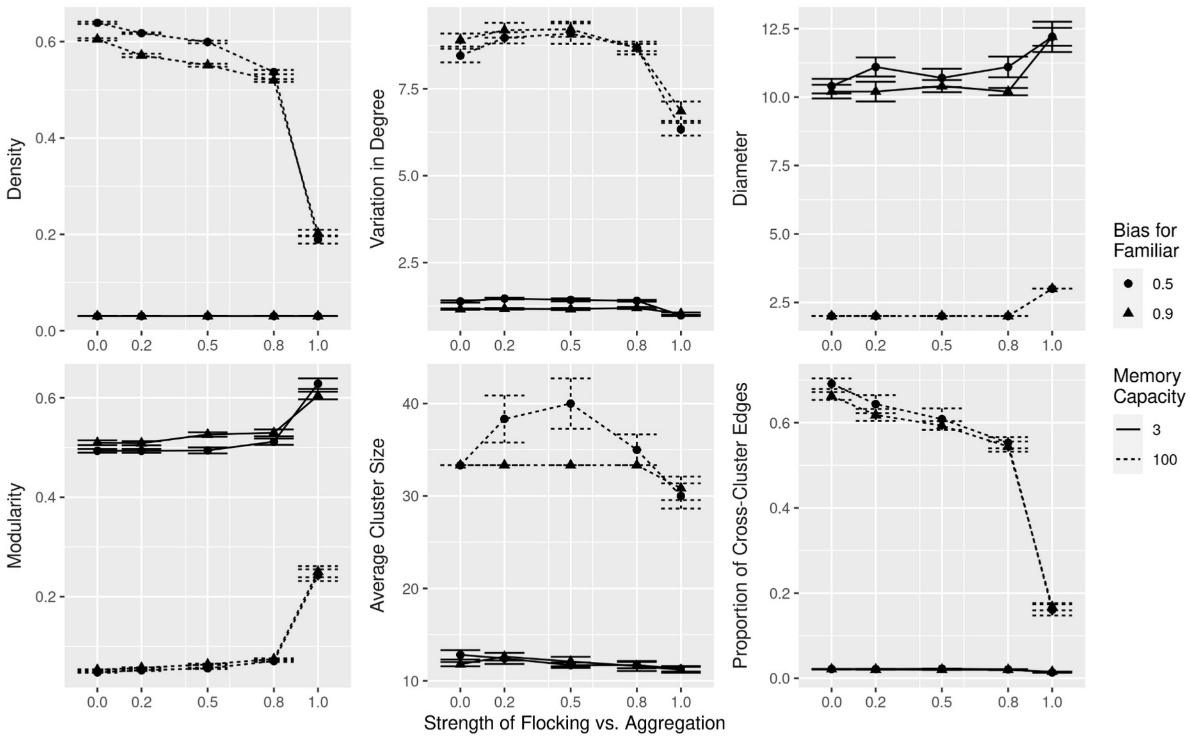


Figure 2. The effect of agents' tendency to align with their neighbors (flocking) as opposed to moving toward their neighbors (aggregating; from only aggregating at 0, to only flocking at 1) on the structure of the emergent network of familiarity relations based on repeated interaction after 5,000 simulation steps, averaged over 10 replicates, with no bias (.5; circles) and strong bias (.9; triangles) for familiar individuals, and limited memory capacity (3; solid line) and unlimited memory (100; dashed). Network structure was analyzed in terms of a) density - the proportion of all possible edges on the network that actually exhibited a familiarity relation, b) standard error in degree - variation in how many familiar associations each individual had, c)

network diameter - shortest distance between the two furthest agents on the network, d) modularity - the extent to which the network was divided into discrete clusters, e) average cluster size - the average size of clusters of highly interconnected agents, and f) proportion of cross-cluster edges - the proportion of all possible edges between clusters that actually exhibited a familiarity relation.

The more agents aligned with their neighbors, as opposed to moving toward them, the less dense the familiarity network became (with unlimited memory; 100; Figure 2A). When agents only aligned with their neighbors and did not move toward them (flocking strength = 1), the variation in degree also decreased (Figure 2B), the diameter increased (Figure 2C), and the network became more modular (Figure 2D). Despite the largely consistent modularity, the flocking strength had a non-monotonic effect on average cluster size (Figure 2E), with the largest clusters when there was an approximately equal amount of alignment and aggregation (flocking strength = .5), especially in the absence of a bias for familiar individuals (bias = .5). However, reflecting the sudden increase in modularity at very high flocking strength (1), there is accordingly a sharp decrease in cluster size at that extreme (with unlimited memory; 100; Figure 2E), and a sharp dip in the proportion of cross-cluster edges (Figure 2F).

Persistence in Movement

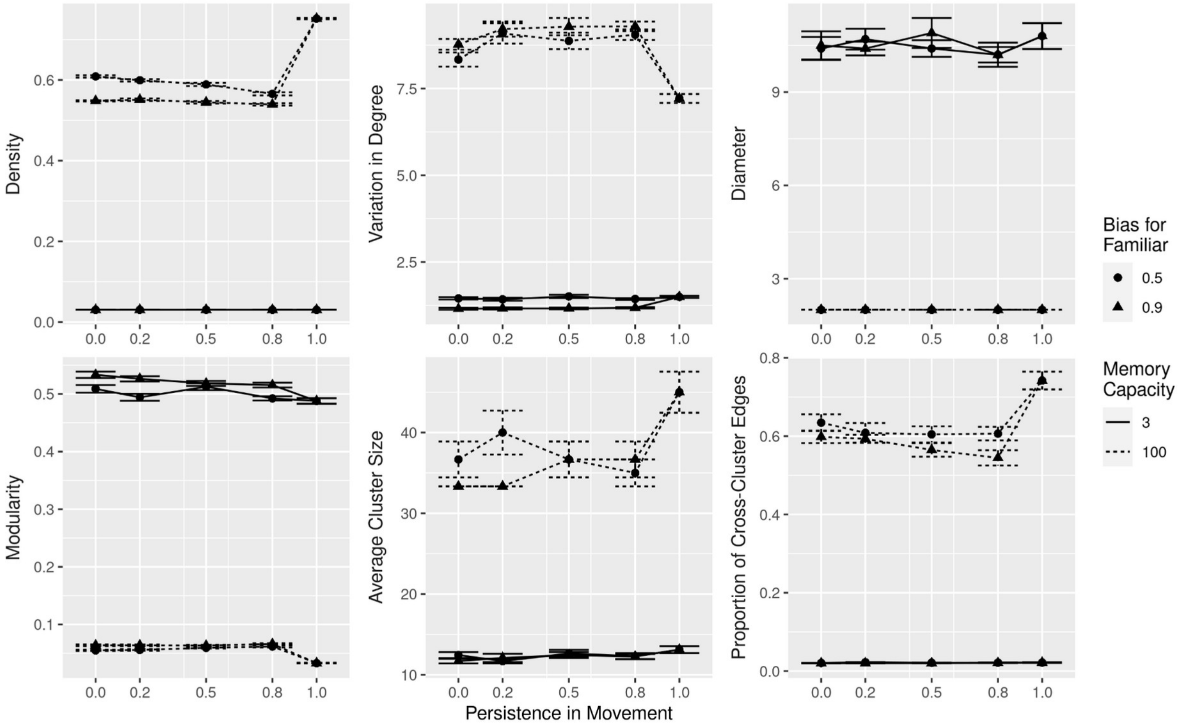


Figure 3. The effect of agents' tendency to persist in moving in the same direction (from no persistence at 0, to only continuing in the same direction with the exception of some small random error at 1) on the structure of the emergent network of familiarity relations based on repeated interaction after 5,000 simulation steps, averaged over 10 replicates, with no bias (.5; circles) and strong bias (.9; triangles) for familiar individuals, and limited memory capacity (3; solid line) and unlimited memory (100; dashed). Network structure was analyzed in terms of a) density - the proportion of all possible edges on the network that actually exhibited a familiarity relation, b) standard error in degree - variation in how many familiar associations each individual had, c) network diameter - shortest distance between the two furthest agents on the network, d) modularity - the extent to which the network was divided into discrete clusters, e) average cluster size - the average size of clusters of highly interconnected agents, and f) proportion of cross-

cluster edges - the proportion of all possible edges between clusters that actually exhibited a familiarity relation.

The tendency of agents to persist in moving in the same direction had a relatively weak effect on network parameters except at extremely high values where agents only persisted in the same direction instead of aggregating with their neighbors (persistence in movement = 1), resulting in essentially random movement (Figure 3). As we found when the aggregation radius was equal to the repulsion radius (aggregation radius = 1; Figure 1), moving randomly resulted in a denser network (Figure 3A), with lower variation in degree (Figure 3B) and somewhat lower modularity (Figure 3D), especially with unlimited memory capacity (memory = 300), as further evidenced by the increase in cluster size (Figure 3E) and edges between clusters (Figure 3F). There was an additional non-monotonic effect of average cluster size (Figure 3E), especially in the absence of a bias in favor of aggregating with familiar individuals (bias = .5), whereas a bias in favor of aggregating with familiar individuals (bias = .9) resulted in a steadier increase in cluster size as the degree of persistence in movement increased with unlimited memory capacity (memory = 300).

Random Error in Movement

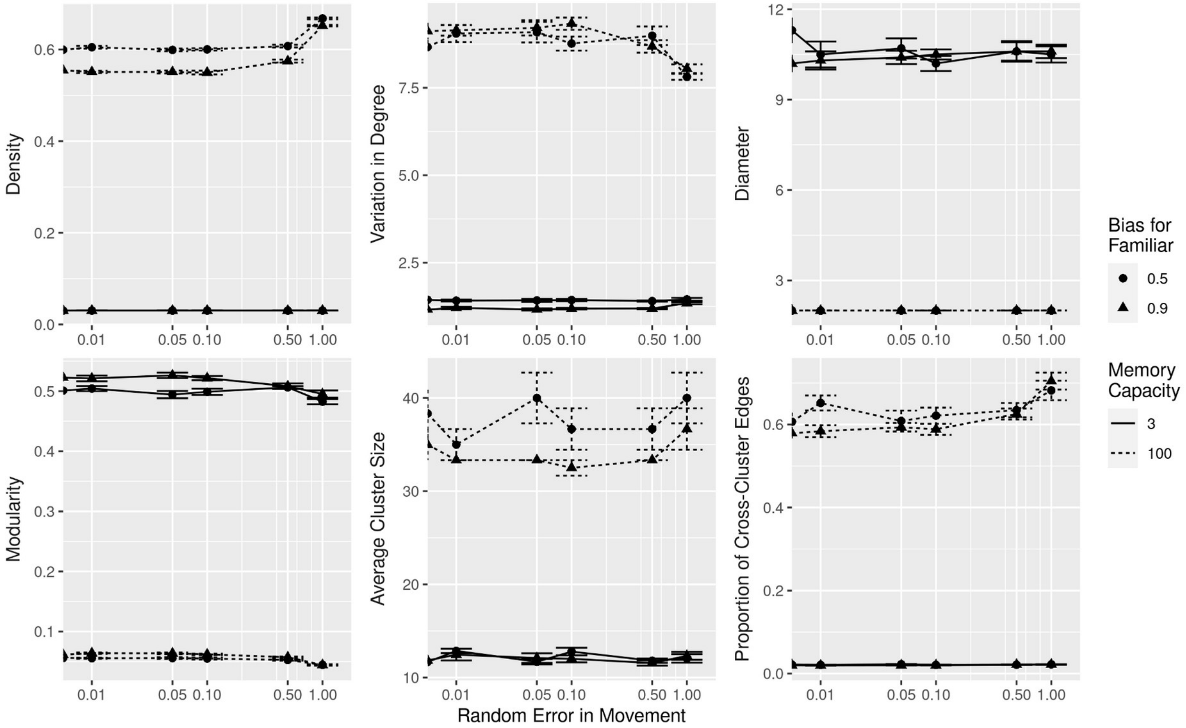


Figure 4. The effect of standard deviation in random error in movement (around a mean of 0; from no random error at 0, to nearly entirely random movement at 1) on the structure of the emergent network of familiarity relations based on repeated interaction after 5,000 simulation steps, averaged over 10 replicates, with no bias (.5; circles) and strong bias (.9; triangles) for familiar individuals, and limited memory capacity (3; solid line) and unlimited memory (100; dashed). Network structure was analyzed in terms of a) density - the proportion of all possible edges on the network that actually exhibited a familiarity relation, b) standard error in degree - variation in how many familiar associations each individual had, c) network diameter - shortest distance between the two furthest agents on the network, d) modularity - the extent to which the network was divided into discrete clusters, e) average cluster size - the average size of clusters of

highly interconnected agents, and f) proportion of cross-cluster edges - the proportion of all possible edges between clusters that actually exhibited a familiarity relation.

The standard deviation of the random error in movement (around a mean of 0) also had a weak effect on the network structure (Figure 4). With a higher standard deviation and therefore more random error in movement (1), the network was somewhat denser (Figure 4A), with less variation in degree (with unlimited memory = 100; Figure 4B), and was less modular (Figure 4D), with more edges between clusters (also with unlimited memory = 100; Figure 4F). Interestingly, there was considerable variation in mean cluster size (Figure 4E), especially with no bias in favor of aggregating with familiar individuals (bias = .5). Additionally, a lack of random error in movement (standard deviation = 0) caused an increase in network diameter (Figure 4C) with no bias (.5) and limited memory capacity (3).

Maximum Turning Angle

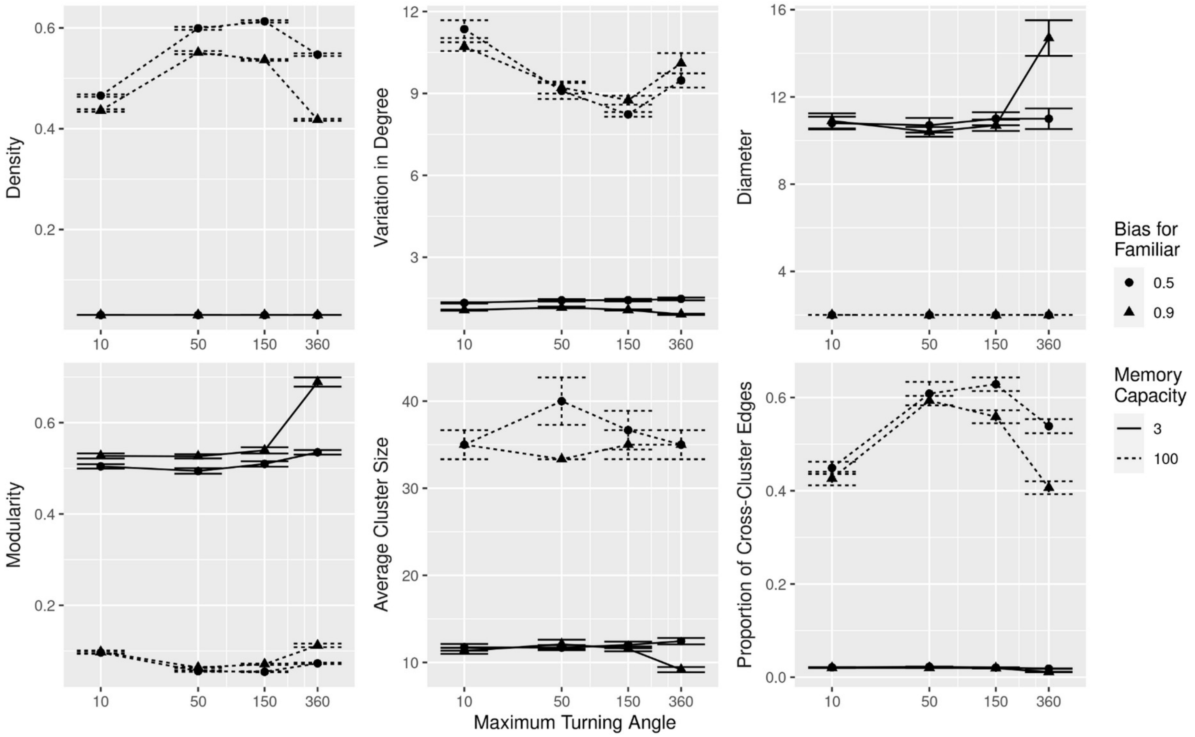


Figure 5. The effect of maximum turning angle (from highly restricted turning at 10 to unrestricted turning at 360) on the structure of the emergent network of familiarity relations based on repeated interaction after 5,000 simulation steps, averaged over 10 replicates, with no bias (.5; circles) and strong bias (.9; triangles) for familiar individuals, and limited memory capacity (3; solid line) and unlimited memory (100; dashed). Network structure was analyzed in terms of a) density - the proportion of all possible edges on the network that actually exhibited a familiarity relation, b) standard error in degree - variation in how many familiar associations each individual had, c) network diameter - shortest distance between the two furthest agents on the network, d) modularity - the extent to which the network was divided into discrete clusters, e) average cluster size - the average size of clusters of highly interconnected agents, and f)

proportion of cross-cluster edges - the proportion of all possible edges between clusters that actually exhibited a familiarity relation.

The maximum turning angle had a remarkably non-monotonic effect on network structure (Figure 5), with the highest density (Figure 5A) but the lowest variation in degree (Figure 5B) at intermediate turning angles (50-150 degrees) with unlimited memory capacity (memory = 100). The absence of a limit on turning angle (maximum angle = 360 degrees) caused a sizable increase in the modularity of the network (Figure 5D), especially when agents preferentially aggregated with familiar individuals (bias = .9) and had limited memory capacities (memory = 3). However, even though modularity was the same as long as there was a limit on maximum turning angle (i.e. maximum angle < 360 degrees), the average cluster size (Figure 3E) and proportion of cross-cluster edges (Figure 3F) exhibited non-monotonic trends. The proportion of cross-cluster edges (Figure 3F) closely followed the network density (Figure 3A), as might be expected. Whereas, the average cluster size (Figure 3E) followed the same trend as modularity (Figure 3D) with limited memory (3) but when agents had unlimited memory (100), the average cluster size exhibited different trends with no bias for familiar individuals (bias = .5) versus a strong bias for aggregating with familiar individuals (bias = .6), with the largest clusters at intermediate turning angles (maximum angle = 50 degrees) with no bias (.5) and smaller clusters at intermediate turning angles (50 degrees) with a strong bias (.9).

Step Size

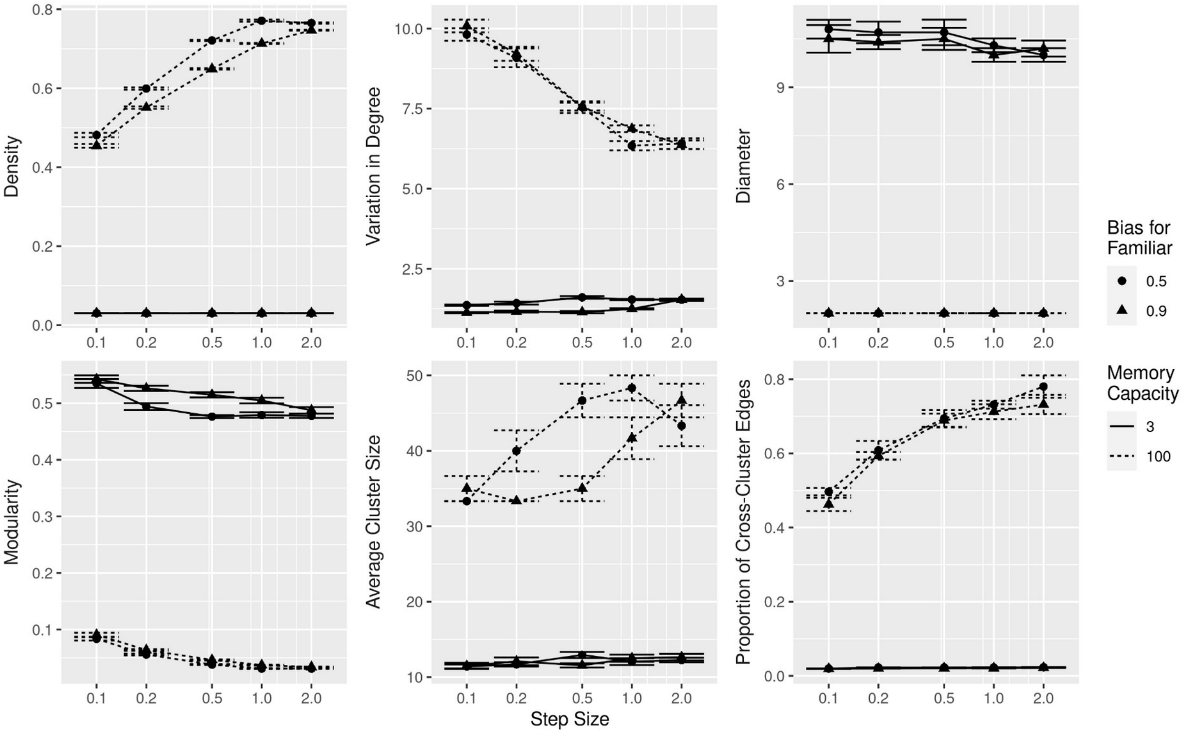


Figure 6. The effect of movement step size on the structure of the emergent network of familiarity relations based on repeated interaction after 5,000 simulation steps, averaged over 10 replicates, with no bias (.5; circles) and strong bias (.9; triangles) for familiar individuals, and limited memory capacity (3; solid line) and unlimited memory (100; dashed). Network structure was analyzed in terms of a) density - the proportion of all possible edges on the network that actually exhibited a familiarity relation, b) standard error in degree - variation in how many familiar associations each individual had, c) network diameter - shortest distance between the two furthest agents on the network, d) modularity - the extent to which the network was divided into discrete clusters, e) average cluster size - the average size of clusters of highly interconnected agents, and f) proportion of cross-cluster edges - the proportion of all possible edges between clusters that actually exhibited a familiarity relation.

Larger step sizes resulted in much denser networks (Figure 6A), but with less variation in degree (Figure 6B) with unlimited memory capacity (memory = 100). The network was accordingly somewhat less modular (Figure 6D), with much larger clusters (Figure 6E) and a higher proportion of cross-cluster edges (Figure 6F), again with unlimited memory capacity (memory = 100). Interestingly, the average cluster size (Figure 6E) was very strongly modulated by whether agents exhibited a bias for aggregating with familiar individuals (bias = .9) or not (bias = .5), with much larger clusters on average at intermediate step size (.2 - 1) when agents exhibited no bias for familiar or unfamiliar individuals.

No change in density (Figure 6A) or proportion of cross-cluster edges (Figure 6F) was observed with a limited memory capacity (memory = 3), and only a very small increase in variance in degree (Figure 6B) and average cluster size (Figure 6E) were observed, slightly modulated by whether agents exhibited a bias for familiar individuals. Diameter was largely unaffected by step size (Figure 6C).

Familiarity Parameters

Interaction Radius

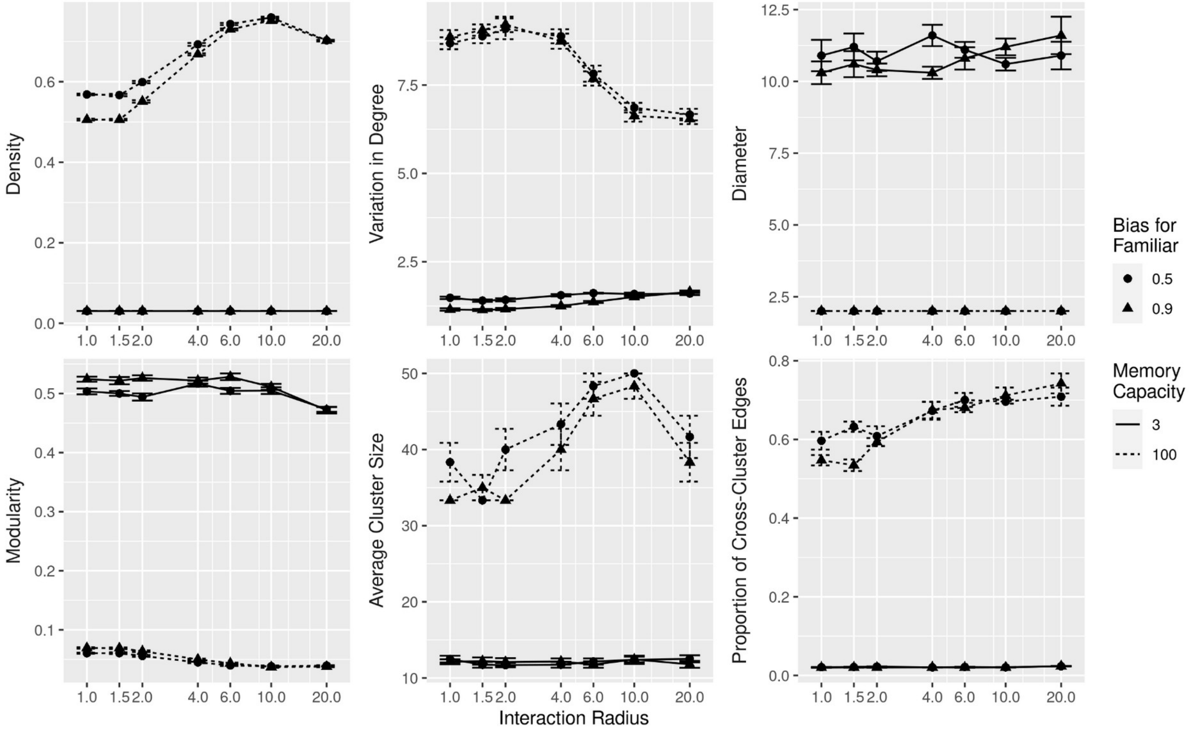


Figure 7. The effect of the radius at which agents choose an interaction partner (from just within their repulsion radius at 1, to almost the entire space at 20) on the structure of the emergent network of familiarity relations based on repeated interaction after 5,000 simulation steps, averaged over 10 replicates, with no bias (.5; circles) and strong bias (.9; triangles) for familiar individuals, and limited memory capacity (3; solid line) and unlimited memory (100; dashed). Network structure was analyzed in terms of a) density - the proportion of all possible edges on the network that actually exhibited a familiarity relation, b) standard error in degree - variation in how many familiar associations each individual had, c) network diameter - shortest distance between the two furthest agents on the network, d) modularity - the extent to which the network was divided into discrete clusters, e) average cluster size - the average size of clusters of highly

interconnected agents, and f) proportion of cross-cluster edges - the proportion of all possible edges between clusters that actually exhibited a familiarity relation.

As the radius at which agents choose their interaction partners, with whom they could become familiar, increased, the network became denser (Figure 7A), but with less variation in degree (Figure 7B). There was no meaningful change in diameter (Figure 7C) or modularity (Figure 7D), but the average cluster size (Figure 7E) and proportion of cross-cluster edges (Figure 7F) increased with interaction radius when agents had unlimited memory capacity (memory = 100). The average cluster size (Figure 7E) exhibited a non-monotonic effect, where it increased to a peak at fairly high interaction radius (10 units), but decreased at extremely high interaction radii (20 units), where agents were essentially choosing their interaction partners at random from the population, which we would expect to result in less opportunity for clustering.

Familiarity Threshold

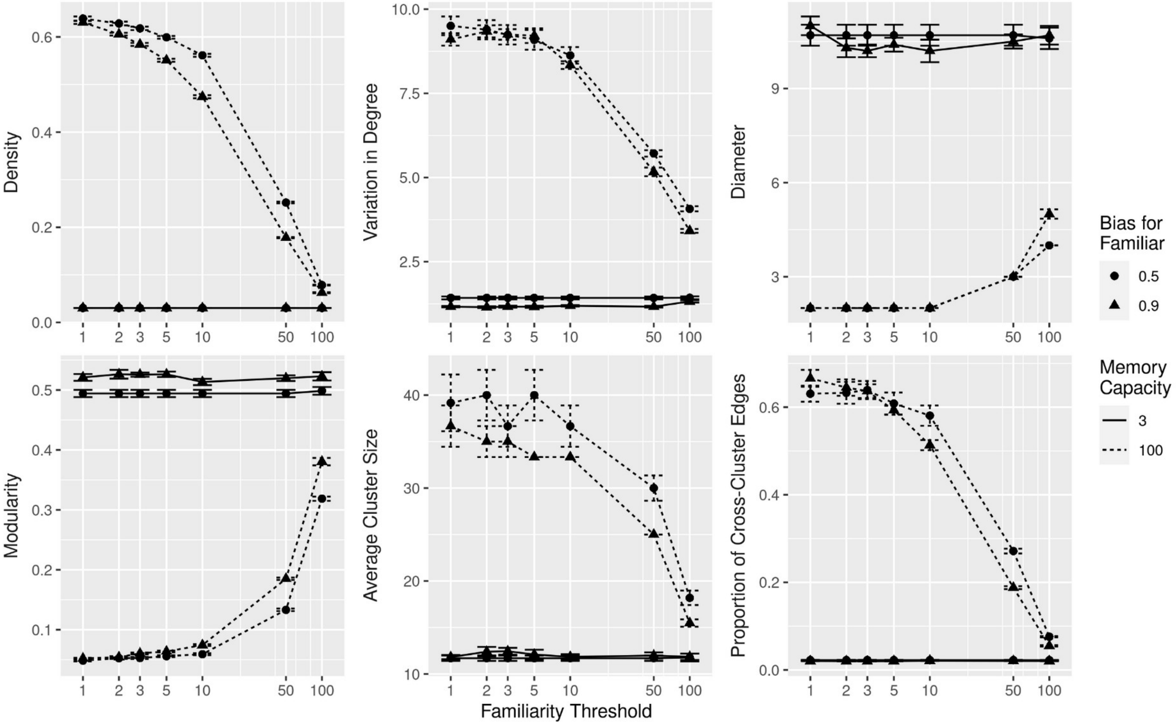


Figure 8. The effect of the threshold for agents recognizing each other as familiar (from 1 interaction to 100 consecutive interactions) on the structure of the emergent network of familiarity relations based on repeated interaction after 5,000 simulation steps, averaged over 10 replicates, with no bias (.5; circles) and strong bias (.9; triangles) for familiar individuals, and limited memory capacity (3; solid line) and unlimited memory (100; dashed). Network structure was analyzed in terms of a) density - the proportion of all possible edges on the network that actually exhibited a familiarity relation, b) standard error in degree - variation in how many familiar associations each individual had, c) network diameter - shortest distance between the two furthest agents on the network, d) modularity - the extent to which the network was divided into discrete clusters, e) average cluster size - the average size of clusters of highly

interconnected agents, and f) proportion of cross-cluster edges - the proportion of all possible edges between clusters that actually exhibited a familiarity relation.

The threshold for becoming familiar had a dramatic effect on network structure with unlimited memory capacity (100; Figure 8), especially at very high thresholds, where agents only became familiar after a long acquaintance (threshold = 50 or 100). Higher thresholds for becoming familiar predictably resulted in less dense familiarity networks (Figure 8A) with less variation in degree (Figure 8B), and a larger diameter (Figure 8C). Higher thresholds also resulted in more modular networks (Figure 8D), with smaller clusters (Figure 8E) and fewer edges between clusters (Figure 8F).

Decay Rate

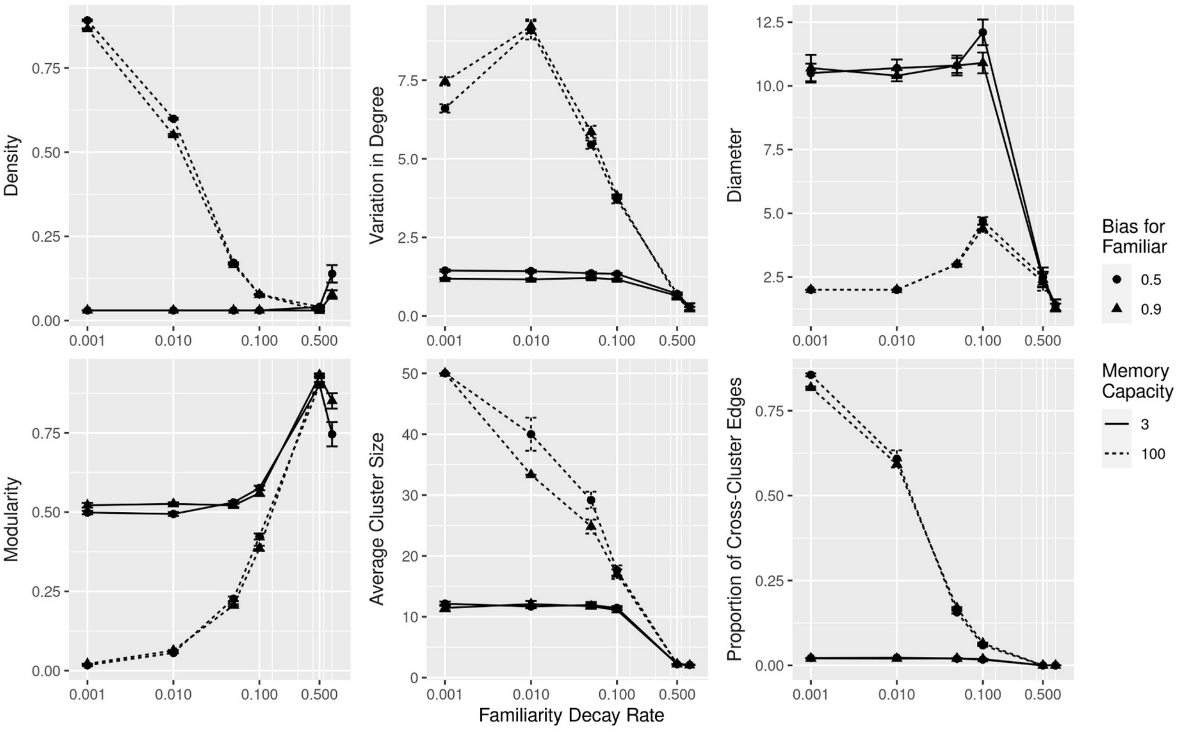


Figure 9. The effect of the rate at which familiarity decreased when agents did not interact (decay rate; from .001 units of familiarity per step to .5 units of familiarity per step) on the structure of the emergent network of familiarity relations based on repeated interaction after 5,000 simulation steps, averaged over 10 replicates, with no bias (.5; circles) and strong bias (.9; triangles) for familiar individuals, and limited memory capacity (3; solid line) and unlimited memory (100; dashed). Network structure was analyzed in terms of a) density - the proportion of all possible edges on the network that actually exhibited a familiarity relation, b) standard error in degree - variation in how many familiar associations each individual had, c) network diameter - shortest distance between the two furthest agents on the network, d) modularity - the extent to which the network was divided into discrete clusters, e) average cluster size - the average size of clusters of highly interconnected agents, and f) proportion of cross-cluster edges - the proportion of all possible edges between clusters that actually exhibited a familiarity relation.

Familiarity decay rate had the largest effect of any parameter (see Supplementary Figures for direct comparison of the effects of each parameter), going from an almost completely connected network with very low decay, especially with unlimited memory capacity (memory = 100), to a highly modular, low-density network at very high decay rates where agents lost familiarity almost as fast as they acquired it (Figure 9). This effect was dampened, but not entirely eliminated in populations with low memory capacity (memory = 3), as extremely high decay resulted in agents forming even fewer connections than they could remember. There was also a small inverse effect at extremely high decay rates (decay = .5), where the network became slightly denser (Figure 9A) and less modular (Figure 9D). Additionally, variation in degree

(Figure 9B) and diameter (Figure 9C) both showed non-monotonic effects with peaks at intermediate decay rates (.01 and .1 respectively), but they sharply dropped off as the decay rate increased.

Discussion

The parameters that define collective movement have been shown to have a significant effect on group structure (Couzin & Krause, 2003), and our results show that this in turn influences the social network structures that emerge. In particular, our simulation demonstrates that the more agents align with their neighbors and the more slowly they move, the less dense and more modular the network of associations becomes. That is to say that if individuals coordinate their movement with others rather than just moving toward them and don’t move too far in each burst, they will form tighter clusters of associated individuals, but the network as a whole will have fewer connections. This emphasizes the importance of coordination for forming long-term associations.

However, movement was not the only factor which had an important effect on network structure. Unsurprisingly, individuals’ memory capacity had a very large effect on the trends we observed, and in fact it was only with limited memory that we were often able to observe any changes in modularity, even as the density of the network remained constant. Perhaps even more striking, the effects of all of the movement parameters were overshadowed by the effects of the threshold for determining how much individuals had to interact to be considered familiar and how quickly familiarity decayed when individuals did not interact. It was really the parameters that defined how familiarity formed which determined the network structure. This result highlights not only the importance of the cognitive processes involved in forming associations, but also the metrics by which we researchers evaluate them. For example, the familiarity threshold which has such an outsize effect could be interpreted as the threshold at which individual animals consider each other to be familiar, or it could be the interaction threshold at which a researcher concludes that a pair of animals has an association. As a result, this model additionally serves as a reminder for researchers to carefully consider the unexpected and often non-linear effects that arbitrary analysis decisions may have on their results.

One parameter which has been perhaps glaringly absent from this discussion is whether individuals preferentially aggregated with those that were familiar to them. Bode et al. (2011) found that when a pre-existing network was imposed upon a similar flocking or schooling population, it decreased the cohesion of the population. However, our model found that whether individuals' aggregative behavior was influenced by an emergent network did not have much of an impact on the resulting structure of the network, especially in comparison with the effects of other parameters. This suggests that because spatial structure limits which individuals can interact in the first place, social preferences, especially those acquired through repeated interaction, may in fact not be able to have such a strong influence on the structure of a population. This may in turn explain why a preference for aggregating with familiar individuals only had an effect on the persistence of cooperation at precarious boundary conditions (see Chapter 2), which in turn emphasizes how important even small differences can be.

However, as with any model, all of these results are mere speculation. Even the apparently simplest creature is more complex than any simulation could capture. The classic optimality models, such as the seminal work by John Maynard Smith (1964) on the evolution of cooperation, are beautiful and useful in their simplicity, but only real organisms can show us how they actually make decisions, and even the most fundamental models are proven wrong time and time again (Pyke, 1984). Meanwhile, advances in technology such as virtual reality and individual tracking using machine learning have enabled empirical work to examine collective movement and social preferences at a more fine grain than ever thought possible in the lab (Oscar et al., 2023) and in the wild (Kays et al., 2015). This explosion of fine-grained data on collective movement makes it more feasible than ever to combine simulation and empirical data to better understand actual organisms and predict future outcomes for populations, especially in a rapidly changing world. It is my hope that this model can serve as a template for future work using agent-based models based on a particular system as a type of null model, as proposed by Farine et al. (2015), or to provide predictions to compare with empirical networks. Agent-based models like this one could be parameterized with empirical movement data and networks could be formed at random or according to hypothesized rules, which could then be analyzed and compared to empirical networks to determine how associations actually form. Further agent-based models of the connection between collective movement and network structure could also be used to simulate the effects of empirical methodology, such as inevitable missing data, on empirical networks, or to predict the response of a population to some disruption, as are becoming increasingly frequent in our rapidly changing world.

Conclusion

Agent-based models are a powerful tool for simulating complex natural phenomena at an otherwise mathematically intractable level of detail. They enable researchers to quantify and explore theoretical possibilities, and to manipulate and analyze processes that are impossible to directly study using empirical approaches. Agent-based models have found particular use in the study of collective behavior and social structure, where not only is the behavior of individuals highly interdependent, but also depends on who they interact with, which is itself dependent on individual behavior (Bode et al., 2011). Agent-based models give researchers the power to manipulate the parameters which dictate individual behavior and the broader context, enabling them to make predictions about the population- and individual-level, and even evolutionary effects of behaviors and environmental factors, which can be impossible to manipulate, and in some cases even to measure, in animals. The first chapter of this dissertation presents a tool for making the most of the remarkable versatility of agent-based models by automating running and gathering data from large swaths of parameter space. However, the versatility and complexity of agent-based models is perhaps also their greatest weakness, as they can easily become as difficult to understand as the complex systems they simulate, but at the same time can be difficult to clearly map on to those natural systems. My final two chapters highlight the strengths and weaknesses of the agent-based modeling approach and suggest a way forward for anyone interested in this highly versatile and complex tool.

As a theoretical tool, agent-based models provide a means of quantifying complex theories and exploring emergent outcomes of individual behavior. My second chapter highlights the potential for emergence in agent-based models as none of the classical mechanisms for the evolution of cooperation (Nowak, 2006) are hard-coded into the model (offspring do not stay with their parents, and agents cooperate or defect indiscriminately), instead assortment among cooperators emerges from the interaction of collective movement dynamics and environmental conditions, which creates interdependence among agents and stabilizes cooperation. However, this insight is not entirely novel; mathematical models have already clearly indicated that the formation of interdependent relationships among individuals can promote the evolution of cooperation even in the absence of more direct reciprocity (Queller, 2011). The contribution of the agent-based model is to extend this result by indicating that population structure, harsh conditions, and a preference for aggregating with familiar individuals are potential sources of interdependence.

In addition, the complexity of agent-based models can be extremely valuable for simulating outcomes for particular populations or ecosystems. Unfortunately, though my second chapter is inspired by cooperation among familiar individuals in a wide variety of species, the theoretical abstractions in the model make it difficult to apply the results to any particular species. The effect I found of a preference for aggregating with familiar individuals on the evolution of cooperation shows how much the particular collective movement dynamics affect the results, but the relationship between collective movement and fitness is unlikely to accurately reflect any vertebrate species because of how much the timeline was condensed to create a functional simulation. In addition, even among fish species, cooperation can take many different forms (Griffiths & Ward, 2011), which may not all even be best represented by the same game (Bshary et al., 2016), let alone exhibit the same spatio-temporal and social dynamics. Likewise, it is difficult to map the results of my third chapter onto any particular species, however, it does provide a template for using agent-based simulations as null models for social network formation. Agent-based models have been proposed as a powerful tool for determining what we would expect networks to look like if individuals interact at random or exhibit some particular kind of preference (Farine et al., 2015). My third chapter also demonstrates the potential value of agent-based models in predicting the effects of particular sampling methodologies on the social networks we construct from empirical data.

Agent-based models can be an extremely powerful tool for increasing our understanding of the natural world, and even help protect it. They are already being used in conservation to predict the effect of environmental changes on sensitive ecosystems (McLane et al., 2011). The recent COVID-19 pandemic further highlighted how the predictive power of agent-based models of disease transmission can contribute to addressing an international crisis, but even in the race to ameliorate the all too real situation, many simulations lacked the fidelity to the real world and validation with empirical data to be useful to policy makers (Lorig et al., 2021). To truly make valuable predictions about real-life phenomena, predictive models must rely on empirical data and be created in close collaboration with the people who engage in empirical work in the field and who will act on the results, so that simulations not only emulate the real world, but also help to further our understanding of it in concrete ways. In theoretical work, it may be especially easy to get lost among the hypothetical fish (as I came to call my agents), but just as empirical work must be guided by theory, theoretical work needs to be guided by the natural world.

References

An, L. (2012). Modeling human decisions in coupled human and natural systems: Review of agent-based models. *Ecological Modelling*, *229*, 25-36. doi: 10.1016/j.ecolmodel.2011.07.010

Andras, P., Lazarus, J., & Roberts, G. (2007). Environmental adversity and uncertainty favour cooperation. *BMC Evolutionary Biology*, *7*, 1-8. doi: 10.1186/1471-2148-7-240

Aureli, F., Schaffner, C. M., Boesch, C., Bearder, S. K., Call, J., Chapman, C. A., . . . Schaik, C. P. V. (2008). Fission-fusion dynamics new research frameworks. *Current Anthropology*, *49*, 627-654. doi: 10.1086/586708

Battiston, F., Perc, M., & Latora, V. (2017). Determinants of public cooperation in multiplex networks. *New Journal of Physics*, *19*, 073017. doi: 10.1088/1367-2630/aa6ea1

Bode, N. W., Wood, A. J., & Franks, D. W. (2011a). The impact of social networks on animal collective motion. *Animal Behaviour* , *82* , 29-38. doi: 10.1016/j.anbehav.2011.04.011

Bode, N. W., Wood, A. J., & Franks, D. W. (2011b). Social networks and models for collective motion in animals. *Behavioral Ecology and Sociobiology* , *65* , 117-130. doi: 10.1007/s00265-010-1111-0

Bradhurst, R. A., Roche, S. E., East, I. J., Kwan, P., & Garner, M. G. (2015). Improving the computational efficiency of an agent-based spatiotemporal model of livestock disease spread and control. *Environmental Modelling Software*, *77* , 1-12. doi: 10.1016/j.envsoft.2015.11.015

Bshary, R., Zuberbuhler, K., & Schaik, C. P. V. (2016). Why mutual helping in most natural systems is neither conflict-free nor based on maximal conflict. *Philosophical Transactions of the Royal Society B: Biological Sciences*, *371* , 20150091. doi: 10.1098/rstb.2015.0091

Cameron, E. Z., Setsaas, T. H., & Linklater, W. L. (2009). Social bonds between unrelated females increase reproductive success in feral horses. *Proceedings of the National Academy of Sciences*, *106* , 13850-13853. doi: 10.1073/PNAS.0900639106

Carlson, N. V., Kelly, E. M. K., & Couzin, I. D. (2020). Individual vocal recognition across taxa: A review of the literature and a look into the future. *Philosophical Transactions of the Royal Society B: Biological Sciences*, *375* , 20190479. doi: 10.1098/rstb.2019.0479

Carter, G. G., & Wilkinson, G. S. (2013). Food sharing in vampire bats: Reciprocal help predicts donations more than relatedness or harassment. *Proceedings of the Royal Society B: Biological Sciences, 280* , 20122573. doi: 10.1098/rspb.2012.2573

Carter, G. G., & Wilkinson, G. S. (2016). Common vampire bat contact calls attract past food-sharing partners. *Animal Behaviour* , *116* , 45-51. doi: 10.1016/J.ANBEHAV.2016.03.005

Chivers, D. P., Brown, G. E., & Smith, R. J. F. (1995). Familiarity and shoal cohesion in fathead minnows (pimephales promelas): implications for antipredator behaviour. *Canadian Journal of Zoology* , *73* , 955-960. doi: 10.1139/z95-111

Coffin, H. R., Watters, J. V., & Mateo, J. M. (2011). Odor-based recognition of familiar and related conspecifics: A first test conducted on captive humboldt penguins (spheniscus humboldti). *PLoS ONE* , *6* , e25002. doi: 10.1371/journal.pone.0025002

Connor, R. C., Mann, J., Tyack, P. L., & Whitehead, H. (1998). Social evolution in toothed whales. *Trends in Ecology and Evolution*, *13* , 228-232. doi: 10.1016/S0169-5347(98)01326-3

Couzin, I. D. (2006). Behavioral ecology: Social organization in fission–fusion societies. *Current Biology* , *16* , R169-R171. doi: 10.1016/j.cub.2006.02.042

Couzin, I. D., & Krause, J. (2003). Self-organization and collective behavior in vertebrates. *Advances in the Study of Behavior, 32,* 1-75.

Couzin, I. D., Krause, J., James, R., Ruxton, G. D., & Franks, N. R. (2002). Collective memory and spatial sorting in animal groups. *Journal of Theoretical Biology*, *218* , 1-11. doi: 10.1006/yjtbi.3065

Csardi, G., & Nepusz, T. (2006). The igraph software package for complex network research. *InterJournal, Complex Systems*, 1695.

Farine, D. R., Montiglio, P.-O., & Spiegel, O. (2015). From individuals to groups and back: The evolutionary implications of group pheno- typic composition. *Trends in Ecology and Evolution*, *30*, 609-621. doi: 10.1016/j.tree.2015.07.005

Fotouhi, B., Momeni, N., Allen, B., & Nowak, M. A. (2019). Evolution of cooperation on large networks with community structure. *Journal of The Royal Society Interface*, *16* , 20180677. doi: 10.1098/rsif.2018.0677

Gherardi, F., Aquiloni, L., & Tricarico, E. (2012). Revisiting social recognition systems in invertebrates. *Animal Cognition*, *15* , 745-762. doi: 10.1007/s10071-012-0513-y

Griffiths, S. W., Brockmark, S., Hojesjo, J., & Johnsson, J. I. (2004). Coping with divided attention: The advantage of familiarity. *Proceedings of the Royal Society B: Biological Sciences*, *271* , 695-699. doi: 10.1098/rspb.2003.2648

Griffiths, S. W., & Ward, A. (2011). Social recognition of conspecifics. Fish cognition and behavior, 2, 186-216.

Grimm, V., Berger, U., Bastiansen, F., Eliassen, S., Ginot, V., Giske, J., . . . DeAngelis, D. L. (2006). A standard protocol for describing individual-based and agent-based models. *Ecological Modelling* , *198* , 115-126. doi: 10.1016/j.ecolmodel.2006.04.023

Grimm, V., Railsback, S. F., Vincenot, C. E., Berger, U., Gallagher, C., Deangelis, D. L., . . . Ayllon, D. (2020). The odd protocol for describing agent-based and other simulation models: A second update to improve clarity, replication, and structural realism. *Journal of Artificial Societies and Social Simulation*, *23* , 7. doi: 10.18564/jasss.4259

Ifti, M., Killingback, T., & Doebeli, M. (2004). Effects of neighbourhood size and connectivity on the spatial continuous prisoner's dilemma. *Journal of Theoretical Biology*, *231,* 97-106. doi: 10.1016/j.jtbi.2004.06.003

Joshi, J., Couzin, I. D., Levin, S. A., & Guttal, V. (2017). Mobility can promote the evolution of cooperation via emergent self-assortment dynamics. *PLoS Computational Biology*, *13*, e1005732. doi: 10.1371/journal.pcbi.1005732

Katz, Y., Tunstrøm, K., Ioannou, C. C., Huepe, C., & Couzin, I. D. (2011). Inferring the structure and dynamics of interactions in schooling fish. *PNAS*, *108*, 18720-18725. doi: 10.1073/pnas.1107583108/-/DCSupplemental

Kays, R., Crofoot, M. C., Jetz, W., & Wikelski, M. (2015). Terrestrial animal tracking as an eye on life and planet. *Science*, *348* , aaa2478. doi: 10.1126/SCIENCE.AAA2478/SUPPL$_F$ *ILE/KAY S.SM.PDF*

Kruschke, J. (2010). *Doing bayesian data analysis: A tutorial introduction with R*. Elsevier Science.

Kun, A. & Scheuring, I. (2008). Evolution of cooperation on dynamical graphs. *BioSystems*, *96* , 65-68. doi: 10.1016/j.biosystems.2008.11.009

Lai, C. D. (2010). Generalized erlang and mortality levelling off distributions. *Mathematical and Computer Modelling*, *51*, 1268-1276. doi: 10.1016/j.mcm.2010.01.011

Lee, J.-S., Filatova, T., Ligmann-Zielinska, A., Hassani-Mahmooei, B., Stonedahl, F., Lorscheid, I., . . . Parker, D. C. (2015). The complexities of agent-based modeling output analysis. *Journal of Artificial Societies and Social Simulation*, *18*. doi: 10.18564/jasss.2897

Lorig, F., Johansson, E., & Davidsson, P. (2021). Agent-based social simulation of the COVID-19 pandemic: A systematic review. *JASSS: Journal of Artificial Societies and Social Simulation, 24*(3).

Luke, S., Cioffi-Revilla, C., Panait, L., Sullivan, K., & Balan, G. (2005). Mason: A multiagent simulation environment. *Simulation, 81* , 517-527. doi: 10.1177/0037549705058073

Luke, S., Simon, R., Crooks, A., Wang, H., Wei, E., Freelan, D., ... & Cioffi-Revilla, C. (2019). The MASON simulation toolkit: past, present, and future. In Multi-Agent-Based Simulation XIX: 19th International Workshop, MABS 2018, Stockholm, Sweden, July 14, 2018, Revised Selected Papers 19 (pp. 75-86). Springer International Publishing.

Marcoux, M., & Lusseau, D. (2013). Network modularity promotes cooperation. *Journal of Theoretical Biology, 324*, 103-108. doi: 10.1016/j.jtbi.2012.12.012

Marshall, J. A. R. (2011). Group selection and kin selection: formally equivalent approaches. *Trends in Ecology and Evolution, 26*, 325-332. doi: 10.1016/j.tree.2011.04.008

Massen, J., Sterck, E., & de Vos, H. (2010). Close social associations in animals and humans: functions and mechanisms of friendship. *Behaviour, 147*, 1379-1412. doi: 10.1163/000579510X528224

McLane, A. J., Semeniuk, C., McDermid, G. J., & Marceau, D. J. (2011). The role of agent-based models in wildlife ecology and management. E*cological modelling, 222*(8), 1544-1556.

Milinski, M., Luthi, J. H., Eggler, R., & Parker, G. A. (1997). Cooperation under predation risk: Experiments on costs and benefits. *Proceedings of the Royal Society B: Biological Sciences, 264*, 831-837.

Moran, P. A. P. (1962). *The statistical processes of evolutionary theory.* Clarendon Press.

Nowak, M. A. (2006). Five rules for the evolution of cooperation. *Science, 314* , 1560-1563. doi: 10.1126/science.1133755

Ohtsuki, H., Pacheco, J. M., & Nowak, M. A. (2007). Evolutionary graph theory: breaking the symmetry between interaction and replacement. *Journal of Theoretical Biology, 246* , 681-694.

Okasha, S. (2006). *Evolution and the levels of selection*. Oxford University Press. doi: 10.1093/acprof:oso/9780199267972.001.0001

Oscar, L., Li, L., Gorbonos, D., Couzin, I. D., & Gov, N. S. (2023). A simple cognitive model explains movement decisions in zebrafish while following leaders. *Physical Biology*, *20*. doi: 10.1088/1478-3975/ACD298

Parker, J., & Epstein, J. M. (2011). A distributed platform for global-scale agent-based models of disease transmission. *ACM Transactions on Modeling and Computer Simulation*, *22*, 2. doi: 10.1145/2043635.2043637

Perc, M., & Szolnoki, A. (2010). Coevolutionary games-a mini review. *BioSystems*, *99*, 109-125. doi: 10.1016/j.biosystems.2009.10.003

Pinter-Wollman, N., Hobson, E. A., Smith, J. E., Edelman, A. J., Shizuka, D., Silva, S. D., . . . McDonald, D. B. (2014). The dynamics of animal social networks: analytical, conceptual, and theoretical advances. *Behavioral Ecology*, *25* , 242-255. doi: 10.1093/BEHECO/ART047

Pyke, G. H. (1984). Optimal foraging theory: A critical review. *Annual Review of Ecological Systems*, *15*, 523-75.

Queller, D. C. (2011). Expanded social fitness and hamilton’s rule for kin, kith, and kind. *Proceedings of the National Academy of Sciences of the United States of America*, *108*, 10792-10799. doi: 10.1073/pnas.1100298108

Rodrigues, A. M., & Kokko, H. (2016). Models of social evolution: Can we do better to predict ‘who helps whom to achieve what’? *Philosophical Transactions of the Royal Society B: Biological Sciences*, *371* , 20150088. doi: 10.1098/rstb.2015.0088

Santos, F. C., & Pacheco, J. M. (2005). Scale-free networks provide a unifying framework for the emergence of cooperation. *Physical Review Letters*, *95*, 098104. doi: 10.1103/PhysRevLett.95.098104

Silk, M. J., Croft, D. P., Tregenza, T., & Bearhop, S. (2014). The importance of fission-fusion social group dynamics in birds. *Ibis*, *156*, 701-715. doi: 10.1111/IBI.12191

Smaldino, P. E., Schank, J. C., & McElreath, R. (2013). Increased costs of cooperation help cooperators in the long run. *American Naturalist*, *181*, 451-463. doi: 10.1086/669615

Smith, J. M. (1964). Group selection and kin selection (and a reply). *Nature*, *201*, 1145-1147. doi: 10.1038/2011145a0

Sueur, C., Petit, O., & Deneubourg, J. L. (2010). Short-term group fission processes in macaques: a social networking approach. *Journal of Experimental Biology*, *213* , 1338-1346. doi: 10.1242/JEB.039016

Sueur, C., Romano, V., Sosa, S., & Puga-Gonzalez, I. (2018). Mechanisms of network evolution: a focus on socioecological factors, intermediary mechanisms, and selection pressures. *Primates 2018 60:3, 60,* 167-181. doi: 10.1007/S10329-018-0682-7

Szabo, G., & Fath, G. (2007). Evolutionary games on graphs. *Physics Reports*, *446*, 97-216.

Szipl, G., Boeckle, M., Wascher, C. A., Spreafico, M., & Bugnyar, T. (2015). With whom to dine? ravens' responses to food-associated calls depend on individual characteristics of the caller. *Animal Behaviour*, *99,* 33-42. doi: 10.1016/J.ANBEHAV.2014.10.015

Thiele, J. C., Kurth, W., & Grimm, V. (2014). Facilitating parameter estimation and sensitivity analysis of agent-based models: A cookbook using Netlogo and R. *Journal of Artificial Societies and Social Simulation*, *17*, 11. doi: 10.18564/JASSS.2503

Traulsen, A., & Nowak, M. A. (2006). Evolution of cooperation by multilevel selection. *Proceedings of the National Academy of Sciences, 103,* 10952-10955. doi: 10.1073/PNAS.0602530103

Wal, E. V., Gagne-Delorme, A., Festa-Bianchet, M., & Pelletier, F. (2016). Dyadic associations and individual sociality in bighorn ewes. *Behavioral Ecology*, *27*, 560-566. doi: 10.1093/beheco

Ward, A., Webster, M., Ward, A., & Webster, M. (2016). Mechanisms: social recognition and social organisation. Sociality: the behaviour of group-living animals, 9-27.

Ward, A. J., & Hart, P. J. (2005). Foraging benefits of shoaling with familiars may be exploited by outsiders. *Animal Behaviour*, *69*, 329-335. doi: 10.1016/j.anbehav.2004.06.005

Webster, M. M., & Hart, P. J. (2007). Prior association reduces kleptoparasitic prey competition in shoals of three-spined sticklebacks. *Animal Behaviour*, *74,* 253-258. doi: 10.1016/j.anbehav.2006.07.021

Wilkinson, G. (1988). Reciprocal altruism in bats and other mammals. *Ethology and sociobiology*, *9*, 85-100.

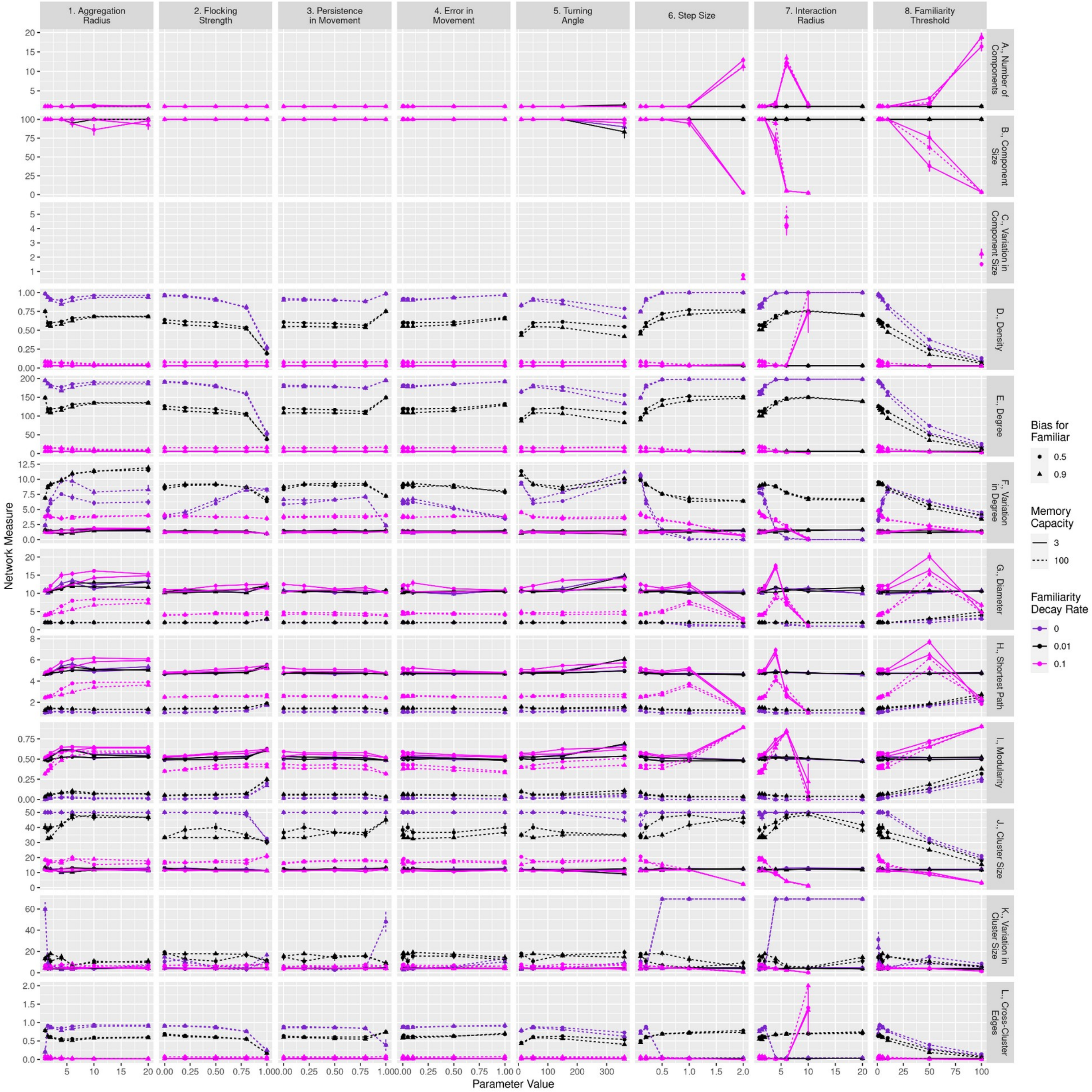
Wilkinson, G. S. (1984). Reciprocal food sharing in the vampire bat. *Nature*, *308*, 181-184. doi: 10.1038/308181a0

Wilkinson, G. S. (1990). Food sharing in vampire bats. *Scientific American*, *262*, 76-83. doi: 10.2307/24996685

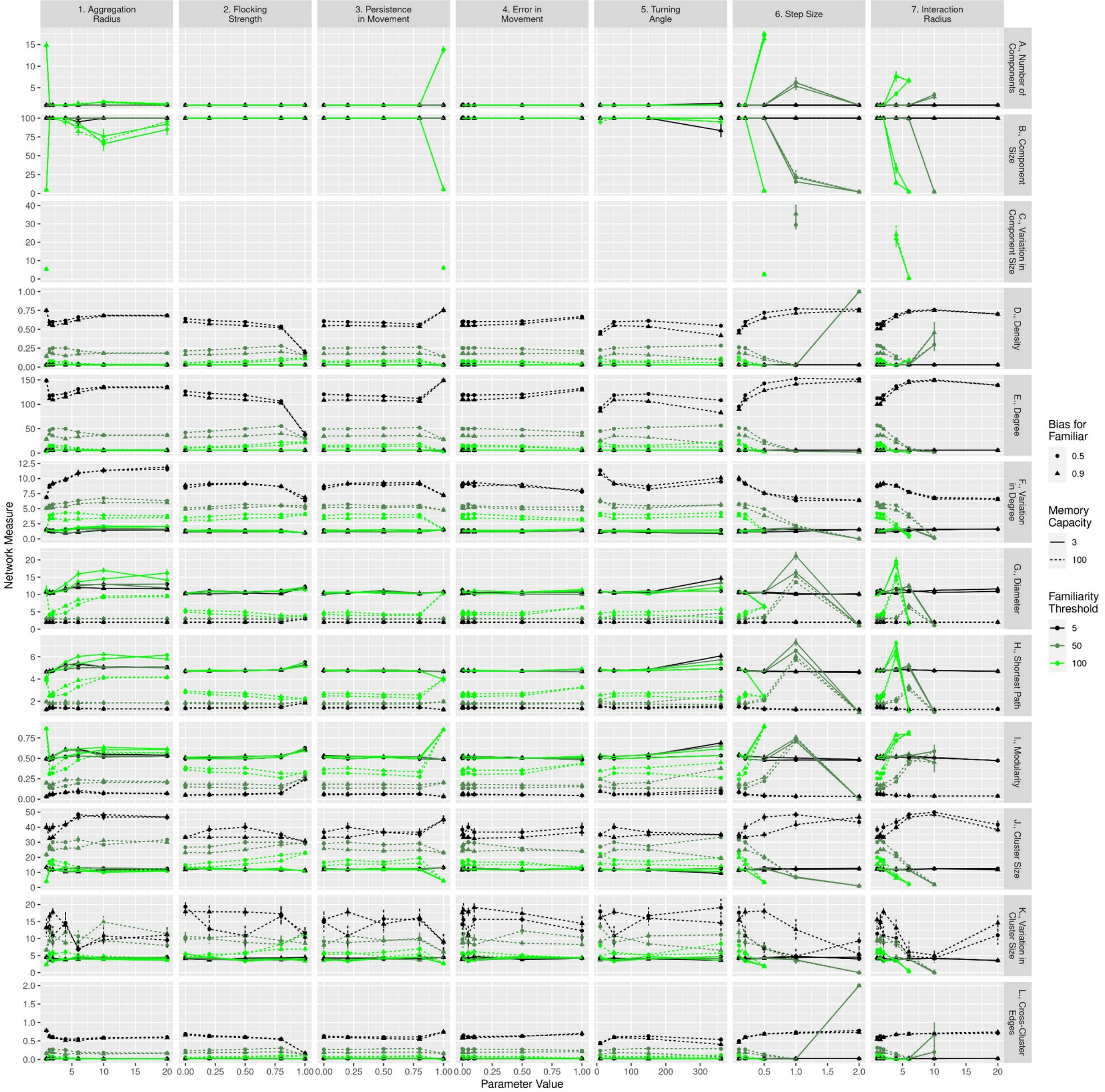
Wilkinson, G. S., Carter, G. G., Bohn, K. M., Caspers, B., Chaverri, G., Farine, D. R., . . . Patriquin, K. J. (2019). Kinship, association, and social complexity in bats. *Behavioral Ecology and Sociobiology*, *73*, 7. doi: 10.1007/S00265-018-2608-1

Wilson, D. S. (1975). A theory of group selection. *Proceedings of the National Academy of Sciences*, *72*, 143-146.

Supplemental Results



Supplementary Figure 1. Number of components, component size, standard error in component size, density, average degree, standard error in degree, diameter, average shortest path, modularity, average cluster size, and proportion of cross-cluster edges of familiarity networks after 5,000 steps, averaged over 10 replicates for a range of values of aggregation radius, strength of flocking versus aggregation, strength of persistence in movement, random error in movement, maximum turning angle, step size, interaction radius, and familiarity threshold, with no bias (.5; circles) and strong bias (.9; triangles) for familiar individuals, limited memory capacity (3; solid line) and unlimited memory (100; dashed), and no familiarity decay (0; dark purple), low familiarity decay rate (.01; black), and high familiarity decay rate (.1 magenta).



Supplementary Figure 2. Number of components, component size, standard error in component size, density, average degree, standard error in degree, diameter, average shortest path, modularity, average cluster size, and proportion of cross-cluster edges of familiarity networks after 5,000 steps, averaged over 10 replicates for a range of values of aggregation radius, strength of flocking versus aggregation, strength of persistence in movement, random error in movement, maximum turning angle, step size, and interaction radius, with no bias (.5; circles) and strong bias (.9; triangles) for familiar individuals, limited memory capacity (3; solid line) and unlimited memory (100; dashed), and low familiarity threshold (5; black), intermediate familiarity threshold (50; dark green), and high familiarity threshold (100 bright green).