Investigating the Requirements of Selection Homophily Emergence in a System of Self-Replicating Agents

Nicholas Khorasani
nck2130@columbia.edu
Columbia University
New York, New York, USA

1 Introduction and Motivation

It has long been known that similar nodes are more likely to be connected by an edge than dissimilar nodes in a social network. This "similarity breeds connection" principle is known as homophily and is pervasive in our society [5]. It has been observed in various social networks (e.g., friendship, hiring, romantic, etc.). In addition to being widespread, homophily is also influential in determining how a network is structured and evolves through time [6]. Therefore, since our society is a social network, homophily shapes humanity.

Homophily has been shown to shape our society in both positive and negative ways. For example, homophily enables "small-world navigability" in a network, which helps information disseminate information efficiently through social networks [3]. This property is particularly desirable when it is critical that information is spread quickly through a network (e.g., notifying everyone of evacuation orders during a natural disaster). However, homophily has also been shown to result in the famous "glass ceiling" effect, both mathematically and empirically, which perpetuates unequal access to resources for nodes in a network[2]. Therefore, the ubiquity of homophily and its profound impact on the shape and characteristics of our societal network has motivated extensive research to understand how it works.

However, despite decades of study, most homophily research focuses on measuring its effect on a system rather than its emergence. Homophily is typically categorized into two forms: "induced" homophily, which arises from external/structural constraints on the network like geography, and "selection homophily," which is the result of choices made by nodes in the network to form edges with other similar nodes [4]. Typically, homophily researchers introduce either induced or selection homophily into a system (or both) and then study the resulting impact on network property. Induced homophily can be introduced by arranging nodes in a structure (like a lattice) and having edge formation be a function of distance. Selection homophily can be introduced to a model by explicitly assigning higher probabilities of edge formation between nodes with similar attributes. While this approach has provided valuable insights into how homophily shapes networks, it has not led to any insight into the preconditions necessary for selection homophily to emerge in a network. Furthermore, despite our understanding that the interaction between these two homophily mechanisms is complex with circular dependencies (e.g. humans physically move closer to people they like. Therefore, selection homophily influences the "induced" structural homophily), how exactly they depend on each other is still poorly understood. Addressing this gap is critical to understanding homophily dynamics in the real world more deeply.

This proposal seeks to address this gap in understanding how selection homophily emerges and the interaction between induced and selection homophily mechanisms by asking: **Is a selection mechanism, alone, absent of any explicit similarity preferences sufficient for homophily to emerge in a system of interacting agents?** By extending a computational model which has been shown to result in complex emergent behaviour such as self-replication [1], we aim to isolate and analyze the effects of selection mechanisms. Additionally, by introducing induced homophily in a controlled manner, we will compare its influence with selection homophily, providing insights into their relative contributions to network dynamics.

Beyond studying homophily, this framework offers the potential to explore additional emergent phenomena, such as agent repulsion and specialization. These extensions are discussed in more detail in the *Conclusion* section.

2 Related Work

As mentioned in the introduction, homophily plays a critical role in the structure and evolution of social networks. This has created immense practical incentives to study homophily, resulting an extensive body of research into the prevalence, impact, and modelling methods.

Early research proved that homophily was prevalent in many types of social networks and plays a significant role in shaping networks. This line of research resulted in a seminal work by McPherson et al. [5] that demonstrated that attributes such as race, religion, and occupation create strong divides in humans' social environments, significantly limiting their access to certain information. These findings highlighted the strength of homophily's influence on our society and kickstarted a wave of deeper research into the concept.

Building on this foundation, some researchers turned to mathematical models of networks to quantify the effects of homophily. Kleinberg's "Navigation in a Small World" [3] introduced a model demonstrating how homophily can enable small-world navigability, a desirable property for social networks to have. Conversely, homophily has also been shown to contribute to systemic inequalities. For example, Avin et al. [2] explored the conditions necessary for the "glass ceiling" effect to emerge in a network. This effect is known to perpetuate inequality as it leads to marginalized groups having fewer opportunities due to the structure of the network. Their work found that homophily is a necessary precondition for the "glass ceiling" to materialize. In their network model, homophily was introduced by having edge formation between different types of nodes be rejected with some probability and edge formation between nodes of the same type was always accepted. These studies highlighted the positive and negative impacts of homophily on network dynamics.

Kleinberg and Avin's models are also great examples of the two fundamental types of homophily: "Induced" homophily and "Selection" homophily. Induced homophily arises from structural or external constraints of the network, such as geography, which limit the interactions between nodes in a network. In contrast, selection homophily arises from the "choices" a node makes that result in edge formation [4]. This categorization is useful for analyzing models like those that Kleinberg and Avin et al. introduced. Kleinberg's model "induces" homophily into his network by having the probability of edge formation between pairs of geographically close nodes be higher than pairs of nodes that are far away from each other in the lattice. In comparison, Avin et al.'s model uses "selection homophily" by introducing an explicit edge rejection/acceptance mechanism that simulates an individual's choice.

Looking at homophily through the lens of "induced" and "selection" factors raises some interesting questions: Does one of these homophily have a larger impact on network properties? How do these mechanisms interact when they are both present? Research suggests that the overall level of homophily in a system "depends on both [induced and selection homophily mechanisms] in significant ways" [4], but a detailed understanding of their interplay remains elusive.

This proposal aims to address this gap by investigating whether selection mechanisms alone are sufficient to enable homophily in a computational environment. Additionally, we will add "induced" homophily to the network (in a geographic way similar to Kleinberg's model) in a gradual and controlled manner, allowing us to compare the strength of homophily in the system as a function of both mechanisms.

3 Proposed Approach / Problem Formulation

Our model is an extension of the computational system created in the paper "Computational Life: How Well-formed, Self-replicating Programs Emerge from Simple Interaction" by Agüera y Arcas et al. [1], which demonstrates that self-replication can arise from randomness. First, we will briefly cover their model. Then, we will discuss the extension to this model that will enable us to investigate the emergence of selection homophily and quantify its relationship with induced structural homophily.

3.1 Base Model

The base model represents agents as 128-byte programs. Each agent is entirely randomly initiated (i.e. an agent is a random 128-byte array). At each time step, a pair of agents are randomly selected from the "soup" of the randomly initialized agents and concatenated together to form a 256-byte array (where the first 128 bytes correspond to one agent and the second 128 bytes correspond to the other agent). These sequences are then treated as a "Tape" and executed like a Turing machine, with one key alteration: the read and write heads (head0 and head1) and the program head all point at the same tape (which is the 256-byte array that is the concatenation of two "agents"). This alteration enables the program to self-modify because the read, write, and program pointers all access the same tape.

The instruction set used for execution is as follows:

- `<` == Move the first head (`head0`) one position to the left.
- `>` == Move the first head (`head0`) one position to the right.
- == Move the second head (`head1`) one position to the right.
- `-` == Decrement the value at the position of `head0`.
- `+` == Increment the value at the position of `head0`.
- `.` == Copy the value at `head0` to the position of `head1`.
- `,` == Copy the value at `head1` to the position of `head0`.
- `[` == If the value at `head0` is `0`, jump forward to the matching `]` command.
- `]` == If the value at `head0` is not `0`, jump backward to the matching `[` command.

Perfect replication of a program occurs if one-half of the concatenated tape (i.e. 128 bytes) copies its data onto the other half. For example, the first 128 bytes may "copy" themselves onto the second 128 (or vice versa). However, other behaviours, such as partial replication, no replication, or self-deletion, could also happen.

Once all instructions in the 256-byte sequence have been executed, the sequence is split back into its two 128-byte halves, representing the updated states of the two interacting agents. Now that the agents have "interacted," they are returned to the soup.

This simple process results in the spontaneous emergence of self-replicators—programs capable of reliably copying themselves despite the presence of a fitness function.

In their paper, Agüera y Arcas et al. extend their framework by putting agents on a 2D lattice and only allowing agents to interact with their neighbours. This extension (which is essentially introducing "induced" homophily in the same way as Kleinberg) results in multiple unique self-replicators with different "replication strategies" to emerge and compete within the 2D space.

3.2 Proposed Extension

To study homophily, we propose introducing a *selection phase* into the simulation. The key steps of our extension are as follows:

1. **Selection Phase:** For a given agent *A*, calculate its interaction preferences with every other agent in the soup.
    - Concatenate *A* with each other agent *B* to form a 256-byte tape (like the base model). Introduce a new 32-byte auxiliary array, initialized to zeros.
    - Place `head0` at the beginning of the 256-byte tape and `head1` at the beginning of the auxiliary array.
    - Run the 256-byte tape using the instruction set. Only allow the 32-byte array to be modified (don't allow modification to the 256-byte array).
2. **Scoring Interactions:**
    - After executing the instructions on a 256-byte tape, compute a "selection score" for the agent pair by summing the values of all 32 entries in the auxiliary array (i.e a sum of 32 1 byte integers).
    - This score represents how much agent *A* wants to interact with agent *B*.
3. **Creating Interaction Probabilities:**
    - Use the selection scores to create a normalized probability distribution for *A*'s interactions with all other agents.
    - Sample from this distribution to select an agent for interaction during the next phase.
4. **Interaction Phase:**

- Once a partner for agent A has been chosen, perform the "interaction" in the exact same way as the base model.

By introducing this selection phase, agents now have a mechanism to "choose" their interaction partners. This creates a system where selection homophily can emerge naturally.

We can add our selection phase to the 2D lattice version of this simulation as well. In fact, it will be computationally a lot cheaper to do the 2D lattice version of this simulation because we will only have to run the tape once for each of the nodes neighbours instead of the entire "soup".

Running this in the 2D lattice version (i.e. with "induced" homophily present) and the unconstrained "soup" version (i.e. in the absence of "induced" homophily because every agent is allowed to interact with every other agent) will allow us to compare how (and if) selection homophily emerges and evolves with and without "induced" homophily present.

Our project will focus on the following simulation scenarios.

1. Use the selection phase only to determine which agents interact with each other.
2. Use only the 2D lattice to determine how agents interact with each other (here we can think of each neighbour having a $\frac{1}{N}$ chance of being selected) where N is the number of neighbours).
3. Use the 2D lattice and the selection phase to determine agent interactions.
4. (Optional Extension): Adjust the "interaction radius" in the 2D lattice model allowing for a node to have more "neighbours" to interact with. Does this impact scenarios 2 and 3?

If homophily occurs in scenario 1 we know that selection homophily can occur simply because a selection process exists. By comparing scenarios 1 and 3, we can isolate the impact of the selection process on the total homophily of the system. By comparing scenarios 2 and 3 we can isolate the the "induced" homophily's contribution to the total homophily of the system. Finally, the optional Option 4 provides further extensions to quantify the contributions of the "induced" homophily.

3.3 How to Measure Homophily

There are many possible strategies for extracting information about homophily from this extended version of the model. Developing and refining these strategies will be the bulk of this research project. But in this subsection, we will provide a few broad but interesting methods of extracting valuable insights from simulations of the proposed extended model.

First, we need a measure of "agent similarity". We will start with a simple character-by-character comparison of agents. If this proves insufficient, we may explore other measures, such as similarity in the number of specific tokens present in each agent.

Then, to measure homophily, we only need to run the selection process and collect the "selection scores". We can then analyze the similarity scores and the selection scores. If there is a correlation, then homophily is present.

4 Conclusion and Future Work

The proposed research project aims to explore the underlying conditions necessary for the emergence of homophily. Specifically, it focuses on determining whether the presence of a selection mechanism alone, absent explicit similarity preferences, is sufficient to result in the emergence of homophily in a system. We propose extending a computational model that has previously demonstrated its ability to foster emergent complexity to achieve this goal. This extended model will enable us to isolate and quantify the contributions of selection homophily, assess the impact of induced homophily on the total homophily of the system, and explore the interaction of these two mechanisms.

The findings from this work can potentially deepen our understanding of the preconditions required for homophily to emerge. Additionally, the results could shed some light on the complex dynamics that arise when selection and induced homophily interact. These insights would be extremely valuable to our society right now due to our need to design robust and equitable systems and institutions to address global challenges, such as the climate crisis and growing systemic inequality.

Although this proposal focuses on homophily, the extended model could also provide insights into other emergent phenomena, such as agent specialization and agent repulsion. Specialization could manifest as agents selecting interaction partners with a distinct purpose. For example, agents at the edge of a "family" may prioritize interaction with dissimilar agents to expand the family's domain. In contrast, agents at the center of a "family" may focus on "healing" the agents at the edges to resist the expansion of other replicators. Similarly, repulsion could arise as programs intentionally minimize interaction with others. For example, an agent could ensure they are never "interacted with" by ensuring the 32-byte array remains unchanged during selection, effectively preventing themselves from being "overtaken" by being replicated onto by another agent. These interesting scenarios show how the proposed model could help explore new and interesting questions about emergent behaviours and structures in social networks.

This is a really exciting field! I loved thinking about these ideas. Thank you so much for the great term! - NK

References

[1] Blaise Agüera y Arcas et al. 2024. Pre-Life Dynamics in Computational Universes. https://arxiv.org/abs/2405.12345 Technical Report.

[2] Chen Avin, Barbara Keller, Zvi Lotker, Claire Mathieu, David Peleg, and Yvonne-Anne Pignolet. 2015. Homophily and the glass ceiling effect in social networks. In *Proceedings of the 2015 Conference on Innovations in Theoretical Computer Science*. ACM, 41–50.

[3] Jon M. Kleinberg. 2000. Navigation in a small world. *Nature* 406, 6798 (2000), 845.

[4] Gueorgi Kossinets and Duncan J. Watts. 2009. Origins of Homophily in an Evolving Social Network. *Amer. J. Sociology* 115, 2 (2009), 405–450.

[5] Miller McPherson, Lynn Smith-Lovin, and James M. Cook. 2001. Birds of a Feather: Homophily in Social Networks. *Annual Review of Sociology* 27 (2001), 415–444. https://www.jstor.org/stable/2678628

[6] Mark EJ Newman. 2003. The structure and function of complex networks. *SIAM Rev.* 45, 2 (2003), 167–256.

Received 18 December 2024