Proceedings of the 2024 Winter Simulation Conference
H. Lam, E. Azar, D. Batur, S. Gao, W. Xie, S. R. Hunter, and M. D. Rossetti, eds.

ACCELERATING HYBRID AGENT-BASED MODELS AND FUZZY COGNITIVE MAPS: HOW TO COMBINE AGENTS WHO THINK ALIKE?

Philippe J. Giabbanelli[1] and Jack T. Beerman[2]

[1]Virginia Modeling, Analysis and Simulation Center, Old Dominion University, Norfolk, VA, USA
[2]School of Data Science, University of Virginia, Charlottesville, VA, USA

ABSTRACT

While Agent-Based Models can create detailed artificial societies based on individual differences and local context, they can be computationally intensive. Modelers may offset these costs through a parsimonious use of the model, for example by using smaller population sizes (which limits analyses in sub-populations), running fewer what-if scenarios, or accepting more uncertainty by performing fewer simulations. Alternatively, researchers may accelerate simulations via hardware solutions (e.g., GPU parallelism) or approximation approaches that operate a tradeoff between accuracy and compute time. In this paper, we present an approximation that combines agents who 'think alike', thus reducing the population size and the compute time. Our innovation relies on representing agent behaviors as networks of rules (Fuzzy Cognitive Maps) and empirically evaluating different measures of distance between these networks. Then, we form groups of think-alike agents via community detection and simplify them to a representative agent. Case studies show that our simplifications remain accuracy.

1 INTRODUCTION

Agent-Based Modeling (ABM) serves to represent physical entities as virtual entities in a simulation for a plethora of real world scenarios. These entities known as 'agents' can imitate living organisms such as cells or physical objects such as cities. Furthermore, ABMs are constructed with specific rule sets and laws governing the actions of agents depending on the problem that modelers are attempting to analyze. Regardless of the problem at hand, there is a balance to be struck between the size of the agent population and the resources required for computation. While improving model resolution (i.e., a simulated agent represents fewer real-world individuals) allows for detailed analyses in sub-groups, we must find a compromise given the limited resources such as compute time. This can be achieved through parallelism (Băbeanu, Filatova, Kwakkel, and Yorke-Smith 2023) and/or hardware accelerators such as GPUs (Lysenko and D'Souza 2008; Ghumrawi et al. 2022) or FPGA (Xiao et al. 2019). However, modelers may not possess such resources or the programming skills to efficiently use them. An alternative is to group agents. Such groups can be achieved by *locally representative agents*, also known as 'super agents' or 'super nodes' (Lippe, Bithell, Gotts, et al. 2019; Parry and Evans 2008; Parry and Bithell 2012). Agents can also be aggregated/disaggregated via a higher level of abstraction such as a continuum model (Cilfone, Kirschner, and Linderman 2015; Minucci, Heise, and Reynolds 2024). In this paper, we focus on creating super agents to reduce the size of a model and accelerate simulations.

In order to find an agent whose behavior is representative of its group, we need to find a meaningful group (e.g., via community detection) and *compare behaviors*. When the behavior of an agent is primarily a function of its individual traits (e.g., agents have different age or socio-economic status), the comparison is based on the distance between the vectors of traits, as exemplified by the Axelrod model of culture. However, real-world individuals with the same traits can still engage in different behaviors. This is not due to randomness, but to the fact that *people follow different rules*: even if two seemingly indistinguishable individuals are presented with the same evidence, they may reach different conclusions. We thus use a

hybrid modeling technique that combines ABMs with Fuzzy Cognitive Maps (FCMs) (Davis, Jetter, and Giabbanelli 2020). Intuitively, each agent has its own rule set (as shown with two agents in Figure 1), which is a simulation model consisting of a directed network with labeled nodes and weighted, directed edges. Node values range from 0 (absence of a concept) to 1 (full presence of a concept). Edge values range from -1 (an increase in the source *decreases* the target) to 1 (an increase in the source *increases* the target). An agent's view of causation is held constant hence edge weights do not change. Observations and circumstances change, hence node values are updated. After an agent interacts with peers or the environment, the observations provide an input to this FCM (similar to the 'virtual brain' of the agent), which iteratively updates the node weights until reaching stabilization for the next decision (as shown for agent *B* with two iterations in Figure 1). This hybrid method allows to represent the heterogeneity of human behaviors. It also illustrates the importance of a compromise since it carries a significant computational cost and even hardware accelerators are limited to populations of dozens of thousands of agents (Ghumrawi et al. 2022).

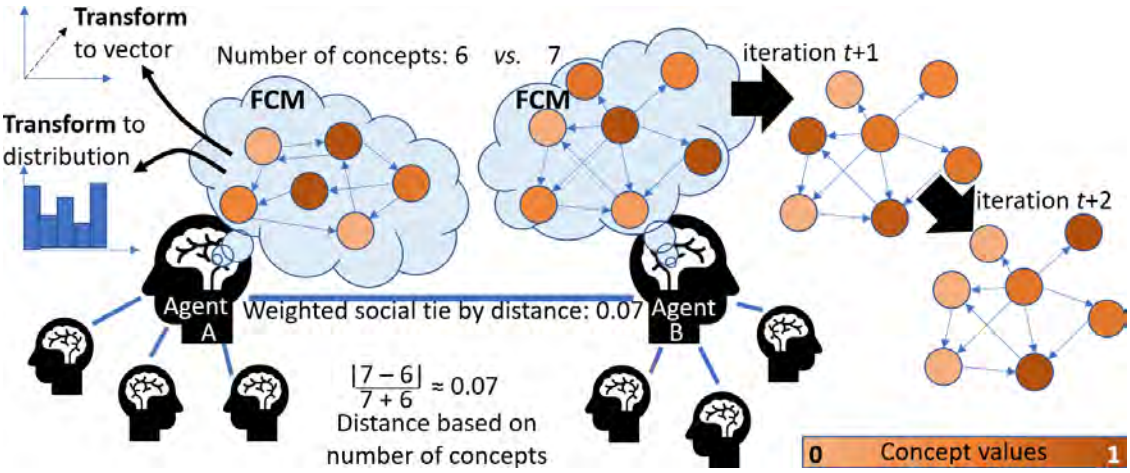

Figure 1: A hybrid ABM/FCM consists of agents who interact with each other (e.g., *A* interacts with *B* and each one has three other neighbors). Interactions are impacted by their 'mental model' in the form of an FCM, which is a network that performs simulations. As exemplified for agent *B*, simulating an FCM changes its node values (in the interval [0, 1]) over discrete iterations. To compare mental models, we can compare FCMs by transforming these networks into distributions (e.g., degree distribution) or vector embeddings, or on the basis of simple criteria such as the number of nodes. Each existing social tie is then weighted to reflect the similarity of mental models. Here, similarity in number of nodes is about 0.07.

Our main contribution is to reduce the population sizes of a hybrid ABM/FCM model by establishing representative agents within their communities while maintaining accuracy. This is achieved by implementing new and existing techniques to compare the mental models of agents, followed by community detection and aggregation. Ultimately our work contributes in two areas: *(i)* we reduce computational costs of existing hybrid models, and *(ii)* we introduce and evaluate new metrics to compare the behavior of agents.

Since behaviors are encoded through a network representation, we compare the behaviors of agents by comparing their FCMs. This becomes a matter of network differences, which can be expressed through network-level metrics (e.g., number of nodes) or by comparing vector- or distribution-based representation of the network (Figure 1). To keep the paper self-contained, we summarize classic network measures in Section 2.2 along with several new proposed measures. We also provide a succinct introduction to the hybrid ABM/FCM paradigm, which has been covered in more details at WinterSim previously. In Section 3 we describe the complete process to compare agents, starting with weighing social ties between agents represent the similarity/dissimilarity between their FCMs, and then running community detection algorithms

to find clusters of behaviors. Agents are selected from these clusters as representatives and reconnected to form smaller hybrid models that are simulated again. In Section 4, we compare the simulations from the smaller model with the original one, in order to assess the effect of simplification onto simulation fidelity.

2 BACKGROUD

2.1 A Brief Introduction to Hybrid Agent-Based Models and Fuzzy Cognitive Maps

The heterogeneity of real-world individuals stems not only from looking different or living in different contexts, but also because people *reflect* differently on facts and experiences. In their reflections, people may be vague or uncertain, leading to fuzzy notions ("if it's too sunny I'll put on some sunscreen") rather than crisp rules ("If UV Index Scale > 4 then apply 30 mg of SPF50 sunscreen"). As they make decisions, individuals can consider a variety of pros and cons, which can be interrelated or form patterns such as cycles. For instance, an individual whose family is facing a financial abyss may consider taking their own lives, which would result in even greater financial insecurity (and trauma) for their family. Fuzzy Cognitive Maps (FCMs) allow to transform the implicit mental models of individuals into explicit simulation models where factors (nodes) can be connected (edges), and these causal connections have numerical weights obtained via fuzzy logic. FCMs have been used in over 20,000 studies, often to model the perspectives of stakeholders in complex socio-environmental problems (Giabbanelli et al. 2024). FCMs have also been integrated to ABMs, where each agent can be equipped with an FCM that has a unique structure and unique concept values, resulting in heterogeneity of behaviors (Giabbanelli 2024). Computationally, an FCM is akin to a neural network, hence it is defined by (i) the network *structure* of concepts and directed, weighted causal edges, representing an agent's ruleset; (ii) the current values of the agent's concept, known as *activation vector*; and (iii) a *transfer function*. In the same way as a neural network, a simulation is performed by applying the transfer function repeatedly to update the concept values, until either reaching stabilization or exceeding a user-defined number of iterations.

2.2 Comparing Behaviors via Networks: New and Established Measures

We can implement a variety of metrics to characterize the Fuzzy Cognitive Map of an agent (Gray, Sterling, Aminpour, et al. 2019) and hence compare maps (Tchupo and Macht 2022; Wills and Meyer 2020). However, using the most appropriate metric is currently an open problem, hence this paper covers many alternatives to evaluate their impact empirically. Since the behavior of each agent is defined by their FCM, we consider metrics that are applicable to FCMs as well as metrics used for networks more broadly. The list of metrics is summarized in Table 1 along with their equation, while this section summarizes the logic of each metric, including established metrics in network science (to keep this paper self-contained). We start with simple and quickly computed metrics (density, number of concepts, receiver/transmitter ratio, clustering coefficient) and then present more computationally intensive measures.

Graph Density is the number of connections in a network with respect to the maximum number of connections possible for all concepts. It depicts the interwovenness of concepts within an FCM (Tchupo and Macht 2022). The density of a network is calculated depending on whether it is directed (as shown in Table 1) or not. The *number of concepts* simply compares the total number of nodes in a network to the total number of nodes in the other network (Tchupo and Macht 2022). Intuitively, it can capture that the behavior of one agent depends on more concepts than another agent. The *receiver-transmitter ratio* is computed based on the number of 'receiver nodes' (denoted as R), which serve exclusively as targets and have no outgoing edge, and the number of 'transmitter' nodes (denoted as T) that always act as a source and have no incoming edge (Tchupo and Macht 2022). The *clustering coefficient* measures the average density of a node's neighbors. That is, we measure the coefficient for each node by looking at the density in the subgraph limited to the node's neighbors. Then, we average the coefficient across the nodes in order to get a clustering coefficient at the level of the graph. The clustering coefficient of a node is given in Table 1 for the directed case. A node without neighborhood is a special case with a clustering of 0.

The measures above can extract one number for each FCM (e.g., number of nodes, graph density) thus two FCMs would be compared through two numbers. However, this may oversimplify important patterns. Instead of summarizing an FCM to a single number, we can thus extract more characteristics and convey them either through a vector or as a discrete distribution.

Weighted Jaccard similarity measures the distance between vectors, where all entries x_i and y_i are positive real numbers. First the similarity coefficient is determined through the summation of the minimum x_i and y_i values divided by the summation of the maximum x_i and y_i values. The *distance* is the inverse of the similarity. In network science, the Jaccard similarity is often defined between two nodes on the basis of the intersection and union of their neighborhood sets (Aljundi, Akyildiz, and Kaya 2022). However, we use the Jaccard similarity between discrete distributions (e.g., the degree distribution) at the network-level, rather than between sets of neighbors at the node-level. For example, our measure takes two FCMs' set of weighted edges and computes the Jaccard similarity coefficient. This allows us to compute a distance between the rulesets of agents.

The difference between two rule sets can be measured based on what is most 'important' to each agent. The importance of a node is its *centrality*, which produces a ranking from most to least important nodes. Common centrality measures include betweenness, closeness, and degree. To compare the rankings of two agents' FCMs, we consider rankings as vectors and perform a cosine similarity. The output ranges from -1 (least similar) to 1 (most similar) (Lahitani, Permanasari, and Setiawan 2016).

Graph kernels decompose a network into subtructures, such as a comprehensive inventory of all trees or loops. A common method is the Triad Significance Profile (TSP) that extracts all 16 possible subgraphs with three nodes (i.e., triads) (Milo, Itzkovitz, Kashtan, et al. 2004). Their significance is assessed statistically via the Z-score, denoted Z_M, where n_M represents the frequency of a triad M in the given network, and $\left\langle n_M^{\text{rand}} \right\rangle$ and σ_M^{rand} represent the mean and standard deviation of M in a set of equivalent random networks, respectively (Juszczyszyn 2018). In other words, the TSP shows for each sub-structure whether it occurs more or less than would be expected at random.

Kullback-Leibler Divergence (KL Divergence) can be used to compare distributions, such as graph kernels or the degree distribution (i.e., distribution of number of edges per node). Assuming that the characteristics of the FCMs for two agents have been summarized through distributions P and Q (which are statistically independent of each other), we then take the summation over all possible values of the random variable x for $Q(x)$ and $P(x)$ – that is, the probabilities assigned to each value x by the distributions Q and P. The output of this metric is always positive and will only equal to zero if the two distributions are identical. The greater the value of KL divergence between two distributions, the greater their difference.

The last statistical test to measure distributions is the *Kolmogorov-Smirnov* (KS) statistic, which computes the maximum difference between the two distributions and determines whether the select distributions are from the same cumulative distribution (Miasnikof, Shestopaloff, Bravo, and Lawryshyn 2023).

3 METHODS

3.1 First Step: Weighing Social Ties by the Expected Similarity of Behavioral Rules

A hybrid model is initialized by creating agents with their individual FCMs, who interact through social ties. We start by identifying whether interacting agents can be merged, thus operating a local simplification. We thus only measure the *similarity for each pair of interacting agents*. That is, each social tie will be weighted using one of the 11 measures (section 2.2): graph density, graph TSP, KL for node and edge weight distributions, Jaccard and KS for edge weight distributions, cosine similarity of betweenness/degree/closeness centrality, number of concepts, clustering coefficient, receiver-transmitter ratio, KL for node, and KS for edge. For example, assume that a modeler defines similarity based on the number of concepts. In Figure 1, agents A and B have $n_A = 6$ and $n_B = 7$ concepts respectively thus their social tie has a weight of $\frac{abs(n_A - n_B)}{(n_A + n_B)} \approx 0.07$, which indicates a small difference. If they had the same number of concepts, the

Table 1: Measures in previous work or implemented for FCM comparison for the first time. V and E are the node and edge sets of the network, respectively. N_i is the neighborhood of a node i. R and T are receivers and transmitters, respectively. A and B are centrality rankings. All measures come from (Tchupo and Macht 2022) at the exception of the last three, which we propose here to compare FCMs.

Measure	**Equation**
Graph Density	$D = \frac{\lvert E\rvert}{2\binom{\lvert V\rvert}{2}} = \frac{\lvert E\rvert}{\lvert V\rvert(\lvert V\rvert-1)}$
Graph Kernels (TSP)	$Z_M = (N_M - \langle N_M^{\text{rand}} \rangle)/\sigma_M^{\text{rand}}$
Number of Concepts	$\lvert V\rvert$
Clustering Coefficient	Average of $C_i = \frac{\lvert e_{jk}\rvert}{\lvert N_i\rvert\cdot(\lvert N_i\rvert-1)}, v_j, v_k \in N_i$ over all nodes i
Receiver-transmitter ratio	ratio $= \frac{R}{T}$
Weighted Jaccard similarity	$d_{JW}(x,y) = 1 - \frac{\sum_i min(x_i,y_i)}{\sum_i max(x_i,y_i)}$
Centralities and cosine similarity	$\frac{A\cdot B}{\lVert A\rVert\lVert B\rVert}$
Kullback-Leibler Divergence	$D(P\lVert Q) = \sum_{i=1}^{k} P_i \log \frac{P_i}{Q_i}$
Kolmogorov-Smirnov	$D_n = \sup_{x\in\mathbb{R}} \lvert F_n(x) - F(x)\rvert$

difference would be 0. As the gap grows, the value tends to 1. Note that the choice of a comparison metric is an *open problem*, hence we evaluate the impact of this choice experimentally in the next section.

Algorithm 1 Evaluate the impact of simplifying a set of agents A using similarity metric S and community detection D onto simulation outcomes. Outcome measures produced by the algorithm are shown in blue.

Run the simulation several times to account for stochasticity and produce distribution of outputs $D_{original}$
//Identify super-agents by weighing social ties, creating communities, and finding a median agent
for every pair of interacting agents $i, j \in A$ **do**
  Assign to the existing edge $e_{i,j}$ a weight via the similarity metric $S(FCM_i, FCM_j)$
Assign each agent to one of c non-overlapping balanced clusters from algorithm D using edge weights
for each cluster $i = 1 \ldots c$ **do**
  Calculate sum of each agent's FCM concept values
  Select as 'super-agent' a_i an agent with the median of these values to represent the community
//Reduce simulation model by linking super-agents, removing all other agents, and initializing super-agents
for each representative agent $a_i, i = 1 \ldots c$ **do**
  Create an edge between a_i and other agents $a_j, i \neq j$ from other clusters using existing edges
Measure the number of agents to remove $\lvert A \setminus \{a_1, \ldots, a_c\}\rvert$
Remove every non-representative agent $A \setminus \{a_1, \ldots, a_c\}$
for each representative agent $a_i, i = 1 \ldots c$ **do**
  Assign the initial values of FCM_{a_i}
//Run the simplified simulation model and compare results with the original model
Run the simulation several times to account for stochasticity and produce distribution of outputs $D_{simplified}$
Measure the KL divergence between distributions $D_{simplified}$ and $D_{original}$
Measure the statistical properties (mean, quartile ranges, std) of $D_{simplified}$ and $D_{original}$

3.2 Second Step: Identify Groups of Like-Minded Agents by Clustering the Weighted Social Ties

Our goal is to reduce a group to a single representative agent. Locality is important: even if an agent had a soulmate at a given time on the other side of the world, the two should not be fused since their interactions with different people/places poses a risk to diverge over time. We posit that people who think alike and are

embedded in the same context are more likely to remain similar throughout a simulation. Consequently, after assigning a weight to existing interactions between agents, we use a community detection algorithm on this network of social interactions to identify clusters of *(i)* like-minded agents who *(ii)* share a context. We sought *balanced* clusters so that a comparable level of simplification is operated throughout the agent population, instead of merging only some massive groups into few agents (i.e., few very large communities) or operating a minimal reduction by merging pairs of agents (i.e., many small communities). In addition, we use non-overlapping clusters to ensure that each super-agent represents a distinct group of the initial agents. We considered four algorithms. The `chinesewhispers` is a randomized algorithm in which nodes are initially assigned different classes, then take the class that dominates in their local neighborhood (Biemann 2006). `DER` (Diffusion Entropy Reducer) applies random walks and a variant of the *k*-means algorithm until the clusters stabilize (Kozdoba and Mannor 2015). `Paris` initially assigns each node to its own class and merges the closest classes recursively as long as the modularity metric increases (i.e., agglomerative hierarchical clustering) (Bonald, Charpentier, Galland, and Hollocou 2018). Finally, `combo` combines the optimization strategies of other algorithms as it involves merging clusters, splitting them, or re-assigning nodes to a different cluster as long as an objective function score improves (Sobolevsky et al. 2014).

3.3 Third Step: Aggregating Each Cluster into a Representative Super-Agent

Each cluster represents a group of agent with similar behavioral rules and local context. We now simplify each group into a super-agent. This could be achieved by morphing the agents into a new composite agent or by identifying a representative agent. We take the latter approach by *(i)* calculating the sum of the concept values for each agent's FCM and *(ii)* selecting an agent in the median of these values. Once these representative agents are identified, we rebuild the simulation model by keeping only these agents. We ensure that two representative agents are connected if original agents in their communities used to interact.

4 RESULTS

4.1 Overview

Our objective is to assess the effect of simplification onto fidelity with respect to the original simulation results. We assume the general case of stochastic models where results consist of a *distribution* of outcomes across repeated simulation runs. The impact of simplification is measured by *(i)* KL Divergence between the original and new distribution, *(ii)* the number of reduced nodes, *(ii)* statistics of distributions (mean, quartiles, std). In addition, we plot the distribution of outcomes as a visual aid. Since modelers control two parameters of our simplification process, results are analyzed with respect to the choice of similarity measure (among 11 possibilities) and the community detection algorithm (out of four choices). Although modelers do not control the case study on which the process would be applied, *characteristics* of a case study can mediate the results. In particular, our process relies on weighing existing ties between agents, hence the structure of social ties can have an effect and we investigate it through different social network structures. Algorithm 1 summarizes the generation of results and their ensuing analysis. To support replicability, our implementation of the Algorithm and its use on each case study can be found at https://osf.io/hpz7c/.

4.2 Case Studies

We considered two case studies that openly provide FCMs on nutrition (Giabbanelli, Bernard, and Cussat-Blanc 2022) and obesity (Giabbanelli, Jackson, and Finegood 2014). That is, these studies provide a collection of rulesets that can make agents behave in different ways. The nutrition case study consisted of 722 unique FCMs, built from observed longitudinal data about real-world individuals using an algorithm called CMA-ES. Each FCM contains the same 15 concepts that are fully connected (Table 2), but weighted differently based on each individual. In the obesity case study, an FCM was created by aggregating the

perspectives of experts on physical exercise and eating behaviors (Table 3). To keep the case studies comparable, we created 722 unique versions of the FCM by varying the edge weights.

Table 2: The CMA-ES case study has 722 FCMs with the same fully connected concepts, but different individual weights (Giabbanelli, Bernard, and Cussat-Blanc 2022).

Concept #	Construct	Operationalization
1.	Awareness	Self-awareness of number of fruits eaten
2.	Attitude	Belief that eating 2 servings of fruits daily is healthy
3.	Attitude Price	Belief that eating 2 servings of fruits daily is expensive
4.	Self-efficacy (belief	...they can eat more fruit daily if they really want to
5.	that in the next 6 months...)	...it is difficult to eat more fruit
6.	Social-influence (belief	...think they should eat 2 pieces of fruit daily.
7.	that most important people...)	...consume two pieces of fruit per day.
8.	Intention	Intention to eat two pieces of fruit per day?
9.		Clear plan for when to eat more fruit.
10.	Action-planning	Clear plan for which fruit to eat more/less.
11.		Clear plan for how many fruits to eat more/less.
12.	Coping planning (plan	...something interferes with plans to eat more fruit.
13.	what to do when...)	...it is difficult to eat more fruit.
14.	Perceived availability	How often are fruit products available at home?
15.	Visibility at home	Visibility of fruits at home

Table 3: Edge values for the FCM in the obesity case study (Giabbanelli, Jackson, and Finegood 2014).

Source node	**List of target nodes (causal weight from -1 to 1)**
Age	Exercise (-0.44)
Income	Exercise (0.548), Fatness perceived as negative (0.478)
Fatness Perceived as Negative	Weight discrimination (0.739)
Belief in Personal Responsibility	Weight discrimination (0.578)
Obesity	Weight discrimination (0.84), Physical health (-0.795)
Weight discrimination	Depression (0.732)
Exercise	Depression (-0.649), Obesity (-0.638), Physical health (0.860)
Depression	Anti-depressants (0.592)
Anti-depressants	Obesity (0.528), Food intake (0.526)
Food intake	Obesity (0.637)
Knowledge	Food intake (-0.5), Exercise (0.5)
Stress	Depression (0.54), Food intake (0.607), Physical health (-0.694)

We created the structure of social ties by considering three network topologies: a random Erdos-Renyi graph (low clustering and normal degree distribution), a small-world Watts-Strogatz graph (high clustering with individuals forming groups), and the scale-free Barabasi-Albert (heavily skewed degree distribution with some individuals serving as hubs). In each case, individual agents were randomly assigned an FCM. In a hybrid ABM/FCM study, social ties only express *who* interacts. To define *how* they interact, we need to state which concepts of an individual can be observed by a peer (i.e., which parts of an FCM influence peers) and which concepts register these observations (i.e., which parts of an FCM are influenced by peers). We created a random connection, so that each agent had an equal chance to influence a particular concept of the other agent. FCM concepts were initialized between 0 and 1, and the agent with the higher value

would influence its peer with a lower value. Intuitively, agents are aligning with their peers to the upside. For example, this represents how individuals acquire *more* knowledge as a result of interacting with others.

Table 4: Properties of communities generated in the CMA-ES case study as a function of network structure, community detection algorithm (`chinese-whispers`, `der`, `paris`, `combo`), and similarity metric.

Measure	Similarity Metric	Scale-free network topology				Small-world network topology			
		chinese-whispers	der	paris	combo	chinese-whispers	der	paris	combo
Average # of Agents	Clustering Coefficient	15.1	361.0	128.7	50.7	5.8	361.0	193.1	38.5
	# of Concepts	102.2	361.0	135.6	48.9	12.1	361.0	240.7	45.1
	Graph Density	16.4	361.0	83.5	48.2	5.9	361.0	185.3	36.3
	R/T Ratio	104.7	361.0	135.6	48.9	12.1	361.0	240.7	45.1
	TSP	15.7	361.0	123.2	48.1	5.9	361.0	187.7	32.0
Max # of Agents	Clustering Coefficient	306.0	414.3	268.6	123.5	13.0	375.4	235.4	44.0
	# of Concepts	532.7	448.9	276.1	123.3	27.7	374.1	256.0	46.0
	Graph Density	331.6	411.9	194.1	125.8	13.7	371.0	223.8	43.3
	R/T Ratio	530.0	422.8	276.1	123.3	27.3	373.0	256.0	46.0
	TSP	305.8	403.3	249.4	126.3	13.1	370.0	223.6	41.6
Min # of Agents	Clustering Coefficient	1.9	307.7	22.4	11.2	3.0	346.6	150.6	33.3
	# of Concepts	2.9	273.1	30.9	6.7	2.6	347.9	210.0	45.0
	Graph Density	1.9	310.1	11.6	11.9	3.0	351.0	153.8	29.6
	R/T Ratio	2.8	299.2	30.9	6.7	2.5	349.0	210.0	45.0
	TSP	1.9	318.7	17.9	7.9	3.0	352.0	147.5	21.6
# of Communities	Clustering Coefficient	55.2	2.0	15.7	14.6	123.8	2.0	4.3	18.8
	# of Concepts	10.6	2.0	13.1	15.1	59.8	2.0	3.0	16.0
	Graph Density	51.4	2.0	19.9	15.3	122.9	2.0	4.2	19.9
	R/T Ratio	10.8	2.0	13.1	15.1	59.9	2.0	3.0	16.0
	TSP	54.3	2.0	10.0	15.3	123.1	2.0	4.0	22.6

4.3 Joint Effect of Community Detection Algorithm, Similarity Metric, and Network Structure

The communities depend on the structure of social ties, the choice of a measure that assigns similarity weights to these ties, and the choice of algorithm to detect communities from weighted social ties. We measure the interplay of these effects on the CMA-ES case study, as shown in Table 4 for two network structures. Communities are characterized by the average/min/max number of agents in each community and the number of communities. The results show that `DER` is not usable as it always yields only two communities. We also note that `combo` does not respond to the choice of the similarity metric, which is undesirable since the notion of a like-minded group should be largely driven by the definition of similarity. For this reason, we identify `chinesewhispers` as the most promising option and use it in our experiments.

4.4 Fidelity of Simulation Results Between the Simplified and the Original Model

We computed the distribution of outputs from the original and simplified models based on 100 runs. In both case studies, we observed that means are similar regardless of the case study. Based on the KL divergence, the distributions are similar (Table 5). We further analyzed these results by plotting the most similar distributions. We found that the main effect of the simplification is a *greater uncertainty around the mean* compared to the original model, as illustrated in Figure 2 for the obesity case study. Complete results on the distributions are provided in Table 6. The extent of the uncertainty depends on the choice of similarity measure, as some yield excessive levels of uncertainty (e.g., centrality and cosine similarity, density) while others have more contained increases. Interestingly, we note that the KL divergence between the distribution of *node weights* had about the same uncertainty as the Jaccard for the distribution of *edge*

weights. This means that the causal mechanisms used by an agent (edge weights) were about as useful as the initial traits of an agent (node weights) to determine the similarity of their simulated behaviors.

Table 5: Entropy of the KL divergence of the distribution of outputs in the simplified vs. original model. Note that the entropy is expressed on a scale of 10^{-4}, hence a value of '9' should be interpreted as 0.0009.

		KS Edges	Jaccard	Centrality	Density	Edge KL	TSP	Node KL	Graphs	RT	Clustering	#nodes
SF	CMAES	9	10	8	9	11	7	10	13	158	10	90
	Nutrition	82	8	151	107	88	85	9	10	171	129	125
SM	CMAES	9	11	9	11	9	10	10	7	23	12	24
	Nutrition	29	9	25	28	18	24	10	10	24	21	25
Rand	CMAES	9	12	6	6	8	6	6	9	47	5	36
	Nutrition	48	7	41	50	44	36	9	9	42	46	32

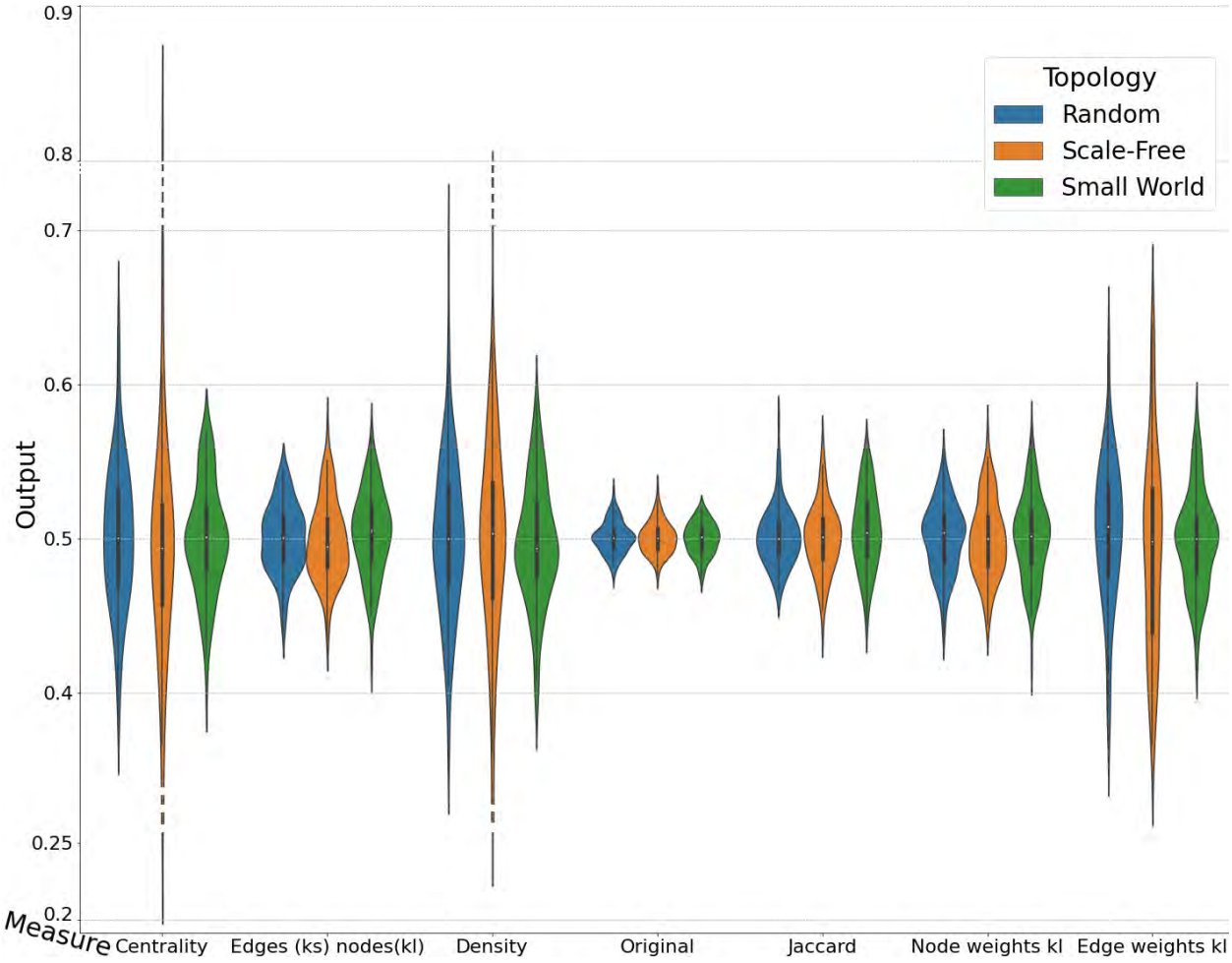

Figure 2: In the obesity case study, mean simulation outcomes in the original model are comparable with the simplified model. However, simplified models have more uncertainty, as shown by their wider distributions.

Table 6: Characteristics of the distributions of simulation outputs in the simplified model.

		Output of the nutrition case study								Output of the CMAES case study						
	Measure	**mean**	**std**	**min**	**25%**	**50%**	**75%**	**max**		**mean**	**std**	**min**	**25%**	**50%**	**75%**	**max**
Random	**Original**	.501	.010	.476	.494	.501	.506	.531		.502	.010	.479	.495	.502	.508	.529
	centrality	.502	.047	.385	.470	.501	.531	.643		.504	.020	.444	.494	.504	.516	.581
	clustering	.503	.048	.382	.474	.505	.529	.623		.502	.019	.450	.490	.498	.514	.556
	compare graphs	.500	.022	.440	.485	.500	.515	.544		.502	.024	.442	.488	.501	.519	.558
	concepts	.502	.041	.420	.473	.507	.530	.588		.499	.044	.389	.465	.496	.530	.604
	density	.501	.052	.363	.471	.500	.534	.688		.500	.018	.462	.487	.500	.512	.549
	edge weight kl	.504	.047	.371	.476	.508	.534	.626		.500	.023	.439	.485	.500	.516	.547
	Jaccard	.501	.019	.464	.489	.500	.511	.577		.500	.027	.439	.481	.500	.515	.581
	KS edges	.504	.051	.373	.472	.500	.537	.624		.504	.023	.445	.488	.502	.519	.578
	node weights kl	.501	.022	.439	.484	.504	.515	.553		.501	.021	.439	.487	.502	.517	.559
	rt ratio	.504	.046	.387	.476	.505	.529	.598		.505	.051	.379	.473	.506	.537	.643
	tsp	.501	.044	.408	.466	.497	.527	.634		.499	.019	.456	.486	.496	.513	.551
Scale-Free	**Original**	.500	.010	.476	.493	.500	.507	.533		.500	.010	.475	.493	.499	.507	.520
	centrality	.493	.085	.265	.457	.494	.522	.807		.499	.022	.448	.483	.500	.514	.546
	clustering	.494	.076	.200	.461	.498	.528	.736		.499	.024	.443	.484	.499	.518	.546
	compare graphs	.499	.026	.435	.481	.495	.513	.571		.500	.028	.437	.484	.500	.516	.568
	concepts	.504	.079	.266	.475	.505	.541	.712		.504	.070	.267	.482	.500	.522	.832
	density	.500	.073	.280	.461	.503	.537	.748		.498	.024	.425	.483	.497	.510	.559
	edge weight kl	.492	.065	.365	.440	.498	.532	.639		.499	.025	.416	.484	.499	.513	.561
	Jaccard	.500	.022	.441	.487	.501	.513	.562		.498	.025	.441	.479	.500	.514	.564
	KS edges	.500	.064	.350	.454	.493	.548	.652		.497	.022	.448	.485	.496	.509	.592
	node weights kl	.500	.025	.444	.483	.500	.514	.567		.499	.025	.431	.482	.499	.518	.555
	rt ratio	.496	.098	.251	.442	.493	.519	.895		.504	.089	.201	.461	.494	.535	.903
	tsp	.504	.068	.316	.472	.502	.532	.768		.499	.022	.438	.486	.501	.514	.550
Small-world	**Original**	.500	.010	.473	.492	.501	.508	.520		.502	.011	.478	.496	.502	.507	.535
	centrality	.500	.035	.403	.480	.501	.521	.569		.498	.023	.447	.482	.498	.514	.554
	clustering	.499	.035	.426	.472	.496	.522	.597		.501	.025	.445	.483	.503	.520	.555
	compare graphs	.504	.027	.422	.487	.505	.522	.566		.504	.022	.433	.488	.506	.518	.560
	concepts	.502	.039	.416	.479	.498	.530	.592		.497	.036	.430	.471	.497	.516	.635
	density	.497	.039	.394	.476	.494	.523	.588		.501	.024	.444	.484	.501	.517	.573
	edge weight kl	.499	.031	.421	.478	.500	.514	.577		.500	.023	.447	.483	.502	.515	.564
	Jaccard	.505	.025	.446	.488	.504	.523	.557		.504	.028	.452	.485	.498	.526	.586
	KS edges	.501	.039	.421	.473	.502	.525	.612		.501	.023	.444	.486	.502	.514	.554
	node weights kl	.501	.027	.420	.484	.502	.518	.568		.502	.025	.455	.483	.501	.518	.572
	rt ratio	.501	.037	.389	.477	.500	.528	.593		.503	.037	.383	.479	.504	.532	.575
	tsp	.499	.036	.418	.476	.495	.527	.583		.502	.023	.450	.484	.503	.517	.561

5 DISCUSSION AND CONCLUSION

Scaling a simulation by creating a small number of representative super-agents that represent a collective "has implications for model forecasts [but] is an underdeveloped field of study" (Wise, Milusheva, Ayling, and Smith 2023). As noted by Wise *et al.*, population sizes can vary hence the extent of scaling is variable. Furthermore, the dynamics of the simplified model can also differ from the original one. In our work, we considered the structure of social ties, the definition of a group (i.e., the clustering algorithm) and the definition of similarity. We showed that `chinese-whispers` can create groups of varying sizes depending on the similarity and social ties. Our work complement Wise *et al.* who focused on the geographical dynamics of a model, while we examined the impact of social ties in a hybrid model. We also contribute to the work of Toupance, Chopard, and Lefèvre (2023), since their reduced model was obtained by a fixed grouping of nearby cells whereas we adapt groups based on social ties and similarity. In our work, the average simulation output was preserved and distributions of simulated outputs were similar between the simplified and original model, although the simplification increases the uncertainty. Future work should comprehensively examine the costs associated with mitigation scenarios for the uncertainty,

as these costs may partly offset the benefits of our proposed simplification. For example, if a simplified simulation requires *more runs* than the original model to reach the same desired confidence interval, then the performance improvements from the simplification are reduced.

We analyzed the impact of reducing the number of nodes in a hybrid ABM/FCM model based on the two parameters of our method and one mediating characteristic of case studies. We incorporated three new metrics along with existing metrics to measure the similarities of FCMs of various agents and studied the impact of clustering agents into representative super agents. Both aspects can be elaborated upon in future works. By applying our methods to more cases, modelers may start to discern additional mediating characteristics, particularly with respect to the rulesets of the agents. While the Jaccard and KL metrics were strong performers in our study (by having less uncertainty than alternatives), metrics to compare agents can continue to be developed. For instance, instead of measuring the total number of concepts, we can use a distribution approach by measuring the Hamming distance between the presence or absence of specific concepts. This could be further weighted by concept categories: an agent motivated by price and perceived availability differs from an agent acting based on social-influence, intention, and visibility.

The hybrid ABM/FCM framework is an asset for our study as the agents' behaviors are defined through simulation models that allow to leverage the rich literature on network comparison. Dozens of studies have employed the ABM/FCM paradigm (Davis, Giabbanelli, and Jetter 2019), with new case studies appearing regularly (Ciftci and Durmusoglu 2023). However, there are many ways to operationalize the internal cognitive processes of an agent. ABMs can integrate causal relationships, but the representation of these relationships is not necessarily performed through a transparent network within each agent (Antosz et al. 2022). Our findings may thus only apply to frameworks in which cognitive processes of the agents are *explicitly* represented as a *network*. For example, hybrid ABM / System Dynamics (SD) models provide a closely related approach by using a network-based simulation model to guide the decision-making activities within each agent (Schieritz and Grobler 2003; Swinerd and McNaught 2014). Our metrics can thus apply to the ABM/SD approach, which could broaden the benefits of our approach for the simulation community.

REFERENCES

Aljundi, A. A., T. A. Akyildiz, and K. Kaya. 2022. "Degree-Aware Kernels for Computing Jaccard Weights on GPUs". In *2022 IEEE International Parallel and Distributed Processing Symposium (IPDPS)*, 897–907.

Antosz, P., T. Szczepanska, L. Bouman, J. G. Polhill and W. Jager. 2022. "Sensemaking of causality in agent-based models". *International Journal of Social Research Methodology* 25(4):557–567.

Băbeanu, A.-I., T. Filatova, J. H. Kwakkel, and N. Yorke-Smith. 2023. "Adaptive parallelization of multi-agent simulations with localized dynamics". *arXiv preprint arXiv:2304.01724*.

Biemann, C. 2006. "Chinese whispers-an efficient graph clustering algorithm and its application to natural language processing problems". In *Proc. TextGraphs: 1st workshop on graph based methods for natural language processing*, 73–80.

Bonald, T., B. Charpentier, A. Galland, and A. Hollocou. 2018. "Hierarchical graph clustering using node pair sampling". *arXiv preprint arXiv:1806.01664*.

Ciftci, P. K. and Z. D. U. Durmusoglu. 2023. "A hybrid agent-based model integrated with a multi-stage learning-based fuzzy cognitive map". *Kybernetes*.

Cilfone, N. A., D. E. Kirschner, and J. J. Linderman. 2015. "Strategies for efficient numerical implementation of hybrid multi-scale agent-based models to describe biological systems". *Cellular and molecular bioengineering* 8:119–136.

Davis, C. W., P. J. Giabbanelli, and A. J. Jetter. 2019. "The intersection of agent based models and fuzzy cognitive maps: a review of an emerging hybrid modeling practice". In *2019 Winter Simulation Conference (WSC)*, 1292–1303. Institute of Electrical and Electronics Engineers. National Harbor, MD, USA.

Davis, C. W., A. J. Jetter, and P. J. Giabbanelli. 2020. "Fuzzy cognitive maps in agent based models: a practicial implementation example". In *Spring Simulation Conference*, edited by F. Barros, X. Hu, H. Kavak, and A. D. Barrio, 1–11. Institute of Electrical and Electronics Engineers. Fairfax, VA, USA.

Ghumrawi, K. A., K. Ha, J. T. Beerman, J.-D. Rudie and P. J. Giabbanelli. 2022. "Software Technology to Develop Large-Scale Self-Adaptive Systems:Accelerating Agent-Based Models and Fuzzy Cognitive Maps via CUDA". In *Hawaii International Conference on System Sciences*. Maui, Hawaii, USA.

Giabbanelli, P. J. 2024. *Hybrid Simulations*, 61–86. Cham: Springer Nature Switzerland.

Giabbanelli, P. J., D. Bernard, and S. Cussat-Blanc. 2022. "Fast Generation of Heterogeneous Mental Models from Longitudinal Data by Combining Genetic Algorithms and Fuzzy Cognitive Maps". In *Hawaii International Conference on System Sciences*, 1–10. Maui, Hawaii, USA.

Giabbanelli, P. J., P. J. Jackson, and D. T. Finegood. 2014. "Modelling the joint effect of social determinants and peers on obesity among Canadian adults". In *Theories and simulations of complex social systems*, 145–160. Springer.

Giabbanelli, P. J., C. B. Knox, K. Furman, A. Jetter and S. Gray. 2024. *Defining and Using Fuzzy Cognitive Mapping*, 1–18. Cham: Springer Nature Switzerland.

Gray, S., E. J. Sterling, P. Aminpour, *et al*. 2019. "Assessing (social-ecological) systems thinking by evaluating cognitive maps". *Sustainability* 11(20):5753.

Juszczyszyn, K. 2018. "Motif Analysis". In *Encyclopedia of Social Network Analysis and Mining*, edited by R. Alhajj and J. Rokne, 1392–1399. New York, NY: Springer New York.

Kozdoba, M. and S. Mannor. 2015. "Community detection via measure space embedding". *Advances in neural information processing systems* 28.

Lahitani, A. R., A. E. Permanasari, and N. A. Setiawan. 2016. "Cosine similarity to determine similarity measure: Study case in online essay assessment". In *2016 4th International Conference on Cyber and IT Service Management*, edited by H. Sukmana, 1–6. Bandung, Indonesia.

Lippe, M., M. Bithell, N. Gotts, *et al*. 2019. "Using agent-based modelling to simulate social-ecological systems across scales". *GeoInformatica* 23:269–298.

Lysenko, M. and R. M. D'Souza. 2008. "A framework for megascale agent based model simulations on graphics processing units". *Journal of Artificial Societies and Social Simulation* 11(4):10.

Miasnikof, P., A. Y. Shestopaloff, C. Bravo, and Y. Lawryshyn. 2023. "Statistical Network Similarity". In *Proc. 11th Int. Conf. Complex Networks and their Applications*, Volume 2, 325–336. Springer.

Milo, R., S. Itzkovitz, N. Kashtan, *et al*. 2004. "Superfamilies of evolved and designed networks". *Science* 303(5663):1538–1542.

Minucci, S. B., R. L. Heise, and A. M. Reynolds. 2024. "Agent-based vs. equation-based multi-scale modeling for macrophage polarization". *Plos one* 19(1):e0270779.

Parry, H. R. and M. Bithell. 2012. "Large scale agent-based modelling: A review and guidelines for model scaling". *Agent-based models of geographical systems*:271–308.

Parry, H. R. and A. J. Evans. 2008. "A comparative analysis of parallel processing and super-individual methods for improving the computational performance of a large individual-based model". *Ecological Modelling* 214(2-4):141–152.

Schieritz, N. and A. Grobler. 2003. "Emergent structures in supply chains-a study integrating agent-based and system dynamics modeling". In *Proc. 36th Annual Hawaii International Conference on System Sciences (HICSS)*, 9. Institute of Electrical and Electronics Engineers. Waikoloa Village, HI, USA.

Sobolevsky, S., R. Campari, A. Belyi, and C. Ratti. 2014. "General optimization technique for high-quality community detection in complex networks". *Physical Review E* 90(1):012811.

Swinerd, C. and K. R. McNaught. 2014. "Simulating the diffusion of technological innovation with an integrated hybrid agent-based system dynamics model". *Journal of simulation* 8(3):231–240.

Tchupo, D. E. and G. A. Macht. 2022. "Comparing fuzzy cognitive maps: Methods and their applications in team communication". *Int. J. Industrial Ergonomics* 92:103344.

Toupance, P.-A., B. Chopard, and L. Lefèvre. 2023. "System reduction: an approach based on probabilistic cellular automata". *Natural Computing*:1–13.

Wills, P. and F. G. Meyer. 2020. "Metrics for graph comparison: a practitioner's guide". *Plos one* 15(2):e0228728.

Wise, S., S. Milusheva, S. Ayling, and R. M. Smith. 2023. "Scale matters: Variations in spatial and temporal patterns of epidemic outbreaks in agent-based models". *Journal of Computational Science* 69:101999.

Xiao, J., P. Andelfinger, D. Eckhoff, W. Cai and A. Knoll. 2019. "A survey on agent-based simulation using hardware accelerators". *ACM Computing Surveys (CSUR)* 51(6):1–35.

AUTHOR BIOGRAPHIES

PHILIPPE J. GIABBANELLI is a Research Professor at the Virginia Modeling, Analysis and Simulation Center (VMASC) at Old Dominion University. He has authored over 130 articles and several books in Modeling & Simulation, with an emphasis on combining M&S with Machine Learning. His email address is pgiabban@odu.edu.

JACK T. BEERMAN is a doctoral student in data science at the University of Virginia and an Operations Research Analyst in the US Air Force. His work with Dr. Giabbanelli has appeared in several venues including the Journal of Computational Science, HICSS'23, and ANNSIM'23. His research interests include network science and AI. His email address is jtb3sud@virginia.edu.