Taylor & Francis
Taylor & Francis Group

Journal of Simulation

ISSN: (Print) (Online) Journal homepage: www.tandfonline.com/journals/tjsm20

An uncertainty quantification framework for agent-based modeling and simulation in networked anagram games

Zhihao Hu, Xueying Liu, Xinwei Deng & Chris J. Kuhlman

To cite this article: Zhihao Hu, Xueying Liu, Xinwei Deng & Chris J. Kuhlman (2024) An uncertainty quantification framework for agent-based modeling and simulation in networked anagram games, Journal of Simulation, 18:4, 505-523, DOI: 10.1080/17477778.2024.2313134

To link to this article: https://doi.org/10.1080/17477778.2024.2313134

Published online: 29 Feb 2024.

Submit your article to this journal

Article views: 67

View related articles

View Crossmark data

Full Terms & Conditions of access and use can be found at
https://www.tandfonline.com/action/journalInformation?journalCode=tjsm20

JOURNAL OF SIMULATION
2024, VOL. 18, NO. 4, 505–523
https://doi.org/10.1080/17477778.2024.2313134

THE OPERATIONAL RESEARCH SOCIETY
Taylor & Francis
Taylor & Francis Group

RESEARCH ARTICLE

Check for updates

An uncertainty quantification framework for agent-based modeling and simulation in networked anagram games

Zhihao Hu[a], Xueying Liu[a], Xinwei Deng[a] and Chris J. Kuhlman[b]

[a]Department of Statistics, Virginia Tech, Blacksburg, VA, USA; [b]Advanced Research Computing, Virginia Tech, Blacksburg, VA, USA

ABSTRACT

In a networked anagram game, players are provided letters with possible actions of requesting letters from their neighbours, replying to letter requests, or forming words. The objective is to form as many words as possible as a team. The experimental data show that behaviours among players can vary significantly. However, simulations using agent-based models (ABM) in the literature often have not incorporated proper uncertainty quantification methods to characterise diverse behaviours of players. In this work, we propose an uncertainty quantification framework to build, exercise, and evaluate agent behaviour models and simulations for networked group anagram games. Specifically, using the data of game experiments, the proposed framework considers the clustering of game players based on their performance to reflect players' heterogeneity. Moreover, we also quantify uncertainty within each cluster through statistical modelling and inference. Numerical studies of networked game configurations are conducted to demonstrate the merits of the proposed framework.

ARTICLE HISTORY
Received 25 February 2022
Accepted 23 January 2024

KEYWORDS
Uncertainty quantification; agent-based models; model construction; simulation; networked group anagram games

1. Introduction

1.1. Background and motivation

Anagram games are word formation games and have been employed in a wide range of research. For example, they are used to study individual mental capabilities, and coordination and cooperation within groups. This is because anagram games are considered to be non-trivial mental tasks (e.g., Cadsby et al. (2007)). The anagram games have a unique combination of features that make them attractive: simple and unambiguous directions, minimal space and equipment requirements to play the game, variable and controllable complexity of task (e.g., requiring greater rearrangement of letters, requiring words with greater numbers of letters, giving lesser time to form words), and straight-forward ways to quantify performance so that success is clearly defined. See Appendix A for descriptions of several works.

Anagram games can be divided into two classes, based on their goal: (*i*) rearranging scrambled letters to form a unique word, or (*ii*) identifying as many words as possible from a collection of letters. In a previous work (Cedeno-Mieles et al., 2020), online networked group anagram games (GrAGs) or experiments were conducted, where players share alphabetic letters to form words. Using experimental data from human subjects, an agent-based model (ABM) was developed to enable simulation of games for conditions beyond those tested (Ren et al., 2018). The experiments and model are described below as background. Here, we use the following terms to denote GrAG game players because they are also agents in ABMs and nodes or vertices in the game network: *node*, *vertex*, *agent*, and *player*.

Figure 1 depicts four consecutive time steps in a hypothetical GrAG with a simpler network configuration to aid the description. Communication channels are in purple, and on these channels a player may request letters and reply to letter requests. Overall, a player may take any of four actions, any number of times, and in any order during a 5 minute game: (*i*) request a letter from a neighbour (request sent), (*ii*) reply to a request with the letter (reply sent), (*iii*) form a word (form word), and (*iv*) think or idle (i.e., a no-op condition).

In our online experiments, human players are recruited using Amazon Mechanical Turk (AMT) and these people play the GrAG remotely using a customised software game platform (Cedeno-Mieles et al., 2020) that they access through their web browsers. Each player is initially given three letters (shown in brown, in the black boxes). During the game, players can request letters from their neighbours, and neighbours can choose to reply with the requested letters or not. For example, v_1 requests a *e* from v_2 at time t, and v_2 sends a reply with the *e* at time $(t+1)$ so that *e* gets added to v_1's letter set, with which it forms words. Received letters are shown in black. If a player shares a letter with the requestor,

CONTACT Chris J. Kuhlman ckuhlman@vt.edu Advanced Research Computing, Virginia Tech, 1311 Research Center Drive, Blacksburg, VA 24060, USA

© 2024 The Operational Research Society

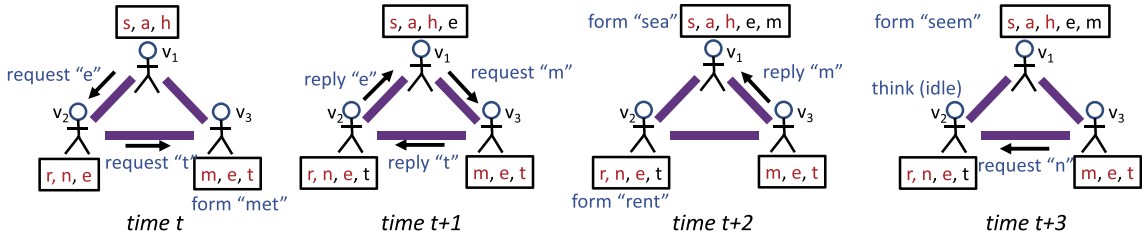


Figure 1. Illustrative play in a group anagram game (GrAG) among three players v_i, $1 \leq i \leq 3$, each with three initial letters in brown, within a black box. Received letters from neighbours are in black. In the model of these games, a player executes one action per time step where a time step is one second. The game is described in the text.

then both the requestor and the player replying have a copy of the letter. A person never loses a letter, even when they share it with others. This is to encourage sharing letters and forming more words. Also, a person may use a letter in any number of words, and any number of times within a single word. For example, v_1 uses *s* and *e* in forming *sea* at time $(t+2)$ and *seem* at $(t+3)$, thus using the single *e* once in the first word and twice in the latter word. Games are played for 300 seconds (5 minutes). Each player typically forms between 10 and 40 words.

In GrAGs, it is seen that behaviours can vary significantly among players. The mean model (Ren et al., 2018) does not capture this heterogeneity. A baseline ABM that only captures mean behaviour is problematic because it implies that all agent behaviours will, over time in one simulation, tend to the same mean behaviour, and all agent behaviours over many simulation instances will also tend to the same mean behaviour. Producing models that contain greater ranges of player performance more faithfully represents the ranges in behaviours observed in the games (Cedeno-Mieles et al., 2020). This underscores the need for methods to quantify the uncertainty in players' behaviours.

Moreover, there are limitations to the amount of experimental data that can be collected. It is well-known that AMT does not provide an unlimited pool of candidate players. Additionally, some candidates do not show up for experiments (Mason & Suri, 2018). We also constrained our experiments so that a person could only play the GrAG one time, to obviate learning from past experience. Consequently, we encountered limitations in the size of our candidate pool, resulting in fewer completed games than we desired. This produced two problems to overcome in building ABMs of game player behaviour: data sparsity and variability.

These challenges motivate the development of a general uncertainty quantification (UQ) framework for building ABMs of human behaviour in the networked anagram game. Our primary objective is not necessarily to generate human-like actions or behaviours, but rather to accurately quantify the uncertainty inherent in players' behaviour based on experimental data. This uncertainty is then integrated into agent-based simulations (ABS) to create more faithful representations of the diversity of players in the real-world game. The proposed UQ framework is designed to study human behaviour in various scenarios that might be expensive or practically impossible to conduct in real-world experiments. The original experiments, conducted with remote participants recruited via AMT, were limited in scope and settings. However, our proposed framework has the potential to simulate GrAGs with a large number of players and explore a variety of scenarios, including different network structures and varying numbers of players' neighbours. This not only allows for the quantification of heterogeneous behaviours among players but also provides a more comprehensive understanding of individual player behaviours. The insights gained from these simulations can then be used to guide further experiments and studies. (We use "simulation" for computations of a simulation, in computing player actions during a GrAG; we use "modeling" for the process of constructing models used in simulation.) We believe that our uncertainty quantification methodology for ABM is also applicable to other types of experiments that involve human behaviour, such as Mason and Watts (2012).

1.2. Novelty and contributions

The proposed framework is to design, implement, and execute a general UQ approach for building ABMs of human behaviour from networked GrAG game data, such that different agents can have heterogeneous behaviours. Based on our best knowledge, it is the first UQ framework for modelling and analysis of networked GrAGs. The key novelty is systematically modelling and simulating the networked anagram game with the considerations of data uncertainty and player's uncertainty. Rigorous hypotheses are conducted to investigate the homogeneity of players with different numbers of neighbours in the networked GrAGs. Furthermore, clustering analysis is performed to refine the quantification of heterogeneous behaviours among players. The cluster results provide a foundation to model player performance based on the experimental data for players within each cluster,

such that the variability of players' abilities to play GrAGs can be better quantified. By using the probabilistic uncertainty of estimated transition probabilities of actions via the asymptotic distribution of the estimated parameters, we can further accommodate data uncertainty into the proposed framework. Incorporating the UQ scheme described above, we thus build ABMs for simulating networked GrAG games where each agent can be endowed with different transition probabilities of actions to better reflect the UQ of human behaviours.

Figure 2 illustrates two new models, the CWM model and the CWUQ model, in comparison with the baseline model (Ren et al., 2018). For the baseline model, players with the same degree will exhibit the same behaviour. In contrast, after grouping players by degree ($d \leq 2$ and $d \geq 3$), the cluster-wise mean (CWM) model considers player behaviours to be different within each group. Then for each group, players are partitioned into four clusters, $c = 1$ through 4, based on their performance. Thus, the CWM model considers four different levels of player behaviours for each degree range. Hence, while the degree d is solely based on network structure, the cluster is specified as an external input that governs players' performance. The cluster-wise uncertainty quantification (CWUQ) model is an extension of the CWM model to account for game data variability within each cluster (indicated by the error bars around each blue data point). These models are formally described and evaluated in Sections 3 and 4. The models are then used to build ABMs and conduct simulations of the anagram game for different numbers of game players and connectivity among them, and different agent performances.

The contributions of the proposed UQ framework are as follows. First, a key contribution is to systematically quantify the uncertainty of the game data through the CWM and CWUQ models. Combining hypothesis testing, statistical analysis, and clustering, player behaviours from games are partitioned by skill level in terms of (a) numbers of interactions with neighbours and (b) numbers of words formed. This immediately provides a way to specify agent models in terms of player performance: through the computed clusters. It also provides meaningful distinctions between poor- and good-performing agents. For modelling the GrAG, a per-player logit-model is constructed to determine a player's next action at time $(t + 1)$ based on the player's most recent action at time t and on a vector of parameters that describe the history of the player's actions and interactions with other players. Moreover, the extension of the CWM model to the CWUQ model enables the quantification of uncertainty within a cluster by accounting for variability of behaviour within it. This is achieved by sampling from distributions of model parameter values in the logit-model at each time during a simulation. Thus, this model incorporates two levels of uncertainty: that from clustering, and uncertainty in model parameters.

The second contribution is incorporating the UQ into ABM simulation of the networked GrAGs. An agent-based modelling and simulation (ABMS) software platform was constructed that executes both of the agent CWM and CWUQ models for arbitrary game configurations and for user-specified assignment of players to clusters (which dictates their performance). An in-depth evaluation of the CWM and CWUQ models, as seen in Section 4, can greatly enhance the interpretation of ABM results. Thus, the proposed framework achieves a good balance among model explainability, model flexibility, and model complexity, e.g. (Baker, 2016; Pearl & Mackenzie, 2018). An ABM of human behaviour is expected to

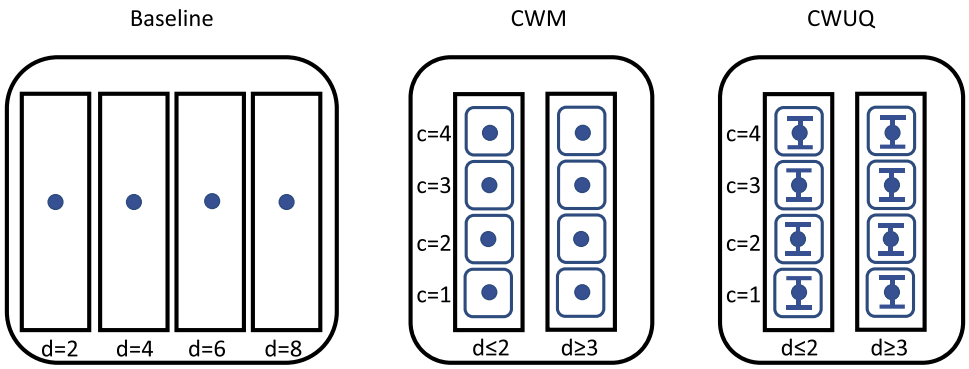


Figure 2. Illustrations in contrasting different ABMs constructed from GrAGs. (LEFT) The ABM of Ren et al. (2018) where a behaviour model is specified for an agent based on a player's (node's) degree in the networked anagram game, with values 2, 4, 6, and 8. (CENTER) The first of two new models is called the cluster-wise mean model, CWM. A behaviour model for an agent is specified by the pair $[g, c]$, where $g = 1$ if the agent has degree $d \leq 2$ in the network and $g = 2$ if $d \geq 3$, and where c is the cluster number 1, 2, 3, or 4. Clusters represent different levels of performance of agents in the game. (RIGHT) The second of two new models is called the cluster-wise uncertainty quantification model, CWUQ. This model is similar to CWM, but now within each cluster, the uncertainty of behaviour is quantified. It is through the last two models that heterogeneous agent behaviours are realized in simulations.

satisfy one or more criteria such as: (1) explainability of human behaviour, (2) sufficient model complexity of human behaviour, (3) a favourable framework for UQ with consideration of computational cost, and (4) other considerations based on specific problems. We demonstrate outcomes that illustrate some of these criteria with the next contribution below.

Our third contribution is the insightful evaluation of the models in the context of ABM for human group behaviour. There are the following insights: (*i*) Variability in model predictions of numbers of words formed are dependent on clusters. Representative simulation results indicate that the number of words formed by one player in a game can vary by factors of 5 to 10. These ranges are consistent with the variability in the game data. (*ii*) Variability in interactions between players is much less than that for numbers of words formed because bounds on the number of interactions of a player are dictated by a player's degree in the network and the number of letters that a player has. (*iii*) For each of the CWM and CWUQ models, variability across agents endowed with the same cluster behaviour generally is less than the variability within a single player and is less than the variability across clusters. The latter point is particularly true when comparing the behaviour of cluster 4, which has the greatest performance compared to the other three clusters. (*iv*) We find that the CWUQ model generates at least as much variability in results as does the CWM model. In many cases, the variability for the two models is comparable. These differences are smaller than the differences produced by changes in behavioural clusters.

1.3. Paper organization

This paper is an extension of preliminary work in Hu et al. (2021). In that work, simulations were conducted using the CWUQ model and a 5-node star-4 network. In this work, we run simulations using both models on the star-4 network and an 18-node graph of four connected cliques. The 18-node graph is designed to address variability of behaviours across agents and across subgraphs of different sizes, significantly extending the conditions evaluated with the star-4 graph. We compare the models and simulation predictions from them on the two networks.

The remainder of the paper is organised as follows, using Figure 3. We first present the formalism for the baseline model in Section 2, which provides a point of departure for our new work, and makes the document self-contained. Then we present the two new models in Section 3, which includes the clustering method for players and quantifying uncertainty for model parameters. The simulation system is also defined. Model evaluation is in Section 4. In Section 5, we provide simulation studies using both models on two networks. A discussion concludes the work in Section 6. Related work comprises Appendix A. Appendix B contains results for a 5-player star game configuration; these results supplement those in Section 5. Appendix C shows the hypothesis testing results to demonstrate the variability of CWM and CWUQ models.

2. Experimentation and baseline model

The anagram game played by a group is described in Section 1.1 and shown in Figure 1. At any time during a game, a player executes one of the actions from the action set A provided in Table 1. In the online experiments, most players spend the majority of their time taking no action (i.e., not requesting a letter, not replying to a letter request, and not forming a word). Hence, in models, we refer to this time as occupied in thinking, or otherwise idle. Over all 243 experiments, it is exceedingly rare for a player to take two or more actions within one second of time; therefore, in our

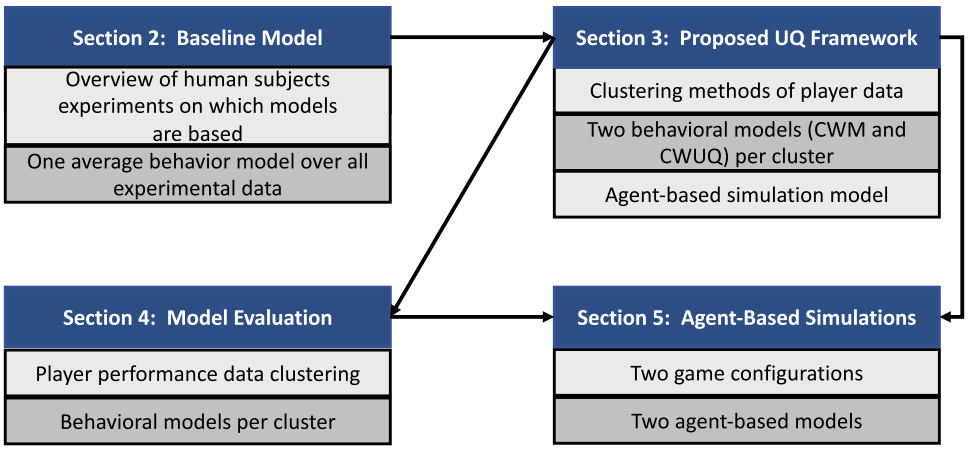


Figure 3. Technical sections of this manuscript, with arrows showing dependencies between them. Section numbers are given for topics. Section 2 is previous work and is provided as a point of departure for the current work, and to make this manuscript self-contained. Sections 3 through 5 contain our new work and contributions.

Table 1. The four actions of players in the GrAG and model. The set a of actions is $A = \{a_1, a_2, a_3, a_4\}$.

Item	Action	Name	Description	Item	Action	Name	Description
1	a_1	idling	Thinking (a no-op).	3	a_3	request	Requesting a letter from a neighbour.
2	a_2	reply	Replying to a neighbour with a requested letter.	4	a_4	word	Forming and submitting a word.

models and simulations, we advance time in one-second intervals over the 300-second game, with each player selecting one action at each second of a simulation.

The network configuration for each experiment was a random regular graph, meaning that each player in one game had the same number d of neighbours. Edges between pairs of players were placed randomly to meet this goal. A network was fixed during an experiment. Experiments were run with $d = 2$, 4, 6, and 8. Each player in each game was assigned three initial letters. Additional game details are in Ren et al. (2018). Note that players evenly split the total earnings from a game, where the earnings are proportional to the total number of words formed by the *team*. Thus, our networked anagram game is a cooperative game (Deutsch, 1949).

Using game data from experiments, Ren et al. (2018) constructed a multinomial logistic regression model to predict a player's action at time $(t+1)$ based on the player's action at time t and the values of four temporal variables provided in Table 2. Let $\mathbf{z} = (1, Z_B(t), Z_L(t), Z_W(t), Z_C(t))^T_{5\times 1}$. Note that $Z_C(t)$ is used to ensure agents do not stagnate in thinking; this parameter forces agents to have a finite deliberation period before acting. Then the multinomial logistic regression to model π_{ij}—the probability of a player taking action a_j at time $(t+1)$, given that the player took action a_i at time t—can be expressed as

$$\pi_{ij} = \frac{\exp(\mathbf{z}^T \beta_j^{(i)})}{\sum_{l=1}^{4} \exp(\mathbf{z}^T \beta_l^{(i)})}, \quad i, j = 1, 2, 3, 4, \qquad (1)$$

where $\beta_j^{(i)} = (\beta_{j,1}^{(i)}, \ldots, \beta_{j,5}^{(i)})^T_{5\times 1}$. For a given i (the index i on action a_i), the parameter set is

$$B^{(i)} = \begin{pmatrix} \beta_{\mathbf{1}}^{(i)T} \\ \beta_{\mathbf{2}}^{(i)T} \\ \vdots \\ \beta_{\mathbf{4}}^{(i)T} \end{pmatrix} = \begin{pmatrix} \beta_{1,1}^{(i)} & \beta_{1,2}^{(i)} & \cdots & \beta_{1,5}^{(i)} \\ \beta_{2,1}^{(i)} & \beta_{2,2}^{(i)} & \cdots & \beta_{2,5}^{(i)} \\ \vdots & \vdots & \ddots & \vdots \\ \beta_{4,1}^{(i)} & \beta_{4,2}^{(i)} & \cdots & \beta_{4,5}^{(i)} \end{pmatrix}. \qquad (2)$$

Game players in the GrAG data are grouped by their degrees d in the network G. To estimate the parameter sets $B^{(i)}$, Ren et al. (2018) used maximum likelihood estimation across the experimental observations for each $d = 2, 4, 6$, and 8. Suppose that there are n observational data having the same "most recent" action a_i and number of neighbours d, denote as $\mathcal{D}_d^{(i)}$, then the next action for observation l, namely y_l, has a multinomial distribution with corresponding probability $\pi_{li1}, \pi_{li2}, \pi_{li3}$, and π_{li4} (π_{lij} is the π_{ij} in Equation (1) for observation l). The probability of observing outcome y_l is

$$f(y_l | B^{(i)}, \mathbf{z}) = \pi_{li1}^{y_{l1}} \times \pi_{li2}^{y_{l2}} \times \pi_{li3}^{y_{l3}} \times \pi_{li4}^{y_{l4}},$$

where y_{lj} indicates whether the next action of observation l is a_j or not. $y_{lj} = 1$ if $y_l = j$, otherwise, it equals 0. Then, they conduct parameter estimation by finding the $B^{(i)}$ that maximises the log-likelihood function

$$\hat{B}^{(i)} = \arg \max_{B^{(i)}} \log L(B^{(i)} | \mathcal{D}_d^{(i)}) = \arg \max_{B^{(i)}} \log \prod_{l}^{n} f(y_l | B(i)) \qquad (3)$$

using the Broyden – Fletcher – Goldfarb – Shanno (BFGS) algorithm of the quasi-Newton optimisation method (Broyden, 1967).

Based on the estimation from the multinominal logistic model, Cedeno-Mieles et al. (2020); Ren et al. (2018) presented an ABM of the GrAG, where the game is modelled as a discrete-time process. At each time step, a player executes one of the actions from the action set. They considered the set V of players and the set E of their communication channels (edges) as an undirected graph $G(V, E)$. Here all agents with the same number of neighbours d in the network G are assigned the same coefficient matrix. Thus, these agents will have the same behaviour in expectation. However, we would like agents to exhibit heterogeneous behaviour. Consequently, we devise a method to produce variability in actions among agents with the same degree d. This is the subject of the next section.

Table 2. The four temporal variables of players $v_k \in V$ in the GrAG and model. The temporal vector is $\mathbf{z} = (1, Z_B(t), Z_L(t), Z_W(t), Z_C(t))^T_{5\times 1}$.

Item	Variable	Description
1	$Z_B(t)$	Size of the buffer of letter requests that v_k has yet to reply to at time t.
2	$Z_L(t)$	Number of letters that player v_k has available to use at t to form words.
3	$Z_W(t)$	Number of valid words that v_k has formed up to t.
4	$Z_C(t)$	Number of consecutive time steps that v_k has taken the same action.

3. The proposed uncertainty quantification framework

In this section, we detail the proposed UQ framework. Section 3.1 focuses on clustering for players based on

their activity in a game, such that one can better quantify their heterogeneity. Section 3.2 describes the UQ for model parameters estimated by the multinomial logistic model for each cluster. Section 3.3 details the ABM simulation using the obtained UQ for GrAGs.

3.1. Clustering methods for players

To quantify the uncertainty of player behaviours, a key goal is to partition players based on their activity. The number of letters a player requests, the number of letter requests a player replies to, and the number of words a player forms in a game are used to quantify a player's activity. We define two variables, *engagements* and *words*. *Engagements* is the sum of the number of requests and number of replies of a player, and *words* is the number of words a player forms in a game. The *engagements* and *words* are used to partition players.

In experiments, the number of neighbours that a player had was either $d = 2$, 4, 6, or 8. All players had the same number of neighbours in one game so that they could gather multiple sets of data on players with the same d. We want to study whether players should be partitioned based on their number of neighbours since a player with more neighbours can request more letters and reply to more letter requests. Figure 4 shows the numbers of *engagements* and *words* for players with different numbers of neighbours. It is shown that *engagements* increase with the number of neighbours, but become saturated when the number of neighbours is greater than four. The numbers of words are nearly the same for different numbers of neighbours. In summary, game data for players with $d = 4$, 6, and 8 neighbours are observed to be similar, and these data are different from those of players with $d = 2$ neighbours.

To determine whether we can divide players into separate groups, we conduct hypothesis testing of two-sample t-tests on *engagements* and *words*:

$$H_0^{r,s,eng}: \mu_r^{eng} = \mu_s^{eng} \text{ vs. } H_1^{r,s,eng}: \mu_r^{eng} \neq \mu_s^{eng},$$

$$H_0^{r,s,word}: \mu_r^{word} = \mu_s^{word} \text{ vs. } H_1^{r,s,word}: \mu_r^{word} \neq \mu_s^{word},$$

$$r, s = 2, 4, 6, 8, \quad \text{and } r < s,$$

where μ_d^{eng} and μ_d^{word} for $d = 2, 4, 6, 8$ are the mean *engagements* and *words* for players with d neighbours, respectively. We also denote μ_{468}^{eng} and μ_{468}^{word} as the mean numbers of *engagements* and *words* for players with either 4, 6, or 8 neighbours, respectively, and perform hypothesis testing on players with 2 neighbours and players with 4, 6, or 8 neighbours:

$$H_0^{eng}: \mu_2^{eng} = \mu_{468}^{eng} \text{ vs. } H_1^{eng}: \mu_2^{eng} \neq \mu_{468}^{eng},$$

$$H_0^{word}: \mu_2^{word} = \mu_{468}^{word} \text{ vs. } H_1^{word}: \mu_2^{word} \neq \mu_{468}^{word}.$$

After partitioning players into two groups based on their number of neighbours, we use the k-means clustering method (Hartigan & Wong, 1979) in each group to quantify the variability in players' abilities. Before clustering players, the *engagement* and *words* are standardised first, so no variable would dominate the clustering. To determine the number of clusters in the k-means clustering, we use the Bayesian Information Criterion (BIC) as our criterion (Li et al., 2016). Based on the BIC and the size of data, we select four clusters, which gives the smallest BIC values. After clustering players in each group, we denote the formed clusters as $\mathcal{D}_{[g,c]}^{(i)}$, $i = 1, 2, 3, 4$ for data with initial action a_i, where $g = 1, 2$ is the group number and $c = 1, 2, 3, 4$ is the cluster number.

3.2. Quantifying uncertainty for model parameters

In the mean multinomial logistic regression model (Cedeno-Mieles et al., 2020) of Section 2, the next

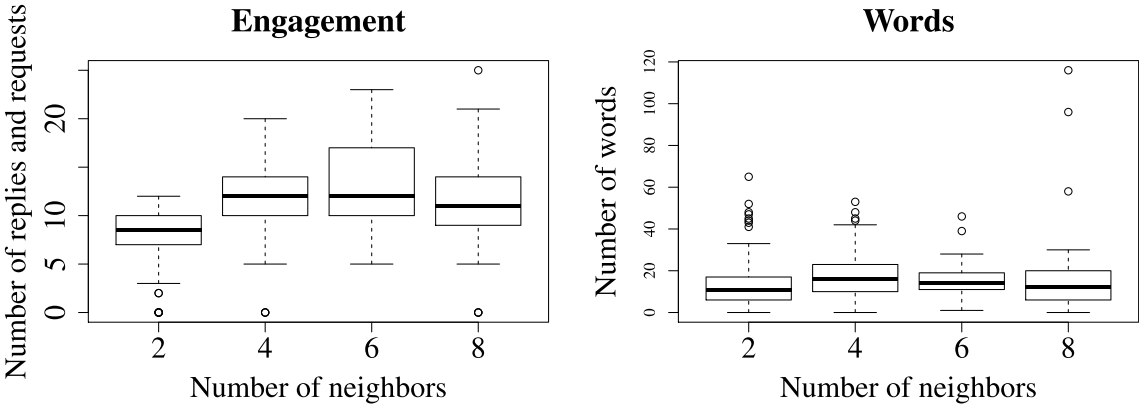


Figure 4. Boxplots of *engagements* and *words*; data come from experiments. The left plot shows *engagements* of each player in games with $d = 2$, 4, 6, 8 neighbours. The right plot shows number of words of each player in games with 2, 4, 6, and 8 neighbours.

player action depends on the parameter matrix $B^{(i)}$ and input vector $\mathbf{z}$ by mapping them to a probability vector $\pi_i = (\pi_{i1}, \pi_{i2}, \pi_{i3}, \pi_{i4})^T$ through Equation (1). The uncertainty in the estimated $\hat{\boldsymbol{B}}^{(i)}$ matrix results in different probabilities π_{ij}, therefore, leading to diversity in outcome actions. To quantify these uncertainties, we use two approaches, the CWM and CWUQ approaches from Figure 2, which are described next.

3.2.1. Method 1: Within-cluster mean CWM approach

We apply the multinomial logistic regression to each cluster, and parameter matrices $B^{(i)}_{[g,c]}$ are estimated for each cluster in each group. In this approach, uncertainty quantification depends on clustering players, and the $B^{(i)}_{[g,c]}$ matrix will be the same for all players in the same cluster. Thus, they will have the same probability vectors given the same set of $\mathbf{z}$ vector values. We compare this approach with the approach below, which quantifies the uncertainty of model parameters.

3.2.2. Method 2: Within-cluster uncertainty quantification (UQ) CWUQ approach

In order to study the heterogeneous behaviour of players in the same cluster, we utilise the asymptotic normality property of MLE to quantify the uncertainty for parameter matrix $B^{(i)}_{[g,c]}$. In this way, different $B^{(i)}_{[g,c]}$ matrices can be sampled from the asymptotic normal distribution, representing different behaviours of players. Without loss of generality, we omit the subscript $[g, c]$ in parameter matrix $B^{(i)}_{[g,c]}$ and transform it to the parameter vector $\beta = \beta^{(i)} = (\beta_2^{(i)T}, \beta_3^{(i)T}, \beta_4^{(i)T})^T_{15\times 1}$ (Action idling a_1 is treated as a reference group so $\beta_1^{(i)}$ will not show up in the parameter vector, leaving the other three 5×1 vectors $\beta_2^{(i)}$, $\beta_3^{(i)}$, and $\beta_4^{(i)}$ in β). Then we use the asymptotic property of MLE (Sweeting, 1980). That is, as sample size increases, the maximum likelihood estimator $\hat{\beta}_{mle}$ of parameter β approximates a multivariate normal random variable

$$\hat{\beta}_{mle} \xrightarrow{d} \mathrm{MN}(\beta, \Sigma = I(\beta)^{-1}/n), \tag{4}$$

where $I(\cdot)$ is the Fisher information matrix (Fisher, 1922) and n is number of observations in $\mathcal{D}^{(i)}_{[g,c]}$. We can estimate β with the MLE $\hat{\beta}_{mle}$ and covariance matrix Σ with $I(\hat{\beta}_{mle})^{-1}/n$. Then, we directly draw samples from the asymptotic normal distribution in Equation (4). Consequently, we can calculate the corresponding probability vector based on the sampled $\hat{\beta}$, again using Equation (1). Note that the use of asymptotic normal distribution requires a large sample size. In our data, there can be situations with small sample sizes for certain pairs of actions. For instance, among 311 observations in group 1 cluster 2, where the initial action is request (a_3), only two observations transition to reply (a_2), three observations transition to request (a_3) again, and there are no observations that transition to forming words (a_4). The remainder of the transitions is to idle (a_1). When the pair of actions occur infrequently, we avoid the use of the asymptotic distribution and instead utilise point estimators.

3.3. Simulation models

The simulation system models the GrAG of Figure 1. Simulation input parameters are provided in Table 3. This table also thereby provides much of the configuration of a simulation. The parameters are divided into three sections by three

Table 3. Summary of parameters and their values used in simulations of GrAGs. The first section contains variables that are physical entities that map directly to a GrAG. The second section contains model parameters that prescribe node (i.e., player, agent) behaviours. The third section contains the simulation parameter. Sections are delineated by three horizontal lines.

Parameter	Description
Networks $G(V, E)$.	Two networks: (i) the star graph of Figure 7 and (ii) the group of cliques in Figure 8.
Number n_ℓ of owned letters.	The number of owned letters initially assigned to a player.
Initial letters L_k^{init}.	The set of initial n_ℓ letters assigned to a player v_k.
Word corpus C^W.	The corpus of 1015 3-letter words is taken from *http://www.wordfind.com/3-letter-words/*, accessed January 12 2018. (At the time of this writing, eight words have presumably been removed, since the web page shows 1007 words.) Only 3-letter words are considered in simulations.
Duration of GrAG t_g.	GrAG duration is fixed at $t_g = 300$ seconds.
Group, g.	There are two groups: $g = 1$ corresponds to nodes with degree $d \leq 2$ in the game network and $g = 2$ corresponds to nodes with degree $d \geq 3$.
Cluster, c.	For each group g, there are four clusters (c): $c = 1$ through 4.
Group-cluster $[g, c]$.	The group-cluster pair $[g, c]$ determines the behaviour regime for each node.
Behaviour classes C.	There are two behaviour classes: the CWM model C_μ and the CWUQ model C_β. Each node is assigned a behaviour class.
Game player behaviour models $\mathcal{M}$.	Each player in a GrAG is assigned a behaviour model M, which consists of the triple $\mathrm{M} = [C, g, c]$.
Player actions a.	The set A of actions a is given in Table 1.
Number of iterations n_{iters}.	Each simulation is composed of $n_{iters} = 50$ individual dynamics instances, where each instance starts from time $t = 0$, with initial conditions reset, and then the dynamics of the system are executed for t_g discrete time steps.

horizontal lines. The first section contains physical parameters of the game. The second section contains parameters of behaviour models that simulate player actions in a GrAG. These are produced from the methods described in this section. The third section of the table, with only one value, is purely a simulation parameter: the number of simulation instances to perform in order to address stochasticity of the models. Details on parameter selections are given in Section 5.1 on the simulations; the purpose here is to list the parameters because they support the simulation models. All parameters, including the word corpus, can easily be changed through configuration files to run additional simulations.

Equation (1) provides the key computations of the ABM in generating the probabilities π_{ij} of an agent v_k taking action a_j, $j \in \{1,2,3,4\}$, at time t given its most recent action a_i at time $(t-1)$, where actions are given in Table 1. This equation is used for each of the CWM and CWUQ models (the $B^{(i)}$ matrices vary between models). Game behaviour data for players in each cluster c are used to fit a per-cluster model. Further, the network structure defines the group g to which each agent belongs. Thus, in total, a game player behaviour model $\mathcal{M}$ is given as $\mathcal{M} = [C, g, c]$, where C is the behaviour class, g is the group, and c is the cluster number. Hence, the CWM model is $\mathcal{M}_\mu = [C_\mu, g, c]$ and the CWUQ model is $\mathcal{M}_\beta = [C_\beta, g, c]$.

A **simulation** is composed of a collection of simulations instances (also called iterations or runs). An **iteration** is a sequence of simulation steps from time $t = 0$ to t_g seconds, in one-second time steps, such that actions of all players are computed at each t. The state of the system at $t = 0$ constitutes the initial conditions for a simulation instance. Within one simulation, all instances have the same initial conditions. These initial conditions and properties (see Table 3) are read from various input files. The processes of forming words and sharing letters in an iteration are shown in Figure 1.

In each iteration, the letters that a player can share with her neighbours are her owned (i.e., initially assigned) letters. There may be duplicate letters between pairs of players, including neighbours of an agent v_k. A player v_k that receives a letter from a neighbour v_i cannot then share that letter with a different neighbour v_ℓ. This is in accordance with the experiment rules. Table 4 defines internal variables used in the algorithms of the simulation system.

Algorithm 1 provides the overall simulation structure: reading inputs, initialising variables, and iterating over simulation iterations, time, and agents $v_k \in V$. Algorithm 2, invoked from Algorithm 1 on step E.2.*i.c.*, provides the steps for computing the next action $a_j(t)$ for node or agent or game player v_k, at each (iteration, time) pair.

Algorithm 1: Algorithm NetworkedGroupAnagramGame.

1 Input: Data in Table 1 through 3.
2 Output: (*i*) a_j; (*ii*) p_{act}; (*iii*) z; and (*iv*) η. Each of these outputs is prefaced with the iteration number *iter*, time t, and node or player or agent ID v_k.
3 Steps:
 # Read inputs from files.
A. $G(V, E)$, and compute $N[k]$ for each agent $v_k \in V$.
B. C^W, n_{iters}, and t_g.
C. **for each** $v_k \in V$:
1. L_k^{init}, C, g, c, and ρ.
2. C, g, c over all $C \in \{C_\mu, C_\beta\}$, $g \in \{1,2\}$, and $c \in \{1,2,3,4\}$.
D. Set L'_k with all L_i^{init} for all $v_i \in N[k]$.
 # Do simulations over iterations and over time and over nodes/agents.
E. **for** *iter* $= 1$ **to** n_{iters}:
1. **for each** ($v_k \in V$): Reset $L_k^{ih} = L_k^{init}$, $z = 0$, $B_k^1 = \emptyset$, $B_k^2 = \emptyset$, $W_k = \emptyset$.
2. **for** $t = 1$ **to** t_g:
i. **for each** $v_k \in V$:
a. Receive all letter requests from neighbors $N[k]$ of v_k, sent to v_k at the previous time $(t-1)$, and put in buffer B_k^1.
b. Receive all letter replies from v_k's neighbors that are in response to v_k's letter requests, sent to v_k at the previous time $(t-1)$, and put in L_k^{ih}; mark this letter request in B_k^2 as fulfilled.
c. Call Algorithm 2, (Vertex Action), computing v_k's next action.
d. Write next action and other variables in **Output** section above.

Algorithm 2: Algorithm Vertex Action for vertex v_k.

1 Input: t, v_k, $a_i(t-1)$, z, $N[k]$, L'_k, L_k^{ih}, C^W, W_k, B_k^1, B_k^2, $\mathcal{M} = [C, g, c]$, and ρ.
2 Output: $a_j(t)$, z, L_k^{ih}, W_k, B_k^1, B_k^2 as appropriate.
3 Steps:
A. Using the property index for v_k from ρ, retrieve properties for $\mathcal{M}$ from values $[C, g, c]$. Compute all π_{ij}, $j \in \{1,2,3,4\}$ from Equation (1), yielding $p_{act} = (\pi_{i1}, \pi_{i2}, \pi_{i3}, \pi_{i4})$.
B. Uniformly draw a random number $r \in [0,1]$ to determine the next action $a_j(t)$ for v_k using p_{act}.
C. **if** $a_j(t)$ **equals** a_1 **do** ## Action $a_j(t)$ is think/idle.
i. Do nothing.
D. **else if** $a_j(t)$ **equals** a_2 **do** ## Action $a_j(t)$ is reply (with letter).
i. If there is a letter request from a neighbor of v_k in B_k^1, that is waiting to be fulfilled, then send a letter reply using FIFO ordering, and mark the request in B_k^1 as fulfilled. Otherwise, do nothing.
E. **else if** $a_j(t)$ **equals** a_3 **do** ## Action $a_j(t)$ is send letter request.
i. If there is a letter $\ell \in L'_k$ that is not in the buffer of letter requests B_k^2 for v_k, choose letter ℓ at random, send a letter request to the appropriate neighbor that possesses ℓ, add the letter to the request buffer B_k^2, and mark the request as sent. Otherwise, do nothing.
F. **else** $a_j(t)$ **equals** a_4 **do** ## Action $a_j(t)$ is form word.
i. Select randomly a word $w \in C^W$, where $w \notin W_k$ and can be formed with letters in L_k^{ih}. Add w to W_k. If there is no such $w \in C^W$, do nothing.
G. Return updated variable values in **Output** section for vertex v_k.

4. Model evaluation

We evaluate the UQ method of Section 3, and in Section 5, we use these evaluation results to reason about ABS output.

Table 4. Summary of additional parameters used in the simulation algorithms. Most of these evolve in time during a simulation.

Parameter	Description
Neighbours of a node $N[k]$.	Set of neighbours of a node v_k in a graph G.
Mapping ρ.	Map of agent v_k, for each $v_k \in V$, to its model M_k.
Action probabilities p_{act}.	The vector of probabilities of taking actions, computed for each v_k at each t, $p_{act} = (\pi_{i1}, \pi_{i2}, \pi_{i3}, \pi_{i4})$ per Equation (1).
Letters in hand L_k^{ih}.	The set of letters that a player v_k has, at any time t during a game (superscript ih is for 'in hand').
Neighbouring letters L'_k.	The set of letters of the neighbours of v_k that v_k can request.
Buffer of letters B_k^1.	The buffer of letter requests that v_k has received.
Buffer of letters B_k^2.	The buffer of letter requests that v_k has made to neighbours.
Words W_k.	The set of words already formed by v_k.
Numbers of actions η.	The counts of actions for each v_k and each t, $\eta = (\eta_{words,k}, \eta_{reqSent,k}, \eta_{reqRec,k}, \eta_{replSent,k}, \eta_{replRec,k})$, which are, respectively, the number of words formed, number of requests sent, number of requests received, number of replies sent, and number of replies received.

Table 5. Pairwise comparisons of engagements and pairwise comparisons of words. The numbers are p-values of two-sided two-sample t-tests.

Engagements		Words	
number of neighbours	p-value	number of neighbours	p-value
2 vs. 4	8.286e-13	2 vs. 4	2.151e-03
2 vs. 6	8.386e-05	2 vs. 6	0.236
2 vs. 8	1.418e-04	2 vs. 8	0.210
4 vs. 6	0.269	4 vs. 6	0.589
4 vs. 8	0.641	4 vs. 8	0.967
6 vs. 8	0.202	6 vs. 8	0.749
2 vs. 468	5.009e-18	2 vs. 468	3.231e-03

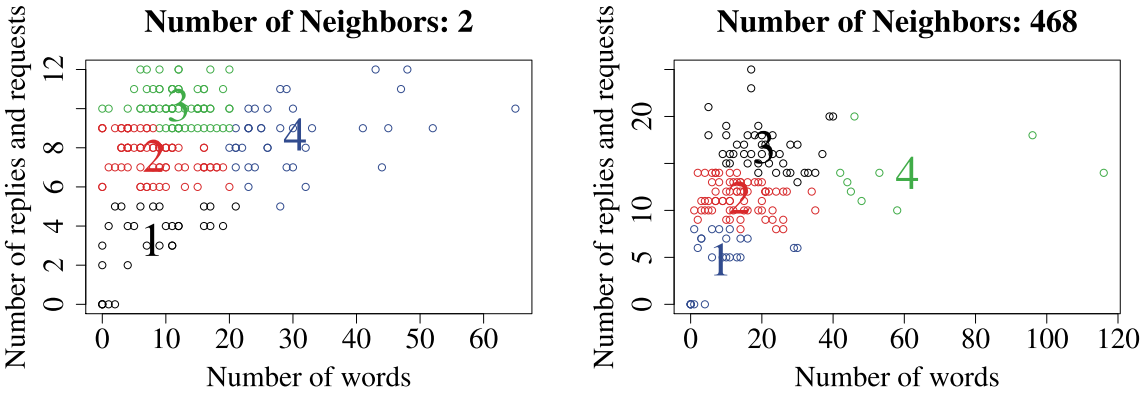
4.1. Clustering players

Table 5 shows the p-values of two-sample t-tests. For *engagements*, the p-values show that 2 neighbours is significantly different from 4, 6, and 8 neighbours, while pairs of values among 4, 6, and 8 neighbours are not significantly different. For *words*, the p-values show that 2 neighbours is significantly different from 4 neighbours, while, again, pairs of values among 4, 6, and 8 neighbours are not significantly different. Though 2 neighbours is not significantly different from 6 and 8 neighbours, respectively, 2 neighbours is significantly different from 4, 6, and 8 neighbours together. This means that we can collect players into two groups: those players with 2 neighbours [group $g = 1$] and those players with 4, 6, or 8 neighbours [group $g = 2$].

Figures 5(a,b) show the clustering results using the k-means method. The left plot is for 2 neighbours, and the right plot is for 4, 6, and 8 neighbours. Different clusters are marked with different colours and numbers, 1 through 4. In Figure 5(a), the black cluster is the least active, and the blue cluster is the most active. In Figure 5(b), the blue cluster is the least active, and the green cluster is the most active. Figure 6 provides the same data, but the mean and median points within each cluster are shown.

4.2. Quantifying uncertainty for model parameters

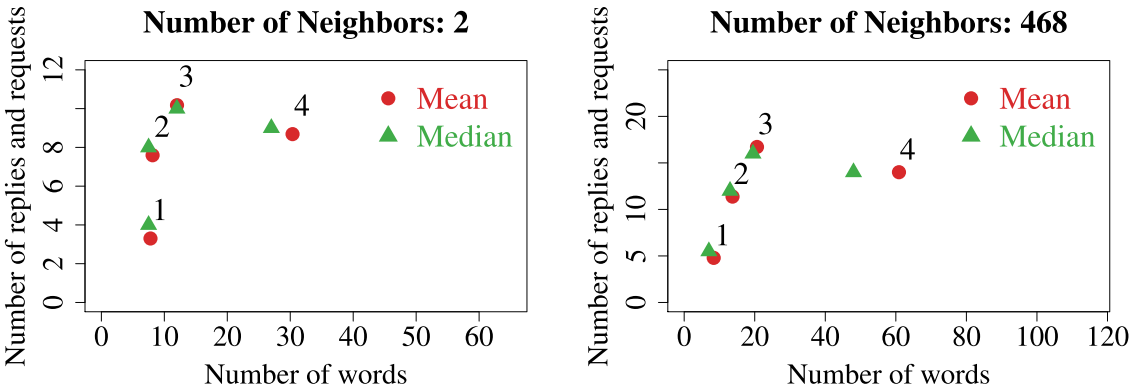
Table 6 shows one set of $\mathbf{z}$ values, for the group $g = 1$, and we use these $\mathbf{z}$ values to generate heterogeneous probability vectors. Table 7 shows generated probability vectors sampled from the asymptotic normal distribution of $\hat{\beta}$ for group $g = 1$. The four clusters correspond to clusters in Figure 5(a). In each cluster, the first row provides the probability vector from the CWM model, and the bottom four rows provide generated probability vectors from the CWUQ model. Players in cluster



(a) Cluster plot for 2 neighbors

(b) Cluster plot for 4, 6, and 8 neighbors

Figure 5. Data from human subject anagram games, showing results of k-means clustering. Scatter plots of number of words formed against number of replies and requests (engagements). Each data point represents one game player. Data points in different clusters are denoted in different colors and are numbered, 1 through 4. (a) Data for $d = 2$ neighbours [group $g = 1$]. (b) Data for $d = 4$, 6, and 8 neighbours [group $g = 2$].



(a) Cluster center plot for 2 neighbors (b) Cluster center plot for 4, 6, and 8 neighbors

Figure 6. Data from human subject anagram games, showing results of k-means clustering. Center plots of number of words formed against number of replies and requests (engagements). (a) Data for $d = 2$ neighbours [group $g = 1$]. (b) Data for $d = 4$, 6, and 8 neighbours [group $g = 2$].

Table 6. z vector values of $Z_B(t)$, $Z_L(t)$, $Z_W(t)$, and $Z_C(t)$.

Initial state	Number of neighbours	buffer	letter	word	constant
a_1 (idle)	2	0	3	1	5

4 have lower to-idle ($a_1 \to a_1$) transition probability than players in other clusters, so players in cluster 4 are the most active ones in group $g = 1$, which can be confirmed in Figures 5(a).

5. Agent-based simulation results from networked group anagram games

In this section, we present simulation scenarios and results and explanations from the simulations. Note that the simulation process of the game follows very closely the actual experimental procedures, by design, so that experimental data could be used to develop agent models of player behaviours for the agent-based simulations (ABSs). The simulation models are presented in Section 3.3.

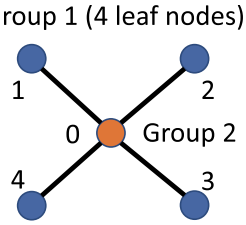


Figure 7. Star network used for simulations. Nodes are game players and edges represent communication channels that can be used to share letters. The centre node (hub) has degree $d = 4$ and hence is in group $g = 2$ and the four leaf nodes each have degree $d = 1$ and so are in group $g = 1$.

5.1. Simulation scenarios

We analyse two anagram game configurations of players. Figure 7 is a star-4 graph, with a hub node 0 and four leaf nodes. This network was analysed in Hu et al. (2021). We include new results for it in Appendix B. The second graph is a graph on 18 nodes consisting of four clique subgraphs (K_3, K_4, K_5, and K_6), where K_j is a clique on j

Table 7. Sampling from the asymptotic normal distribution of $\hat{\beta}$, where the initial state is a_1 (idle) and the group $g = 1$. The top left part is for cluster 1, the top right part is for cluster 2, the bottom left part is for cluster 3, and the bottom right part is for cluster 4. One can see these clusters in Figures 5(a). Each row is a probability vector for next actions. The first row (mean) is from the cluster-wise mean model (CWM), and the bottom four rows are four samples from the cluster-wise uncertainty quantification model (CWUQ). $a_i \to a_j$ means transition from a_i to a_j, so the value under each column represents the probability of next action (e.g., idling, replying with letter, requesting letter, and forming words).

Cluster 1					Cluster 2				
	$a_1 \to a_1$	$a_1 \to a_2$	$a_1 \to a_3$	$a_1 \to a_4$		$a_1 \to a_1$	$a_1 \to a_2$	$a_1 \to a_3$	$a_1 \to a_4$
Mean	0.957	0.006	0.014	0.022	mean	0.942	0.010	0.031	0.017
Sample 1	0.955	0.008	0.014	0.023	sample 1	0.944	0.009	0.028	0.019
Sample 2	0.957	0.006	0.015	0.023	sample 2	0.944	0.010	0.029	0.018
Sample 3	0.958	0.009	0.013	0.019	sample 3	0.941	0.010	0.031	0.018
Sample 4	0.951	0.006	0.019	0.024	sample 4	0.942	0.008	0.031	0.019
Cluster 3					**Cluster 4**				
	$a_1 \to a_1$	$a_1 \to a_2$	$a_1 \to a_3$	$a_1 \to a_4$		$a_1 \to a_1$	$a_1 \to a_2$	$a_1 \to a_3$	$a_1 \to a_4$
Mean	0.917	0.018	0.043	0.023	mean	0.880	0.020	0.051	0.049
Sample 1	0.923	0.017	0.040	0.020	sample 1	0.868	0.019	0.060	0.054
Sample 2	0.922	0.018	0.040	0.020	sample 2	0.868	0.019	0.057	0.056
Sample 3	0.912	0.020	0.046	0.022	sample 3	0.874	0.023	0.053	0.050
Sample 4	0.914	0.018	0.047	0.022	sample 4	0.869	0.022	0.060	0.050

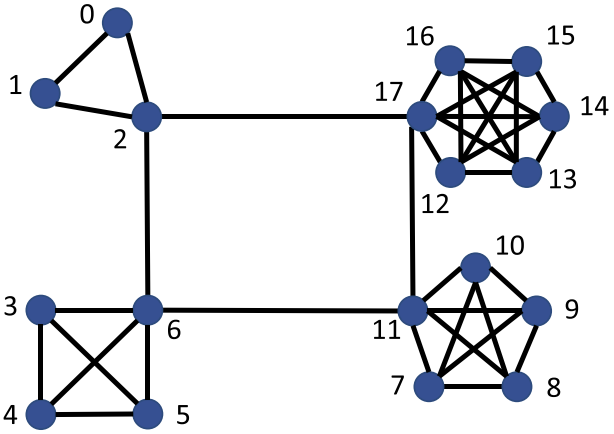


Figure 8. Network of four connected cliques (K_3, K_4, K_5, and K_6) used for simulations. Nodes are game players and edges represent communication channels that can be used to share letters. Nodes 0 and 1 are the only nodes with degree $d \leq 2$ so are in group $g = 1$. All other nodes have degree $d \geq 3$ and hence are in group $g = 2$.

nodes. These four cliques are connected in a circle arrangement. See Figure 8.

These two graphs are motivated by the following considerations. First, our two groups ($g = 1$ and 2) are based on the degrees of nodes. Hence, we devised one graph that had multiple nodes of degree $d = 1$ so that these nodes are in $g = 1$. This is the star graph. We devised the second graph that had multiple nodes of degree $d \geq 3$ so that these nodes are in $g = 2$. This is the 4-clique graph. Second, we want multiple nodes in a graph to have the same degrees and neighbourhoods so that we can assess variability in behaviours across nodes. For this goal, we have four nodes with degree $d = 1$ in the star graph, and we have multiple similar nodes in each clique of the 4-clique graph. Third, for the 4-clique graph, we want multiple cliques, of different sizes, so that we can assess differences in agent or node behaviours as degree increases. Also, these network configurations are different from those used in experiments, thus illustrating the ability of simulations to address a wide range of player interaction patterns.

This section uses the 4-clique graph because it is larger and more nuanced, enabling more detailed analyses. Many simulations are performed on each game configuration where input parameters are varied.

Numbers n_ℓ of letters are specified and specific letter assignments are made so that players (agents) can form words when they choose this action. Four letters are used per player in the star graph, since some players have only one neighbour, and three letters are used per player in the cliques graph because each node has more neighbours. One goal in these simulations is to determine how many words a player can form. Consequently, the initial letter assignments to players are done by human decision-making, to ensure that a sufficient number of vowels and that often-occurring consonants are either owned by players or can be requested from neighbours. At the other end of the spectrum, we can assign players very poor letters (e.g., x, y, z, q) so that no matter the model, a player cannot form words. We have not done this in this work because we want to understand model behaviours. The game duration is the same as that used in experiments, and a word corpus is used to determine valid words that agents form.

In this work, each simulation is comprised of $n_{iters} = 50$ iterations, and results are presented as time point-wise averages over all 50 instances and as boxplots that also account for the data of all 50 iterations. Fifty iterations provide mean results consistent with those for simulations between 30 and 50 iterations.

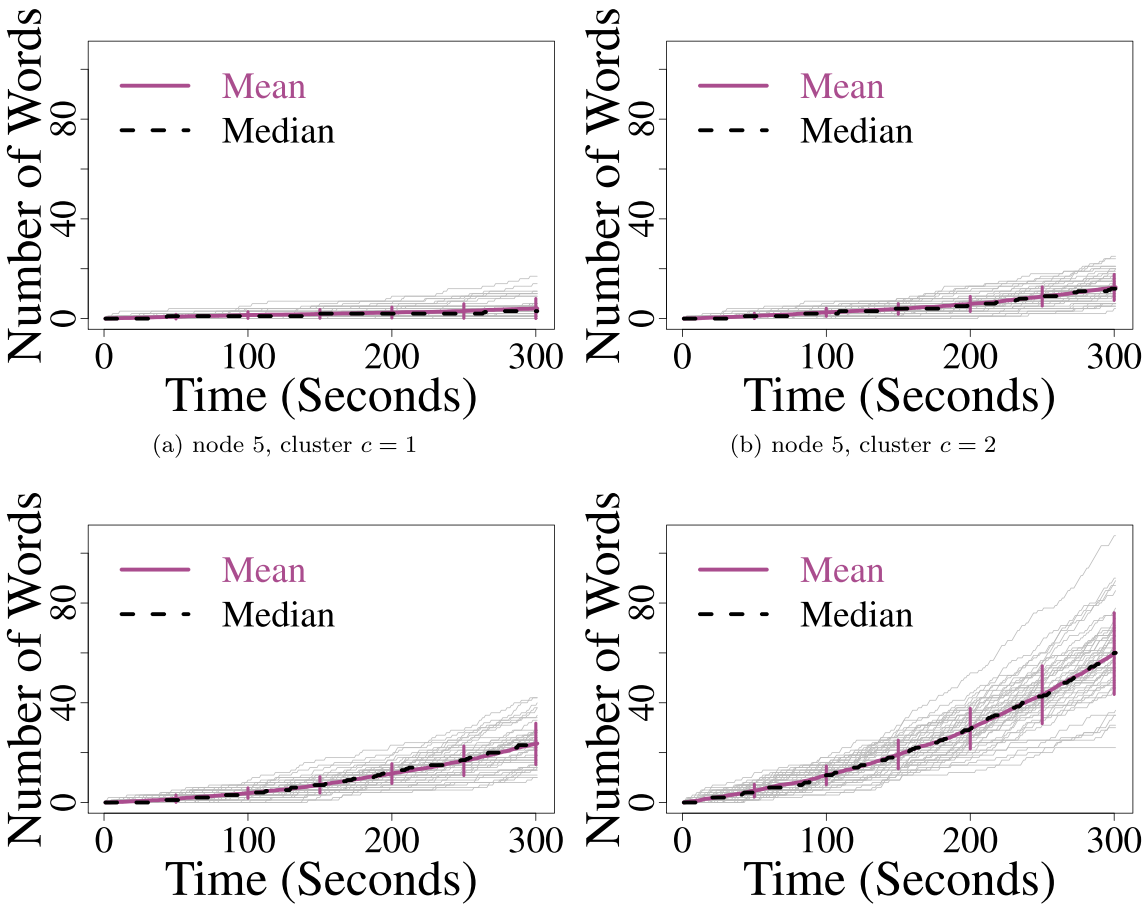
5.2. Simulation results

5.2.1. Basic time history results

Figure 9 shows variability in results across all 50 runs or instances of one simulation, for the CWM model and various values of group g and cluster c, i.e., $[g, c]$. The results are number of words formed by node (i.e., player) 5 as a function of game time. The four plots correspond to node 5 of the 4-clique network with behaviour models assigned according to $[g, c] = [2, 1]$, [2, 2], [2, 3], and [2, 4], respectively, where $g = 2$ because node 5 has degree $d \geq 3$, i.e., $d_5 = 3$. In each plot, the 50 gray curves are results from the 50 runs, the magenta curve is the time point-wise average with $\pm$ one standard deviation, and the black curve is the time point-wise median value. The results show that the individual curves (i.e., simulation instance results in gray) across the 50 runs can vary considerably, with the largest variations occurring for $[2, 4]$ in Figure 9(d): the range in numbers of words at $t = 300$ seconds is from 20 to 110 words. Also, the cluster-to-cluster differences can be large, particularly when comparing with the behaviour of cluster 4. These ranges of variability can change with C and $[g, c]$, i.e., with model $\mathcal{M}$, and result from the variability in the data of Section 4.

5.2.2. Comparisons of time histories of similar nodes over all player actions

We focus on the K_6 clique of the graph in Figure 8 and specifically, the behaviours of nodes 12 through 14. Figure 10 shows the time histories for the five types of events: number of letter replies received (replRec) in response to this player's requests for letters from its neighbours; number of letter replies sent (replSent) in response to letter requests that it receives; the number of requests for letters that the node receives from neighbours (reqRec); the number of requests for letters that this player sends to neighbours (reqSent); and the number of words (words) that this player forms. Each plot is data for one node. The first column of plots is for the CWM model $\mathcal{M}_\mu = [C, g, c] = [C_\mu, 2, 2]$, and



(a) node 5, cluster $c = 1$ (b) node 5, cluster $c = 2$

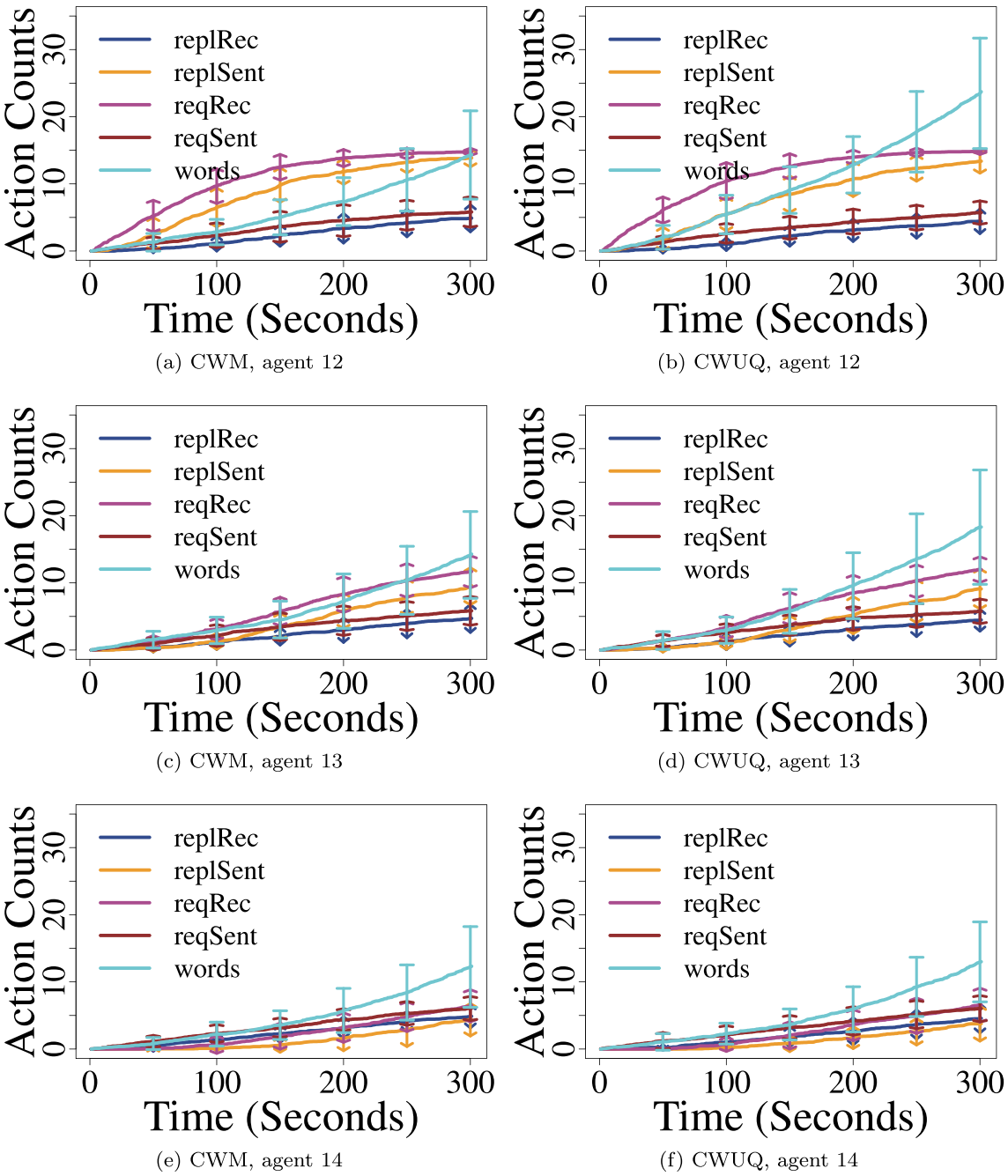
(c) node 5, cluster $c = 3$ (d) node 5, cluster $c = 4$

Figure 9. Agent-based simulation results for node 5, for a game configuration of a 4-clique graph (see Figure 8). Plots are the number of words formed by this player as a function of game time. Each plot has 50 gray curves of numbers of words formed as a function of time; one curve for each simulation run or instance. The magenta curve is the time point-wise average over the 50 instances, with error bars for $\pm$ one standard deviation. The black curve is the median time point-wise average for the 50 instances. In all simulations, the two degree $d = 2$ nodes have C of CWM class and $[g, c] = [1, 2]$. All 16 nodes with $d \geq 3$ use the same class C and $[g, c]$ according to: (a) $[2, 1]$; (b) $[2, 2]$; (c) $[2, 3]$; and (d) $[2, 4]$. These results demonstrate that the cluster c assigned to node 5 results in different behaviours.

the second column of plots is the corresponding data for the CWUQ model $\mathcal{M}_\beta = [C, g, c] = [C_\beta, 2, 2]$. (The two low degree nodes in the 4-clique graph use the same two models $\mathcal{M}_\mu$ and $\mathcal{M}_\beta$, with $[g, c] = [1, 2]$.)

Nodes 12 through 14 have the same properties and same connectivity (i.e., the same neighbours), so their behaviours should be the same, modulo stochasticity. We see that for both models, the behaviours of these nodes can vary node-to-node. Nodes 13 and 14 have differences in numbers of letter requests received (magenta, reqRec) and of letter replies sent (orange, replSent). The differences are greatest for node 12, in both models, where the magenta and orange curves are concave down; the corresponding curves for nodes 13 and 14 are less distinctive. Across all nodes, the time histories of letter requests sent (brown, reqSent) and letter replies received (blue, replRec) are similar. The average number of words formed varies between 11 and 24 across nodes. These data indicate that variability in results can be generated due to the stochasticity of each behaviour model, per Section 4. However, the error bars cover these differences in average values. It is also observed that replies sent (in response to letter requests, orange) lags the letter requests received (magenta) for all nodes.

Examining general trends in the behaviours of all models and conditions, we observe the following. Players request letters throughout the game. They reply to letters throughout the game. That is, they do not request all neighbouring letters at the outset of a game, which is one strategy; the game data do not exhibit this behaviour and hence neither do our models. Generally, simulation results show that players request all neighbour letters during a game. In a game, the number of letter requests that can be made of neighbours (reqSent) is bounded by the number of letters a player originally possesses and by the number of neighbours. This, in turn, affects all other sharing types of actions: letter requests received (by a neighbour), letter replies sent, and letter replies received. Hence, the variability in these quantities is



(a) CWM, agent 12 (b) CWUQ, agent 12

(c) CWM, agent 13 (d) CWUQ, agent 13

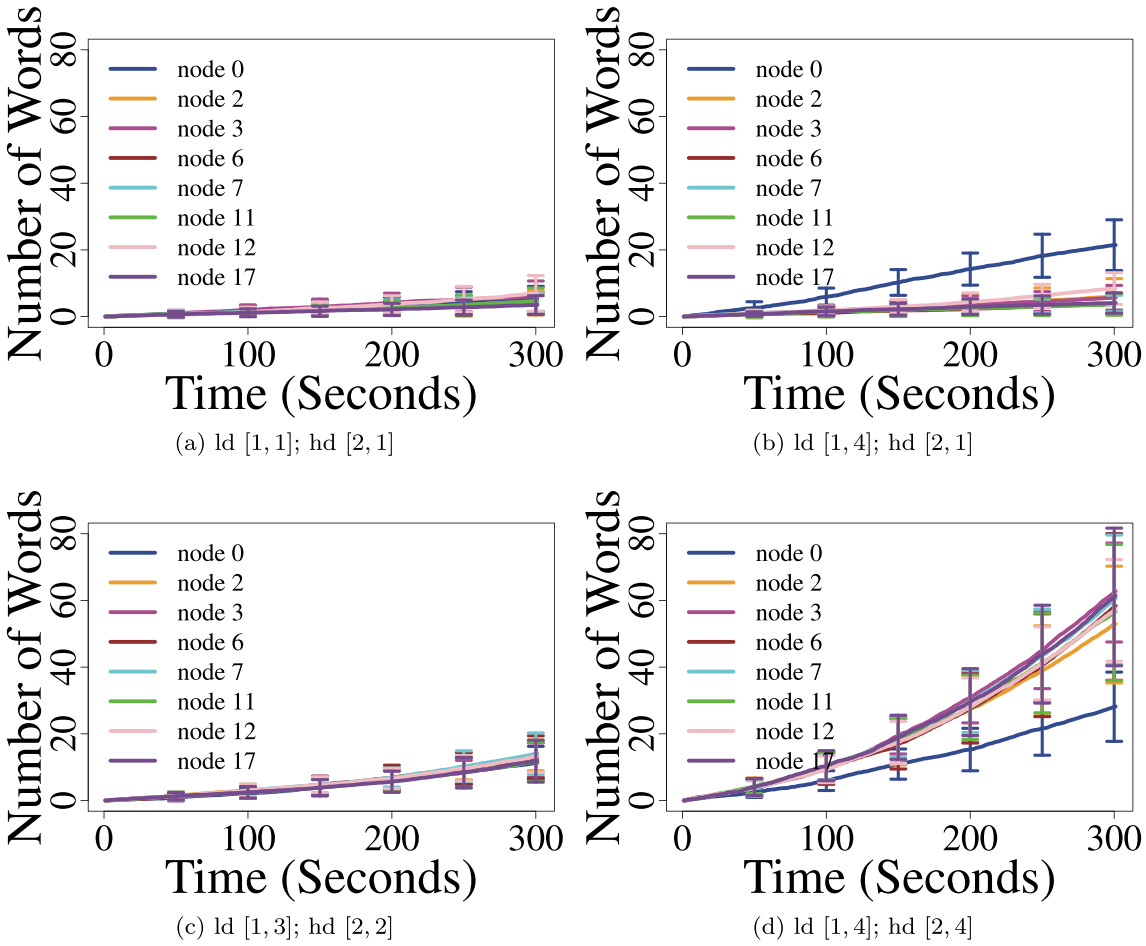
(e) CWM, agent 14 (f) CWUQ, agent 14

Figure 10. Agent-based simulation results for game configuration of the 4-clique network. Curves are mean time point-wise data for nodes 12 through 14; all nodes are in the K_6 clique. The first column of results are for the CWM model $\mathcal{M}_\mu$ and the second column contains corresponding results for the CWUQ model $\mathcal{M}_\beta$. All nodes have $[g, c] = [2, 2]$; $g = 2$ since all degrees are $d \geq 3$. The magenta curves for (letter) requests received (legend: reqRec) must be greater than or equal to the orange curves for (letter) replies sent (legend: replSent), and similarly the brown curve must be greater than or equal to the blue curve because a player must send at least as many (letter) requests sent (legend: reqSent) as (letter) replies received (legend: replRec).

lesser than that for words formed. There is no practical limit on the number of words a player can form and hence variability is greater. This is why we focus on numbers of words formed in subsequent results.

5.2.3. Comparisons of time histories of nodes from cliques using models that incorporate behaviours from different clusters

In this section, we examine the time histories of numbers of words formed for eight nodes of the 4-clique graph of Figure 8. For each of the four cliques, we chose two nodes: the unique node that is connected to two other cliques and one of the remaining nodes that is only connected to other nodes of the particular clique. Node IDs are given in the plot legends. Of the eight nodes, seven are high degree nodes, i.e., degree $d \geq 3$, and so are in group $g = 2$; only node 0 is low degree, i.e., degree $d \leq 2$, and so is in group $g = 1$. We intentionally select combinations of clusters for high and low degree groups to demonstrate wide range of behaviours that can be produced with the models.



(a) ld [1, 1]; hd [2, 1]

(b) ld [1, 4]; hd [2, 1]

(c) ld [1, 3]; hd [2, 2]

(d) ld [1, 4]; hd [2, 4]

Figure 11. ABS results for the CWM model $\mathcal{M}_\mu$ and the 4-clique network of Figure 8. In each plot, results for the same eight nodes are provided. Two nodes come from each of the four cliques, where one node is connected only to other nodes of the clique, and the other node is also connected to two other cliques in the overall network. The values plotted are mean time point-wise averages, $\pm$ one standard deviation, of numbers of words formed across all 50 instances of a simulation. The results for nodes 2 through 17 in each plot are similar (to within stochastic variation) because all of these nodes are in group $g = 2$ since each has degree $d \geq 3$. Thus, as the cluster for the high degree nodes changes across plots, the cluster is constant for each plot, so all of these nodes are assigned the same behaviour model. Only node 0 is in group $g = 1$ because it has degree $d = 2 \leq 2$. The $[g, c]$ pairs for low degree (ld) and high degree (hd) nodes are given under each plot. The plots show that different combinations of $[g, c]$ for ld and hd nodes can have large effects on the number of words formed in the game. For example, compare the blue curve versus the high-degree node curves in each of (b) and (d).

Figure 11 provides time histories for the eight chosen nodes for different combinations of clusters for group $g = 1$ low degree and group $g = 2$ high degree nodes, for the CWM model $\mathcal{M}_\mu$. In the first plot, low degree (ld) nodes have $[g, c] = [1, 1]$, while high degree (hd) nodes have $[g, c] = [2, 1]$. The blue curve for node 0 is in mid-range compared to the curves for other nodes. In the second plot, Figure 11(b), the only change is that the lone group $g = 1$ node is now assigned cluster 4 behaviour. The number of words formed by this node greatly increases (by 4×), while those for the high degree nodes remain similar to those in the first plot. In the third plot, Figure 11(c), the behaviour clusters for low degree and high degree nodes are 3 and 2, respectively. The numbers of words formed by each node are greater than those in Figure 11(a). The last combination of clusters in Figure 11(d) results in node 0 forming as many words as it did in Figure 11(b), but now nodes 2 through 17 roughly form 2× to 3× the number of words that are formed by node 0.

These results indicate that the behaviours of agents can change markedly when the clusters assigned to low and high degree agents change. Further, it demonstrates the efficacy of clustering player behaviours, by engagement and by words in Section 3, to capture differences in player performance. Clearly, heterogeneity in player behaviours is achieved.

5.2.4. Comparisons of time histories and end-of-game data between the CWM and CWUQ models

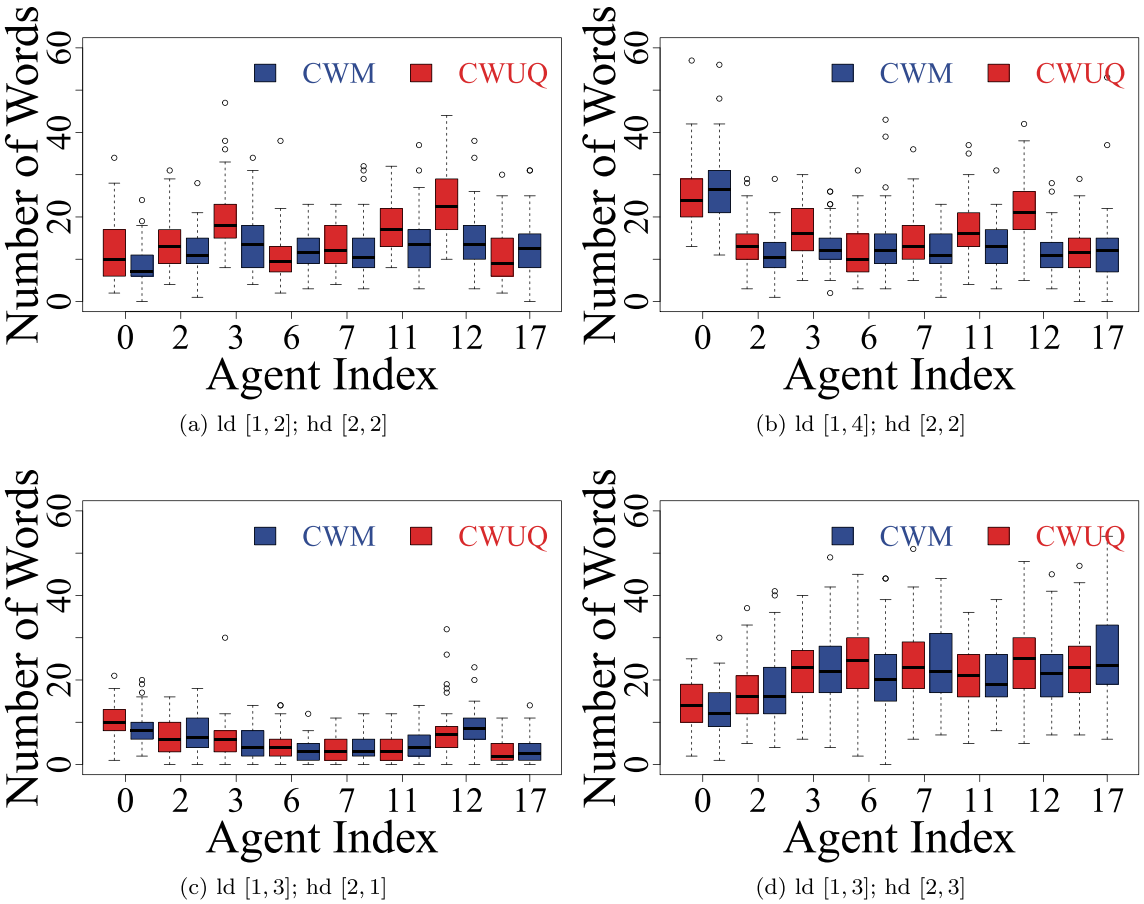
As noticed in Figure 10, the average curves are similar between the two models $\mathcal{M}_\mu$ and $\mathcal{M}_\beta$. However, there are differences: (i) the average number of words formed by each of nodes 12 and 13 are different between the two models; and (ii) the variability of results (quantified by

standard deviation) are greater for the CWUQ model $\mathcal{M}_\beta$. To the first point, in Figures 10(a,b), the average number of words formed are 14 and 24, respectively, for node 12. That is, $\mathcal{M}_\beta$ generates more words than does $\mathcal{M}_\mu$. To the second point, in comparing these same two figures, the standard deviation is much greater for the $\mathcal{M}_\beta$ model-generated results.

To examine the variability of median behaviours in numbers of words formed for nodes between the two models $\mathcal{M}_\mu$ and $\mathcal{M}_\beta$, we continue to look at the eight nodes studied in Figure 11 for the 4-clique graph. The new results that we address here are provided in Figure 12 and we focus on high degree (i.e., group $g = 2$) nodes. Each plot shows on the x-axis two values of each node ID. Red boxplots are for $\mathcal{M}_\beta$ and blue boxplots are for $\mathcal{M}_\mu$. Again, because nodes 2 through 17 all have degree $d \geq 3$ (i.e., are denoted hd for high degree nodes), they all correspond to group $g = 2$ and hence are assigned the same behaviour model. In Figures 12(a,b), the clusters assigned to high degree (hd) and low degree (ld) nodes produce behaviours such that the red bars for $\mathcal{M}_\beta$ show greater variability in median numbers of words formed across nodes 2 through 17 than does $\mathcal{M}_\mu$.

However, note that not all combinations of clusters generate more variability in median numbers of words formed in the CWUQ model. In Figure 12(c), $\mathcal{M}_\mu$ and $\mathcal{M}_\beta$ produce comparable variability in median values across nodes. In Figure 12(d), too, $\mathcal{M}_\mu$ and $\mathcal{M}_\beta$ generate similar median values that vary across nodes. The main result is that the CWUQ model ($\mathcal{M}_\beta$) produces at least as much variability as does the CWM model ($\mathcal{M}_\mu$).

Hypothesis tests are conducted to demonstrate the variability of CWM and CWUQ models (see results in Appendix C). First, we test if there is no difference between mean number of words formed by nodes with high degrees. The results in Table C1 show that in Figures 12(a,b), nodes assigned the CWM model have no difference in the mean numbers of words formed, while nodes assigned the CWUQ model have significant differences in the mean numbers of words formed. In Figures 12(c,d),



(a) ld [1, 2]; hd [2, 2]

(b) ld [1, 4]; hd [2, 2]

(c) ld [1, 3]; hd [2, 1]

(d) ld [1, 3]; hd [2, 3]

Figure 12. Simulation results of word counts for game configuration of a 4-clique graph. Results are provided for the same eight nodes (two nodes per clique) across all plots: one node in a clique that is only connected to other nodes in the clique, and the one unique node that is also connected to two other cliques. Each plot assigns to low degree (ld) and high degree (hd) nodes different clusters c as part of their behaviour models; the two clusters are fixed in each plot. The plots (a) and (b) have combinations of $[g, c]$ for ld and hd nodes that result in greater variability in median values of numbers of words formed across nodes for the CWUQ model ($\mathcal{M}_\beta$, in red) compared to that for the CWM model ($\mathcal{M}_\mu$, in blue). However, there is not always such a difference. In (c), $\mathcal{M}_\mu$ and $\mathcal{M}_\beta$ produce similar variability. Similar variabilities between models are observed in (d). In all cases, there is always at least as much variability in numbers of words formed for $\mathcal{M}_\beta$ as for $\mathcal{M}_\mu$.

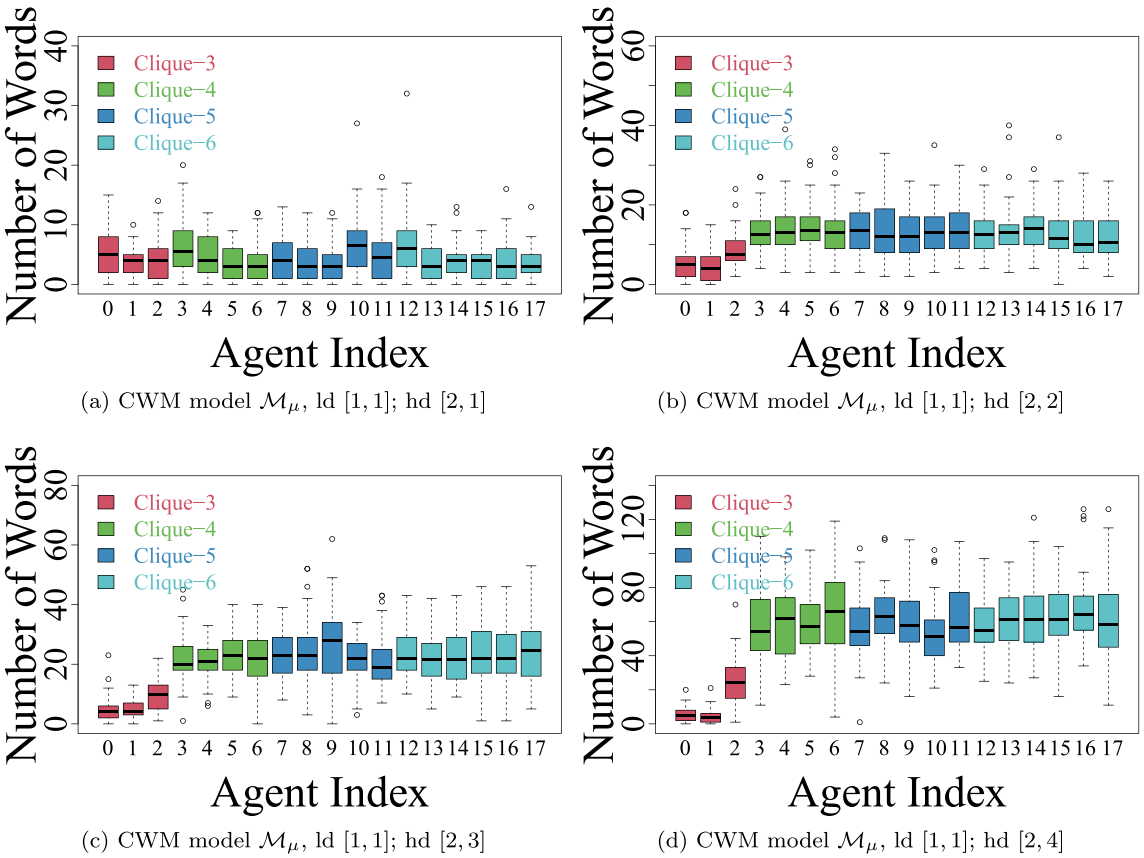
both CWM and CWUQ models show significant differences among mean numbers of words by nodes.

5.2.5. Comparisons of CWM and CWUQ model behaviours within and across cliques

Comparisons are now made between the two models, $\mathcal{M}_\mu$ and $\mathcal{M}_\beta$, for all nodes of the 4-clique network and for all four clusters of behaviour for the high degree nodes. We focus on the number of words formed by each node at the end of the game, i.e., at time $t = 300$ seconds. Data for all 50 runs of a simulation are presented as boxplots. Data for the CWM model are provided in Figure 13 and the corresponding data for the CWUQ model are given in Figure 14. Plots are for different model clusters c, and the boxplots of nodes are colored for the clique to which they belong.

Figure 13 shows four plots of number of words formed for each of the 18 nodes. The change in the plots is the cluster $c = 1, 2, 3$, and 4, assigned to the 16 high degree nodes. The cluster, of course, changes the behaviour model for the nodes, and this is observed in the median values of numbers of words for all nodes, which increase as the cluster number increases from 1 to 4. The variability for each node also increases as cluster number increases. The point of interest is the variability of these results within cliques and across cliques, viewing each plot separately. In Figure 13(a-c), it is seen that the median values do not change appreciably within cliques, nor across cliques. For cluster 4, the last plot, there is variability in median values among nodes within K_4 and within K_5, but not across cliques. Note that we do not expect massive variability in this case because all nodes of all cliques K_4, K_5, and K_6 use the same $g = 2$ models; the increased degree of nodes in K_6, compared to the degrees of nodes in K_4, for example, play no discriminating role because this model holds for all nodes with degree $d \geq 3$. Hence, the conclusion is that the CWM model does not produce great levels of variability across nodes.

Also, it is observed that the 16 nodes with $g = 2$ have increasingly different behaviour from those of nodes 0 and 1, where degree $d = 2$ and therefore $g = 1$, as cluster c increases from 1 to 4. This again



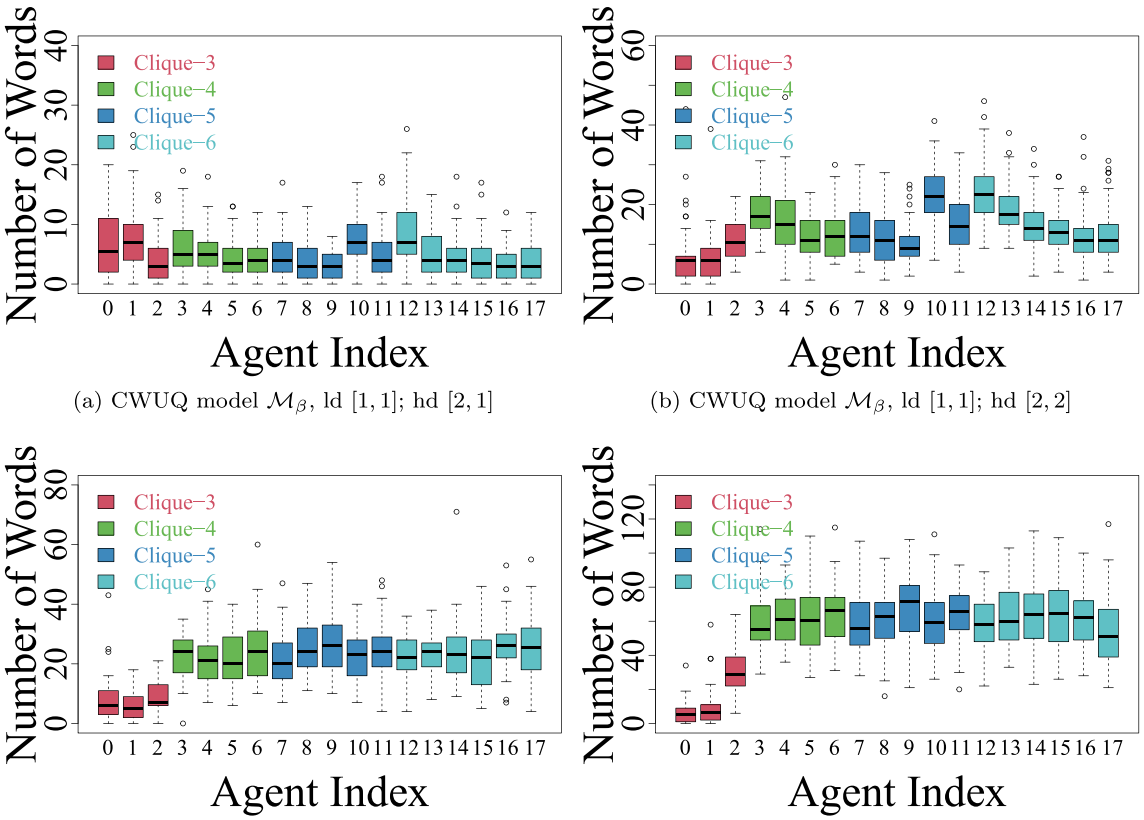
(a) CWM model $\mathcal{M}_\mu$, ld $[1, 1]$; hd $[2, 1]$

(b) CWM model $\mathcal{M}_\mu$, ld $[1, 1]$; hd $[2, 2]$

(c) CWM model $\mathcal{M}_\mu$, ld $[1, 1]$; hd $[2, 3]$

(d) CWM model $\mathcal{M}_\mu$, ld $[1, 1]$; hd $[2, 4]$

Figure 13. ABS results of final number of words formed per agent in the 4-clique network for the CWM behavioural model $\mathcal{M}_\mu = [C_\mu, g, c]$ (these are the number of words formed through the 300 second game). Each plot shows boxplots of words formed for every node in the network along the x-axis. The y-axis range varies across the plots. The difference across plots is the cluster used for the high degree (hd) node behaviour, which varies in $[g = 2, c]$: (a) $[2, 1]$; (b)$[2, 2]$; (c) $[2, 3]$; and (d) $[2, 4]$. The low degree nodes—only nodes 0 and 1—are $\mathcal{M}_\mu = [C, g, c] = [C_\mu, 1, 1]$. The boxes in each plot are color coded by the clique in which a node resides. The boxes of one color are nodes in one clique. We focus on cliques K_4, K_5, and K_6 because each agent in each clique has the same model, so we can compare the node behaviours of each clique. There is little variability in the median values, within and across cliques, for clusters $c = 1$, 2, and 3. There is greater variability within cliques for cluster 4, but there is little variability across cliques.



(a) CWUQ model $\mathcal{M}_\beta$, ld [1, 1]; hd [2, 1]

(b) CWUQ model $\mathcal{M}_\beta$, ld [1, 1]; hd [2, 2]

(c) CWUQ model $\mathcal{M}_\beta$, ld [1, 1]; hd [2, 3]

(d) CWUQ model $\mathcal{M}_\beta$, ld [1, 1]; hd [2, 4]

Figure 14. ABS results of final number of words formed per agent in the 4-clique network for the CWUQ behavioural model $\mathcal{M}_\beta = [C_\beta, g, c]$ (these are the number of words formed through the 300 second game). Each plot shows boxplots of words formed for every node in the network along the x-axis. The y-axis range varies across the plots. The difference across plots is the cluster used for the high degree (hd) node behaviour, which varies in $[g = 2, c]$: (a) [2, 1]; (b) [2, 2]; (c) [2, 3]; and (d) [2, 4]. The low degree nodes—only nodes 0 and 1—are $\mathcal{M}_\beta = [C, g, c] = [C_\beta, 1, 1]$. The boxes in each plot are color coded by the clique in which a node resides. We focus on cliques K_4, K_5, and K_6 because each agent in each clique has the same model, so we can compare the node behaviours of each clique. The greatest variation of the median values of words generated for nodes in each cluster occurs for clusters 2 and 4. There is some variability in clusters 1 and 3.

points to the efficacy of partitioning game player behaviour into different clusters, to form separate models for them – this enables greater diversity of player behaviours in simulations.

Turning now to Figure 14 for the CWUQ model $\mathcal{M}_\beta$, the same four types of plots are shown as in Figure 13. Some of the same trends hold as in the previous plot: numbers of words formed increases as c increases, the differences between the behaviours of nodes with $g = 1$ and $g = 2$ increases as c increases for the 16 $g = 2$ nodes, and the variability in results per node increases as c increases.

But there are differences between these two sets of plots. The variability in results across nodes in Figure 14(a) for $\mathcal{M}_\beta$ is 18.6% greater compared to that in Figure 13(a) for $\mathcal{M}_\mu$. Figure 14(b) shows a 16% greater variability compared to Figure 13(b). The variability difference is reduced in comparing Figures 14(c) and 13(c), but it still exists. Both Figures 14(d) and 13(d) exhibit variability across nodes in cliques. Hence, we conclude that variability in behaviours, via node-by-node results comparisons, are greater for CWUQ model $\mathcal{M}_\beta$ compared to the CWM model $\mathcal{M}_\mu$.

6. Discussion

This work presents and evaluates two methods to quantify uncertainty and build ABMs of human behaviour. Based on the data from group anagram games, the proposed methods provide a comprehensive uncertainty quantification framework for agent-based modelling and simulation. Such a frame is not limited to modelling anagram games, but can be widely applicable to other systems. Motivation, novelty, and contributions of our uncertainty quantification approach and ABMs are provided in Section 1. The methods work best when a data set can be partitioned along natural parameter dimensions as is the case in this work.

Through the comparison of model outputs via simulations of two networked GrAGs, we find that the CWUQ model can better quantify and generate uncertainty than the CWM model. Note that in

some cases, the uncertainty generated from the two models is comparable in Section 5. A possible explanation is that some results reported in Section 5 are based on averaging data over 100 simulations at each time step of the 300-second game. The overall behaviour of the two models across many simulation instances could be end up being similar, indicating that the variability may be averaged over the entire game in the CWUQ model.

There are several directions for future research. Note that the multinomial logistic regression used in this work assumes a linear parametric form, which may not be satisfied in some sophisticated social experiments. However, we can modify the parametric statistical model (i.e., multinomial logistic regression) by some nonparametric statistical model such as generalised Gaussian process. Moreover, we can explore Bayesian approaches for modelling and uncertainty quantification (van de Schoot et al., 2021) to alleviate extreme value problems caused by data scarcity in the asymptotic normal distribution. The proposed method can also be extended beyond the situation of three-letter words by adjusting the word corpus to include words of varying lengths.

Acknowledgments

We thank the editor, associate editor, and anonymous reviewers for their constructive comments for improving this manuscript. Research Computing at The University of Virginia for providing computational resources and technical support. This work has been partially supported by NSF CRISP 2.0 (CMMI Grant 1916670) and NSF CISE Expeditions (CCF-1918770).

Disclosure statement

No potential conflict of interest was reported by the author(s).

Funding

This work was supported by the National Science Foundation [CRISP 2.0 (CMMI Grant 1916670)]; National Science Foundation [CISE Expeditions (CCF-1918770)].

References

Alam, M., Deng, X., Philipson, C., Bassaganya-Riera, J., Bisset, K., Carbo, A., Eubank, S., Hontecillas, R., Hoops, S., Mei, Y., Abedi, V & Marathe, M. (2015). Sensitivity analysis of an enteric immunity simulator (ENISI)-based model of immune responses to helicobacter pylori infection. *PloS One*, *10*(9), e0136139. https://doi.org/10.1371/journal.pone.0136139

Anderson, N. H. (1961). Group performance in an anagram task. *The Journal of Social Psychology*, *55*(1), 67–75. https://doi.org/10.1080/00224545.1961.9922160

Baker, A. (2016). Simplicity. In E. N. Zalta (Ed.), *The Stanford encyclopedia of philosophy*. Metaphysics Research Lab, Stanford University. https://plato.stanford.edu/archives/win2016/entries/simplicity/

Broyden, C. G. (1967). Quasi-newton methods and their application to function minimisation. *Mathematics of Computation*, *21*(99), 368–381. https://doi.org/10.1090/S0025-5718-1967-0224273-2

Cadsby, C. B., Song, F., & Tapon, F. (2007). Sorting and incentive effects of pay for performance: An experimental investigation. *Academy of Management Journal*, *50*(2), 387–405. https://doi.org/10.5465/amj.2007.24634448

Cedeno-Mieles, V., Hu, Z., Ren, Y., Deng, X., Adiga, A., Barrett, C. L. & Self, N. (2020). Networked experiments and modeling for producing collective identity in a group of human subjects using an iterative abduction framework. *Social Network Analysis and Mining (SNAM)*, *10*(1), 43. https://doi.org/10.1007/s13278-019-0620-8

Cedeno-Mieles, V., Hu, Z., Ren, Y., Deng, X., Adiga, A., Barrett, C. & Self, N. (2019). Mechanistic and data-driven agent-based models to explain human behavior in online networked group anagram games. In *Proceedings of the International Conference on Advances in Social Network Analysis and Mining (ASONAM)*, Vancouver, Canada (pp. 357–364).

Charness, G., Cobo-Reyes, R., & Jimenez, N. (2014). Identities, selection, and contributions in a public-goods game. *Games and Economic Behavior*, *87*, 322–338. https://doi.org/10.1016/j.geb.2014.05.002

Deng, X., Lin, C. D., Liu, K.-W., & Rowe, R. (2017). Additive gaussian process for computer models with qualitative and quantitative factors. *Technometrics*, *59*(3), 283–292. https://doi.org/10.1080/00401706.2016.1211554

Deutsch, M. (1949). An experimental study of the effects of cooperation and competition upon group process. *Human Relations*, *2*(3), 199–231. https://doi.org/10.1177/001872674900200301

Fadikar, A., Higdon, D., Chen, J., Lewis, B., Venkatramanan, S., & Marathe, M. (2018). Calibrating a stochastic, agent-based model using quantile-based emulation. *SIAM/ASA Journal on Uncertainty Quantification*, *6*(4), 1685–1706. https://doi.org/10.1137/17M1161233

Fisher, R. A. (1922). On the mathematical foundations of theoretical statistics. *Philosophical Transactions of the Royal Society of London: Series A, Containing Papers of a Mathematical or Physical Character*. *222*(594–604), 309–368.

Goldman, M., Stockbauer, J. W., & McAuliffe, T. G. (1977). Intergroup and intragroup competition and cooperation. *Journal of Experimental Social Psychology*, *13*(1), 81–88. https://doi.org/10.1016/0022-1031(77)90015-4

Gramacy, R. B. (2020). *Surrogates: Gaussian process modeling, design and optimization for the applied sciences*. Chapman and Hall/CRC.

Gugole, F., Coffeng, L. E., Edeling, W., Sanderse, B., De Vlas, S. J., Crommelin, D., & Klinkenberg, D. (2021). Uncertainty quantification and sensitivity analysis of COVID-19 exit strategies in an individual-based transmission model. *PloS Computational Biology*, *17*(9), e1009355. https://doi.org/10.1371/journal.pcbi.1009355

Hartigan, J. A., & Wong, M. A. (1979). Algorithm as 136: A k-means clustering algorithm. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, *28*(1), 100–108. https://doi.org/10.2307/2346830

Henderson, D. A., Boys, R. J., Krishnan, K. J., Lawless, C., & Wilkinson, D. J. (2009). Bayesian emulation and

calibration of a stochastic computer model of mitochondrial dna deletions in substantia nigra neurons. *Journal of the American Statistical Association*, *104*(485), 76–87. https://doi.org/10.1198/jasa.2009.0005

House-Peters, L. A., & Chang, H. (2011). Urban water demand modeling: Review of concepts, methods, and organizing principles. *Water Resources Research*, *47*(5), 1–15. https://doi.org/10.1029/2010WR009624

Hu, Z., Deng, X., Goode, B. J., Ramakrishnan, N., Saraf, P., Self, N. & others. (2019). On the modeling and agent-based simulation of a cooperative group anagram game. In *2019 Winter Simulation Conference (WSC)* National Harbor, Maryland (pp. 169–180).

Hu, Z., Deng, X., & Kuhlman, C. J. (2021). An uncertainty quantification approach for agent-based modeling of human behavior in networked anagram games. In *Proceedings of the 2021 winter simulation conference*. Piscataway, New Jersey.

Jones, B., Lin, D., & Nachtsheim, C. (2008). Bayesian d-optimal supersaturated designs. *Journal of Statistical Planning and Inference*, *138*(1), 86–92. https://doi.org/10.1016/j.jspi.2007.05.021

Krige, D. G. (1951). A statistical approach to some basic mine valuation problems on the witwatersrand. *Journal of the Southern African Institute of Mining and Metallurgy*, *52*(6), 119–139.

Lee, S.-J., & Wentz, E. A. (2008). Applying bayesian maximum entropy to extrapolating local- scale water consumption in maricopa county, arizona. *Water Resources Research*, *44*(1), 1–13. https://doi.org/10.1029/2007WR006101

Li, H., Deng, X., Dolloff, C., & Smith, E. (2016). Bivariate functional data clustering: Grouping streams based on a varying coefficient model of the stream water and air temperature relationship. *Environmetrics*, *27*(1), 15–26. https://doi.org/10.1002/env.2370

Marrel, A., Iooss, B., Da Veiga, S., & Ribatet, M. (2012). Global sensitivity analysis of stochastic computer models with joint metamodels. *Statistics and Computing*, *22*(3), 833–847. https://doi.org/10.1007/s11222-011-9274-8

Mason, W., & Suri, S. (2018). On the predictive validity of various corpus-based frequency norms in L2 English lexical processing. *Behavior Research Methods*, *50*(1), 1–25. https://doi.org/10.3758/s13428-017-1001-8

Mason, W., & Watts, D. J. (2012). Collaborative learning in networks. *Proceedings of the National Academy of Sciences*, *109*(3), 764–769. https://doi.org/10.1073/pnas.1110069108

Papadelis, S., & Flamos, A. (2019). An application of calibration and uncertainty quantification techniques for agent-based models. *Understanding Risks and Uncertainties in Energy and Climate Policy*, 79. https://doi.org/10.1007/978-3-030-03152-7_3

Pearl, J., & Mackenzie, D. (2018). *The book of why: The new science of cause and effect*. Basic Books.

Ren, Y., Cedeno-Mieles, V., Hu, Z., Deng, X., Adiga, A., Barrett, C. & Macy, M. W. (2018). Generative modeling of human behavior and social interactions using abductive analysis. In *2018 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, Barcelona, Spain (pp. 413–420).

Ryzhov, I. O., Zhang, Q., & Chen, Y. (2020). Advanced statistical methods: Inference, variable selection, and experimental design. In *2020 Winter Simulation Conference (WSC)* (pp. 1–15).

Sanchez, A. (2022). Group identity and charitable contributions: Experimental evidence. *Journal of Economic Behavior and Organization*, *194*, 542–549. https://doi.org/10.1016/j.jebo.2021.12.032

Sevcikova, H., Raftery, A. E., & Waddell, P. A. (2007). Assessing uncertainty in urban simulations using bayesian melding. *Transportation Research Part B: Methodological*, *41*(6), 652–669. https://doi.org/10.1016/j.trb.2006.11.001

Sheikholeslami, R., & Razavi, S. (2017). Progressive latin hypercube sampling: An efficient approach for robust sampling-based analysis of environmental models. *Environmental Modelling & Software*, *93*, 109–126. https://doi.org/10.1016/j.envsoft.2017.03.010

Sweeting, T. J. (1980). Uniform asymptotic normality of the maximum likelihood estimator. *The Annals of Statistics*, *8*(6), 1375–1381. https://doi.org/10.1214/aos/1176345208

van de Schoot, R., Depaoli, S., King, R., Kramer, B., Märtens, K., Tadesse, M. G., & others. (2021). Bayesian statistics and modelling. *Nature Reviews Methods Primers*, *1*(1), 1–26. https://doi.org/10.1038/s43586-020-00001-2

Watson, G. B. (1928). Do groups think more effectively than individuals? *Journal of Abnormal and Social Psychology*, *23*(3), 328–336. https://doi.org/10.1037/h0072661

Wu, C. J., & Hamada, M. S. (2011). *Experiments: Planning, analysis, and optimization* (Vol. 552). John Wiley & Sons.