Proceedings of the 2023 Winter Simulation Conference
C. G. Corlu, S. R. Hunter, H. Lam, B. S. Onggo, J. Shortle, and B. Biller, eds.

A CALIBRATION MODEL FOR BOT-LIKE BEHAVIORS IN AGENT-BASED ANAGRAM GAME SIMULATION

Xueying Liu
Zhihao Hu
Xinwei Deng

Department of Statistics
Virginia Tech
800 Washington Street Southwest
Blacksburg, VA 24061, USA

Chris J. Kuhlman

Biocomplexity Institute & Initiative
University of Virginia
1827 University Avenue
Charlottesville, VA 22904, USA

ABSTRACT

Experiments that are games played among a network of players are widely used to study human behavior. Furthermore, bots or intelligent systems can be used in these games to produce contexts that elicit particular types of human responses. Bot behaviors could be specified solely based on experimental data. In this work, we take a different perspective, called the Probability Calibration (PC) approach, to simulate networked group anagram games with certain players having bot-like behaviors. The proposed method starts with data-driven models and calibrates in principled ways the parameters that alter player behaviors. It can alter the performance of each type of agent (e.g., bot) in group anagram games. Further, statistical methods are used to test whether the PC models produce results that are statistically different from those of the original models. Case studies demonstrate the merits of the proposed method.

1 INTRODUCTION

1.1 Background

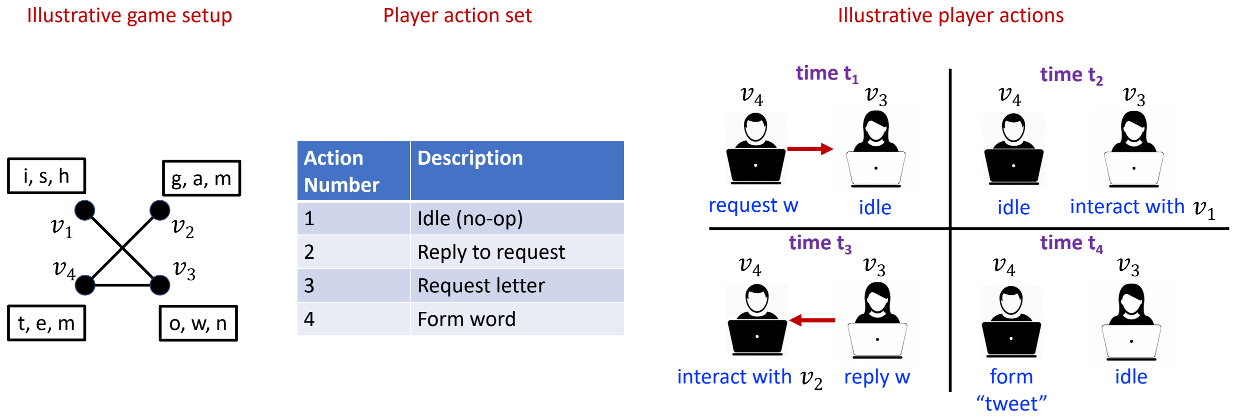
Networked games—games or experiments that assign human subjects as nodes in graphs and interaction channels between pairs of humans as edges—are used in many different contexts. Economics uses these types of experiments for many purposes, e.g., coordination (Kearns et al. 2012), bargaining (Chakraborty et al. 2010), and decisions under conditions of incomplete information (Charness et al. 2014). Explore-exploit problems, e.g., Mason and Watts (2012), are of interest in anthropology and evolutionary studies (Gopnik 2020), cognitive science (Feng et al. 2021), and business (den Hamer and Frenken 2021). Social scientists conduct networked experiments to study common knowledge (Korkmaz et al. 2018), collective identity (Charness et al. 2014), and collective action (Centola 2010; Mønsted et al. 2017). They have also been used to study anagram games (Charness et al. 2014).

In this work, we focus on a networked anagram game detailed in Cedeno-Mieles et al. (2020) and overviewed in the next subsection. Existing work has used data from over 200 experiments to build statistical models of game player behavior, e.g., Liu et al. (2022). Our scope here is to produce behaviors for players that are not observed in the experimental data, especially when the players are assisted by intelligent systems such as ChatGPT. This enables us to incorporate such a calibration model into agent-based simulations (ABSs) and gain a deeper understanding of the dynamics of player behavior in networked group anagram games (NGrAGs) when intelligent systems are involved in the game.

979-8-3503-6966-3/23/$31.00 ©2023 IEEE

1.2 Overview of Networked Group Anagram Game

Figure 1 conveys many of the ideas of NGrAGs that were conducted online. Games or experiments were performed with four to fifteen players per game. Each game lasts for $t_g = 300$ seconds. A network $G(V,E)$ is specified on the game players, where V is the set of players and E is the set of communication channels between pairs of players. A network is provided at the left of Figure 1. The game platform takes all players through game instructions and example actions. Players are told the true goal of the game: to have the *team* form as many words as possible, that all players get the same remuneration, and that team earnings are based on the number of words that the team forms. This is to foster cooperation. Initially, each player is given three letters, assigned at random, but with an eye toward providing letters that appear most often in words, e.g., not using letters like x and z. Each player sees their own assigned letters and those of its neighbors. Once the game timer begins, players can, at any time in the game, think about what they want to do (action idle a_1), reply to a letter request from a neighbor (action a_2), request a letter from a neighbor (action a_3), or form a word (action a_4). Examples are provided on the right of Figure 1 using two of the four players in the network. At time t_1, player v_4 requests the letter w from player v_3. At a later time t_3, v_3 responds to v_4 with the requested letter w while v_4 is interacting (e.g., requesting a letter from) v_2. In experiments, time is continuous, so that actions by multiple players typically do not happen at the same instant, but they can. These examples are to illustrate that pairs of agents do not have to interact with each other in any given instant, e.g., v_i can interact with v_j, who in turn, interacts with v_k. The actions in the center table can be repeated by all players any number of times, as they desire. Note, for example, that a player does not have to request neighbor letters and does not have to reply to neighbor letter requests. Each player's game screen shows the letters it has to form words, its letter requests, the collection of outstanding letter requests made to this player, and the words it has formed up through the current time. See Cedeno-Mieles et al. (2020) for further details.



Action Number	Description
1	Idle (no-op)
2	Reply to request
3	Request letter
4	Form word

Figure 1: Illustrative networked group anagram game (NGrAG). The network $G(V,E)$ of four players is on the left, and sets L_j^{init} of $n_\ell = 3$ initial letters are assigned to each $v_j \in V$, $j \in \{1,2,3,4\}$. All player action types are given in the center. The quad chart on the right shows representative interactions between players v_3 and v_4 in carrying out the various actions of a game at time steps $0 < t_1 < t_2 < t_3 < t_4 \leq t_g$. Players can use a letter multiple times in the same word, e.g., using t and e twice each in *tweet*. Each player may repeat these actions, in any order, throughout the $t_g = 300$ seconds game duration.

1.3 Motivation

In this work, we construct new models for characterizing players' behaviors to generate behaviors not observed in networked anagram experiments. The proposed model is called the *probability calibration*

(PC) model. This model forms the basis of a new agent-based model (ABM) for simulation of the NGrAG. The proposed agent-based simulation (ABS) is named the PC-based ABS approach.

The PC-based ABS can be used for bots or simulated agents in experiments. Bots can be used to create particular environments for a player test subject that may otherwise be difficult and/or time consuming to create with only human subjects (e.g., to find and coach human confederates to play in particular ways). For example, we can construct bot agents that do not reply to letter requests, thus destroying productive player interactions that can give rise to collective identity (Polletta and Jasper 2001). In a similar way, we can test other theories like tit-for-tat (i.e., a person only responds to a neighbor to the degree that the neighbor responds to her) (Kollock 1998). We can do this by having bots behave in extreme ways (e.g., replying very quickly to letter requests or making many letter requests) and determining whether human neighbors respond in kind. As another case, in single person anagram games, experiments have been performed where a player is given poor letters and good letter in different experiments, leading to poor and good performance, respectively. Interestingly, players are dichotomous in their explanations of their behaviors: they attribute good performance to skill but poor performance to bad luck (Feather and Simon 1971). In networked versions of these experiments, we could use bots to exhibit various levels of cooperation and determine whether players blame their neighbors for poor performance or complement them for good performance (Monroe and Malle 2017), in addition to skill and luck attributions.

Given this motivation, it is essential to investigate the significance of differences generated by the PC model when compared to results from the original models. In other words, when using a new model to generate player behaviors in the ABS, it is important to quantify whether these differences are significant or merely noise.

1.4 Novelty and Contributions

The novelty of this work is to propose a probability calibration model to characterize a player's behavior beyond the model developed from the experimental data. Such a modeling strategy enables ABS of NGrAGs to be much more flexible and enables surgical use of human subjects and intelligent systems in playing the game. We have also adopted the functional analysis of variance (FANOVA) to analyze the significance of these new models in terms of player performance in the agent-based simulation. Previous modeling efforts of NGrAGs are confined to building models from data and assessing uncertainty in the models and simulation results, e.g., Liu et al. (2022). In contrast, the proposed model is developed to accommodate varying degrees of player utilization of intelligent systems, and investigate the consequent effects on players' behaviors for the NGrAGs. This has not been done before for NGrAGs. The calibration concepts and techniques developed in this work can also be applied to more sophisticated anagram game models (Cedeno-Mieles et al. 2019) and to models of other games (Mason and Watts 2012).

Our first contribution is a principled methodology to enable a proper calibration of data-driven models to behaviors beyond those gathered from experimental data. Our use of the word "calibration" comes from the fact that we are generating new models by quantifying via $\boldsymbol{\alpha}$ a deviation or departure from the behavior of data-driven models, so that there is always a reference model (i.e., the data-driven model) for each new PC model. By introducing the calibration parameters $\boldsymbol{\alpha}$ for the four player actions, the model can change player probabilities of taking each action at each time step. It is in this sense that we use the word "bot" in this paper, i.e., an agent whose behavior is exogenously controlled by game administrators (through $\boldsymbol{\alpha}$). Hence, the core or baseline probabilities of actions are preserved, but are scaled according to the $\boldsymbol{\alpha}$ values. By appropriately choosing calibration parameters, one can simulate different scenarios to examine how the performance of intelligent agent players adapt in the NGrAGs.

Our second contribution is an evaluation methodology to determine whether two behaviors are the same or different. We adopt the use of the functional ANOVA method to compare the players' behaviors under different deviations from the reference model. It provides insights on the degrees of deviation from the reference model using the calibration parameters that lead to a significant difference in players' behaviors in the NGrAGs. Such insights and understanding can help in developing new monitoring and mitigation methods for adversary attacks in online social games.

Our third contribution is a simulation-based case study that employs both the reference models and the proposed PC models (that are recast in software as ABMs) in the ABS of the NGrAG. In this case study, the impact that different (bot) behaviors have on the actions of the agents are assessed. We specify constant values for the $\boldsymbol{\alpha}$ parameters as a first step. Using different settings for calibration parameters, the simulation results uncover several interesting findings on the dynamics of players' behaviors. For example, we demonstrate that decreasing the probability of the idle action leads to more words formed by agents than increasing the probability of the forming word action. Such findings can imply the promotion of collective identity, which encourages more interaction among players. Note that although we consider relatively simple scenarios of specifying the calibration parameters $\boldsymbol{\alpha}$, the setting of $\boldsymbol{\alpha}$ parameters in the PC model can be a function of time, previous player actions, and other game variables.

The remainder of the paper is organized as follows. Section 2 contains related work. Section 3 presents the PC model. Section 4 describes the simulation process, which includes the PC model, and simulation results from the case study. Conclusions and future work are in Section 5.

2 RELATED WORK

Networked experiments and modeling. Complex contagion (Centola and Macy 2007) was studied in group online networked experiments (Centola 2010). A model of common knowledge (Korkmaz et al. 2014) was used to specify online networked experiments of collective action. Experiments were used to enhance the model (Korkmaz et al. 2018). Explore-exploit network experiments were run, with a small amount of modeling, in Mason and Watts (2012). Many economics-based games were run and modeled on networks. A series of networked games, involving strategic complements and strategic substitutes, with complete and incomplete information, were performed on networks and modeled (Charness et al. 2014). Network formation and coordination games were run and modeled in Corbae and Duffy (2008). The experiments on which our modeling herein is based is given in Cedeno-Mieles et al. (2020); illustrative modeling works to characterize uncertainty include Liu et al. (2022). The modeling work in this paper is different: here we seek to extend the models of player behavior for use as bots and to explore behaviors that are reasonable but have not been gathered through experiments, owing to experimental constraints.

Experiments and modeling with bots. Bots have been used to establish broader consensus through coordination in networks of humans, above the consensus established with humans alone (Shirado and Christakis 2017). They have also been used in experiments on Twitter to determine whether social contagions are spread by simple or complex mechanisms (Mønsted et al. 2017). Much work exists on detecting bots, e.g., Mendoza et al. (2020).

Sequence analysis of game data. Game data can be viewed as a collection of action sequences or time series data, representing the progression of in-game events over time. Numerous studies have utilized the activity sequence in a game session to capture player behavior in massively-multiplayer online games (MMOGs). These studies have employed various methods, such as machine learning binary classification techniques (Ahmad et al. 2009), Levenshtein distance (Platzer 2011), time series classification (Bernardi et al. 2017), and more, to detect bot players.

3 TRANSITION PROBABILITY CALIBRATION MODEL

In our early works, e.g., Cedeno-Mieles et al. (2020), Liu et al. (2022), a clustering-based method was developed to quantify players' behaviors in NGrAGs. These methods provide a way to quantify the heterogeneous behavior of players in the experimental data and incorporate the quantified uncertainty into ABSs. Incorporation of uncertainties into ABS enables a deeper understanding of the dynamics of human behavior in NGrAGs and bridges the gap between experimental data and real-world applications. However, these previous methods only produce behaviors that are represented in the experimental data.

This section describes a transition probability calibration model that produces a wider range of player behaviors, beyond the experimental data. These may be useful to represent bots, or humans that are aided by intelligent systems. It can further provide insights into the larger field of human-computer interaction and collaboration.

3.1 Previous Models

The previous model provides a generative approach to modeling the group anagram game as a discrete-time stochastic process (Cedeno-Mieles et al. 2020). That work also includes model validation, using Kullback-Leibler (KL) divergence to compare game data and model output distributions. In the model, at each time step, players choose one of four pre-defined actions: idling (a_1), replying to a neighbor's letter request (a_2), requesting a letter from a neighboring player (a_3), or forming a word (a_4). Instead of solely relying on the current action of a player, it is essential to consider the influence of her current state on her decision-making process. This includes several factors at time t: the size of the buffer of letter requests that a player has yet to reply to, denoted as $z_B(t)$; the number of letters the player has, denoted as $z_L(t)$; the number of valid words the player has formed, denoted as $z_W(t)$; and the number of consecutive time steps that the player has taken the same action, denoted as $z_C(t)$. Therefore, we introduce these four variables in our model to accurately capture the evolving nature of the NGrAG and provide a more realistic representation of player behavior and decision-making dynamics. These variables are combined into a vector $\mathbf{z(t)} = (1, z_L(t), z_W(t), z_B(t), z_C(t))^T_{5\times 1}$ that evolves over time.

At time $(t+1)$, given a player's most recent action $a_i(t)$, we use multinomial logistic regression to model the probabilities of their next action $a_j(t+1)$ as $\pi_{ij}(t+1)$. The formula is as follows, where idle (a_1) is chosen as the reference level for comparison with other actions (a_2, a_3, and a_4):

$$\begin{aligned} log(\frac{\pi_{ij}(t+1)}{\pi_{i1}(t+1)}) &= \beta_{j0}^{(i)} + \beta_{j1}^{(i)} z_L(t) + \beta_{j2}^{(i)} z_W(t) + \beta_{j3}^{(i)} z_B(t) + \beta_{j4}^{(i)} z_C(t) \\ &= \mathbf{z(t)^T} \boldsymbol{\beta}_{\mathbf{j}}^{\mathbf{(i)}},\ j = 2,3,4, \end{aligned} \tag{1}$$

where $\boldsymbol{\beta}_j^{(i)} = (\beta_{j0}^{(i)}, \beta_{j1}^{(i)}, \beta_{j2}^{(i)}, \beta_{j3}^{(i)}, \beta_{j4}^{(i)})^T_{5\times 1}$. The parameters are written as a matrix $\boldsymbol{B}^{(i)} = (\boldsymbol{\beta}_2^{(i)}, \boldsymbol{\beta}_3^{(i)}, \boldsymbol{\beta}_4^{(i)})^T_{3\times 5}$, and the probability of a player taking each action a_j at time $t+1$, given their current action a_i at time t, is represented by the vector $\boldsymbol{\pi}_i(t+1) = (\pi_{i1}(t+1), \pi_{i2}(t+1), \pi_{i3}(t+1), \pi_{i4}(t+1))^T$.

Furthermore, Liu et al. (2022) accounted for the observed heterogeneity in player behavior and activity levels by developing an uncertainty quantification framework to capture this variation in the observed data. Specifically, the players were first partitioned into two groups based on their number of neighbors (i.e., degree of the node in the network structure), and then clustering methods were used within each group to further divide the players into four distinct clusters. Players with a higher cluster number within a group were found to have higher activity levels and better ability to form words. Then, Equation (1) is estimated for each cluster within each group, resulting in a separate set of parameters for each cluster. This allows us to model and simulate the group anagram game in a more realistic manner, taking into consideration the diverse behaviors exhibited by different players.

3.2 Proposed Probability Calibration Model and Its Inference

Building on the previous model in Section 3.1, we can investigate individual and collective behaviors among players in the NGrAG utilizing experimental data as a foundation. Note that the original experiments were carried out with remote participants from Amazon Mechanical Turk, operating under specific settings and a limited number of players. However, as intelligent systems continue to advance, there is a growing interest in understanding the characteristics of NGrAGs when the players are aided by intelligent systems, or when the intelligent system acts as a remote player.

To enable ABMs to simulate NGrAGs with possible intelligent systems involved in playing the game, we propose a so-called probability calibration (PC) model to adjust the transition probability of each action

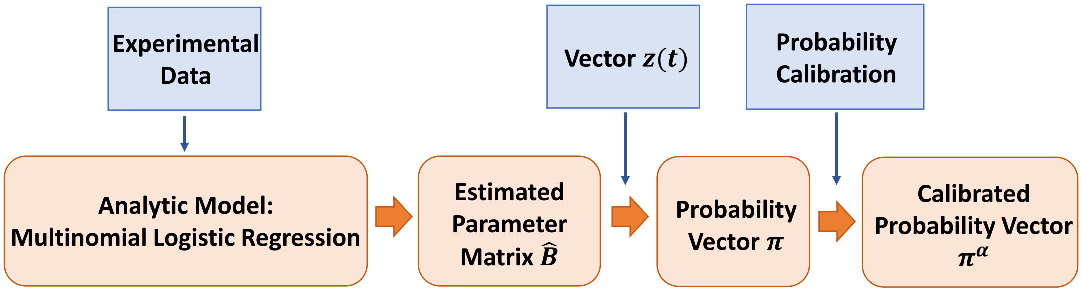

Experimental Data
Vector $z(t)$
Probability Calibration
Analytic Model: Multinomial Logistic Regression
Estimated Parameter Matrix $\hat{B}$
Probability Vector π
Calibrated Probability Vector π^{α}

Figure 2: Flowchart for the probability calibration (PC) model to adjust the probability of each action, using $\boldsymbol{\alpha}$, at every time step.

at every time step. The proposed method differs from pre-specifying the transition probability for the players with intelligent systems. Instead, the key idea of the proposed PC model is to first unitize an analytic model to obtain the transition probability based on the experiment data and the player's historical behaviors, and then conduct a probability calibration to reflect the changes due to intelligent systems as shown in Figure 2. Therefore, the proposed method offers flexibility in accommodating varying degrees of player utilization of intelligent systems. When intelligent systems are involved in a player's decision-making process, calibration parameters can be adjusted. In particular, the previous model in Section 3.1 becomes a special case of the proposed method in the sense that there is no calibration used. Specifically, we define the calibration parameters $\boldsymbol{\alpha}^{(i)} = (\alpha_1^{(i)}, \alpha_2^{(i)}, \alpha_3^{(i)}, \alpha_4^{(i)}),\ i = 1,2,3,4$, where i denotes the current action $a_i(t)$. With the transition probability vector $\boldsymbol{\pi}_i(t+1)$ obtained using the analytic model in Equation (1), the proposed PC model produces a calibrated transition probability vector as

$$\begin{aligned}\boldsymbol{\pi}_i^{\alpha}(t+1) &= (1+\boldsymbol{\alpha}^{(i)})\cdot\boldsymbol{\pi}_i(t+1)\\ &= \frac{1}{C}\Big((1+\alpha_1^{(i)})\pi_{i1}(t+1), (1+\alpha_2^{(i)})\pi_{i2}(t+1), (1+\alpha_3^{(i)})\pi_{i3}(t+1), (1+\alpha_4^{(i)})\pi_{i4}(t+1)\Big)^T,\end{aligned}$$

where the constant $C = \sum_{j=1}^{4}(1+\alpha_j^{(i)})\pi_{ij}(t+1)$ is to make the probabilities of the four actions sum to 1. Then, $\boldsymbol{\pi}_i^{\alpha}(t+1)$ is used to determine the next action $a_j(t+1)$. Note that the values of $\alpha_j^{(i)}$ can be either positive or negative, reflecting an increase or decrease in the probabilities of particular actions, respectively. Moreover, the $\boldsymbol{\alpha}^{(i)}$ vector allows the assignment of different transition probability vectors to individual players. This enables a more flexible and customizable approach to modeling player behaviors in the game.

Using the proposed PC model, we can simulate NGrAGs where some of the players are assisted by the intelligent agents. To analyze the effect of incorporating the PC model into the ABS, we consider the action sequence of a player in the game as a function of time. In this perspective, we can adopt the functional analysis of variance (FANOVA) (Fan and Lin 1998; Cuevas et al. 2004; Shen and Faraway 2004; Zhang 2011; Górecki and Smaga 2015) to investigate differences in the mean functions across different conditions. For FANOVA, the test statistic is the ratio of between-group variance to within-group variance. A large test statistic value implies that between-group variability is considerably greater than within-group variability, indicating a significant difference between the mean functions of the two groups. Specifically, for $t \in T = \{1,\ldots,300\}$, we define the means of the functions $X_{aj}(t),\ j = 1,\ldots,n_a$ and $X_{bj}(t),\ j = 1,\ldots,n_b$ as $\mu_a(t)$ and $\mu_b(t)$, respectively. Here, n_a and n_b represent the number of simulation iterations under conditions a and b, respectively. For example, $X_{aj}(t)$ can be the number of formed words up to time t for a player without the assistance of the intelligent system, while the $X_{bj}(t)$ can be the number of formed words up to time t for a player with the assistance of the intelligent system. The FANOVA aims to test the null hypothesis H_0 that there is no significant difference between the mean functions under condition a

and condition b:

$$H_0 : \mu_a(t) = \mu_b(t), t \in T$$
$$H_1 : \mu_a(t) \neq \mu_b(t), t \in T.$$

In this work, we adopt the FANOVA test to compare the behavior of players with different calibration parameters $\boldsymbol{\alpha}^{(i)}$. By calculating the p-value and comparing it with the significance level ($\alpha = 0.05$), we determine if there is a significant difference between the mean functions of the two conditions. A p-value less than or equal to 0.05 indicates a significant difference, while a p-value greater than 0.05 means there is no significant difference. Such inferences using hypothesis testing can enhance the understanding of the PC model in NGrAGs and provide insights into the effect of intelligent agents for the networked players.

Here we would like to emphasize that the proposed PC model allows for greater control over the decision-making process, enabling the comparison of various behaviors between players and intelligent agents in NGrAGs. By fine-tuning the calibration parameters, we can simulate different scenarios and analyze how artificial intelligence adapts and optimizes its performance. Ultimately, the proposed model can make a contribution to the development of more advanced intelligent systems that can adapt to changing environments, and can also enhance our understanding of the interaction between a human and AI in complex tasks.

4 AGENT-BASED SIMULATION AND EVALUATION

4.1 Simulation Scenarios and Process

In this section, simulation results are provided for a NGrAG with eight players. Seven cases are studied to compare original data-driven model results and analogous results generated with the PC model (e.g., using bots): (1) a baseline case where all $\alpha_j = 0$ (i.e., no effect of $\boldsymbol{\alpha}$), (2) agents increase their probabilities of forming words (three cases), and (3) agents reduce their probabilities of being idle (three cases). Specifically, we have:

(1) $\alpha_1^{(i)} = \alpha_2^{(i)} = \alpha_3^{(i)} = \alpha_4^{(i)} = 0$, $i = 1$, 2, 3, and 4.
(2) $\alpha_1^{(i)} = \alpha_2^{(i)} = \alpha_3^{(i)} = 0, \alpha_4^{(i)} = c_1$, where $c_1 = 0.1$, 0.2, and 0.5, $i = 1$, 2, 3, and 4.
(3) $\alpha_1^{(i)} = c_2, \alpha_2^{(i)} = \alpha_3^{(i)} = \alpha_4^{(i)} = 0$, where $c_2 = -0.05$, -0.1, and -0.5, $i = 1$, 2, 3, and 4.

For each simulation, these respective assignments of α values are made to all agents so that α values for one simulation are homogeneous.

A *simulation* is a collection of 100 simulation *instances* or *runs*. Each instance models the NGrAG as described in Section 1.2. We now provide details of simulation inputs for one instance. A graph $G(V,E)$ is specified for the game and this fixes the agent interactions (agents interact with their distance-1 neighbors in G). It also fixes the group g to which each player is assigned. Initial conditions for each agent are specified: behavior parameters (the cluster c and $\boldsymbol{\alpha}^{(i)}$ for $i \in \{1,2,3,4\}$) and the initial letters L_k^{init} assigned to each player $v_k \in V$. There are $|L_k^{init}| = 4$ letters assigned to each player in a game. For space reasons, we focus on $[g,c] = [1,3]$ for low-degree nodes (i.e., agents v_k with $d_k \leq 2$), and $[2,3]$ for high-degree nodes (i.e., agents v_k with $d_k \geq 3$), and on the Frequentist model (McCullagh 2019). The $[g,c]$ pair fixes the $\boldsymbol{\beta}_j^{(i)}$ vectors used in Equation (1). A single 6000-word corpus C^W is assigned to all players from which players choose a word to form when the action is a_4. The duration of a simulated instance is $t_g = 300$ seconds, consistent with experiments. The $n_{runs} = 100$ instances per simulation all use the same initial conditions, so that run-to-run differences in results are due solely to the stochasticity of a simulation (e.g., the behavior model). The game network $G(V,E)$ used in all simulations is provided in Figure 3.

Our simulation process is a discrete time process, which is justified by the fact that in over 200 games (Cedeno-Mieles et al. 2020), players did not take successive actions within one-second intervals. A one-second discrete time increment is used for each time step in a simulation instance.

The simulation process is as follows for one run; this process is repeated n_{runs} times. At time $t = 0$ initial conditions are assigned to nodes (agents) of the graph. It is assumed that at $t = 0$ all previous actions are idle a_1. For each time $0 \leq t < t_g$, the following computations are performed to determine the next action at time $(t+1)$, $a_j(t+1)$. Each player $v_k \in V$ receives all letter requests from its neighbors that were made at time t and all letter replies to v_k at time t from earlier letter requests by v_k. Then, based on v_k's most recent action $a_i(t)$, the base probability for each of the four next possible actions at time $t+1$ is computed from the Frequentist model: $\boldsymbol{\pi}_i(t+1) = (\pi_{i1}(t+1), \pi_{i2}(t+1), \pi_{i3}(t+1), \pi_{i4}(t+1))^T$. These probabilities are then scaled by the appropriate calibration parameters $\boldsymbol{\alpha}^{(i)}$ and renormalized so that the probabilities for the four actions sum to 1. Based on these calibrated probabilities, we use a multinomial distribution to randomly select the next action $a_j(t+1)$ of v_k. If the selected action cannot be executed (e.g., form word a_4 is the next action but an agent cannot form a word with its letters, or request letter is the next action but all neighbor letters have already been requested), then the action is changed to idle a_1. When the computations at $t = (t_g - 1)$ complete, all $a_j(t_g)$ are determined for all $v_k \in V$, the current simulation instance is complete and all agent parameters are reset to the initial conditions for the start of the next instance.

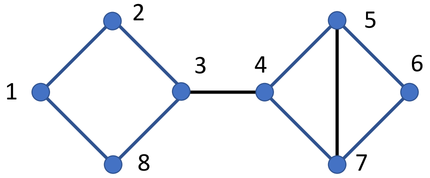


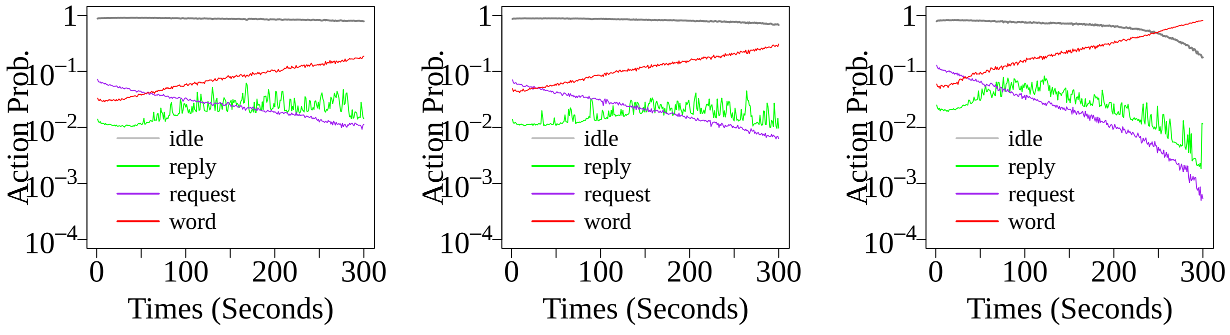
Figure 3: Anagram game network $G(V,E)$ with $|V| = 8$ and $|E| = 10$, on which simulations are run. Nodes v_k with degree $d_k \leq 2$ are in group $g = 1$ degree $d_k \geq 3$ are in group $g = 2$.

4.2 Simulation Results

The time histories of probabilities for node 3 of Figure 3 for different simulations that use different $\boldsymbol{\alpha}$ vectors are shown in Figure 4. The four curves in each plot correspond to the four actions (idle a_1, replying to letter requests a_2, requesting letters a_3, and forming words a_4; see legends), which represent the time point-wise averages of 100 simulation instances. In Figure 4a, the probabilities for the original method (no calibration) are shown. Here, the probability of idling (thinking) is about 0.9, with the remaining probability distributed among the three actions involving letters and words. This observation is reasonable since the actions are computed per second, and experimental data show that players spend most of their time deciding what to do next.

To simulate the actions of intelligent agents, such as bots, based on observed experimental data from human subjects, we make adjustments to represent faster decision-making processes of bots. Figure 4b illustrates the outcome of applying the PC approach with $\alpha_4 = 0.5$ ($\boldsymbol{\alpha} = (\alpha_1 = 0, \alpha_2 = 0, \alpha_3 = 0, \alpha_4 = 0.5)$), which means increasing the probability of forming words by 50% and adjusting the probabilities of the four actions to ensure that they add up to 1.0 at each time point. The plot indicates that the probabilities of replying and requesting are comparable to the original method, while the probability of forming a word is more than 1.5 times greater than the original method, and the probability of being idle decreases as time progresses.

Results of decreasing the probability of idling by 50% (PC approach with $\alpha_1 = -0.5, \alpha_2 = \alpha_3 = \alpha_4 = 0$) are shown in Figure 4c. This $\boldsymbol{\alpha}$ setting increases the probabilities of all three player actions involving letters and words. The probability of idling starts at 0.8 and decreases to 0.17 by the end of the game, while the probability of forming words increases over time from 0.06 to 0.6. It is interesting that early in the simulation, π_2, π_3, and π_4 are all greater than in the base case of Figure 4a, consistent with the prescribed $\boldsymbol{\alpha}$. But as time increases, π_2 and π_3 both decrease to lesser values than in the other two plots,



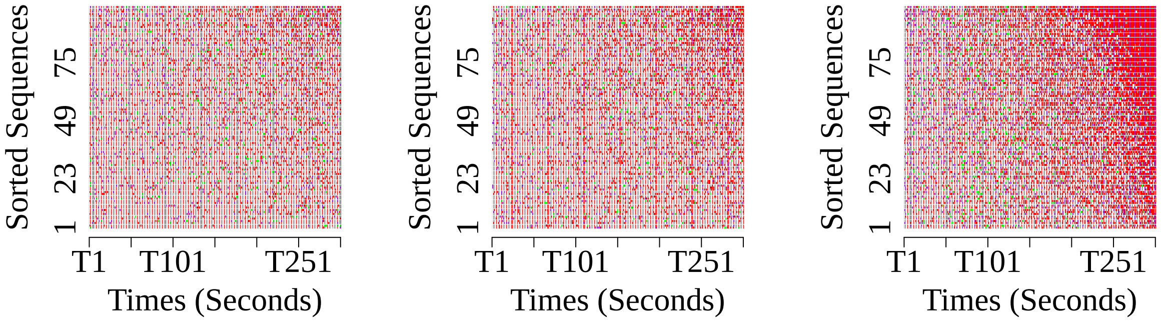
(a) node 3, original method

(b) node 3, $\alpha_4 = 0.5$

(c) node 3, $\alpha_1 = -0.5$

Figure 4: The time histories of probabilities for node 3 in the simulation setting that low-degree (ld) nodes have $[g,c] = [1,3]$ and high-degree (hd) nodes have $[g,c] = [2,3]$. (a) Probabilities for the original method (i.e., all $\alpha_j = 0$). (b) Probabilities for PC approach where $\alpha_4 = 0.5$, other $\alpha_j = 0$. (c) Probabilities for PC approach where $\alpha_1 = -0.5$, other $\alpha_j = 0$. Each curve in (a) to (c) represents the probability of an agent taking a specific action. The displayed data are the average probabilities over 100 instances at each time point.

and π_4 increases to greater values. This is due to $\pi_4 > \pi_2$, π_3, so that forming words gets amplified more by $\boldsymbol{\alpha}$, and due to the evolution of $\boldsymbol{z}$.



(a) node 3, original method

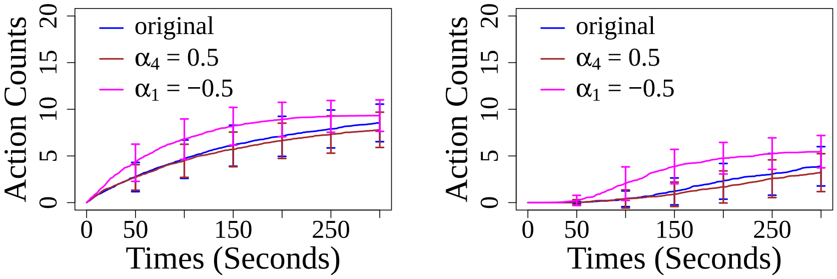
(b) node 3, $\alpha_4 = 0.5$

(c) node 3, $\alpha_1 = -0.5$

Figure 5: The time histories of action sequences for node 3. (a) Action sequences for the original method (i.e., all $\alpha_j = 0$). (b) Action sequences for PC approach with $\alpha_4 = 0.5$, other $\alpha_j = 0$. (c) Action sequences for PC approach with $\alpha_1 = -0.5$, other $\alpha_j = 0$. Each iteration is represented by horizontally stacked boxes, which are colored based on the action taken at the time, and the 100 sorted iterations are arranged vertically. The sequences are sorted according to the scores of a multidimensional scaling analysis of the dissimilarities between sequences. The colors of the actions used in this figure are consistent with those used in Figure 4.

Figure 5 displays the action sequence plot of 100 iterations from the same three simulations for node 3. Each iteration is represented by horizontally stacked boxes, which are colored based on the action taken at the time, and the 100 sorted iterations are arranged vertically. The colors of the actions used in this figure are the same as those used in Figure 4. The sequences are sorted according to the scores of a multidimensional scaling analysis of the dissimilarities between sequences. This plot allows for clearer visualization of the action transitions and duration spent in each action throughout the simulation. There is progressively more red, for forming words, moving from the left-most plot to the right-most plot. They also show the increased words formed in Figure 5c, particularly in the later stage of the game.

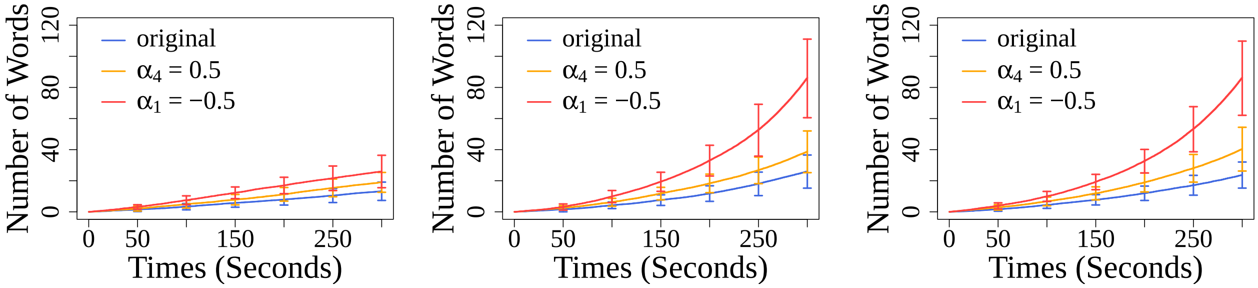
It is important to recognize that each probability is dependent on players' current state vectors $\mathbf{z}(\mathbf{t})$ and the $\boldsymbol{B}^{(i)}$ matrices, as described in Equation (1). By decreasing the probability of idling, the PC approach encourages players to be more active and responsive, resulting in more requests and replies sent by players as shown for node 3 in Figure 6. Additionally, the increase in activity level also contributes to a more significant increase in the probability of forming words compared to merely increasing the probability of forming words directly (using the PC approach with $\alpha_4 = 0.5$). It is worth noting that the case $\alpha_4 = 0.5$ produces little change in action counts for requests sent and replies sent, as expected. Moreover, unlike forming words, the numbers of possible requests and replies are bounded by a player's number of neighbors and number of letters per neighbor.



(a) node 3, request sent (b) node 3, reply sent

Figure 6: (a) and (b) show the average number of requests sent and replies sent by node 3, respectively. The curves are the time point-wise averages over the 100 instances, with error bars for ± one standard deviation. The three scenarios in the legend are original (i.e., all $\alpha_j = 0$); $\alpha_4 = 0.5$, other $\alpha_j = 0$; and $\alpha_1 = -0.5$, other $\alpha_j = 0$.

Figure 7 provides the time histories of the number of words formed for selected nodes. The curves are the time point-wise averages over the 100 instances, with error bars for ± one standard deviation. The figure shows that the number of words formed at the end of the NGrAG increases from node 1 to 4, as the latter two nodes have $d = 3$ and node 4 is connected to more high-degree nodes, per Figure 3. The chosen nodes can form more than 3 times the number of words by the end of the game using the PC approach with $\alpha_1 = -0.5$, compared to the original model (i.e., baseline). From these various plots, it is clear that the PC approach is versatile in changing player behavior in a controlled fashion.



(a) node 1 (b) node 3 (c) node 4

Figure 7: (a), (b), and (c) show the average number of words formed using the PC approach for nodes 1, 3, and 4, respectively. The curves are the time point-wise averages over the 100 instances, with error bars for ± one standard deviation. The three scenarios in the legend are original (i.e., all $\alpha_j = 0$); $\alpha_4 = 0.5$, other $\alpha_j = 0$; and $\alpha_1 = -0.5$, other $\alpha_j = 0$.

Table 1 presents the results of hypothesis testing using FANOVA. Each test has $n_a = n_b = 100$, as there are 100 simulation iterations under each condition. The objective is to demonstrate statistically that some values of $\boldsymbol{\alpha}$ do not significantly change the action sequence or the number of words formed sequence from those of the original method that does not employ the PC approach. These data show that the cases $\alpha_4 = 0.1$ and 0.2, and $\alpha_1 = -0.05$ do not have significant effects on player behavior.

Table 1: Functional analysis of variance (FANOVA) test results for the null hypothesis that there are no significant differences in the mean of sequences between the PC approach and the original method. If the p-value obtained from the test is ≥ 0.05, we fail to reject the null hypothesis and conclude that there is no significant difference in the mean of sequences. Conversely, if the p-value is ≤ 0.05, we reject the null hypothesis and conclude that there is a statistically significant difference in the mean of sequence.

	Number of Words		Action Sequence	
	p-value	Decision	p-value	Decision
$\alpha_4 = 0.1$	0.47	fail to reject	0.483	fail to reject
$\alpha_4 = 0.2$	0.068	fail to reject	0.083	fail to reject
$\alpha_4 = 0.5$	<0.001	reject	<0.001	reject
$\alpha_1 = -0.05$	0.736	fail to reject	0.389	fail to reject
$\alpha_1 = -0.1$	0.009	reject	0.005	reject
$\alpha_1 = -0.5$	<0.001	reject	<0.001	reject

5 CONCLUSION AND FUTURE WORK

In this work, we propose a Probability Calibration (PC) approach to simulate group anagram games with certain players having bot-like behaviors. The PC model starts with data-driven models and calibrates the parameters that alter player behaviors in principled ways. Our PC-based agent-based simulations demonstrate how this approach can change the performance of players in the game. In addition, statistical methods are used to compare the behaviors of players with different calibration parameters. Our contributions are listed in Section 1.4. This PC approach can also be applied to many other models where taking actions or changing states are governed by probabilities. Future work includes exploring more complicated expressions for calibration parameters $\boldsymbol{\alpha}$, investigating the effects of heterogeneous assignments of calibration parameters to players, and developing additional evaluation methodologies.

ACKNOWLEDGMENT

This work has been partially supported by NSF CRISP 2.0 (CMMI Grant 1916670) and COVA-CCI (Commonwealth Cyber Initiative-Coastal Virginia).

REFERENCES

Ahmad, M. A., B. Keegan, J. Srivastava, D. Williams, and N. Contractor. 2009. “Mining for Gold Farmers: Automatic Detection of Deviant Players in Mmogs”. In *2009 Int. Conf. on Computational Science and Engineering*, Volume 4, 340–345.

Bernardi, M. L., M. Cimitile, F. Martinelli, and F. Mercaldo. 2017. “A Time Series Classification Approach to Game Bot Detection”. In *Proceedings of the 7th International Conference on Web Intelligence, Mining and Semantics*, 1–11.

Cedeno-Mieles, V., Z. Hu, Y. Ren et al. 2019. “Mechanistic and Data-Driven Agent-Based Models to Explain Human Behavior in Networked Online Group Anagram Games”. In *IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, 357–364. IEEE.

Cedeno-Mieles, V., Z. Hu, Y. Ren et al. 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.

Centola, D. 2010. “The Spread of Behavior in an Online Social Network Experiment”. *Science* 329:1194–1197.

Centola, D., and M. Macy. 2007. “Complex Contagions and the Weakness of Long Ties”. *Am. J. Sociology* 113(3):702–734.

Chakraborty, T., S. Judd, M. Kearns, and J. Tan. 2010. “A Behavioral Study of Bargaining in Social Networks”. In *ACM Electronic Commerce*, 243–252.

Charness, G., R. Cobo-Reyes, and N. Jimenez. 2014. “Identities, Selection, and Contributions in a Public-Goods Game”. *Games and Economic Behavior* 87:322–338.

Charness, G., F. Feri, M. A. Melendez-Jimenez, and M. Sutter. 2014. "Experimental Games on Networks: Underpinnings of Behavior and Equilibrium Selection". *Econometrica* 82:1615–1670.

Corbae, D., and J. Duffy. 2008. "Experiments with Network Formation". *Games and Economic Behavior* 64:81–120.

Cuevas, A., M. Febrero, and R. Fraiman. 2004. "An Anova Test for Functional Data". *Computational Statistics & Data Analysis* 47(1):111–122.

den Hamer, P., and K. Frenken. 2021. "A Network-Based Model of Exploration and Exploitation". *Journal of Business Research* 129:589–599.

Fan, J., and S.-K. Lin. 1998. "Test of Significance When Data are Curves". *Journal of the American Statistical Association* 93(443):1007–1021.

Feather, N. T., and J. G. Simon. 1971. "Attribution of Responsibility and Valence of Outcome in Relation to Initial Confidence and Success and Failure of Self and Other". *Journal of Personality and Social Psychology* 18:173–188.

Feng, S. F., S. Wang, S. Zarnescu, and R. C. Wilson. 2021. "The Dynamics of Explore–Exploit Decisions Reveal a Signal-to-Noise Mechanism for Random Exploration". *Scientific Reports* 11:3077–1–3077–15.

Gopnik, A. 2020. "Childhood as a Solution to Explore-Exploit Tensions". *Trans. Royal Soc. B* 375:20190502–1–20190502–10.

Górecki, T., and Ł. Smaga. 2015. "A Comparison of Tests for the One-Way ANOVA Problem in Functional Data". *Computational Statistics* 30:987–1010.

Kearns, M., S. Judd, and Y. Vorobeychik. 2012. "Behavioral Experiments on a Network Formation Game". In *ACM Economics and Computation*, 690–704.

Kollock, P. 1998. "Social Dilemmas: The Anatomy of Cooperation". *Annual Review of Sociology* 24(1):183–214.

Korkmaz, G., M. Capra et al. 2018. "Coordination and Common Knowledge on Communication Networks". In *Proceedings of the 17th international conference on autonomous agents and multiagent systems*, 1062–1070.

Korkmaz, G., C. J. Kuhlman et al. 2014. "Collective Action Through Common Knowledge Using a Facebook Model". In *AAMAS*.

Liu, X., Z. Hu, X. Deng, and C. J. Kuhlman. 2022. "A Bayesian Uncertainty Quantification Approach for Agent-Based Modeling of Networked Anagram Games". In *Winter Simulation Conference (WSC)*, edited by B. Feng, G. Pedrielli, Y. Peng, S. Shashaani, C. C. E. Song, L. Lee, E. Chew, T. Roeder, and P. Lendermann, 310–321. Piscataway, New Jersey: IEEE.

Mason, W., and D. J. Watts. 2012. "Collaborative Learning in Networks". *PNAS* 109(3):764–769.

McCullagh, P. 2019. *Generalized Linear Models*. Routledge.

Mendoza, M., M. Tesconi, and S. Cresci. 2020. "Bots in Social and Interaction Networks: Detection and Impact Estimation". *ACM Transactions on Information Systems* 39.

Monroe, A. E., and B. F. Malle. 2017. "Two Paths to Blame: How Intentionality Directs Moral Information Processing along Dual Tracks". *Journal of Experimental Psychology* 146:123–133.

Mønsted, B., P. Sapieżyński, E. Ferrara, and S. Lehmann. 2017. "Evidence of Complex Contagion of Information in Social Media: An Experiment Using Twitter Bots". *Plos One* 12:e0184148–1–e0184148–12.

Platzer, C. 2011. "Sequence-Based Bot Detection in Massive Multiplayer Online Games". In *2011 8th International Conference on Information, Communications & Signal Processing*, 1–5. IEEE.

Polletta, F., and J. M. Jasper. 2001. "Collective Identity and Social Movements". *Annual Review of Sociology* 27:283–305.

Shen, Q., and J. Faraway. 2004. "An F Test for Linear Models with Functional Responses". *Statistica Sinica*:1239–1257.

Shirado, H., and N. A. Christakis. 2017. "Locally Noisy Autonomous Agents Improve Global Human Coordination in Network Experiments". *Nature* 545:370–374.

Zhang, J.-T. 2011. "Statistical Inferences for Linear Models with Functional Responses". *Statistica Sinica*:1431–1451.

AUTHOR BIOGRAPHIES

XUEYING LIU is a Ph.D. candidate in the Department of Statistics at Virginia Tech. She received her M.S. in Statistics from Georgia Tech. Her email address is xliu96@vt.edu.

ZHIHAO HU is a recently-graduated Ph.D. student in the Department of Statistics at Virginia Tech. He received his M.S. in Materials Science & Engineering from Virginia Tech. His email address is huzhihao@vt.edu.

XINWEI DENG is a professor in the Department of Statistics at Virginia Tech. He received his Ph.D. in Industrial Engineering from Georgia Tech. His email address is xdeng@vt.edu.

CHRIS J. KUHLMAN is a Research Associate Professor at the University of Virginia. He received a Ph.D. in Computer Science at Virginia Tech. His email address is hugo3751@gmail.com.