university of groningen
faculty of science and engineering
mathematics and applied mathematics

Modeling Opinion Dynamics: Biased Agents in Voter and Majority Rule Models

Bachelor's Project Mathematics

March 2025

Student: L.A. van Dijk

First supervisor: Dr. R. Szabó

Second assessor: Prof. dr. J.P. Trapman

Bachelor's Project Mathematics

Modeling Opinion Dynamics:
Biased Agents in Voter and Majority Rule Models

Linda van Dijk

S3051226

March 5, 2025

Abstract

Opinion dynamics in social networks studies the interplay between individual beliefs and collective behaviour. This thesis investigates binary opinion dynamics within connected social networks influenced by biased agents, as studied by Mukhopadhyay et al. ("Voter and majority dynamics with biased and stubborn agents," *Journal of Statistical Physics*, vol. 181, pp. 1239–1265, 2020). Biased agents have a preference for a specified opinion, the preferred opinion. Two updating rules are analysed; the voter and the majority rule. The voter rule stipulates that agents update their opinion by randomly sampling the opinion of another agent, whereas in the majority rule model, agents take a sample of $2K$ agents and adopt the majority opinion of this random sample. It is of interest to study whether consensus is reached, that is, when all agents hold the same opinion. The voter model analysis will demonstrate that the exit probability, the probability that consensus is achieved on the preferred opinion, converges to 1 as the network size increases to infinity. The analysis of the majority rule model will show that a phase transition exists in the exit probability. Moreover, the mean consensus time for both models is characterised to be $\Theta(\log N)$, where N is the network size.

Supervisors:

First supervisor: Dr. R. Szabó

Second assessor: Prof. dr. J.P. Trapman

Contents

1 Introduction

Mathematical modelling has been used to examine opinion dynamics in social networks, highlighting the complex interaction between personal beliefs and group behaviour. Understanding how opinions evolve and consensus forms in such networks is not only of academic interest but also holds practical significance. Applications of opinion dynamics can be seen in fields ranging from sociology and marketing to biology and physics, for example, in studies on species conflict [1] and interacting particle systems [2]. Models such as the voter and majority rule have yielded significant insights into these dynamics, although they frequently assume that individual decision making is unbiased. However, in reality, people can be influenced by various biases, which can significantly impact the formation of opinions. This thesis reviews the work of Mukhopadhyay et al. [3], providing an in-depth explanation of the proofs presented in their article. The focus here is on binary opinion dynamics modelled by the voter and majority rule models with biased agents. This means that the agents are biased to one of two opinions and the opinion which the agents are biased towards is called the *preferred opinion*. In their research, Mukhopadhyay et al. investigate how biases influence opinion formation within a connected network and the conditions under which consensus, meaning all agents hold the same opinion, is achieved. The analysis examines the convergence of the exit probability, the probability that the network reaches consensus on the preferred opinion, as the network size approaches infinity. In addition, the mean time needed to reach consensus within both models is characterised, showing that the expected time until consensus is reached is logarithmic in the network size.

The field of opinion dynamics is a relatively new area of study, but it is quickly increasing in popularity. Its foundations are found in sociology and socio-psychology. In the nineteenth century, phenomena such as herd mentality and mass hysteria were observed, leading to psychological studies in the twentieth century on conformity, social norm, and obedience, which showed that people tend to conform to the majority [4]. Economists later integrated this into analyses involving corporate investments and financial markets, while sociologists examined group polarisation, minority influence, and models of social influence [4]. In 1956, the use of a Markov chain model was introduced to represent the spread of social influence and the development of public opinions within social networks [5]. In political research, understanding the development of public opinions on issues or candidates is of great interest; this inspired the development of models such as the voter model. The voter model is based on the idea that people update their opinions through interactions with others. On the other hand, the majority rule model posits that individuals conform to the opinion held by the majority. The use of mathematical and computational tools has since expanded this research into a multidisciplinary framework, integrating social sciences with physics, mathematics, and complex systems to explore how opinions spread and evolve [4].

Early studies primarily examined opinion dynamics in societies structured on regular lattices. Recently, however, the focus has shifted toward complex networks, reflecting the growing significance of complex network science [4]. Studies on the voter model have looked at heterogeneous graphs [6], complete graphs with three states [7], random networks [8], connected d-regular graphs [9]. In contrast to the majority rule model presented in Mukhopadhyay et al., in the first study of the majority rule model, agents formed random groups, within which all members adopt the majority opinion of the group [10]. Cruised and Ganesh examined a broader model based on the majority rule model [11]. The voter model consisting of agents with a preference for one opinion on finite-dimensional lattices was initially explored by Mobilia [12]. A variation on biased agents is the stubborn agent who never changes their opinion, this has been studied by Mobilia et. al [13] and Yildiz et. al [14]. Various overviews of research on opinion dynamics have been created, including a broad history and overview of opinion dynamics from a multidisciplinary viewpoint [4], a review from the perspective of statistical physics [15], and a more recent survey that includes an algorithmic perspective [2].

Within the field of opinion dynamics, there are models that represent more than the binary choice of opinion of the voter and majority rule models discussed here, such as the culture dissemination model [16], [17], and the bounded confidence model [18]–[20]. Despite the binary choice of opinion being a simpler model, we find many practical sociological applications, such as voting for one of two political candidates, choosing to quit or continue smoking, and buying or selling decisions in financial markets [4]. Opinion dynamics is a relatively young field, and recent computational progress has significantly expanded the body of research. The analysis presented here is an exploration of foundations for the stochastic analysis of the voter and

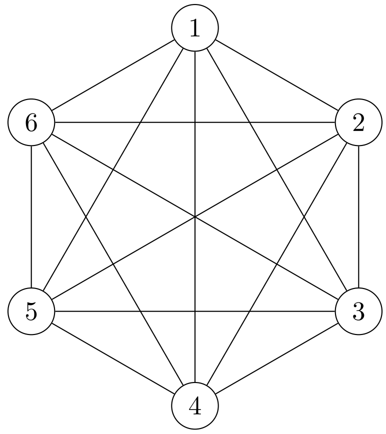


Figure 1: Complete graph with $N = 6$ nodes.

majority rule model with biased agents.

The model we consider consists of a network of N agents, each holding one of two opinions, 0 or 1. The networks examined are finite complete graphs, so that all agents can interact with every other agent in the network, depicted in Figure 1 for $N = 6$. Initially, a given fraction of all agents in the network hold opinion 1, while the rest hold opinion 0. We denote the number of agents starting with opinion 1 by $\lfloor \alpha N \rfloor$ for some $\alpha \in (0, 1)$, where $\lfloor \cdot \rfloor$ is the floor function. Each agent in the network considers updating their opinion at random times. The times at which agents consider updating their opinion follow independent Poisson point processes with rate 1, that is, each agent has an independent unit rate Poisson point process associated with themselves. Therefore, the inter-arrival times for an agent to consider updating their opinion is exponentially distributed with parameter 1. When an agent considers updating their opinion, they either retain their current opinion with probability $p_i = 1 - q_i$ or update their opinion with probability q_i, where $i \in \{0, 1\}$ indicates the opinion currently held by the agent.

Biased agents are individuals who are influenced by certain predispositions or preferences that affect their decision-making process. These biases can manifest as a tendency to favour one opinion over another, regardless of the opinions held by their neighbours. For example, a biased agent may be more inclined to adopt the opinion of a specific subgroup within the network, or it may have a predetermined inclination towards a particular opinion due to personal beliefs or external influences. In the model considered here, we assume that all agents are biased towards opinion 1, therefore, we set $q_1 < q_0$.

If an agent decides to update their opinion, they do so according to either the voter or majority rule model. When updating according to the voter model, the agent adopts the opinion of another agent chosen uniformly at random. If updating according to the majority rule model, the agent samples $2K$ agents uniformly at random and adopts the majority opinion among the sample plus themselves. This process of agents updating or keeping their opinion occurring at the times of their Poisson process continues indefinitely or until consensus is reached.

We can represent this model as a continuous-time Markov process, $X^{(N)}(t)$, with state space $\{0, 1, \ldots, N\}$ representing the number of agents with opinion 1 at time $t \geq 0$. The rate of transition from state i to j is denoted $q(i \to j)$. These transition rates differ for the voter and majority rule model, as these rates reflect the probability of agents switching opinions based on interactions and bias. For a continuous-time Markov process, we can define an embedded discrete-time Markov chain that is simpler to work with. The embedded discrete-time Markov chain is a random walk on $\{0, 1, ..., N\}$, with jump probabilities $p(n)$ to the right and $q(n) = 1 - p(n)$ to the left when the chain is in state n. We use this embedded chain to draw conclusions about the continuous-time process. Since the Markov chain is not irreducible, the chain will almost surely be absorbed into one of its essential states in finite time. In this case, that means that the Markov chain is absorbed in state 0 or N in finite time. In other words, consensus is reached on opinion 0 (absorption in state 0) or opinion 1 (absorption in state N). This random walk is depicted in Figure 2.

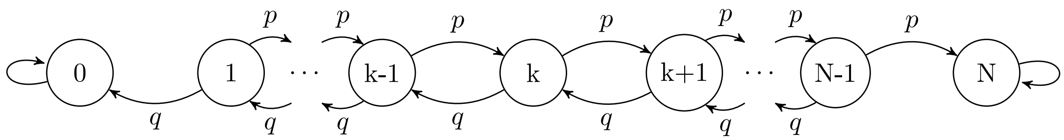


Figure 2: Random walk with absorbing states 0 and N.

We are interested in the likelihood of reaching consensus on the preferred opinion, also referred to as the exit probability and denoted by $E_N(\alpha)$, for network of size N and $\lfloor \alpha N \rfloor$ initial agents with the preferred opinion. The voter model analysis will demonstrate that the exit probability converges exponentially to 1 as the network size increases to infinity. This is in contrast to the voter model without biased agents, where the probability that consensus is reached on opinion i is constant and equals the initial fraction of agents with opinion i [3]. For the majority rule model, it is shown that there exists a phase transition in the exit probability, meaning that below a certain α, the exit probability converges to 0 as the network size increases to infinity, and above that threshold, it converges to 1. However, for the majority rule model without biased agents, consensus is reached on the opinion with the initial majority with high probability [3].

Furthermore, we are interested in the mean time to reach consensus, denoted by $t_N(\alpha)$. For both the voter and majority rule models, the mean consensus time is shown to be bounded from below and from above by a function that is logarithmic in N. We write $t_N(\alpha) = \Theta(\log N)$, where the function $f(n) = \Theta(g(n))$ indicates that for large enough n, there exist some constants C_1 and C_2 such that the function $f(n)$ is bounded from above by $C_1 \cdot g(n)$ and from below by $C_2 \cdot g(n)$. The mean consensus time for the unbiased majority rule model has also been shown to be logarithmic in N [3]. However, the mean consensus time for the unbiased voter model increases linearly with N [3].

This article is organised as follows. Before the analysis of the voter and majority rule model is presented, in Section 2 we review some key concepts from probability theory that will be used in the main results. In Section 3, the voter model with biased agents is presented, the exit probability is found, and the mean consensus time is characterised. Moreover, the theorems for the voter model are tested by running simulations. Next, in Section 4 the majority rule model with biased agents is presented, the exit probability is formulated, and the mean consensus time is characterised. Again, the theorems for the majority rule model are tested by running simulations. Finally, in Section 5, conclusions are presented, summarising the results, identifying limitations, and putting forward suggestions for future areas of research.

2 Theory

This section aims to outline and briefly explain some of the relevant theory that will be used in the analysis of the voter and majority rule models in Sections 3 and 4, respectively. In Section 2.1, we cover one-dimensional random walks and the gambler's ruin, including an important lemma on the probability of reaching one state before another. We discuss continuous-time Markov chains and Poisson processes in Section 2.2, as the agents in the voter and majority rule models update their opinion according to a unit rate Poisson process. In Section 2.3, we briefly explain the branching process with immigration, as this emerges in the demonstration of Theorem 3.2.

Firstly, the following lemmas give formulations of Markov's inequality [21] and Wald's identity [22], which are used throughout.

Lemma 2.1. *Markov's Inequality.* Let X be a non-negative random variable with a finite mean. For any constant $a \geq 0$,

$$\mathbb{P}(X \geq a) \leq \frac{\mathbb{E}[\mathbb{X}]}{a}.$$

Lemma 2.2. *Wald's identity.* Let $(X_n)_{n\in\mathbb{N}}$ be a sequence of independent identically distributed (i.i.d.) random variables and let $N \geq 0$ be an integer random variable that is independent of X_n, for all $n \in \mathbb{N}$. Assume that X_n and N are all integrable random variables, that is, they have finite mean. Then

$$\mathbb{E}\left[\sum_{i=1}^{N} X_i\right] = \mathbb{E}[N]\mathbb{E}[X_1].$$

2.1 One-dimensional Random Walks and the Gambler's Ruin

A random walk on $\mathbb{Z}$ is a stochastic process which takes steps along the integer line. Denote the random walk by $X(n)$, $n \geq 0$. The process takes a step to the right with a certain probability, p, and to the left with probability $q = 1 - p$. The gambler's ruin [23], [24] considers a random walk on the integers as the amount of money won by a gambler, with each move of the game the gambler can gain or lose \$1. The gambler starts with an initial sum of money and when the random walk reaches 0 the gambler has lost all his money and therefore cannot continue, meaning the state 0 is an absorbing state for this type of random walk. A gambler might set a goal for a sum of money to reach before he stops gambling, say \$$N$, then N is also an absorbing state for the random walk. It is of interest to study the probability that the gambler reaches 0 before N. To do this, we consider the first hitting time, T_k, of state k,

$$T_k := \inf\{n \geq 0 : X(n) = k\}.$$

A random walk is said to be symmetric if $p = q = 1/2$ and asymmetric if $p \neq q$. Assume that the random walk is asymmetric, specifically that $p > q$ and define $r := q/p < 1$. We are considering a one-dimensional random walk on the state space $\{0, 1, ..., N\}$, where 0 and N are absorbing states, as depicted in Figure 2. The following lemma [3] expresses the probability of reaching state a before reaching state b when starting the random walk in state x, with $0 \leq a < x < b \leq N$.

Lemma 2.3. Let $\mathbb{P}_x(\cdot) := \mathbb{P}(\cdot|X(0) = x)$. For any states a, b and x such that $0 \leq a < x < b \leq N$, we have

$$\mathbb{P}_x(T_a < T_b) = \frac{r^x - r^b}{r^a - r^b}. \tag{1}$$

Proof. We have $p \neq q$, and $r = q/p < 1$. Define $f(x) := \mathbb{P}_x(T_a < T_b)$, and note that $f(a) = 1$ and $f(b) = 0$. Furthermore, for any $x \in (a, b)$, we can write

$$\begin{aligned}
f(x) &= pf(x+1) + qf(x-1) \\
(p+q)f(x) &= pf(x+1) + qf(x-1) \\
p(f(x+1) - f(x)) &= q(f(x) - f(x-1)) \\
f(x+1) - f(x) &= \frac{q}{p}(f(x) - f(x-1)) \\
&= r(f(x) - f(x-1)).
\end{aligned}$$

From here we see that,

$$f(x+1) - f(x) = r^{x-a}(f(a+1) - f(a)).$$

We use $f(a) = 1$ and $f(b) = 0$ to obtain an equation in which $f(a+1)$ is the only unknown,

$$
\begin{aligned}
-1 = f(b) - f(a) &= \sum_{k=a}^{b-1} (f(k+1) - f(k)) \\
&= (f(a+1) - f(a)) \sum_{k=a}^{b-1} r^{k-a} \\
&= (f(a+1) - f(a)) \sum_{u=0}^{b-a-1} r^{u} \\
&= (f(a+1) - f(a)) \frac{r^{b-a} - 1}{r - 1}.
\end{aligned}
$$

This implies that,

$$
f(a+1) - f(a) = \frac{1 - r}{r^{b-a} - 1} = \frac{r - 1}{1 - r^{b-a}}. \tag{2}
$$

For any $x \in (a, b)$,

$$
\begin{aligned}
f(x) - f(a) &= \sum_{k=a}^{x-1} (f(k+1) - f(k)) \\
&= (f(a+1) - f(a)) \sum_{k=a}^{x-1} r^{k-a} \\
&= (f(a+1) - f(a)) \sum_{u=0}^{x-a-1} r^{u} \\
&= (f(a+1) - f(a)) \frac{r^{x-a} - 1}{r - 1} \\
&= \frac{r - 1}{1 - r^{b-a}} \cdot \frac{r^{x-a} - 1}{r - 1} && \text{(Substituting in equation (2))} \\
&= \frac{r^{x-a} - 1}{1 - r^{b-a}}.
\end{aligned}
$$

Using $f(a) = 1$ we get,

$$
\begin{aligned}
f(x) &= \frac{r^{x-a} - 1}{1 - r^{b-a}} + \frac{1 - r^{b-a}}{1 - r^{b-a}} \\
&= \frac{r^{x-a} - r^{b-a}}{1 - r^{b-a}} \\
&= \frac{r^{-a}(r^{x} - r^{b})}{r^{-a}(r^{a} - r^{b})} \\
&= \frac{r^{x} - r^{b}}{r^{a} - r^{b}},
\end{aligned}
$$

which completes the proof. $\square$

2.2 Markov Chains and Poisson Processes

A sequence of random variables $(X_t)_{t\in\mathbb{N}}$ is a *discrete-time Markov chain*, with state space χ and transition probabilities between states given by $p_{x,y} \in [0,1]$ for any $x, y \in \chi$, and

$$\mathbb{P}(X_{t+1} = x_{t+1}|X_t = x_t, \dots, X_1 = x_1, X_0 = x_0) = \mathbb{P}(X_{t+1} = x_{t+1}|X_t = x_t). \tag{3}$$

equation (3) is called the *Markov property* [25]. Clearly, the discrete-time Markov chain has transitions at distinct time intervals, since $t \in \mathbb{N}$.

Given $x, y \in \chi$, we say that y is *accessible* from x, denoted by $x \to y$, if the Markov chain can move from x to y in a finite number of steps. For $x, y \in \chi$, if y is accessible from x and x is accessible from y, we say that x *communicates* with y. A Markov chain is called *irreducible* if for any two states $x, y \in \chi$ there is a non-zero probability of reaching y from x in finite time. In other words, every state communicates with all the others. A state $x \in \chi$ is an *essential state* if for all y such that $x \to y$ it is also true that $y \to x$. An essential class is a subset of χ from which, once entered, the chain cannot exit. Every finite chain has at least one essential class. If $\{x\}$ is an essential class, then once the chain visits state x it never leaves, we then call x an *absorbing state* [25]. For a Markov chain that is not irreducible, it is known that the chain will almost surely be absorbed into one of its essential classes in finite time. If the Markov chain has one or more absorbing states, then the Markov chain will almost surely be absorbed into one of its absorbing states. These definitions and results are analogously applicable to continuous-time Markov chains.

In contrast to discrete-time Markov chains, a *continuous-time Markov chain* does not require that transitions between states occur at predetermined regular time intervals. Let $(X_t)_{t\in[0,\infty)}$ be a continuous-time process. The intervals $T_1, T_2, \dots,$ between transitions, are i.i.d. exponential random variables with rate λ. That is, each T_i is such that

$$\mathbb{P}(T_i \le t) = \begin{cases} 1 - e^{-\lambda t} & \text{if } t \ge 0, \\ 0 & \text{if } t < 0. \end{cases}$$

Let $(Y_k)_{k\in\mathbb{N}}$ be a discrete-time Markov chain with transition rates independent of the random transition times $(T_k)_{k\in\mathbb{N}}$. Define $S_0 = 0$ and for all $k \ge 1$, let $S_k := \sum_{i=1}^{k} T_i$. Let us define the continuous-time Markov chain by $X_t := Y_k$ for any $S_k \le t < S_{k+1}$. Changes in state will only occur at *transition times* $S_1, S_2, \dots$. However, if there is a non-zero probability that the chain remains in state n, then it is possible that when in state n the chain does not change state at a transition time, but rather remains in state n [25]. The sequence $(S_k)_{k\in\mathbb{N}}$ is also an *arrival process* [26]. The discrete-time Markov chain $(Y_k)_{k\in\mathbb{N}}$ is also called the *embedded discrete-time Markov chain* for the continuous-time Markov chain $(X_t)_{t\in[0,\infty)}$. The embedded chain can be used in the analysis of the continuous-time Markov chain, since it represents the state changes made without considering the time spent in each state.

Let $N_t := \max\{k : S_k \le t\}$, that is the number of transitions up to and including time t. Note that $N_t = k$ if and only if $S_k \le t < S_{k+1}$. Assume the starting state of the process is x, then

$$\mathbb{P}_x(X_t = n|N_t = k) = \mathbb{P}_x(Y_k = n).$$

The process described by $(N_t)_{t\ge0}$ is in fact a counting process, specifically a *Poisson process*, since it has exponential inter-arrival times [26]. Therefore, N_t has a Poisson distribution with mean λt [25], that is

$$\mathbb{P}(N_t = n) = \frac{e^{-\lambda t}(\lambda t)^n}{n!}.$$

For any time interval of length s, the expected number of arrivals is λs, and λ is also called the arrival rate of the process. From this we see that we can represent the voter and majority rule models using a continuous-time Markov chain. The following are two important theorems for Poisson processes.

Theorem 2.4. *Superposition theorem.* Let P_1 and P_2 be independent Poisson processes with rates λ_1 and λ_2, respectively. Then their superposition $P_1 + P_2$ is also a Poisson process and has rate $\lambda_1 + \lambda_2$.

Theorem 2.5. *Thinning theorem.* Let P be a Poisson process with rate λ. From P two independent Poisson processes, P_1 and P_2, are constructed by assigning each point of the process P to the process P_1 with probability p and to the process P_2 with probability $q = 1 - p$. The resulting processes P_1 and P_2 are independent Poisson processes with rates $p\lambda$ and $(1 - p)\lambda$.

2.3 Branching Processes with Immigration

A *branching process* is a stochastic process used to represent the behaviour of populations or families, how they grow over time. This is done by considering generations of individuals who give birth to offsprings creating the next generation. A branching process is often also called the Galton-Watson process. Let $\xi_{n,k}$, $n \geq 0$ and $k \geq 1$, be i.i.d. non-negative integer random variables. Set Z_0 to be some positive integer, and

$$Z_{n+1} = \begin{cases} \sum_{k=1}^{Z_n} \xi_{n,k}, & \text{for } Z_n > 0, \\ 0 & \text{for } Z_n = 0, \end{cases}$$

for all $n \geq 0$. Then $(Z_n)_{n\in\mathbb{N}}$ forms a Galton–Watson branching process with an initial population of Z_0 [27]. A modification of this branching process is the addition of immigration [28]. Assume that a random number Y_n of immigrants enter the process in the n^{th} generation, again Y_n are i.i.d. non-negative integer random variables. Moreover, assume that Y_n is independent of $\xi_{n,k}$ for all n and k. Then the sequence $(Z_n)_{n\in\mathbb{N}}$ forms a Galton-Watson process with immigration, where

$$Z_{n+1} = \begin{cases} \sum_{k=1}^{Z_n} \xi_{n,k} + Y_n, & \text{for } Z_n > 0, \\ 0 & \text{for } Z_n = 0. \end{cases}$$

Although there exist more complex formulations of branching processes with immigration, for this thesis, we will only require the scenario where the immigration rate is set at 1, meaning $Y_n = 1$ for every n.

3 Voter Model with Biased Agents

In this section, the voter model with biased agents is explored. It is demonstrated that the exit probability converges to one as the network size increases and the average time for consensus to be reached is characterised.

The network has N agents and initially each agent adopts one of two opinions, 0 or 1. The fraction of agents who initially have opinion 1 is given by $\lfloor \alpha N \rfloor$. Each agent independently considers updating their opinion at the points of a Poisson point process with parameter 1, that is, the inter-arrival times for an agent to consider an opinion update are exponential with parameter 1. When considering an update, an agent with opinion $i \in \{0, 1\}$ decides to update their opinion with probability q_i and retains their opinion with probability $p_i = 1 - q_i$. We choose $q_1 < q_0$, making the agents biased towards opinion 1. If the agent decides to retain their opinion, they continue with their held opinion until the next point at which they consider updating their opinion. If they decide to update their opinion, they do so according to the voter rule, this states that the agent selects an agent uniformly at random from the N agents in the network.

Let $X^N(t)$ denote the number of agents with opinion 1 at time $t \geq 0$. $X^{(N)}$ is a continuous-time Markov process on state space $\{0, 1, ..., N\}$, with absorbing states 0 and N. The rates of transition out of state k are

$$q(k \to k + 1) = q_0 k \frac{N - k}{N},$$
$$q(k \to k - 1) = q_1 k \frac{N - k}{N}.$$

To go from state k to $k + 1$, an agent with opinion 0 needs to update their opinion to 1. First, an agent with opinion 0 must consider updating their opinion. Since each individual agent considers updating their opinion according to independent Poisson processes with rate 1, we can apply Theorem 2.4, the Superposition

Theorem for Poisson processes. The combined Poisson processes of the $N-k$ agents with opinion 0 is a Poisson process with rate $N-k$. The interarrival times of this combined Poisson process follows an exponential distribution with rate $N-k$. So, at a rate of $N-k$ an agent with opinion 0 will consider updating their opinion. Then this agent with opinion 0 will update their opinion with probability q_0. Since we are looking at the voter model, if the agent has decided to update their opinion, we select an agent uniformly at random, and the agent will adopt this agent's opinion. Since there are k agents with opinion 1, the probability that the selected agent is of opinion 1 and therefore that the agent adopts opinion 1 is k/N.

Similarly, to go from state k to $k-1$, an agent with opinion 1 must update their opinion to 0. An agent with opinion 1 must be considering updating their opinion, this happens at a rate of k. The probability that the agent considers updating their opinion is q_1 and the probability they select an agent with opinion 0 to adopt the opinion of is $(N-k)/N$.

The embedded discrete-time Markov chain $\tilde{X}^{(N)}$ for $X^{(N)}$ is a one-dimensional random walk on $\{0, 1, ..., N\}$ with jump probabilities $p = q_0/(q_0+q_1)$ to the right and $q = 1-p$ to the left. Define $r = q/p < 1$ and $\bar{r} = 1/r$.

When the Markov process is absorbed in state N then consensus has been achieved on the preferred opinion. It is of interest to study the average time it takes for consensus to be reached and the probability that the process is absorbed into state N, also known as the exit probability. Denote the mean time to reach consensus by $t_N(\alpha) := \mathbb{E}_{\lfloor \alpha N \rfloor}[T_0 \wedge T_N]$, where $A \wedge B = \min(A, B)$ and $\mathbb{E}_x[\cdot]$ is the expectation conditioned on the initial state or the starting state of the Markov chain being x. The exit probability to reach consensus on the preferred opinion is denoted by $E_N(\alpha) := \mathbb{P}_{\lfloor \alpha N \rfloor}(T_N < T_0)$.

3.1 Exit Probability

In this section, it is demonstrated that for the voter model, the exit probability converges to 1 as the network size increases to infinity.

Theorem 3.1. Let $E_N(n)$ denote the probability that the process $X^(N)$ is absorbed into state N when starting from state n. Let $E_N(\alpha) := E_N(\lfloor \alpha N \rfloor)$, for $\alpha \in (0, 1)$. Then, $E_N(\alpha) \to 1$ as $N \to \infty$ and this convergence is exponential in N.

Proof. We have

$$
\begin{aligned}
E_N(\alpha) &= \mathbb{P}_{\lfloor \alpha N \rfloor}(T_N < T_0) && \text{(By definition)} \\
&= \frac{1 - r^{\lfloor \alpha N \rfloor}}{1 - r^N} && \text{(By Lemma 2.3)} \\
&\geq 1 - r^{\lfloor \alpha N \rfloor} && \text{(Since } 1 - r^N < 1) \\
&= 1 - \exp(\ln(r^{\lfloor \alpha N \rfloor})) \\
&= 1 - \exp(-cN), && (4)
\end{aligned}
$$

for some constant $c > 0$ (since $r < 1$). So, $E_N(\alpha) \geq 1 - \exp(-cN)$, meaning that as the network size increases, the probability of having consensus on the preferred opinion approaches one exponentially fast. □

In contrast, the probability that the network reaches consensus on the non-preferred opinion, that is, $1 - E_N(\alpha) = \exp(-cN)$, converges to zero exponentially fast.

Remark. In the voter model with unbiased agents, the probability of reaching consensus on either opinion has been found to be constant with respect to N, in particular the exit probability on any opinion equals the initial fraction of agents with that opinion [3].

3.2 Mean Consensus Time

Here we characterise the mean time to reach consensus, $t_N(\alpha) = \mathbb{E}_{\lfloor \alpha N \rfloor}[T_0 \wedge T_N]$. This is done in the following theorem, which states that the average time to reach consensus is logarithmic in N, the size of the network, for a large enough N.

Theorem 3.2. For all $\alpha \in (0,1)$ we have $t_N(\alpha) = \Theta(\log N)$.

Remark. The mean consensus time for the unbiased voter model increases linearly with N, and is approximated by $Nh(\alpha)$ for large N, with $h(\alpha) = -[\alpha \ln(\alpha) + (1-\alpha)\ln(1-\alpha)]$ [3].

Theorem 3.2 claims that the mean consensus time is bounded from above and below by a function that is logarithmic in N, for N large enough. In other words, there exists an N_0 and constants C_1 and C_2 such that for all $N > N_0$, $t_N(\alpha)$ is bounded from below by a function $C_1 \cdot \log(N)$ and bounded from above by a function $C_2 \cdot \log(N)$. We prove Theorem 3.2 by finding this lower and upper bound for $t_N(\alpha)$.

Proof. Let $T := T_0 \wedge T_N$ denote the random time to reach consensus. We can rewrite T as

$$T = \sum_{k=1}^{N-1} \sum_{j=1}^{Z_k} M_{k,j}, \tag{5}$$

where Z_k is the number of visits to state k before absorption and $M_{k,j}$ is the time spent in the j^{th} visit to state k. In other words, to get T we are summing the time spent in each state, by splitting the time spent in each state into time spent in the state per visit to that state. Each $M_{k,j}$ is an exponential random variable with rate $\lambda = (q_0 + q_1)k(N-k)/N$, that is, the sum of the rate of transition from state k to $k-1$ and k to $k+1$. It is important to note that Z_k and $(M_{k,j})_{j \geq 1}$ are independent, because the time spent in a state is based on exponentially distributed inter-arrival times, which are independent of everything else. The expectation of $M_{k,j}$ is

$$\begin{aligned}
\mathbb{E}_{\lfloor \alpha N \rfloor}[M_k, j] &= \frac{1}{\lambda} \\
&= \frac{1}{(q_0+q_1)k(N-k)/N} \\
&= \frac{N}{(q_0+q_1)k(N-k)} \\
&= \frac{1}{q_0+q_1}\left(\frac{N}{k(N-k)}\right) \\
&= \frac{1}{q_0+q_1}\left(\frac{1}{k} + \frac{1}{N-k}\right). \qquad (6)
\end{aligned}$$

The mean consensus time can be rewritten as follows,

$$\begin{aligned}
t_N(\alpha) &= \mathbb{E}_{\lfloor \alpha N \rfloor}[T] \\
&= \mathbb{E}_{\lfloor \alpha N \rfloor}\left[\sum_{k=1}^{N-1} \sum_{j=1}^{Z_k} M_{k,j}\right] \\
&= \sum_{k=1}^{N-1} \mathbb{E}_{\lfloor \alpha N \rfloor}\left[\sum_{j=1}^{Z_k} M_{k,j}\right] \\
&= \sum_{k=1}^{N-1} \mathbb{E}_{\lfloor \alpha N \rfloor}[Z_k]\mathbb{E}_{\lfloor \alpha N \rfloor}\left[M_{k,j}\right] && \text{(Lemma 2.2: Wald's identity)} \\
&= \frac{1}{q_0+q_1} \sum_{k=1}^{N-1} \left(\frac{1}{k} + \frac{1}{N-k}\right)\mathbb{E}_{\lfloor \alpha N \rfloor}[Z_k] && \text{(Using equation (6)).} \qquad (7)
\end{aligned}$$

We find lower and upper bounds for $t_N(\alpha)$. Let $A = \{\omega : T_N(\omega) < T_0(\omega)\}$ be the event that the Markov chain gets absorbed in state N. The expectation of Z_k can be split into the two possible absorbing outcomes, either the Markov chain being absorbed into state 0 or state N.

$$\mathbb{E}_{\lfloor \alpha N \rfloor}[Z_k] = \mathbb{E}_{\lfloor \alpha N \rfloor}[Z_k|A]\mathbb{P}_{\lfloor \alpha N \rfloor}(A) + \mathbb{E}_{\lfloor \alpha N \rfloor}\left[Z_k|A^c\right](1 - \mathbb{P}_{\lfloor \alpha N \rfloor}(A)). \tag{8}$$

Lower bound of $t_N(\alpha)$. The following holds for Z_k,

$$\begin{aligned} Z_k|A &\geq 1 \quad \text{for all } k \geq \lfloor \alpha N \rfloor \\ Z_k|A &\geq 0 \quad \text{for all } k < \lfloor \alpha N \rfloor \\ Z_k|A^c &\geq 1 \quad \text{for all } k \leq \lfloor \alpha N \rfloor \\ Z_k|A^c &\geq 0 \quad \text{for all } k > \lfloor \alpha N \rfloor. \end{aligned}$$

This states that for all $k \geq \lfloor \alpha N \rfloor$ and in event A, the Markov chain must visit state k at least once, hence $Z_k|A \geq 1$, for all $k \geq \lfloor \alpha N \rfloor$. Similarly, if $k \leq \lfloor \alpha N \rfloor$ and the Markov chain is absorbed into state 0 (event A^c), then the Markov chain must visit state k at least once. On the other hand, in event A^c and $k > \lfloor \alpha N \rfloor$ then the Markov chain need not necessarily visit state k (that is not to say it won't, but it is not necessary), therefore we say $Z_k|A^c \geq 0$ for all $k > \lfloor \alpha N \rfloor$. Likewise, in the case of event A and $k < \lfloor \alpha N \rfloor$ then the Markov chain need not necessarily visit state k, so we say $Z_k|A \geq 0$ for all $k < \lfloor \alpha N \rfloor$. Using the above four cases in combination with equation (8) a lower bound is obtained for $\mathbb{E}_{\lfloor \alpha N \rfloor}[Z_k]$,

$$\mathbb{E}_{\lfloor \alpha N \rfloor}[Z_k] \geq \mathbb{P}_{\lfloor \alpha N \rfloor}(A)\mathbb{1}_{\{k \geq \lfloor \alpha N \rfloor\}} + (1 - \mathbb{P}_{\lfloor \alpha N \rfloor}(A))\mathbb{1}_{\{k \leq \lfloor \alpha N \rfloor\}}. \tag{9}$$

This can be input into equation (7) to give,

$$\begin{aligned} t_N(\alpha) &\geq \frac{1}{q_0 + q_1} \sum_{k=1}^{N-1} \left(\frac{1}{k} + \frac{1}{N-k}\right)\left[\mathbb{P}_{\lfloor \alpha N \rfloor}(A)\mathbb{1}_{\{k \geq \lfloor \alpha N \rfloor\}} + (1 - \mathbb{P}_{\lfloor \alpha N \rfloor}(A))\mathbb{1}_{\{k \leq \lfloor \alpha N \rfloor\}}\right] \\ &= \frac{\mathbb{P}_{\lfloor \alpha N \rfloor}(A)}{q_0 + q_1} \sum_{k=\lfloor \alpha N \rfloor}^{N-1} \left(\frac{1}{k} + \frac{1}{N-k}\right) + \frac{1 - \mathbb{P}_{\lfloor \alpha N \rfloor}(A)}{q_0 + q_1} \sum_{k=1}^{\lfloor \alpha N \rfloor} \left(\frac{1}{k} + \frac{1}{N-k}\right) \\ &= \frac{\mathbb{P}_{\lfloor \alpha N \rfloor}(A)}{q_0 + q_1} \sum_{k=\lfloor \alpha N \rfloor}^{N-1} \left(\frac{1}{k}\right) + \frac{\mathbb{P}_{\lfloor \alpha N \rfloor}(A)}{q_0 + q_1} \sum_{k=1}^{N-\lfloor \alpha N \rfloor} \left(\frac{1}{k}\right) \\ &\qquad + \frac{1 - \mathbb{P}_{\lfloor \alpha N \rfloor}(A)}{q_0 + q_1} \sum_{k=1}^{\lfloor \alpha N \rfloor} \left(\frac{1}{k}\right) + \frac{1 - \mathbb{P}_{\lfloor \alpha N \rfloor}(A)}{q_0 + q_1} \sum_{k=N-\lfloor \alpha N \rfloor}^{N-1} \left(\frac{1}{k}\right). \end{aligned}$$

In this next step, we simplify this expression of the lower bound. We do this by considering that the summations have different number of terms, namely $N - \lfloor \alpha N \rfloor$ and $\lfloor \alpha N \rfloor$ terms. Taking the minimum of these values gives $\lfloor \alpha N \rfloor \wedge (N - \lfloor \alpha N \rfloor) > (\alpha N - 1) \wedge N(1 - \alpha) = N((\alpha - 1/N) \wedge (1 - \alpha))$. If N large enough, then $1/N < 10^{-6}\alpha$, so we can take this minimum as $N(\alpha \wedge (1 - \alpha))$. Taking all summations from 1 to this minimum, we get a further lower bound. This allows us to simplify the equation, cancelling out the $\mathbb{P}_{\lfloor \alpha N \rfloor}(A)$ terms. For simplicity, we also divide the expression by 2; we can do this because we are finding a lower bound. Putting this together gives

$$\begin{aligned} t_N(\alpha) &\geq \frac{1}{q_0 + q_1} \sum_{k=1}^{N(\alpha \wedge (1-\alpha))} \frac{1}{k} \\ &> \frac{1}{q_0 + q_1} \log(N(\alpha \wedge (1 - \alpha))). \end{aligned}$$

Hereby, obtaining a lower bound for $t_N(\alpha)$ as a function of $\log(N)$.

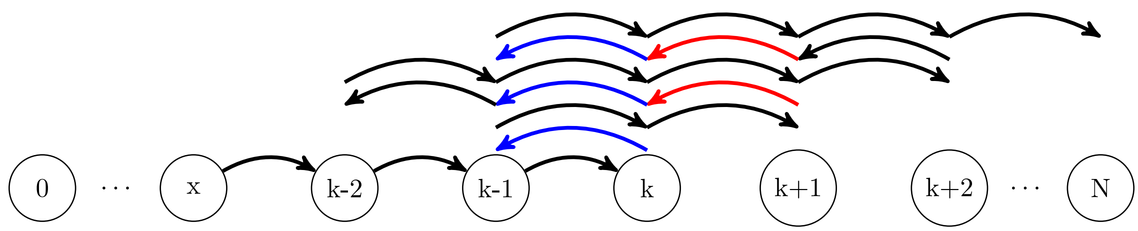


Figure 3: Depiction of the number of visits to state k, Z_k, given absorption in state N, with ζ_k in blue and ζ_{k+1} in red.

Upper bound for $t_N(\alpha)$. In order to find an upper bound for $t_N(\alpha)$, we first find an upper bound on $\mathbb{E}_x[Z_k; A]$ for all $k \geq x$ with any $0 < x < N$. We define $\mathbb{E}_x[X; A] := \sum_x x\mathbb{P}(\{X = x\} \cap A)$.

Assuming that the Markov chain is absorbed in state N. Conditioned on A the embedded discrete-time Markov chain and the jump probabilities are modified as follows,

$$\begin{aligned}
p^A_{k,k+1} &= \mathbb{P}(\tilde{X}_{t+1} = k+1 | \tilde{X}_t = k, A) \\
&= \frac{\mathbb{P}(\tilde{X}_{t+1} = k+1, \tilde{X}_t = k, A)}{\mathbb{P}(\tilde{X}_t = k, A)} \\
&= \frac{\mathbb{P}(\tilde{X}_{t+1} = k+1, A | \tilde{X}_t = k) \cdot \mathbb{P}(\tilde{X}_t = k)}{\mathbb{P}(A | \tilde{X}_t = k) \cdot \mathbb{P}(\tilde{X}_t = k)} \\
&= \frac{\mathbb{P}(\tilde{X}_{t+1} = k+1, A, \tilde{X}_t = k)}{\mathbb{P}(\tilde{X}_t = k) \cdot \mathbb{P}_k(A)} \\
&= \frac{\mathbb{P}(A | \tilde{X}_{t+1} = k+1, \tilde{X}_t = k) \cdot \mathbb{P}(\tilde{X}_{t+1} = k+1, \tilde{X}_t = k)}{\mathbb{P}(\tilde{X}_t = k) \cdot \mathbb{P}_k(A)} \\
&= \frac{\mathbb{P}_{k+1}(A) \cdot \mathbb{P}(\tilde{X}_{t+1} = k+1 | \tilde{X}_t = k) \cdot \mathbb{P}(\tilde{X}_t = k)}{\mathbb{P}(\tilde{X}_t = k) \cdot \mathbb{P}_k(A)} \\
&= p\frac{\mathbb{P}_{k+1}(A)}{\mathbb{P}_k(A)} \\
&= p\frac{\mathbb{P}_{k+1}(T_N < T_0)}{\mathbb{P}_k(T_N < T_0)} \\
&= p\frac{r^{k+1} - r^0}{r^N - r^0} \cdot \frac{r^N - r^0}{r^k - r^0} && \text{(By Lemma 2.3)} \\
&= p\frac{1 - r^{k+1}}{1 - r^k}, && (10)
\end{aligned}$$

and $p^A_{k,k+1} = 1 - p^A_{k,k-1}$. Let ζ_k denote the number of times the embedded chain moves from k to $k-1$ before absorption. We have following formulation for the number of visits to state k conditioned on absorption in state N.

$$Z_k | A = 1 + \zeta_k + \zeta_{k+1}, \text{ for } x \leq k \leq N-1. \tag{11}$$

This follows from the fact that the states $k \geq x$ are visited at least once, since we start in state x and we have conditioned on absorption in state N. This first visit occurs before any left jumps from k or $k+1$. Every time the chain jumps to the left of state k, the chain must visit state k again, since we have conditioned on absorption in N. And every left jump from $k+1$ is necessarily a visit to k. This is depicted in Figure 3, where the chain shown visits state k six times before absorption in N, with ζ_k represented in blue and ζ_{k+1} represented in red.

Given that the Markov chain is absorbed in state N, it must be the case that $\zeta_N = 0$. Let $\xi_{l,k}$ denote the random number of left-jumps from state k between the l^{th} and $(l+1)^{th}$ left-jumps from state $k+1$.

Then $(\xi_{l,k})_{l\geq 0}$ are random variables that are i.i.d. with geometric distribution from zero having mean $p^A_{k,k-1}/p^A_{k,k+1}$. They are geometrically distributed from zero because they describe the number of trials or failures before the first success; in this situation, the left-jumps from state k are the failures and the first success is the next right-jump from k. The mean for the geometric distribution from zero is the probability of failure over the probability of success, that is $p^A_{k,k-1}/p^A_{k,k+1}$.

Observe that ζ_k can be written using the following recursion,

$$\zeta_k = \sum_{l=0}^{\zeta_{k+1}} \xi_{l,k}, \text{ for } x \leq k \leq N-1, \tag{12}$$

which breaks the number of left-jumps from state k before absorption into the number of left-jumps from k between consecutive left-jumps from state $k+1$. We sum to ζ_{k+1} because this is the number of left-jumps from state $k+1$ before absorption. The reason the sum starts from $l=0$ is that, for all $k \geq x$, left-jumps from k can occur even before the chain visits $k+1$ for the first time.

Note that $(\zeta_k)_{x\leq k\leq N}$ forms a branching process with immigration of one individual in each generation. This means that in addition to the offspring of the previous generation, there is an extra individual who immigrated into the process [28]. Consider $\xi_{l,k}$ as the number of offsprings of the l^{th} individual in generation k, every state being a generation. The generation above generation k is generation $k+1$, because the number of left-jumps $\xi_{l,k}$ depends on the left-jumps from state $k+1$. The immigration of one comes from the fact that with $k \geq x$, the left-jumps from k can occur before we have reached state $k+1$. In the summation in equation (12) this is $l=0$, and in terms of a branching process we treat this case as one immigrant joining generation k.

Applying Wald's identity (Lemma 2.2) we find the expectation of equation (12), for all $x \leq k \leq N$,

$$\begin{aligned}
\mathbb{E}_x[\zeta_k] &= \mathbb{E}_x\left[\sum_{l=0}^{\zeta_{k+1}} \xi_{l,k}\right] \\
&= \mathbb{E}_x[\xi_{0,k}] + \mathbb{E}_x\left[\sum_{l=1}^{\zeta_{k+1}} \xi_{l,k}\right] \\
&= \mathbb{E}_x[\xi_{0,k}] + \mathbb{E}_x[\zeta_{k+1}]\mathbb{E}_x[\xi_{l,k}] && \text{(Lemma 2.2: Wald's identity)} \\
&= \frac{p^A_{k,k-1}}{p^A_{k,k+1}} + \frac{p^A_{k,k-1}}{p^A_{k,k+1}}\mathbb{E}_x\left[\sum_{l=0}^{\zeta_{k+2}} \xi_{l,k+1}\right] \\
&= \sum_{n=k}^{N-1}\prod_{i=k}^{n} \frac{p^A_{i,i-1}}{p^A_{i,i+1}} && \text{(from continuing the recursion)} \\
&= \sum_{n=k}^{N-1}\prod_{i=k}^{n} \left(1 - p\frac{1-r^{i+1}}{1-r^i}\right)\left(\frac{1}{p}\cdot\frac{1-r^i}{1-r^{i+1}}\right) && \text{(from equation (10))} \\
&= \sum_{n=k}^{N-1}\prod_{i=k}^{n} \frac{1}{p}\cdot\frac{1-r^i}{1-r^{i+1}} - 1 \\
&= \sum_{n=k}^{N-1}\prod_{i=k}^{n} \frac{1-r^i-p+pr^{i+1}}{p(1-r^{i+1})} \\
&= \sum_{n=k}^{N-1}\prod_{i=k}^{n} \frac{pr\cdot r^i - r^i + 1 - p}{p(1-r^{i+1})} \\
&= \sum_{n=k}^{N-1}\prod_{i=k}^{n} \frac{(1-p)r^i - r^i + pr}{p(1-r^{i+1})} && \text{(since } pr = 1-p)
\end{aligned}$$

$$
\begin{aligned}
&= \sum_{n=k}^{N-1} \prod_{i=k}^{n} \frac{pr - pr^i}{p(1-r^i)} \\
&= \sum_{n=k}^{N-1} \prod_{i=k}^{n} \frac{r - r^i}{1-r^{i+1}} \\
&= \sum_{n=k}^{N-1} \prod_{i=k}^{n} \frac{r(1 - r^{i-1})}{1-r^{i+1}} \\
&= \sum_{n=k}^{N-1} r^{n-k+1} \cdot \frac{(1-r^{k-1})(1-r^k)}{(1-r^n)(1-r^{n+1})} \\
&= r(1-r^{k-1})(1-r^k) \sum_{n=k}^{N-1} \frac{r^{n-k}}{(1-r^n)(1-r^{n+1})} \\
&= \frac{r(1-r^{k-1})(1-r^{N-k})}{(1-r)(1-r^N)}. \qquad (13)
\end{aligned}
$$

The last equality can be proven using induction on N. Taking the expectation of (11) and substituting in (13) we see that, for all $x \le k \le N-1$,

$$
\begin{aligned}
\mathbb{E}_x[Z_k|A] &= 1 + \mathbb{E}_x[\zeta_k] + \mathbb{E}_x[\zeta_{k+1}] \\
&= 1 + \frac{r(1-r^{k-1})(1-r^{N-k})}{(1-r)(1-r^N)} + \frac{r(1-r^k)(1-r^{N-k-1})}{(1-r)(1-r^N)} \\
&= \frac{1 - r - r^N + r^{N+1} + r - r^k - r^{N-k+1} + r^N + r - r^{k+1} - r^{N-k} + r^N}{(1-r)(1-r^N)} \\
&= \frac{1+r}{1-r} \cdot \frac{(1-r^{N-k})(1-r^k)}{1-r^N} \\
&\le \frac{1+r}{1-r}. \qquad (14)
\end{aligned}
$$

Note that,

$$
\mathbb{E}[X|A] = \sum_x x \cdot \mathbb{P}(X = x|A) = \sum_x \frac{x \cdot \mathbb{P}(\{X = x\} \cup A)}{\mathbb{P}(A)} = \mathbb{E}[X; A] \cdot \frac{1}{\mathbb{P}(A)},
$$

and with $0 < \mathbb{P}(A) \le 1$ it is clear that $\mathbb{E}[X; A] \le \mathbb{E}[X|A]$. Putting this together with equation (14) we have,

$$
\mathbb{E}_x[Z_k; A] \le \mathbb{E}_x[Z_k|A] \le \frac{1+r}{1-r},
$$

for all $x \le k \le N-1$. This bound is independent of the choice of x and thus holds for $x = \lfloor \alpha N \rfloor$, as such we have found an upper bound on $\mathbb{E}_{\lfloor \alpha N \rfloor}[Z_k; A]$ for all $k \ge \lfloor \alpha N \rfloor$. For all $1 \le k < x$,

$$
\begin{aligned}
\mathbb{E}_x[Z_k; A] &= \mathbb{E}_x[Z_k; T_k < T_N < T_0] \\
&= \mathbb{E}_x[Z_k|T_k < T_N < T_0]\mathbb{P}_x(T_k < T_N < T_0) \\
&= \mathbb{E}_k[Z_k|T_N < T_0]\mathbb{P}_x(T_k < T_N < T_0) && \text{(By the Markov property)} \\
&\le \frac{1+r}{1-r} && \text{(By equation (14)).}
\end{aligned}
$$

Combining these results we have an upper bound for $\mathbb{E}_x[Z_k; A]$, for all $0 < x < N$,

$$\mathbb{E}_x[Z_k; A] \leq \frac{1+r}{1-r}.$$

Consider the event A^c, the Markov chain is absorbed into state 0. We find an upper bound for $E_x[Z_k; A^c]$ for any $0 < x < N$. We have,

$$p_{k,k+1}^{A^c} = p\frac{\mathbb{P}_{k+1}(A^c)}{\mathbb{P}_k(A^c)},$$

and,

$$p_{k,k-1}^{A^c} = q\frac{\mathbb{P}_{k-1}(A^c)}{\mathbb{P}_k(A^c)}.$$

Define $\bar{\zeta}_k$ to be the number of times the embedded chain jumps from k to $k+1$, then $Z_k|A^c = 1 + \bar{\zeta}_{k-1} + \bar{\zeta}_k$ for all $1 \leq k \leq x$. Given that the Markov chain is absorbed in state 0, we have $\bar{\zeta}_0 = 0$. Given A^c, denote by $\bar{\xi}_{l,k}$ the random number of right-jumps from k between the l^{th} and $(l+1)^{th}$ right-jumps f rom $k-1$. Again, these $\bar{\xi}_{l,k}$ are i.i.d. geometric from zero with mean $p_{k,k+1}^{A^c}/p_{k,k-1}^{A^c}$. Moreover,

$$\bar{\zeta}_k = \sum_{l=0}^{\bar{\zeta}_{k-1}} \xi_{l,k},$$

for all $1 \leq k \leq x$. Again, this is a branching process with immigration of one, and we derive the following expression for the expectation of $\bar{\zeta}_k$.

$$\begin{aligned}
\mathbb{E}_x[\bar{\zeta}_k] &= \sum_{n=1}^{k}\prod_{i=n}^{k}\frac{p_{i,i+1}^{A^c}}{p_{i,i-1}^{A^c}} \\
&= \sum_{n=1}^{k}\prod_{i=n}^{k}\bar{r}\frac{r^{i+1}-r^N}{r^{i-1}-r^N} \\
&= \bar{r}^{k+1}(r^k - r^N)(r^{k+1}-r^N)\sum_{n=1}^{k}\frac{\bar{r}^{-n}}{(r^{n-1}-r^N)(r^n-r^N)} \\
&= \bar{r}^{k+1}(\bar{r}^{N-k}-1)(\bar{r}^{N-k-1}-1)\sum_{n=1}^{k}\frac{\bar{r}^{-n}}{(\bar{r}^{N-n+1}-1)(\bar{r}^{N-n}-1)} \\
&= \frac{\bar{r}(\bar{r}^k-1)(\bar{r}^{N-k-1}-1)}{(\bar{r}-1)(\bar{r}^N-1)},
\end{aligned}$$

where $\bar{r} = 1/r$ and the final equality can be shown by induction on k. Combining the above we see

$$\begin{aligned}
\mathbb{E}_x[Z_k; A^c] &\leq \mathbb{E}_x[Z_k|A^c] \\
&= 1 + \mathbb{E}_x[\bar{\zeta}_{k-1}] + \mathbb{E}_x[\bar{\zeta}_k] \\
&= \frac{(\bar{r}-1)(\bar{r}^N-1) + \bar{r}(\bar{r}^{k-1}-1)(\bar{r}^{N-k}-1) + \bar{r}(\bar{r}^k-1)(\bar{r}^{N-k-1}-1)}{(\bar{r}-1)(\bar{r}^N-1)} \\
&= \frac{\bar{r}+1}{\bar{r}-1}\frac{(\bar{r}^{N-k}-1)(\bar{r}^k-1)}{\bar{r}^N-1} \\
&\leq \frac{\bar{r}+1}{\bar{r}-1} = \frac{1+r}{1-r},
\end{aligned}$$

where the inequality follows from the fact that $\bar{r} = \frac{1}{r} > 1$. Furthermore, for all $x < k < N$, we have,

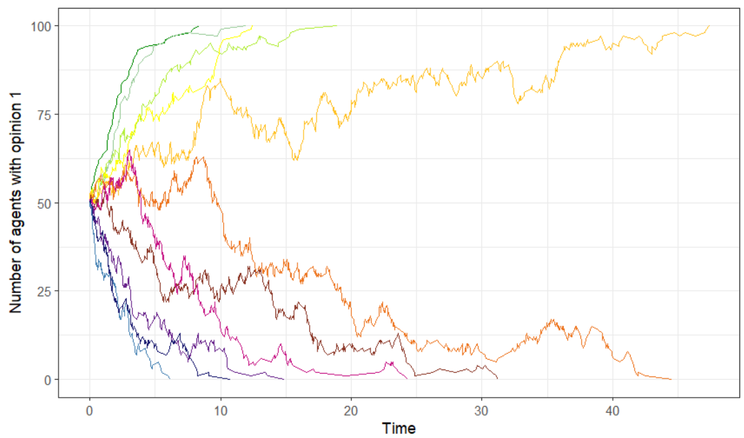


Figure 4: Simulations of the voter model for varying levels of bias. Parameters: $N = 100$, $\alpha = 0.5$, and $q_0 = 0.5$. Simulations run for $q_1 \in \{0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1\}$.

$$\begin{aligned}\mathbb{E}_x[Z_k; A^c] &= \mathbb{E}_x[Z_k; T_k < T_0 < T_N] \\ &= \mathbb{E}_x[Z_k | T_k < T_0 < T_N]\mathbb{P}_x(T_k < T_0 < T_N) \\ &= \mathbb{E}_k[Z_k | T_0 < T_N]\mathbb{P}_x(T_k < T_0 < T_N) \\ &\leq \frac{1+r}{1-r}.\end{aligned}$$

Combining all these findings establishes the upper bound for $\mathbb{E}_{\lfloor \alpha N \rfloor}[Z_k]$,

$$\mathbb{E}_{\lfloor \alpha N \rfloor}[Z_k] \leq \frac{1+r}{1-r},$$

for all $0 < k < N$. Applying this to equation (7) yields the upper bound for $t_n(\alpha)$,

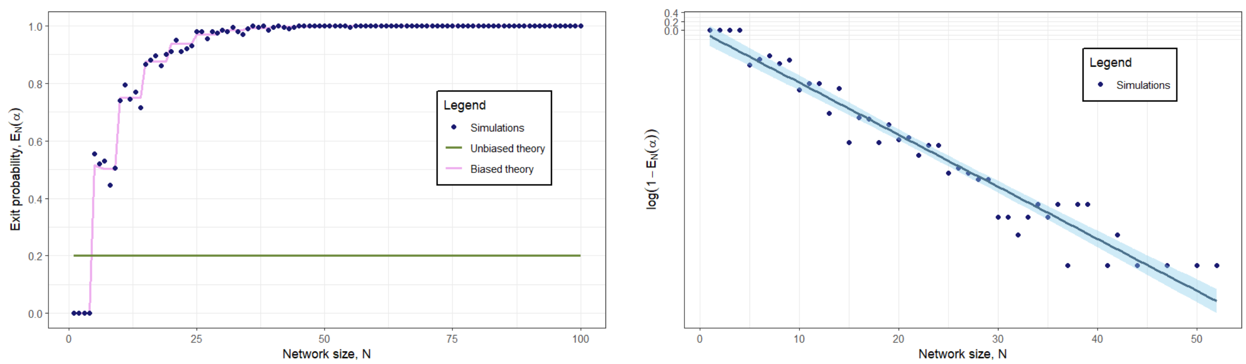
$$\begin{aligned}t_N(\alpha) &\leq \frac{2}{q_0 + q_1}\frac{1+r}{1-r}\sum_{k=1}^{N-1}\frac{1}{k} \\ &\leq \frac{2}{q_0 + q_1}\frac{1+r}{1-r}(\log(N-1) + 1),\end{aligned}$$

which, again, depends on the logarithm of N, thus completing the proof. □

3.3 Simulations

To illustrate the impact of bias on the opinion dynamics of the voter model, simulations were run; the R script for this can be found in Appendix A. In Figure 4, the simulations show the evolution of the opinions of the agents for varying biases. This was done by setting $N = 100$, $\alpha = 0.5$, and $q_0 = 0.5$, then simulating the voter model for different values of q_1. The figure shows that when q_1 is larger than q_0, the network tends to reach consensus on opinion 1, with the higher values of q_1 achieving consensus faster. Conversely, for values of q_1 less than q_0, the network tends to achieve consensus on opinion 0, with the lower values of q_1 achieving consensus faster.

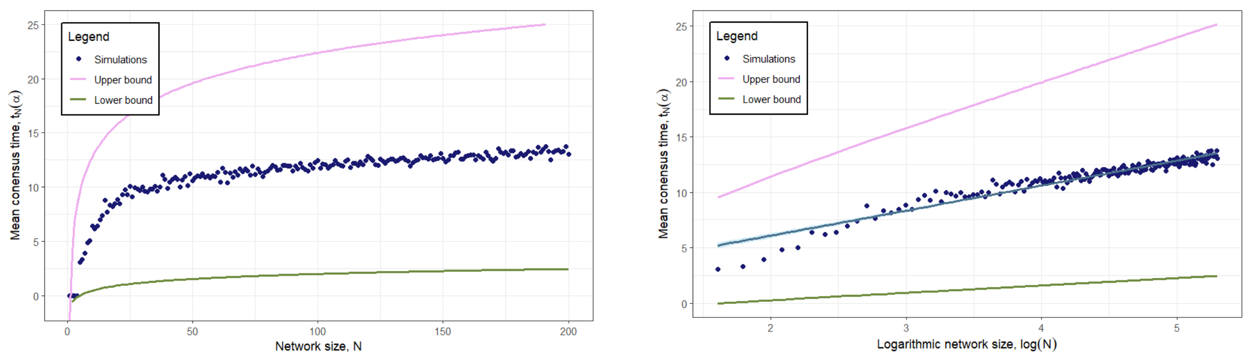
Figure 5 shows the exit probability from 200 simulations for each value of N for the voter model (Figure 5a), as well as a semi-log plot (Figure 5b) to show the convergence is exponential. Figure 5a shows the



(a) Exit probability, $E_N(\alpha)$, as a function of network size, N, under the voter model with biased agents. Parameters: $q_0 = 1$, $q_1 = 0.5$, and $\alpha = 0.2$.

(b) Semi log plot taking $\log(1 - E_N(\alpha))$, as a function of network size, N. Parameters: $q_0 = 1$, $q_1 = 0.5$, and $\alpha = 0.2$.

Figure 5: Simulations for exit probability under the voter model with biased agents.



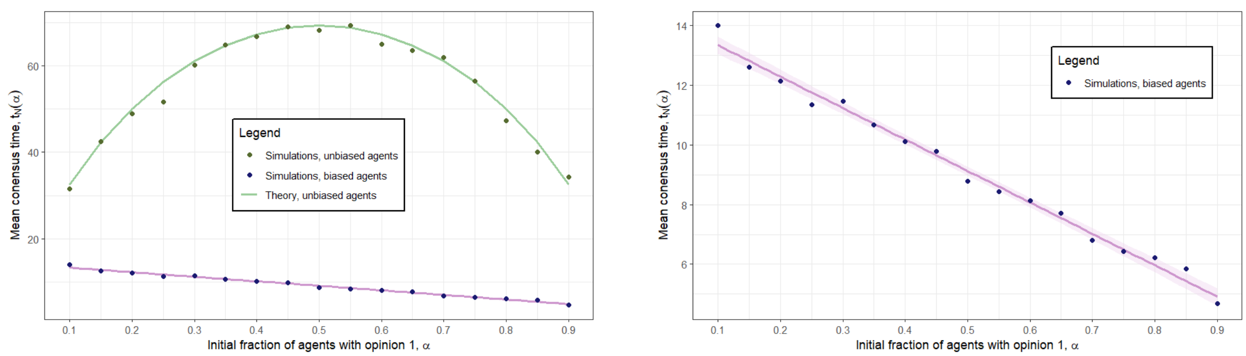
(a) Simulations and theoretical bounds for mean consensus time, $t_N(\alpha)$, as a function of network size, N. Parameters: $q_0 = 1$, $q_1 = 0.5$, and $\alpha = 0.2$.

(b) Simulations and theoretical bounds for mean consensus time, $t_N(\alpha)$, plotted against the logarithm of the network size, $\log(N)$. Parameters: $q_0 = 1$, $q_1 = 0.5$, and $\alpha = 0.2$.

Figure 6: Simulations for mean consensus time for varying values for network size, N, under the voter with biased agents.

results for the exit probability for different values of N, with $q_0 = 1$, $q_1 = 0.5$, and $\alpha = 0.2$. These simulations are plotted along with the theoretical exit probability obtained in Section 3.1, $E_N(\alpha) = \frac{1-r^{\lfloor \alpha N \rfloor}}{1-r^N}$, as well as the theoretical exit probability for the voter model with unbiased agents which equals the initial fraction of agents with the preferred opinion, $\alpha = 0.2$. This figure shows that the simulations align well with the theory that the exit probability increases to 1 as the network size increases. To show that this convergence is exponential in Figure 5b we plot $\log(1 - E_N(\alpha))$ against network size N, where $E_N(\alpha)$ here is the exit probability to state N obtained from the simulations with network size N. To study the behaviour of the convergence of the exit probability for $E_N(\alpha)$, we consider the exit probability to state 0; $1 - E_N(\alpha)$. Note that $1 - E_N(\alpha) \leq \exp(-cN)$, suggests that the exit probability to state 0 converges to zero exponentially fast. If the exit probability to state 0 converges exponentially in N, then $E_N(\alpha)$ also converges exponentially in N. This is seen in Figure 5b, in which the semi-log shows a linear relationship in N, meaning that the convergence is indeed exponential.

In Figure 6 the mean consensus time for the voter model is plotted as a function of network size (Figure 6a) and $\log(N)$ (Figure 6b), where 100 simulations are run with $q_0 = 1$, $q_1 = 0.5$, and $\alpha = 0.2$. In Figure 6a, we observe for the simulations a logarithmic relationship of the mean consensus time in N, supporting the claims of Theorem 3.2. Additionally, the lower and upper bounds found in the proof of Theorem 3.2 are plotted,



(a) Mean consensus time from simulations with biased and unbiased agents, as well as the theoretical mean consensus time, $t_N(\alpha)$. Plotted as a function of initial fraction of agents with opinion 1, α. Parameters: $q_0 = 1$, $q_1 = 0.5$ for biased agents, $q_1 = 1$ for unbiased agents, and $N = 100$.

(b) Mean consensus time, $t_N(\alpha)$, from simulations for biased agents as a function of initial fraction of agents with opinion 1, α. Parameters: $q_0 = 1$, $q_1 = 0.5$, and $N = 100$.

Figure 7: Simulations for mean consensus time for varying values for initial fraction of agents with the preferred opinion, α, under the voter with biased agents.

with the simulations falling within these bounds. If the relationship between mean consensus time and network size is indeed linear, then the semi-log plot with mean consensus time plotted against $\log(N)$ would show a linear relationship. This semi-log plot is shown in Figure 6b, demonstrating a linear relationship between the mean consensus time and $\log(N)$. Thus providing evidence that the mean consensus time is logarithmic in network size.

Finally, in Figure 7 the mean consensus time is again plotted, this time as a function of α. Figure 7a shows simulations for biased agents ($q_0 = 1$ and $q_1 = 0.5$) and unbiased agents ($q_0 = 1$ and $q_1 = 1$), along with the theory for the approximation of the mean consensus time for unbiased agents (see Section 3.2). Figure 7b shows only the mean consensus time from the simulations for biased agents. The mean consensus time was obtained from 100 simulations for biased agents and 500 simulations for unbiased agents, with $N = 100$. We see that the simulations for the unbiased agents aligns with the theory for the unbiased case. In the unbiased case, consensus on a specific opinion is reached more quickly when the initial number of agents with this opinion is larger than the number of agents with the opposing opinion. In the case of biased agents, the process achieves consensus on the preferred opinion with high probability. Consequently, increasing the initial proportion of agents with the preferred opinion will always decrease the mean time to reach consensus, as seen clearly in Figure 7b.

4 Majority Rule Model

In this section, we explore the majority rule model with biased agents. The exit probability is shown to have a phase transition, implying that despite the bias towards one opinion, consensus may not be reached on the preferred opinion. Furthermore, the average time to reach consensus is characterised.

Again, the network has N agents and initially each agent adopts one of two opinions, 0 or 1, with $\lfloor \alpha N \rfloor$ agents adopting opinion 1. Each agent independently considers updating their opinion at the points of a Poisson point process with rate 1. When considering an update, an agent with opinion $i \in \{0, 1\}$ decides to update their opinion with probability q_i and retains their opinion with probability $p_i = 1 - q_i$. If the agent decides to retain their opinion, they continue with their held opinion until the next point at which they consider updating their opinion. If they decide to update their opinion they do so according to the majority rule, which dictates that the agent updating their opinion should take a sample of $2K$ agents from the network; $1 \leq K \leq \frac{N}{2}$, they do this uniformly at random. The updating agent should adopt the opinion of the majority of the sampled agents plus themselves. The inclusion of the updating agent themselves ensures that there will be one opinion that is the majority among the $2K+1$ agents.

Let $X^{(N)}(t)$ denote the number of agents with opinion 1 at time $t \geq 0$. As in the Voter model, this is a continuous-time Markov process on state space $\{0, 1, ..., N\}$ with absorbing states 0 and N. The transition rates out of state n are given by

$$\begin{aligned} q(n \to n+1) &= (N-n)q_0 \sum_{i=K+1}^{2K} \binom{2K}{i} \Big(\frac{n}{N}\Big)^i \Big(\frac{N-n}{N}\Big)^{2K-i} \\ &= (N-n)q_0 \mathbb{P}\Big(\text{Bin}\Big(2K, \frac{n}{N}\Big) \geq K+1\Big), \end{aligned} \tag{15}$$

$$\begin{aligned} q(n \to n-1) &= nq_1 \sum_{i=K+1}^{2K} \binom{2K}{i} \Big(\frac{N-n}{N}\Big)^i \Big(\frac{n}{N}\Big)^{2K-i} \\ &= nq_1 \mathbb{P}\Big(\text{Bin}\Big(2K, 1-\frac{n}{N}\Big) \geq K+1\Big). \end{aligned} \tag{16}$$

We get the equation for $q(n \to n+1)$ because to go from state n to $n+1$, an agent with opinion 0 needs to update their opinion to 1. First, an agent with opinion 0 must consider updating their opinion. Since each individual agent considers updating their opinion according to a Poisson process with rate 1, by the Superposition Theorem (2.4) we can combine the $N-n$ agents with opinion 0 into a Poisson process with rate $N-n$. The interarrival times of this combined Poisson process follows an exponential distribution with rate $N-n$. So, at a rate of $N-n$, an agent with opinion 0 will consider updating their opinion. Then this agent with opinion 0 will update their opinion with probability q_0. If the agent decides to update their opinion, they do so according to the majority rule, that is, by selecting a sample of $2K$ agents uniformly at random. Therefore, we are interested in the probability that of the chosen $2K$ agents, $K+1$ agents have opinion 1. Since only when $K+1$ or more agents have opinion 1, will the majority opinion of the $2K+1$ agents, the sampled agents plus the updating agent, be opinion 1. The probability that any given agent has opinion 1 is n/N. This means that the probability that at least $K+1$ of the $2K$ agents have opinion 1 can be written as the probability that a binomial random variable, with $2K$ as the number of trials and n/N as the probability of success, is larger than or equal to $K+1$. With similar reasoning for $q(n \to n-1)$.

Let $\tilde{X}^{(N)}$ be the embedded Markov chain corresponding to $X^{(N)}$. The jump probabilities are,

$$\begin{aligned} p_{n,n+1} &= 1 - p_{n,n-1} \\ &= \frac{q(n \to n+1)}{q(n \to n-1) + q(n \to n+1)} \\ &= \frac{g_K(n/N)}{g_K(n/N) + r}, \end{aligned} \tag{17}$$

where $1 \leq n \leq N-1$, $r = q_1/q_0 < 1$, and $g_K : (0,1) \to (0, \infty)$ is defined as

$$g_K(x) = \frac{\frac{1}{x}\mathbb{P}(\text{Bin}(2K, x) \geq K+1)}{\frac{1}{1-x}\mathbb{P}(\text{Bin}(2K, 1-x) \geq K+1)}. \tag{18}$$

With probability 1, $X^{(N)}$ is absorbed in state 0 or N in finite time, as explained in Section 2.2 the Markov chain is not irreducible and therefore will almost surely be absorbed into one of its essential states in finite time. As with the Voter model, let $E_N(\alpha)$ denote the exit probability and $t_N(\alpha)$ denote the mean consensus time.

In order to show the phase transition of the exit probability and characterise the mean consensus time for the majority model, we will need the following lemma.

Lemma 4.1. The function $g_K : (0,1) \to (0, \infty)$ as defined in equation (18) is strictly increasing and therefore also one-to-one.

Proof. First, say $g_K(x) = \phi(\psi(x))$, with $\psi(x) = \frac{x}{1-x} : [0, 1) \to [0, \infty)$ and

$$\phi(t) = \frac{\sum_{i=K+1}^{2K} \binom{2K}{i} t^i}{\sum_{i=K+1}^{2K} \binom{2K}{i} t^{2K+1-i}} = \frac{\sum_{i=K+1}^{2K} \binom{2K}{i} t^i}{\sum_{i=1}^{K} \binom{2K}{i-1} t^i},$$

where the denominator in the final equality is achieved by noticing that for any $i \in \{K, ..., 2K\}$ or $i \in \{0, ..., K\}$ we have $\binom{2K}{i} = \binom{2K}{2K-i}$ and changing the basis in the sum using $j = 2K + 1 - i$.

Note that $\psi(x)$ is strictly increasing; this can be shown by taking the derivative and seeing that $\psi'(x) > 0$. We have

$$\psi'(x) = \frac{1}{(1-x)^2} > 0.$$

Therefore, we can say that $\psi(x)$ is strictly increasing. To show that g_K is strictly increasing, it is sufficient to show that $\phi(t)$ is also strictly increasing. Taking the derivative of $\phi(t)$ gives

$$\begin{aligned} \phi'(t) &= \frac{\left(\sum_{i=K+1}^{2K} i\binom{2K}{i} t^{i-1}\right)\left(\sum_{i=1}^{K} \binom{2K}{i-1} t^i\right) - \left(\sum_{i=K+1}^{2K} \binom{2K}{i} t^i\right)\left(\sum_{i=1}^{K} i\binom{2K}{i-1} t^{i-1}\right)}{\left(\sum_{i=1}^{K} \binom{2K}{i-1} t^i\right)^2} \qquad (19) \\ &= \frac{\sum_{j=K+1}^{3K-1} M_j t^j}{\left(\sum_{i=1}^{K} \binom{2K}{i-1} t^i\right)^2}, \end{aligned}$$

with

$$M_j = \sum_i \binom{2K}{i-1}\binom{2K}{j-i+1}(j+1-2i),$$

where the values for i are determined by considering the following two cases for j,

1. If $K + 1 \leq j \leq 2K$, then i runs from 1 to $j - K$.
2. If $2K + 1 \leq j \leq 3K - 1$, then i runs from $j + 1 - 2K$ to K.

We derive M_j by looking at the numerator of equation (19) and seeing that the lowest power of t will be t^{K+1} and the highest will be t^{3K-1}. Finding the coefficients of these t^j in equation (19) we see that they are equal to M_j. Observe that t^{K+1} is obtained only with the first sum evaluated at $i = K + 1$, the second at $i = 1$, the third at $i = K + 1$, and the fourth at $i = 0$; this gives

$$(K+1)\binom{2K}{K+1}\binom{2K}{0} t^{K+1} - \binom{2K}{K+1}\binom{2K}{0} t^{K+1} = K\binom{2K}{K+1}\binom{2K}{0} t^{K+1}.$$

If we evaluate M_j at $j = K+1$, we get $\binom{2K}{0}\binom{2K}{K+1}K$. We continue and find this holds for all $j = K+1$ until $j = 2K$. With the coefficient of t^{2K} from equation (19) being

$$\binom{2K}{2K}\binom{2K}{0}(2K-1) + \cdots + \binom{2K}{K+1}\binom{2K}{K-1} = M_{2K}.$$

We consider coefficients for t^{2K+1} and find,

$$(2K-2)\binom{2K}{2K}\binom{2K}{2} + \cdots + 2\binom{2K}{K+2}\binom{2K}{K-1} = M_{2K+1}.$$

Continuing this until we get the coefficient for t^{3K-1},

$$K\binom{2K}{2K}\binom{2K}{K-1} = M_{3K-1}.$$

The number of terms in the sum M_j are increasing for $j = 1$ to $2K$, when the sum M_j has the most terms, namely K. Then from $j = 2K$ the number of terms decreases again to one term for $3K-1$. This is because the number of ways to obtain the coefficient for t^j from the multiplication of two sums will increase until the midway point between the lowest, t^{K+1}, and the highest, t^{3K-1}, powers of t; this midway point being $2K$. Therefore, we see the summation for the coefficients M_j run from 1 to $j-K$ for $K+1 \leq j \leq 2K$ and from $j+1-2K$ to K for $2K+1 \leq j \leq 3K-1$.

If $M_j > 0$, then $\phi'(t) > 0$, proving that $\phi(t)$ is strictly increasing. Due to the binomial coefficients by definition being greater than zero, it remains to show $j+1-2i > 0$, reformulated this is $i < \frac{j+1}{2}$. Note that we have the following.

1. For all $K+1 \leq j \leq 2K$, i runs to $j-K$.
2. For all $2K+1 \leq j \leq 3K-1$, i runs to K.

This means that when $K+1 \leq j \leq 2K$, then $i \leq j-K$. We can rewrite $i \leq j-K$ as $-2i \geq 2K-2j$, which in turn gives

$$\begin{aligned} j+1-2i &\geq j+1+2K-2j \\ &= 2K+1-j \\ &\overset{\text{(a)}}{\geq} 2K+1-2K \\ &= 1 > 0, \end{aligned}$$

where inequality (a) comes from the fact that $j \leq 2K$ implies that $-j \geq -2K$. Similarly, for all $2K+1 \leq j \leq 3K-1$, we have $i \leq K$, giving

$$\begin{aligned} j+1-2i &\geq j+1-2K \\ &\geq 2K+1+1-2K \\ &= 2 > 0. \end{aligned}$$

Therefore, $M_j > 0$ for all $K+1 \leq j \leq 3K-1$ and any $K \geq 1$, proving that $\phi(t)$ is strictly increasing, as required. □

4.1 Exit Probability with Phase Transition

The following theorem describes the exit probability to state N, $E_N(\alpha)$, as well as showing that the convergence of the exit probability has a phase transition and delineating the boundary for this phase transition.

Theorem 4.2. Let $E_N(n)$ denote the probability that the process $X^{(N)}$ is absorbed in state N when starting from state n.

1. Then, we have
$$E_N(n) = \frac{\sum_{t=0}^{n-1}\prod_{j=1}^{t}\frac{r}{g_K(j/N)}}{\sum_{t=0}^{N-1}\prod_{j=1}^{t}\frac{r}{g_K(j/N)}}, \tag{20}$$
2. Define $E_N(\alpha) := E_N(\lfloor \alpha N \rfloor)$ and $\beta := g_K^{-1}(r)$, $\beta \in (0,1)$. Then $E_N(\alpha) \to 1$ (respectively $E_N(\alpha) \to 0$) as $N \to \infty$ if $\alpha > \beta$ (respectively $\alpha < \beta$) and this convergence is exponential in N.

The phase transition for the exit probability occurs at $\beta = g_K^{-1}(r)$ for all $K \geq 1$.

Remark. In the case of unbiased agents, we have $r = 1$, giving $\beta = g_K^{-1}(1) = 1/2$, which is consistent with the results of previous studies on the majority rule model with unbiased agents [3].

Proof. First, we prove the expression for $E_N(n)$. Looking at the embedded Markov chain, we can rewrite $E_N(n)$ as follows,

$$E_N(n) = p_{n,n+1}E_N(n+1) + p_{n,n-1}E_N(n-1),$$

which can be rearranged to

$$E_N(n+1) - E_N(n) = \frac{p_{n,n-1}}{p_{n,n+1}}(E_N(n) - E_N(n-1)).$$

Let $D_N(n) := E_N(n+1) - E_N(n)$ and substituting this in the above equation yields a recursion in $D_N(n)$, for all $1 \leq n \leq N-1$,

$$\begin{aligned} D_N(n) &= \frac{p_{n,n-1}}{p_{n,n+1}} D_N(n-1) \\ &= \frac{\frac{r}{g_K(n/N)+r}}{\frac{g_K(n/N)}{g_K(n/N)+r}} D_N(n-1) \\ &= \frac{r}{g_K(n/N)} D_N(n-1). \end{aligned} \tag{21}$$

We have the boundary conditions $E_N(0) = 0$ and $E_N(N) = 1$, which implies,

$$\sum_{n=0}^{N-1} D_N(n) = \sum_{n=0}^{N-1}(E_N(n+1) - E_N(n)) = E_N(N) - E_N(0) = 1.$$

To compute $D_N(0)$,

$$\begin{aligned} D_N(t) &= \frac{r}{g_K(t/N)} D_N(t-1) \\ &= \frac{r}{g_K(t/N)} \frac{r}{g_K(\frac{t-1}{N})} D_N(t-2) \\ &= \dots = \prod_{j=1}^{t} \frac{r}{g_K(j/N)} D_N(0). \end{aligned}$$

Summing both sides over t from 0 to $N-1$ gives,

$$1 = \sum_{t=0}^{N-1} D_N(t) = D_N(0) \sum_{t=0}^{N-1} \prod_{j=1}^{t} \frac{r}{g_K(j/N)},$$

which upon rearranging gives,

$$D_N(0) = \frac{1}{\sum_{t=0}^{N-1} \prod_{j=1}^{t} \frac{r}{g_K(j/N)}}. \tag{22}$$

Using equation (21) and equation (22) we find $E_N(n)$,

$$E_N(n) = \sum_{k=0}^{n-1} D_N(k) = \sum_{k=0}^{n-1} \prod_{j=1}^{k} \frac{r}{g_K(j/N)} D_N(0) = \frac{\sum_{t=0}^{n-1} \prod_{j=1}^{t} \frac{r}{g_K(j/N)}}{\sum_{t=0}^{N-1} \prod_{j=1}^{t} \frac{r}{g_K(j/N)}},$$

the required expression for $E_N(n)$ for all $0 \leq n \leq N$.

In the second part of Theorem 4.2 the convergence of the exit probability with phase transition is shown. Note that D_N defines a probability distribution on state space $\{0, 1, ..., N-1\}$, since $D_N(n)$ assigns probabilities to the set $\{0, 1, ..., N-1\}$ and $\sum_{t=0}^{N-1} D_N(n) = 1$. We have the following relationships;

$$
\begin{aligned}
&D_N(n) < D_N(n-1) \text{ for all } n \geq \lfloor \beta N \rfloor + 1, \\
&D_N(n) > D_N(n-1) \text{ for all } n \leq \lfloor \beta N \rfloor.
\end{aligned}
$$

This comes from the monotonicity of g_K shown in Lemma 4.1 and equation (21). For $D_N(n) < D_N(n-1)$ we need $\frac{r}{g_K(n/N)} < 1$, which holds when $g^{-1}(r) = \beta < n/N$, that is $\beta N < n$. And for $D_N(n) > D_N(n-1)$ we need $\frac{r}{g_K(n/N)} > 1$, which holds when $\beta N > n$. Since βN does not need to be an integer, we take the floor function, resulting in the above conditions. From this we can deduce that the mode of the distribution of D_N is at $\lfloor \beta N \rfloor$.

First, we consider $\alpha > \beta$ and show that the exit probability, $E_N(\alpha)$, converges to 1. For any value $\alpha > \beta$, we can choose a $\beta' \in \mathbb{R}$ such that $\alpha > \beta' > \beta$, we will use this β' to find a lower bound for $E_N(\alpha)$. By the monotonicity of g_K,

$$
r' := \frac{r}{g_K(\beta')} < \frac{r}{g_K(\beta)} = 1.
$$

Moreover, from the monotonicity of g_K and equation (21) we have for any $j \geq 1$,

$$
\begin{aligned}
D_N(\lfloor \beta' N \rfloor + j) &= \frac{r}{g_K(\frac{\lfloor \beta' N \rfloor + j}{N})} D_N(\lfloor \beta' N \rfloor + j - 1) \\
&= \prod_{l=1}^{j} \frac{r}{g_K(\frac{\lfloor \beta' N \rfloor + l}{N})} D_N(\lfloor \beta' N \rfloor) \\
&\leq \left(\frac{r}{g_K(\frac{\lfloor \beta' N \rfloor + 1}{N})} \right)^j D_N(\lfloor \beta' N \rfloor) \\
&\leq \left(\frac{r}{g_K(\beta')} \right)^j D_N(\lfloor \beta' N \rfloor) \qquad \text{since } \beta' N < \lfloor \beta' N \rfloor + 1 \\
&= (r')^j D_N(\lfloor \beta' N \rfloor).
\end{aligned} \tag{23}
$$

To find a lower bound for $E_N(\alpha)$, we write,

$$
\begin{aligned}
E_N(\alpha) &= \sum_{t=1}^{\lfloor \alpha N \rfloor - 1} D_N(t) \\
&= 1 - \sum_{t=\lfloor \alpha N \rfloor}^{N-1} D_N(t),
\end{aligned}
$$

and notice that

$$
\begin{aligned}
\sum_{t=\lfloor \alpha N \rfloor}^{N-1} D_N(t) &= \sum_{l=0}^{N-1-\lfloor \alpha N \rfloor} D_N(l + \lfloor \alpha N \rfloor) \\
&= \sum_{l=0}^{N-1-\lfloor \alpha N \rfloor} D_N(\lfloor \beta' N \rfloor + l + \lfloor \alpha N \rfloor - \lfloor \beta' N \rfloor) \\
&\leq \sum_{l=0}^{N-1-\lfloor \alpha N \rfloor} (r')^{l + \lfloor \alpha N \rfloor - \lfloor \beta' N \rfloor} D_N(\lfloor \beta' N \rfloor),
\end{aligned}
$$

where the inequality follows from equation (23). This then gives

$$E_N(\alpha) \geq 1 - D_N(\lfloor \beta' N \rfloor)(r')^{\lfloor \alpha N \rfloor - \lfloor \beta' N \rfloor} \sum_{t=0}^{N-1-\lfloor \alpha N \rfloor} (r')^t.$$

For any t, we have $D_N(t) < 1$, so

$$\begin{aligned} E_N(\alpha) &\geq 1 - (r')^{\lfloor \alpha N \rfloor - \lfloor \beta' N \rfloor} \frac{1 - (r')^{N - \lfloor \alpha N \rfloor}}{1 - r'} \\ &\to 1 \text{ as } N \to \infty. \end{aligned}$$

It can be seen that the expression converges to one, because $(r')^{N-\lfloor \alpha N \rfloor} < (r')^{(1-\alpha)N} \to 0$ and $(r')^{\lfloor \alpha N \rfloor - \lfloor \beta' N \rfloor} < (r')^{(\alpha - \beta')N - 1} \to 0$ as $N \to \infty$, since $1 - \alpha > 0$ and $\alpha - \beta' > 0$. And the convergence is exponential in N.

Assume $\alpha < \beta$. For any $\alpha < \beta$, we can choose a $\beta' \in \mathbb{R}$ such that $\alpha < \beta' < \beta$. Since g_K is monotone, we have

$$r' = \frac{r}{g_K(\beta')} > \frac{r}{g_K(\beta)} = 1.$$

We find an upper bound for $E_N(\alpha)$ in order to prove convergence to zero. Rewriting equation (21), along with the monotonicity of g_K gives, for any $1 \leq j \leq \lfloor \beta' N \rfloor - 1$,

$$\begin{aligned} D_N(\lfloor \beta' N \rfloor - j) &= \frac{g_K\left(\frac{\lfloor \beta' N \rfloor - j + 1}{N}\right)}{r} D_N(\lfloor \beta' N \rfloor - j + 1) \\ &= \prod_{l=0}^{j-1} \frac{g_K\left(\frac{\lfloor \beta' N \rfloor - l}{N}\right)}{r} D_N(\lfloor \beta' N \rfloor) \\ &\leq \left(\frac{g_K\left(\frac{\lfloor \beta' N \rfloor}{N}\right)}{r}\right)^{j-1} D_N(\lfloor \beta' N \rfloor) \\ &\leq \left(\frac{g_K(\beta')}{r}\right)^{j-1} D_N(\lfloor \beta' N \rfloor) \\ &= \left(\frac{1}{r'}\right)^{j-1} D_N(\lfloor \beta' N \rfloor). \end{aligned}$$

Using this, we find the upper bound on $E_N(\alpha)$ as follows.

$$\begin{aligned} E_N(\alpha) &= \sum_{t=1}^{\lfloor \alpha N \rfloor - 1} D_N(t) \\ &= \sum_{t=1}^{\lfloor \alpha N \rfloor - 1} D_N(\lfloor \alpha N \rfloor - t) \\ &= \sum_{t=1}^{\lfloor \alpha N \rfloor - 1} D_N(\lfloor \beta' N \rfloor - (\lfloor \beta' N \rfloor - \lfloor \alpha N \rfloor + t)) \\ &\leq \sum_{t=1}^{\lfloor \alpha N \rfloor - 1} \left(\frac{1}{r'}\right)^{\lfloor \beta' N \rfloor - \lfloor \alpha N \rfloor + t - 1} D_N(\lfloor \beta' N \rfloor) \end{aligned}$$

$$
\begin{aligned}
&= \left(\frac{1}{r'}\right)^{\lfloor \beta' N \rfloor - \lfloor \alpha N \rfloor - 1} D_N(\lfloor \beta' N \rfloor) \sum_{t=1}^{\lfloor \alpha N \rfloor - 1} \left(\frac{1}{r'}\right)^{t} \\
&\leq \left(\frac{1}{r'}\right)^{\lfloor \beta' N \rfloor - \lfloor \alpha N \rfloor} \frac{1-\left(\frac{1}{r'}\right)^{\lfloor \alpha N \rfloor - 1}}{1-\left(\frac{1}{r'}\right)} \\
&\to 0 \text{ as } N \to \infty,
\end{aligned}
$$

where the convergence to zero is due to the fact that $\left(\frac{1}{r'}\right)^{\lfloor \beta' N \rfloor - \lfloor \alpha N \rfloor} < \left(\frac{1}{r'}\right)^{(\beta'-\alpha)N-1} \to 0$ as $N \to \infty$, since $\lfloor \beta' N \rfloor - \lfloor \alpha N \rfloor > \beta N - \alpha N - 1$ and $\beta' - \alpha > 0$, and $\left(\frac{1}{r'}\right)^{\lfloor \alpha N \rfloor} < \left(\frac{1}{r'}\right)^{\alpha N - 1} \to 0$ as $N \to \infty$, since $\alpha N - 1 < \lfloor \alpha N \rfloor$ and $\alpha > 0$. Moreover, the convergence is also exponential in N. ☐

Thus, we have demonstrated that a phase transition occurs in the exit probability at the point $\beta = g_K^{-1}(r)$ for any $K \geq 1$, where $r = q_1/q_0 < 1$. When the initial proportion of agents holding the preferred opinion surpasses the threshold value β, the system reaches consensus on this preferred opinion with high probability. Conversely, if the initial proportion of these agents is below this threshold β, the system may not necessarily converge to consensus on the preferred opinion. This contrasts with the dynamics observed in the voter model, where consensus is obtained on the preferred opinion regardless of the original distribution of opinions within the system.

4.2 Mean Consensus Time

In the following theorem, the mean consensus time, $t_N(\alpha)$, is characterised for a network which initially has $\lfloor \alpha N \rfloor$ agents with opinion 1. Again, this shows that the mean consensus time is logarithmic in N.

Theorem 4.3. For $\alpha \in (0,\beta) \cup (\beta, 1)$, we have $t_N(\alpha) = \Theta(\log N)$.

Remark. The mean consensus time for the unbiased majority rule model is also logarithmic in N [3].

Proof. First, let us recap the beginnings of the proof of Theorem 3.2. We define T_n as the first hitting time that the process reaches the state n. Formally, $T_n := \inf\{t \geq 0 : X^{(N)}(t) = n\}$. Furthermore, designate the variable T as the minimum of two first hitting times, $T = T_0 \wedge T_N$; T is the random time it takes to achieve consensus on the network. Moreover,

$$
T = \sum_{n=1}^{N-1} \sum_{j=1}^{Z_n} M_{n,j},
$$

with Z_n denoting the number of visits to state n prior to absorption, and $M_{n,j}$ the duration of the j^{th} visit to state n. The random variables Z_n and $(M_{n,j})_{j\geq 1}$ are independent, where $M_{n,j}$ follows an exponential distribution with a rate $q(n \to n+1) + q(n \to n-1)$. Applying Wald's identity gives

$$
\begin{aligned}
t_N(\alpha) &= \mathbb{E}_{\lfloor \alpha N \rfloor}[T] \\
&= \sum_{n=1}^{N-1} \mathbb{E}_{\lfloor \alpha N \rfloor}[Z_n] \mathbb{E}_{\lfloor \alpha N \rfloor}[M_{n,j}] \\
&= \sum_{k=1}^{N-1} \frac{\mathbb{E}_{\lfloor \alpha N \rfloor}[Z_k]}{q(n \to n+1) + q(n \to n-1)}.
\end{aligned} \tag{24}
$$

We determine the lower and upper bounds for $t_N(\alpha)$. Define $A := \{\omega : T_N(\omega) < T_0(\omega)\}$ as the event in which the Markov chain is absorbed at state N. We have

$$
\mathbb{E}_{\lfloor \alpha N \rfloor}[Z_n] = \mathbb{E}_{\lfloor \alpha N \rfloor}[Z_n|A]\mathbb{P}_{\lfloor \alpha N \rfloor}(A) + \mathbb{E}_{\lfloor \alpha N \rfloor}[Z_n|A^c](1 - \mathbb{P}_{\lfloor \alpha N \rfloor}(A)).
$$

Lower bound of $t_N(\alpha)$. By applying Markov's inequality (Lemma 2.1) to equations (15) and (16), we obtain

$$\begin{aligned} q(n \to n+1) + q(n \to n-1) &\le (N-n)q_0 \frac{2K\frac{n}{N}}{K+1} + nq_1 \frac{2K(1-\frac{n}{N})}{K+1} \\ &= (q_0+q_1)\frac{2K}{K+1}\frac{n(N-1)}{N}. \end{aligned}$$

As in equation (9) of the voter model, we again have

$$\mathbb{E}_{\lfloor \alpha N \rfloor}[Z_n] \ge \mathbb{P}_{\lfloor \alpha N \rfloor}(A)\mathbb{1}_{\{n \ge \lfloor \alpha N \rfloor\}} + (1-\mathbb{P}_{\lfloor \alpha N \rfloor}(A))\mathbb{1}_{\{n \le \lfloor \alpha N \rfloor\}}.$$

Combining the above inequalities with equation (24) gives

$$\begin{aligned} t_N(\alpha) &\ge \frac{N(K+1)}{2K(q_0+q_1)} \sum_{n=1}^{N-1} \frac{1}{n(N-n)} \Big[\mathbb{P}_{\lfloor \alpha N \rfloor}(A)\mathbb{1}_{\{n \ge \lfloor \alpha N \rfloor\}} + (1-\mathbb{P}_{\lfloor \alpha N \rfloor}(A))\mathbb{1}_{\{n \le \lfloor \alpha N \rfloor\}}\Big] \\ &= \frac{N(K+1)}{2K(q_0+q_1)}\mathbb{P}_{\lfloor \alpha N \rfloor}(A) \sum_{n=\lfloor \alpha N \rfloor}^{N-1} \left(\frac{1}{n} + \frac{1}{N-n}\right) \\ &\quad + \frac{N(K+1)}{2K(q_0+q_1)}(1-\mathbb{P}_{\lfloor \alpha N \rfloor}(A)) \sum_{n=1}^{\lfloor \alpha N \rfloor} \left(\frac{1}{n} + \frac{1}{N-n}\right). \end{aligned}$$

To simplify the expression of this lower bound, we consider that the two summations have $N-\lfloor \alpha N \rfloor$ and $\lfloor \alpha N \rfloor$ terms, respectively. Taking the minimum of these values gives $\lfloor N\alpha \rfloor \wedge (N - \lfloor N\alpha \rfloor) > (N\alpha - 1) \wedge N(1-\alpha) = N((\alpha - 1/N) \wedge (1-\alpha))$. For sufficiently large N, this can be approximated by $N(\alpha \wedge (1-\alpha))$. Summing from 1 to this minimum yields a reduced lower bound, allowing cancellation of $\mathbb{P}_{\lfloor \alpha N \rfloor}(A)$ terms. Dividing by N provides a simpler lower bound. Thus, we have

$$\begin{aligned} t_N(\alpha) &\ge \frac{1}{q_0+q_1}\frac{K+1}{2K} \sum_{k=1}^{N(\alpha \wedge (1-\alpha))} \frac{1}{k} \\ &> \frac{1}{q_0+q_1}\frac{K+1}{2K} \log(N(\alpha \wedge (1-\alpha))). \end{aligned}$$

Note that this expression is similar to that found in the voter model, with the addition of the term $(K+1)/2K$.

Upper bound for $t_N(\alpha)$. When $\frac{n}{N} \le \frac{1}{2}$, then

$$\begin{aligned} \mathbb{P}\Big(\text{Bin}\Big(2K, \frac{n}{N}\Big) \ge K+1\Big) &\le \mathbb{P}\Big(\text{Bin}\Big(2K, \frac{1}{2}\Big) \ge K+1\Big) \\ &\le \mathbb{P}\Big(\text{Bin}\Big(2K, 1-\frac{n}{N}\Big) \ge K+1\Big). \end{aligned}$$

Conversely, when $\frac{n}{N} > \frac{1}{2}$, we have

$$\begin{aligned} \mathbb{P}\Big(\text{Bin}\Big(2K, 1-\frac{n}{N}\Big) \ge K+1\Big) &\le \mathbb{P}\Big(\text{Bin}\Big(2K, \frac{1}{2}\Big) \ge K+1\Big) \\ &\le \mathbb{P}\Big(\text{Bin}\Big(2K, \frac{n}{N}\Big) \ge K+1\Big). \end{aligned}$$

This is because a higher probability of success of the binomial random variable corresponds to a higher number of agents in the network holding opinion 1, meaning there is a higher probability that at least $K+1$

of the chosen sample of $2K$ agents has opinion 1. Moreover, since $0 < q_1 < q_0 < 1$, from equations (15) and (16) we have

$$q(n \to n+1) + q(n \to n-1) \geq q_1 \Big[(N-n)\mathbb{P}(\text{Bin}(2K, \frac{n}{N}) \geq K+1) + n\mathbb{P}(\text{Bin}(2K, 1-\frac{n}{N}) \geq K+1) \Big].$$

Observe that for $\frac{n}{N} \leq \frac{1}{2}$,

$$\begin{aligned} q(n \to n+1) + q(n \to n-1) &\geq q_1 n\mathbb{P}(\text{Bin}(2K, 1-\frac{n}{N}) \geq K+1) \\ &\geq q_1 n\mathbb{P}(\text{Bin}(2K, \frac{1}{2}) \geq K+1), \end{aligned}$$

and for $\frac{n}{N} > \frac{1}{2}$,

$$\begin{aligned} q(n \to n+1) + q(n \to n-1) &\geq q_1 (N-n)\mathbb{P}(\text{Bin}(2K, \frac{n}{N}) \geq K+1) \\ &\geq q_1 (N-n)\mathbb{P}(\text{Bin}(2K, \frac{1}{2}) \geq K+1). \end{aligned}$$

Putting this together gives

$$q(n \to n+1) + q(n \to n-1) \geq \begin{cases} cn, & \text{for } \frac{n}{N} \leq \frac{1}{2}, \\ c(N-n), & \text{for } \frac{n}{N} > \frac{1}{2}, \end{cases} \tag{25}$$

where $c = q_1 \mathbb{P}(\text{Bin}(2K, \frac{1}{2}) \geq K+1)$. Using the inequalities of equation (25) with equation (24) gives,

$$t_N(\alpha) \leq \sum_{n=1}^{\lfloor N/2 \rfloor} \frac{\mathbb{E}_{\lfloor \alpha N \rfloor}[Z_n]}{cn} + \sum_{n=\lfloor N/2 \rfloor + 1}^{N-1} \frac{\mathbb{E}_{\lfloor \alpha N \rfloor}[Z_n]}{c(N-n)}. \tag{26}$$

From the above equation, we observe that if $\mathbb{E}_{\lfloor \alpha N \rfloor}[Z_n]$ is bounded from above by a constant for N large enough, then $t_N(\alpha)$ is bounded from above by a function logarithmic in N for large enough N. Therefore, in order to show $t_N(\alpha) = \Theta(\log N)$ it is sufficient to prove $\mathbb{E}_{\lfloor \alpha N \rfloor}[Z_n] = O(1)$ for all $1 \leq n \leq N-1$.

Let $x = \lfloor \alpha N \rfloor$. First, assume $\alpha > \beta$. Additionally, assume event A, i.e. absorption in state N. We aim to find an upper bound for $\mathbb{E}_x[Z_n; A]$. Conditioned on A, the embedded process $\tilde{X}^(N)$ is a Markov chain characterised by the transition probabilities

$$p^A_{n,n+1} = 1 - p^A_{n,n-1} = p_{n,n+1} \frac{\mathbb{P}_{n+1}(A)}{\mathbb{P}_n(A)}. \tag{27}$$

Moreover, using an argument similar to that used for equation (10) we see that $p^A_{n,n-1}$ can be expressed as

$$\begin{aligned} p^A_{n,n-1} &= \mathbb{P}(\tilde{X}_{t+1} = n-1 | A, \tilde{X}_t = n) \\ &= p_{n,n-1} \frac{\mathbb{P}_{n-1}(A)}{\mathbb{P}_n(A)}. \end{aligned} \tag{28}$$

Conditioned on event A occurring, let ζ_n denote the number of transitions made by the embedded chain $\tilde{X}^{(N)}$ from state n to $n-1$ prior to absorption. Therefore, we have

$$Z_n \mid A = \begin{cases} 1 + \zeta_n + \zeta_{n+1}, & \text{for } x \leq n \leq N-1, \\ \zeta_n + \zeta_{n+1}, & \text{for } 1 \leq n < x. \end{cases} \tag{29}$$

Note that $\zeta_N = 0$ and let $\xi_{l,n}$ denote the random number of left-jumps from state n between the l^{th} and $(l+1)^{th}$ left-jumps from state $n+1$. Similar to the voter model, the sequence $(\xi_{l,n})_{l \geq 0}$ consists of i.i.d. random variables with a geometric distribution from zero, having a mean of $p^A_{n,n-1}/p^A_{n,n+1}$. Using equation (17) and Theorem 4.2 we derive

$$\begin{aligned}\frac{p^A_{n,n-1}}{p^A_{n,n+1}} &= \frac{p_{n,n-1}}{p_{n,n+1}}\frac{\mathbb{P}_{n-1}(A)}{\mathbb{P}_{n+1}(A)}, && \text{from equations (27) and (28)}\\ &= \frac{r}{g_K\left(\frac{n}{N}\right)}\frac{\sum_{t=0}^{n-2}\prod_{j=1}^{t}\frac{r}{g_K\left(\frac{j}{N}\right)}}{\sum_{t=0}^{n}\prod_{j=1}^{t}\frac{r}{g_K\left(\frac{j}{N}\right)}}. && (30)\end{aligned}$$

A monotonically non-increasing non-negative sequence $(y_n)_{n\geq 1}$ satisfies the following inequality

$$y_n\frac{\sum_{t=0}^{n-2}\prod_{j=1}^{t}y_j}{\sum_{t=0}^{n}\prod_{j=1}^{t}y_j} \leq 1. \tag{31}$$

Since $(y_n)_{n\geq 1}$ is non-increasing, $y_k \geq y_{k+1}$ for all $k \geq 1$. Expanding equation (31)

$$\frac{y_n + y_1y_n + y_1y_2y_n + \cdots + y_1y_2 \cdot y_{n-2}y_n}{1 + y_1 + y_1y_2 + y_1y_2y_3 + \cdots + y_1y_2\cdots y_{n-1} + y_1y_2\cdots y_n},$$

and comparing the terms in the numerator with the middle $n-1$ terms of the denominator, remembering that all y_k are non-negative, we see

$$\begin{aligned} y_1y_n &\leq y_1y_2,\\ y_1y_2y_n &\leq y_1y_2y_3,\\ &\vdots\\ y_1y_2\cdots y_{n-2}y_n &\leq y_1y_2\cdots y_{n-2}y_{n-1}.\end{aligned}$$

Therefore,

$$y_n\sum_{t=0}^{n-2}\prod_{j=1}^{t}y_j \leq \sum_{t=0}^{n}\prod_{j=1}^{t}y_j,$$

giving the inequality (31). The sequence $(r/g_K(\frac{n}{N}))_{n\geq 1}$ is a monotonically non-increasing non-negative sequence; as such, we can apply inequality (31) to (30). This means that equation (30) can be bounded from above by 1. Moreover, observe that $\mathbb{P}_{n-1}(A) \leq \mathbb{P}_{n+1}(A)$; intuitively, the closer we are to state N, the higher the probability that the Markov chain will be absorbed into state N. This suggests that for $r/g_K(\frac{n}{N}) < 1$ we can create a further upper bound for equation (30). Putting this together, we obtain the following bound on $p^A_{n,n-1}/p^A_{n,n+1}$

$$\frac{p^A_{n,n-1}}{p^A_{n,n+1}} \leq \min\left(1, \frac{r}{g_K\left(\frac{n}{N}\right)}\right). \tag{32}$$

As with the voter model, we notice the recursion

$$\zeta_n = \begin{cases}\sum_{l=0}^{\zeta_n+1}\xi_{l,n}, & \text{for } x \leq n \leq N-1,\\ \sum_{l=1}^{\zeta_n+1}\xi_{l,n}, & \text{for } 1 \leq n < x.\end{cases} \tag{33}$$

So, we apply Wald's identity, with the same argumentation as seen in the proof for Theorem 3.2 for any $x \leq n \leq N-1$ and a similar argument for any $1 \leq n < x$, giving

$$\mathbb{E}_x[\zeta_n] = \begin{cases} \sum_{t=n}^{N-1} \prod_{i=n}^{t} \frac{p^A_{i,i-1}}{p^A_{i,i+1}}, & \text{for } x \le n \le N-1 \\ \left(\prod_{i=n}^{x-1} \frac{p^A_{i,i-1}}{p^A_{i,i+1}}\right) \mathbb{E}_x[\zeta_x], & \text{for } 1 \le n < x. \end{cases} \tag{34}$$

From our assumption that $\alpha > \beta$ we have that $x = \lfloor \alpha N \rfloor > \lfloor \beta N \rfloor$. Using this and the monotonicity of g_K, it holds that $1 > r_\alpha := r/g_K(\alpha) \ge r/g_K(n/N)$ for all $n \ge x$, remembering that $\beta := g_K^{-1}(r)$. Therefore, using inequality (32) see that $p^A_{n,n-1}/p^A_{n,n+1} \le r_\alpha$ and thus for all $n \ge x$,

$$\mathbb{E}_x[\zeta_n] \le \sum_{t=n}^{N-1} \prod_{i=n}^{t} r_\alpha = \sum_{t=1}^{N-n} \prod_{i=1}^{t} r_\alpha = r_\alpha + r_\alpha^2 + \cdots + r_\alpha^{N-n}.$$

The infinite sum $\sum_{n=1}^{\infty} r_\alpha^n = \frac{r_\alpha}{1-r_\alpha}$, as $r_\alpha < 1$. Clearly, the finite sum is bounded from above by the infinite sum, giving

$$\mathbb{E}_x[\zeta_n] \le \frac{r_\alpha}{1-r_\alpha}. \tag{35}$$

Thus, by taking the expectation of (29), it follows that for all $n \ge x$

$$\begin{aligned} \mathbb{E}_x[Z_n; A] &\le \mathbb{E}_x[Z_n | A] \\ &= 1 + \mathbb{E}_x[\zeta_n] + \mathbb{E}_x[\zeta_{n+1}] \\ &\le 1 + \frac{r_\alpha}{1-r_\alpha} + \frac{r_\alpha}{1-r_\alpha} \\ &= \frac{1+r_\alpha}{1-r_\alpha} = O(1). \end{aligned} \tag{36}$$

As for $n < x$, we have from inequality (32) that $\prod_{i=n}^{x-1} \frac{p_{i,i-1}}{p_{i,i+1}} \le 1$, meaning that from equation (34) we see that $\mathbb{E}_x[\zeta_n] \le \mathbb{E}_x[\zeta_x]$. Using equation (35) with $n = x$ gives

$$\mathbb{E}_x[\zeta_x] \le \frac{r_\alpha}{1-r_\alpha}.$$

Again, we take the expectation of equation (29) to show that for all $n < x = \lfloor \alpha N \rfloor$

$$\mathbb{E}_x[Z_n; A] \le \mathbb{E}_x[Z_n | A] = \mathbb{E}_x[\zeta_n] + \mathbb{E}_x[\zeta_{n+1}] \le \frac{2r_\alpha}{1-r_\alpha} = O(1).$$

Thus, we have shown that $\mathbb{E}_x[Z_n; A] = O(1)$, for all $\alpha > \beta$.

Similarly, for A^c, let $\bar{\zeta}_n$ denote the number of times the embedded chain $\tilde{X}^{(N)}$ jumps to the right from state n given that the chain is absorbed in state 0. In this case, the number of visits to state n conditioned on A^c is given by

$$Z_n \big| A^c = \begin{cases} 1 + \bar{\zeta}_n + \bar{\zeta}_{n-1} & \text{for } 1 \le n \le x, \\ \bar{\zeta}_n + \bar{\zeta}_{n-1} & \text{for } x < n < N-1, \end{cases} \tag{37}$$

and $\bar{\zeta}_n$ follows the recursive relation

$$\bar{\zeta}_n = \begin{cases} \sum_{l=0}^{\bar{\zeta}_{n-1}} \bar{\xi}_{l,n}, & \text{for } 1 \le n \le x, \\ \sum_{l=1}^{\bar{\zeta}_{n-1}} \bar{\xi}_{l,n}, & \text{for } x < n < N-1, \end{cases} \tag{38}$$

with $\bar{\zeta}_n = 0$, and $\bar{\xi}_{l,n}$ denoting the random number of transitions from state n to $n+1$ occurring between the l^{th} and $(l+1)^{th}$ right-jumps from state $n-1$, given that the embedded chain is absorbed into state 0. Moreover, $(\bar{\xi}_{l,n})_{l \ge 0}$ are i.i.d. following a geometric distribution from zero, with a mean of $p^{A^c}_{n,n+1}/p^{A^c}_{n,n-1}$, where

$$p^{A^c}_{n,n+1} = 1 - p^{A^c}_{n,n-1} = p_{n,n+1}\frac{\mathbb{P}_{n+1}(A^c)}{\mathbb{P}_n(A^c)}.$$

Moreover, we also have

$$p^{A^c}_{n,n-1} = p_{n,n-1}\frac{\mathbb{P}_{n-1}(A^c)}{\mathbb{P}_n(A^c)}.$$

Employing an analogous argument to that used in the proof of Theorem 4.2 for the exit probability to state N, we find

$$\mathbb{P}_n(A^c) = \frac{\sum_{t=n}^{N-1}\prod_{j=t+1}^{N-1}\frac{g_K(j/N)}{r}}{\sum_{t=0}^{N-1}\prod_{j=t+1}^{N-1}\frac{g_K(j/N)}{r}}.$$

Putting this together with equation (17) gives

$$\frac{p^{A^c}_{n,n+1}}{p^{A^c}_{n,n-1}} = \frac{g_K\left(\frac{n}{N}\right)}{r}\frac{\sum_{t=n+1}^{N-1}\prod_{j=t+1}^{N-1}\frac{g_K(j/N)}{r}}{\sum_{t=n-1}^{N-1}\prod_{j=t+1}^{N-1}\frac{g_K(j/N)}{r}}.$$

Since $(g_K(\frac{n}{N})/r)_{n\geq 1}$ is a monotonically non-decreasing non-negative sequence, consider the sequence $(y_n)_{n\geq 1}$ also monotonically non-decreasing non-negative, so that $y_k \leq y_{k+1}$ for all $k \geq 1$. As before,

$$y_j\frac{\sum_{t=n+1}^{N-1}\prod_{j=t+1}^{N-1}y_j}{\sum_{t=n-1}^{N-1}\prod_{j=t+1}^{N-1}y_j} = \frac{y_n + y_{N-1}y_n + y_{N-1}y_{N-2}y_n + \cdots + y_{N-1}\ldots y_{n+3}y_n + y_{N-1}\ldots y_{n+2}y_n}{1 + y_{N-1} + y_{N-1}y_{N-2} + \cdots + y_{N-1}\ldots y_{n+2}y_{n+1} + y_{N-1}\ldots y_{n+1}y_n},$$

where the numerator has $N-n-1$ terms and the denominator has $N-n+1$. We compare the terms in the numerator with the middle $N-n-1$ terms in the denominator to see

$$\begin{aligned} y_n &\leq y_{N-1}, \\ y_{N-1}y_n &\leq y_{N-1}y_{N-2}, \\ &\vdots \\ y_{N-1}\cdots y_{n+2}y_n &\leq y_{N-1}\cdots y_{n+2}y_{n+1}. \end{aligned}$$

Hereby, we conclude that

$$y_j\frac{\sum_{t=n+1}^{N-1}\prod_{j=t+1}^{N-1}y_j}{\sum_{t=n-1}^{N-1}\prod_{j=t+1}^{N-1}y_j} \leq 1.$$

Therefore, similar to earlier, we obtain

$$\frac{p^{A^c}_{n,n+1}}{p^{A^c}_{n,n-1}} \leq \min\left(1, \frac{g_K\left(\frac{n}{N}\right)}{r}\right). \tag{39}$$

Using computations similar to those used to obtain equation (34) for finding the expectation of equation (38)

$$\mathbb{E}_x[\bar{\zeta}_n] = \begin{cases} \sum_{t=1}^{n}\prod_{i=t}^{n}\frac{p^{A^c}_{i,i+1}}{p^{A^c}_{i,i-1}}, & \text{for } 1 \leq n \leq x, \\ \left(\prod_{i=x+1}^{n}\frac{p^{A^c}_{i,i+1}}{p^{A^c}_{i,i-1}}\right)\mathbb{E}_x[\bar{\zeta}_x], & \text{for } x < n < N-1. \end{cases} \tag{40}$$

For any $1 \leq n \leq x$, after simplifying equation (40), we find

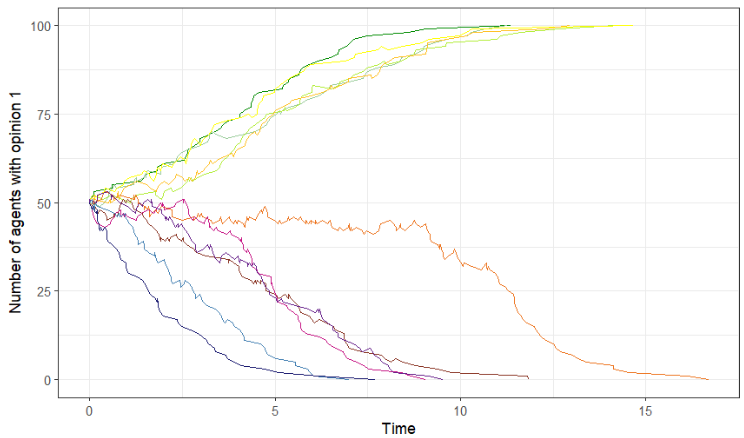


Figure 8: Simulations of the majority rule model for varying levels of bias. Parameters: $N = 100$, $\alpha = 0.5$, and $q_0 = 0.5$. Simulations run for $q_1 \in \{0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1\}$.

$$\mathbb{E}_x[\bar{\zeta}_n] = \frac{\prod_{j=1}^{n} \frac{g_K(j/N)}{r}}{\prod_{j=1}^{N-1} \frac{g_K(j/N)}{r}} \mathbb{P}_{n+1}(A^c)\left(1 - \mathbb{P}_n(A^c)\right). \tag{41}$$

From our assumption that $\alpha > \beta$, we have that for all $j \leq \lfloor \beta N \rfloor$, it holds that $\frac{g_K(j/N)}{r} \leq \frac{g_K(\beta)}{r} = 1$. Moreover, from equation (18) we see that we have the property $g_K(x) = 1/g_K(1-x)$. This means that we have $g_K(1/N) = 1/g_K(\frac{N-1}{N})$, $g_K(2/N) = 1/g_K(\frac{N-2}{N})$, ..., $g_K(1/2) = 1/g_K(1/2)$. We use this property to see that we get $\prod_{j=1}^{N-1} \frac{g_K(j/N)}{r} = 1/r^{N-1}$ by cancelling out the $g_K(j/N)$ terms. Therefore, from equation (41) it follows that, for all $n \leq \lfloor \beta N \rfloor$,

$$\mathbb{E}_x[\bar{\zeta}_n] \leq \frac{\prod_{j=1}^{n} \frac{g_K(j/N)}{r}}{\prod_{j=1}^{N-1} \frac{g_K(j/N)}{r}} \leq r^{N-1} \leq 1.$$

Additionally, for all $j > \lfloor \beta N \rfloor$, we have $\frac{g_K(j/N)}{r} \geq 1$. Thus, for all $\lfloor \beta N \rfloor < n \leq x$, it holds that

$$\mathbb{E}_x\left[\bar{\zeta}_n\right] \leq \frac{1}{\prod_{j=n+1}^{N-1} \frac{g_K(j/N)}{r}} \leq 1.$$

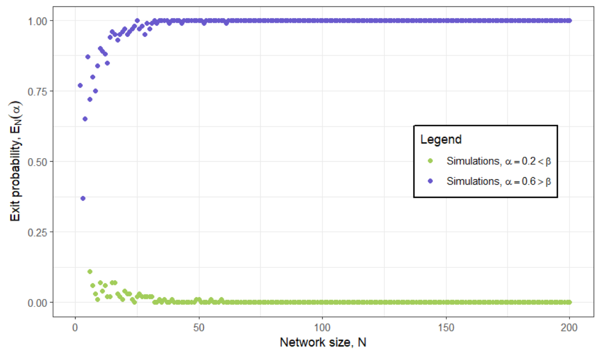
Hereby, we have demonstrated that $\mathbb{E}_x[\bar{\zeta}_n] = O(1)$ for all $1 \leq n \leq x$. From equations (39) and (40) we find $\mathbb{E}_x[\bar{\zeta}_n] \leq \mathbb{E}_x[\bar{\zeta}_x] = O(1)$ for all $x < n < N-1$. Then, using equation (37) we can conclude that $\mathbb{E}_x[Z_n; A^c] \leq \mathbb{E}_x[Z_n | A^c] = O(1)$.

Now assume $\alpha < \beta$, that is, $x = \lfloor \alpha N \rfloor \leq \lfloor \beta N \rfloor$. Again we condition on A, noting that we can use an argument similar to that used for $\alpha > \beta$ conditioned on A^c. In this case, we use the fact that for $j \geq \lfloor \beta N \rfloor$ we have $\frac{r}{g_K(j/N)} \leq 1$ and for $j < \lfloor \beta N \rfloor$ we have $\frac{r}{g_K(j/N)} \geq 1$. This gives us $\mathbb{E}_x[\zeta_n] \leq 1$, which implies that $\mathbb{E}_x[\zeta_n] = O(1)$, for $x \leq n \leq N-1$. Then, we also have for $1 \leq n < x$ that $\mathbb{E}_x[\zeta_n] \leq \mathbb{E}_x[\zeta_x] = O(1)$. Thus, proving that $\mathbb{E}_x[Z_n; A] = O(1)$ for $\alpha < \beta$.

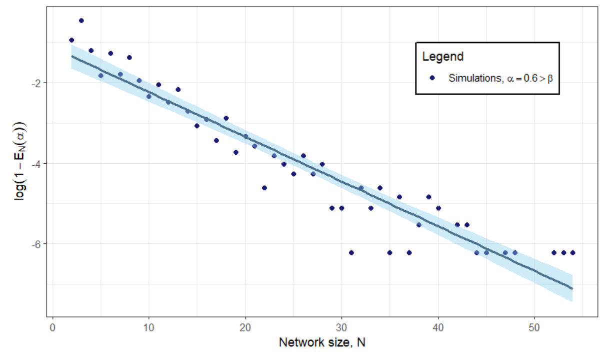
Then conditioning on A^c, for all $1 \leq n \leq x \leq \lfloor \beta N \rfloor$, we use the same argument as for the case that $\alpha > \beta$ with absorption in state N. In this case, note that for $n \leq x$ we use $\frac{g_K(n/N)}{r} \leq \frac{g_K(\alpha)}{r} < 1$, leading to $\mathbb{E}_x[\bar{\zeta}_n] = O(1)$. Then for $n > x$, we have $\mathbb{E}_x[\bar{\zeta}_n] \leq \mathbb{E}_x[\bar{\zeta}_x] = O(1)$. As such, proving that $\mathbb{E}_x[Z_n; A^c] = O(1)$ for $\alpha < \beta$.

Thus, for $\alpha \in (0, \beta) \cup (\beta, 1)$, we have proved that $\mathbb{E}_x[Z_n] = O(1)$, therefore concluding the proof.

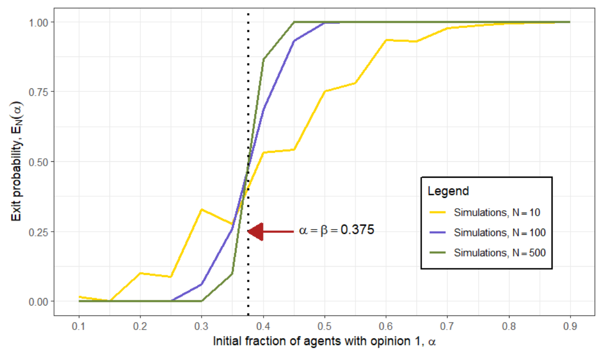
□



(a) Simulations for the exit probability, $E_N(\alpha)$, as a function of network size, N, for two values of α.



(b) Semi log plot taking $\log(1 - E_N(\alpha))$ as a function of network size, N, for $\alpha = 0.6$.



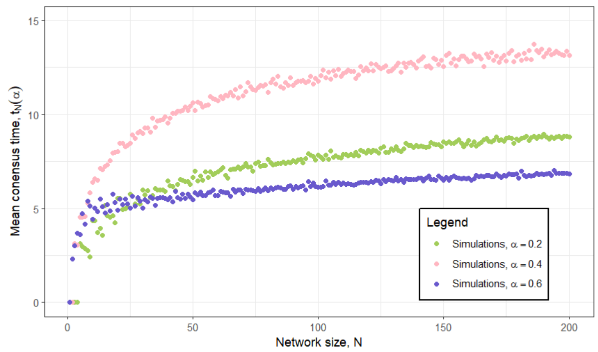
(c) Simulations for the exit probability, $E_N(\alpha)$, as a function of initial fraction of agents with opinion 1, α, for three values of N.

Figure 9: Simulations for the exit probability under the majority rule with biased agents. Parameters: $q_0 = 1$, $q_1 = 0.6$, and $K = 1$.

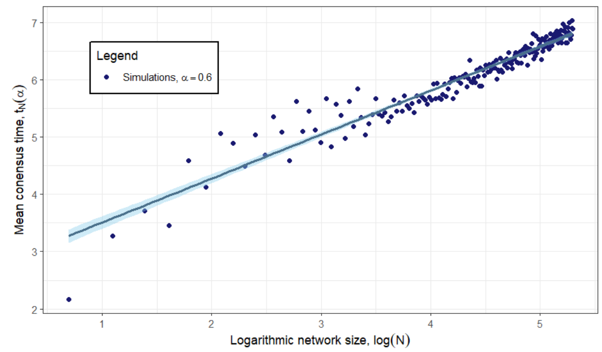
4.3 Simulations

The effect of bias on opinion dynamics within the majority rule model is demonstrated through simulations, the R script for this can be found in Appendix A. Figure 8 illustrates the evolution of agents' opinions on varying biases. This was done by setting $N = 100$, $\alpha = 0.5$, and $q_0 = 0.5$, then simulating the majority rule model for different values of q_1. The figure shows that when q_1 is larger than q_0, the network tends to reach consensus on opinion 1, with higher values of q_1 usually achieving consensus faster. Conversely, for values of q_1 less than q_0, the network tends to achieve consensus on opinion 0, with the lower values of q_1 usually achieving consensus faster. Moreover, the figure indicates that even for values of q_1 close to $q_0 = 0.5$, consensus is achieved fairly quickly in comparison to that seen in the voter model.

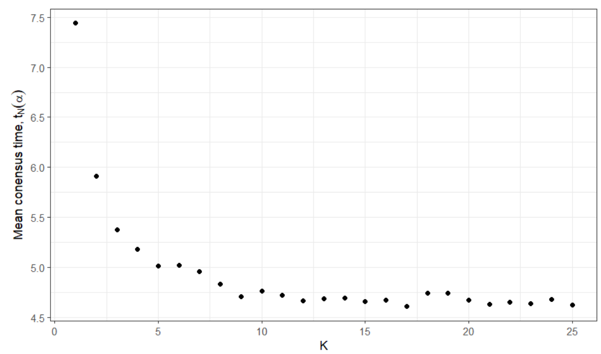
In Figure 9, the exit probability for simulations of the majority rule model is presented, with parameters $K = 1$, $q_0 = 1$ and $q_1 = 0.6$. For these values, the threshold for the phase transition of the exit probability can be computed to be $\beta = 0.375$. In Figure 9a, the exit probability for two values of α is plotted against network size, N. The exit probabilities are obtained by running 100 simulations for values of N from 1 to 200. It shows that for $\alpha = 0.2$ the exit probability converges to zero and for $\alpha = 0.6$ the exit probability converges to one, with convergence appearing exponential. This aligns with the theory that below β the exit probability converges exponentially to zero and above β the exit probability converges exponentially to one. Figure 9b is a semi-log plot, where we plot $\log(1 - E_N(\alpha))$ against N for $\alpha = 0.6$. As with the voter model, we see a linear relationship in this figure, as such providing evidence for the exponential convergence



(a) Simulations for mean consensus time, $t_N(\alpha)$, as a function of network size, N, for three values of alpha. Parameters: $K = 1$.



(b) Simulations for mean consensus time, $t_N(\alpha)$, plotted against the logarithm of the network size, $\log(N)$. Parameters: $K = 1$ and $\alpha = 0.6$.



(c) Simulations for mean consensus time, $t_N(\alpha)$, as a function of K. Parameters: $N = 50$ and $\alpha = 0.5$.

Figure 10: Simulations for mean consensus time under the majority rule with biased agents. Parameters: $q_0 = 1$ and $q_1 = 0.6$.

of the exit probability. In Figure 9c, the exit probability is plotted against values of α for three values of N; $N = 10$, 100, and 500. For this, 500 simulations were performed for the values of α from 0.1 to 0.9 in steps of 0.05. This figure clearly shows the phase transition at $\beta = 0.375$, with a sharper transition for larger values of N.

In Figure 10, the mean consensus time for simulations of the majority rule model is plotted for different values of N (Figure 10a), $\log(N)$ (Figure 10b) and K (Figure 10c). In the majority rule model, an agent updating their opinion randomly samples $2K$ agents in the network and adopts the majority opinion of these agents plus themselves. In all figures, we use the parameters $q_0 = 1$ and $q_1 = 0.6$, so the model is biased towards opinion 1. Figure 10a is obtained by running 200 simulations for each value of N, showing the mean consensus time for three values of α, with $K = 1$. The relationship between the mean consensus time and network size appears to be logarithmic. We see that for $\alpha = 0.6$ the mean consensus time is the lowest, with $\alpha = 0.2$ only slightly higher, and $\alpha = 0.4$ significantly higher. This is due to the fact that the exit probability for $\alpha = 0.4$ is closest to the phase transition value $\beta = 0.375$, meaning the exit probability takes longer to converge as N increases than for $\alpha = 0.6$ or $\alpha = 0.2$. As can be observed in Figure 9c, even for $N = 500$, the exit probability is still below 1. This suggests that it takes longer for the process to absorbed into either state 0 or N. Intuitively, if the chain has a probability of absorption into state N closer to 0.5, then even though the chain will be absorbed into either absorbing states in finite time, the time taken for this to happen will be longer, as there will be more transitions between the states $\{1, 2, ..., N-1\}$. To investigate the convergence of the mean consensus time, Figure 10b plots the mean consensus time for $\alpha = 0.6$ against $\log(N)$. This supports the claim that the convergence is exponential, because we see a linear relationship between mean consensus time and $\log(N)$. Figure 10c plots the mean consensus time from 500 simulations against various values of K, with $N = 50$ and $\alpha = 0.5$. From equation (26), we see that the upper bound depends on K only in the constant c. As K increases, c increases and, as such, the upper bound on the mean consensus time decreases. From this we conclude that as K increases, the mean consensus time decreases, this is reflected in Figure 10c. To give a more intuitive explanation of this, for larger K, agents take a larger sample of agents in the network, meaning that as more agents in the network adopt opinion 1 there is a higher probability that an agent will update their opinion to the preferred opinion, ensuring faster consensus.

5 Conclusion

This project explores the work of Mukhopadhyay et al. [3] on the influence of biased agents on opinion dynamics within voter and majority rule models, demonstrating that biases significantly influence consensus outcomes and time to consensus in social networks. The analysis shows that in both the voter model and the majority rule model, the mean consensus time is logarithmic in network size. Within the voter model, biased agents significantly reduce the mean consensus time, leading to exponentially faster convergence compared to the unbiased case. However, for the majority rule model with unbiased agents, the mean consensus time is also logarithmic. For the voter model, the exit probability converges to one as the network size grows, compared to the unbiased voter model, for which the exit probability remains constant. In the majority rule model with biased agents, the results demonstrate that there exists a phase transition in the exit probability. Above this certain α, the initial fraction of agents with opinion 1, the exit probability increases exponentially to one, and below this the exit probability exponentially decreases to zero. Whereas, for the majority rule model with unbiased agents, the network achieves consensus on the opinion with the initial majority with high probability. Finally, simulations of both the voter and majority rule models support the theorems presented.

The results offer significant insights into opinion dynamics, although some limitations must be noted. Addressing these limitations will deepen the understanding of opinion dynamics and extend the applicability of the findings to more complex and realistic real-world scenarios. Three primary directions for advancing these models in future research are: modifying the rules by which agents update their opinions, changing the spectrum of opinions available to agents, and changing the social structure and network topology [4].

The binary opinion framework, while useful for its simplicity, does not capture the full complexity of real-world beliefs. More advanced models could explore multidimensional opinion spaces, such as discrete arrays, continuous real values, or multidimensional vectors [4]. These would better capture the spectrum of opinions and attitudes found in society. However, numerous real-world scenarios involve binary choices, such as voters choosing between two candidates, making these models suitable for certain real-world applications.

The assumption that all agents share the same bias is limiting; a more realistic approach would allow for heterogeneous biases, stubborn agents, strategists, or opposing groups. It could be of interest to investigate models where agents have diverse biases, with some favouring one opinion, others favouring another, and some remaining neutral or stubborn. Various studies have already been conducted on networks with stubborn agents who do not change their opinion [3]. Moreover, another approach could be to consider agents as strategists, where they act as decision makers who exhibit selfishness, prioritising their own interests [29]. Additionally, the bias in the models presented can be seen as weak, as agents who hold the preferred opinion merely update their opinion less frequently than those who hold the alternative opinion. Therefore, it could be interesting to explore different ways of applying bias, such as models in which agents have a stronger bias toward a so-called superior opinion [30].

The network structure of a complete graph can be seen as an oversimplification, since real social interactions are rarely fully connected or uniformly distributed. Investigating more complex network topologies and adaptive social structures could develop understanding of opinion dynamics and improve the applicability of the model to real-world scenarios. Such topologies could be small-world networks, scale-free networks, cubic graphs, d-regular graphs [31], or rooted regular trees [32]. Another avenue of research could be dynamic networks in which connections between agents evolve over time, reflecting changes in social relationships, influence, and varying communication patterns. Changes in social structure could mean incorporating varying agent characteristics, such as fluctuating levels of influence or susceptibility to change. This could better reflect the roles of opinion leaders and followers on social networks.

Furthermore, it would be of interest to compare the model predictions with empirical data to assess the predictive power and real-world applicability of the models. Using real-world data would help in increasing the applicability of these models in fields such as political science, public health, and marketing. This empirical validation can be done by applying the models to specific case studies, such as opinions on politically polarising topics, public health campaigns, or the influence of marketing strategies.

Studies such as those of Mukhopadhyay et al. [3] contribute to the ever-growing field of opinion dynamics. Developing these ideas by adapting the models to be more complex and reflect real-world scenarios would advance both the theoretical and practical applications of opinion dynamics research.

References

[1] P. Clifford and A. Sudbury, "A model for spatial conflict," *Biometrika*, vol. 60, no. 3, pp. 581–588, 1973. DOI: 10.1093/biomet/60.3.581.

[2] L. Becchetti, A. Clementi, and E. Natale, "Consensus Dynamics: An Overview," *ACM SIGACT News*, vol. 51, no. 1, pp. 58–104, 2020. DOI: 10.1145/3388392.3388403.

[3] A. Mukhopadhyay, R. R. Mazumdar, and R. Roy, "Voter and majority dynamics with biased and stubborn agents," *Journal of Statistical Physics*, vol. 181, pp. 1239v–1265, 2020. DOI: 10.1007/s10955-020-02625-w.

[4] H. Xia, H. Wang, and Z. Xuan, "Opinion Dynamics: A Multidisciplinary Review and Perspective on Future Research," *International Journal of Knowledge and Systems Science*, vol. 2, no. 4, pp. 72–91, 2011. DOI: 10.4018/jkss.2011100106.

[5] J. R. P. French, "A formal theory of social power," *Psychological review*, vol. 63, no. 3, pp. 181–194, 1956. DOI: 10.1037/h0046123.

[6] V. Sood and S. Redner, "Voter model on heterogeneous graphs," *Physical Review Letters*, vol. 94, no. 17, 178701(4), 2005. DOI: 10.1103/PhysRevLett.94.178701.

[7] E. Perron, D. Vasudevan, and M. Vojnovic, "Using three states for binary consensus on complete graphs," in *IEEE INFOCOM 2009*, Rio de Janeiro, Brazil, 2009, pp. 2527–2535. DOI: 10.1109/INFCOM.2009.5062181.

[8] M. Yildiz, R. Pagliari, A. Ozdaglar, and A. Scaglione, "Voting models in random networks," in *2010 Information Theory and Applications Workshop, ITA 2010 - Conference Proceedings*, ser. 2010 Information Theory and Applications Workshop, ITA 2010 - Conference Proceedings, La Jolla, CA, USA, 2010, pp. 1–7. DOI: 10.1109/ITA.2010.5454090.

[9] C. Cooper, R. Elsässer, H. Ono, and T. Radzik, "Coalescing random walks and voting on connected graphs," *SIAM Journal on Discrete Mathematics*, vol. 27, no. 4, pp. 1748–1758, 2013. DOI: 10.1137/120900368.

[10] S. Galam, "Minority opinion spreading in random geometry," *The European Physical Journal B*, vol. 25, no. 4, pp. 403–406, 2002. DOI: 10.1140/epjb/e20020045.

[11] J. Cruise and A. Ganesh, "Probabilistic consensus via polling and majority rules," *Queueing Systems*, vol. 78, no. 2, pp. 99–120, 2014. DOI: 10.1007/s11134-014-9397-7.

[12] M. Mobilia, "Does a single zealot affect an infinite group of voters?" *Physical Review Letters*, vol. 91, no. 2, 028701(4), 2003. DOI: 10.1103/PhysRevLett.91.028701.

[13] M. Mobilia, A. Petersen, and S. Redner, "On the role of zealotry in the voter model," *Journal of Statistical Mechanics: Theory and Experiment*, vol. 2007, no. 8, 08029(17), 2007. DOI: 10.1088/1742-5468/2007/08/p08029.

[14] E. Yildiz, D. Acemoglu, A. Ozdaglar, A. Saberi, and A. Scaglione, "Binary opinion dynamics with stubborn agents," *SSRN Electronic Journal*, vol. 1, pp. 1–39, 2011. DOI: 10.2139/ssrn.1744113.

[15] C. Castellano, S. Fortunato, and V. Loreto, "Statistical physics of social dynamics," *Reviews of Modern Physics*, vol. 81, no. 2, pp. 591–646, 2009. DOI: 10.1103/revmodphys.81.591.

[16] R. Axelrod, "The dissemination of culture: A model with local convergence and global polarization," *Journal of Conflict Resolution*, vol. 41, no. 2, pp. 203–226, 1997. DOI: 10.1177/0022002797041002001.

[17] S. Vitzthum, "Dissemination of culture in a networked world: The influence of network structure on the outcomes of the axelrod cultural model," in *AMCIS 2007 Proceedings*, vol. 413, Association for Information Systems Electronic Library, 2007. [Online]. Available: http://aisel.aisnet.org/amcis2007/413 (visited on Jan. 25, 2025).

[18] G. Deffuant, D. Neau, F. Amblard, and G. Weisbuch, "Mixing beliefs among interacting agents," *Advances in Complex Systems*, vol. 03, pp. 87–98, 2000. DOI: 10.1142/S0219525900000078.

[19] D. Stauffer, A. Sousa, and C. Schulze, “Discretized opinion dynamics of the deffuant model on scale-free networks,” *Journal of Artificial Societies and Social Simulation*, vol. 7, no. 3, pp. 1–7, 2004. [Online]. Available: https://api.semanticscholar.org/CorpusID:40688346.

[20] F. Amblard and G. Deffuant, “The role of network topology on extremism propagation with the relative agreement opinion dynamics,” *Physica A: Statistical Mechanics and its Applications*, vol. 343, pp. 725–738, 2004. DOI: 10.1016/j.physa.2004.06.102.

[21] M. Huber, “Halving the bounds for the markov, chebyshev, and chernoff inequalities using smoothing,” *The American Mathematical Monthly*, vol. 126, no. 10, pp. 915–927, 2019. DOI: 10.1080/00029890.2019.1656484.

[22] D. L. Sun, “Wald’s identity and geometric expectation,” *The American Mathematical Monthly*, vol. 127, no. 8, p. 716, 2020. DOI: 10.1080/00029890.2020.1790909.

[23] T. Leighton and R. Rubinfeld. “Lecture notes: Random walks.” (2006), [Online]. Available: http://web.mit.edu/neboat/Public/6.042/randomwalks.pdf (visited on Jan. 15, 2025).

[24] S. Song and J. Song, “A note on the history of the gambler’s ruin problem,” *Communications for Statistical Applications and Methods*, vol. 20, no. 2, pp. 157–168, 2013. DOI: 10.5351/CSAM.2013.20.2.157.

[25] D. A. Levin, Y. Peres, E. L. Wilmer, J. Propp, and D. B. Wilson, *Markov Chains and Mixing Times*, Second edition. American Mathematical Society, 2017.

[26] R. Gallager. “Course notes: Chapter 2: Poisson processes.” (2011), [Online]. Available: https://ocw.mit.edu/courses/6-262-discrete-stochastic-processes-spring-2011/resources/mit6_262s11_chap02/ (visited on Jan. 23, 2025).

[27] K. B. Athreya and P. E. Ney, “T. E. Harris and branching processes,” *The Annals of Probability*, vol. 39, no. 2, pp. 429–434, 2011. DOI: 10.1214/10-AOP599.

[28] K. B. Athreya and P. E. Ney, “Special processes,” in *Branching Processes*. Springer, 2004, ch. VI. Pp. 229–267. DOI: 10.1007/978-3-642-65371-1_6.

[29] W. Cao, H. Zhang, G. Kou, and B. Zhang, “Discrete opinion dynamics in social networks with stubborn agents and limited information,” *Information Fusion*, vol. 109, p. 102 410, 2024. DOI: 10.1016/j.inffus.2024.102410.

[30] A. Mukhopadhyay, “Phase transitions in biased opinion dynamics with 2-choices rule,” *Probability in the Engineering and Informational Sciences*, vol. 38, no. 2, pp. 227–244, 2024. DOI: 10.1017/S0269964823000098.

[31] H. Lesfari, F. Giroire, and S. Pérennes, “Biased Majority Opinion Dynamics: Exploiting graph k-domination,” in *IJCAI 2022 - International Joint Conference on Artificial Intelligence*, Vienna, Austria: International Joint Conferences on Artificial Intelligence Organization, 2022, pp. 377–383. DOI: 10.24963/ijcai.2022/54. [Online]. Available: https://hal.science/hal-03676312 (visited on Feb. 4, 2025).

[32] M. Podder and A. Sarkar, *Learning models on rooted regular trees with majority update policy: Convergence and phase transition*, 2024. arXiv: 2405.12418 [math.PR]. [Online]. Available: https://arxiv.org/abs/2405.12418 (visited on Feb. 4, 2025).

A Appendix

```
# Set up
rm(list= ls()) # Removes all objects from workspace.
set.seed(0)

# Library packages used
library("latex2exp")
library("tidyverse")
library("dplyr")
library("reshape2")

#function for CTMC
voter_CTMC <- function(q0,q1,N,alpha){
  X = vector()  # X the embedded markov chain
  X[1] = floor(N*alpha) # set initial state as proportion alpha of N total
                        #agents having opinion 1
  t <- vector()
  t[1] <- 0
  while(TRUE){
    if(X[length(X)] == N) break #stop the simulation when state N is reached
    else if (X[length(X)] == 0) break
    #stop the simulation when state 0 is reached
    else {
      q_plus <- q0*X[length(X)]*((N-X[length(X)])/N) # rate q(k->k+1)
      q_minus <- q1*X[length(X)]*((N-X[length(X)])/N) # rate q(k->k-1)
      M.exp.rate <- X[length(X)]*(q0+q1)*((N-X[length(X)])/N)
          # rate of exponential variable M_{j,k}, jth visit to state k
      X_change <- X[length(X)] + sample(c(-1,1), 1, replace = TRUE,
                                        prob = c(q_minus,q_plus))
      X <- c(X, X_change) # for all other states the chain takes a step
                          #to the left or right
      t <- c(t, t[length(t)] + rexp(1,rate=M.exp.rate))
      next
    }
  }
  return(list(X = X, time = t)) # returns the states that the MC goes
                                #through and the corresponding time step
}

#####################################################

#### Plotting chains for multiple values of q1
q0 <- 0.5
N <- 100
alpha <- 0.5
q1.vec <-  vector()
q1.times <- vector()
q1.X <- vector()
q1.mat <- vector()
q1.X.mat <- vector()
```

```
q1.times.mat <- vector()
counter <- 0

for (i in seq(0,1,0.1)) {
  counter <- counter +1
  q1.sim <- voter_CTMC(q0,i,N,alpha)
  q1.vec <- rep(i, length(q1.sim$X))
  q1.times <- q1.sim$time
  q1.X <- q1.sim$X
  q1.mat <- qpcR:::cbind.na(q1.mat, q1.vec)
  q1.times.mat <- qpcR:::cbind.na(q1.times.mat, q1.times)
  q1.X.mat <- qpcR:::cbind.na(q1.X.mat, q1.X)
}


df.q1.chain <- data.frame(q1.mat[,-1],
                          q1.times.mat[,-1],
                          q1.X.mat[,-1])


ggplot(df.q1.chain) +
  geom_line(aes(x = q1.times, y = q1.X), colour = "green4") +
  geom_line(aes(x = q1.times.1, y = q1.X.1), colour = "darkseagreen3") +
  geom_line(aes(x = q1.times.2, y = q1.X.2), colour = "olivedrab2") +
  geom_line(aes(x = q1.times.3, y = q1.X.3), colour = "yellow1") +
  geom_line(aes(x = q1.times.4, y = q1.X.4), colour = "goldenrod1") +
  geom_line(aes(x = q1.times.5, y = q1.X.5), colour = "chocolate2") +
  geom_line(aes(x = q1.times.6, y = q1.X.6), colour = "tomato4") +
  geom_line(aes(x = q1.times.7, y = q1.X.7), colour = "mediumvioletred") +
  geom_line(aes(x = q1.times.8, y = q1.X.8), colour = "darkorchid4") +
  geom_line(aes(x = q1.times.9, y = q1.X.9), colour = "steelblue") +
  geom_line(aes(x = q1.times.10, y = q1.X.10), colour = "midnightblue") +
  theme_bw() +
  labs(x = TeX("Time"), y = TeX("Number of agents with opinion $1$"))


####################################################

#### Simulations for mean consensus time for different values of N
# Values & set up
nsims <- 100
N_min <- 1
N_max <- 200
N_steps <- 1
q0 <- 1
q1 <- 0.5
alpha <- 0.2
time.sims <- matrix(nrow = nsims, ncol = length(seq(N_min,N_max,N_steps)))
sim.num <-matrix(nrow = nsims, ncol = length(seq(N_min,N_max,N_steps)))
end.sim <- matrix(nrow = nsims, ncol = length(seq(N_min,N_max,N_steps)))
mean.time <- vector()
```

```
N.sims <- vector()

# Running simulations
for (j in seq(N_min,N_max,N_steps)){
  N.sims[j] <- j
  for (i in 1:nsims){
    a.sim <- voter_CTMC(q0,q1,j,alpha)
    time.sims[i,j] <- a.sim$time[length(a.sim$time)]
    end.sim[i,j] <- a.sim$X[length(a.sim$X)]
  }
  mean.time[j] <- mean(time.sims[,j])
}

# Editing and plotting data
N.data <- vector()
time.data <- vector()
end.data <- vector()

for (j in N_min:N_max) {
  N.data <- c(N.data, rep(j, nsims))
  time.data <- c(time.data, time.sims[,j])
  end.data <- c(end.data, end.sim[,j])
}

data.sims <- data.frame(N.data,
                        time.data,
                        end.data)


### Theoretical upper and lower bound for mean consensus time
mean.low <- vector()
mean.up <- vector()
p <- q0/(q0+q1)
r <- (1-p)/p
for (j in N_min:N_max) {
  mean.low[j] <- (1/(q0+q1))*log((j)*min(alpha, 1-alpha))
  mean.up[j] <- (2/(q0+q1))*((1+r)/(1-r))*(log(j-1)+1)
}
df.mean <- data.frame(1:N_max, mean.time, mean.low, mean.up)
names(df.mean) <- c("N", "Mean.time", "Lower.bound", "Upper.bound")

ggplot(df.mean) +
  geom_point(aes(x = N, y = Mean.time, colour = 'black')) +
  geom_line(aes(x = N, y = Lower.bound, colour = 'red'), size = 1) +
  geom_line(aes(x = N, y = Upper.bound, colour = 'blue'), size = 1) +
  theme_bw() +
  ylim(-1, 25) +
  labs(x = TeX("Network size, $N$"),
       y = TeX("Mean conensus time, $t_{N}(\\alpha)$"), colour = "Legend") +
  scale_color_manual(labels = c("Simulations", "Upper bound", "Lower bound"),
                # for some reason the colouring doesn't match, so we adjust it
```

```r
                    values = c("midnightblue", "plum2", "darkolivegreen4")) +
  theme(legend.position = c(0.12, 0.81),
        legend.background = element_rect(fill="white",
                                         size=0.75, linetype="solid",
                                         colour ="black"))


######################################################
### Plotting against log(N)
N.log <- log(1:N_max)
log.df.mean <- data.frame(N.log, mean.time, mean.low, mean.up)
names(log.df.mean) <- c("log.N", "Mean.time", "Lower.bound", "Upper.bound")
edit.log.df.mean <- log.df.mean[log.df.mean$Mean.time != 0,]

ggplot(edit.log.df.mean) +
  geom_point(aes(x = log.N, y = Mean.time, colour = 'black')) +
  geom_line(aes(x = log.N, y = Lower.bound, colour = 'red'), size = 1) +
  geom_line(aes(x = log.N, y = Upper.bound, colour = 'blue'), size = 1) +
  geom_smooth(aes(x = log.N, y = Mean.time), method = lm, colour = "skyblue4",
              fill = "skyblue") +
  theme_bw() +
  labs(x = TeX("Logarithmic network size, $log(N)$"),
       y = TeX("Mean conensus time, $t_{N}(\\alpha)$"), colour = "Legend") +
  scale_color_manual(labels = c("Simulations", "Upper bound", "Lower bound"),
                    # for some reason the colouring doesn't match, so we adjust it
                    values = c("midnightblue", "plum2", "darkolivegreen4")) +
  theme(legend.position = c(0.12, 0.81),
        legend.background = element_rect(fill="white",
                                         size=0.75, linetype="solid",
                                         colour ="black"))


######################################################
#### Simulations for exit probability for different values of N
# Values & set up
nsims <- 200
N_min <- 1
N_max <- 100
N_steps <- 1
q0 <- 1
q1 <- 0.5
alpha <- 0.2
time.sims <- matrix(nrow = nsims, ncol = length(seq(N_min,N_max,N_steps)))
sim.num <-matrix(nrow = nsims, ncol = length(seq(N_min,N_max,N_steps)))
end.sim <- matrix(nrow = nsims, ncol = length(seq(N_min,N_max,N_steps)))
mean.time <- vector()
N.sims <- vector()

# Running simulations
for (j in seq(N_min,N_max,N_steps)){
  N.sims[j] <- j
```

```
  for (i in 1:nsims){
    a.sim <- voter_CTMC(q0,q1,j,alpha)
    time.sims[i,j] <- a.sim$time[length(a.sim$time)]
    end.sim[i,j] <- a.sim$X[length(a.sim$X)]
  }
  mean.time[j] <- mean(time.sims[,j])
}

# Editing and plotting data
N.data <- vector()
time.data <- vector()
end.data <- vector()

for (j in N_min:N_max) {
  N.data <- c(N.data, rep(j, nsims))
  time.data <- c(time.data, time.sims[,j])
  end.data <- c(end.data, end.sim[,j])
}

data.sims <- data.frame(N.data,
                        time.data,
                        end.data)

# Exit probability from simulations
data.exitprob <- vector()
for (i in N_min:N_max) {
  data.exitprob[i] <-  length(data.sims$end.data[data.sims$end.data == i])/nsims
}
N.exit.sims <- seq(from = N_min, to = N_max, by = N_steps)
df.exitprob.sims <- data.frame(
  N.exit.sims,
  data.exitprob
)

# exit probability theory
N.theory <- seq(from = N_min, to = N_max, by = N_steps)
p <- q0/(q0+q1)
r <- (1-p)/p
E_N <- (1-r^{floor(alpha*N.theory)})/(1-r^N.theory)
theory.exit <- data.frame(N.theory,
                          E_N)
unbiased.theory <- rep(alpha, length(N.theory))
df.exit.prob <- cbind(theory.exit, df.exitprob.sims$data.exitprob, unbiased.theory)
names(df.exit.prob) <- c("N", "theory.exit.prob", "sim.exit.prob", "unbiased.theory")

### plotting data and theory in one graph
ggplot(data = df.exit.prob) +
  geom_line(mapping=aes(x = N, y = theory.exit.prob, colour ='red'), linewidth = 1) +
  geom_line(mapping=aes(x = N, y = unbiased.theory, colour = "darkolivegreen4"),
            linewidth = 1)+
  geom_point(mapping = aes(x= N, y = sim.exit.prob, colour = 'black')) +
```

```
  labs(x = TeX("Network size, $N$"), y = TeX("Exit probability, $E_N(\\alpha)$"),
       colour = "Legend") +
  theme_bw() +
  scale_y_continuous(breaks=seq(0,1,0.2)) +
  scale_color_manual(labels = c("Simulations", "Unbiased theory",
                                "Biased theory"),
                     values = c("midnightblue", "darkolivegreen4", "plum2")) +
  theme(legend.position = c(0.8, 0.7),
        legend.background = element_rect(fill="white",
                                         size=0.75, linetype="solid",
                                         colour ="black"))


### Plotting log(1-E_N) vs N to show growth is exponential
log.sim.exit.prob <- log(1-df.exit.prob$sim.exit.prob)
log.df.exit.prob <- data.frame(df.exit.prob$N, log.sim.exit.prob)
names(log.df.exit.prob) <- c("N", "log.sim.exit.prob")
# remove -infinity values, cleans up the data for plotting
edit.log.df.exit.prob <- log.df.exit.prob[log.df.exit.prob$log.sim.exit.prob != -Inf,]
# plotting
ggplot(data = edit.log.df.exit.prob) +
  geom_point(mapping = aes(x= N, y = log.sim.exit.prob, colour = 'black')) +
  geom_smooth(aes(x = N, y = log.sim.exit.prob, colour = 'red'), method = lm,
              colour = "skyblue4", fill = "skyblue") +
  labs(x = TeX("Network size, $N$"), y = TeX("$log(1-E_N(\\alpha))$"),
       colour = "Legend") +
  theme_bw() +
  scale_y_continuous(breaks=seq(0,1,0.2)) +
  scale_color_manual(labels = c("Simulations"),
                     values = c("midnightblue")) +
  theme(legend.position = c(0.8, 0.8),
        legend.background = element_rect(fill="white",
                                         size=0.75, linetype="solid",
                                         colour ="black"))


#####################################################
#### Biased simulations mean consensus time for different values of alpha
# Values & set up
nsims <- 100
q0 <- 1
q1 <- 0.5
N <- 100
k <- 0

time.sims <- matrix(nrow = nsims, ncol = length(seq(0.1,0.9,0.05)))
sim.num <-matrix(nrow = nsims, ncol = length(seq(0.1,0.9,0.05)))
end.sim <- matrix(nrow = nsims, ncol = length(seq(0.1,0.9,0.05)))
mean.time <- vector()
alpha.sims <- vector()
```

```
alpha.data <- vector()
time.data <- vector()
end.data <- vector()

### simulations biased
for (j in seq(0.1,0.9,0.05)){
  k <- k+1
  alpha.sims[k] <- j
  for (i in 1:nsims){
    a.sim <- voter_CTMC(q0,q1,N,j)
    time.sims[i,k] <- a.sim$time[length(a.sim$time)]
    end.sim[i,k] <- a.sim$X[length(a.sim$X)]
  }
  mean.time[k] <- mean(time.sims[,k])
  alpha.data <- c(alpha.data, rep(j, nsims))
  time.data <- c(time.data, time.sims[,k])
  end.data <- c(end.data, end.sim[,k])
}

# Editing data
df.mean.alpha.biased <- data.frame(seq(0.1,0.9,0.05), mean.time)
names(df.mean.alpha.biased) <- c("mean.alpha", "mean.time")


#### Unbiased simulations mean consensus time for different values of alpha
# values & set up
nsims <- 500
q0 <- 1
q1 <- 1
N <- 100
k <- 0

time.sims <- matrix(nrow = nsims, ncol = length(seq(0.1,0.9,0.05)))
sim.num <-matrix(nrow = nsims, ncol = length(seq(0.1,0.9,0.05)))
end.sim <- matrix(nrow = nsims, ncol = length(seq(0.1,0.9,0.05)))
mean.time <- vector()
alpha.sims <- vector()

alpha.data <- vector()
time.data <- vector()
end.data <- vector()

### simulations unbiased
for (j in seq(0.1,0.9,0.05)){
  k <- k+1
  alpha.sims[k] <- j
  for (i in 1:nsims){
    a.sim <- voter_CTMC(q0,q1,N,j)
    time.sims[i,k] <- a.sim$time[length(a.sim$time)]
    end.sim[i,k] <- a.sim$X[length(a.sim$X)]
  }
```

```r
  mean.time[k] <- mean(time.sims[,k])
  alpha.data <- c(alpha.data, rep(j, nsims))
  time.data <- c(time.data, time.sims[,k])
  end.data <- c(end.data, end.sim[,k])
}

# making data for mean consensus time as function of alpha
df.mean.alpha.unbiased <- data.frame(seq(0.1,0.9,0.05), mean.time)
names(df.mean.alpha.unbiased) <- c("mean.alpha", "mean.time")

# Theory mean consensus time as function of alpha
mean.alpha.theory.unbiased <- vector()
k <- 0
for (j in seq(0.1,0.9,0.05)) {
  k <- k+1
  mean.alpha.theory.unbiased[k] <- N*(-((j*log(j))+((1-j)*log(1-j))))
}

# Collecting all mean consensus times in one data frame
df.mean.alpha <- data.frame(seq(0.1,0.9,0.05), df.mean.alpha.unbiased$mean.time,
                            mean.alpha.theory.unbiased,
                            df.mean.alpha.biased$mean.time,
                            mean.alpha.theory.biased)
names(df.mean.alpha) <- c("mean.alpha", "mean.time.unbiased", "mean.theory.unbiased",
                          "mean.time.biased", "mean.theory.biased")

### plotting everything in one graph
ggplot(df.mean.alpha) +
  geom_smooth(aes(x = mean.alpha, y = mean.time.biased, colour = 'green'),
              method = lm, size = 1, colour = "plum3", fill = "thistle2")+
  geom_line(aes(x = mean.alpha, y = mean.theory.unbiased, colour = 'red'), size = 1) +
  geom_point(aes(x = mean.alpha, y = mean.time.unbiased, colour = 'black')) +
  geom_point(aes(x = mean.alpha, y = mean.time.biased, colour = 'blue')) +
  theme_bw() +
  scale_x_continuous(breaks=seq(0.1,0.9,0.1)) +
  labs(x = TeX("Initial fraction of agents with opinion 1, $\\alpha$"),
       y = TeX("Mean conensus time, $t_{N}(\\alpha)$"), colour = "Legend") +
  scale_color_manual(labels = c("Simulations, unbiased agents",
                                "Simulations, biased agents",
                                "Theory, unbiased agents"),
                     # for some reason the colouring doesn't match
                     values = c("darkolivegreen", "midnightblue",
                                "darkseagreen3"))+
  theme(legend.position = c(0.5, 0.5),
        legend.background = element_rect(fill="white",
                                         size=0.75, linetype="solid",
                                         colour ="black"))


### plotting just the simulations of the mean consensus time for biased agents
ggplot(df.mean.alpha) +
```

```
  geom_smooth(aes(x = mean.alpha, y = mean.time.biased),
              method = lm, size = 1, colour = "plum3", fill = "thistle2") +
  geom_point(aes(x = mean.alpha, y = mean.time.biased, colour = 'blue')) +
  theme_bw() +
  scale_x_continuous(breaks=seq(0.1,0.9,0.1)) +
  labs(x = TeX("Initial fraction of agents with opinion 1, $\\alpha$"),
       y = TeX("Mean conensus time, $t_{N}(\\alpha)$"), colour = "Legend") +
  scale_color_manual(labels = c("Simulations, biased agents"),
                     values = c("midnightblue"))+
  theme(legend.position = c(0.8, 0.8),
        legend.background = element_rect(fill="white",
                                          size=0.75, linetype="solid",
                                          colour ="black"))


############################################################

### Function for majority model CTMC
majority_CTMC <- function(q0,q1,N,alpha,K){
  X = vector()  # X the embedded markov chain
  X[1] = floor(N*alpha) # set initial state as proportion alpha of N total
                        # agents having opinion 1
  t <- vector() # time vector for transitions
  t[1] <- 0
  while(TRUE){
    if(X[length(X)] == N) break #stop the simulation when state N is reached
    else if (X[length(X)] == 0) break #stop the simulation when state 0 is reached
    else {
      q_plus <- q0*(N-X[length(X)])*(1-pbinom(K,2*K,X[length(X)]/N))
      # rate q(k->k+1)
      q_minus <- q1*X[length(X)]*(1-pbinom(K,2*K,1-X[length(X)]/N))
      # rate q(k->k-1)
      M.exp.rate <- q_minus + q_plus # rate of exponential variable M_{j,k},
                                      # jth visit to state k
      X_change <- X[length(X)] + sample(c(-1,1), 1, replace = TRUE,
                                         prob = c(q_minus,q_plus))
      X <- c(X, X_change) # for all other states the chain takes a step to the
                          # left or right
      t_change <- t[length(t)] + rexp(1,rate=M.exp.rate)
      t <- c(t, t_change)
      next
    }
  }
  return(list(X = X, time = t)) # returns the states that the MC goes through
                                # and the corresponding time step
}

############################################################

### plotting chains for multiple values of q1
q0 <- 0.5
```

```
N <- 100
alpha <- 0.5
K <- 1
q1.vec <-  vector()
q1.times <- vector()
q1.X <- vector()
q1.mat <- vector()
q1.X.mat <- vector()
q1.times.mat <- vector()
counter <- 0

for (i in seq(0,1,0.1)) {
  counter <- counter +1
  q1.sim <- majority_CTMC(q0,i,N,alpha, K)
  q1.vec <- rep(i, length(q1.sim$X))
  q1.times <- q1.sim$time
  q1.X <- q1.sim$X
  q1.mat <- qpcR:::cbind.na(q1.mat, q1.vec)
  q1.times.mat <- qpcR:::cbind.na(q1.times.mat, q1.times)
  q1.X.mat <- qpcR:::cbind.na(q1.X.mat, q1.X)
}


df.q1.chain <- data.frame(q1.mat[,-1],
                          q1.times.mat[,-1],
                          q1.X.mat[,-1])


ggplot(df.q1.chain) +
  geom_line(aes(x = q1.times, y = q1.X), colour = "green4") +
  geom_line(aes(x = q1.times.1, y = q1.X.1), colour = "darkseagreen3") +
  geom_line(aes(x = q1.times.2, y = q1.X.2), colour = "olivedrab2") +
  geom_line(aes(x = q1.times.3, y = q1.X.3), colour = "yellow1") +
  geom_line(aes(x = q1.times.4, y = q1.X.4), colour = "goldenrod1") +
  geom_line(aes(x = q1.times.5, y = q1.X.5), colour = "chocolate2") +
  geom_line(aes(x = q1.times.6, y = q1.X.6), colour = "tomato4") +
  geom_line(aes(x = q1.times.7, y = q1.X.7), colour = "mediumvioletred") +
  geom_line(aes(x = q1.times.8, y = q1.X.8), colour = "darkorchid4") +
  geom_line(aes(x = q1.times.9, y = q1.X.9), colour = "steelblue") +
  geom_line(aes(x = q1.times.10, y = q1.X.10), colour = "midnightblue") +
  theme_bw() +
  labs(x = TeX("Time"), y = TeX("Number of agents with opinion $1$"))

#####################################################

### function g_K
g_K <- function(x,K){
  ((1/x)*(1-pbinom(K,2*K,x)))/((1/(1-x))*(1-pbinom(K,2*K,1-x)))
}

#####################################################
```

```
### Function for majority model simulations for different values of N
maj_sims_N <- function(q0,q1,alpha,K,N_max,nsims){
  N_min <- 1
  N_steps <- 1
  time.sims <- matrix(nrow = nsims, ncol = length(seq(N_min,N_max,N_steps)))
  sim.num <-matrix(nrow = nsims, ncol = length(seq(N_min,N_max,N_steps)))
  end.sim <- matrix(nrow = nsims, ncol = length(seq(N_min,N_max,N_steps)))
  mean.time <- vector()
  N.sims <- vector()

  for (j in seq(N_min,N_max,N_steps)){
    N.sims[j] <- j
    for (i in 1:nsims){
      a.sim <- majority_CTMC(q0,q1,j,alpha,K)
      time.sims[i,j] <- a.sim$time[length(a.sim$time)]
      end.sim[i,j] <- a.sim$X[length(a.sim$X)]
    }
    mean.time[j] <- mean(time.sims[,j])
  }

  #### making data frame
  N.data <- vector()
  time.data <- vector()
  end.data <- vector()

  for (j in N_min:N_max) {
    N.data <- c(N.data, rep(j, nsims))
    time.data <- c(time.data, time.sims[,j])
    end.data <- c(end.data, end.sim[,j])
  }

  data.sims <- data.frame(N.data,
                          time.data,
                          end.data)
  return(data.sims)
}


##################################################

### simulations for mean consensus time for three values of alpha
q0 <- 1
q1 <- 0.6
K <- 1
N_steps <- 1
N_min <- 1
N_max <- 200
nsims <- 200

sims_0.2_mean <- maj_sims_N(q0,q1,0.2,K,N_max,nsims)
```

```
sims_0.4_mean <- maj_sims_N(q0,q1,0.4,K,N_max,nsims)
sims_0.6_mean <- maj_sims_N(q0,q1,0.6,K,N_max,nsims)

### making data for mean time
data.mean.0.2 <- vector()
data.mean.0.4 <- vector()
data.mean.0.6 <- vector()

for (i in 1:N_max) {
  data.mean.0.2[i] <-  mean(sims_0.2_mean[sims_0.2_mean$N.data == i, 2])
}

for (i in 1:N_max) {
  data.mean.0.4[i] <-  mean(sims_0.4_mean[sims_0.4_mean$N.data == i, 2])
}

for (i in 1:N_max) {
  data.mean.0.6[i] <-  mean(sims_0.6_mean[sims_0.6_mean$N.data == i, 2])
}

df.mean <- data.frame(1:N_max, data.mean.0.2, data.mean.0.4, data.mean.0.6)
names(df.mean) <- c("N", "mean.time.0.2", "mean.time.0.4", "mean.time.0.6")

df.mean.long <- melt(df.mean, id = "N")

### plotting mean time
ggplot(df.mean.long,
       aes(x = N,
           y = value,
           color = variable)) +
  geom_point() +
  theme_bw() +
  labs(x = TeX("Network size, $N$"), y = TeX("Mean conensus time,
                                              $t_{N}(\\alpha)$"),
       colour = "Legend") +
  scale_color_manual(labels = c(TeX("Simulations, $\\alpha = 0.2$"),
                                TeX("Simulations, $\\alpha = 0.4$"),
                                TeX("Simulations, $\\alpha = 0.6$")),
                     # for some reason the colouring doesn't match, so adjust
                     values = c("darkolivegreen3", "lightpink", "slateblue3"))+
  theme(legend.position = c(0.8, 0.2),
        legend.background = element_rect(fill="white",
                                         size=0.75, linetype="solid",
                                         colour ="black"))

### plotting log(N)
# creating the data frame
N.log <- log(1:N_max)
log.df.mean <- data.frame(N.log, data.mean.0.6)
names(log.df.mean) <- c("log.N", "mean.time.0.6")
# removing values zero from mean consensus time, since this adds nothing useful,
```

```r
# and removing these values creates a nicer graph
edit.log.df.mean <- log.df.mean[log.df.mean$mean.time.0.6 != 0,]

ggplot(edit.log.df.mean) +
  geom_point(aes(x = log.N, y = mean.time.0.6, colour = 'black')) +
  geom_smooth(aes(x = log.N, y = mean.time.0.6), method = lm, colour = "skyblue4",
              fill = "skyblue") +
  theme_bw() +
  labs(x = TeX("Logarithmic network size, $log(N)$"),
       y = TeX("Mean conensus time, $t_{N}(\\alpha)$"), colour = "Legend") +
  scale_color_manual(labels = c(TeX("Simulations, $\\alpha = 0.6$")),
                     # for some reason the colouring doesn't match, so we adjust it
                     values = c("midnightblue")) +
  theme(legend.position = c(0.2, 0.8),
        legend.background = element_rect(fill="white",
                                         size=0.75, linetype="solid",
                                         colour ="black"))



#####################################################

### function for simulations of majority model for different values of K
maj_sims_K <- function(q0,q1,alpha,N,nsims){
  K_end <- floor(N/2)
  time.sims <- matrix(nrow = nsims, ncol = length(seq(1,K_end)))
  sim.num <-matrix(nrow = nsims, ncol = length(seq(1,K_end)))
  end.sim <- matrix(nrow = nsims, ncol = length(seq(1,K_end)))
  mean.time <- vector()
  K.sims <- vector()
  K_start <- 1

  for (j in seq(1,K_end)){
    K.sims[j] <- j
    for (i in 1:nsims){
      a.sim <- majority_CTMC(q0,q1,N,alpha,j)
      time.sims[i,j] <- a.sim$time[length(a.sim$time)]
      end.sim[i,j] <- a.sim$X[length(a.sim$X)]
    }
    mean.time[j] <- mean(time.sims[,j])
  }

  #### making and editing data
  K.data <- vector()
  time.data <- vector()
  end.data <- vector()

  for (j in seq(1,K_end)) {
    K.data <- c(K.data, rep(j, nsims))
    time.data <- c(time.data, time.sims[,j])
    end.data <- c(end.data, end.sim[,j])
```

```
  }

  data.sims <- data.frame(K.data,
                          time.data,
                          end.data)
  return(data.sims)
}

########################################################
### Values
nsims <- 500
N <- 50
q0 <- 1
q1 <- 0.6
alpha <- 0.5
K_mean_sims <- maj_sims_K(q0,q1,alpha,N,nsims)


### making data frame for mean time
data.mean.K <- vector()

for (i in 1:floor(N/2)) {
  data.mean.K[i] <-  mean(K_mean_sims[K_mean_sims$K.data == i, 2])
}

df.mean.K <- data.frame(1:floor(N/2), data.mean.K)
names(df.mean.K) <- c("K.mean", "mean.time")

### plotting

ggplot(data = df.mean.K) +
  geom_point(mapping = aes(x=K.mean, y=mean.time)) +
  labs(x = TeX("$K$"), y = TeX("Mean conensus time, $t_{N}(\\alpha)$")) +
  theme_bw()

########################################################

### simulations for Exit probability for two values of alpha
q0 <- 1
q1 <- 0.6
K <- 1
N_steps <- 1
N_min <- 1
N_max <- 200
nsims <- 100
sims_0.2 <- maj_sims_N(q0,q1,0.2,K,N_max,nsims)
sims_0.6 <- maj_sims_N(q0,q1,0.6,K,N_max,nsims)

### exit probability from simulations w/ alpha=0.2
data.exitprob.0.2 <- vector()
for (i in N_min:N_max) {
```

```
  data.exitprob.0.2[i] <- length(sims_0.2$end.data[sims_0.2$end.data == i])/nsims
}
data.exitprob.0.2[1:5] <- NA

### exit probability from simulations w/ alpha=0.6
data.exitprob.0.6 <- vector()
for (i in N_min:N_max) {
  data.exitprob.0.6[i] <- length(sims_0.6$end.data[sims_0.6$end.data == i])/nsims
}
data.exitprob.0.6[1] <- NA

### data frame and plotting

N.exit.sims <- seq(from = N_min, to = N_max, by = N_steps)
df.exitprob.sims <- data.frame(
  N.exit.sims,
  data.exitprob.0.2,
  data.exitprob.0.6
)

df.exitprob.melt <- melt(df.exitprob.sims, id = "N.exit.sims")

# plotting data
ggplot(data = df.exitprob.melt,
       aes(x = N.exit.sims, y=value, colour = variable)) +
  geom_point() +
  labs(x = TeX("Network size, $N$"), y = TeX("Exit probability, $E_N(\\alpha)$"),
       colour = "Legend") +
  theme_bw() +
  scale_color_manual(labels = c(TeX("Simulations, $\\alpha = 0.2<\\beta$"),
                                TeX("Simulations, $\\alpha = 0.6>\\beta$")),
                     values = c("darkolivegreen3", "slateblue3")) +
  theme(legend.position = c(0.8, 0.5),
        legend.background = element_rect(fill="white",
                                         size=0.75, linetype="solid",
                                         colour ="black"))



### Plotting log(1-E_N) vs N to show growth is exponential
# simulations
q0 <- 1
q1 <- 0.6
K <- 1
N_steps <- 1
N_min <- 1
N_max <- 200
nsims <- 500
sims_0.6 <- maj_sims_N(q0,q1,0.6,K,N_max,nsims)

# make exit probability
```

```
data.exitprob.0.6 <- vector()
for (i in N_min:N_max) {
  data.exitprob.0.6[i] <-  length(sims_0.6$end.data[sims_0.6$end.data == i])/nsims
}
data.exitprob.0.6[1] <- NA

# make data frame
log.sim.exit.prob <- log(1-data.exitprob.0.6)
log.df.exit.prob <- data.frame(N.exit.sims, log.sim.exit.prob)
names(log.df.exit.prob) <- c("N", "log.sim.exit.prob")

# remove -infinity values, cleans up the data for plotting
edit.log.df.exit.prob <- log.df.exit.prob[log.df.exit.prob$log.sim.exit.prob != -Inf,]

# plotting
ggplot(data = edit.log.df.exit.prob) +
  geom_point(mapping = aes(x= N, y = log.sim.exit.prob, colour = 'black')) +
  geom_smooth(aes(x = N, y = log.sim.exit.prob, colour = 'red'), method = lm,
              colour = "skyblue4", fill = "skyblue") +
  labs(x = TeX("Network size, $N$"), y = TeX("$log(1-E_N(\\alpha))$"),
       colour = "Legend") +
  theme_bw() +
  scale_color_manual(labels = c(TeX("Simulations, $\\alpha = 0.6>\\beta$")),
                     values = c("midnightblue")) +
  theme(legend.position = c(0.8, 0.8),
        legend.background = element_rect(fill="white",
                                         size=0.75, linetype="solid",
                                         colour ="black"))


#####################################################

#### Function for simulations of majority model for different values of alpha
maj_sims_alpha <- function(q0,q1,K,N,nsims){
  N_min <- 1
  N_steps <- 1

  time.sims <- matrix(nrow = nsims, ncol = length(seq(0.1,0.9,0.05)))
  sim.num <-matrix(nrow = nsims, ncol = length(seq(0.1,0.9,0.05)))
  end.sim <- matrix(nrow = nsims, ncol = length(seq(0.1,0.9,0.05)))

  mean.time <- vector()
  alpha.data <- vector()
  time.data <- vector()
  end.data <- vector()
  alpha.sims <- vector()
  c<- 0
  for (j in seq(0.1,0.9,0.05)){
    c <- c+1
    alpha.sims[c] <- j
    for (i in 1:nsims){
```

```
      a.sim <- majority_CTMC(q0,q1,N,j,K)
      time.sims[i,c] <- a.sim$time[length(a.sim$time)]
      end.sim[i,c] <- a.sim$X[length(a.sim$X)]
    }
    mean.time[c] <- mean(time.sims[,c])
    alpha.data <- c(alpha.data, rep(j, nsims))
    time.data <- c(time.data, time.sims[,c])
    end.data <- c(end.data, end.sim[,c])
  }

  data.sims <- data.frame(alpha.data,
                          time.data,
                          end.data)
  return(data.sims)
}

##################################################
### Exit probability vs alpha for different values of N
# set up values and run simulations
q0 <- 1
q1 <- 0.6
K <- 1
nsims <- 500

sims_10 <- maj_sims_alpha(q0,q1,K,10,nsims)
sims_100 <- maj_sims_alpha(q0,q1,K,100,nsims)
sims_500 <- maj_sims_alpha(q0,q1,K,500,nsims)

# exit probability from simulations w/ N=10
data.exitprob.10 <- vector()
counter <- 0
for (i in seq(0.1,0.9,0.05)){
  counter <- counter +1
  data.exitprob.10[counter] <- length(sims_10$end.data[sims_10$end.data == 10
                                & sims_10$alpha.data == i])/nsims
}

# exit probability from simulations w/ N=100
data.exitprob.100 <- vector()
counter <- 0
for (i in seq(0.1,0.9,0.05)){
  counter <- counter +1
  data.exitprob.100[counter] <- length(sims_100$end.data[sims_100$end.data == 100
                                & sims_100$alpha.data == i])/nsims
}

# exit probability from simulations w/ N=500
data.exitprob.500 <- vector()
counter <- 0
for (i in seq(0.1,0.9,0.05)){
  counter <- counter +1
```

```
  data.exitprob.500[counter] <- length(sims_500$end.data[sims_500$end.data == 500
                                                & sims_500$alpha.data == i])/nsims
}

# data frame and plotting
alpha.exit.sims <- seq(0.1,0.9,0.05)
df.exitprob.alpha <- data.frame(
  alpha.exit.sims,
  data.exitprob.10,
  data.exitprob.100,
  data.exitprob.500
)

data_long <- melt(df.exitprob.alpha, id = "alpha.exit.sims")

ggplot(data_long,
       aes(x = alpha.exit.sims,
           y = value,
           color = variable)) +
  geom_line(size=1) +
  theme_bw() +
  geom_vline(xintercept = 0.375, linetype="dotted",
             color = "black", size=1) +
  annotate("text", x = 0.52,
           y = 0.255, label = TeX("$\\alpha = \\beta = 0.375$"), size=4,
           colour='black', face="bold") +
  geom_segment(aes(x = 0.45, y = 0.25,
                   xend = 0.375, yend = 0.25),
               colour='firebrick', size=1,arrow = arrow(length = unit(0.5, "cm"),
                                                      type = "closed")) +
  labs(x = TeX("Initial fraction of agents with opinion 1, $\\alpha$"),
       y = TeX("Exit probability, $E_N(\\alpha)$"),
       colour = "Legend") +
  scale_x_continuous(breaks=seq(0.1,0.9,0.1)) +
  scale_color_manual(labels = c(TeX("Simulations, $N = 10$"),
                                TeX("Simulations, $N=100$"),
                                TeX("Simulations, $N=500$")),
                     values = c("gold1", "slateblue3", "darkolivegreen4"))+
  theme(legend.position = c(0.8, 0.3),
        legend.background = element_rect(fill="white",
                                         size=0.75, linetype="solid",
                                         colour ="black"))
```