Averaging dynamics, mortal random walkers and information aggregation on graphs

Orowa Sikder
Department of Computer Science, University College London, London, WC1E 6BT, United Kingdom
* Author to whom any correspondence should be addressed.
E-mail: orowa.sikder.15@ucl.ac.uk

Keywords: random walks, network dynamics, information aggregation, network topology, degree distribution, random graphs

Abstract
Many models of dynamics on graphs implicitly capture some tendency towards alignment or synchronisation between neighbouring nodes. This mechanism often leads to a tension between independent dynamics at each node preserving local variation, and alignment between neighbours encouraging global smoothing. In this paper, we explore some of the intuition behind this phenomenon by considering a simplified set of dynamics where the states of agents are determined by a combination of private signals and averaging dynamics with their neighbours. We show that outcomes of this mechanism correspond to the behaviour of mortal random walkers on the graph, and that steady state outcomes are captured precisely in an object called the fundamental matrix, which summarises expected visitation between pairs of nodes. The bulk of the paper approximates the elements of the fundamental matrix as a function of the topology of the graph, in the case of undirected and unweighted graphs. In doing so we show intuitively how features such as degree distribution, community structure and clustering impact the trade-off between local variation and global smoothing in the outcomes, and can shed light on more complex instances of dynamics on graph. We consider as an application how the results can be used to predict and better understand the steady state outcomes of an information aggregation process.

1. Introduction
The field of network science has evolved to demonstrate that many seemingly unrelated processes can be well understood by a common framework of complex networks. For example, core ideas in graph theory have been applied to drive insight into opinion dynamics in social networks [1–3], the functioning of the brain [4, 5], the structure of the World Wide Web [6], the dynamics of financial networks [7, 8], and many more phenomena [9, 10].

The study of a diverse range of dynamics of graphs can often be framed in terms of a tug-of-war between local variation and global smoothing, where the topology of the graph can help determine the relative influence of these two effects. For example, the study of Ising models demonstrate how the dimensionality of the lattice determine the possibility of phase transitions between ordered and disordered phases [11]. In models of large systems of coupled oscillators, the topology of the coupling matrix help us uncover the synchronizability of the oscillators—increasingly connected networks are in general more synchronizable (see for example [12] for a demonstration in small world networks).

In general, we can consider two extremes: a completely isolated set of vertices and a fully connected graph. Dynamics on the former generally support individual variation, and the latter will generally favour homogeneity across outcomes for all nodes. The topology of the graph effectively interpolates between these two extremes through features such as connectivity, community structure, clustering, etc. This tug-of-war between localised variation and global smoothing can be exploited in analysing graph-based data in frameworks such as graph signal processing [13]. In such frameworks, the graph acts as an operator of sorts, and we can use methods such as a graph fourier transform to decompose the variation in the states of the nodes into the relative influence of global averages and higher order variation driven by the topology of the graph (for example, see [14]).
Such frameworks contribute in turn more recent developments such as graph convolutional networks [15] and graph neural networks [16], which implicitly balance local and global information to provide better predictions at each node.

This paper will explore this idea of local-global trade-off by considering an illustrative example of averaging dynamics over graphs. In particular, we show that for a certain characterisation of averaging dynamics (that we will formalise shortly), the steady state outcomes correspond precisely to the behaviour of ‘mortal’ random walkers. These are random walks on graphs that terminate arbitrarily with some given probability (also called evanescent random walkers) [17]. It turns out mortal walkers provide a much more intuitive framework to analyse our averaging dynamics of interest. Analysing this framework allows us to provide not only a deeper intuition as to the nature of the averaging dynamics, but also develop analytic expressions that relate topological properties of the graphs to the steady state outcomes of such processes. As far as the author is aware, this link between mortal random walkers and averaging dynamics is novel.

In our warm-up we introduce more precisely the averaging dynamics of interest, the definition of the mortal random walkers, and the ‘fundamental matrix’ that links these two phenomenon closely. For completeness, we also illustrate the relationship with a few close ideas in graph theory.

We then consider the main results, where we demonstrate how the behaviour of the mortal walkers (and implicitly the averaging dynamics) can be approximated very closely through the use of generating functions. We begin with the simplest example of an infinitely large tree, then show how we can extend these results by dropping simplifying assumptions (infiniteness and tree-ness).

In our final section, we consider an application to the study of information aggregation dynamics in social networks. We show how the results we have developed so far allow us not only to intuitively capture notions such as ‘echo chambers’ and ‘polarisation’, but how they interact with—and stem from—more fundamental features such as network topology.

1.1. Averaging dynamics with private signals

Simple averaging dynamics on graphs with private signals provide us with a very straightforward example of how the states of nodes are influenced over time by that of their neighbours. We define this formally as follows: for some undirected connected graph \(G\) with \(n\) nodes, let \(B\) be the adjacency matrix, such that \(B_{ij} = B_{ji} = 1\) if \(i,j\) are connected and \(0\) otherwise. Consider also the row-normalised adjacency matrix \(A = D^{-1}B\), where \(D\) is the diagonal matrix where \(D_{ii} = d_i\), the degree of the \(i\)th node. Each \(i \in G\) possesses some state \(x_i \in \mathbb{R}^d\), and updates this at each time step by interpolating between the states of their neighbours (denoted by the set \(N(i)\)) and a private signal \(b_i\). That is:

\[
x_i(t + 1) = \alpha \sum_{j \in N(i)} A_{ij}x_j(t) + (1 - \alpha)b_i
\]

\[
\Rightarrow x(t + 1) = \alpha Ax(t) + (1 - \alpha)b,
\]

where \(\alpha \in (0, 1)\) denotes the weight of ‘social’ update vs the ‘private’ or ‘local’ update (we can also absorb the term \((1 - \alpha)\) into \(b\) for a more general presentation, but it is useful for intuition to consider this weighted average format).

The dynamics described in equation (1) can be used to model a broad series of processes from opinion dynamics to swarm behaviour to games on networks (see for example a review in [18]). Of course, there are still many possible interaction dynamics that such linear rules will fail to capture (for instance, models where the states of the agents change the topology of the graph, such as bounded confidence models, dynamic communication networks, etc). However, the linear rules are sufficient for us to make precise our intuition about how the topology of the network can characterise the tug-of-war between local and global forces in shaping outcomes across a graph. As such, we use it as a stand-in for the more general phenomenon we are seeking to understand.

In our averaging dynamics, we can see that the parameter \(\alpha\) effectively captures the strength of network interactions. If it is large, then the states of agents are heavily influenced by their neighbours. If it is small, then the interaction is weak and the dynamics are dominated by private signals. In the case of specific models, this parameter \(\alpha\) can often be translated as some decay or imperfection in the interaction mechanism. We will see an example of this in the last section when we review an information aggregation model with imperfect transmission, where the parameter \(\alpha\) denotes the probability of a successful transmission.

The steady state outcomes of the dynamics in equation (1) are simply:

\[
x^* = (1 - \alpha)(I - \alpha A)^{-1}b = (1 - \alpha)F(\alpha)b = \tilde{F}(\alpha)b.
\]

Here, we use \(F = (I - \alpha A)^{-1}\) to denote what we call the ‘fundamental matrix’ and use the tilde to denote the row-normalized counterpart (i.e. all rows sum to 1; it is row stochastic). To see why the term \((1 - \alpha)\) achieves...
this normalization, note that since $A$ is a row stochastic matrix, it must have a leading eigenvector $\mathbb{1}$ with eigenvalue 1. It follows that the fundamental matrix $F = (I - \alpha A)^{-1}$ shares the eigenvector $\mathbb{1}$ with eigenvalue $(1 - \alpha)^{-1}$. In other words, all rows must sum to $(1 - \alpha)^{-1}$, and the term $(1 - \alpha)$ ensures this sums to 1 instead.

We utilise the term fundamental matrix from the corresponding idea in the theory of absorbing Markov chains [19]. Suppose we have some set of transient states in our chain, where the transition matrix submatrix is denoted $Q$, $Q$ is by definition a strictly sub-stochastic matrix, so we can define the fundamental matrix $F = (I - Q)^{-1}$, which captures the expected number of visits between transient states before the random walk escapes into the absorbing set [20].

In our example, since $A$ is a row stochastic matrix, $Q = \alpha A$ must be a sub-stochastic matrix for some $0 < \alpha < 1$, and can be thought of as a transient set of states in a larger set of states. More precisely, we can consider our graph in question to be a subgraph of a larger graph containing one or more ‘ghost nodes’—nodes that can be reached from the original graph but cannot be escaped once they are entered (the absorbing set). This conceptual framework often has a natural interpretation in models of interest. For example, we can allow each node $i$ to have their own private ghost node whose state is fixed at the private signal we are modelling $(b_i)$. In this case, visits by a Markov chain to this absorbing set draw a precise correspondence to the influence of the private signals on the averaging dynamics in the original model. The details of this correspondence are explored in considerable detail elsewhere (for example, see [18]), so we do not consider this analogy further, apart from noting that considered in this manner the fundamental matrix terminology is precise.

Equation (2) demonstrates how we can decompose the steady state outcome of the averaging dynamics into the effect of graph topology, summarised by the fundamental matrix $F(\alpha)$, and the distribution of private signals in $b$ (which we assume is non-trivial i.e. $b \neq \mathbb{1}$ where $\mathbb{1}$ is the vector of all 1s). We can see therefore that understanding the nature of the fundamental matrix can unlock deep insights as to the level of variation we might expect to see in the steady state of some dynamic process on a graph.

For example, if the graph is totally disconnected then $F(\alpha) = I$, meaning of course that the final steady states are just the local/private signals (i.e. $x^* = b$). On the other hand, if the graph is fully connected we can see that:

$$F(\alpha) = \alpha N^{-1} \mathbb{1}\mathbb{1}^T + (1 - \alpha)I.$$  

Here, $\mathbb{1}\mathbb{1}^T$ refers to the outer product of the vector of all ones, so $N^{-1} \mathbb{1}\mathbb{1}^T$ is a matrix with $\frac{1}{N}$ for every entry. Therefore:

$$x^* = F(\alpha)b = \alpha b1 + (1 - \alpha)b,$$

where $b = N^{-1} \mathbb{1}\mathbb{1}^Tb$ is the mean value of the private signals. That is, the steady state outcomes for the nodes will be weighted towards the global average, tempered by some individual variation. In other words, the outcomes would be smoother across the graph (relative to the disconnected graph). The smoothness grows with interaction strength $\alpha$. Proofs for the preceding comparisons are provided in appendix A.

As mentioned qualitatively in the introduction, other graph topologies provide some interpolation between these extremes. The question we seek to answer is exactly what this might look like, and estimating how $F$ varies with the topology is our key in doing so.

1.2. Mortal random walkers

In order to do so, we introduce the idea of mortal random walkers and demonstrate how they map precisely to our averaging dynamics.

Consider again our undirected connected graph $G$. A (simple) mortal random walk on such a graph proceeds as follows. Suppose the walk is at some node $i$. With probability $0 < (1 - \alpha) < 1$, the walk terminates. With probability $\alpha$, the walk picks a random neighbour of $i$ uniformly at random and steps to it. The transition probabilities between nodes can be summarised by the row-normalised matrix $A = D^{-1}B$. An immortal walker is simply one where $\alpha = 1$.

The fundamental matrix $F(\alpha)$ encodes a key property of a mortal random walker: the $(i,j)$th entry encodes the expected number of visits that a mortal walker that originates at site $i$ makes to site $j$ before it terminates. In order to see this, note that:

$$F(\alpha) = I + \alpha A + \alpha^2 A^2 + \cdots = \sum_{i=0}^{\infty} (\alpha A)^i = (I - \alpha A)^{-1}.$$  

Since $A$ captures the transition probabilities, $[A^i]_{ij}$ captures the probability of an immortal walk from $i$ hitting $j$ at time $t$, and the factor $\alpha^i$ captures the survival probability. Summing these across all times gives us our expected lifetime visits. The normalized quantity $\tilde{F}_0(\alpha)$ provides the expected fraction of visits that a walker starting from $i$ makes to $j$. 


This correspondence delivers a useful conceptual bridge between mortal random walks and the averaging dynamics\(^1\). Rewriting equation (2) we can see:

\[
x^*_i = \sum_j \tilde{F}_{ij} b_j = \sum_{d=0}^\infty \sum_{j \in \mathcal{N}(i,d)} \tilde{F}_{ij} b_j,
\]

where \(\mathcal{N}(i,d)\) denotes the neighbours of node \(i\) at distance \(d\) (where \(d\) is the shortest path length between \(i\) and \(j\)). It follows therefore that we can decompose the steady state outcomes for each node as the expected fraction of visits (since \(\tilde{F}\) is normalized) by a mortal walker from \(i\) to each node \(j\), and the contribution of the private signal from node \(j\), \(b_j\). If the weights \(\tilde{F}_{ij}\) are mostly allocated to a small set of nearby neighbours for each node \(i\), then each node’s steady state outcome is shaped mostly by its close neighbours and local variation is encouraged. On the other hand, if the weights are mostly allocated to distant neighbours, and given that there are many more nodes at greater distance, then weights are spread very thinly for a large set of neighbours across the graph. Since this holds for all nodes, this encourages global smoothing. We can see therefore that simple properties of the mortal walker behaviour can translate very quickly into deep intuition as to the outcomes of a dynamic process on the graph.

The key observation that relates the behaviour of the random walks with these dynamics is to note that the behaviour of random walks on an (unweighted) graph is effectively dictated by the number of possible pathways a random walk can traverse to get between different points. Similarly, the co-evolution of a pair or nodes during a dynamical process on a graph is often related to the number of causal pathways that exist between the nodes—if there are many possible pathways, then the influence they have on each other’s states is larger. The mortality of the random walk corresponds with the common sense notion that longer causal pathways in a dynamical process are weaker (replacing with immortal walkers both removes this desired property and extinguishes the possibility of long term variation\(^2\)). The dynamics we describe in the simple process in equation (1) are perhaps one of the simplest ways to formalize this notion, but we can see intuitively how this relates to a very general idea of dynamics on graphs.

1.3. The fundamental role of the fundamental matrix

As we can see, the fundamental matrix is the key object that allows us to map between the averaging dynamics and mortal walker framework. The importance nature of the fundamental matrix is of course not accidental, and it is an important concept that crops up throughout many applications of graph theory. It is worth highlighting some interesting examples.

We have already discussed the natural inheritance of the object from the study of Markov Chains. Closely related also is the ubiquitous notion of PageRank. The classic PageRank vector \(\pi\) is just:

\[
\pi = (1 - \alpha)(I - \alpha A^T)^{-1} \mathbb{1} = \tilde{F}(\alpha)^T \mathbb{1}.
\]

Here \(\alpha\) is the damping constant. For undirected graphs, it is generally remarked that the classic PageRank of each node just scales with the degree of the nodes [26]. Since we estimate \(F(\alpha)\) in full in our analysis, a by-product of our analysis is that we can refine this statement to show that the (expected) PageRank of each node is an affine function of its degree. The intercept and slope can be computed in closed form from \(\alpha\) and the mean degree of the network \(\langle k \rangle\). Details for this are provided in appendix B.

Fundamental matrices can also arise in Markov reward processes (MRPs) that are central in reinforcement learning and optimal control. An MRP is defined as a tuple \((\mathcal{S}, P, r, \alpha)\) where \(\mathcal{S}\) is a state space, \(P\) defines a Markov chain over the states, \(r(s)\) defines the expected immediate reward from hitting state \(s\), and \(\alpha\) is the discount applied to future rewards. A key idea is the value function \(V(s)\), which assigns to each state the expected discounted lifetime rewards of commencing in the state \(s\) and continuing with the dynamics of the MRP. We can use Bellman equations to recursively define this for the case where the state space is discrete and finite [27]:

\[
V(s) = r(s) + \alpha \sum_{s' \in \mathcal{S}} P(s'|s) V(s').
\]

1 As far as the author is aware, this explicit link between mortal random walks, the fundamental matrix, and averaging dynamics on arbitrary graphs is novel. While the study of mortal walkers is not new, it is only recently experiencing a surge in popularity, and many early studies focus either on general extensions of well-studied properties in immortal random walkers (i.e. first passage probabilities) [17, 21], very specific topologies [22], or specific applications to i.e. physical chemistry or biophysics [23, 24].

2 For completeness: for immortal walkers i.e. \(\alpha = 1\), there are no private signals, and the dynamics are purely driven by neighbourhood averaging i.e. the basic Degroot model [25]. In such a model with the undirected connected graph \(G\) persistent individual variation is impossible in the long run, so there is no heterogeneity of interest to analyse. In general, note that as \(\alpha \rightarrow 1\), many of the objects of discussion in this paper translate to their analogous objects in the study of traditional immortal random walks: the ‘fundamental matrix’ is replaced with \(\lim_{\alpha \to 1} A^t = \mathbb{1} \pi^\prime\), the stationary matrix of the associated Markov chain where \(\pi^\prime\) is the steady state visitation to node \(j\). The steady states of each node become \(x_i^* \to \sum_j \pi_j x_j(0)\forall j\), meaning total consensus across the nodes.
We can see this follows a similar format in matrix format:

\[ V = \alpha P V + r \]
\[ \Rightarrow V = (I - \alpha P)^{-1} r = F(\alpha) r \]
\[ \Rightarrow V(s) = \sum_{s'} F_{s,s'}(\alpha) r(s') . \] (9)

This means that we can intuitively think of the value function of state \( s \) as being equal to the expected number of visits made from \( s \) to \( s' \) under a mortal random walker with survival probability \( \alpha \), times the expected reward at each visit.

Knowledge of the fundamental matrix in principle therefore provides a shortcut to the value function\(^3\). Of course, in practical applications where the value function must be estimated, the transition matrix \( P \) is usually more complex than the simple random walk we focus on in this study. However, even a simplified approximation could provide for example a more accurate starting estimate before more rigorous methods are deployed.

Another important example arises in the study of the resolvent of matrices (see for example [29] for a review). The resolvent of matrix \( A \) is:

\[ G_A(z) = (A - zI)^{-1} = -z^{-1}F(z^{-1}) : z \in \mathbb{C} \setminus \sigma(A) , \] (10)

where \( \sigma(A) \) denotes the spectrum of \( A \). The resolvent is a deeply useful object in random matrix theory, as it encodes information about the eigenvalues and eigenvectors of the matrix \( A \). This is especially useful when \( A \) represents the (sparse) adjacency matrix of a graph. Methods to estimate the resolvent are developed in [30] for certain classes of tree-like graphs, but closed form solutions are unavailable for all but the most trivial structures.

The bulk of this paper will therefore seek to estimate the form of \( F(\alpha) \) for undirected graphs of varying topologies, and show how this can shed light on our intuitions about dynamics on graphs. We use the mortal walker interpretation in order to do this, as it allows us to estimate the fundamental matrix by simply reasoning about the expected visitation behaviour of these random walks across a graph.

Our general strategy to characterise the behaviour of these mortal walks is to estimate firstly the distribution of visits at various distances from some arbitrary root. That is, we estimate on average the number of visits a mortal walker has on the set of nodes at some distance \( d \) from an arbitrary root, for all \( d \). We then ‘distribute’ these visits amongst the set of nodes at distance \( d \). In order to carry out these two steps, we develop a general approach to treat the neighbours of a node at each distance \( d \) as a single ‘layer’ that we collapse into a single representative node. We are left with a chain with certain transition probabilities between these ‘layers’, then solve for the steady state distribution using a generating function approach. We first consider a baseline case where our target graph is arbitrarily large and a tree. We show that this results in an accurate prediction of mortal random walk visits even for graphs that do not strictly adhere to the stated assumptions. We then relax each assumption explicitly, first demonstrating when and how the behaviour changes under finite size effects, and then how it changes for graphs that violate the tree assumptions. Finally, we leverage the insight we have gained in terms of the behaviour of the mortal walkers to solve a simple model of information aggregation over graphs. We show that the ideas developed allow us to quickly estimate how real-life models of dynamics on networks might behave, through the lens of approximating the relative strength of local and global effects.

### 2. Estimating steady state outcomes

We will build up our estimate of the fundamental matrix in steps by considering the behaviour of the mortal random walkers on graphs with different constraints. Following the main set of results, we will seek to relax these constraints to see the effects. In doing so, we will be able to build up a picture of when our baseline approximations are robust, and when they need modification to handle specific cases.

We will first consider the simplest case where the graph of interest \( G \) is an arbitrarily large tree with uncorrelated degrees, but still possesses some given degree distribution \( P(k) \). This affords analytic tractability while retaining important features such as degree heterogeneity. Note this is approximately equivalent to a configuration model with degree distribution \( P(k) \) as \( N \to \infty \), but we use an exact tree instead to be a bit neater in our derivations.

\(^3\) In fact, recent algorithms such as expected eligibility traces [28] implicitly try to estimate the elements of \( F \) to speed up learning (personal communication, authors of [28]).
As before, we define the mortal random walkers as walkers that at each time step either terminate with probability \((1 - \alpha)\), or with probability \(\alpha\) pick a random neighbour uniformly at random and walk to it. Our objective is to estimate the expected number of visits a walker that begins at a node \(i\) will make to some target node \(j\), which as we have shown corresponds to \(\tilde{F}_{ij}\).

2.1. Mapping to biased random walks on a chain

For an arbitrary root node \(i\), we can partition the remaining nodes \(j \in G\) into neighbours at various distances (with the distance between \(i\) and \(j\) denoting the length of the shortest path between them). We will focus for now on an intermediate problem, which is to estimate the total expected number of random walk visitations from a root node \(i\) to all nodes at some layer \(d\). Denote this value \(\tau(d, \alpha)\) (we sometimes drop the reference to \(\alpha\) for brevity). Once we have established this, we can 'distribute' the visitations between all nodes at this layer. For convenience, we also define \(\tilde{\tau}(d) = (1 - \alpha)\tau(d)\) which like the corresponding quantity with the fundamental matrices normalizes the total visits so that \(\tilde{\tau}(d)\) represents the fraction of visits at distance \(d\).

We begin by collapsing all the nodes at distance \(d\) into a single node representing that layer (as illustrated in figure 1). The key observation which makes this tractable is the calculation of the transition probability between the ‘layers’ in the now (infinite) one-dimensional chain representing our graph. Suppose the random walk is currently at some node \(j\) in layer \(d \geq 1\). We wish to compute the probability of moving ‘up’ the chain (i.e. closer to the origin):

\[
\mathbb{P}[d \rightarrow d - 1] = \sum_{k_j} P(k_j) P(d \rightarrow d - 1 | k_j),
\]

where \(P(k_j)\) represents the degree distribution of the node \(j\). Recall that since \(G\) is uncorrelated, and the node \(j\) is reached via a random walk, the degree distribution of \(j\) will be skewed, and is represented as \(P(k_j) = \frac{k_j P(k_j)}{\langle k \rangle}\), where \(\langle k \rangle\) is the mean degree of \(G\) (see for example [31]). Secondly, since the graph is a tree, the node \(j\) will have only a single edge to traverse back up the chain, out of its total edges \(k_j\). Putting these together, we get:

\[
\mathbb{P}[d \rightarrow d - 1] = \sum_{k_j} \frac{k_j P(k_j)}{\langle k \rangle} \frac{1}{k_j} \sum_{k_j} P(k_j) = \frac{1}{\langle k \rangle}.
\]

The probability of moving ‘down’ the chain is simply \(\langle k \rangle - 1\), since by the tree assumption, no edges exist to other nodes in the same layer. This result holds at any point in the chain (barring the origin). As a result, we can see the problem takes on a very simple and well-studied phenomenon: a biased random walk on a chain with a reflection at the origin. Let \(p_{d,t}\) represent the probability that a random walk is at layer \(d\) at time \(t\). We can then define the generating function(s):

\[
G_d(z) = \sum_{t=0}^{\infty} p_{d,t} z^t.
\]

Crucially, note that for \(z = \alpha\):

\[
G_d(\alpha) = \sum_{t=0}^{\infty} \alpha^t p_{d,t} = \mathbb{E}[\text{total visits at } d|\alpha] = \tau(d, \alpha).
\]
In other words, we can compute our desired quantity \( \tau(d, \alpha) = G_d(\alpha) \) without having to compute the intermediate probabilities \( P_{d \alpha} \) at all! Together with the boundary conditions \( P_{d \alpha} = \delta_{d,0} \), we can solve the generating functions (see appendix C for details) to obtain:

\[
\tau(d, \alpha) = \begin{cases} 
G_0(\alpha) & \text{for } d = 0 \\
G_1(\alpha)C(\alpha)^{d-1} & \text{for } d \geq 1,
\end{cases}
\]  \hspace{1cm} (15)

where the generating functions \( G_d \) and decay term \( C \) can be obtained in closed form:

\[
G_0(z, \langle k \rangle) = \frac{2(\langle k \rangle - 1)}{(\langle k \rangle s(z, \langle k \rangle) + 1) - 2}
\]  \hspace{1cm} (16)

\[
G_1(z, \langle k \rangle) = \frac{2(k)z}{(\langle k \rangle s(z, \langle k \rangle) + 1) - 2z^2}
\]  \hspace{1cm} (17)

\[
s(z, \langle k \rangle) = \sqrt{1 - \frac{4((\langle k \rangle - 1)z^2}{\langle k \rangle^2}}
\]  \hspace{1cm} (18)

\[
C(z, \langle k \rangle) = \frac{(\langle k \rangle - 1) - s(z, \langle k \rangle)}{2z}
\]  \hspace{1cm} (19)

In order to sense check this result, we perform numerical simulations and record the number of visits by random walks from some random root node to other nodes at various distances. The results are provided in figure 2, for a variety of networks, average degrees \( \langle k \rangle \), and exit probabilities \( \alpha \).

We can see that the results are quite robust, even though some of these networks do not follow the simplifying tree assumption (in particular the BA network, which can possess a non-trivial clustering coefficient). Our results are most robust for small \( d \), but some discrepancies exist for larger \( d \). This discrepancy arises because our derivation depends on an assumption of the asymptotic limit \( N \to \infty \), which results in an infinite one-dimensional chain. Of course, in reality, our networks are finite. Therefore, there is a ‘bounce back’ effect when random walks reach the end of the (finite) chain, where the position of the discrepancy is determined by the expected ‘radius’ of the network, denoted with a blue dashed line. It turns out for our estimation of the elements of \( F(\alpha) \), these small discrepancies contribute very little to the overall accuracy, so we will ignore them for now. In section 2.3 we revisit these finite size effects and graph radius in more detail, and how we can correct for them by explicitly modelling a finite chain.

### 2.2. Expected visits between specific nodes

Once we have our expression for the total expected hits at distance \( d \), we can approximate the elements of the fundamental matrix \( F(\alpha) \) as follows, given the node degrees \( k_i \):

\[
F_{ij} \approx \frac{\tau(d(i,j))}{\mathbb{E}[|N(i, d(i,j))]|]}
\]

\[
\Rightarrow F_{ij} = \begin{cases} 
\tau(0) & \text{for } i = j \\
\frac{\tau(d(i,j))}{k_i h^{d(i,j)-1}} & \text{otherwise},
\end{cases}
\]  \hspace{1cm} (20)

where in the denominator we have made use of the fact that the expected size of the \( d \)th layer from the root node with degree \( k_i \) in a tree scales with the expected excess degree \( h = \frac{\langle k^2 \rangle - \langle k \rangle}{\langle k \rangle} \), which effectively measures the degree heterogeneity of the graph [31].

It is also useful to compute the expected column sums of the matrix \( F \). The \( i \)th column sum \( F_i \) denotes the total expected number of random walks that arrive at the node \( i \) from all other nodes (this is precisely the classic PageRank score of each node):

\[
F_i = F_{ii} + \sum_{d=1}^{\infty} \sum_{j \in N(i, d)} F_{ij} = F_{ii} + \sum_{d=1}^{\infty} \frac{k_i}{k_j} \sum_{j \in N(i, d)} F_{ij}
\]

\[
\approx \tau(0) + \frac{k_i}{\langle k \rangle} \sum_{d=1}^{\infty} \sum_{j \in N(i, d)} F_{ij} \approx \tau(0) + \frac{k_i}{\langle k \rangle} \sum_{d=1}^{\infty} \tau(d)
\]

\[
= \tau(0) + \frac{k_i}{\langle k \rangle} \left( \frac{1}{1 - \alpha} - \tau(0) \right).
\]  \hspace{1cm} (23)
Figure 2. Comparison of numerical simulations vs theory of the total number of visits by a mortal random walk to nodes at distance \(d\) from the origin node. Crosses indicate the mean number of hits, with the intervals denote one standard deviation. The blue dashed line denotes the expected radius of the network (see section 2.3 for a derivation). A variety of networks (Erdos–Renyi (ER), \(k\)-regular (KREG), Barabasi–Albert (BA), stochastic block model (SBM) respectively), average degree \(\langle k \rangle\) and exit probability \(\alpha\) are depicted. In all cases, \(N = 2000\). For SBM networks, the graph is built from two blocks of size \(N/2\), within block connection probability \(p_{\text{in}} = 3 \langle k \rangle / N\) and between block \(p_{\text{out}} = p_{\text{in}}/3\).

In the second step we made use of the fact that \(F_{ij} = \frac{m}{k_j} F_{ij}\), which follows from the fact that any possible path \((i \rightarrow a \rightarrow \cdots \rightarrow z \rightarrow j)\), when reversed, is identical to the transition probabilities in the intermediate points \((a \leftrightarrow z)\) and only needs to be corrected by reversing the direction on the endpoints \(i, j\). In the third step, we approximate \(\frac{1}{k_j}\) with its expected value. Because the node \(j\) is reached via random walk, \(\mathbb{E}[\frac{1}{k_j}] = \sum_k \frac{k P(k)}{\langle k \rangle} = \frac{1}{\langle k \rangle}\). The final step makes use of the fact that the sum of expected random walk visits to all distances will just be the expected lifetime of a random walk with termination probability \((1 - \alpha)\), which is just the expected value of a geometric distribution \((\frac{1}{1-\alpha})\).

We verify these results numerically in figure 3, where we consider the elements \(F_{ij}\) for \(d(i, j) \in \{1, 2, 3\}\) for the same networks and combinations of \(\langle k \rangle\) and \(\alpha\) as used in figure 1 (barring the KREG network, since there is no degree heterogeneity to depict). In all cases, we can see that the approximations are quite accurate. We also include the column sums against node degrees \(k_i\) for the SBM network, and can see that the theory closely matches prediction.

In order to avoid cherry-picking specific instances, we compute some measures of error for the matrix \(F(\alpha)\). We consider the mean squared error (MSE) for the elements \(F_{ij}\). That is \(\text{MSE} = \frac{1}{N^2} \sum_{i,j} (1 - \alpha)(F_{ij} - \tilde{F}_{ij})^2\), where the scaling by \((1 - \alpha)\) ensures that all examples we consider have rows that sum to 1 and ensures that different matrices are comparable. We also compare the Frobenius norms of the true matrix and the estimated matrix and take the absolute percentage error (APE). That is \(\text{APE} = \frac{\|\tilde{F}\|_F - \|F\|_F}{\|F\|_F}\). We computed both these

\[^4\text{For } d = 0, F_{ii} = \tau(0)\text{ which can be verified in figure 2.}\]
Figure 3. Comparison of numerical simulations vs theory for the elements $F_{ij}$ for nodes at distance $d(i,j) \in \{1, 2, 3\}$. Crosses indicate the mean number of hits, with the intervals denote one standard deviation. The colour denotes the distance $d(i,j)$. A variety of networks (ER, small world, BA, respectively), average degree $\langle k \rangle$ and exit probability $\alpha$ are depicted. On the right-hand side, we also denote $F_i$ against $k_i$ for the corresponding SBM network. In all cases, $N = 2000$.

As we can see, our results are quite accurate in general, both in terms of individual elements (MSE) and the matrix norm (APE). Since the Fundamental Matrix approximation can be computed in $O(N^2)$ instead of $O(N^3)$ for the exact expression, this represents a cost saving if one is interested in utilising the fundamental matrix in computation. Of greater utility, however, is the insight it provides us with regard to how the fundamental matrix and therefore behaviour of random walks is affected by changes in the interaction distance (modulated by $\alpha$) and the degree distribution (modulated by $\langle k \rangle$ and $\langle k^2 \rangle$).

For example, consider the expected fraction of visits occurring at each layer $\tilde{\tau}(d, \alpha) = (1 - \alpha)\tau(d, \alpha)$ and how this varies as $\alpha$ and $\langle k \rangle$ vary. How does the behaviour of the mortal walkers change as survival probability $\alpha$ changes? Taking the partial derivative, it can be shown that:

$$\frac{\partial \tilde{\tau}(d, \alpha)}{\partial \alpha} \geq 0 \iff d \geq \frac{\tilde{\tau}(1)}{(1 - C(\alpha))^2} = R_W. \quad (24)$$

That is to say, for all neighbours outside the ‘random walk radius’ $R_W$ (we will see why we chose this terminology shortly), increasing the survival probability $\alpha$ will increase the expected fraction of random walk visits. For all neighbours within this radius, however, increasing $\alpha$ will reduce the expected fraction of visits. Correspondingly, if we are interested in the averaging dynamics on the graph instead, this tells us that an increased interaction strength $\alpha$ will allow for more influence from nodes that are further away. Since there are more nodes in further layers, this means that each node is influenced by a very large set of neighbours, and the process results in more homogeneous outcomes.
The role of $\langle k \rangle$ is also informative. For small $\langle k \rangle$, the probability of ‘reverting’ up the chain (i.e. closer towards the root node) increases, and as a result we can see that the visitations are more heavily skewed towards the local neighbourhood of the nodes. As a simple illustration, consider a KREG graph with $\langle k \rangle = 3$. In this case, at each step, there is a 1 in 3 chance that a random walk moves ‘up’ the tree back towards the root. If $\langle k \rangle = 100$ on the other hand, the chances of reverting back to the root 1 in 100. In other words, the less connected the network, the more likely a random walker will stay close to the origin, as it will keep reverting back up the tree instead of exploring new layers of the tree.

Furthermore, each layer consists of fewer neighbours, so the random walk visits are distributed amongst a smaller set, so the visits to each nearby neighbour increases. In the context of our averaging dynamics we therefore get a very intuitive result: as the connectivity of the graph decreases, the influence of each node’s local neighbours increases considerably, and local variation increases.

Finally, $\langle k^2 \rangle$ interestingly has no effect on the distribution across layers $\tilde{\tau}(d)$, but it does determine how populated each layer is in expectation, and therefore how thinly spread the visits are. Therefore, we can see that for more heterogeneous graphs, each node is influenced by a much larger set of neighbours, which encourages more global smoothing.

### 2.3. Finite size effects and the random walk radius

In the previous sections, we have leveraged the assumption of arbitrarily large networks to provide a tractable analytic approximation for the fundamental matrix. We showed that this approximation was robust even when the true network was finite. As we hinted in figure 2, however, this approach has its limitations when the network is small enough for the mortal walker to ‘bounce back’ from the end of the network, adding discrepancies. In fact, the maximum error that occurs in table 1 occurs precisely for networks with large $\langle k \rangle$, large $\alpha$ and small $N$.

There are two preliminary concepts to introduce to analyse this formally. We first consider the expected radius of the graph $(R_{D})$, which we have alluded to earlier in figure 2. We define this to be the expected number of layers required to reach the whole graph, starting from a random source node. With the assumption of the tree, the expected number of neighbours at distance $d \geq 1$ take a convenient form $\langle k \rangle h^{d-1}$, meaning:

$$
\mathbb{E}[\text{neighbors upto } D] = \langle k \rangle \sum_{d=1}^{D} h^{d-1} = \langle k \rangle h^D - \frac{1}{h - 1} = N
$$

$$
\Rightarrow R_{D} = \frac{\log((h - 1)N/\langle k \rangle + 1)}{\log(h)}.
$$

The quantity $R_{D}$ effectively tells us the approximate length of the finite chain representation of the graph, and is purely a function of graph features. Note that as $N \rightarrow \infty$, this will grow arbitrarily large, so long as $h$ is finite.

Another phenomenon to consider is the expected exit layer of the random walk on the biased chain. This can be computed as:

$$
\mathbb{E}[\text{final layer of walk}] = \frac{\tilde{\tau}(1)}{(1 - C(\alpha))^2} = R_{W}. \tag{26}
$$

| Network | $N$  | MSE (max) | APE (max) |
|---------|------|-----------|-----------|
| BA      | 100  | $5 \times 10^{-6}(4 \times 10^{-5})$ | 1.7% (5.0%) |
|         | 1000 | $3 \times 10^{-7}(2 \times 10^{-6})$ | 0.8% (1.2%) |
|         | 2000 | $1 \times 10^{-7}(1 \times 10^{-6})$ | 0.8% (1.4%) |
| ER      | 100  | $3 \times 10^{-6}(2 \times 10^{-5})$ | 1.2% (4.2%) |
|         | 1000 | $8 \times 10^{-8}(1 \times 10^{-6})$ | 0.2% (0.6%) |
|         | 2000 | $3 \times 10^{-8}(4 \times 10^{-7})$ | 0.1% (0.3%) |
| KREG    | 100  | $2 \times 10^{-6}(2 \times 10^{-5})$ | 1.1% (4.1%) |
|         | 1000 | $1 \times 10^{-8}(1 \times 10^{-6})$ | 0.1% (0.4%) |
|         | 2000 | $3 \times 10^{-10}(2 \times 10^{-8})$ | 0.1% (0.2%) |
| SBM     | 100  | $3 \times 10^{-6}(2 \times 10^{-5})$ | 1.3% (4.6%) |
|         | 1000 | $8 \times 10^{-8}(1 \times 10^{-6})$ | 0.2% (0.9%) |
|         | 2000 | $4 \times 10^{-8}(5 \times 10^{-7})$ | 0.1% (0.7%) |

Table 1. Verification that our estimate of the fundamental matrix produces accurate results. For each network type, we generated graphs at three scales $\langle N \rangle \in \{100, 1000, 2000\}$ and for each instance we generated all combinations of $\langle k \rangle \in \{4, 6, 8, 10\}$ and $\alpha \in \{0.25, 0.3, 0.75\}$. We report the median MSE and APE in all cases, with the maximum MSE and APE in brackets. The minimum in all cases is virtually zero (not included). Generally the larger the graph, the more accurate the approximation.
Figure 4. Comparison of predictions by infinite chain (red) and finite chain (blue). Crosses indicate average visitations on graphs with parameters outlined in main text. The blue dashed line is $R_G$ and the green dashed line is $R_W$.

(The full derivation is provided in appendix D.) Here we can see that the expected exit layer is precisely the ‘random walk radius’ we introduced in the previous section. This result is verified numerically in the appendix.

We can use the quantities $R_G$ and $R_W$ to characterise when finite size effects have a bigger impact in our estimate of the random walk behaviour. Clearly, when $R_W \ll R_G$, the random walk terminates on average far before reaching the end of the chain, and our infinite chain estimate is appropriate. However, if the two radii are close (or $R_W > R_G$), then the ‘bounce back’ effect will inflate the number of visits to layers around the graph radius.

In order to sense check this, we can compute analytically the distribution of visits for a random walk on a finite chain with a reflection on both ends. One easy way to do this is to exploit the fact that the transition probabilities of this finite chain of length $D$ can be written as a tri-diagonal matrix, for which analytic solutions are available for inversion [32]. Details including the full closed form solutions are provided in the appendix.

For the sake of illustration, we consider two examples in figure 4. Here, we consider a BA graph ($N = 1000, \alpha = 0.2, \langle k \rangle = 6$) and ER ($N = 1000, \alpha = 0.2, \langle k \rangle = 8$). In each case, we denote $R_G$ and $R_W$ as the blue and green dashed vertical lines, respectively. We can see that the two radii are very close, and as such there is considerable ‘bounce back’, inflating the visits to intermediate layers as indicated by numerical simulations (crosses). The original infinite chain prediction is in red, whereas the adjusted finite chain prediction is in blue. We can see that the approximation improves slightly, but not perfectly. This imperfection arises in part because we are forced to discretize the graph radius $R_G$ in order to compute the analytic terms. Regardless, the finite size correction provides a small improvement to our estimates (for the BA graph, the APE and MSE fall by $\sim 20\%$ and $\sim 2\%$ and for the ER graph, the APE and MSE fall by $\sim 3\%$ and $\sim 8\%$).

We can also consider how changes in the parameters affect the visitation behaviour. Consider for example the expected fraction of visits at the origin ($\tilde{\tau}(0)$). It can be shown easily that as $R_G$ increases, this quantity falls. That is to say, as the graph becomes larger, the (mortal) random walk is less likely to return to the origin. This is because as the graph grows, the bounce-back is limited, which reduces visits back to the source. In the context of averaging dynamics, this means that processes on smaller graphs can more easily support localized variation.

2.4. Non tree-like effects

So far we have assumed our graphs are trees (or at least tree-like), and we now consider the effect of violating this assumption. In particular, for the sake of illustration, we will consider a lattice structure as follows. Suppose we have a one-dimensional lattice where nodes are connected to their $m$ closest neighbours. If $m = 1$ we simply have a one-dimensional chain (which is a tree), but as $m > 1$, we generate a graph with a high clustering coefficient (specifically, the global clustering coefficients, denoted with $\gamma_{G}$, grows with $m$ as $\frac{3m-1}{4(2m-1)}$).

5 The results here can be extended straightforwardly to higher dimension lattices by condensing each dimension as a separate chain and rebalancing the transition probabilities.
Examining the behaviour of mortal random walks on such graphs help illustrate the effect of such topological features in general.

In order to solve for such clustered graphs, we can once again map the layers of neighbours from an arbitrary root into a chain. We can then leverage the regularity of the lattice structure to compute the transition probabilities between layers. By simply enumerating the number of edges within and between layers and leveraging symmetry, it is straightforward to show that the transition probabilities are:

\[
P[d \to d - 1] = \frac{(m + 1)}{4m}, \quad (27)
\]
\[
P[d \to d] = \frac{(m - 1)}{2m}, \quad (28)
\]
\[
P[d \to d + 1] = \frac{(m + 1)}{4m}. \quad (29)
\]

Notice here that the positive clustering coefficient means that random walk can remain in the same layer, which we did not have for trees. Once again, we can solve for the expected number of visits explicitly using a generating function approach:

\[
\tau_L(d) = \begin{cases} 
H_0(\alpha) & \text{for } d = 0 \\
H_1(\alpha)D(\alpha)^{d-1} & \text{for } d \geq 1,
\end{cases}
\quad (30)
\]

where the details including closed forms are provided in the appendix. We compare the tree and lattice theory with tree and lattice numerical simulations in figure 5. We can see that the violation of the tree-like assumption means the random walk circulates much more heavily within the immediate neighbourhood of the root node, as one would expect. Furthermore, from the full expression for equation (30) in the appendix see that as \( m \) increases (and alongside, the clustering coefficient), the weight at distances 0 and 1 increases monotonically.

For the general case, note that the one-dimensional lattice model and the baseline ER model (with the same mean degree) represent the two extremes of the rewiring parameterisation of a small world network. For intermediate levels of rewiring the clustering coefficient is not as strong, and we might expect the two solutions to interpolate, which we denote with circular markers in figure 5 for rewiring \( r = 0.1 \). We can see the visitations interpolate precisely between the lattice and the tree, as expected.

Similarly to before, we can show how the random walk behaviour changes as the parameters change. For example, as the number of neighbours \( m \) (and alongside, the clustering coefficient) increases, we can see that the fraction of visits to the first layer \( \tilde{v}(1) \) will always increase, but actually decrease for the returns to the origin \( \tilde{v}(0) \). Intuitively this is because as the clustering coefficient increases, a random walk can become ‘trapped’ at the first set of neighbours for a long time, increasing visitations amongst these nodes. Overall, however, it can be shown that relative visits to distances 0 and 1 are always higher for the lattice than the corresponding tree-like baseline with the same mean degree. In the language of our averaging dynamics, this means that graphs with
Figure 6. Illustration of the general idea to determine steady state influence in the averaging dynamics. (Top left) We use the network topology to get summary features such as degree heterogeneity and clustering. Top right: we attempt to translate these to transition probabilities between layers of a chain representing layers of neighbours around nodes, which we can solve for via generating functions. (Bottom) We sketch the steady state ‘influence’ through the size and vertical elevation of the neighbours at each distance from a central node (yellow), with the influence decaying from the centre out. The ‘forward probability’ \(a\) pushes influence further out from the centre, whereas the ‘retention’ and ‘backward’ probability \(b + c\) bolster the influence of local agents. The graph radius \(R_G\) is illustrated as a ‘bounce-back’ effect distinct from these dynamics that folds influence back towards the intermediate layers.

larger clustering coefficients will have nodes that are more heavily influenced by their immediate neighbours, encouraging local variation.

2.5. General method

We briefly summarize what we have covered so far. Our objective was to understand how the network topology determines the steady state outcomes of an averaging dynamic. We illustrate the general idea in figure 6. We begin with a network from which we take some summary topological features such as degree distribution (i.e. \(\langle k^2 \rangle, \langle k \rangle \)), the global clustering coefficient (i.e. \(\gamma_G\)) and perhaps even the size of the network or segregated communities in the graph (i.e. \(N\)). We use these features to try to determine a set of transition probabilities for random walks between nodes at various distances to each other (for generality, we refer to the forward probability \(P[d \rightarrow d + 1]\) as \(a\), the recurrent probability \(P[d \rightarrow d]\) as \(b\) and the backward probability \(P[d \rightarrow d - 1]\) as \(c\)).

If we can construct appropriate transition probabilities, then the steady state outcomes of the averaging process can be solved via generating functions (a general solution is provided in the appendix). Critically, the distribution of visits by walkers over various distances tells us the influence that nodes at this distance have on the steady state outcome. If most of the visits are to nearby nodes, then most of the influence is local, and agents will have steady state outcomes that allow for local variation. If more of the visits are further away, then there is long-range influence, and global smoothing occurs.

The transition probabilities dictate where the influence lies: intuitively, the ‘forward’ probability \(a\) will push visits (and influence) further and further away, and conversely the ‘backward’ and ‘retention’ probabilities \(b + c\)
pull influence closer to the local area. The finite size effects are distinct from these transition probabilities, and push influence away from distant nodes back towards the centre.

One particularly useful graph we can use to summarise this process is a ‘clustered random graph’ we developed for the purpose of this illustration. The graph in question starts with an existing random graph without any specific clustering (KREG, ER, BA, etc), then rewires edges if and only if they increase the clustering while maintaining the degree of each node. This allows us to construct a graph with a desired degree distribution as well as clustering coefficient. Details and verification are provided in the appendix; we focus here on intuition.

It can be shown for such a graph that:

\[ b + c = \frac{\gamma_G (\langle k^2 \rangle / \langle k \rangle - 1) + 1}{\langle k \rangle} \approx \frac{\gamma_G \langle k^2 \rangle}{\langle k \rangle^2} \]

(31)

\[ a = 1 - (b + c) \approx 1 - \frac{\gamma_G \langle k^2 \rangle}{\langle k \rangle^2} \]

(32)

This allows us to summarize the general ideas quite succinctly: in such a graph we can see that for a fixed mean degree, increasing clustering and degree heterogeneity (as captured by \( \langle k^2 \rangle \)) will increase the retention/backward probability of walkers, ensuring they circulate more locally. In turn, this implies the steady state outcomes will be weighted heavily towards local nodes. In short, degree heterogeneity and clustering will act to reduce global smoothing for the averaging process in this specific graph.

Illustration of the relative influence of nodes at each distance are provided in figure 7 (we symmetrize the graph for the sake of visual intuition). Here, we started with an ER graph with no transitivity and little degree heterogeneity (\( \langle k^2 \rangle = 21, \langle k \rangle = 4 \)) (green). The forward probability is therefore \( a \approx 0.75 \), pushing influence away from the local nodes and towards further nodes. We then increase the clustering coefficient to 0.17 while maintaining the degree distribution (red). This reduces the forward probability to around 0.6 and pulls the influence closer to the nearby distances. Finally, we take a BA network with much higher degree heterogeneity to start with (\( \langle k^2 \rangle = 32, \langle k \rangle = 4 \)) and increased transitivity to 0.17 also. We can see the forward probability falls even further to 0.4 and the influence is concentrated on the immediate neighbours with a visual ‘valley effect’ (the influence of the distance 1 neighbours higher than the influence of the source node’s own private signals).
3. Applications to information aggregation

We now consider a specific application in the form of a simple model of information aggregation over a graph. Consider a set of $N$ agents connected over some graph $\mathcal{G}$ where each agent $i$ accurses a set of signals in an information set $I_i(t)$. Each agent acquires a new signal at each time step in one of two ways. With probability $(1 - \alpha)$, the agent draws a signal from a private information source, which takes a value $b_i$. Instead, with probability $\alpha$ they sample the latest signal of a random neighbour.

The following intuition is useful. Suppose each agent represents a user learning news about the world. At each time step they can either look at their preferred news source (private information), or they can have a conversation with a random friend, who passes on the latest bit of information they have learnt. Let us suppose an agent’s private news source always generates signals of either $b_i = +1$ or $b_i = -1$. The orientation of an agent’s news source is determined i.i.d. at $t = 0$ and is positive with probability $p$. The results that follow can be easily generalised to a case where the private information sources themselves generate signals according to some arbitrary distribution, and the distributions are generated hierarchically, but we leave the details of this to the appendix.

For large $t$, we can see that each signal that arrives at the node $i$ can be modelled as a mortal walker. To see this, consider the process from two directions. Moving forward in time, it looks as if signals are generated from private information sources, then diffuses via the network, with a copy deposited in the information set of each node it visits. However, consider moving backward in time for a node that has just acquired a signal. We ask a ‘contact tracing’ question: where did this signal come from?

For example, suppose at time step $t$ some signal $y$ arrives at node $u$. This means at the beginning of $t$, node $u$ sampled a signal from one neighbour $v$ of $u$ (uniformly at random from all of $u$’s neighbours). At $t - 1$, suppose that node $v$ received that signal in question from another neighbour $w$ (picked uniformly at random from $v$’s neighbours). Eventually, going back enough steps, we will reach the original node where the signal was sampled from the private information source and not via a neighbour. With the clock running backwards, the signal will look like it ‘terminated’ upon reaching this source (as opposed to originating from it in forward view). Each step it made in the backward run has followed the transition probabilities we outlined for a random walker (i.e. the walker moves to a random neighbour for node $u$ with probability $\frac{1}{k_u}$, where $k_u$ is $u$’s degree). Furthermore, each node could have been a ‘source’ (and therefore terminated in backward view) with probability $(1 - \alpha)$. In short, when we model the process backwards we are just dealing with mortal random walkers again, and all of our analysis from the previous sections can be applied.

It is sometimes helpful to consider the forward view the ‘diffusion’ problem (how far do signals travel from a source?), and the backward view as the ‘aggregation’ problem (where did a target accrue signals from?).

3.1. Solving for the steady state information sets

Intuitively, we can see that the set of signals accrued by each node in its information set will be determined by the set of nodes $j$ from which a node $i$ accrued private signals, and the values of the private signals those nodes would supply ($b_i$). In other words, if we can determine where the mortal random walkers from $i$ end up ‘sampling’ in the graph, we can conclude a great deal about the steady state outcomes of this process.

For example, let $x_i^*$ denote the asymptotic mean value of the signals $I_i(\infty)$. Since $\tilde{F}_{ij} = (1 - \alpha)F_{ij}$ denotes the fraction of visits by a mortal random walk from node $i$ to $j$, we can see that, as we saw in equation (6):

$$x_i^* = \sum_j \tilde{F}_{ij}b_j.$$  

(33)

That is, the mean value of the signals in $i$’s information set is the values of the private signals each node $j$ would sample from their private source, weighted by the probability of $i$ sampling from $j$. Vectorising this, we get the familiar:

$$\Rightarrow x^* = \tilde{F}(\alpha)b.$$  

(34)

From here we can determine the expected value (given some graph $\mathcal{G}$) across the nodes:

$$\mathbb{E}[x_i^*] = \sum_j \tilde{F}_{ij}\mathbb{E}[b_j] = \sum_{j=1}^{np} \tilde{F}_{ij} = \mu_p.$$  

(35)

We can also get the expected value of nodes whose private sources are positive or negative:

$$\mathbb{E}[x_i^*|b_i = +1] = \tilde{F}_{ii}(+1) + \mu_p \sum_{j \neq i} \tilde{F}_{ij}$$  

(36)

$$\approx \mu_p(1 - \tilde{\tau}(0)) + \tilde{\tau}(0)$$  

(37)

15
which makes intuitive sense. Interestingly, the degree heterogeneity higher fraction of mortal walkers return to origin, meaning that sparser graphs encourage greater polarisation, (from section 2.1), with a large tree-like graph, we recall that graphs with lower mean degree the divergence between positive and negatively oriented readers.

The end result is quite intuitive: the more agents indirectly sample their network. The extent to which an agent experiences their own information because it is echoed back at them through their network. This is the extent to which an agent experiences their own information because it is echoed back at them through their network. The end result is quite intuitive: the more agents indirectly sample their own private signals.

Note, interestingly, the fraction of signals the node samples from private sources directly is just \((1 - \alpha)\) by design, so that the quantity \(\tilde{\tau}(0) - (1 - \alpha) > 0\) measures the ‘echo chamber effect’ in a very literal manner. It is the extent to which an agent experiences their own information because it is echoed back at them through their network. The end result is quite intuitive: the more agents indirectly sample their own signals, the greater the divergence between positive and negatively oriented readers.

We can now utilise our previous results to quickly draw some conclusions about the size of the ‘polarisation’ on different graph topologies, simply by considering how \(\tilde{\tau}(0)\) varies with the parameters. In our baseline case (from section 2.1), with a large tree-like graph, we recall that graphs with lower mean degree \(\langle k \rangle\) will have a higher fraction of mortal walkers return to origin, meaning that sparser graphs encourage greater polarisation, which makes intuitive sense. Interestingly, the degree heterogeneity \(\langle k^2 \rangle\) will have by itself no impact on the distance between the groups, as it does not affect self-visits of the random walk. We can also see that smaller graphs (introduced in section 2.3) with strong bounce-backs will encourage larger self-weight, for the less obvious reason that the smaller a graph is the more an individual’s own signals circulate back to themselves. This is especially useful if we consider graphs with very sharp community structure, as this effectively means that nodes are contained in sub-graphs that mimic the effects of a smaller network overall.

Compared to the baseline tree-like graph, a lattice-like network with high clustering in the manner discussed in section 2.4 will increase the self-weight \(\tilde{\tau}(0)\) due to signals overwhelmingly circulating in close neighbourhoods, which increases the distance between opposing groups. Finally, for the clustered random graph we introduced in section 2.5, we can see that clustering and degree heterogeneity will reinforce each other to produce a higher \(\tilde{\tau}(0)\) and polarisation. Put more intuitively—the combination of ‘influencers’ (hubs) and ‘cliques’ (clustering) will combine to aggravate polarisation, since insular groups with high clustering will form around highly connected influencers.

We can also examine differences within groups (e.g. the variance of steady state outcomes for the positive readership agents). For the sake of simplicity, we focus on infinitely large graphs for both tree-like and lattice-like structures:

\[
\mathbb{V}[x_i^* | b_i = +1, k_i] = \sum_{j \neq i} F^2_{ij} \mathbb{V}[b_j] = \sigma^2_k \sum_{j \neq i} F^2_{ij} \\
\approx \sigma^2_k \mathbb{E}_k \left[ \frac{1}{k} \right] (\tilde{\tau}(1))^2. \tag{40}
\]

(The full derivation is provided in appendix 1.) Here, \(\sigma^2_k = \mathbb{V}[b_j] = 4p(1 - p)\) measures the variance of the private signal distribution across the population, which is invariant to the topology. The quantity \((\tilde{\tau}(1))^2\) scales with the fraction of visits to immediate neighbours. As we have already discussed, this quantity is considerably higher for highly connected graphs, and particularly so for lattices (or in general with graphs with larger clustering coefficients). Finally, the quantity \(\mathbb{E}_k \left[ \frac{1}{k} \right]\) is the expected degree reciprocal of the graph. For graphs with no degree heterogeneity (for example the lattice where all nodes have degree 2\(m = \langle k \rangle\), or more general KREG graphs), this is simply \(\frac{4}{3\langle k \rangle}\). But as the degree heterogeneity increases, this quantity inflates. For example for BA graphs we can see that:

\[
\mathbb{E}_k \left[ \frac{1}{k} \right] = \sum_{k=m}^{\infty} p(k) \frac{1}{k} \approx \int_{k=m}^{\infty} \frac{2m^2}{3m^3} = \frac{4}{3\langle k \rangle}, \tag{41}
\]

where \(m\) in this case refers to the number of edges added by each new node. In the appendix, we provide an approximation for a generic degree distribution with finite moments that is shown to increase with \(\langle k^2 \rangle\).

It follows therefore that the within-group variance will be strongest for graphs with a large clustering coefficient and degree heterogeneity. In the (bottom panel) of figure 8, we visualise the steady state distribution of \(x_i^*\) for a KREG random graph, a BA graph, and the lattice, where \(\langle k \rangle = 12\) and \(N = 2500\) in each case. The
distribution of positive and negative agents are clearly separated for the tree-like graphs, as their within-group variance is quite low. Inspecting this carefully visually, one can see that the BA network has a slightly larger variance, as predicted. For the lattice, the variance within group is much higher, and as a result the two groups overlap heavily. We can see therefore that the within-group variance is much more heavily dominated by the effect of clustering than the degree heterogeneity.

We verify this by measuring the within-group variance of the positive agents for a large number of simulations for increasing $p$, pictured in the (top left panel). The predictions are very precise for the tree-like graphs, and very slightly under-estimated for the lattice. We can see, as conjectured, that the effect of the lattice’s clustering is substantially more pronounced in determining within-group variance than the degree heterogeneity.

We can finally combine our variation within and between groups to get the variance across all nodes:

$$\mathbb{V}[x_i^*] = \mathbb{E}_k[\mathbb{V}[x_i^* | k_i]] + \mathbb{V}_k[\mathbb{E}_k[x_i^* | k_i]]$$

$$\approx \sigma_p^2 \left( \bar{\tau}(0)^2 + \mathbb{E}_k \left[ \frac{1}{k} h(\bar{\tau}(1))^2 \right] \right)$$

$$\approx \sigma_p^2 \left( \bar{\tau}(0)^2 + \mathbb{E}_k \left[ \frac{1}{k} (\bar{\tau}(1))^2 \right] \right).$$

We verify our results in the (top right panel) of figure 8, where we show the total variance for different levels of the positive readership bias $p$, for the KREG random, BA, and lattice graph. As before, the analytic predictions are quite accurate, and demonstrate that the clustering in the lattice results in a much larger variance than any degree heterogeneity.
Referring back to the local vs global tug-of-war we described in the introduction, we can see a quite clear story here. The variance of the steady state outcomes—a literal measure of the heterogeneity supported by a dynamic process on the graph—is determined primarily by what fraction of the mortal walk visits are local (at the origin and immediate neighbours). If visits are mostly local (as is the case with i.e. large clustering or low \(\langle k \rangle\)), then each node is influenced primarily by a very small radius of neighbours, and local variation is preserved. Conversely, if the visits are ‘pushed out’ to a large radius around each node, then each node is influenced by a very large global population, and as a result local variation is suppressed and the outcomes are much more homogeneous across the network.

4. Conclusions

Over the course of this paper, we have estimated the behaviour of mortal walkers on graphs. We first quantified the behaviour by considering the total expected visits by the mortal walkers between sites on the graph, summarised in the fundamental matrix \(F(\alpha)\). We showed that in the case of large, tree-like graphs, we can express the behaviour of the mortal walkers as a function of the distance between nodes by producing a correspondence with a one-dimensional walk on a chain. Our analytic approximations matched numerical simulations closely, validating our conceptual approach. We then extended this method to consider finite graphs and those which are non-tree-like. In each case, we were able to show how simple parameters like mean connectivity and degree heterogeneity dictated the specific behaviour of the mortal walkers. We leveraged this to develop simple intuition as to how changing the topology of the graph would influence the mortal walkers.

Throughout our analysis, we also demonstrated how the behaviour of the mortal walkers corresponded precisely to a dynamical process of the graph where agents influence each other’s states linearly and with some decay factor. This culminated in a worked example where we considered a model of information aggregation on the graph. With very little overhead, we were able to show how the results from our mortal walker analysis could be quickly applied to build a clear picture of how the topology of the graph affected the variability in information sets agents could accrue over time.

While the dynamical processes we considered in this paper were purposefully simple, we hope they demonstrate how reasoning about interactions in terms of mortal walkers can shed light onto more complex dynamics on graphs. The mortal random walks in this case merely provided a convenient and intuitive characterisation of the number of causal pathways that exist between nodes in the network. To say that the visits by the walk are mostly local, for example, is to say that a large fraction of the causal paths that influence a given node occur in its immediate neighbourhood, assuming that the causal influence between nodes decays with distance. As such, we hope that is provides a simple building block for network scientists to reason about the 'tug-of-war' between local variation and global smoothing that characterises many diverse models.

Acknowledgments

OS would like to thank Pierpaolo Vivo, Giacomo Livan, Irtzea Ishraq and Isabelle Ogland for many useful comments and feedback.

Data availability statement

No new data were created or analysed in this study.

Appendix A. Solving fully connected and disconnected graphs

When the graph is fully disconnected, we represent the adjacency matrix with \(A = I\). Computing \(\tilde{F}(\alpha)\) is then straightforward:

\[
\tilde{F}(\alpha) = (1 - \alpha)(I - \alpha I)^{-1} = (1 - \alpha) \frac{1}{1 - \alpha} I = I.
\]  

(A.1)

When the graph is fully connected, we represent the (row-normalized) adjacency matrix with \(N^{-1} \mathbf{1} \mathbf{1}'\) where \(\mathbf{1}\) is the vector of all 1s. Note that this matrix is idempotent:

\[
(N^{-1} \mathbf{1} \mathbf{1}') (N^{-1} \mathbf{1} \mathbf{1}') = N^{-2} (\mathbf{1} \mathbf{1}') = N^{-1} \mathbf{1} \mathbf{1}'.
\]  

(A.2)
We can then proceed by taking the summation representation of the adjacency matrix:

$$\tilde{F}(\alpha) = (1 - \alpha)\sum_{t=0}^{\infty} (\alpha A)^t = (1 - \alpha) \left( I + \sum_{t=1}^{\infty} \alpha^t \left( \frac{N^{-1}11'}{N^{-1}11'} \right)^t \right) \quad (A.3)$$

$$= (1 - \alpha) \left( I + \sum_{t=1}^{\infty} \alpha^t \right) = (1 - \alpha) \left( I + \frac{\alpha}{1 - \alpha} \right) \quad (A.4)$$

$$= (1 - \alpha) \left( I + \frac{\alpha}{1 - \alpha} \left( N^{-1}11' \right) \right) \quad (A.5)$$

$$= (1 - \alpha) \left( I + \frac{\alpha}{1 - \alpha} \left( N^{-1}11' \right) \right) \quad (A.6)$$

$$\Rightarrow \tilde{F}(\alpha) = (1 - \alpha)I + \alpha N^{-1}11'. \quad (A.7)$$

### Appendix B. Approximating PageRank

In the main text, we alluded to the fact that computing the fundamental matrix and the visitation probabilities provides a shortcut to approximating the PageRank values of the nodes in the graph. This is because, as highlighted in equation (7), the classic PageRank vector is just the column sums of the normalized fundamental matrix:

$$\pi = \tilde{F}(\alpha)^T1. \quad (B.1)$$

Furthermore, as we showed in equation (23), the expected PageRank for degree $k$ can be approximated as the mean column sum of nodes with degree $k$, i.e.:

$$\pi_k \approx \tau(0) + \frac{k}{\langle k \rangle} \left( \frac{1}{1 - \alpha} + \tau(0) \right). \quad (B.2)$$

We now verify this claim by computing the expected PageRank by degree for all remaining heterogeneous graphs we used in this paper (excluding the SBM, since this was already shown in figure 3, and the homogeneous graphs i.e. KREG and lattice since there is no meaningful variation in PageRank to display). The results are presented in figure B1. As we can see, the approximation is quite accurate, even in the case of the clustered graphs where the quantity $\tau(0)$ now relies on more complex features such as the global clustering coefficient and degree heterogeneity.

For the non-clustered graphs where we have computed the full matrix $\tilde{F}(\alpha)$, we can go one step further and approximate the PageRank for each node, $i$ since this is just the column sum of our approximation of $\tilde{F}(\alpha)$. Here again the approximation was quite accurate. For the ER graph, the mean APE was under 1%, for the BA graph it was ~ 8% and for the SBM it was under 5%. While a comprehensive analysis of PageRank against our model is beyond the scope of this paper, these together indicate that the fundamental matrix approximations provide a useful perspective to reconsider the traditionally expensive PageRank computation (at least in the case of undirected graphs).

### Appendix C. Solving the generating functions

Suppose we take the general question of a random walk on the chain with a reflection at the origin, illustrated in the top right of figure 6. The behaviour of this walk can be parameterised by two values: $a$ is the probability of moving forward in the chain and $b$ is the probability of remaining at the current distance. The probability of moving back up the chain is then $c = (1 - a - b)$.

For $d \geq 2$, we can then write the following relationship:

$$p_{d,t} = ap_{d-1,t-1} + bp_{d-1,t} + cp_{d+1,t-1}. \quad (C.1)$$

Multiplying both sides by $z^t$ and summing from $t = 1$ to $t = \infty$ we get:

$$\sum_{t=1}^{\infty} p_{d,t}z^t = a \sum_{t=1}^{\infty} p_{d-1,t-1}z^t + b \sum_{t=1}^{\infty} p_{d-1,t}z^t + c \sum_{t=1}^{\infty} p_{d+1,t-1}z^t. \quad (C.2)$$
Figure B1. Comparison of numerical simulations vs theory for the mean PageRank of nodes of degree \(k\) in each of the heterogeneous graphs used in the main section of the paper. In all cases the parameters match those in the paper: starting on the far left with the ‘unclustered’ ER (figure 2; \(\langle k \rangle = 3, \alpha = 0.8, N = 2000\)), then the ‘unclustered’ BA (figure 2; \(\langle k \rangle = 6, \alpha = 0.5, N = 2000\)), the ‘clustered ER (figure 7; \(\langle k \rangle = 4, \alpha = 0.75, N = 1000\)), and finally the ‘clustered BA (figure 7; \(\langle k \rangle = 4, \alpha = 0.75, N = 1000\)).

Note that since \(p_{d,0} = 0\) for all \(d > 0\), the left-hand side is equivalent to \(\sum_{t=0}^{\infty} p_{d,t}z^t = G_d(z)\). On the right-hand side, substitute \(s\) for \(t-1\) to get:

\[
G_d(z) = a z \sum_{t=0}^{\infty} p_{d-1,t}z^t + b z \sum_{t=0}^{\infty} p_{d,t}z^t + c \sum_{t=0}^{\infty} p_{d+1,t}z^t
\]

\[\Rightarrow G_d(z) = a z G_{d-1}(z) + b z G_d(z) + c z G_{d+1}(z)\]  

\[\Rightarrow c z G_{d+1}(z) + (b z - 1)G_d(z) + a z G_{d-1}(z) = 0.\]

We now have a recurrence relation, so we can write each \(G_d(z)\) as \(G_{d-1}(z)C\) i.e. there is some constant decay rate \(C\) for \(d \geq 2\). We can therefore write \(G_d(z) = G_1(z)C^{d-1}\) for \(d \geq 2\) and substitute this in to get:

\[(czC^2 + (b z - 1)C + az)C^{d-2} = 0.\]

We can now just solve for the quadratic expression to get the two non-zero roots:

\[C(a, b, z) = \frac{-(b z - 1) \pm \sqrt{(b z - 1)^2 - 4a(1 - a - b)z^2}}{2(1 - a - b)z}.\]

We can then use additionally the fact that it must hold that \(C \to 0\) as \(z \to 0\) (since this would imply the random walk terminates immediately) to retain only the smaller solution as viable, i.e. the solution for the
decay rate for the visitations at each layer is:

\[ C(a, b, z) = \frac{-(bz - 1) - \sqrt{(bz - 1)^2 - 4a(1 - a - b)z^2}}{2(1 - a - b)z}. \] (C.8)

For our tree solution, where there were no connections within a single layer, we could substitute \( a = \frac{(k - 1)}{2} \) and \( b = 0 \) to retrieve the ratio \( C \) used in equation (19). For the lattice solution, we can use \( a = \frac{m + 1}{2m} \) and \( b = \frac{m - 1}{2m} \) to retrieve the ratio \( D \) used in appendix F, equation (F.4). For the clustered graph, we obtain transition probabilities in appendix G below.

In order to get the full expression, we make use of the boundary conditions. For generality, we label the forward and retention probabilities for distance 1 as \( a_1 \) and \( b_1 \), since there could be some special case we might want to capture from the topology (such as a lattice structure, or more generally this may help obtain a more precise solution for edge cases).

Since there is always a reflection at the origin (and no self-loops), we can see for \( t > 0 \):

\[ p_{0,t} = (1 - a_1 - b_1)p_{1,t-1}. \] (C.9)

Repeating the process above, we get:

\[ G_0(z) = (1 - a_1 - b_1)zG_1(z) + 1. \] (C.10)

We then note that:

\[ p_{1,t} = p_{0,t-1} + b_1p_{1,t-1} + cp_{2,t-1}. \] (C.11)

Repeating the process again, we get:

\[ zG_0(z) + (b_1z - 1)G_1(z) + czG_2(z) = 0. \] (C.12)

Note now however that \( G_z(z) = G_1(z)C \), so substituting in all variables, we get:

\[ z((1 - a_1 - b_1)zG_1(z) + 1) + (b_1z - 1)G_1(z) + \]

\[ (1 - a - b)zG_1(z)C = 0 \]

\[ \Rightarrow ((1 - a_1 - b_1)z^2 + ((1 - a - b)C + b_1)z - 1)G_1(z) = -z \]

\[ \Rightarrow G_1(z) = \frac{z}{(b_1 + a_1 - 1)z^2 + ((b + a - 1)C - b_1)z + 1}. \] (C.16)

We can finally determine \( G_0(z) \) by substituting this expression into equation (C.10).

In order to obtain the results in the paper, for the tree solution, we substitute \( a = a_1 = \frac{(k - 1)}{2} \) and \( b = b_1 = 0 \) to get the expressions in equations (16) and (17). For the lattice, we substitute \( a = \frac{m + 1}{2m}, \ a_1 = \frac{m + 1}{2m}, \ b = \frac{m - 1}{2m} \) and \( b_1 = \frac{m - 1}{2m} \) (these can be obtained by just enumerating the number of edges between layers) to get the expressions in equations (F.2) and (F.3). For the clustered graph we again derive these appendix G.

**Appendix D. The random walk radius**

Computing the random walk radius in full:

\[
\mathbb{E}[\text{final layer of walk}] = \sum_{t=0}^{\infty} \mathbb{E}[\text{final layer|walk is of length } t] \mathbb{P}[\text{walk is of length } t] \tag{D.1}
\]

\[
= \sum_{t=0}^{\infty} \left[ \sum_{d=0}^{\infty} dP_{d,t} \right] \alpha^t(1 - \alpha) = (1 - \alpha)\sum_{d=0}^{\infty} d \left[ \sum_{t=0}^{\infty} P_{d,t} \alpha^t \right] \tag{D.2}
\]

\[
= (1 - \alpha)\sum_{d=1}^{\infty} d\tau(d) = (1 - \alpha)\sum_{d=1}^{\infty} d\tau(1)C(\alpha)^{d-1} \tag{D.3}
\]

\[
= (1 - \alpha)\tau(1)\frac{\partial}{\partial C} \sum_{d=1}^{\infty} C(\alpha)^d \tag{D.4}
\]

\[
= (1 - \alpha)\tau(1)\frac{\partial}{\partial C} \frac{C}{1 - C} = \frac{\tau(1)}{(1 - C(\alpha))^2} = R_W. \tag{D.6}
\]
We verify this numerically in figure D1 by simulating a mortal random walk on a chain with the transition probabilities $\alpha$ and termination probabilities as dictated in the tree graph with mean degree $\langle k \rangle$.

Appendix E. Expected visits for finite graphs

The closed form solutions for $\tau(d)$, the expected number of visits by a mortal walk to distance $d$ in a finite chain of length $n$ are defined piecewise for the origin ($d = 0$), termination distance ($d = n$) and distance in between ($0 < d < n$):

\[
\tau(0) = \left( (\alpha^2 - 1)\langle k \rangle((s + 1)(1 - s)^n - (1 - s)(1 + s)^n) \right)^{-1} \\
\cdot (\langle k \rangle - 1) \left( (1 - s)^n(-2\alpha^2 + (s + 1)\langle k \rangle) \\
+ (1 + s)^n(2\alpha^2 + (s - 1)\langle k \rangle) \right)
\]

\[
\tau(n) = \frac{2\alpha(2\alpha^{(\langle k \rangle - 1)} \langle k \rangle)^n}{(\alpha^2 - 1)((\langle k \rangle - 1)((1 + s)(1 - sw)^n - (1 - s)(1 + s)^n)}
\]

\[
\tau(d) = \alpha \left( \frac{\langle k \rangle((\langle k \rangle - 1)^2}{2\alpha} \right)^d \\
\cdot \left( ((1 - s)^d((s + 1)\langle k \rangle + 2)(1 + s)^n) \\
+ (1 + s)^d((s + 1)(\langle k \rangle - 2)(1 - s)^n) \right) \\
\cdot \left( (\alpha^2 - 1)(\langle k \rangle - 1)((s + 1)(1 - s)^n + (s - 1)(s + 1)^n) \right)^{-1}.
\]

In the above, $s$ is a shorthand for $s(z, \langle k \rangle)$ as defined in equation (18).

As mentioned in the main text, the solution can be derived by noting that the transition probabilities between layers of a chain can be encoded in a tri-diagonal matrix, and solving for the inverse. Denote a transition matrix $T$:

\[
\begin{bmatrix}
  a_1 & b_1 & 0 & \ldots & 0 \\
  c_1 & a_2 & b_2 & \ldots & 0 \\
  0 & c_2 & a_3 & \ldots & 0 \\
  \vdots & \vdots & \ddots & \ddots & b_{n-1} \\
  0 & 0 & \ldots & c_{n-1} & a_n
\end{bmatrix},
\]

(E.1)
where \( n \) is the expected radius of the graph. From the transition probabilities on the tree-like graph, we can use \( a_i = 0, b_1 = 1, b_{j+1} = 1 - \frac{1}{z} \) and \( c_i = \frac{1}{z} \) (this can be substituted as required with the transition probabilities for the different topologies). We then adjust for the walker mortality (= \( \alpha T \)), and finally to get the ‘fundamental matrix’ for this finite chain, we take:

\[
F_T = (I - \alpha T)^{-1} = S^{-1} \quad (E.2)
\]

\[
S = \begin{bmatrix}
1 & -\alpha & 0 & \ldots & 0 \\
\alpha/\langle k \rangle & 1 - \alpha/\langle k \rangle & -\alpha/\langle k \rangle & \ldots & 0 \\
0 & \alpha/\langle k \rangle & 1 & \ldots & 0 \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
0 & 0 & 0 & \ldots & 1
\end{bmatrix} \quad (E.3)
\]

Since the mortal walker always starts at the origin, we are only interested in obtaining the first row \( S_{1j} \). Solving for these elements simply require plugging in the terms into the known closed form solutions for the tri-diagonal inverse (see, for example, [32, 33]). This will return the expressions listed above.

### Appendix F. Expected visits for lattices

The closed form solutions for \( \tau_L(d) \), the expected number of visits by a mortal walk to distance \( d \) in a one-dimensional lattice where each node is connected to its \( m \) nearest neighbours on the lattice:

\[
\tau_L(d) = \begin{cases} 
H_0(\alpha) & \text{for } d = 0 \\
H_1(\alpha)D(\alpha)^{d-1} & \text{for } d \geq 1,
\end{cases} \quad (F.1)
\]

where:

\[
H_0(z, m) = \frac{zm}{zm} \left( \sqrt{\frac{1 - z(1 + m)}{z^2m}} - 1 \right) + m + z \quad (F.2)
\]

\[
H_1(z, m) = \frac{2zm}{zm} \left( \sqrt{\frac{1 - z(1 + m)}{z^2m}} - 1 \right) + m + z - z^2 \quad (F.3)
\]

\[
D(z) = \frac{z + 2m - zm}{z(m + 1)} \left( \sqrt{\frac{1 - z(1 + m)}{z^2m}} + 1 \right) \quad (F.4)
\]

The derivation follows from the solution of the generating function detailed in appendix C.

### Appendix G. Solving the clustered random graph

In order to increase the clustering coefficient of a target graph \( G \) we first sample a node \( i \) uniformly at random. A disconnected pair \( j, k \) of its neighbours are selected. Two further neighbours are selected from these nodes: \( j_2 \) is another neighbour of \( j \) and \( k_2 \) is another neighbour of \( k \). We then rewire as follows: \( j \) and \( k \) are wired to each other and unwired from \( j_2 \) and \( k_2 \). Then \( j_2 \) and \( k_2 \) are wired to each other. This rewiring ensures that the degree of each node after the operation is unchanged. Before the operation is finalized, a check is done to make sure this operation does not disconnect any existing triangles, in which case it is allowed to proceed. The operation proceeds until the global clustering coefficient \( \gamma_G \) reaches the target level.

In order to deduce the transition probabilities through layers of this graph, we begin by discussing how the ‘local’ clustering coefficient will vary across nodes of various degrees in the graph. While the global clustering coefficient \( \gamma_G \) is defined as the ratio of triangles to connected triples in the graph, the local clustering coefficient \( \gamma_k \) defined as the ratio of triangles connected to vertex \( i \) and the number of potential triples centred at vertex \( i \), which will be \( k(k - 1)/2 \). Clearly, the degree of a vertex will be a factor in its expected local clustering. We aim to determine \( \gamma_k \) which is the expected local clustering for nodes with degree \( k \).

\[
\gamma_k = \frac{\sum_{k_i=k} \gamma_i}{N(k)} = \frac{1}{N(k)} \sum_{k_i=k} \Delta_i = \frac{2\Delta_k}{N(k)k(k - 1)} \quad (G.1)
\]
Here, \(N(k)\) is the number of nodes with degree \(k\), \(\Delta_i\) is the number of triangles with a corner at \(i\), and \(\Delta_k\) sums this across all nodes with degree \(k\). In order to determine this latter quantity in expectation, consider the following argument: there are \(\Delta_G\) triangles across the whole graph, each of which has three corners. Each corner must so to speak, ‘attach’ to a node. The probability of picking a node with degree \(k\) by following a corner in a random graph is precisely the edge-sampled degree distribution \(\frac{P(k)}{\langle k \rangle}\). It follows the total number of expected triangles attached to nodes of degree \(k\) is just:

\[
\Delta_k = \frac{3\Delta_G k P(k)}{\langle k \rangle} = \frac{3\Delta_G k N(k)}{\langle k \rangle N}. \tag{G.2}
\]

Leading to:

\[
\gamma_k = \frac{6\Delta_G k N(k)}{N(k)k(k-1)\langle k \rangle N} = \frac{6\Delta_G}{(k-1)\langle k \rangle N}. \tag{G.3}
\]

The value of \(\Delta_G\) can be acquired by re-arranging the definition of the global clustering coefficient:

\[
\gamma_G = \frac{\sum_i \Delta_i}{\sum_i k_i(k_i-1)/2} \Rightarrow \gamma_G = \frac{6\Delta_G}{N\sum_k P(k)k(k-1)} = \frac{6\Delta_G}{N\langle k^2 \rangle - \langle k \rangle}. \tag{G.4}
\]

Finally:

\[
\gamma_k = \frac{\gamma_G N\langle k^2 \rangle - \langle k \rangle}{(k-1)\langle k \rangle N} = \gamma_G \frac{\langle k^2 \rangle - 1}{k-1} = \tilde{\gamma} = \frac{\gamma_G}{k-1}, \tag{G.5}
\]

where \(\tilde{\gamma} = \gamma_G \frac{\langle k^2 \rangle - 1}{(k-1)\langle k \rangle} \) is a constant term capturing the effect of the global clustering coefficient and expected excess degree.

The final expression in equation (G.6) tells us the expected fraction of a node’s pairs of neighbours should be connected. We can leverage this directly into understanding transition probabilities. We consider what happens at \(d = 0, 1\) and \(d \geq 2\) separately, since the clustering may invoke different transition probabilities around the origin (as was the case with the lattice).

We first consider determining \(b_1\), that is the probability of a random walk at layer 1 staying at layer 1. Consider first a random walk that departs from some origin node \(i\) to a neighbour \(j\) at \(d = 1\). Suppose \(i\) has degree \(k_i\) and \(j\) has degree \(k_j\). The expected number of connections from node \(j\) to other neighbours of \(i\) in the same layer will be the probability of being connected to any one of them (which would close a triangle) and the number of potential neighbours to connect to. For the former, since we are closing a triangle with a corner in a random graph is precisely the edge-sampled degree distribution \(\frac{P(k_i)}{\langle k \rangle}\). Therefore, the expected number of edges at the same layer is just \(\tilde{\gamma}\).

The expected fraction of edges (which determines the transition probability) is just \(\frac{\tilde{\gamma}}{k_i}\). Taking the expectation of this over the distribution \(k_i\) (which we recall is edge-sampled as per the tree example in the main text), we get \(b_1 = \frac{\tilde{\gamma}}{k_i}\).

We can use similar reasoning and the fact that there can only be one edge back to \(d = 0\) (the origin node) from \(j\) to determine \(c_1 = P[1 \rightarrow 0] = \frac{\tilde{\gamma}}{k_j}\), as per the tree in the main text.

Finally, we determine \(c\) (backward probability) and \(b\) (retention probability) for \(d \geq 2\). In order to do this, it is useful to imagine some fixed target node \(j\) at the layer \(d\) we reach, which has \(k_j\) neighbours. All random walks that reach \(j\) must arrive via one of the \(k_j\) neighbours, so the partition of these neighbours into being in layer \(d\), \(d - 1\) or \(d + 1\) depend on what direction the random walk arrives from.

What is the expected number of the \(k_j\) neighbours at \(d - 1\)? Clearly, there must be at least one (in order for the walk to arrive at \(j\)). In a random graph, we should not expect any more than this: the probability of two neighbours of \(j\) both being at distance \(d - 1\) from any source is asymptotically 0 as \(N \to \infty\) (since we are approaching them not via \(j\), so they are independent at layer \(d - 1\)). In the clustered graph we might expect this to grow to multiple nodes if two or more neighbours of \(j\) share a common ‘parent’ from some direction due to transitive closure if they are already connected—this would form a ‘diamond’ shape in the graph between a node at layer \(d - 2\), two or more nodes at \(d - 1\) and \(j\) at \(d\). However, it can be shown that the probability of each of these diamonds grows with \(O(\gamma_G^2)\), so we can effectively ignore this so long as \(\gamma_G\) is not too large, leaving us with only one neighbour at \(d - 1\) in expectation. Following previous reasoning, this leaves us with an expected probability \(c = c_1 = \frac{\tilde{\gamma}}{k_j}\).

For \(b = P[d \rightarrow d]\) we need to compute the number of the neighbours of \(j\) at the same level. However, note that we already have one expected neighbour at level \(d - 1\), call it node \(m\). Any of the other neighbours of \(j\) this node \(m\) is connected to must be at layer \(d\) also. But there are simply \(k_j - 1\) other potential neighbours,
and each of the triangles close with probability $\tilde{\gamma}_{kj} - 1$. So following the same reasoning as before we are left with

$$b = b_1 = \frac{\tilde{\gamma}}{k}. \tag{G.7}$$

In summary:

$$c = c_1 = \frac{1}{\langle k \rangle} \quad \tag{G.7}$$

$$b = b_1 = \gamma_G \left( \frac{\langle k^2 \rangle}{\langle k \rangle} - 1 \right) \quad \tag{G.8}$$

$$a = 1 - b - c = 1 - \gamma_G \left( \frac{\langle k^2 \rangle}{\langle k \rangle} - 1 \right) + 1 \quad \frac{\langle k \rangle}{(k)} \tag{G.9}$$

We verify these in figure G1 by simulating a random walk over three types of graphs (KREG, ER and BA) with increasing transitivity. For each, we measured the fraction of times the random walk starting at a random node moved further from the origin ($a$), stayed at the same distance from the origin ($b$), or moved closer to the origin ($c$), over the first 4 layers. We can see from the results the estimates are indeed quite accurate.

In order to get the final distribution of visits (and influence) as per figure 7, we simply plug these transition probabilities into the generalised generating function solution provided in appendix C.

**Appendix H. Generalising the private signal sampling distribution**

When we considered our model of information aggregation, we assumed for simplicity that each agent $j$ draws a private signal $b_j \in \{+1, -1\}$ with a positive signal with uniform probability $p$. We can generalise this by considering a hierarchical signal generation model where each agent draws signals $s_j \in S$ with an agent-specific distribution $s_j \sim p_j(s_j) \in \mathbb{P}[S]$, the space of possible signal sampling distributions over $S$. Furthermore, $p_j \sim \rho$ for all $j$, where $\rho$ is some measure over $\mathbb{P}[S]$.

We define the empirical mean of an agent $i$’s information set as $x_i(t) = \sum_{s \in S_i(t)} \frac{4}{|s_i(t)|}$. We can see from the law of large numbers that:

$$x_i(t) \to x_i^* = \mathbb{E}[s \text{ arriving at } i] \tag{H.1}$$

$$= \sum_j P[s \text{ arrives from } j]E_{p_j}[s] = \sum_j \tilde{F}_{ij}E_{p_j}[s]. \tag{H.2}$$
We can therefore compute the expected value of the limiting signal mean for a given graph $G$ and agent $i$ as:

$$
E[x_i^*] = \sum_j F_{ij} \int d\rho(p_j) E_p[s_i]
$$

(H.3)

$$
= \sum_j F_{ij} \int d\rho(p_j) E_p[s_i] = \mu_j \sum_j F_{ij} = \mu, \tag{H.4}
$$

where we make use of the fact that $\int d\rho(p_j) E_p[s_i] = \int d\rho(p) E_p[s_i]$ (the private signal distribution for each agent is drawn uniformly from the hierarchical measure $\rho$).

Similarly, for the variance of the limiting signals $x_i^*$ we find that:

$$
\mathbb{V}[x_i^*] = \sum_j \mathbb{V}_{ij} \mathbb{V}[s_i]
$$

(H.5)

$$
= \left( \int d\rho(p) \mathbb{V}_{ij}(s_i) + \mathbb{V}_{ij}(E_p[s_i]) \right) \sum_j \mathbb{F}_{ij}^2 \approx \sigma_p^2 \sum_{d=0}^{\infty} \frac{\bar{\tau}(d)^2}{|\mathcal{N}(i,d)|^2}, \tag{H.6}
$$

where in the last step we follow the same steps as in the main text.

**Appendix I. Estimating the steady state variance for information aggregating agents**

$$
\mathbb{V}[x_i^*|b_i = 1, k_i] = \sum_{j \neq i} \mathbb{F}_{ij}^2 \mathbb{V}[b_j] = \sigma_p^2 \sum_{j \neq i} \mathbb{F}_{ij}^2
$$

(I.1)

$$
= \sigma_p^2 \sum_{d=1}^{\infty} \sum_{j \in \mathcal{N}(i,d)} \mathbb{F}_{ij}^2 \approx \sigma_p^2 \sum_{d=1}^{\infty} \sum_{j \in \mathcal{N}(i,d)} \frac{\bar{\tau}(d)^2}{|\mathcal{N}(i,d)|^2}, \tag{I.2}
$$

$$
= \sigma_p^2 \sum_{d=1}^{\infty} \frac{\bar{\tau}(d)^2}{|\mathcal{N}(i,d)|^2} \approx \sigma_p^2 \frac{\bar{\tau}(1)^2}{k_i} \sum_{d=0}^{\infty} \left( \frac{C(\alpha)^2}{h} \right)^d, \tag{I.3}
$$

$$
= \sigma_p^2 \left( \frac{\bar{\tau}(1)^2}{k_i} \right) \frac{1}{1 - C(\alpha)^2/h} \approx \sigma_p^2 \left( \frac{\bar{\tau}(1)^2}{k_i} \right), \tag{I.4}
$$

$$
\Rightarrow \mathbb{V}[x_i^*|b_i = +1] = \mathbb{E}[\mathbb{V}[x_i^*|b_i = +1, k_i]], \tag{I.5}
$$

$$
\approx \sigma_p^2 \mathbb{E}_h \left[ \frac{1}{k} \right] \left( \bar{\tau}(1)^2 \right), \tag{I.6}
$$

**Appendix J. Estimating the expected degree reciprocal for an arbitrary degree distribution**

For arbitrary degree distributions with finite moments, one can make use of the following Taylor series approximation around $\langle k \rangle$:

$$
\frac{1}{k} = f(k) \approx \frac{1}{\langle k \rangle} - \frac{1}{\langle k \rangle^2} (k - \langle k \rangle) + \frac{2}{\langle k \rangle^3} (k - \langle k \rangle)^2 \tag{J.1}
$$

$$
\Rightarrow \mathbb{E} \left[ \frac{1}{k} \right] \approx \frac{1}{\langle k \rangle} + \frac{2}{\langle k \rangle^3} \left( \langle k^2 \rangle - \langle k \rangle^2 \right) = \frac{2\langle k^2 \rangle - \langle k \rangle^2}{\langle k \rangle^3}, \tag{J.2}
$$

which, as we asserted, grows with $\langle k^2 \rangle$. In other words, all else being held equal, more networks with higher degree heterogeneity will have more heterogeneous steady state outcomes in the averaging dynamics model.
ORCID iDs

Orowa Sikder © https://orcid.org/0000-0001-6982-0144

References

[1] Castellano C, Fortunato S and Loreto V 2009 Statistical physics of social dynamics Rev. Mod. Phys. 81 591
[2] Golub B and Sadler E 2016 Learning in social networks The Oxford Handbook of the Economics of Networks (Oxford: Oxford University Press)
[3] Sikder O, Smith R E, Vivo P and Livan G 2020 A minimalistic model of bias, polarization and misinformation in social networks Sci. Rep. 10 5493
[4] Stam C J and Reijneveld J C 2007 Graph theoretical analysis of complex networks in the brain Nonlinear Biomed. Phys. 1 3
[5] Barabási A L 2016 Network Science (Cambridge: Cambridge University Press)
[6] Newman M 2018 Networks (Oxford: Oxford University Press)
[7] Da Fonseca C M 2007 On the eigenvalues of some tridiagonal matrices J. Comput. Appl. Math. 200 283–6