Tractable Bayesian Social Learning on Trees*

Yashodhan Kanoria† and Omer Tamuz‡

July 12, 2011

Abstract

We consider a set of Bayesian agents who are attempting to iteratively estimate an unknown 'state of the world' s from initial private signals, and the past actions of their neighbors in a social network. We investigate the computational problem the agents face in implementing the (myopic) Bayesian decision rule. When private signals are independent conditioned on s, and when the social network graph is a tree, we provide a new 'dynamic cavity algorithm' for the agents' calculations, with computational effort that is exponentially lower than what is currently known.

We conjecture that on d-regular trees, myopic Bayesian agents learn s as quickly as agents who practice 'majority dynamics', i.e., at each round adopt the majority opinion of their neighbors. Under mild conditions, we show that with majority dynamics, agents learn s with probability $1-\epsilon$ in $O(\log\log(1/\epsilon))$ rounds. Using our algorithm, the conjecture implies that the computational effort required of Bayesian agents to learn s is only polylogarithmic in $1/\epsilon$ on d-regular trees.

We use our algorithm to perform the first numerical simulations of interacting Bayesian agents on networks with hundreds of nodes. Our numerical results support our conjecture, and indicate rapid learning of s on regular trees.

This work makes the following technical contributions: (1) We introduce a new tool, the 'dynamic cavity method', to the algorithms community. This technique was developed recently to analyze random dynamical processes on trees. We use it to give an efficient algorithm for the agents' computations, demonstrating the algorithmic utility of this method. Our algorithmic approach should generalize beyond the model considered here. (2) We prove doubly exponentially fast convergence of majority updates on regular trees. This result could be of independent interest.

Keywords: social learning, Bayesian agents, computational efficiency, convergence, algorithm, dynamic cavity method.

*Eligible for the best student paper award.

†Microsoft Research New England and Department of Electrical Engineering, Stanford University. Supported by 3Com Corporation Stanford Graduate Fellowship.

‡Microsoft Research New England and Weizmann Institute. Supported by ISF grant 1300/08

1 Introduction

The importance of social learning in networks has been demonstrated in a wide variety of settings, for example, adoption of agricultural technology in Ghana [8] and Mozambique [5], and choice of contraceptives by European women [17]).

Accordingly, developing and understanding models of social learning has been a goal of theoretical economics for the past few decades (cf., Goyal [13]). Typical models in this context assume a *pure information externality*; agent payoffs depend only on the action they choose and an underlying 'state of the world', and not on the actions of others. Agents observe the actions of their 'neighbors', but payoffs are not observed (or observed with noise) ex interim. Typically, all agents have the same utility function. Each agent receives a private signal that contains noisy information about the state of the world. Agents choose actions to maximize expected payoff, given their own private signal and their observations of the actions chosen by others. The premise in such models is that "actions speak louder than words" – agents learn by observing each others' actions, and not by communicating directly.

A standard modeling approach in economics assumes that people (agents) are Bayesian or 'rational' (cf., [3, 24]); agents assume a prior distribution over possible events and pick actions that maximize the expectation of some utility function, conditioned on the information available to them. Accordingly, many studies of social learning assume Bayesian agents [6, 7, 25, 12]. Alternatively, agents in a social network are modeled as 'boundedly rational', attempting to maximize payoffs using some heuristic or 'naïve' rule (e.g., [11, 4]).

Fully Bayesian models have two advantages over models that assume bounded rationality and prescribe thumb rules for agent behavior: First, any bounded rationality approach is bound to involve a somewhat arbitrary decision of which heuristics the agents use. Second, a game theoretic analysis of strategic players is possible only if the players choose actions that are optimal by some criterion. Hence game-theoretic analyses of learning on networks (e.g., [23]) often opt for the more difficult but fully Bayesian model.

Much progress has been achieved in models where Bayesian agents act *sequentially*, such as the herd behavior models of Banerjee [6], Bikhchandani, Hirshleifer and Welch [7], Smith and Sørenson [25], and Acemoglu et al [2]. Here, the interaction is *not bidirectional*: each agent acts only once, taking into account the actions of her predecessors.

In comparison, our understanding of Bayesian agents who act *repeatedly* is much more limited. Gale and Kariv [12] consider Bayesian agents on a network who choose actions repeatedly. They show, in the spirit of Aumann's agreement theorem [3], that agents on a network converge to the same action under some conditions. Related work by Rosenberg, Solan and Vieille [23] and Ménager [18] sheds more light on the phenomenon of agreement on actions and the conditions in which it arises. However, several questions remain essentially unanswered, including: (I) What action do the agents converge to, e.g., what is the distribution of this consensus action? (II) What are the dynamics of such interactions, e.g., what is the rate of agreement/convergence?

The difficulty in answering these questions, and in analyzing bidirectional models in general, seems to be closely related to a lack of understanding of the computational/algorithmic aspects of such models[1]. Moreover, computational tractability is a natural requirement for any decision rule that aims to model behavior of real agents. This motivates the focal question of the current work:

(III) *Are the computations required of Bayesian agents feasible?*

We consider a model that features repeated bidirectional interaction between fully Bayesian agents connected by a social network. Our model is a specialization of the model of Gale and

[1]Mathematical and computational tractability often coincide, e.g., see [14].

Kariv [12]. We consider a group of Bayesian agents, each with a private signal that carries information on an unknown state of the world s. The individuals form a social network, so that each observes the actions of some subset of others, whom we call her neighbors. The agents must repeatedly choose between a set of possible actions, the relative merit of which depends on the state of the world s. The agents iteratively learn by observing their neighbors' actions, and picking an action that is myopically optimal, given their information. Thus, the interaction between agents is not strategic, and is characterized by information externalities.

Even in the simple case of two states of the world, binary private signals and two possible actions, the required calculations appear to be very complicated. A naïve dynamic programming algorithm[2] is exponential in the number of individuals. Since at iteration t one may consider only agents at distance t, then in graphs of maximum degree d (on which we focus) the number of individuals to consider is $O(\min(n, d^t))$, and the computational effort required of each individual to compute their action at time t is $t2^{O(\min(n,d^t))}$. Obviously, this grows very rapidly. As Gale and Kariv remark [12], "The computational difficulty of solving the model is massive even in the case of three persons." This prevents them from even simulating networks with more than three nodes.

We describe a novel algorithm for the agents' calculation in our model, when the social network graph is a tree. This algorithm has running time that is exponentially smaller than the naïve dynamic program, reducing the computational effort to $2^{O(\min(n,td))}$.

Using our algorithm we are able to run numerical simulations of the social learning process. This extends the work of Gale and Kariv [12], who simulated the process for three agents, to much larger networks[3]. We use our algorithm to investigate questions (I) and (II): We numerically evaluate the probability that the agents learn the optimal action, and its progress with time. We observe rapid learning of the optimal action in certain previously unexplored settings. Numerical results are presented in the full version of the paper.

We conjecture that the error probability under Bayesian updates is no larger than the error probability under a different 'majority' update rule, in which agents adopt the opinion of the majority of their neighbors in the previous round. Our numerical results support this conjecture. We prove that for the majority update rule, the number of iterations needed to estimate s correctly with probability $1 - \epsilon$ is $O(\log\log(1/\epsilon))$, for regular trees of degree at least five. Our conjecture then implies that the number of iterations needed to estimate s correctly with probability $1 - \epsilon$ is $O(\log\log(1/\epsilon))$. Thus, assuming the conjecture, the computational effort required of Bayesian agents drops from quasi-polynomial in $1/\epsilon$ (using the naïve dynamic program) to polynomial in $\log(1/\epsilon)$ (i.e., polylogarithmic in $1/\epsilon$), making Bayesian learning computationally tractable. Thus, our results shed new light on question (III), suggesting a positive answer in the case of tree graphs.

The restriction of the discussion to tree or tree-like social networks certainly excludes many natural settings that tend to exhibit highly clustered social graphs. However, in some cases artificially constructed networks have no or few loops by design; these include some highly hierarchical or compartmentalized organizations, as well as some physical communication networks where redundancy is expensive, and the least expensive connected network is a tree. Furthermore, the fact that this non-trivial class of networks does not present a major computational hurdle for fully Bayesian calculations is in itself somewhat surprising.

The work most closely related in spirit to the present one is that of Aaronson on the complexity of agreement [1]. There, as in here, a positive computational result was found somewhat unexpectedly. We present a detailed comparison with Aaronson's work at the end of the full version of this paper.

Technical contributions. A key technique used in this paper is the dynamic cavity method,

[2]Although this algorithm seems to be well known, we could not find a complete description of it in the literature and hence supply it for completeness in Proposition 3.1.

[3]In our numerical analyses, agents receive information (directly or indirectly) from hundreds of distinct nodes.

introduced by Kanoria and Montanari [15] in their study of 'majority updates' on trees. This technique is a dynamical version of the cavity method of statistical physics and appears promising for the analysis of iterative tree processes in general. The key idea is the following: In a dynamical setting on a tree graph, there is correlation in the trajectories of neighbors of a node due to a nodes own past actions. The dynamic cavity method allows to exactly account for these correlations. In this work, we use this method for the first time to give a new algorithmic result, enabling efficient computation by nodes. This is in contrast to the case of majority updates, where the update rule is computationally trivial. Our algorithmic and analytical approach leveraging the dynamic cavity method should be applicable to a range of iterative update situations on trees.

Our second main technical contribution is our proof of doubly exponentially fast convergence of majority dynamics on regular trees. Though this result is somewhat auxiliary in the context of the current work, it should be of independent interest, since, e.g., majority dynamics may serve as a very simple distributed computation protocol or consensus protocol in a distributed system[22]. Besides, majority dynamics is a reasonable model of social learning with bounded rationality.

2 Model

The model we consider is a simplified version of the model of social learning introduced by Gale and Kariv [12]. We first state our model. In Section 2.1, we motivate our model in the context of rational agents.

Consider a directed graph $G = (V, E)$, representing a network of $n \equiv |V|$ agents, with V being the set of agents and E being the social ties between them. A directed edge (i, j) indicates that agent i observes agent j. In most of this paper, we study the special case of undirected graphs, where relationships between agents are bidirectional.

Agents attempt to learn the true *state of the world* $s \in \mathcal{S}$, where $\mathcal{S}$ is finite. The information available to them are their *private signals* x_i, where $x_i \in \mathcal{X}$ and $\mathcal{X}$ is finite. We assume a general distribution of $(s, x_1, \ldots, x_n)$, under the condition that *private signals are independent conditioned on* s, i.e. $\mathbb{P}[s, x_1, \ldots, x_n] = \mathbb{P}[s] \prod_{i \in V} \mathbb{P}[x_i|s]$.

In each discrete time period (or round) $t = 0, 1, \ldots$, agent i chooses action $\sigma_i(t) \in \mathcal{S}$, which we call a 'vote'[4]. Agents observe the votes cast by their neighbors in G. Thus, at the time of voting in round $t \geq 1$, the information available to an agent consists of the private signal she received initially, along with the votes cast by her neighbors in rounds up to $t - 1$. In each round, each agent votes for the state of the world that she currently believes is most likely, given the Bayesian posterior distribution she computes.

Formally, let $\sigma_i^t \equiv (\sigma_i(0), \sigma_i(1), \ldots, \sigma_i(t))$ denote all of agent i's votes, up to and including time t. Let ∂i denote neighbors of agent i, not including i, i.e., $\partial i \equiv \{j : (i, j) \in E\}$. We write $\sigma_{\partial i}^t = \{\sigma_j^t\}_{j \in \partial i}$, i.e., $\sigma_{\partial i}^t$ are the votes of i's neighbors up to and including time t. Then the agents' votes $\sigma_i(t)$ are given by

$$\sigma_i(t) = \arg\max_{s \in \mathcal{S}} \mathbb{P}\left[s \middle| x_i, \sigma_{\partial i}^{t-1}\right],$$

where, if the maximum is attained by more than one value, some deterministic tie breaking rule is used. We denote $\sigma_i = (\sigma_i(0), \sigma_i(1), \ldots)$.

[4]We choose to use "vote" rather than the more usual "act" since, in this simplified model, the set of actions and the set of states of the world are identical, so choosing an action is equivalent to picking a possible state of the world as a guess for the true s.

Note that $\sigma_i(t)$ is a deterministic function of x_i and $\sigma_{\partial i}^{t-1}$. We denote this function $g_i(t) : \mathcal{X} \times |\mathcal{S}|^{t|\partial i|} \to \mathcal{S}$:

$$\sigma_i(t) = g_{i,t}(x_i, \sigma_{\partial i}^{t-1})$$

For convenience, we also define the vector function g_i^t that returns the entire history of i's votes up to time t, $g_i^t \equiv (g_{i,0}, g_{i,1}, \ldots, g_{i,t})$, so that

$$\sigma_i^t = g_i^t(x_i, \sigma_{\partial i}^{t-1}) \,.$$

2.1 Discussion of our Model

The decision rules can be interpreted/motivated as follows. Suppose $\mathbb{P}[s]$, $\mathbb{P}[\underline{x}|s]$ and G are common knowledge[5]. Suppose that for each state of the world s, action σ has utility one when the state of the world is $s = \sigma$, and zero otherwise. Thus, the action that myopically maximizes the expected utility corresponds to the maximum *a posteriori* probability (MAP) estimator of the state of the world. This leads to the decision rule we consider[6], with $\sigma_i(t)$ being chosen as $\arg\max_{s\in\mathcal{S}} \mathbb{P}\left[s \middle| x_i, \sigma_{\partial i}^{t-1}\right]$.

A natural objection to such a model of behavior is that the agents should want to maximize the discounted sum of their future utilities, instead of making the myopic optimal choice. Gale and Kariv [12] deal with this by assuming a continuum of agents at each node, so that no one of them can hope to influence the future by their choice of votes. We can do the same here: Then $\{\sigma_i(t)\}$ form a weak perfect Bayesian equilibrium (cf. [12, Definition 1]) for the right utility function (see above). Alternatively, the myopic model we consider can be thought of as a stepping stone towards the strategic model, being a special case corresponding to discount factor 0.

The scaling regime we consider is the following: We treat the cardinalities of the sets $\mathcal{S}$ and $\mathcal{X}$ and the maximum degree d of the network as fixed parameters, whereas the scaling parameters are the number of agents $n \equiv |V|$, and the number of iterations t. An alternative scaling parameter to t is $1/\epsilon$, where $\epsilon > 0$ is the desired probability of error.

The full version contains a more detailed discussion of the model, including extensions to a general action set and general utility function, and a comparison with the Gale-Kariv model [12].

3 Main results

3.1 Efficient computation

To the best of our knowledge, the literature (e.g., [12, 23]) does not contain an explicit description of an algorithm to compute the actions chosen by agents in our model. However, it seems that a dynamic programming algorithm that performs this computation is well known. The proposition below states the computational complexity of this algorithm.

Proposition 3.1. *On any graph G, there is a dynamic programming (DP) based algorithm that allows agents to compute their actions up to time t with computational effort $t2^{O(\min(n,(d-1)^t))}$, where d is the maximum degree of the graph.*

The algorithm and proof is included in the full version of this paper. This proposition provides the benchmark that we compare our other algorithmic results to. In particular, we do not consider this algorithm a major contribution of this work.

[5] We say that G is common knowledge, if, for each pair of agents i and j, both know G, both know that they both know G, both know that they both know that they both know G, and so on.

[6] Here, the decision rules are also taken as common knowledge, as per the norm in definitions of equilibria (like Nash equilibrium for games) in economics.

A key advantage of the DP algorithm is that it works for any graph G. The disadvantage is that the computational effort required grows doubly exponentially in the number of iterations t.

Our main result concerns the computational effort needed when the graph G is a tree[7]. We show that computational effort exponentially lower than that of the naive DP suffices in this case.

Theorem 3.2. *In a tree graph G with maximum degree d, each agent can calculate her actions up to time t with computational effort $t2^{O(\min(n,td))}$.*

The algorithm we use employs a technique called the dynamic cavity method [15], previously used only in analytical contexts. Section 4 contains a description of the algorithm and analysis leading to Theorem 3.2.

An apparent issue is that the computational effort required is exponential in t; typically, exponentially growing effort is considered as large. However, in this case, we expect the number of iterations t to be typically quite small, for two reasons: (1) In many settings, agents appear to converge to the 'right' answer in a very small number of iterations [12]. In Section 3.2 below, we conjecture that if ϵ is the desired probability of error, then the number of rounds required should be only $O(\log\log(1/\epsilon))$, leading to computational effort of only polylog$(1/\epsilon)$. Having obtained an approximately correct estimate, the agents would have little incentive to continue observing their neighbors actions and updating their beliefs.[8] (2) In many situations we would like to model, we might expect only a small number (e.g., single digit) number of iterative updates to occur, irrespective of network size etc. For instance, voters may discuss an upcoming election with each other over a short period of time, ending on the election day when ballots are cast.

3.2 Convergence

Since an agent gains information at each round, and since she is Bayesian, then the probability that she votes correctly is non-decreasing in t, the number of rounds. We say that the agent *converges* if this probability converges to one, or equivalently if the probability that the agent votes incorrectly converges to zero[9].

We say that there is *doubly exponential convergence* to the state of the world s if the maximum single node error probability $\max_{i\in V}\mathbb{P}\left[\sigma_i(t)\neq s\right]$ decays with round number t as

$$\max_{i\in V}\mathbb{P}\left[\sigma_i(t)\neq s\right]=\exp\left(-\Omega(b^t)\right), \tag{1}$$

where $b>1$ is some constant.

The following is an immediate corollary of Theorem 3.2.

Corollary 3.3. *Consider iterative Bayesian learning on a tree of with maximum degree d. If we have doubly exponential convergence to s, then computational effort that is polynomial in* $\log(1/\epsilon)$ *(i.e., polylogarithmic in $1/\epsilon$) suffices to achieve error probability $\mathbb{P}\left[\sigma_i(t)\neq s\right]\leq\epsilon$ for all i in V.*

Note that if weaken our assumption to doubly exponential convergence in only a subset $V_{\mathrm{c}}\subseteq V$ of nodes, i.e., $\max_{i\in V_{\mathrm{c}}}\mathbb{P}\left[\sigma_i(t)\neq s\right]=\exp\left(-\Omega(b^t)\right)$, we still obtain a similar result with nodes in V_{c} efficiently learning s.

We state below, and provide numerical evidence for (in the full version), a conjecture that implies doubly exponential convergence of iterative Bayesian learning.

[7] By *tree*, we mean a graph that contains no loops. This is sometimes called a 'forest' in the literature.

[8] Thus, $1/\epsilon$ serves as an alternative scaling parameter to t.

[9] Note that this notion of 'convergence' differs greatly from the 'agreement on actions' sense in which the term is sometimes used.

3.2.1 Bayesian vs. 'majority' updates

We conjecture that on regular trees, iterative Bayesian learning leads to lower error probabilities (in the weak sense) than a very simple alternative update rule we call 'majority dynamics'[15]. Under this rule, the agents adopt the action taken by the majority of their neighbors in the previous iteration (this is made precise in Definition 5.1). Our conjecture seems natural since the iterative Bayesian update rule chooses the vote in each round that (myopically) minimizes the error probability. We use $\widehat{\sigma}_i(t)$ to denote votes under the majority dynamics.

Conjecture 3.4. *Consider binary $s \sim$ Bernoulli(1/2), and binary private signals that are independent identically distributed given s, with $\mathbb{P}[x_i \neq s] = 1 - \delta$ for some $\delta \in (0, 1/2)$. Let the majority dynamics (see Definition 5.1) be initialized with the private signals, i.e., $\widehat{\sigma}_i(0) = x_i$ for all $i \in V$. Then on any infinite regular tree, for all $t \geq 0$, we have*

$$\mathbb{P}[\sigma_i(t) \neq s] \leq \mathbb{P}[\widehat{\sigma}_i(t) \neq s] \ . \tag{2}$$

In Section 5, we show doubly exponential convergence for majority dynamics on regular trees:

Theorem 3.5. *Consider binary $s \sim$ Bernoulli(1/2), and binary initial votes $\widehat{\sigma}_i(0)$ that are independent identically distributed given s, with $\mathbb{P}[\widehat{\sigma}_i(0) \neq s] = 1 - \delta$ for some $\delta \in (0, 1/2)$. Let i be any node in an (undirected) d regular tree for $d \geq 5$. Then, under the majority dynamics,*

$$\mathbb{P}[\widehat{\sigma}_i(t) \neq s] = \exp\Big[-\Omega\Big(\big(\tfrac{1}{2}(d-2)\big)^t \Big) \Big].$$

when $\delta < (2e(d-1)/(d-2))^{-\frac{d-2}{d-4}}$.

Thus, if Conjecture 3.4 holds:

- We have doubly exponential convergence for iterative Bayesian learning on regular trees with $d \geq 5$, implying that for any $\epsilon > 0$, an error probability ϵ can be achieved in $O(\log\log(1/\epsilon))$ iterations with iterative Bayesian learning.
- Combining with Corollary 3.3), we see that the computational effort that is polylogarithmic in $(1/\epsilon)$ suffices to achieve error probability $1/\epsilon$.

This compares favorably with the quasi-poly$(1/\epsilon)$ (i.e., $\exp\big(\text{polylog}(1/\epsilon)\big)$) upper bound on computational effort that we can derive by combining Conjecture 3.4 and the naïve dynamic program described. Indeed, based on recent results on subexponential decay of error probability with the number of private signals being aggregated [16], it would be natural to conjecture that the number of iterations T needed to obtain an error probability of ϵ obeys $(d-1)^T \geq C\log(1/\epsilon)$ for any $C < \infty$, for ϵ small enough. This would then imply that the required computational effort using the naïve DP on a regular tree of degree d grows faster than any polynomial in $1/\epsilon$.

Since we are unable to prove our conjecture, we instead provide numerical evidence for it (see the full version of the paper), which is consistent with our conjecture over different values of d and $\mathbb{P}[x_i \neq s]$. The full version also discusses difficulties in proving the conjecture.

We would like to emphasize that several of the error probability values could be feasibly computed only because of our new efficient approach to computing the decision functions employed by the nodes. For instance, with $d = 5$, computing the decision function at iteration 3 using the dynamic program (see Proposition 3.1) would require enumeration over $2^{80} \sim 10^{24}$ possibilities, which is infeasible even on state-of-the-art supercomputers. With our approach, we are able to compute the decision function at iteration 3 and even at iteration 4, on a desktop machine. This aggregates information from the ~ 400 nodes within 4 hops of a given node. Our numerical results indicate very rapid decay of error probability on regular trees (cf. questions (I) and (II) in Section 1).

4 The Dynamic Cavity Algorithm on Trees

In this section we develop the dynamic cavity algorithm leading to Theorem 3.2. We present the core construction and key technical lemmas in Section 4.1. In Section 4.2, we show how this leads to an efficient algorithm for the Bayesian computations on tree graphs, and prove Theorem 3.2.

Assume in this section that the graph G is a tree with finite degree nodes. For $j \in \partial i$, let $G_{j\to i} = (V_{j\to i}, E_{j\to i})$ denote the connected component containing node j in the graph G with the edge (i,j) removed. That is, $G_{j\to i}$ is j's subtree when G is rooted at i.

4.1 The Dynamic Cavity Method

We consider a modified process where agent i is replaced by a *zombie agent* who takes a fixed sequence of actions $\tau_i = (\tau_i(0), \tau_i(1), \ldots)$, and *the true state of the world is assumed to be some fixed s*. Furthermore, this 'fixing' goes unnoticed by the agents (except i, who is a zombie anyway) who perform their calculations assuming that i is her regular Bayesian self. Formally:

$$\sigma_j(t) = \begin{cases} \tau_i(t) & \text{for } j = i\,, \\ g_{j,t}(x_j, \sigma_{\partial j}^{t-1}) & \text{for } j \neq i\,. \end{cases}$$

We denote by $\mathbb{Q}\left[A||\tau_i, s\right]$ the probability of event A in this modified process.

Remark 4.1. *We emphasize that the modified process with a 'zombie' agent is a* theoretical construct *we use to derive an efficient implementation for the iterative Bayesian decision rules. Our algorithm does not involve actual replacement of nodes in the network.*

This modified process is easier to analyze, as the processes on each of the subtrees $V_{j\to i}$ for $j \in \partial i$ are independent: Recall that private signals are independent conditioned on s, and the zombie agent ensures that the subtrees stay independent of each other. This is formalized in the following claim, which is immediate to see:

Claim 4.2. *For any $i \in V$, $s \in \mathcal{S}$ and any trajectory τ_i, we have*

$$\mathbb{Q}\left[\sigma_{\partial i}^t \middle|\middle| \tau_i, s\right] = \prod_{j\in\partial i} \mathbb{Q}\left[\sigma_j^t \middle|\middle| \tau_i^t, s\right]. \tag{3}$$

(Since σ_j^t is unaffected by $\tau_i(t')$ for all $t' > t$, we only need to specify τ_i^t, and not the entire τ_i.)

Now, it might so happen that for some number of steps the 'zombie' agent behaves exactly as may be expected of a rational player. More precisely, given $\sigma_{\partial i}^{t-1}$, it may be the case that $\tau_i^t = g_i^t\left(x_i, \sigma_{\partial i}^{t-1}\right)$ for some x_i. This event provides the connection between the modified process and the original process, and is the inspiration for the following theorem.

Theorem 4.3. *Consider any $i \in V$, $s \in \mathcal{S}$, $t \in \mathbb{N}$, trajectory τ_i and $\sigma_{\partial i}^{t-1}$. For any x_i such that $\mathbb{P}\left[x_i|s\right] > 0$, we have*

$$\mathbb{P}\left[\sigma_{\partial i}^{t-1}\middle|s, x_i\right] \mathbf{1}\left(\tau_i^t = g_i^t\left(x_i, \sigma_{\partial i}^{t-1}\right)\right) = \mathbb{Q}\left[\sigma_{\partial i}^{t-1}\middle|\middle|\tau_i, s\right] \mathbf{1}\left(\tau_i^t = g_i^t\left(x_i, \sigma_{\partial i}^{t-1}\right)\right). \tag{4}$$

Proof. We couple the original process, after choosing s, to the modified processes by setting the private signals to be identical in both.

Now, clearly if it so happens that $\tau_i^t = g_i^t\left(x_i, \sigma_{\partial i}^{t-1}\right)$ then the two processes will be identical up to time t. Hence the probabilities of events measurable up to time t will be identical when multiplied by $\mathbf{1}\left(\tau_i^t = g_i^t\left(x_i, \sigma_{\partial i}^{t-1}\right)\right)$, and the theorem follows. □

Using Eqs. (3) and (4), we can write the posterior on s computed by node i at time t, in terms of the probabilities $\mathbb{Q}[\cdot||\cdot]$:

$$\begin{aligned}\mathbb{P}\left[s|x_i,\sigma_{\partial i}^{t-1}\right] &\propto \mathbb{P}[s]\,\mathbb{P}\left[x_i,\sigma_{\partial i}^{t-1}|s\right]\\ &= \mathbb{P}[s]\,\mathbb{P}[x_i|s]\,\mathbb{P}\left[\sigma_{\partial i}^{t-1}|s,x_i\right]\\ &= \mathbb{P}[s]\,\mathbb{P}[x_i|s]\prod_{j\in\partial i}\mathbb{Q}\left[\sigma_j^{t-1}\middle|\middle|\sigma_i^{t-1},s\right]\end{aligned} \tag{5}$$

Recall that

$$\sigma_i(t) = g_{i,t}(x_i,\sigma_{\partial i}^{t-1}) = \arg\max_{s\in\mathcal{S}}\mathbb{P}\left[s|x_i,\sigma_{\partial i}^{t-1}\right]. \tag{6}$$

We have therefore reduced the problem of calculating $\sigma_i(t)$ to calculating $\mathbb{Q}[\cdot||\cdot]$. The following theorem is the heart of the dynamic cavity method and allows us to perform this calculation:

Theorem 4.4. *For any $i\in V$, $j\in\partial i$, $s\in\mathcal{S}$, $t\in\mathbb{N}$, τ_i^t and σ_j^t, we have*

$$\mathbb{Q}\left[\sigma_j^t\middle|\middle|\tau_i^t,s\right] = \sum_{\sigma_1^{t-1}\dots\sigma_{d-1}^{t-1}}\sum_{x_j}\mathbb{P}[x_j|s]\,\mathbf{1}\left[\sigma_j^t = g_j^t\left(x_j,(\tau_i^{t-1},\sigma_{\partial j\setminus i}^{t-1})\right)\right]\cdot\prod_{l=1}^{d-1}\mathbb{Q}\left[\sigma_l^{t-1}\middle|\middle|\sigma_j^{t-1},s\right], \tag{7}$$

where the neighbors of node j are $\partial j=\{i,1,2,\dots,d-1\}$.

The proof of this theorem is similar to the proof of Lemma 2.1 in [15], where the dynamic cavity method is introduced and applied to a different process.

Proof. In the modified process, the events in the different branches that i sees are independent. We therefore consider $G_{j\to i}$ only, and view it as a tree rooted at j. Also, for convenience we define $\sigma_i^t\equiv\tau_i^t$; note that the random variable σ_i^t does not exist in the modified process, as i's trajectory is fixed to τ_i.

Let $\underline{x}$ be the vector of private signals of j and all the vertices up to a depth t in $G_{j\to i}$ (call this set of vertices $V_{j\to i}^t$). For each $l\in\{1,\dots,d-1\}$, let $\underline{x}_l$ be the vector of private signals of $V_{l\to j}^{t-1}$. Thus, $\underline{x}=(x_j,\underline{x}_1,\underline{x}_2,\dots,\underline{x}_{d-1})$.

The trajectory σ_j^t is a function -deterministic, by our assumption- of $\underline{x}$ and τ_i^t. We shall denote this function by $F_{j\to i}$ and write $\sigma_j^t = F_{j\to i}^t(\underline{x},\tau_i^t)$. This function is uniquely determined by the update rules $g_l^t\left(x_l,\sigma_{\partial l}^{t-1}\right)$ for $l\in V_{j\to i}^t$.

We have therefore

$$\mathbb{Q}\left[\sigma_j^t=\lambda^t\middle|\middle|\tau_i^t,s\right] = \sum_{\underline{x}}\mathbb{P}[\underline{x}|s]\,\mathbf{1}\big(\lambda^t = F_{j\to i}^t(\underline{x},\tau_i^t)\big). \tag{8}$$

We now analyze each of the terms appearing in this sum. Since the private signals are independent conditioned on s, we have

$$\mathbb{P}[\underline{x}\,|s] = \mathbb{P}[x_j|s]\,\mathbb{P}[\underline{x}_1|s]\,\mathbb{P}[\underline{x}_2|s]\dots\mathbb{P}\left[\underline{x}_{d-1}\middle|s\right]. \tag{9}$$

The function $F_{j\to i}^t(\cdots)$ can be decomposed as follows:

$$\mathbf{1}\big(\lambda^t = F_{j\to i}^t(\underline{x},\tau_i^t)\big) = \sum_{\sigma_1^{t-1}\dots\sigma_{d-1}^{t-1}}\mathbf{1}\Big(\lambda^t = g_j^t(x_j,\sigma_{\partial j}^{t-1})\Big)\prod_{l=1}^{d-1}\mathbf{1}\Big(\sigma_l^{t-1} = F_{l\to j}^{t-1}(\underline{x}_l,\lambda^{t-1})\Big). \tag{10}$$

Using Eqs. (9) and (10) in Eq. (8) and separating terms that depend only on $\underline{x}_i$, we get

$$\mathbb{Q}\left[\sigma_j^t = \lambda^t \middle| \middle| \tau_i^t, s\right] = \sum_{\sigma_1^{t-1} \dots \sigma_{d-1}^{t-1}} \sum_{x_j} \mathbb{P}\left[x_j|s\right] \mathbf{1}\left(\lambda^t = g_j^t(x_j, \sigma_{\partial j}^{t-1}\right) \cdot$$
$$\cdot \prod_{l=1}^{d-1} \sum_{\underline{x}_l} \mathbb{P}\left[\underline{x}_l|s\right] \mathbf{1}\left(\sigma_l^{t-1} = F_{l\to j}^{t-1}(\underline{x}_l, \lambda^{t-1})\right).$$

The recursion follows by identifying that the product over l has argument $\mathbb{Q}\left[\sigma_l^{t-1} \middle| \middle| \sigma_j^{t-1}, s\right]$. □

4.2 The Agents' Calculations

We now describe how to perform the agents' calculations. At time $t = 0$ these calculations are trivial. Assume then that up to time t each agent has calculated the following quantities:

1. $\mathbb{Q}\left[\sigma_j^{t-1} \middle| \middle| \tau_i^{t-1}, s\right]$, for all $s \in \mathcal{S}$, for all $i, j \in V$ such that $j \in \partial i$, and for all τ_i^{t-1} and σ_j^{t-1}.
2. $g_i^t(x_i, \sigma_{\partial i}^{t-1})$ for all i, x_i and $\sigma_{\partial i}^{t-1}$.

Note that these can be calculated without making any observations – only knowledge of the graph G, $\mathbb{P}[s]$ and $\mathbb{P}[\underline{x}|s]$ is needed.

At time $t+1$ each agent makes the following calculations:

1. $\mathbb{Q}\left[\sigma_j^t \middle| \middle| \tau_i^t, s\right]$ for all $s, i, j, \sigma_j^t, \tau_i^t$. These can be calculated using Eq. (7), given the quantities from the previous iteration.
2. $g_i^{t+1}(x_i, \sigma_{\partial i}^t)$ for all i, x_i and $\sigma_{\partial i}^t$. These can be calculated using Eqs. (5) and (6) and the the newly calculated $\mathbb{Q}\left[\sigma_j^t \middle| \middle| \tau_i^t, s\right]$.

Since agent j calculates g_i^{t+1} for all i, then she, in particular, calculates g_j^{t+1}. This allows her to choose the (myopic) Bayes optimal action in rounds up to $t+1$, based on her neighbors' past actions. A simple calculation yields the following lemma.

Lemma 4.5. *In a tree graph G with maximum degree d, the agents can calculate their actions up to time t with computational effort $n2^{O(td)}$.*

In fact, each agent does not need to perform calculations for the entire graph. It suffices for node i to calculate quantities up to time t' for nodes at distance $t - t'$ from node i (there are at most $(d-1)^{t-t'}$ such nodes). A short calculation yields an improved bound on computational effort, stated in Theorem 3.2. The proof of Theorem 3.2 is relatively straightforward and is provided in the full version of this paper.

5 Majority dynamics: Proof of Theorem 3.5

We use $\widehat{\sigma}_i(t) \in \{-1, +1\}$ to denote votes under the majority dynamics.

Definition 5.1. *Under majority dynamics, each agent $i \in V$ chooses her vote in round $t+1$ according to the majority of the votes of her neighbors in round t, i.e.,*

$$\widehat{\sigma}_i(t+1) = \text{sign}\left(\sum_{j \in \partial i} \widehat{\sigma}_j(t)\right)$$

Ties are broken by flipping an unbiased coin.

Let $s \in \{-1, +1\}$ be drawn from a uniform prior and nodes receive 'private signals' $\widehat{\sigma}_i(0)$ that are correct with probability $1-\delta$, and independent conditioned on s. We consider an undirected d regular tree. The analysis is complicated by dependencies which have to be carefully handled. Our analytical approach here is again closely related to the dynamic cavity method.

Lemma 5.2. *Consider the setting in Theorem 3.5. Let i and j be adjacent nodes in the tree. Then for all $(\widehat{\sigma}_i^{t-1}, \widehat{\sigma}_j^{t-1}) \in \{-1, +1\}^{2t}$*

$$\mathbb{P}\left[\widehat{\sigma}_i(t) = -1 | \widehat{\sigma}_i^{t-1}, \widehat{\sigma}_j^{t-1}, s = +1\right] \le \delta_t \tag{11}$$

where δ_t is defined recursively by $\delta_0 \equiv \delta$, and

$$\delta_t \equiv \mathbb{P}\left[\text{Binomial}(d-1, \delta_{t-1}) \ge d/2 - 1\right] \tag{12}$$

Proof. We proceed by induction. Clearly Eq. (11) holds for $t = 0$. Suppose Eq. (11) holds for some t. We want to show

$$\mathbb{P}\left[\widehat{\sigma}_i(t+1) = -1 \,|\, \widehat{\sigma}_i^t, \widehat{\sigma}_j^t, s = +1\right] \le \delta_{t+1}\,, \tag{13}$$

for all $(\widehat{\sigma}_i^t, \widehat{\sigma}_j^t) \in \{-1, +1\}^{2(t+1)}$.

Let $l_1, l_2, \ldots, l_{d-1}$ be the other neighbors of node i (besides j). We will show that, in fact,

$$\mathbb{P}\left[\widehat{\sigma}_i(t+1) = -1 \,|\, \widehat{\sigma}_i^t, \widehat{\sigma}_j^t, \widehat{\sigma}_{l_1}^{t-1}, \ldots, \widehat{\sigma}_{l_{d-1}}^{t-1}, s = +1\right] \le \delta_{t+1}\,, \tag{14}$$

for all possible $\xi \equiv (\widehat{\sigma}_i^t, \widehat{\sigma}_j^t, \widehat{\sigma}_{l_1}^{t-1}, \widehat{\sigma}_{l_2}^{t-1}, \ldots, \widehat{\sigma}_{l_{d-1}}^{t-1})$.

We reason as follows. Fix the state of the world s and the trajectories $\widehat{\sigma}_i^t$ and $\widehat{\sigma}_j^t$. Now this induces correlations between the trajectories of the neighbors $l_1, \ldots, l_{d-1}$, caused by the requirement of consistency with the majority rule at node i, but *only up to time $t-1$*. If we further fix $\widehat{\sigma}_{l_m}^{t-1}$, then $\widehat{\sigma}_{l_m}(t)$ (and $\widehat{\sigma}_{l_m}$ at all future times) is conditionally independent of $\left(\widehat{\sigma}_{l_{m'}}^t\right)_{m' \neq m}$. Thus, we have

$$\mathbb{P}\left[\widehat{\sigma}_{l_m}(t) = -1 | \,\xi, s = +1\right] \;=\; \mathbb{P}\left[\widehat{\sigma}_{l_m}(t) = -1 | \,\widehat{\sigma}_{l_m}^{t-1}, \widehat{\sigma}_i^{t-1}, s = +1\right]\,,$$

and therefore, using the induction hypothesis

$$\mathbb{P}\left[\widehat{\sigma}_{l_m}(t) = -1 | \,\xi, s = +1\right] \le \delta_t \tag{15}$$

for all $m \in \{1, 2, \ldots, d-1\}$. Also, the actions $\widehat{\sigma}_{l_1}(t), \ldots, \widehat{\sigma}_{l_{d-1}}(t)$ are conditionally independent of each other given $\xi, s = +1$. We have

$$\widehat{\sigma}_i(t+1) = \text{sgn}(\widehat{\sigma}_j(t) + \widehat{\sigma}_{l_1}(t) + \ldots + \widehat{\sigma}_{l_{d-1}}(t))\,,$$

with sgn(0) being assigned value -1 or $+1$ with equal probability. This yields

$$\mathbb{P}\left[\widehat{\sigma}_i(t+1) = -1 \,|\, \xi, s = +1\right] \;\le\; \mathbb{P}\left[\text{Binomial}(d-1, \delta_t) \ge d/2 - 1\right]$$

from Eq. (15) and conditional independence of $\widehat{\sigma}_{l_1}(t), \ldots, \widehat{\sigma}_{l_{d-1}}(t)$. Thus, we obtain obtain Eq. (14). Eq. (13) follows by summing over $\widehat{\sigma}_{l_1}^{t-1}, \widehat{\sigma}_{l_2}^{t-1}, \ldots, \widehat{\sigma}_{l_{d-1}}^{t-1}$. □

Theorem 3.5 follows from Lemma 5.2 using a Chernoff bound (see the full version).

Theorem 3.5 is non-trivial for $d \ge 5$. The upper limit of the 'noise' δ for which it establishes rapid convergence approaches $(2e)^{-1}$ as d grows large.

Acknowledgments. We would like to thank Andrea Montanari, Elchanan Mossel and Allan Sly for valuable discussions.

References

[1] S. Aaronson, *The complexity of agreement*, in Proceedings of the thirty-seventh annual ACM symposium on Theory of computing, STOC '05, ACM, 2005, pp. 634–643.

[2] D. Acemoglu, M. A. Dahleh, I. Lobel, and A. Ozdaglar, *Bayesian learning in social networks*, The Review of Economic Studies, (2011).

[3] R. Aumann, *Agreeing to disagree*, The Annals of Statistics, 4 (1976), pp. 1236–1239.

[4] V. Bala and S. Goyal, *Learning from neighbours*, Review of Economic Studies, 65 (1998), pp. 595–621.

[5] O. Bandiera and I. Rasul, *Social networks and technology adoption in northern mozambique**, The Economic Journal, 116 (2006), pp. 869–902.

[6] A. V. Banerjee, *A simple model of herd behavior*, The Quarterly Journal of Economics, 107 (1992), pp. 797–817.

[7] D. H. Bikhchandani, S. and I. Welch, *A theory of fads, fashion, custom, and cultural change as informational cascade*, Journal of Political Economy, 100 (1992), pp. 992–1026.

[8] T. Conley and C. Udry, *Social learning through networks: The adoption of new agricultural technologies in ghana*, American Journal of Agricultural Economics, 83 (2001), pp. 668–673.

[9] M. H. DeGroot, *Reaching a consensus*, Journal of the American Statistical Association, 69 (1974), pp. 118–121.

[10] P. M. DeMarzo, D. Vayanos, and J. Zwiebel, *Persuasion bias, social influence, and unidimensional opinions.*, Quarterly Journal of Economics, 118 (2003), pp. 909 – 968.

[11] G. Ellison and D. Fudenberg, *Rules of thumb for social learning*, Journal of Political Economy, 110 (1995), pp. 93–126.

[12] D. Gale and S. Kariv, *Bayesian learning in social networks*, Games and Economic Behavior, 45 (2003), pp. 329–346.

[13] S. Goyal, *Connections: An Introduction to the Economics of Networks*, Princeton University Press, 2007.

[14] S. Istrail, *Statistical mechanics, three-dimensionality and np-completeness: I. universality of intracatability for the partition function of the ising model across non-planar surfaces (extended abstract)*, in Proceedings of the thirty-second annual ACM symposium on Theory of computing, STOC '00, New York, NY, USA, 2000, ACM, pp. 87–96.

[15] Y. Kanoria and A. Montanari, *Majority dynamics on trees and the dynamic cavity method*. Preprint. To appear in Annals of Applied Probability., 2011.

[16] Y. Kanoria and A. Montanari, *Subexponential convergence for information aggregation on regular trees*. 2011.

[17] H. Kohler, *Learning in social networks and contraceptive choice*, Demography, 34 (1997), pp. 369–383.

[18] L. Ménager, *Consensus, communication and knowledge: an extension with bayesian agents*, Mathematical Social Sciences, 51 (2006), pp. 274–279.

[19] M. Mueller-Frank, *A Comment on 'Bayesian Learning in Social Networks'*, SSRN eLibrary, (2011).

[20] ——, *Mixed Communication Structures: Opinion Formation in Social Networks with Bayesian and Non-Bayesian Agents*, SSRN eLibrary, (2011).

[21] R. Parikh and P. Krasucki, *Communication, consensus, and knowledge* 1*, Journal of Economic Theory, 52 (1990), pp. 178–189.

[22] M. Raynal, *Fault-tolerant agreement in synchronous message-passing systems*, Synthesis Lectures on Distributed Computing Theory, 1 (2010), pp. 1–189.

[23] D. Rosenberg, E. Solan, and N. Vieille, *Informational externalities and emergence of consensus*, Games and Economic Behavior, 66 (2009), pp. 979 – 994. Special Section In Honor of David Gale.

[24] J. Sebenius and J. Geanakoplos, *Don't bet on it: Contingent agreements with asymmetric information*, Journal of the American Statistical Association, 78 (1983), pp. 424–426.

[25] L. Smith and P. Sørensen, *Pathological outcomes of observational learning*, Econometrica, 68 (2000), pp. 371–398.

Full version of the paper

This full version includes additional discussion (including several clarifying remarks and footnotes), missing proofs and numerical results. There are three additional sections: an unnumbered section between Section 3 and 4 that proves Proposition 3.1, Section 6 containing numerical results, and Section 7 that contains a concluding discussion and a comparison with Aaronson's work on the complexity of agreement. We attempted to have consistent numbering of sections, theorems, lemmas etc. in the extended abstract and in this full version.

1 Introduction

The importance of social learning in networks has been demonstrated in a wide variety of settings, for example, adoption of agricultural technology in Ghana [8] and Mozambique [5], and choice of contraceptives by European women [17]).

Accordingly, developing and understanding models of social learning has been a goal of theoretical economics for the past few decades (cf., Goyal [13]). Typical models in this context assume a *pure information externality*; agent payoffs depend only on the action they choose and an underlying 'state of the world', and not on the actions of others. Agents observe the actions of their 'neighbors', but payoffs are not observed (or observed with noise) ex interim. Typically, all agents have the same utility function. Each agent receives a private signal that contains noisy information about the state of the world. Agents choose actions to maximize expected payoff, given their own private signal and their observations of the actions chosen by others. The premise in such models is that "actions speak louder than words" – agents learn by observing each others' actions, and not by communicating directly.

A standard modeling approach in economics assumes that people (agents) are Bayesian or 'rational' (cf., [3, 24]); agents assume a prior distribution over possible events and pick actions that maximize the expectation of some utility function, conditioned on the information available to them. Accordingly, many studies of social learning assume Bayesian agents [6, 7, 25, 12]. Alternatively, agents in a social network are modeled as 'boundedly rational', attempting to maximize payoffs using some heuristic or 'naïve' rule (e.g., [11, 4]).

Fully Bayesian models have two advantages over models that assume bounded rationality and prescribe thumb rules for agent behavior: First, any bounded rationality approach is bound to involve a somewhat arbitrary decision of which heuristics the agents use. Second, a game theoretic analysis of strategic players is possible only if the players choose actions that are optimal by some criterion. Hence game-theoretic analyses of learning on networks (e.g., [23]) often opt for the more difficult but fully Bayesian model.

Much progress has been achieved in models where Bayesian agents act *sequentially*, such as the herd behavior models of Banerjee [6], Bikhchandani, Hirshleifer and Welch [7], Smith and Sørenson [25], and Acemoglu et al [2]. Here, the interaction is *not bidirectional*: each agent acts only once, taking into account the actions of her predecessors.

In comparison, our understanding of Bayesian agents who act *repeatedly* is much more limited. Gale and Kariv [12] consider Bayesian agents on a network who choose actions repeatedly. They show, in the spirit of Aumann's agreement theorem [3], that agents on a network converge to the same action under some conditions[1]. Related work by Rosenberg, Solan and Vieille [23] and Ménager [18] sheds more light on the phenomenon of agreement on actions and the conditions in

[1] A gap in the proof of Gale and Kariv's agreement theorem was recently pointed out [19]. However, recent works [23, 20] establish similar results in more general settings.

which it arises. However, several questions remain essentially unanswered, including: (I) What action do the agents converge to, e.g., what is the distribution of this consensus action?[2] (II) What are the dynamics of such interactions, e.g., what is the rate of agreement/convergence?

The difficulty in answering these questions, and in analyzing bidirectional models in general, seems to be closely related to a lack of understanding of the computational/algorithmic aspects of such models[3]. Moreover, computational tractability is a natural requirement for any decision rule that aims to model behavior of real agents. This motivates the focal question of the current work:

(III) *Are the computations required of Bayesian agents feasible?*

We consider a model that features repeated bidirectional interaction between fully Bayesian agents connected by a social network. Our model is a specialization of the model of Gale and Kariv [12]. We consider a group of Bayesian agents, each with a private signal that carries information on an unknown state of the world s. The individuals form a social network, so that each observes the actions of some subset of others, whom we call her neighbors. The agents must repeatedly choose between a set of possible actions, the relative merit of which depends on the state of the world s. The agents iteratively learn by observing their neighbors' actions, and picking an action that is myopically optimal, given their information. Thus, the interaction between agents is not strategic, and is characterized by information externalities.

Even in the simple case of two states of the world, binary private signals and two possible actions, the required calculations appear to be very complicated. A naïve dynamic programming algorithm[4] is exponential in the number of individuals. Since at iteration t one may consider only agents at distance t, then in graphs of maximum degree d (on which we focus) the number of individuals to consider is $O(\min(n, d^t))$, and the computational effort required of each individual to compute their action at time t is $t2^{O(\min(n,d^t))}$. Obviously, this grows very rapidly. As Gale and Kariv remark [12], "The computational difficulty of solving the model is massive even in the case of three persons." This prevents them from even simulating networks with more than three nodes.

We describe a novel algorithm for the agents' calculation in our model, when the social network graph is a tree. This algorithm has running time that is exponentially smaller than the naïve dynamic program, reducing the computational effort to $2^{O(\min(n,td))}$.

Using our algorithm we are able to run numerical simulations of the social learning process. This extends the work of Gale and Kariv [12], who simulated the process for three agents, to much larger networks[5]. We use our algorithm to investigate questions (I) and (II): We numerically evaluate the probability that the agents learn the optimal action, and its progress with time. We observe rapid learning of the optimal action in certain previously unexplored settings. : We consider a model with two possible states of the world and two corresponding actions ('votes'), so the agents are in effect trying to estimate the state of the world and revealing their estimates to their neighbors. The social networks in these analyses were chosen to be d-regular (infinite) trees, i.e., trees in which each node has d neighbors. The simulations suggest that, on regular trees, the number of iterations needed under Bayesian learning to estimate s correctly with probability $1 - \epsilon$ is $O(\log\log(1/\epsilon))$.

We conjecture that the error probability under Bayesian updates is no larger than the error probability under a different 'majority' update rule, in which agents adopt the opinion of the majority of their neighbors in the previous round. Our numerical results support this conjecture.

[2] In fact, convergence is *not* always obtained [23]. The exact conditions under which convergence occurs are not known.

[3] Mathematical and computational tractability often coincide, e.g., see [14].

[4] Although this algorithm seems to be well known, we could not find a complete description of it in the literature and hence supply it for completeness in Proposition 3.1.

[5] In our numerical analyses, agents receive information (directly or indirectly) from hundreds of distinct nodes.

We prove that for the majority update rule, the number of iterations needed to estimate s correctly with probability $1-\epsilon$ is $O(\log\log(1/\epsilon))$, for regular trees of degree at least five. Our conjecture then implies, again, that the number of iterations needed to estimate s correctly with probability $1-\epsilon$ is $O(\log\log(1/\epsilon))$. Thus, assuming the conjecture, the computational effort required of Bayesian agents drops from quasi-polynomial in $1/\epsilon$ (using the naïve dynamic program) to polynomial in $\log(1/\epsilon)$ (i.e., polylogarithmic in $1/\epsilon$), making Bayesian learning computationally tractable. Thus, our results shed new light on question (III), suggesting a positive answer in the case of tree graphs.

The restriction of the discussion to tree or tree-like social networks certainly excludes many natural settings that tend to exhibit highly clustered social graphs. However, in some cases artificially constructed networks have no or few loops by design; these include some highly hierarchical or compartmentalized organizations, as well as some physical communication networks where redundancy is expensive, and the least expensive connected network is a tree. Furthermore, the fact that this non-trivial class of networks does not present a major computational hurdle for fully Bayesian calculations is in itself somewhat surprising.

The work most closely related in spirit to the present one is that of Aaronson on the complexity of agreement [1]. There, as in here, a positive computational result was found somewhat unexpectedly. We present a detailed comparison with Aaronson's work at the end of the full version of this paper.

Technical contributions. A key technique used in this paper is the dynamic cavity method, introduced by Kanoria and Montanari [15] in their study of 'majority updates' on trees. This technique is a dynamical version of the cavity method of statistical physics and appears promising for the analysis of iterative tree processes in general. The key idea is the following: In a dynamical setting on a tree graph, there is correlation in the trajectories of neighbors of a node due to a nodes own past actions. The dynamic cavity method allows to exactly account for these correlations. In this work, we use this method for the first time to give a new algorithmic result, enabling efficient computation by nodes. This is in contrast to the case of majority updates, where the update rule is computationally trivial. Our algorithmic and analytical approach leveraging the dynamic cavity method should be applicable to a range of iterative update situations on trees.

Our second main technical contribution is our proof of doubly exponentially fast convergence of majority dynamics on regular trees. Though this result is somewhat auxiliary in the context of the current work, it should be of independent interest, since, e.g., majority dynamics may serve as a very simple distributed computation protocol or consensus protocol in a distributed system[22]. Besides, majority dynamics is a reasonable model of social learning with bounded rationality.

1.1 Outline of the paper

We describe and discuss our model in Section 2. We state our main results in Section 3. Section 3.2.1 presents a naïve dynamic programming algorithm. Section 4 presents our main contribution: a dynamic cavity method based algorithm for tree graphs, along with a proof of correctness and analysis of running time. We prove rapid convergence of majority dynamics in Section 5. Section 6 discusses our conjecture regarding convergence (Conjecture 3.4) and presents numerical results.

2 Model

The model we consider is a simplified version of the model of social learning introduced by Gale and Kariv [12]. We first state our model. In Section 2.1, we motivate our model in the context of rational agents. Finally, Section 2.3 compares our model with other models, including that of Gale and Kariv [12].

Consider a directed graph $G = (V, E)$, representing a network of $n \equiv |V|$ agents, with V being the set of agents and E being the social ties between them. A directed edge (i, j) indicates that agent i observes agent j. In most of this paper, we study the special case of undirected graphs, where relationships between agents are bidirectional.

Agents attempt to learn the true *state of the world* $s \in \mathcal{S}$, where $\mathcal{S}$ is finite. The information available to them are their *private signals* x_i, where $x_i \in \mathcal{X}$ and $\mathcal{X}$ is finite. We assume a general distribution of $(s, x_1, \ldots, x_n)$, under the condition that *private signals are independent conditioned on* s, i.e. $\mathbb{P}[s, x_1, \ldots, x_n] = \mathbb{P}[s] \prod_{i \in V} \mathbb{P}[x_i|s]$.

In each discrete time period (or round) $t = 0, 1, \ldots$, agent i chooses action $\sigma_i(t) \in \mathcal{S}$, which we call a 'vote'[6]. Agents observe the votes cast by their neighbors in G. Thus, at the time of voting in round $t \geq 1$, the information available to an agent consists of the private signal she received initially, along with the votes cast by her neighbors in rounds up to $t - 1$. In each round, each agent votes for the state of the world that she currently believes is most likely, given the Bayesian posterior distribution she computes.

Formally, let $\sigma_i^t \equiv (\sigma_i(0), \sigma_i(1), \ldots, \sigma_i(t))$ denote all of agent i's votes, up to and including time t. Let ∂i denote neighbors of agent i, not including i, i.e., $\partial i \equiv \{j : (i, j) \in E\}$. We write $\sigma_{\partial i}^t = \{\sigma_j^t\}_{j \in \partial i}$, i.e., $\sigma_{\partial i}^t$ are the votes of i's neighbors up to and including time t. Then the agents' votes $\sigma_i(t)$ are given by

$$\sigma_i(t) = \arg\max_{s \in \mathcal{S}} \mathbb{P}\left[s \middle| x_i, \sigma_{\partial i}^{t-1}\right],$$

where, if the maximum is attained by more than one value, some deterministic tie breaking rule is used. We denote $\sigma_i = (\sigma_i(0), \sigma_i(1), \ldots)$.

Note that $\sigma_i(t)$ is a deterministic function of x_i and $\sigma_{\partial i}^{t-1}$. We denote this function $g_i(t) : \mathcal{X} \times |\mathcal{S}|^{t|\partial i|} \to \mathcal{S}$:

$$\sigma_i(t) = g_{i,t}(x_i, \sigma_{\partial i}^{t-1})$$

For convenience, we also define the vector function g_i^t that returns the entire history of i's votes up to time t, $g_i^t \equiv (g_{i,0}, g_{i,1}, \ldots, g_{i,t})$, so that

$$\sigma_i^t = g_i^t(x_i, \sigma_{\partial i}^{t-1}).$$

2.1 Discussion of our Model

The decision rules can be interpreted/motivated as follows. Suppose $\mathbb{P}[s]$, $\mathbb{P}[\underline{x}|s]$ and G are common knowledge[7]. Suppose that for each state of the world s, action σ has utility one when the state of the world is $s = \sigma$, and zero otherwise. Thus, the action that myopically maximizes the expected utility corresponds to the maximum *a posteriori* probability (MAP) estimator of the state of the world. This leads to the decision rule we consider[8], with $\sigma_i(t)$ being chosen as $\arg\max_{s \in \mathcal{S}} \mathbb{P}\left[s \middle| x_i, \sigma_{\partial i}^{t-1}\right]$. We would like to emphasize that we only restrict the 'action' space $\mathcal{A}$ to $\mathcal{S}$ (thus calling actions as 'votes'), with this simple "1 if you vote correctly, 0 otherwise" utility function, for simplicity of presentation. Indeed, our main computational result, Theorem 3.2 admits a trivial generalization

[6]We choose to use "vote" rather than the more usual "act" since, in this simplified model, the set of actions and the set of states of the world are identical, so choosing an action is equivalent to picking a possible state of the world as a guess for the true s.

[7]We say that G is common knowledge, if, for each pair of agents i and j, both know G, both know that they both know G, both know that they both know that they both know G, and so on.

[8]Here, the decision rules are also taken as common knowledge, as per the norm in definitions of equilibria (like Nash equilibrium for games) in economics.

to the case of a general finite action space $\mathcal{A}$ and a general common utility function $U : \mathcal{A} \times \mathcal{S} \to \mathbb{R}$. Section 4 includes a precise description of why this is the case.

A natural objection to such a model of behavior is that the agents should want to maximize the discounted sum of their future utilities, instead of making the myopic optimal choice. Gale and Kariv [12] deal with this by assuming a continuum of agents at each node, so that no one of them can hope to influence the future by their choice of votes. We can do the same here: Then $\{\sigma_i(t)\}$ form a weak perfect Bayesian equilibrium (cf. [12, Definition 1]) for the right utility function (see above).

Rosenberg, Solan and Vieille[23] consider a model with fully rational agents - one per node - maximizing the discounted sum of payoffs. In this setting strategic behavior arises, and so they study the properties of the model's Nash equilibria. They extend to this model many agreement results (e.g., those of Gale and Kariv [12], Parikh and Krasucki [21] and Sebenius and Geanakoplos [24]) that were previously known to apply to myopic agents. Our results apply only to the myopic model, which can be thought of as a stepping stone towards the strategic model, being a special case corresponding to discount factor 0.

2.2 Scaling regime

The scaling regime we consider is the following: We treat the cardinalities of the sets $\mathcal{S}$ and $\mathcal{X}$ and the maximum degree d of the network as fixed parameters, whereas the scaling parameters are the number of agents $n \equiv |V|$, and the number of iterations t. An alternative scaling parameter to t is $1/\epsilon$, where $\epsilon > 0$ is the desired probability of error.

Such a scaling regime is of much interest with the emergence of massive online networks, where non-expert agents interact on a variety of issues, and individual agents are expected to have limited private information, and typically choose from a (relatively) small set of available actions.

Our choice of scaling variables is in contrast, for instance, to the work of Aaronson on the complexity of agreement [1]. Aaronson focuses on the case of two agents, and allows the set of possible private signals to grow, aiming to reach agreement with minimum communication and computational cost. In our case, the objective must clearly be the computational cost, since the 'protocol' is defined by the model itself, leading to a fixed communication cost.

2.3 Comparison with other models

The model presented above is a special case of the Gale-Kariv model [12], which we refer to as the GK model henceforth.

In the GK model there is an all-encompassing 'state of nature' $\omega \in \Omega$, the agents have a set of possible actions $\mathcal{A}$, and the utility of the actions is a general function $U : \mathcal{A} \times \Omega \to \mathbb{R}$. The utility functions of all agents are the same. The agents receive general private signals $(x_1(\omega), \ldots, x_n(\omega))$.

We specialize the GK model as follows:

- We reduce ω to $(x_1, \ldots, x_n, s)$, where $x_i \in \mathcal{X}$ is agent i's private signal and s belongs to a given (finite) set $\mathcal{S}$. We restrict the utility function to depend only on s and a, i.e., $U : \mathcal{A} \times \mathcal{S} \to \mathbb{R}$. In particular, U does not depend directly on the x_i's.
- We demand that the x_i's be conditionally independent of each other, given s.
- We demand that $\mathcal{S}$, $\mathcal{X}$ and $\mathcal{A}$ be *finite* sets with bounded cardinalities.

Our choice of a 'state of the world' s and conditionally independent private signals, with a utility function dependent only on s and a, is typical in herd behavior models (e.g., Banerjee [6]

Bikhchandani, Hirshleifer and Welch [7], Smith and Sørensen [25]). It is also the basis of the model of boundedly-rational agents on social networks studied by Bala and Goyal [4]. Nevertheless it is important to note that our first and second assumptions represents an important specialization of the GK model. The third assumption corresponds to our choice of scaling regime (see Section 2.2).

Our assumptions play a crucial role in the efficient approach we develop to enable the computation of Bayesian posteriors. As discussed above, we allow the number of agents n to scale. Hence, one might expect a general utility function that depends on all private signals to cause a computational burden that grows exponentially in n, just to enumerate the different utility functions possible. Our assumption 1 above eliminates this difficulty. Similarly, dependent private signals might lead to the problem of summing over exponentially many different possibilities.

While we know of no formal computational hardness results for Bayesian calculations on social networks, we conjecture that the removal of any of the first two assumptions, or the consideration of general graphs (i.e., not tree or tree-like graphs), makes the agents' calculations #P hard. A proof of this conjecture would be a natural complement to this work.

3 Main results

3.1 Efficient computation

To the best of our knowledge, the literature (e.g., [12, 23]) does not contain an explicit description of an algorithm to compute the actions chosen by agents in our model. However, it seems that a dynamic programming algorithm that performs this computation is well known. The proposition below states the computational complexity of this algorithm.

Proposition 3.1. *On any graph G, there is a dynamic programming (DP) based algorithm that allows agents to compute their actions up to time t with computational effort $t2^{O(\min(n,(d-1)^t))}$, where d is the maximum degree of the graph.*

The algorithm leading to Proposition 3.1 is described in the unnumbered Section immediately after this one, and before Section 4.. This proposition provides the baseline or benchmark that we compare our other algorithmic results to. In particular, we do not consider this algorithm a major contribution of this work.

A key advantage of the DP algorithm is that it works for any graph G. The disadvantage, of course, is that the computational effort required grows doubly exponentially in the number of iterations t.

Our main result concerns the computational effort needed when the graph G is a tree[9]. We show that computational effort exponentially lower than that of the naive DP suffices in this case.

Theorem 3.2. *In a tree graph G with maximum degree d, each agent can calculate her actions up to time t with computational effort $t2^{O(\min(n,td))}$.*

The algorithm we use employs a technique called the dynamic cavity method [15], previously used only in analytical contexts. Section 4 contains a full description of the algorithm and analysis leading to Theorem 3.2.

An apparent issue is that the computational effort required is exponential in t; typically, exponentially growing effort is considered as large. However, in this case, we expect the number of iterations t to be typically quite small, for two reasons: (1) In many settings, agents appear to converge to the 'right' answer in a very small number of iterations [12]. In Section 3.2 below, we

[9]By *tree*, we mean a graph that contains no loops. This is sometimes called a 'forest' in the literature.

conjecture that if ϵ is the desired probability of error, then the number of rounds required should be only $O(\log\log(1/\epsilon))$, leading to computational effort of only polylog$(1/\epsilon)$. Having obtained an approximately correct estimate, the agents would have little incentive to continue observing their neighbors actions and updating their beliefs.[10] (2) In many situations we would like to model, we might expect only a small number (e.g., single digit) number of iterative updates to occur, irrespective of network size etc. For instance, voters may discuss an upcoming election with each other over a short period of time, ending on the election day when ballots are cast.

3.2 Convergence

Since an agent gains information at each round, and since she is Bayesian, then the probability that she votes correctly is non-decreasing in t, the number of rounds. We say that the agent *converges* if this probability converges to one, or equivalently if the probability that the agent votes incorrectly converges to zero[11].

We say that there is *doubly exponential convergence* to the state of the world s if the maximum single node error probability $\max_{i\in V}\mathbb{P}\left[\sigma_i(t)\neq s\right]$ decays with round number t as

$$\max_{i\in V}\mathbb{P}\left[\sigma_i(t)\neq s\right]=\exp\left(-\Omega(b^t)\right), \tag{1}$$

where $b>1$ is some constant.

The following is an immediate corollary of Theorem 3.2.

Corollary 3.3. *Consider iterative Bayesian learning on a tree of with maximum degree d. If we have doubly exponential convergence to s, then computational effort that is polynomial in* $\log(1/\epsilon)$ *(i.e., polylogarithmic in* $1/\epsilon$*) suffices to achieve error probability* $\mathbb{P}\left[\sigma_i(t)\neq s\right]\leq\epsilon$ *for all i in V.*

Note that if weaken our assumption to doubly exponential convergence in only a subset $V_{\mathrm{c}}\subseteq V$ of nodes, i.e., $\max_{i\in V_{\mathrm{c}}}\mathbb{P}\left[\sigma_i(t)\neq s\right]=\exp\left(-\Omega(b^t)\right)$, we still obtain a similar result with nodes in V_{c} efficiently learning s.

We are handicapped by the fact that very little in known rigorously about convergence of iterative Bayesian learning in this sense (cf. questions (I) and (II) in Section 1). Nevertheless, we provide the evidence for doubly exponential convergence on trees: We study a situation with two possible states of the world and two possible private signal values. First, on a regular *directed* tree we show that except for the case of very noisy signals, we have doubly exponential convergence if the degree is at least five. Second, we state a conjecture and show that it implies doubly exponential convergence of iterative Bayesian learning also on undirected trees. We provide numerical evidence in support of our conjecture.

3.2.1 Bayesian vs. 'majority' updates

We conjecture that on regular trees, iterative Bayesian learning leads to lower error probabilities (in the weak sense) than a very simple alternative update rule we call 'majority dynamics'[15]. Under this rule, the agents adopt the action taken by the majority of their neighbors in the previous iteration (this is made precise in Definition 5.1). Our conjecture seems natural since the iterative Bayesian update rule chooses the vote in each round that (myopically) minimizes the error probability. We use $\widehat{\sigma}_i(t)$ to denote votes under the majority dynamics.

[10] Thus, $1/\epsilon$ serves as an alternative scaling parameter to t.

[11] Note that this notion of 'convergence' differs greatly from the 'agreement on actions' sense in which the term is sometimes used.

Round	Bayesian	Majority
0	0.15	0.15
1	$2.66119 \cdot 10^{-2}$	$2.66119 \cdot 10^{-2}$
2	$7.61832 \cdot 10^{-4}$	$1.67525 \cdot 10^{-3}$
3	$2.83839 \cdot 10^{-7}$	$8.37462 \cdot 10^{-6}$
4	$1.41065 \cdot 10^{-12}$	$2.48525 \cdot 10^{-10}$

Table 1: Error probability on a regular tree with $d = 5$ and $\mathbb{P}[x_i \neq s] = 0.15$, for (i) Bayesian and (ii) majority updates. The agents break ties by picking their original private signals.

Conjecture 3.4. *Consider binary $s \sim$ Bernoulli$(1/2)$, and binary private signals that are independent identically distributed given s, with $\mathbb{P}[x_i \neq s] = 1 - \delta$ for some $\delta \in (0, 1/2)$. Let the majority dynamics (see Definition 5.1) be initialized with the private signals, i.e., $\widehat{\sigma}_i(0) = x_i$ for all $i \in V$. Then on any infinite regular tree, for all $t \geq 0$, we have*

$$\mathbb{P}[\sigma_i(t) \neq s] \leq \mathbb{P}[\widehat{\sigma}_i(t) \neq s] \ . \tag{2}$$

In words, the error probability under iterative Bayesian learning is no larger than the error probability under majority dynamics, after the same number of iterations.

In Section 5, we show doubly exponential convergence for majority dynamics on regular trees:

Theorem 3.5. *Consider binary $s \sim$ Bernoulli$(1/2)$, and binary initial votes $\widehat{\sigma}_i(0)$ that are independent identically distributed given s, with $\mathbb{P}[\widehat{\sigma}_i(0) \neq s] = 1 - \delta$ for some $\delta \in (0, 1/2)$. Let i be any node in an (undirected) d regular tree for $d \geq 5$. Then, under the majority dynamics,*

$$\mathbb{P}[\widehat{\sigma}_i(t) \neq s] = \exp\Big[- \Omega\Big(\big(\tfrac{1}{2}(d-2)\big)^t \Big) \Big].$$

when $\delta < (2e(d-1)/(d-2))^{-\frac{d-2}{d-4}}$.

Thus, if Conjecture 3.4 holds:

- We have doubly exponential convergence for iterative Bayesian learning on regular trees with $d \geq 5$, implying that for any $\epsilon > 0$, an error probability ϵ can be achieved in $O(\log\log(1/\epsilon))$ iterations with iterative Bayesian learning.
- Combining with Corollary 3.3), we see that the computational effort that is polylogarithmic in $(1/\epsilon)$ suffices to achieve error probability $1/\epsilon$.

This compares favorably with the quasi-poly$(1/\epsilon)$ (i.e., $\exp\big(\text{polylog}(1/\epsilon)\big)$) upper bound on computational effort that we can derive by combining Conjecture 3.4 and the naïve dynamic program described. Indeed, based on recent results on subexponential decay of error probability with the number of private signals being aggregated [16], it would be natural to conjecture that the number of iterations T needed to obtain an error probability of ϵ obeys $(d-1)^T \geq C\log(1/\epsilon)$ for any $C < \infty$, for ϵ small enough. This would then imply that the required computational effort using the naïve DP on a regular tree of degree d grows faster than any polynomial in $1/\epsilon$.

Since we are unable to prove our conjecture, we instead provide numerical evidence for it in Table 1. Further numerical results are presented in Section 6, along with a discussion of the difficulties in proving Conjecture 3.4. All computations leading to our numerical results are exact (modulo finite precision arithmetic), and were performed using the dynamic cavity equations. The results are all consistent with our conjecture over different values of d and $\mathbb{P}[x_i \neq s]$.

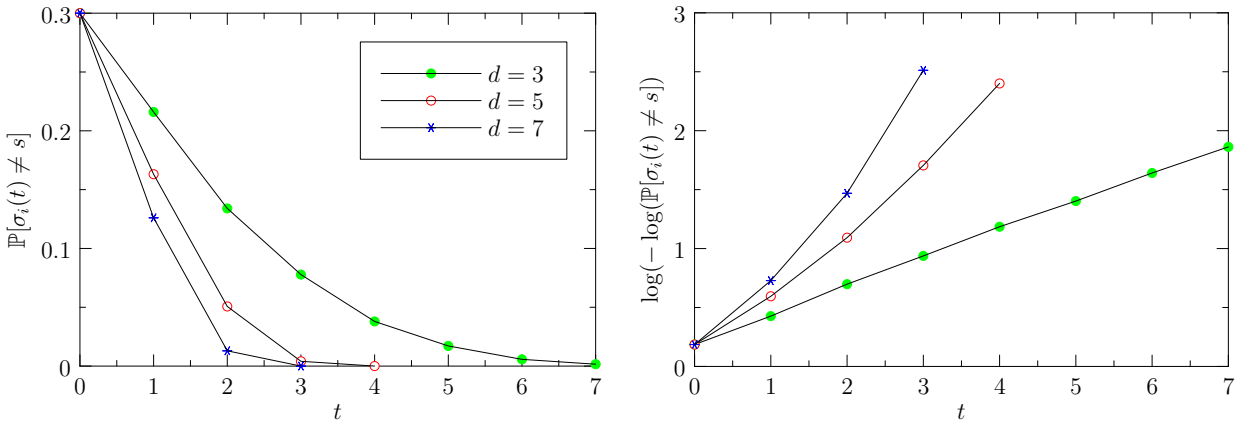

Figure 1: Error probability decay on regular trees for iterative Bayesian learning, with $\mathbb{P}[x_i \neq s] = 0.3$ (see Section 6). The data used to generate this figure is presented in Table 3.

We would like to emphasize that several of the error probability values could be feasibly computed only because of our new efficient approach to computing the decision functions employed by the nodes. For instance, with $d = 5$, computing the decision function at iteration 3 using the dynamic program (see Proposition 3.1and Section 3.2.1) would require enumeration over $2^{80} \sim 10^{24}$ possibilities, which is infeasible even on state-of-the-art supercomputers. With our approach, we are able to compute the decision function at iteration 3 and even at iteration 4, on a desktop machine. This aggregates information from the ~ 400 nodes within 4 hops of a given node.

Figure 1 plots decay of error probabilities in regular trees for iterative Bayesian learning with $\mathbb{P}[x_i \neq s] = 0.3$, where the agents break ties by picking their original private signals. Each of the curves (for different values of d) in the plot of $\log(-\log \mathbb{P}[\sigma_i(t) \neq s])$ vs. t appear to be bounded below by straight lines with positive slope, suggesting doubly exponential decay of error probabilities with t. The empirical rapidity of convergence, particularly for $d = 5, 7$, is noteworthy.

A Simple Algorithm: Proof of Proposition 3.1

A sign of the complexity of evaluating the Bayesian decision function $g_i^t(x_i, \sigma_{\partial i}^{t-1})$, is that even the brute-force solution approach to it is not trivial. We therefore describe it here.

One way of thinking of the agents' calculation is to imagine that they keep a long list of all the possible combinations of private signals of all the other agents, and at each iteration cross out entries that are inconsistent with the signals that they've observed from their neighbors up to that point. Then, they calculate the probabilities of the different possible states of the world by summing over the entries that have yet to be crossed out.

This may not be as simple as it seems. To understand which private signal vectors are ruled out by the observed actions of neighbors, an agent "simulates" the network for every possible private signal vector: Each agent calculates the function g_i^t for every other agent i and every possible set of observations by i. We formalize this below.

Let $\underline{x} \in \mathcal{X}^n$ be the vector of private signals $(x_i)_{i \in V}$. The trajectory of i, denoted by σ_i, is a deterministic function of $\underline{x}$. Assume then that up to time $t-1$ each agent has calculated the trajectory $\sigma_i^{t-1}(\underline{x})$ for all possible private signal vectors $\underline{x}$ and all agents i. This is trivial for $t - 1 = 0$.

We say that $\underline{y} \in \mathcal{X}^n$ is feasible for i at time t if $x_i = y_i$ and $\sigma_{\partial i}^t = \sigma_{\partial i}^t(\underline{y})$. We denote this set of

feasible private signal vectors by $I_i^t(x_i, \sigma_{\partial i}^t) \subseteq \mathcal{X}^n$. To calculate $\sigma_i^t(\underline{x})$, one observe that for all i, x_i and $\sigma_{\partial i}^{t-1}$, we have

$$\begin{aligned}\mathbb{P}\left[s|x_i, \sigma_{\partial i}^{t-1}\right] &\propto \mathbb{P}\left[s\right]\mathbb{P}\left[x_i, \sigma_{\partial i}^{t-1}|s\right] \\ &= \mathbb{P}\left[s\right] \sum_{\underline{y}\in I_i^{t-1}\left(y_i, \sigma_{\partial i}^{t-1}\right)} \mathbb{P}\left[\underline{x} = \underline{y}\middle|s\right]\end{aligned}$$

and

$$g_{i,t}(x_i, \sigma_{\partial i}^{t-1}) = \arg\max_{s\in\mathcal{S}} \mathbb{P}\left[s|x_i, \sigma_{\partial i}^{t-1}\right]$$

by definition. We use the standard abusive notation $\mathbb{P}\left[x_i\right]$ instead of $\mathbb{P}\left[x_i = y_i\right]$, $\mathbb{P}\left[\sigma_j^t\right]$ instead of $\mathbb{P}\left[\sigma_j^t = \omega_j^t\right]$, etc.

It is easy to verify that using the equations above, the 'simulation' can be advanced from $t-1$ to t with additional computational effort $O(n|\mathcal{X}|^n)$. Thus, the calculation of $\sigma_i^t(\underline{x})$ for all i and $\underline{x}$ requires total effort $O(tn|\mathcal{X}|^n)$. This leads to an upper bound of $t2^{O(n)}$ for this method. Note that up to time t an agent need only consider agents at distance at most t, so on a graph with maximum degree d, we obtain a bound of $t2^{O((d-1)^t)}$. This improves the bound above for 'large' graphs, i.e., graphs for which $n > (d-1)^t$ for relevant values of t. Thus, we obtain the result stated in Proposition 3.1.

We call this algorithm 'the naïve dynamic program'.

4 The Dynamic Cavity Algorithm on Trees

In this section we develop the dynamic cavity algorithm leading to Theorem 3.2. We present the core construction and key technical lemmas in Section 4.1. In Section 4.2, we show how this leads to an efficient algorithm for the Bayesian computations on tree graphs, and prove Theorem 3.2.

Assume in this section that the graph G is a tree with finite degree nodes. For $j \in \partial i$, let $G_{j\to i} = (V_{j\to i}, E_{j\to i})$ denote the connected component containing node j in the graph G with the edge (i, j) removed. That is, $G_{j\to i}$ is j's subtree when G is rooted at i.

4.1 The Dynamic Cavity Method

We consider a modified process where agent i is replaced by a *zombie agent* who takes a fixed sequence of actions $\tau_i = (\tau_i(0), \tau_i(1), \ldots)$, and *the true state of the world is assumed to be some fixed s*. Furthermore, this 'fixing' goes unnoticed by the agents (except i, who is a zombie anyway) who perform their calculations assuming that i is her regular Bayesian self. Formally:

$$\sigma_j(t) = \begin{cases} \tau_i(t) & \text{for } j = i\,, \\ g_{j,t}(x_j, \sigma_{\partial j}^{t-1}) & \text{for } j \neq i\,. \end{cases}$$

We denote by $\mathbb{Q}\left[A||\tau_i, s\right]$ the probability of event A in this modified process.

Remark 4.1. *We emphasize that the modified process with a 'zombie' agent is a* theoretical construct *we use to derive an efficient implementation for the iterative Bayesian decision rules. Our algorithm does not involve actual replacement of nodes in the network.*

This modified process is easier to analyze, as the processes on each of the subtrees $V_{j\to i}$ for $j \in \partial i$ are independent: Recall that private signals are independent conditioned on s, and the zombie agent ensures that the subtrees stay independent of each other. This is formalized in the following claim, which is immediate to see:

Claim 4.2. *For any $i \in V$, $s \in \mathcal{S}$ and any trajectory τ_i, we have*

$$\mathbb{Q}\left[\sigma_{\partial i}^t \middle|\middle| \tau_i, s\right] = \prod_{j\in\partial i} \mathbb{Q}\left[\sigma_j^t \middle|\middle| \tau_i^t, s\right]. \tag{3}$$

(Since σ_j^t is unaffected by $\tau_i(t')$ for all $t' > t$, we only need to specify τ_i^t, and not the entire τ_i.)

Now, it might so happen that for some number of steps the 'zombie' agent behaves exactly as may be expected of a rational player. More precisely, given $\sigma_{\partial i}^{t-1}$, it may be the case that $\tau_i^t = g_i^t\left(x_i, \sigma_{\partial i}^{t-1}\right)$ for some x_i. This event provides the connection between the modified process and the original process, and is the inspiration for the following theorem.

Theorem 4.3. *Consider any $i \in V$, $s \in \mathcal{S}$, $t \in \mathbb{N}$, trajectory τ_i and $\sigma_{\partial i}^{t-1}$. For any x_i such that $\mathbb{P}[x_i|s] > 0$, we have*

$$\mathbb{P}\left[\sigma_{\partial i}^{t-1}\middle|s, x_i\right] \mathbf{1}\left(\tau_i^t = g_i^t\left(x_i, \sigma_{\partial i}^{t-1}\right)\right) = \mathbb{Q}\left[\sigma_{\partial i}^{t-1}\middle|\middle|\tau_i, s\right] \mathbf{1}\left(\tau_i^t = g_i^t\left(x_i, \sigma_{\partial i}^{t-1}\right)\right). \tag{4}$$

Proof. We couple the original process, after choosing s, to the modified processes by setting the private signals to be identical in both.

Now, clearly if it so happens that $\tau_i^t = g_i^t\left(x_i, \sigma_{\partial i}^{t-1}\right)$ then the two processes will be identical up to time t. Hence the probabilities of events measurable up to time t will be identical when multiplied by $\mathbf{1}\left(\tau_i^t = g_i^t\left(x_i, \sigma_{\partial i}^{t-1}\right)\right)$, and the theorem follows. □

Using Eqs. (3) and (4), we can write the posterior on s computed by node i at time t, in terms of the probabilities $\mathbb{Q}[\cdot||\cdot]$:

$$\begin{aligned}
\mathbb{P}\left[s|x_i, \sigma_{\partial i}^{t-1}\right] &\propto \mathbb{P}[s]\,\mathbb{P}\left[x_i, \sigma_{\partial i}^{t-1}|s\right] \\
&= \mathbb{P}[s]\,\mathbb{P}[x_i|s]\,\mathbb{P}\left[\sigma_{\partial i}^{t-1}|s, x_i\right] \\
&= \mathbb{P}[s]\,\mathbb{P}[x_i|s] \prod_{j\in\partial i} \mathbb{Q}\left[\sigma_j^{t-1}\middle|\middle|\sigma_i^{t-1}, s\right]
\end{aligned} \tag{5}$$

(Recall that σ_i^{t-1} is a deterministic function of $(x_i, \sigma_{\partial i}^{t-1})$. Also, note that if $\mathbb{P}[x_i|s] = 0$, we simply obtain $\mathbb{P}\left[s|x_i, \sigma_{\partial i}^{t-1}\right] = 0$. Eq. (5) deals with the non-trivial case $\mathbb{P}[x_i|s] > 0$.)

Remark. *A naïve (and incorrect) method to estimate the posterior $\mathbb{P}\left[s|x_i, \sigma_{\partial i}^{t-1}\right]$ would be to treat the trajectories of the neighbors and x_i as being independent conditioned on s, leading to the estimate $\tilde{\mathbb{P}}\left[s|x_i, \sigma_{\partial i}^{t-1}\right] \propto \mathbb{P}[s]\,\mathbb{P}[x_i|s] \prod_{j\in\partial i} \mathbb{P}\left[\sigma_j^{t-1}\middle|s\right]$ for posterior beliefs[12]. Eq. (5) gives us a variation on this estimate that is exact on trees. In other words, it provides the right way to 'combine' information from neighbors to compute the Bayesian posterior on s.*

Recall that

$$\sigma_i(t) = g_{i,t}(x_i, \sigma_{\partial i}^{t-1}) = \arg\max_{s\in\mathcal{S}} \mathbb{P}\left[s|x_i, \sigma_{\partial i}^{t-1}\right]. \tag{6}$$

[12]Thus, the logarithm of this estimated belief is a linear combination of information from neighbors. This has motivated some of the heuristic updates rules studied in the literature [9, 10].

Remark. *Suppose, instead, that the action set $\mathcal{A}$ is distinct from $\mathcal{S}$, and the agents have some common utility function $U : \mathcal{A} \times \mathcal{S} \to \mathbb{R}$. Eq.* (6) *changes to*

$$g_{i,t}(x_i, \sigma_{\partial i}^{t-1}) = \arg\max_{a \in \mathcal{A}} \sum_{s \in \mathcal{S}} \mathbb{P}\left[s | \mathcal{F}_i^t\right] U(a, s)\,, \tag{7}$$

and all results in this section remain unchanged.

We have therefore reduced the problem of calculating $\sigma_i(t)$ to calculating $\mathbb{Q}\left[\cdot||\cdot\right]$. The following theorem is the heart of the dynamic cavity method and allows us to perform this calculation:

Theorem 4.4. *For any $i \in V$, $j \in \partial i$, $s \in \mathcal{S}$, $t \in \mathbb{N}$, τ_i^t and σ_j^t, we have*

$$\mathbb{Q}\left[\sigma_j^t \middle|\middle| \tau_i^t, s\right] = \sum_{\sigma_1^{t-1} \dots \sigma_{d-1}^{t-1}} \sum_{x_j} \mathbb{P}\left[x_j | s\right] \mathbf{1}\left[\sigma_j^t = g_j^t\left(x_j, (\tau_i^{t-1}, \sigma_{\partial j \setminus i}^{t-1})\right)\right] \cdot \prod_{l=1}^{d-1} \mathbb{Q}\left[\sigma_l^{t-1} \middle|\middle| \sigma_j^{t-1}, s\right]\,, \tag{8}$$

where the neighbors of node j are $\partial j = \{i, 1, 2, \dots, d-1\}$.

We mention without proof that the recursion easily generalizes to the case of a *random* tie-breaking rule, provided the rule is common knowledge; it is a matter of replacing the expression $\mathbf{1}\left[\sigma_j^t = \cdots\right]$ with $\mathbb{P}\left[\sigma_j^t = \cdots\right]$, where this probability is over the randomness of the rule. Eq. (5) continues to be valid in this case.

The proof of this theorem is similar to the proof of Lemma 2.1 in [15], where the dynamic cavity method is introduced and applied to a different process.

Proof. In the modified process, the events in the different branches that i sees are independent. We therefore consider $G_{j \to i}$ only, and view it as a tree rooted at j. Also, for convenience we define $\sigma_i^t \equiv \tau_i^t$; note that the random variable σ_i^t does not exist in the modified process, as i's trajectory is fixed to τ_i.

Let $\underline{x}$ be the vector of private signals of j and all the vertices up to a depth t in $G_{j \to i}$ (call this set of vertices $V_{j \to i}^t$). For each $l \in \{1, \dots, d-1\}$, let $\underline{x}_l$ be the vector of private signals of $V_{l \to j}^{t-1}$. Thus, $\underline{x} = (x_j, \underline{x}_1, \underline{x}_2, \dots, \underline{x}_{d-1})$.

The trajectory σ_j^t is a function -deterministic, by our assumption- of $\underline{x}$ and τ_i^t. We shall denote this function by $F_{j \to i}$ and write $\sigma_j^t = F_{j \to i}^t(\underline{x}, \tau_i^t)$. This function is uniquely determined by the update rules $g_l^t\left(x_l, \sigma_{\partial l}^{t-1}\right)$ for $l \in V_{j \to i}^t$.

We have therefore

$$\mathbb{Q}\left[\sigma_j^t = \lambda^t \middle|\middle| \tau_i^t, s\right] = \sum_{\underline{x}} \mathbb{P}\left[\underline{x} | s\right] \mathbf{1}\big(\lambda^t = F_{j \to i}^t(\underline{x}, \tau_i^t)\big)\,. \tag{9}$$

We now analyze each of the terms appearing in this sum. Since the private signals are independent conditioned on s, we have

$$\mathbb{P}\left[\underline{x}\,|s\right] = \mathbb{P}\left[x_j | s\right] \mathbb{P}\left[\underline{x}_1 | s\right] \mathbb{P}\left[\underline{x}_2 | s\right] \dots \mathbb{P}\left[\underline{x}_{d-1} \middle| s\right]\,. \tag{10}$$

The function $F_{j \to i}^t(\cdots)$ can be decomposed as follows:

$$\mathbf{1}\big(\lambda^t = F_{j \to i}^t(\underline{x}, \tau_i^t)\big) = \sum_{\sigma_1^{t-1} \dots \sigma_{d-1}^{t-1}} \mathbf{1}\Big(\lambda^t = g_j^t(x_j, \sigma_{\partial j}^{t-1})\Big) \prod_{l=1}^{d-1} \mathbf{1}\Big(\sigma_l^{t-1} = F_{l \to j}^{t-1}(\underline{x}_l, \lambda^{t-1})\Big)\,. \tag{11}$$

Using Eqs. (10) and (11) in Eq. (9) and separating terms that depend only on $\underline{x}_i$, we get

$$\mathbb{Q}\left[\sigma_j^t = \lambda^t \middle| \middle| \tau_i^t, s\right] = \sum_{\sigma_1^{t-1} \dots \sigma_{d-1}^{t-1}} \sum_{x_j} \mathbb{P}\left[x_j|s\right] \mathbf{1}\left(\lambda^t = g_j^t(x_j, \sigma_{\partial j}^{t-1}\right) \cdot \\ \cdot \prod_{l=1}^{d-1} \sum_{\underline{x}_l} \mathbb{P}\left[\underline{x}_l|s\right] \mathbf{1}\left(\sigma_l^{t-1} = F_{l\to j}^{t-1}(\underline{x}_l, \lambda^{t-1})\right) .$$

The recursion follows by identifying that the product over l has argument $\mathbb{Q}\left[\sigma_l^{t-1} \middle| \middle| \sigma_j^{t-1}, s\right]$. □

4.2 The Agents' Calculations

We now describe how to perform the agents' calculations. At time $t = 0$ these calculations are trivial. Assume then that up to time t each agent has calculated the following quantities:

1. $\mathbb{Q}\left[\sigma_j^{t-1} \middle| \middle| \tau_i^{t-1}, s\right]$, for all $s \in \mathcal{S}$, for all $i, j \in V$ such that $j \in \partial i$, and for all τ_i^{t-1} and σ_j^{t-1}.
2. $g_i^t(x_i, \sigma_{\partial i}^{t-1})$ for all i, x_i and $\sigma_{\partial i}^{t-1}$.

Note that these can be calculated without making any observations – only knowledge of the graph G, $\mathbb{P}\left[s\right]$ and $\mathbb{P}\left[\underline{x}|s\right]$ is needed.

At time $t+1$ each agent makes the following calculations:

1. $\mathbb{Q}\left[\sigma_j^t \middle| \middle| \tau_i^t, s\right]$ for all $s, i, j, \sigma_j^t, \tau_i^t$. These can be calculated using Eq. (8), given the quantities from the previous iteration.
2. $g_i^{t+1}(x_i, \sigma_{\partial i}^t)$ for all i, x_i and $\sigma_{\partial i}^t$. These can be calculated using Eqs. (5) and (6) and the the newly calculated $\mathbb{Q}\left[\sigma_j^t \middle| \middle| \tau_i^t, s\right]$.

Since agent j calculates g_i^{t+1} for all i, then she, in particular, calculates g_j^{t+1}. This allows her to choose the (myopic) Bayes optimal action in rounds up to $t+1$, based on her neighbors' past actions. A simple calculation yields the following lemma.

Lemma 4.5. *In a tree graph G with maximum degree d, the agents can calculate their actions up to time t with computational effort $n2^{O(td)}$.*

In fact, each agent does not need to perform calculations for the entire graph. It suffices for node i to calculate quantities up to time t' for nodes at distance $t - t'$ from node i (there are at most $(d-1)^{t-t'}$ such nodes). A short calculation yields an improved bound on computational effort, stated in Theorem 3.2.

Proof of Theorem 3.2. Consider an agent j, who wants to determine her own decision function up to round t, i.e., she wants to determine $g_j^t(\cdot, \cdot)$. The computation is performed in t steps, that we number 0, 1, ..., $t-1$. Step 0 involves the following: (i) Evaluate $g_i^0(x_i) = \arg\max_{s\in\mathcal{S}} \mathbb{P}\left[s|x_i\right]$ for all i at a distance at most t from j. (ii) Evaluate $\mathbb{Q}\left[\sigma_i^0 \middle| \middle| \tau_k^0, s\right]$ for all k at distance at most $t-1$ from j, for all $i \in \partial k$, and for all σ_i^0, τ_k^0, s, using Eq. (8).

For any $1 \le t' \le t-1$, step $t - t'$ proceeds as follows. Consider any agent i at distance at most $t' \ge 1$ from j. Suppose that we have already computed $\mathbb{Q}\left[\sigma_l^{t-t'-1} \middle| \middle| \tau_i^{t-t'-1}, s\right]$ for all such i, for all $l \in \partial i$, and for all possible $\sigma_l^{t-t'-1}, \tau_i^{t-t'-1}, s$. Then we can use Eqs. (5) and (6) to

compute $g_i^{t-t'}(x_i, \sigma_{\partial i}^{t-t'-1})$ for all possible x_i, $\sigma_{\partial i}^{t-t'-1}$. Using these values, for any k at a distance $t'-1$ from j, we can compute $\mathbb{Q}\left[\sigma_i^{t-t'} \middle| \tau_k^{t-t'}, s\right]$ for all $i \in \partial k$, for all $\sigma_i^{t-t'}$,$\tau_k^{t-t'}$, s, using Eq. (8). The computational effort involved is bounded by $C(d-1)^{t'}|\mathcal{S}|^{d(t-t')+1}|\mathcal{X}|$ for the computation of $g_i^{t-t'}(\,\cdot\,,\,\cdot\,)$'s and bounded by $C(d-1)^{t'}|\mathcal{S}|^{(d+1)(t-t'+1)}|\mathcal{X}|$ for the computation of $\mathbb{Q}\left[\sigma_i^{t-t'} \middle| \tau_k^{t-t'}, s\right]$'s. Here d is maximum degree, and $C = C(d) < \infty$ is a constant. Thus, step $t-t'$ requires effort bounded by $2^{C'td}$ for some $C' = C'(d, |\mathcal{S}|, |\mathcal{X}|) < \infty$. This bound also holds for step 0. Thus, the overall computational effort is bounded by $t2^{C'td} = 2^{O(td)}$.

□

5 Majority dynamics: Proof of Theorem 3.5

In this section we study a very simple update rule, 'majority dynamics'. We prove the following theorem.

Theorem. *(Theorem 3.5) Consider binary $s \sim$ Bernoulli(1/2), and binary initial votes $\widehat{\sigma}_i(0)$ that are independent identically distributed given s, with $\mathbb{P}[\widehat{\sigma}_i(0) \neq s] = 1-\delta$ for some $\delta \in (0, 1/2)$. Let i be any node in an (undirected) d regular tree for $d \geq 5$. Then, under the majority dynamics,*

$$\mathbb{P}[\widehat{\sigma}_i(t) \neq s] = \exp\Big[-\Omega\Big(\big(\tfrac{1}{2}(d-2)\big)^t\Big)\Big].$$

when $\delta < (2e(d-1)/(d-2))^{-\frac{d-2}{d-4}}$.

We use $\widehat{\sigma}_i(t) \in \{-1, +1\}$ to denote votes under the majority dynamics.

Definition 5.1. *Under majority dynamics, each agent $i \in V$ chooses her vote in round $t+1$ according to the majority of the votes of her neighbors in round t, i.e.,*

$$\widehat{\sigma}_i(t+1) = \text{sign}\left(\sum_{j \in \partial i} \widehat{\sigma}_j(t)\right)$$

Ties are broken by flipping an unbiased coin.

Let $s \in \{-1, +1\}$ be drawn from a uniform prior and nodes receive 'private signals' $\widehat{\sigma}_i(0)$ that are correct with probability $1-\delta$, and independent conditioned on s. We consider an undirected d regular tree. The analysis is complicated by dependencies which have to be carefully handled. Our analytical approach here is again closely related to the dynamic cavity method.

Lemma 5.2. *Consider the setting in Theorem 3.5. Let i and j be adjacent nodes in the tree. Then for all $(\widehat{\sigma}_i^{t-1}, \widehat{\sigma}_j^{t-1}) \in \{-1, +1\}^{2t}$*

$$\mathbb{P}\left[\widehat{\sigma}_i(t) = -1 | \widehat{\sigma}_i^{t-1}, \widehat{\sigma}_j^{t-1}, s = +1\right] \leq \delta_t \tag{12}$$

where δ_t is defined recursively by $\delta_0 \equiv \delta$, and

$$\delta_t \equiv \mathbb{P}[\text{Binomial}(d-1, \delta_{t-1}) \geq d/2 - 1] \tag{13}$$

Proof. We proceed by induction. Clearly Eq. (12) holds for $t = 0$. Suppose Eq. (12) holds for some t. We want to show

$$\mathbb{P}\left[\widehat{\sigma}_i(t+1) = -1 \,|\, \widehat{\sigma}_i^t, \widehat{\sigma}_j^t, s = +1\right] \le \delta_{t+1}\,, \tag{14}$$

for all $(\widehat{\sigma}_i^t, \widehat{\sigma}_j^t) \in \{-1, +1\}^{2(t+1)}$.

Let $l_1, l_2, \ldots, l_{d-1}$ be the other neighbors of node i (besides j). We will show that, in fact,

$$\mathbb{P}\left[\widehat{\sigma}_i(t+1) = -1 \,|\, \widehat{\sigma}_i^t, \widehat{\sigma}_j^t, \widehat{\sigma}_{l_1}^{t-1}, \ldots, \widehat{\sigma}_{l_{d-1}}^{t-1}, s = +1\right] \le \delta_{t+1}\,, \tag{15}$$

for all possible $\xi \equiv (\widehat{\sigma}_i^t, \widehat{\sigma}_j^t, \widehat{\sigma}_{l_1}^{t-1}, \widehat{\sigma}_{l_2}^{t-1}, \ldots, \widehat{\sigma}_{l_{d-1}}^{t-1})$.

We reason as follows. Fix the state of the world s and the trajectories $\widehat{\sigma}_i^t$ and $\widehat{\sigma}_j^t$. Now this induces correlations between the trajectories of the neighbors $l_1, \ldots, l_{d-1}$, caused by the requirement of consistency with the majority rule at node i, but *only up to time* $t-1$. If we further fix $\widehat{\sigma}_{l_m}^{t-1}$, then $\widehat{\sigma}_{l_m}(t)$ (and $\widehat{\sigma}_{l_m}$ at all future times) is conditionally independent of $\left(\widehat{\sigma}_{l_{m'}}^t\right)_{m' \neq m}$. Thus, we have [13]

$$\mathbb{P}\left[\widehat{\sigma}_{l_m}(t) = -1 | \, \xi, s = +1\right] = \mathbb{P}\left[\widehat{\sigma}_{l_m}(t) = -1 | \, \widehat{\sigma}_{l_m}^{t-1}, \widehat{\sigma}_i^{t-1}, s = +1\right]\,,$$

and therefore, using the induction hypothesis

$$\mathbb{P}\left[\widehat{\sigma}_{l_m}(t) = -1 | \, \xi, s = +1\right] \le \delta_t \tag{16}$$

for all $m \in \{1, 2, \ldots, d-1\}$. Also, the actions $\widehat{\sigma}_{l_1}(t), \ldots, \widehat{\sigma}_{l_{d-1}}(t)$ are conditionally independent of each other given $\xi, s = +1$. We have

$$\widehat{\sigma}_i(t+1) = \mathrm{sgn}(\widehat{\sigma}_j(t) + \widehat{\sigma}_{l_1}(t) + \ldots + \widehat{\sigma}_{l_{d-1}}(t))\,,$$

with $\mathrm{sgn}(0)$ being assigned value -1 or $+1$ with equal probability. This yields

$$\mathbb{P}\left[\widehat{\sigma}_i(t+1) = -1 \,|\, \xi, s = +1\right] \le \mathbb{P}\left[\mathrm{Binomial}(d-1, \delta_t) \ge d/2 - 1\right]$$

from Eq. (16) and conditional independence of $\widehat{\sigma}_{l_1}(t), \ldots, \widehat{\sigma}_{l_{d-1}}(t)$. Thus, we obtain obtain Eq. (15). Eq. (14) follows by summing over $\widehat{\sigma}_{l_1}^{t-1}, \widehat{\sigma}_{l_2}^{t-1}, \ldots, \widehat{\sigma}_{l_{d-1}}^{t-1}$. □

Proof of Theorem 3.5. By applying the multiplicative version of the Chernoff bound[14] to Eq. (13) we have that

$$\delta_{t+1} \le e^{(d-2)/2 - (d-1)\delta_t} (2\delta_t (d-1)/(d-2))^{(d-2)/2}$$

Dropping the term $e^{-(d-1)\delta_t}$, we obtain

$$\delta_{t+1} \le (2e\delta_t (d-1)/(d-2))^{\frac{1}{2}(d-2)}. \tag{17}$$

This is a first order non-homogeneous linear recursion in $\log \delta_t$. If it were an equality it would yield

$$\begin{aligned}\log \delta_t = &\left(\log \delta + \frac{d-2}{d-4} \log[2e(d-1)/(d-2)]\right) \left[\tfrac{1}{2}(d-2)\right]^t \\ &- \frac{d-2}{d-4} \log[2e(d-1)/(d-2)]\,,\end{aligned}$$

[13] A alternate argument can be constructed using the modified process with a zombie agent, mirroring the reasoning used in the proof of Theorem 4.4.

[14] $\mathbb{P}\left[X \ge (1+\eta)\mathbb{E}\left[X\right]\right] \le \left(\frac{\exp \eta}{(1+\eta)^{1+\eta}}\right)^{\mathbb{E}[X]}$. We substitute $\mathbb{E}\left[X\right] = \delta_t(d-1)$ and $1 + \eta = (d/2 - 1)/[\delta_t(d-1)]$.

Round	Bayesian	Majority
0	0.15	0.15
1	$6.075 \cdot 10^{-2}$	$6.075 \cdot 10^{-2}$
2	$1.57158 \cdot 10^{-2}$	$2.95136 \cdot 10^{-2}$
3	$2.99170 \cdot 10^{-3}$	$1.59849 \cdot 10^{-2}$
4	$3.39853 \cdot 10^{-4}$	$9.15458 \cdot 10^{-3}$
5	$2.72958 \cdot 10^{-5}$	$5.46501 \cdot 10^{-3}$
6	$2.21981 \cdot 10^{-6}$	$3.35117 \cdot 10^{-3}$

Table 2: Error probabilities for Bayesian agents on a regular tree of degree $d = 3$, with $\mathbb{P}[x_i \neq s] = 0.15$.

and so

$$-\log \delta_t \in \Omega\left(\left(\tfrac{1}{2}(d-2)\right)^t\right), \tag{18}$$

as long as

$$-\log \delta < \frac{d-2}{d-4} \log[2e(d-1)/(d-2)]\,.$$

□

Theorem 3.5 is non-trivial for $d \geq 5$. The upper limit of the 'noise' δ for which it establishes rapid convergence approaches $(2e)^{-1}$ as d grows large. (see also the discussion below for large d)

5.0.1 Convergence for large d

We present now a short informal discussion on the limit $d \to \infty$. We can, in fact, use Lemma 5.2 to show convergence is doubly exponential for $\delta < 1/2 - c/d$ for some $c < \infty$ that does not depend on d.

Here is a sketch of the argument. Suppose $\delta = 1/2 - c_1/d$. Then, for all $d > d_1$ where $d_1 < \infty$, there exists $c_2 < \infty$ such that $\mathbb{P}[\text{Binomial}(d-1, \delta) \geq d/2 - 1] < 1/2 - c_2/\sqrt{d}$. This can be seen, for instance, by coupling with the Binomial$(d-1, 1/2)$ process and using an appropriate local central limit theorem (e.g., see [15, Theorem 4.4]). Thus, $\delta_1 < 1/2 - c_2/\sqrt{d}$. Further, c_2 can be made arbitrarily large by choosing large enough c_1. Next, with a simple application of Azuma's inequality, we arrive at $\delta_2 < c_3$ (where $c_3 \to 0$ as $c_2 \to \infty$). Now, for small enough c_3, we use the Chernoff bound analysis in the proof of Theorem 3.5 and obtain doubly exponential convergence.

6 Further numerical results and discussion on Conjecture 3.4

Table 2, together with Table 1 above, contrast the error probabilities of Bayesian updates with those of majority updates. All cases exhibit lower error probabilities (in the weak sense) for the Bayesian update, consistent with Conjecture 3.4. Table 3 contains the data plotted in Figure 1. Also for these parameters, we found that the Bayesian updates showed lower error probabilities than the majority updates (compare with Table 4).

The running time to generate these tables was less than a minute on a standard desktop machine. We did not proceed with more rounds because of numerical instability issues which begin to appear as error probabilities decrease.

We now discuss briefly the difficulties in proving Conjecture 3.4. Order the possible private signals by the implied likelihood ratio of s, with higher x_j corresponding to $s = +1$ being more likely. We say a learning rule with successive rounds of 'voting' is *monotonic* if the following

Round	$d = 3$	$d = 5$	$d = 7$
0	0.30	0.30	0.30
1	0.216	0.16308	0.126036
2	0.134038	$5.07053 \cdot 10^{-2}$	$1.1966 \cdot 10^{-2}$
3	$7.77755 \cdot 10^{-2}$	$4.06495 \cdot 10^{-3}$	$3.67884 \cdot 10^{-6}$
4	$3.79502 \cdot 10^{-2}$	$1.61786 \cdot 10^{-5}$	
5	$1.71209 \cdot 10^{-2}$		
6	$5.73294 \cdot 10^{-3}$		
7	$1.59587 \cdot 10^{-3}$		

Table 3: Error probabilities for Bayesian agents with $\mathbb{P}[x_i \neq s] = 0.3$, for regular trees of different degrees d. This data is displayed in Figure 1 in Section 3.2.

Round	$d = 3$	$d = 5$	$d = 7$
0	0.30	0.30	0.30
1	0.216	0.16308	0.126036
2	0.170489	0.0733673	0.0232861
3	0.146010	0.0215952	$2.99165 \cdot 10^{-4}$
4	0.130070	$2.61093 \cdot 10^{-3}$	
5	0.119647		
6	0.112267		
7	0.107006		

Table 4: Error probabilities for agents using majority updates with $\mathbb{P}[x_i \neq s] = 0.3$, for regular trees of different degrees d.

occurs: If some $\underline{x}$ leads to $\sigma_i(t) = 1$, then increasing x_j in $\underline{x}$ for some $j \in V$ leaves $\sigma_i(t)$ unchanged. One might expect most reasonable learning rules, including iterative Bayesian learning, to satisfy *monotonicity*. For instance, there is a simple proof that the majority rule is monotonic [15]. However, it turns out that iterative Bayesian learning is not always monotonic[15]! It is not very surprising, then, that it is hard to prove convergence of Bayesian learning to the 'right' answer, even in simple settings. Controlling the rate of convergence, as in Conjecture 3.4, is even harder.

Despite non-monotonicity, it is tempting to hope for a direct proof of Conjecture 3.4, by showing inductively (in time) that iterative Bayesian learning is always at least as good majority dynamics. The difficulty that arises here is that though iterative Bayesian learning minimizes the error probability at a node, given the available information, this is not the case if we condition on the state of the world. After conditioning on the state of the world, iterative Bayesian learning does better than majority dynamics on some nodes, and worse on others. It is very hard to control the difference between the two processes beyond a small number of iterations, making a direct proof of Conjecture 3.4 difficult.

7 Discussion

We presented a new algorithmic approach that questions the belief that fully Bayesian computations for agents interacting on a social network are computationally intractable. The chief drawback is that our approach does not seem amenable to graphs with multiple short loops, though many real networks possess this feature. A significant open question suggested by our results is: What is the

[15]Elchanan Mossel and Omer Tamuz, private communication.

'computational boundary' between networks where exact Bayesian calculations can be efficiently performed, and networks where this is not possible?

7.1 Relation to Aaronson's work on the complexity of agreement

The work most closely related in spirit to the present one is that of Aaronson on the complexity of agreement [1]. In that work, as in this one, the author started out aiming to establish complexity theoretic 'hardness': "communication complexity might provide a fundamental reason for why ... people could agree to disagree ... this was our conjecture when we began studying the topic", but instead discovered an efficient procedure to achieve the objective in question under some conditions. We briefly describe Aaronson's work next and compare it to our own.

Aaronson investigated the question of whether Aumann's classic theorem on agreement [3] is supported by an efficient procedure by which agents with a common prior can reach agreement. In that setting, the 'communication protocol' itself is unspecified, and the objective is to formulate an efficient communication protocol, along with an efficient computational procedure to implement this protocol, so as to facilitate agreement between agents. The scaling variables chosen are the number of bits n of private information, and the inverse 'error probability' $1/\epsilon$. Aaronson shows that agreement can be achieved after a 'conversation' of reasonable length (that does not depend on n) in the case of two interacting agents, and also in the case of more than two agents on a strongly connected network. Further, he shows that for two agents, the computational effort required to adequately 'simulate' this conversation is again independent of n ([1] does not establish a computational bound for networks of three or more agents). Given a desired error probability ϵ, the conversation length required grows as $\text{poly}(1/\epsilon)$, whereas the bound on computational effort grows as $\exp(\text{poly}(1/\epsilon))$.

There are several evident differences with the current work. First, the 'communication protocol' is specified implicitly by the model itself in our problem. Thus, the single objective for us is to minimize computational effort. Second, our scaling regime is very different, in that we let the number of nodes n grow large (whereas Aaronson focusses on the two agent case), but demand that private signals of agents belong to a finite set. In other words, we study the effect of large network size on computational difficulty, whereas Aaronson focusses on the effect of a large amount of private information[16]. In terms of dependence of computational effort on 'error probability', our bound of $\text{polylog}(1/\epsilon)$ on computational effort (assuming Conjecture 3.4) is doubly exponentially smaller than the bound of $\exp(\text{poly}(1/\epsilon))$ obtained by Aaronson as evidence of 'efficient computability'[17].

In the spirit of Aaronson's approach [1] to simulating a conversation between Bayesian agents using limited computations, it may be possible to develop a Monte Carlo sampling based version of the naïve dynamic program (Section 3.2.1) that is much more efficient. The chief additional difficulty would be in handling the effects of network structure, since Aaronson only deals with the case of two agents. This is an interesting direction of future study.

Acknowledgments. We would like to thank Andrea Montanari, Elchanan Mossel and Allan Sly for valuable discussions.

References

[1] S. Aaronson, *The complexity of agreement*, in Proceedings of the thirty-seventh annual ACM symposium on Theory of computing, STOC '05, ACM, 2005, pp. 634–643.

[16] Interestingly the bounds obtained are independent of n in both works, for n large.

[17] Of course, a direct comparison is somewhat unfair since the problems being addressed are quite different.

[2] D. Acemoglu, M. A. Dahleh, I. Lobel, and A. Ozdaglar, *Bayesian learning in social networks*, The Review of Economic Studies, (2011).

[3] R. Aumann, *Agreeing to disagree*, The Annals of Statistics, 4 (1976), pp. 1236–1239.

[4] V. Bala and S. Goyal, *Learning from neighbours*, Review of Economic Studies, 65 (1998), pp. 595–621.

[5] O. Bandiera and I. Rasul, *Social networks and technology adoption in northern mozambique**, The Economic Journal, 116 (2006), pp. 869–902.

[6] A. V. Banerjee, *A simple model of herd behavior*, The Quarterly Journal of Economics, 107 (1992), pp. 797–817.

[7] D. H. Bikhchandani, S. and I. Welch, *A theory of fads, fashion, custom, and cultural change as informational cascade*, Journal of Political Economy, 100 (1992), pp. 992–1026.

[8] T. Conley and C. Udry, *Social learning through networks: The adoption of new agricultural technologies in ghana*, American Journal of Agricultural Economics, 83 (2001), pp. 668–673.

[9] M. H. DeGroot, *Reaching a consensus*, Journal of the American Statistical Association, 69 (1974), pp. 118–121.

[10] P. M. DeMarzo, D. Vayanos, and J. Zwiebel, *Persuasion bias, social influence, and unidimensional opinions.*, Quarterly Journal of Economics, 118 (2003), pp. 909 – 968.

[11] G. Ellison and D. Fudenberg, *Rules of thumb for social learning*, Journal of Political Economy, 110 (1995), pp. 93–126.

[12] D. Gale and S. Kariv, *Bayesian learning in social networks*, Games and Economic Behavior, 45 (2003), pp. 329–346.

[13] S. Goyal, *Connections: An Introduction to the Economics of Networks*, Princeton University Press, 2007.

[14] S. Istrail, *Statistical mechanics, three-dimensionality and np-completeness: I. universality of intracatability for the partition function of the ising model across non-planar surfaces (extended abstract)*, in Proceedings of the thirty-second annual ACM symposium on Theory of computing, STOC '00, New York, NY, USA, 2000, ACM, pp. 87–96.

[15] Y. Kanoria and A. Montanari, *Majority dynamics on trees and the dynamic cavity method*. Preprint. To appear in Annals of Applied Probability., 2011.

[16] Y. Kanoria and A. Montanari, *Subexponential convergence for information aggregation on regular trees*. 2011.

[17] H. Kohler, *Learning in social networks and contraceptive choice*, Demography, 34 (1997), pp. 369–383.

[18] L. Ménager, *Consensus, communication and knowledge: an extension with bayesian agents*, Mathematical Social Sciences, 51 (2006), pp. 274–279.

[19] M. Mueller-Frank, *A Comment on 'Bayesian Learning in Social Networks'*, SSRN eLibrary, (2011).

[20] ——, *Mixed Communication Structures: Opinion Formation in Social Networks with Bayesian and Non-Bayesian Agents*, SSRN eLibrary, (2011).

[21] R. Parikh and P. Krasucki, *Communication, consensus, and knowledge* 1*, Journal of Economic Theory, 52 (1990), pp. 178–189.

[22] M. Raynal, *Fault-tolerant agreement in synchronous message-passing systems*, Synthesis Lectures on Distributed Computing Theory, 1 (2010), pp. 1–189.

[23] D. Rosenberg, E. Solan, and N. Vieille, *Informational externalities and emergence of consensus*, Games and Economic Behavior, 66 (2009), pp. 979 – 994. Special Section In Honor of David Gale.

[24] J. Sebenius and J. Geanakoplos, *Don't bet on it: Contingent agreements with asymmetric information*, Journal of the American Statistical Association, 78 (1983), pp. 424–426.

[25] L. Smith and P. Sørensen, *Pathological outcomes of observational learning*, Econometrica, 68 (2000), pp. 371–398.