Tractable Bayesian Social Learning on Trees

Yashodhan Kanoria
Microsoft Research New England and
Department of Electrical Engineering
Stanford University
Email: ykanoria@stanford.edu

Omer Tamuz
Microsoft Research New England
and
Weizmann Institute
Email: omer.tamuz@weizmann.ac.il

***Abstract*—We study a model of Bayesian agents in social networks who learn from the actions of their neighbors. Agents attempt to iteratively estimate an unknown 'state of the world' s from initial private signals, and the past actions of their neighbors in the 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 a naive dynamic program.**

We use this algorithm to perform the first numerical simulations of Bayesian agents on networks with hundreds of nodes, and observe rapid learning of s in some settings.

I. Introduction

The importance of social learning in networks has been demonstrated in a wide variety of settings, such as the adoption of agricultural technology in Ghana [1], and choice of contraceptives by European women [2].

Accordingly, developing and understanding models of social learning has been a goal of theoretical economics for the past few decades (cf., [3], [4] and references therein). 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. The premise in such models is that "actions speak louder than words" – agents learn by observing each others' actions, and not by communicating directly.

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 [5]. 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. Gale and Kariv [5] showed that, in this model, agents converge to the same action under some conditions. Related work [6] sheds more light on the phenomenon of agreement on actions and the conditions in which it arises.

We are interested in the following questions in the context of this model, which have not been previously addressed:

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

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 (cf. Proposition III.1) 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 $t2^{O(\min(n,d^t))}$. This grows very rapidly, restricting previous simulations to networks of three nodes [5]

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 [5], who simulated the process for three agents, to much larger networks[1]. 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

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

previously unexplored settings.

We conjecture that on regular trees, 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. 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 trees.

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 highly hierarchical organizations, as well as some physical communication networks. 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.

See the full version of the paper [7] for a more detailed discussion, literature survey and proof details.

Technical contributions. A key technique used in this paper is the dynamic cavity method, introduced by Kanoria and Montanari [8] in their study of 'majority updates' on trees. This technique is a dynamical version of the cavity method of statistical physics. Our algorithmic and analytical approach leveraging the dynamic cavity method should be applicable to a range of models involving iterative updates on trees.

Our second main technical contribution is our proof, using a dynamic cavity type approach, of doubly exponentially fast convergence of majority dynamics on regular trees. This result should be of independent interest.

II. Model

The model we consider is a simplified version of the model of social learning introduced by Gale and Kariv [5].

Consider a directed graph $G=(V,E)$, representing a network of $n=|V|$ 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'. 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 $\sigma_i^t=(\sigma_i(0),\sigma_i(1),\ldots,\sigma_i(t))$ 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., $\partial i=\{j:(i,j)\in E\}$. 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, $g_i^t=(g_{i,0},g_{i,1},\ldots,g_{i,t})$, so that

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

The full version [7] motivates our model in the context of rational agents, and also presents a detailed comparison with the model of Gale and Kariv [5].

III. Main results

A. Efficient computation

A fairly straightforward dynamic programming algorithm can be used to compute the actions chosen by agents in our model. The proposition below states the computational complexity of this algorithm.

Proposition III.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 [7] 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.

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 (i.e., a graph with no loops). We show that computational effort exponentially lower than that of the naive DP suffices in this case.

Theorem III.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 [8], previously used only in analytical contexts. Section IV contains a description of the algorithm and analysis leading to Theorem III.2.

We would like to thank our anonymous referee for pointing out that it may also be possible to prove Theorem III.2 using Bayesian Networks (BN). The proof would involve constructing the appropriate BN and showing that its tree-width is $\min(n, td)$.

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 [5]. If ϵ is the desired probability of error, then assuming a reasonable conjecture (Conjecture III.4), we show that computational effort only $\text{polylog}(1/\epsilon)$ is required on trees. Having obtained an approximately correct estimate, the agents would have little incentive to continue updating their beliefs. (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.

B. 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.

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

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

where $b > 1$ is some constant.

The following is an immediate corollary of Theorem III.2.

Corollary III.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* $\mathbb{P}[\sigma_i(t) \neq s] \leq \epsilon$ *for all i in V.*

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

We state below, and provide numerical evidence for, a conjecture that implies doubly exponential convergence of iterative Bayesian learning.

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'[8]. Under this rule, the agents adopt the action taken by the majority of their neighbors in the previous iteration. 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 III.4. *Consider binary* $s \sim \text{Bernoulli}(1/2)$*, 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 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*

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

Using a dynamic cavity approach, we show doubly exponential convergence for majority dynamics on regular trees (the full version [7] contains a proof):

Theorem III.5. *Consider binary* $s \sim \text{Bernoulli}(1/2)$*, 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 III.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 III.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 III.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 [9], 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. Our numerical results indicate very rapid decay of error probability on regular trees (cf. questions (I) and (II) in Section I).

Figure 1 plots decay of error probabilities in regular trees for iterative Bayesian learning with $\mathbb{P}[x_i \neq s] = 0.3$ 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. See the full version [7] for more numerical results.

IV. The Dynamic Cavity Algorithm on Trees

In this section we develop the dynamic cavity algorithm leading to Theorem III.2. We present the core construction and key technical lemmas in Section IV-A. In Section IV-B, we show how this leads to an efficient algorithm for the Bayesian computations on tree graphs, and prove Theorem III.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.

A. 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.

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 IV.1. *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 IV.2. *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*

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

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

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

Recall that

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

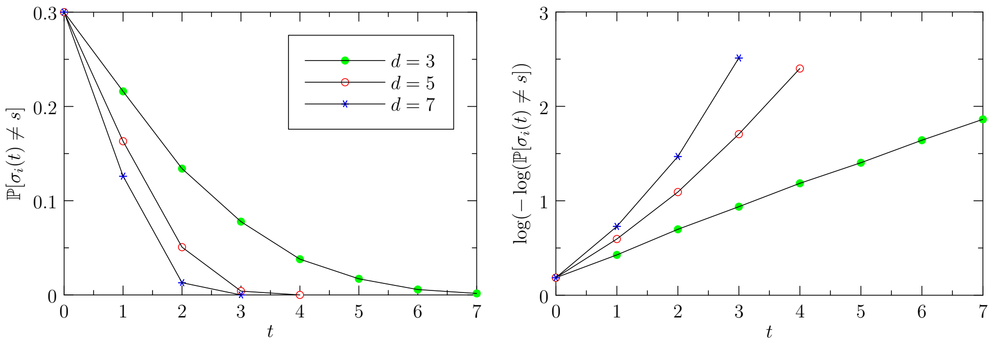

Fig. 1. Error probability decay on regular trees for iterative Bayesian learning, with $\mathbb{P}[x_i \neq s] = 0.3$.

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 IV.3. *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*

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

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

The proof of this theorem is similar to the proof of Lemma 2.1 in [8], where the dynamic cavity method is introduced and applied to a different process.

B. 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 IV.4. *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 III.2. The proof of Theorem III.2 is relatively straightforward and is provided in the full version [7] of this paper.

Acknowledgments. We would like to thank A. Montanari, E. Mossel and A. Sly for valuable discussions, and the anonymous referees for their valuable comments.

Y. Kanoria was supported by a 3Com Corporation Stanford Graduate Fellowship. Omer Tamuz is supported by a Google Europe Fellowship in Social Computing.

References

[1] T. Conley and C. Udry, "Social learning through networks: The adoption of new agricultural technologies in ghana," *American Journal of Agricultural Economics*, vol. 83, no. 3, 2001.

[2] H. Kohler, "Learning in social networks and contraceptive choice," *Demography*, vol. 34, no. 3, pp. 369–383, 1997.

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

[4] D. Acemoglu, M. A. Dahleh, I. Lobel, and A. Ozdaglar, "Bayesian learning in social networks," *The Review of Economic Studies*, vol. 78, no. 4, 2011.

[5] D. Gale and S. Kariv, "Bayesian learning in social networks," *Games and Economic Behavior*, vol. 45, no. 2, pp. 329–346, 2003.

[6] D. Rosenberg, E. Solan, and N. Vieille, "Informational externalities and emergence of consensus," *Games and Economic Behavior*, vol. 66, no. 2, pp. 979 – 994, 2009.

[7] Y. Kanoria and O. Tamuz, "Tractable bayesian social learning," 2011, preprint at http://arxiv.org/abs/1102.1398.

[8] Y. Kanoria and A. Montanari, "Majority dynamics on trees and the dynamic cavity method," *Annals of Applied Probability.*, vol. 21, no. 5, 2011.

[9] ——, "Subexponential convergence for information aggregation on regular trees," in *CDC-ECC*, 2011.