Trust Evaluation in Anarchy: A Case Study on Autonomous Networks

Tao Jiang
Institute for Systems Research and
Department of Electrical and Computer Engineering
University of Maryland, College Park 20742
Email: tjiang@isr.umd.edu

John S. Baras
Institute for Systems Research and
Department of Electrical and Computer Engineering
University of Maryland, College Park 20742
Email: baras@isr.umd.edu

***Abstract*— With the explosive growth of network techniques, in particular wireless communications, the traditional centralized, fixed networks can no longer satisfy the enormous demands on network connectivity, data storage and information exchanges. New types of networks, such as pervasive computing networks, mobile ad hoc networks and P2P networks, emerged in recent years in order to provide solutions for the increasing requirements on networked services. All those networks are *autonomous networks*, because they are distributed and self-organized. As a case study, we employ a specific application – distributed trust management – to understand and analyze the behavior and properties of these "anarchical" autonomous networks. We propose a statistical trust evaluation rule, prove its convergence and investigate its characteristics when the system is at the steady state. Our investigation results in several conclusions for the design of trust evaluation rules, some of which are unexpected if we do not have the stationary distribution at hand. Our study shows the importance and necessity of applying theoretical analyses to understand the complex characteristics of distributed, self-organized, autonomous networks.**

I. Introduction

With the explosive growth of network techniques in the last decade, connecting to the world from any place, at any time and for any body is no longer just a dream. In the meanwhile, the fast proliferation of networked devices and applications, such as sensor networks and pervasive computing, integrates information technology into our environments. These dramatic changes create unique challenges for network management and control. Innovative solutions are required for managing network mobility and dynamics, astronomical number of data and enormous information exchanges.

The traditional centralized server-based management can no longer satisfy the requirements of next generation networks, so people started to propose new concepts of network infrastructure and management. For instance, mobile ad hoc networks (MANETs) [1] aim to provide wireless network services without relying on any infrastructure. The wireless mesh network, which has been implemented by a number of wireless network groups [2], [3], is essentially a MANET over a 802.11 wireless LAN, which can be set up with almost "zero-cost" in highly mobile environments. Another example is peer-to-peer (P2P) networks [4], [5], where a large number of data are shared among millions of network users. All the aforementioned new types of networks share a common characteristic: they are distributed and self-organized, thus they are sometimes called *autonomous/autonomic networks* [6] in the literature. In this paper, our focus is on the fundamental principles and properties of these networks, rather than narrowing on a particular network prototype.

An autonomous network is one that is decentralized, self-configuring and self-protecting. Such a network requires minimal administration, which mostly only involves policy-level management. Entities in autonomous networks all participate in network control through individual interactions. To achieve desired network management goals under such "anarchy" is not an easy job. A small misbehavior by an individual might lead to a network-wise "avalanche". The goal of our paper is to understand and analyze the behavior and properties of these "anarchical" autonomous networks.

Autonomous systems have been studied in various scientific fields: in biological systems, swarms of bacteria, insects and animals yield sophisticated collective behaviors based on simple individual interactions; in physics, a group of particles interacting with their neighbors to form a magnet; even in human society, economists have modeled human individual interactions using iterated games, etc. Our work is inspired from those studies, in particular the Ising model and spin glasses model in statistical physics, which will be discussed in more detail in Sec. V. The Ising model was originally developed to explain the physical alignment of particles within a magnetically-charged material, such as iron. Recently, the Ising model has been applied to a diverse range of applications where individuals interact with others in their vicinity, such as the associative memory model in neural networks (e.g. Hopfield networks) and cooperation in social networks [7].

As a case study, we employ a specific application in autonomous networks – distributed trust management – for our study. Trust is important and critical for network security. It integrates with several components of network management, such as risk management, access control and authentication. Trust management is to collect, analyze and present trust-related evidence and to make assessments and decisions regarding trust relationships between entities in a network [8]. In this paper, we will study the evaluation of entity trust based on trust information provided by its neighbors in the network. In Sec. II, we will specify the properties of trust management

1-4244-0222-0/06/$20.00 (c)2006 IEEE

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the Proceedings IEEE Infocom.

– in particular for autonomous networks.

Our focus for distributed trust management is on its theoretical analysis rather than providing a strict definition, or a system model for trust management. By defining a trust evaluation rule based on local voting, we study how the trustworthiness of the whole network evolves with time. The trust evaluation is identified as an iterated stochastic process. The convergence of this process and the stationary distribution at the steady state is provided. We further investigate the resulting trust values at the steady state. Our investigation gives several important conclusions, some of which are even surprising, such as the choice of threshold and phase transition phenomena. Those results provide a way to properly design a feasible evaluation rule. Furthermore, we extend the local voting rule to a general local evaluation rule and discuss their relation. The fast emergence of autonomous networks has led to a tremendous amount of related publications, while few have solid theoretical analyses. This paper is the starting point of our effort to theoretically understand the complex characteristics of distributed, self-organized, autonomous networks.

This paper is organized as the follows. Section II discusses the unique properties of trust management in autonomous networks as opposed to traditional Internet. A general trust evaluation rule based on local voting is provided in Sec. III. This general rule is further specified as an iterated stochastic rule in Sec. IV, and the Markov chain interpretation and convergence of the stochastic rule is presented. Based on the derived stationary distribution in Sec. IV, we study the properties of the resulting trust values at the steady state in Sec. V, where the analyses from the Ising model and the spin glasses model are applied, and also the effect of network topology is discussed. Section VI extends the local voting rule to a more general rule and studies their relation. Section VII reviews related work in the literature and Sec. VIII concludes this paper and discusses the future work.

II. Distributed Trust Management

Trust is interpreted as a set of relations among entities participating in network activities [9]. In traditional networks, such as Internet, sources of trust evidence are centralized control servers, such as trusted third parties (TTPs) and authentication servers (ASs). Those servers are trusted and available all the time. Most prior research efforts in this framework [10], [11], [12] assume an underlying hierarchical structure within which trust relationships between ASs are constrained.

In contrast, autonomous networks have neither fixed infrastructures, nor centralized control servers. In these networks, the sources of trust evidence are peers, i.e. the entities that form the network. We summarize the essential and unique properties of distributed trust management in autonomous networks as opposed to traditional centralized approaches:

- **uncertainty and incompleteness**: Trust evidence is provided by peers, which can be incomplete and even incorrect.
- **locality**: Trust information is exchanged locally through individual interactions.
- **distributed computation**: Trust evaluation is performed in a distributed manner.

To manage trust in such a distributed way has several advantages. Because of locality, it saves network resources (power, bandwidth, computation, etc.). It avoids the single point of failure problem as well. Moreover, the networks we are interested in are dynamic with frequent topology and membership changes, and distributed trust has the desired emergent property [13] as entities only contact a few other entities that are easy-to-reach.

However, because of its distributed nature, the control of such distributed trust systems is much more difficult, which includes trust evidence collection, policy specifications, evaluation rules, etc.. In this paper, we are not concened with the problems of how peers obtain the necessary pieces of trust evidence on others. We rather analyse what kind of procedure entities in the network could use for deriving conclusions, once they have obtained the necessary information. Our objective is to design an evaluation rule that has desired performance even under "anarchy".

III. Problem Formulation

A. Network Model

We model an autonomous network as a directed graph $G(V, E)$, in which nodes are the entities/peers in the network and links represent trust relations. Graph G is called the *trust graph*, in order to distinguish the physical graph, in which nodes are connected if they are one hop away in terms of physical transmissions. Suppose that the number of nodes in the network is N, i.e. $|V| = N$ and nodes are labeled with indices $V = \{1, \ldots, N\}$.

In a distributed environment, there is no centralized system to manage trustworthiness of entities. However, entities may still rate each other based on their previous interactions. For example, when node i requests files from node j, i may rate j based on whether j replies to his requests and the quality of these files. A directed link from node i to node j in G, denoted as (i, j), corresponds to the *direct* trust relation that entity i has on entity j[1] and the weight on the link represents the degree of confidence i has on j, denoted as $c_{ij} : V \times V \to [-1, 1]$. $c_{ij} = 1$ represents completely positive confidence i has on j, and $c_{ij} = -1$ represents completely negative confidence. $c_{ij} = 0$ means totally uncertain, so if i and j have no interactions, i.e. $(i, j) \notin E$, c_{ij} may set to be 0. Trust relations are asymmetric, so generally $c_{ij} \neq c_{ji}$. Throughout the paper, we assume nodes' opinions are fixed for the sake of analyses. We define the neighbor set of node i as

$$\mathcal{N}_i = \{j | (i, j) \text{ or } (j, i) \in E\} \subseteq V \setminus \{i\},$$

which is the set of nodes that are directly connected to i. Fig. 1 is an example of a trust graph, where $\mathcal{N}_1 = \{2, 4\}$.

[1]Trust relations based on recommendation are not direct relations. A trust relation from i to j does not necessarily mean that i trusts j. Trust relations include distrust (i.e. negative opinions) as well.

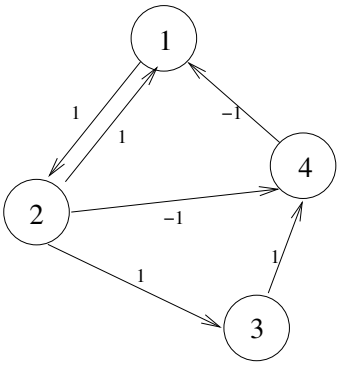

Fig. 1. A trust graph

B. Trust and Confidence Values

Nodes in the network are assumed to be either GOOD or BAD, denoted by $t_i = 1$ or -1 for node i. The vector $T = [t_1, \ldots, t_N]$ is called the *real* trust vector in order to distinguish it from the *estimated* trust vector below. Mathematically speaking, trust evaluation is to estimate the trustworthiness of nodes. Let s_i be the estimated trust value of node i and vector $S = [s_1, \ldots, s_N]$ be the estimated trust vector or simply called trust vector. If $s_i = 1$, we call node i trusted, which is a subjective concept, while $t_i = 1$ means node i is a good node, which is an existing but unknown fact. The evaluation result is the estimate s_i rather than the real trust value t_i, which is the value to be estimated.

The confidence value c_{ij}, also called the c-value, is the degree of confidence node i has on node j, where $c_{ij} \in [-1, 1]$. Without centralized trusted authority, confidence values may not be able to represent true states of the target even from good nodes. For example, in the network with active attackers, the target node that used to be good may be compromised by bad nodes, or because of communication constraints, the past experience may not completely represent the current behaviors of the target. So c_{ij} is modeled as a random variable depending on the real trust values of i and j. The conditional probability $\Pr[c_{ij}|t_i, t_j]$ represents the probability of the c-value equal to c_{ij} given t_i and t_j. Trivially, if $(i, j) \notin E$, $\Pr[c_{ij} = 0|t_i, t_j] = 1$. We assume that all c-values are independent with each other, i.e.,

$$\Pr[c_{ij}, c_{kl}|t_i, t_j, t_k, t_l] = \Pr[c_{ij}|t_i, t_j] \cdot \Pr[c_{kl}|t_k, t_l],$$

where i, j, k, l may not be distinct.

C. Local Voting Rule

For a homogeneous distributed network, all nodes are equal. There is no reason to specialize any particular node. Therefore, trust evaluation should take all available trust information into account. Suppose node i is the target of trust evaluation. The natural approach is to aggregate all its neighbors' opinions. This approach is called a *local voting rule*, in which votes are neighbors' c-values on the target. However, a rule using naïve summation is not a good estimate, because of the following reasons:

1) Trustworthiness of voters: Opinions from nodes with high (estimated) trust values are more credible, so they should carry larger weights. On the other hand, a vote from a distrusted voter should not be valid or even has negative effects. Suppose node i gets positive votes from a couple of distrusted voters, it is reasonable for others to doubt on the trustworthiness of i. So the voting rule should be a weighted sum.
2) Conflicting opinions between the target and voters: Suppose j is one of the voters of target i and their opinions on each other are conflicting, say $c_{ij} = 1$, while $c_{ji} = -1$. Then with high probability, either one of them has made a wrong decision or one of them is a bad node. Therefore, votes from conflicting pairs are less credible. In order to mitigate the effect of such conflicting votes, our solution is to use *effective* votes, which are defined as

$$\hat{c}_{ji} = c_{ji} + \alpha c_{ij} \tag{1}$$

where α is a constant. To simplify our analyses, from now on α is set to be 1. Our discussion in Sec. VI also confirms that it is reasonable to take $\alpha = 1$. By applying effective votes, the absolute values of conflicting votes are reduced, i.e. the conflicting votes are mitigated, while votes from pairs in consistency are strengthened.

Based on the above arguments, we define the local voting rule as the following

$$s_i = f(\hat{c}_{ji} s_j | j \in \mathcal{N}_i) \tag{2}$$

where $f : R \to [-1, 1]$. Apparently, the trust value s_j depends on trust values of j's neighbors and their votes on j. Notice that s_j is also evaluated at the same time, and so are j's neighbors. The whole evaluation therefore evolves as the local interactions iterate throughout the network and Eqn. (2) can be written as

$$s_i(k+1) = f\left(\hat{c}_{ji} s_j(k) | j \in N_i\right). \tag{3}$$

Thus the trust evaluation can be considered as a dynamic process which evolves with time.

Our interest is to study the evolution of the estimated trust vector S and its values at the equilibrium. The motivation of trust management is, of course, to be able to detect bad nodes and trust good nodes. It is important to investigate whether S can correctly estimate the trust vector T at the steady state.

IV. A Stochastic Threshold Rule

Guided by the voting rule in Eqn. (3), we design a specific evaluation rule for analysis. At each iteration, assume that the voting result is binary, i.e. $s_i(k) \in \{1, -1\}$. So the target node is either trusted or distrusted and the voting result is decided by the following threshold rule

$$s_i(k+1) = \begin{cases} 1, & \text{if } m_i(k) \geq \eta \\ -1, & \text{if } m_i(k) < \eta \end{cases}, \tag{4}$$

where

$$m_i(k) = \sum_{j \in \mathcal{N}_i} \hat{c}_{ji} s_j(k) \tag{5}$$

is the weighted sum of the votes from i's neighbors.

However, as we have discussed, uncertainty of opinions by peers is inevitable for autonomous networks. Thus we introduce randomness into our rule. Obviously, if the weighted sum m_i is large, s_i will take value 1 with high probability and vice versa. If m_i is right on the threshold η, it should choose 1 or -1 with equal probability. So our stochastic threshold rule is defined as:

$$\Pr[s_i(k+1)=1|m_i(k)] = \frac{e^{b(m_i(k)-\eta)}}{Z_i(k)} \quad (6)$$

$$\Pr[s_i(k+1)=-1|m_i(k)] = \frac{e^{-b(m_i(k)-\eta)}}{Z_i(k)} \quad (7)$$

where $Z_i(k)$ is the normalization factor

$$Z_i(k) = e^{b(m_i(k)-\eta)} + e^{-b(m_i(k)-\eta)}, \quad (8)$$

and $b > 0$ is a constant representing the degree of certainty. A small b represents a highly uncertain scenario. By placing the value of $s_i(k+1)$ into the right hand sides of both Eqn. (6) and Eqn. (7), the stochastic voting rule can be combined into one formula

$$\Pr[s_i(k+1)|m_i(k)] = \frac{e^{bs_i(k+1)(m_i(k)-\eta)}}{Z_i(k)}. \quad (9)$$

A. Update Sequence

Our evaluation rule is essentially an updating rule. Another component of an updating rule is the update sequence. We list three possible types of update sequences

- *Synchronous updates*: Trust values of all nodes are updated at the same time.
- *Ordered asynchronous updates*: Trust values are updated one at a time in a predefined order.
- *Random asynchronous updates*: Trust values are updated one at a time. The updated node is randomly chosen following a distribution.

In the autonomous environment, it is very difficult to achieve synchronicity, which involves complicated algorithms and substantial control messages. Thus the system should only use asynchronous updates. The ordered updates require extra control as well, so we only study random asynchronous updates, but the analyses in the following are also valid for the ordered asynchronous updates with only a few modifications. In random asynchronous updates, the probability that node i is chosen as the target is defined as q_i, and $\sum_{i\in V} q_i = 1$.

B. Markov Chain Interpretation

Our trust evaluation rule in the form of Eqn. (9) has the obvious Markov property: the value of s_i at time $k+1$ only depends on m_i at time k and is independent of all the rest history. Furthermore, according to the *local* voting rule, $s_i(k+1)$ is independent of $s_h(k), \forall h \notin \mathcal{N}_i$. Thus we have

$$\Pr[s_i(k+1)|m_i(k)] = \Pr[s_i(k+1)|S(k)] \quad (10)$$

Notice that Eqn (10) exhibits the Markov type property in the spatial situation. The distribution with such property is called a Markov random field (MRF), whose detailed definition will be given in Sec. IV-D. At present, only properties related to Markov chains are going to be discussed.

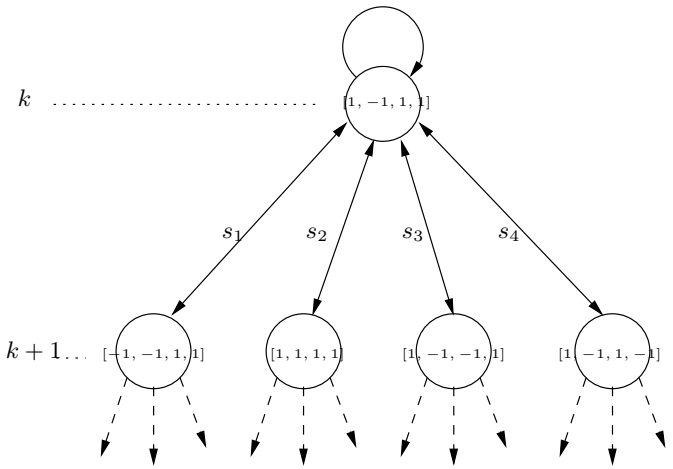

Fig. 2. Part of the Markov Chain. Suppose $S(k) = [1, -1, 1, 1]$, then $S(k+1)$ either flips one of the element in $S(k)$ or stays at the state of $S(k)$.

From Eqn. (10), the state of the Markov chain at time k is a configuration of S, and at time $k+1$, s_i changes while all other nodes keep unchanged. Combining Eqn. (10) and the random asynchronous updates, our iterated voting rule can be considered as a Markov chain, where the states are composed of all the possible configurations of vector S. Suppose at time k, we are at state $S = [s_1, \ldots, s_i, \ldots, s_N]$. Denote the transition probability from state S to state $\bar{S}^i = [s_1, \ldots, \bar{s}_i, \ldots, s_N]$ as $p_{S,\bar{S}^i}$, where $\bar{s}_i = -s_i$, then we have

$$p_{S,\bar{S}^i} = q_i \Pr[\bar{s}_i|S] = q_i \Pr[\bar{s}_i|m_i]. \quad (11)$$

Then the probability of staying still at S at time $k+1$ is

$$p_{S,S} = 1 - \sum_{i\in V} p_{S,\bar{S}^i}. \quad (12)$$

Fig. 2 is an example of the Markov chain for a network with four nodes, such as the one in Fig. 1. We only depict one state transition due to space constraints. Suppose at time k, $S(k) = [1, -1, 1, 1]$, i.e. the state at the top of Fig. 2. Each outgoing edge represents a state transition. The one with s_i on the edge means transition from S to $\bar{S}^i$. For instance, the leftmost transition is to flip the value of s_1. In Fig. 1, node 2 and 4 are the neighbors of 1. By Eqn. (5), we get

$$m_4(k) = (c_{12} + c_{21})s_2(k) + c_{41}s_4(k) \quad (13)$$

If we substitute $c_{12} = c_{21} = 1$, $c_{41} = -1$ and $s_2(k) = -1$, $s_4(k) = 1$, we have $m_1(k) = -3$. Then the transition probability

$$p_{S,\bar{S}^4} = q_1 \Pr[s_1 = -1|m_1 = -3] = \frac{e^{3b}}{4(e^{3b} + e^{-3b})}, \quad (14)$$

where we assumed $q_i = 1/N = 1/4$.

C. Convergence

Having developed the probabilistic interpretation of our local voting rule in terms of a Markov chain, in this subsection, we study convergence of the Markov chain. We will show that the Markov chain converges and give the explicit formula for the stationary distribution. First is a lemma used for the proof of convergence.

Lemma 1: The Markov chain with transition probability defined as in Eqns. (11) and (12) is irreducible and aperiodic, given $b \in (0, \infty)$ and $q_i > 0, \forall i \in V$.

This lemma is easy to verify by proving that all states have a self-loop with nonzero probability and all are connected with each other under the conditions for b and q_i.

Furthermore, the Markov chain is finite with dimension 2^N, so it is a regular Markov chain. It is known that there is a unique stationary distribution π for a regular Markov chain. Therefore, our voting rule does converge under the trivial conditions $b \in (0, \infty)$ and $q_i > 0, \forall i \in V$. In order to derive the stationary distribution, we introduce the notion of a *reversible* Markov chain.

Definition 1: A Markov chain is reversible if

$$\pi_S p_{S,R} = \pi_R p_{R,S} \text{ for all } R, S, \tag{15}$$

where R, S are the states and π_S is the probability of being in state S at the steady state.

In other words, for a reversible Markov chain at the steady state, the process looks the same forward as well as backward. Hence it is said to be *reversible*[2]. Furthermore, the following theorem which is the reverse of Definition 1 provides a way to compute the stationary distribution [14].

Lemma 2: Suppose that π is a probability distribution satisfying Eqn. (15), then π is the unique stationary distribution and the chain is reversible.

Proof: This is true because Eqn. (15), sometimes called the *detailed balance equation*, implies

$$\sum_S \pi_S p_{S,R} = \pi_R \sum_S p_{R,S} = \pi_R \text{ for all } R \tag{16}$$

and therefore π satisfies the *balance equation* of the stationary distribution. ■

Inspired from [15], we define the *energy* of configuration S as $U(S) = \sum_{(i,j)\in E}(c_{ij} + c_{ji})s_i s_j - \eta \sum_{i\in V} s_i$ and a distribution π on the states of the Markov chain as following

$$\pi_S = \frac{e^{bU(S)}}{Z} \tag{17}$$

where Z is the normalization constant, also called the partition function with

$$Z = \sum_S e^{bU(S)} \tag{18}$$

Then we have the following theorem.

Theorem 3: For the stochastic voting rule defined by Eqn. (9) and using random asynchronous updates, if $b \in (0, \infty)$ and $q_i > 0, \forall i \in V$, we have that

[2]Notice that not all regular Markov chains are reversible.

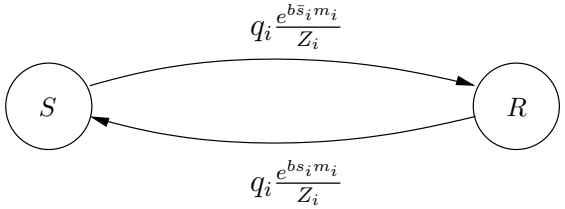

Fig. 3. Transitions between S and R. $p_{S,R} = q_i \frac{e^{b\bar{s}_i m_i}}{Z_i}$ and $p_{R,S} = q_i \frac{e^{bs_i m_i}}{Z_i}$.

1) the voting rule converges to the steady state with a unique stationary distribution;
2) the distribution $\pi_S = \frac{e^{bU(S)}}{Z}$ is the unique stationary distribution.

Proof: The convergence has been verified based on Lemma 1. To prove π is the stationary distribution, we just need to check if π satisfies Eqn. (15) according to Lemma 2.

Consider now two adjacent states S and R in the Markov chain. $S = [s_1, \ldots, s_i, \ldots, s_N]$ and $R = [s_1, \ldots, \bar{s}_i, \ldots, s_N]$. So the transitions between R and S are just flipping s_i. Since all the other nodes are the same, m_i of both R and S is the same, which is

$$m_i = \sum_{j\in\mathcal{N}_i} (c_{ji} + c_{ij}) s_j.$$

Fig. 3 shows the transitions between R and S. We need only verify Eqn. (15), i.e.

$$\frac{e^{bU(S)}}{Z} p_{S,R} = \frac{e^{bU(R)}}{Z} p_{R,S} \tag{19}$$

or

$$\frac{p_{S,R}}{p_{R,S}} = \frac{e^{bU(R)}}{e^{bU(S)}} \tag{20}$$

We know that

$$\begin{aligned} \frac{p_{S,R}}{p_{R,S}} &= e^{b(\bar{s}_i - s_i)m_i} \\ &= e^{b\sum_{j\in\mathcal{N}_i}((J_{ji}+Jij)\bar{s}_i s_j - (J_{ji}+Jij)s_i s_j)} \\ &= e^{b(U(R)-U(S))} \end{aligned}$$

Thus, Eqn. (15) is satisfied and π is the stationary distribution of our voting rule. ■

Having derived the stationary distribution, we are able to compute the probability of correct estimation. Let vector SS be equal to the trust vector S at the steady state. Then the probability of correct estimation, including trusting good nodes and detecting bad nodes, is

$$\begin{aligned} P_{correct} &= \{ \text{ Expected \# of } SS_i = T_i \} \\ &= \mathbf{E}\left[1 - \frac{\|SS - T\|_1}{2N}\right]. \end{aligned}$$

where $\|SS - T\|_1 = \sum_{i\in V} |SS_i - T_i|$. The last equation is true because $SS, T \in \{1, -1\}$.

The stationary distribution

$$\pi_S = \frac{e^{bU(S)}}{Z}$$

is called *Gibbs distribution*. The Gibbs distribution is closely related to local interactions of our voting rule, as explained in the next subsection.

D. Markov Random Field

Recall Eqn.(10) in Sec. IV-B with a small modification: replacing m_i with $s_j(k), j \in \mathcal{N}_i$.

$$\Pr\left[s_i(k+1)|S(k)\right] = \Pr\left[s_i(k+1)|s_j(k), j \in \mathcal{N}_i\right]. \quad (21)$$

Equation (21) in fact presents a Markov type property, i.e., the probability of the estimated trust value for a certain node i, s_i, given the estimated trust values of all the other nodes in the network, is the same as the probability of s_i, given only the estimated trust values of the neighbors of i. As opposed to the Markov chain, which has the Markov property with respect to time, Eqn. (21) shows Markov property in the space dimension. A distribution with such property is called a *Markov random field* (MRF).

The well-known Hammersley-Clifford theorem [15] proves the equivalence between a MRF on a graph and the Gibbs distribution. So our voting rule at the steady state is a MRF, and the essential reason is because our rule only depends on *local interactions*.

V. Trust at the Steady State

In this section, we investigate properties of the estimated trust values when the voting rule reaches the steady state. At first, we introduce an important model that models local interactions of magnets in physics – the Ising model.

A. Ising Model and Spin Glasses

For those familiar with statistical physics, it is very easy to link the Gibbs distribution of π with the Ising model [16] in statistical physics. The Ising model describes interaction of magnetic moments or "spins" of particles, where some particles seek to align with one another (ferromagnetism), while others try to anti-align (antiferromagnetism). In the Ising model, s_i is the orientation of the spin at particle i. $s_i = 1$ or -1 indicates the spin at i is "up" or "down" respectively. A Hamiltonian, or energy, for a configuration S is given by

$$H(S) = -\sum_{(i,j)} J_{ij} s_i s_j - mH \sum_i s_i. \quad (22)$$

The first term represents the interaction between spins. The second term represents the effect of the external (applied) magnetic field. Then the probability of configuration S is given by

$$\Pr[S] = \frac{e^{-\frac{1}{kT}H(S)}}{Z}, \quad (23)$$

where T is the temperature and k is the Boltzmann constant. In the Ising model, the local interaction "strengths" J_{ij}'s are all equal to a constant J, which is either 1 or -1. The Ising model has been extensively studied ever since Ernst Ising published his results on this model in 1920s. In recent years, an extension of the Ising model called the Edwards-Anderson model of spin glasses is used to study local interactions with independently random J_{ij} [17], which corresponds to c_{ij} in our voting rule. The rich literature in statistical physics will no doubt help us to understand our voting model at the steady state.

B. Virtuous Network

Now go back to our discussion on trust at the steady state. We start with the simplest case: a virtuous network, where all nodes are good and they always have full confidence on their neighbors, so $t_i = 1, \forall i \in V$ and $c_{ij} = 1, \forall (i,j) \in E$. Then the stationary distribution π is exactly the same as the one in the Ising model with

$$b = \frac{1}{2kT} \text{ and } \eta = -\frac{mH}{kT}. \quad (24)$$

Since all nodes are good with $t_i = 1$ and SS_i is either 1 or -1, the probability of correct estimation can also be written as

$$P_{correct} = \frac{\mathbf{E}\left[\langle SS\rangle\right] + N}{2N},$$

where $\langle SS\rangle = \sum_{i\in V} SS_i$. In the terminology of physics, $\langle SS\rangle$ is called the total magnetization. It is known that when the external field $H > 0$, $\mathbf{E}\left[\langle SS\rangle\right]$ is positive and when $H < 0$, it is negative. According to (24), the threshold $\eta > 0$ corresponds to $H < 0$, thus $\mathbf{E}\left[\langle SS\rangle\right] < 0$ and $P_{correct} < 0.5$. Similarly when η is negative, $P_{correct} > 0.5$.

We use simulations to study the value of $P_{correct}$ with respect to parameters η and b. In this section, the network topology for all the simulations is a two-dimensional lattice with periodic boundary. The number of nodes is 100 and each takes four nearest nodes as their neighbors. We chose the lattice because most theoretical results for the Ising model are for the 2-D lattice. In Sec. V-D, we will discuss the effect of network topology.

Fig. 4 and Fig. 5 represent the probability of correct estimation as a function of b for η being negative, positive or zero. While $\eta > 0$, which means that the rule is chosen to be conservative, the probability of correct estimation is less than half. Such a result is obviously undesired, since it is even worse than randomly choosing the trust values, which can achieve the mean 0.5. Therefore, it is infeasible to use a positive threshold. This conclusion is rather surprising given the natural thought of setting a threshold when using a voting rule. On the other hand, for $\eta = 0$ or $\eta < 0$, if the value b is properly chosen ($b > 0.6$), $P_{correct}$ is close to 1. Therefore, the threshold η must be non-positive.

Without our effort of deriving the stationary distribution π, the conclusion that η cannot be greater than zero is not an obvious fact if we only look at the local voting rule. This simple result elucidates the power and necessity of conducting analytical studies.

The other interesting property is the phase transition phenomenon observed in Fig. 5 when b is in $[0.4, 0.5]$. Phase transition has been extensively studied by physicists in the Ising model. For the Ising model on a two-dimensional lattice with $H = 0$, there exists a so-called *critical temperature* T_c.

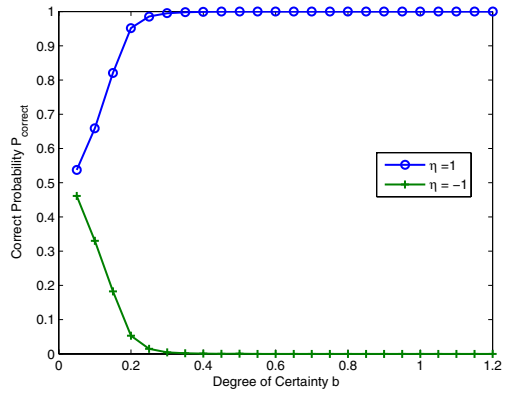

Fig. 4. P_c vs. b with $\eta < 0$ and $\eta > 0$.

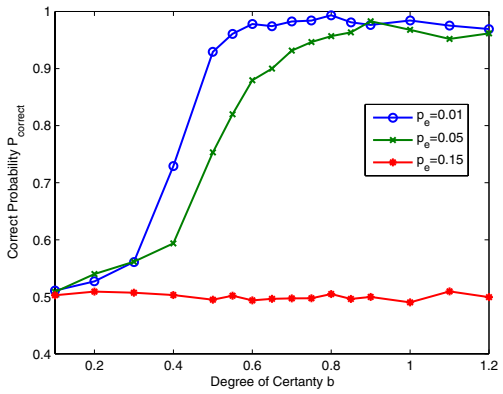

Fig. 6. P_c vs. b with link errors p_e. $\eta = 0$.

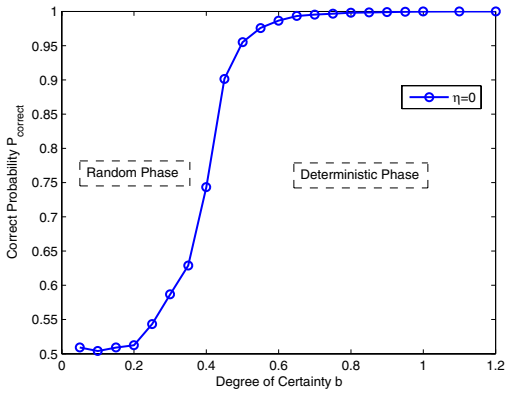

Fig. 5. $P_{correct}$ vs. b with $\eta = 0$.

When the temperature T is above T_c, all the spins behave nearly independently (no long-range correlation), whereas when temperature is below T_c, all the spins tend to stay the same (i.e., cooperative performance). The above observation is expressed in the following in terms of the total magnetization $\langle SS \rangle$,

$$\mathbf{E}\left[\langle SS \rangle\right] \begin{cases} = 0, & \text{if } T > T_c \\ \neq 0, & \text{if } T < T_c \end{cases} . \tag{25}$$

For a 2-D lattice, the value of the critical temperature can be accurately calculated as $2kT_c = \frac{2}{1+\sqrt{2}} = 2.269$, which is consistent with our simulation result where the critical value of b, denoted as b_c, is in $[0.4, 0.5]$, given the relation of b and T in Eqn. (24).

If we look closer into the interval when $b < 0.4$, the estimated trust value of each node is changing all the time and looks like random jumping. While when b is above the critical value, all values converge steadily to 1. So we call the first interval the *random phase*, while the second is the *deterministic phase*.

The discovery of phase transition in our voting rule is quite surprising given that the rule itself is very simple. More importantly, the fact that a small change in the parameter might result in a totally opposite performance of our voting rule proves the necessity of doing more analyses before applying any distributed algorithms. A rule with arbitrary or unverified parameters may ruin performance of the whole system.

As we have discussed, due to uncertainty and incompleteness of trust evidence, c_{ij} should be modeled as a random variable rather than being always 1. Let's assume $c_{ij} \in \{-1, 0, 1\}$ and define the probability that a good node has an incorrect opinion on its neighbors as p_e, then we have

$$p_e = \Pr\left[c_{ij} \neq t_j | t_i = 1\right] \text{ for all } i \text{ good.}$$

Thus in a virtuous network, the distribution of c_{ij} is

$$\Pr(c_{ij} = 1) = 1 - p_e; \quad \Pr(c_{ij} = -1) = p_e, \tag{26}$$

which is simplified as $c \sim (1-p_e, p_e)$. This model corresponds to the $\pm J$ model in spin glasses, where $J \sim (p, 1-p)$.

We again investigate the phase transition. As shown in Fig. 6, the phase transition still happens when $\eta = 0$. However, as p_e increases, the wrong votes with value -1 gradually destabilize the votes of value 1. Thus it is harder to keep s_i's equal to 1, which means that b_c becomes larger and the system more probably stays in the random phase given a high link error p_e. When p_e is large enough, as shown in the figure, where $p_e = 0.15$, the system always stays in the random phase.

In [17], the authors theoretically studied phase transitions between random and deterministic phases, and introduced the replica symmetry method to solve them analytically. Based on this method, very good approximations of values, such as $\mathbf{E}[\langle SS \rangle]$ and $\mathbf{E}[SS_i^2]$, can be derived. The mathematical manipulation of the replica symmetry method is beyond the scope of this paper, but it is definitely a very good direction for our future work. Given explicit expressions for these values, they will provide even better guide for network management and control design.

C. Adversary Model

The motivation of trust evaluation is to be able to identify good nodes and detect bad nodes. In this section, we study a

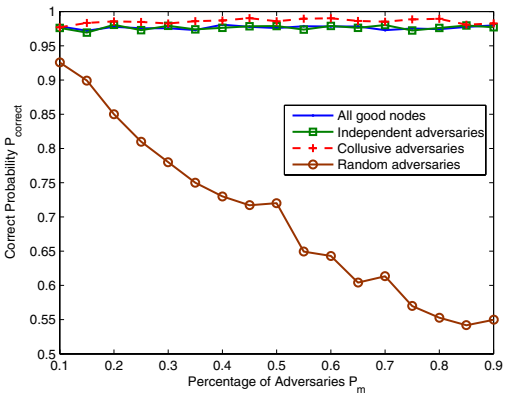

Fig. 7. P_c vs. the percentage of adversaries P_m. The link error $p_e = 0.05$ and the degree of certainty $b = 1$.

network in the presence of adversaries, i.e., bad nodes. Three types of adversaries are considered:

- **Independent**: adversaries do not collude with each other and have normal power. We assume their probabilities of wrongly identifying whether a node is good or bad are p_e the same as good nodes. Once they identity a node as their friend, i.e. a bad node as well, they rate it with value 1 and vice versa.
- **Collusive**: adversaries know each other. They always vote for their friends with value 1 and vote for good nodes with value -1.
- **Random**: to confuse the evaluation rule, they randomly assign confidence values on others.

The voting rule used here is the one with threshold $\eta = 0$, while the following results can be extended to the case where $\eta < 0$. First consider independent adversaries. Define a new network G^* which has the same topology as the current network G and same probability of link error p_e, but with all nodes being good. Suppose the probability of correct estimation for G^* is $P^*_{correct}$, we have the following theorem:

Theorem 4: The probability of correct estimation in the presence of independent adversaries $P_{correct}$ is equal to the probability of correct estimation $P^*_{correct}$ for the network G^* with all good nodes. $P_{correct}$ is independent of the number of adversaries.

Proof: Please see the Appendix ■

Therefore, the performance of our voting rule is independent of the number of adversaries in the case where all adversaries are independent. This is a very valuable property for a trust evaluation rule, and we can apply all the analyses in the last two subsections to the network with independent adversaries.

We use simulations to compare the three types of adversaries. In each simulation, the only difference is the behavior of bad nodes. Fig. 7 shows the simulation results. In order to verify Theorem 4, the curve for all good nodes is plotted as well. The curves of all good nodes and independent adversaries nearly overlap except for some small random perturbations, and the performance of independent adversaries does not change with respect to the percentage of adversaries in the network. Both verify the conclusion of Theorem 4. Furthermore, these two curves have quite high $P_{correct}$ because we are able to carefully choose parameters based on the above analyses.

Surprisingly, the curve of collusive adversaries performs better than the ones for all good nodes and independent adversaries. This is because the voting rule has considered distrusted voters and conflicting votes. In the independent case, because of link errors, some adversaries gain trust from good nodes. While in the collusive case, bad nodes always vote good ones negatively, then they are much easier to be detected. The worst performance is the one of random adversaries, because our voting rule does not capture such malicious behavior. We leave as future work to study which adversary strategy leads to the worst performance of the local voting rule.

D. Network Topology

So far the network topology being used is the two-dimensional lattice. While in reality, a trust graph is not just a lattice. In this section we further investigate our trust evaluation rule with respect to network topology. We will show the significant influences of network topology on trust evaluation.

The network model we use is the small-world model, which has its roots in social systems. The first experiment on small-world networks was studied in [18], where mails were delivered using acquaintances. This experiment resulted in "six degree of separation". In the past five years, there has been substantial research on the small-world model in various complex networks, such as Internet and biological systems. As a social concept, distributed trust networks exhibit small-world properties too. In [19], it was shown that the PGP certificate graph has the small-world property. Therefore, the study of network topology using the small-world model will help us to understand practical trust systems.

Several small-world models have been proposed in order to resemble actual networks. In this paper, we use the small-world model proposed by Watts and Strogatz in [20](WS model), because it is relatively simple but retains the fundamental properties of practical networks. In the WS model, we start from a ring lattice with the number of nodes $N = 100$ and degree of each node $k = 4$. Each edge is rewired at random so as to create shortcuts with the percentage of shortcuts being the parameter P_{rw}. This construction models the graph transition between regular lattices ($P_{rw} = 0$) and chaotic random graphs ($P_{rw} = 1$).

Our simulation results are shown in Fig. 8, where different curves represent different shortcut percentages P_{rw}. We observe that the performance improves as the model changes from regular lattices to random graphs. For instance, as $b = 0.4$, $P_{correct} = 0.55$ for regular laatices, while $P_{correct} = 0.85$ for random graphs. This is because a more random network has shorter average distance between any two nodes in the network. Therefore, the number of hops which trust information takes to reach any node in the network is

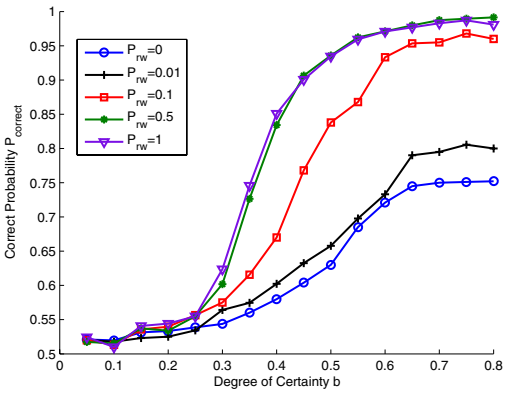

Fig. 8. The effect of network topology. P_{rw} is the percentage of shortcuts in WS small world model. The network with $P_{rw} \in [0.01, 0.1]$ has the small world property. $p_e = 0.1$.

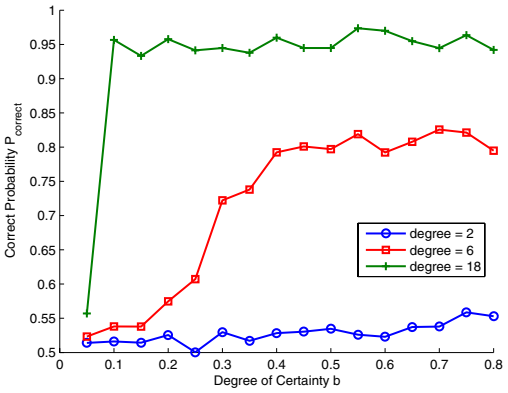

Fig. 9. The effect of average degree with $p_e = 0.1$.

small. The accuracy of trust information degenerates over the path length, so short spreading paths have more accurate information and lead to good results. In particular, the most obvious improvement happens when P_{rw} increase from 0.01 to 0.1, which corresponds to the small-world topology. Therefore, a few shortcuts in the network will greatly improve performance of the trust evaluation rule.

We also investigated the effect of node degree, as shown in Fig. 9. The network is a 100-node ring lattice without shortcuts. The neighbors of a node are those within k hopes distance, where degree k varies. The simulations show that the performance improves as the degree increases. One reason is still that the path length reduces between any two nodes by increasing the number of their neighbors.

From the above discussion, clearly the network topology has great influence on the system performance. As our future work, it is interesting to study our trust evaluation rule under real trust network topologies, and to investigate what kind of network topology has the best performance in terms of trust evaluation.

VI. General Local Evaluation Rule

We have shown that our evaluation rule based on local voting can achieve reasonably good results. In this section, we extend the local voting rule, which is specially designed, to a generalized local evaluation rule. Then we will show that the general rule is in fact a local voting rule under certain conditions.

A. Global and Local Evaluation

Before introducing the general evaluation rule based on local interactions, we first consider the optimal estimation of T without the constraint on locality, so-called *global trust evaluation*. In the case of global trust evaluation, all c-values are collected in one place and serve as the observations for the estimation of T. Therefore, we have the *posterior* of T given c-values as follows

$$\Pr[T|\mathcal{C}] = \frac{\Pr[\mathcal{C}|T]\Pr[T]}{\sum_T \Pr[\mathcal{C}|T]\Pr[T]}, \tag{27}$$

where $\mathcal{C} = \{c_{ij}, \forall (i,j) \in E\}$ is the set of all c-values and $\Pr[T]$ is the *prior* probability of nodes being good or bad, which is assumed to be independent of each other. Due to the independence of c-values and true trust values, we have

$$\Pr[\mathcal{C}|T] = \Pi_{(i,j)\in E} \Pr[c_{ij}|t_i, t_j], \tag{28}$$

and

$$\Pr[T] = \Pi_{i\in V} \Pr[t_i]. \tag{29}$$

Therefore, substituting Eqn. (28) and (29) into Eqn. (27), the distribution of the *estimated* trust vector S

$$\begin{aligned} \Pr[S] &= \Pr[T = S|\mathcal{C}] \\ &= \frac{\Pi_{(i,j)\in E} \Pr[c_{ij}|s_i, s_j] \Pi_{i\in V} \Pr[s_i]}{Z}, \end{aligned} \tag{30}$$

where $Z = \sum_T \Pi_{(i,j)\in E} \Pr[c_{ij}|s_i, s_j] \Pi_{i\in V} \Pr[s_i]$ is the normalization factor. Next, we present an important property of $\Pr[S]$.

Lemma 5: The distribution of the random vector S is a Markov random field, that is

$$\Pr[s_i|S/\{s_i\}] = \Pr[s_i|s_j, \forall j \in \mathcal{N}_i]. \tag{31}$$

Proof: According to the definition of conditional probability

$$\Pr[s_i|S/\{s_i\}] = \frac{Pr[S]}{Pr[S/\{s_i\}]}.$$

Substitute from Eqn. (30) to get

$$\begin{aligned} \Pr[s_i|S/\{s_i\}] &= \frac{\Pi_{(k,l)\in E} \Pr[c_{kl}|s_k, s_l] \Pi_{k\in V} \Pr[s_k]}{\sum_{s_i} \Pi_{(k,l)\in E} \Pr[c_{kl}|s_k, s_l] \Pi_{k\in V} \Pr[s_k]} \\ &= \frac{\Pi_{j\in\mathcal{N}_i} \Pr[c_{ij}, c_{ji}|s_i, s_j] \Pr[s_i]}{Z_i}, \end{aligned} \tag{32}$$

where $Z_i = \sum_{s_i} \Pi_{j\in\mathcal{N}_i} \Pr[c_{ij}, c_{ji}|s_i, s_j] \Pr[s_i]$. The last equation shows that the conditional probability only depends on trust values of nodes that are neighbors of node i. ■

From the conditional probability of s_i in Eqn. (32), we can design a local evaluation rule as follows:

$$\begin{aligned} &\Pr\left[s_i(k+1)|s_j(k), j \in \mathcal{N}_i\right] \\ = \quad &\frac{\Pi_{j\in\mathcal{N}_i} \Pr[c_{ij}, c_{ji}|s_i, s_j] \Pr[s_i]}{Z_i(k)}, \end{aligned} \tag{33}$$

for all nodes in the network. Apparently, this rule only requires that each node knows its own c-values and those of its neighbors. Similarly to the local voting rule, it iterates throughout the network.

Following the same arguments on convergence of the local voting rule in Sec. IV-C, we are able to prove that the stationary distribution π_S of Eqn. (33) is equivalent to the *posterior* probability $\Pr[S]$ in Eqn. (30). This is a significant result, because by using the local information, we are able to obtain the same result as in the case where global information is available. Apparently, there are some disadvantages if we only use local information. The local evaluation rule requires long convergence time. Furthermore, the global rule directly calculates the distribution of S, but the local rule only gives a sampling value of the random variable S following the distribution $\Pr[S]$ at each iteration, and the distribution can only be calculated by averaging over time after the evaluation reaches the steady state.

B. General Rule and Voting Rule

Now assume the *error probability* for a node to make a wrong decision is fixed as p_e and the adversaries are independent, then the iterated evaluation rule can be written in the following form

$$\begin{aligned} &\Pr[s_i(k+1)|s_j(k), j \in \mathcal{N}_i] \\ = \quad &\frac{e^{\sum_{j\in\mathcal{N}_i} b[(c_{ij}+c_{ji})s_i(k+1)s_j(k) - \eta s_i(k+1)]}}{Z_i'(k)} \end{aligned} \tag{34}$$

where b and η' are constants with values $b = \frac{1}{2}\ln[(1-p_e)/p_e]$ and $\eta = \frac{1}{2b}\ln[(1-p_G)/p_G]$, and Z_i' is the modified normalization factor which is equal to a constant times Z_i. Obviously, Eqn. (34) has the same form as the local voting rule in Eqn. (9). The derivation of Eqn. (34) is given in the Appendix.

Since the iterated local rule converges to the global evaluation result, the local voting rule defined in Eqn. (34) is optimal given p_e and p_G. However, in practice, it is difficult to derive this optimal rule, because p_e and p_G are usually unknown, and the value of p_e may vary among different users. Thus it is important to have a good estimate of these two values.

In this section, we extended the local voting rule to a more general rule. In future work, we plan to study the properties of the general rule and thus to be able to design evaluation rules that are feasible for different scenarios.

VII. Related Work

Some of the work on trust we will mention is put into the framework of public key certificates. However they can also be viewed as trust evaluation, in the sense that a certificate represents the trust opinion of the issuer on the target. Blaze et al. [21] is commonly acknowledged as the first to coin the term "trust management", and identify it as a separate component of security services in network management. They designed and implemented the PolicyMaker [21] and subsequent KeyNote [8] to provide a unified framework for describing trust policies, credentials and trust relationships. Their main concerns are on the assertion language for credentials and policies, and the compliance-checking algorithms.

PGP [22] is the most widely used trust management system. In PGP, a distinction is made between the validity of a public key and its trust level. Any user can issue a certificate on another user's key if he/she believes the validity of this key. The trust levels of keys are assigned in any way users want. PGP only determines the validity of a key according to how many keys have signed it based on a set of simple rules. For instance, a key is valid if it is signed by two marginally trusted keys.

In the literature, most of distributed trust evaluation is formulated as a path problem on a weighted, directed graph. In this graph, nodes represent agents (entities or keys), and edges represent trust relations, weighted by the amount of trust that the first agent has on the second. The direct trust relations can be direct trust or recommendation depending on trust policies. The aim is to establish an indirect relation between two agents that have not previously linked. [23], [24], [25] fall into this approach, where one or multiple trust paths are identified between two agents and the trust value is combined by aggregating along and across paths.

There have been several works on trust evaluation based on local interactions as well. In [26], first-hand observations are exchanged between neighboring nodes. Assume Alice receives her neighbors' opinions about a particular node in the network. Alice merges her neighbors' opinions if they are close to her opinion on that node. This work provides an innovative model to link nodes' trustworthiness with the quality of the evidence they provide.

In this paper, we study the inference of trust value given necessary trust evidence rather than generation of direct trust. In this sense, the EigenTrust by Kamvar et al. [27] is similar with our solution. In EigenTrust, in order to aggregate local trust values, a node, say i, asks its neighbors of their opinions about other peers. Neighbors' opinions are weighted by the trust i places on them:

$$t_{ik} = \sum_j c_{ij} c_{jk}, \tag{35}$$

where c_{ij} is i's local trust value for node j and trust values are normalized: $\forall i : \sum_j c_{ij} = 1$. To address the adversary collusion problem, they assume there are peers in the network that can be pre-trusted. [28], [29] proposed similar algorithms that evaluate trust by combining opinions from a selected group of users. One possible selection for the selected users is the neighbors. All these algorithms showed promising results against a variety of threat models by simulations.

VIII. Conlusions and Future Work

The management of distributed and self-organized networks has gained increasing attention because of their wide applications and control difficulties. The interactions in such network management can only be local. Without the global management and control on the network, a small change in local domain may result in dramatic behavior changes on the whole network. Therefore, it is essential to understand the behavior of such autonomous networks before conducting any network management and control. In this work, we study the characteristics of autonomous networks under the context of distributed trust management. Even though the trust evaluation rule we used is very simple, our analyses show extraordinary complexity in terms of the system performance. The analytic results enable us to design the evaluation rule that achieves desired performance.

Our work is just the first step on the exploration of understanding autonomous networks. The simple local voting is not necessarily the best scheme; however it performs well in certain scenarios and presents amazingly complex results. We also proposed a more general evaluation rule based on the global estimation result. This general rule can help to design rules that are feasible for different situations. Evaluation rules based on local interactions have the desired emergent property, which makes the evaluation adaptive to trust dynamics. In this paper, we assume the confidence values are fixed. However trust dynamics is one of the main issues in autonomous networks. So it is necessary to integrate more trust contents into the evaluation rule, such as the worth of each transaction and decay of trust values in time. For the measurement of performance, another metric we are particularly interested is the connectivity of trusted graph [30], which measures the probability of two non-neighbors being able to communicate with each other through a trusted path.

One of our interesting observations is the phase transition phenomenon. Phase transition is very common in any combinatorial structure, where a large combinatorial structure can be modeled as a system consisting of many locally interacting components. Many research fields are mainly studying the phase transitions, such as Ising model, random graph [31] and percolation theory[32]. The phase transition phenomenon is also studied within the networking community, For example, Franceschetti et al. [33] work on phase transitions of wireless network capacity, and Goel et al. [34] proved that all properties in random geometric graphs, which are used in MANETs and sensor networks, undergo sharp phase transitions as long as they are monotone. We believe that study of phase transitions will become more and more popular for autonomous networks.

Appendix I
Derivation of Equation (34)

Given $c_{ij} \in \{1, 0, -1\}$ and the confidence value node i has on j is correct, we have $c_{ij}t_it_j = 1$. For instance, $t_i = 1, t_j = -1$, then the correct confidence value should be $c_{ij} = -1$. Inversely, i makes wrong decision if $c_{ij}t_it_j = -1$. Thus

$$\Pr[c_{ij}t_it_j = 1] = 1 - p_e, \Pr[c_{ij}t_it_j = -1] = p_e,$$

where p_e is the error probability. Therefore the conditional probability of c_{ij} is a function of $c_{ij}t_it_j$ and can be written in the following form

$$\Pr[c_{ij}|t_i, t_j] = e^{b_1 c_{ij} t_i t_j + d_1},$$

where $b_1 = \frac{1}{2}\ln[(1-p_e)/p_e]$ and $d_1 = \frac{1}{2}\ln[(1-p_e)p_e]$. Similarly, the *prior* probability can be written as

$$\Pr[t_i] = e^{b_2 t_i + d_2},$$

where $b_2 = \frac{1}{2}\ln[p_G/(1-p_G)]$ and $d_2 = \frac{1}{2}\ln[p_G(1-p_G)]$. Then substituting the above two equations into Eqn. (9), we have that

$$\Pr[s_i|s_j, j \in \mathcal{N}_i] = \frac{e^{\sum_{j\in\mathcal{N}_i} b_1(c_{ij}+c_{ji})t_it_j + b_2 t_i + d_1 + d_2}}{Z_i}.$$

Therefore, by replacing $b_1 = b$, $b_2 = -b_1\eta$ and $Z_i' = Z_i e^{-d_1-d_2}$, we get Eqn. (34).

Appendix II
Proof of Theorem 4

Theorem 4: The probability of correct estimation in the presence of independent adversaries $P_{correct}$ is equal to the probability of correct estimation $P^*_{correct}$ for the network G^* with all good nodes. $P_{correct}$ is independent of the number of adversaries.

Proof: We know that the distribution of J^*_{ij} is $J^*_{ij} \sim (1-p_e, p_e)$, while the distribution of J_{ij} is

$$J_{ij} \sim \begin{cases} (1-p_e, p_e) & \text{if } t_i \cdot t_j = 1 \\ (p_e, 1-p_e) & \text{if } t_i \cdot t_j = -1 \end{cases}.$$

Therefore we have

$$J^*_{ij} \sim J_{ij}t_it_j. \tag{36}$$

By the definition,

$$P_{correct} = \mathbf{E}\left[1 - \|S - T\|_1/(2N)\right].$$

In probability theory, we know that

$$\mathbf{E}[X] = \mathbf{E}\left[\mathbf{E}[X|Y]\right].$$

Now let's set $X = \|S-T\|_1$, $Y = \{J_{ij}t_it_j\}_{(i,j)\in E}$ and $X^* = \|S^* - T^*\|_1$, $Y^* = \{J^*_{ij}\}_{(i,j)\in E}$. So we are trying to prove

$$\mathbf{E}\left[\mathbf{E}[X|Y]\right] = \mathbf{E}\left[\mathbf{E}[X^*|Y^*]\right] \tag{37}$$

From Eqn. (36), we have $Y \sim Y^*$, so if we can prove

$$\mathbf{E}[X|Y] = \mathbf{E}[X^*|Y^*],$$

given $Y = Y^*$, then Eqn. (37) is true. First consider the left hand side,

$$\begin{aligned} \mathbf{E}[X|Y] &= \sum_S \|S - T\|_1 e^{b\sum J_{ij}s_is_j} \\ &= \sum_S \sum_i |s_i - t_i| e^{b\sum (J_{ij}t_it_j)(s_it_i)(s_jt_j)} \end{aligned}$$

The last equation is because $t_i^2 = 1, \forall i$. Notice that

$$|s_i - t_i| = |t_i||s_i - t_i| = |s_i t_i - t_i^*|$$

because $|t_i| = 1$ and $t_i^* = 1$. If for all i, we take $s_i^* = s_i t_i$, and $J_{ij} t_i t_j = J_{ij}^*$ (which is the assumption), we have

$$\mathbf{E}[X|Y] = \sum_{S^*} \sum_i |s_i^* - t_i^*| e^{b \sum J_{ij}^* s_i^* s_j^*} = \mathbf{E}[X^*|Y^*].$$

■

ACKNOWLEDGEMENT

This work is prepared through collaborative participation in the Communications and Networks Consortium sponsored by the U.S. Army Research Laboratory under the Collaborative Technology Alliance Program, Cooperative Agreement DAAD19-01-2-0011. Research is also supported by the U.S. Army Research Office under grant No DAAD19-01-1-0494. The authors would also like to thank Professor George V. Moustakides for several fruitful discussions and all the anonymous reviewers for their valuable comments.

REFERENCES

[1] "IETF Mobile Ad-hoc Networks (manet) working group." [Online]. Available: http://www.ietf.org/html.charters/manet-charter.html

[2] D. Aguayo, J. Bicket, S. Biswas, and G. J. R. Morriss, "Link-level measurements from an 802.11b mesh network," in *SIGCOMM '04: Proceedings of the 2004 conference on Applications, technologies, architectures, and protocols for computer communications*, Portland, Oregon, USA, 2004, pp. 121–132.

[3] R. Draves, J. Padhye, and B. Zill, "Routing in multi-radio, multi-hop wireless mesh networks," in *MobiCom '04: Proceedings of the 10th annual international conference on Mobile computing and networking*, Philadelphia, PA, USA, 2004, pp. 114–128.

[4] Gnutella. [Online]. Available: http://www.gnutella.com

[5] I. Stoica, R. Morris, D. Karger, M. F. Kaashoek, and H. Balakrishnan, "Chord: A scalable peer-to-peer lookup service for internet applications," in *Proceedings of the ACM SIGCOMM '01 Conference*, San Diego, California, August 2001, pp. 149–160.

[6] "Autonomic communication." [Online]. Available: http://www.autonomic-communication.org/

[7] S. Battiston, G. Weisbuch, and E. Bonabeau, "Decision spread in the corporate board network," *Advances in Complex Systems*, vol. 6, no. 4, pp. 631–664, 2003.

[8] M. Blaze, J. Feigenbaum, J. Ioannidis, and A. D. Keromytis, "The role of trust management in distributed systems security," *Secure Internet Programming: Security Issues for Mobile and Distributed Objects*, pp. 185–210, 1999.

[9] L. Eschenauer, "On trust establishment in mobile ad-hoc networks," M.S. Thesis, ECE Department, University of Maryland, College Park, 2002.

[10] J. Steiner, C. Neuman, and J. I. Schiller, "Kerberos: An authentication service for open network systems," in *USENIX Workshop Proceedings, UNIX Security Workshop*, 1988, pp. 191–202.

[11] B. Lampson, M. Abadi, M. Burrows, and E. Wobber, "Authentication in distributed systems: Theory and practice," in *Proceedings of the 13th ACM Symposium on Operating Systems Principles*, Oct 1991, pp. 265–310.

[12] V. D. Gligor, S.-W. Luan, and J. Pato, "On inter-realm authentication in large distributed systems," in *Proceedings of the IEEE Conference on Security and Privacy*, 1992, pp. 2–17.

[13] V. D. Gligor, "Security of emergent properties in ad-hoc networks," in *Proceeding of the Security Protocols Workshop*, Sidney Sussex College, Cambridge, UK, April 2004.

[14] D. Aldou and J. A. Fill, "Reversible Markov chains and random walks on graphs," monograph in preparation. [Online]. Available: http://www.stat.berkeley.edu/users/aldous/RWG/book.html

[15] R. Kindermann and J. L. Snell, *Markov Random Fields and Their Applications*, ser. Contemporary mathematics. American Mathematical Society, 1980, vol. 1.

[16] S. Wierzchon. (2002) Ising model. Eric Weisstein's World of Physics. [Online]. Available: http://scienceworld.wolfram.com/physics/IsingModel.html

[17] H. Nishimori, *Statistical Physics of Spin Glasses and Information Processing: An Introduction*. Oxford University Press, 2001.

[18] S. Milgram, "The small world problem," *Psychology Today*, vol. 1, no. 61, pp. 60–67, 1967.

[19] S. Capkun, L. Buttyán, and J. P. Hubaux, "Small worlds in security systems: an analysis of the PGP certificate graph," in *Proceedings of The ACM New Security Paradigms Workshop 2002*, Norfolk, Virginia Beach, USA, September 2002, pp. 28–35.

[20] D. J. Watts and S. H. Strogatz, "Collective dynamics of 'small-world' networks," *Nature*, vol. 393, pp. 440–442, 1998.

[21] M. Blaze, J. Feigenbaum, and J. Lacy, "Decentralized trust management," in *Proceedings of 1996 IEEE Symposium on Security an Privacy*, May 1996, pp. 164–173.

[22] P. Zimmermann, *PGP User's Guide*. MIT press, 1994.

[23] T. Beth, M. Borcherding, and B. Klein, "Valuation of trust in open networks," in *Proceedings of 3rd European Symposium on Research in Computer Security – ESORICS'94*, 1994, pp. 3–18.

[24] U. Maurer, "Modelling a public-key infrastructure," in *Proceedings of 1996 European Symposium on Research in Computer Security – ESORICS'96*, 1996, pp. 325–350.

[25] G. Theodorakopoulos and J. S. Baras, "Trust evaluation in ad-hoc networks," in *WiSe '04: Proceedings of the 2004 ACM workshop on Wireless security*. ACM Press, 2004, pp. 1–10.

[26] S. Buchegger and J. L. Boudec, "The effect of rumor spreading in reputation systems for mobile ad-hoc networks," in *Proceeding of WiOpt'03 (Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks)*, 2003.

[27] S. D. Kamvar, M. T. Schlosser, and H. Garcia-Molina, "The EigenTrust algorithm for reputation management in p2p networks," in *Proceedings of the 12th Initernational World Wid Web Conference*, Budapest, Hungary, 2003, pp. 640–651.

[28] S. Marti and H. Garcia-Molina, "Limited reputation sharing in p2p systems," in *Proceedings of the 5th ACM conference on Electronic commerce*. New York, NY, USA: ACM Press, 2004, pp. 91–101. [Online]. Available: http://doi.acm.org/10.1145/988772.988787

[29] L. Xiong and L. Liu, "PeerTrust: Supporting reputation-based trust in peer-to-peer communities," *IEEE Transactions on Knowledge and Data Engineering, Special Issue on Peer-to-Peer Based Data Management*, vol. 16, no. 7, pp. 843–857, July 2004.

[30] S. Capkun and J. P. Hubaux, "BISS: Building secure routing out of an incomplete set of security associations," in *Proceedings of the 2003 ACM workshop on Wireless Security (WiSe)*, San Diego, USA, September 2003, pp. 21–29.

[31] B. Bollobás, *Random Graphs*, 2nd ed. Cambridge University Press, 2001.

[32] G. Grimmett, *Percolation*. New York: Springer-Verlag, 1989.

[33] M. Franceschetti, O. Dousse, D. Tse, and P. Thiran, "On the throughput capacity of random wireless networks," Under revision.

[34] A. Goel, S. Rai, and B. Krishnamachari, "Sharp thresholds for monotone properties in random geometric graphs," in *Proceeding of the thirty-sixth annual ACM Symposium on Theory of Computing (STOC)*, Chicago, IL, 2004, pp. 580–586.