On Deception Detection in Multiagent Systems

Eugene Santos, Jr., *Senior Member, IEEE*, and Deqing Li, *Member, IEEE*

***Abstract*—Deception detection plays an important role in safely and reliably using multientity advisory models such as multiagent intelligence systems. The benevolence assumption people have based their implementations of multiagent (human and/or synthetic) systems on is rarely valid in the real world. Unfortunately, deception detection is extremely challenging. The average detection rate by humans alone is only above chance, and the skill for detection has been shown to be difficult to improve even with training. In psychological studies, deception detection is typically based on examining a person's nonverbal cues and expressions such as facial expressions, gestures, and movements. In this paper, our approach instead is focused on the agent's reasoning process. We detect deception by observing the correlations between agents, which can be used to make a reasonable prediction of the agents' reasoning processes. Our experiments demonstrate the effectiveness of this method and show the impact of different factors on detection rate. We further conduct some preliminary experiments to explore its performance at detecting both disinformation and misinformation and that of identifying more than one deceiver in the system.**

***Index Terms*—Bayesian networks (BNs), deception detection, multiagent system, parametric study.**

I. Introduction

DEFINITIONS of deception arise from numerous disciplines and situations studied [1]–[3]. In particular, we focus on two such definitions: 1) Whaley [4] defines deception as information designed to "manipulate the behavior of others by inducing them to accept a false or distorted presentation of their environment—physical, social, or political" and 2) Burgoon and Buller [5] define deception as a "deliberate act perpetrated by a sender to engender in a receiver's beliefs contrary to what the sender believes is true to put the receiver at a disadvantage." Both definitions point out that deception leads to consequences less favorable to the receiver. Failure to identify deception in time may bring long-term and irreparable harm to the receiver. Unfortunately, deception detection is a challenging task. Humans can only identify 45% to 65% of all deceptions in face-to-face interactions [6]. It is even more difficult when people interact through electronic media [6] and through the Internet [7]. Research on how to successfully detect deception has been gaining ground. In particular, Johnson *et al.* [8] examined the way auditors detect malicious manipulations of financial information by management so as to make the company appear more profitable than it actually is. They noticed that people learn knowledge about how to apply detection heuristics from past experience if a particular form of deception is frequent. However, deception detection is a low base-rate task, as deception occurs infrequently, particularly in domains where interactions and feedback are available. Therefore, people's experience in detecting deception is fraught with failure. In order to address this problem, Johnson *et al.* proposed a model that identifies inconsistencies between agent's actions and goals. The main components of the model are the following [8].

1) *Activation:* Compare expectations and the observed values. The magnitude of the discrepancy between them determines whether to activate further checks.
2) *Hypothesis generation:* Propose hypotheses to explain the inconsistencies.
3) *Hypothesis evaluation:* Assess hypotheses on the basis of their materiality.
4) *Global evaluation:* Aggregate all accepted hypotheses and produce the final judgment.

Following Johnson *et al.*'s model, Santos and Johnson [1] developed a detection method based on multiagent systems, which is able to address the activation step. A multiagent system is a system composed of a group of intelligent agents where each acts according to some role in order to achieve his goal. Thus, in a multiagent system, agents solve problems that may not be solvable by a single agent, by sharing the burden of a task or playing different roles in the society—such as a group of advisors or a collection of experts with varying specialties. In Santos and Johnson's work, a multiagent system is used to simulate a group of human experts who give opinions on a specific task based on their respective knowledge. Details of the work can be found in [1], in which they provided some preliminary ideas about how to apply Johnson *et al.*'s [8] components to deception detection using multiagent systems and conducted a pilot experiment to evaluate its performance in the activation stage.

In this paper, we (re)validate their results more comprehensively and further explore the behavior of the model by studying how stable it is under changes to the testing environment. More specifically, we isolate each parameter of the model to analyze how they influence performance. In practice, the motivation to deceive (intentionally or unintentionally) and the way to deceive (single deceiver or multiple deceivers) vary. Thus, we will also study the practicality of the model by applying it to a multiagent system with multiple deceivers and also evaluating how the model performs with misinformation so as to propose a method to distinguish misinformation from disinformation.

In the next section, we first introduce some related work and briefly describe Bayesian networks (BNs) [9], which are used to simulate the human reasoning process. Section III, which

Manuscript received February 12, 2009. First published November 20, 2009; current version published February 18, 2010. This work was supported in part by the Air Force Office of Scientific Research under Grants F49620-03-1-0014 and FA9550-07-1-0050. This paper was recommended by Associate Editor K. W. Hipel.

The authors are with the Thayer School of Engineering, Dartmouth College, Hanover, NH 03755 USA (e-mail: eugene.santos.jr@dartmouth.edu; deqing.li@dartmouth.edu).

Digital Object Identifier 10.1109/TSMCA.2009.2034862

describes the detection method is followed by a discussion on how to construct the testbed in Section IV. We then present our experimental results and parametric study in Sections V and VI. Further explorations including simulating misinformation and simulating multiple deceivers will be discussed in Section VII. Finally, we present our conclusions and an outlook on future work.

II. Background

In this section, we discuss some related work in deception detection and compare and contrast them with Santos and Johnson [1]. Next, we provide an overview of BNs which serves as the knowledge representation scheme in Santos and Johnson's approach.

A. Related Work

Recent research focusing on identifying deception using multiagent systems has included the concept of "trust management" or "reputation management" as introduced by Schillo *et al.* [10]. Their model of trustworthiness is built upon the agents' knowledge of the other agents' past behavior, honest or deceptive. The model may converge accurately after several rounds of decision making. However, the failure to catch the deceiver in the early rounds may already have caused irreparable damage. Moreover, as we mentioned earlier, deception does not frequently occur in real-life situations [8]. Therefore, a method that can alert the victim as soon as deception occurs is ideal. Santos and Johnson's approach stands out since it is able to respond as soon as deception occurs by predicting agents' opinions whenever the agents are consulted. The prediction comes from the correlation in decision between the agents in previous tasks.

Another approach which prototypes a model combining different deception-detection techniques was proposed by Vyas and Zhou [11]. The model covers a holistic detection process, including searching for vulnerabilities and indications, analyzing logged information, and undoing the damage from deception. The intent of the deceiver and the environment are taken into consideration in order to collect more precise indicators. For example, potential deceptions are indicated from specific vulnerabilities of the environment. The vulnerabilities may motivate the malicious intent of an agent and lead him into the manipulation of the environmental information. However, some processes and assumptions in the approach may not be consistent with real-world expectations. For example, all members in the society are assumed to have up-to-date and genuine knowledge about both the environment and the other agents, which is likely to be impossible in the real world. In practice, deceivers may hide information and, more seriously, provide incorrect information to confuse the receivers. In comparison, Santos and Johnson's model successfully identifies deception even with incomplete information about the environment. Another problem arises from Vyas and Zhou's approach to generate deception indicators. Conflicts between agents are retrieved as an indication of the vulnerability of the society, which will be used as evidence suggesting possible deception. However, deception may come from cooperative agents who do not have significant conflicts of interest, in which case, it is hard to find any vulnerability. In contrast, Santos and Johnson's model is independent of the knowledge domain of the expert and thus, can be applied in any environment with the agents pursuing different or common interests with only the assumption that the experts share similar knowledge.

Other detection research such as Rowe [12] and Wang *et al.* [13] are primarily focused toward their specific applications. One approach that is similar to Santos and Johnson in using reasoning systems is Stech and Elässer [14]. They employ an adversarial planner together with an analysis of competing-hypotheses (ACH) approach to generate potential hypothesis and actions for adversaries semiautomatically. They also use BNs to infer the most probable hypothesis from observed actions. However, the effectiveness of their approach depends on the choice of hypothesis and the user's assessment of probabilities, while in Santos and Johnson, the detection rate does not involve human interpretations and is stable with respect to environmental parameters, as will be shown later in Section V.

B. Bayesian Networks (BNs)

In Santos and Johnson's approach [1], each agent in the multiagent system represents the decision-making process of a human expert. A decision-making process involves knowledge and reasoning about the knowledge. The knowledge is captured in a *knowledge base,* and the brain that the system uses to reason about the knowledge is called the *inference engine*. How to represent the experts' knowledge is one of the principle fields of study for knowledge-based systems. For the problem of deception detection in Santos and Johnson [1], the system must also be capable of coping with uncertainty. As such, a probabilistic knowledge representation based on a graphical representation of conditional probabilistic dependences called BNs [9] was chosen. BNs have been gaining popularity in deception detection to support causal reasoning such as in the ACH-counter deception approach [14]. Our group has extensively studied BNs and their underlying reasoning mechanisms necessary for this work [15].

A BN is an annotated directed acyclic graph, which is composed of nodes and arcs. Nodes store the experts' knowledge in the form of random variables, and directed arcs connecting two nodes represent a conditional/causal relationship between them. The uncertainty of the relationship is encoded in a conditional probability. The conditional probabilities between any random variable and its parents are contained in an associated conditional probability table (CPT). Under the conditional independence assumption, the chain rule, which is also the product of the CPTs, is expressed as

$$P(X_1, X_2, \ldots, X_n) = \prod_{1}^{n} P\left(X_i \,|\, parents(X_i)\right). \quad (1)$$

This provides a representation of the joint probability distribution, with which a BN is able to present the direct relationships between variables and form a structural organization of

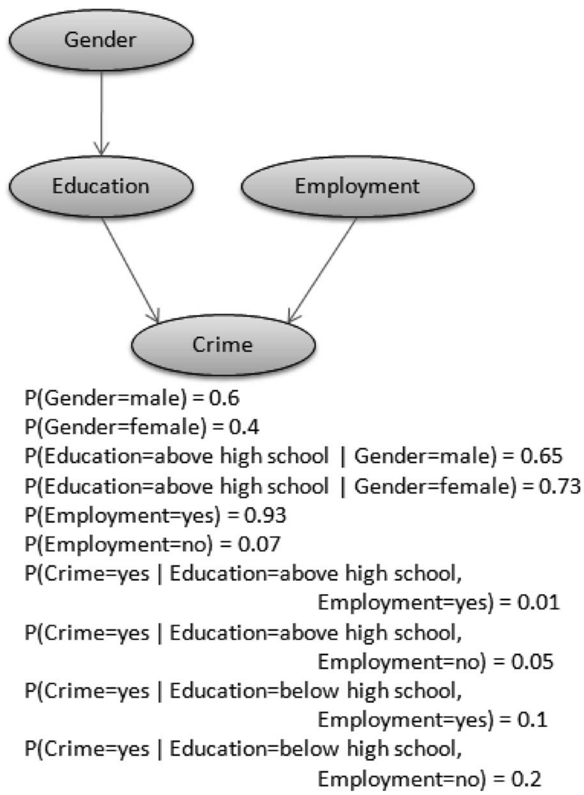

Fig. 1. Simple BN example.

information. During an inference, the probability of each state of a random variable is updated given that the states of other variables are observed. The process of computing the posterior probability of each random variable can be called probabilistic inference, which is achieved by applying Bayes' theorem.

Fig. 1 is a simple example of a BN. It represents the relationship between possible causes and consequences of committing a crime. Each random variable in the example has two states. The arcs between each two nodes denote the causal relationship between possible states of the two random variables. For example, if someone is a male, then his education level is above high school with a probability of 0.65. The roots of the network (*Gender* and *Employment* in this case) have prior probabilities instead of conditional probabilities, which represent the probability of a person being male and that of a person being employed regardless of any evidence.

A BN is a complete model of the reasoning structure of the expert knowledge. In Santos and Johnson's approach [1], each agent of the multiagent system is represented by some BN so that they can simulate experts in the same knowledge domain or experts working on the same task.

III. Deception Detection in Multiagent Systems

In Santos and Johnson's model [1], it is assumed that all agents in the multiagent system share a significant portion of knowledge. Thus, they are expected to provide opinions on a given problem that share similar knowledge. The assumption is reasonable in areas that need highly expert knowledge such as law and medicine. For example, given the same symptoms, multiple doctors will likely provide similar diagnoses (though there can be multiple diagnoses in total). This assumption results in the fact that the agents' opinions are highly correlated because of shared knowledge. In other words, agents who deviated from the majority in the past are expected to have a larger difference with others in the future, while those who were similar to the majority in the past tend to have a smaller difference in the future. Based on this observation, we can regard inconsistent opinions as a possible result of deception. By "inconsistent" we mean that the expert's opinions are inconsistent with his correlations with others, rather than that the expert disagrees with the other experts. We check inconsistency in this way because conflicting opinions are not necessarily wrong, and sometimes, they even form a more comprehensive view about the problem for the decision makers, but, intuitively, people always reason in a similar way given that their knowledge often remains the same. Since it is possible to anticipate agents' opinions based on his correlations with others (following [1]), we can use prediction techniques to predict each agent's potential opinion [16] and compare the prediction against his actual opinion.

The methodology of the model can be summarized as follows: First, calculate the correlations between each two agents by comparing their past opinions. Next, based on the *GroupLens* prediction technique [16], we predict each agent's opinion about the current task. Finally, deceptions will be identified if the predicted opinions are far different from the actual opinions. More specifically, the steps are as follows:

1) *Compare Opinions:* The assumption that agents share similar knowledge indicates that the agents' opinions are correlated with each other. This observation enables us to predict one's opinion based on his correlation with others. Therefore, the first step is to calculate the correlation between two agents based on their opinions from past tasks. The agents' historical inferencing processes are also called the *training processes*, and the opinions generated in the past are called the *training* data. We assume that the training data does not contain any deceptive opinion. Thus, it does not play a role in identifying deception but is used to obtain the correlation values. The correlation measure we use is the *Pearson correlation*, which is calculated as follows:

$$r_{AB} = \frac{Cov(A,B)}{\sigma_A \sigma_B} = \frac{\sum_i (A_i - \overline{A})(B_i - \overline{B})}{\sqrt{\sum_i (A_i - \overline{A})^2 \sum_i (B_i - \overline{B})^2}} \tag{2}$$

where r_{AB} represents the Pearson-correlation coefficient between expert A and expert B. For the ith set of evidence, we define A_i as the posterior probability of expert A and B_i as the posterior probability of expert B. $\overline{A}$ denotes the average of all probabilities assigned to expert A's knowledge base given different sets of evidence and likewise for B.

2) *Predict Opinions:* After the correlations are obtained, we predict each agent's opinion over a set of evidence using the other agent's opinions. Evidence is pieces of information that have been already observed before consulting

the experts. The agents' current inferencing processes are also called the *testing processes*, and the opinions generated in the current task are called the *test* data. The technique we use to predict opinion is based on GroupLens prediction as shown in (3), which allows us to estimate what opinion is expected for each agent provided that the agent's historical opinions are sufficient

$$A_{X_{\text{Prediction}}} = \overline{A} + \frac{\sum (B_{iX} - \overline{B}_i) r_{AB_i}}{\sum_i |r_{AB_i}|} \tag{3}$$

with $A_{X_{\text{Prediction}}}$ denoting the predicted posterior probability of A. For the ith agent, r_{AB_i} is defined as the Pearson correlation coefficient between A and B_i, and X denotes the random variable whose state is unknown to A but is available to B_i.

3) *Identify Deception:* If an agent's actual opinion on a given problem is very different from the predicted one, then it means that he provided an inconsistent opinion, which might be an indication of deception. In this case, we will identify him as a candidate deceiver and activate further detection processes. In practice, we regard an agent as a candidate deceiver if the error between his opinion and expectation is larger than four standard deviations, which covers 99.99% of the normal decision error. Later, in the parametric study, we will adjust the error that we can accept between expected and actual opinions from four standard deviations to one standard deviation for the purpose of understanding the critical factors of the model.

IV. Testbed Construction

In this section, we describe how the testbed is constructed. In order to evaluate this deception-detection methodology, a multiagent system testbed was employed as in [1]. To ease the construction of the testbed, we used existing BNs to simulate the agents. The *Alarm Network* [17], which was originally built to monitor patients with intensive care, was chosen in our pilot experiment (as well as in [1]) because of its moderate size and structure. Multiple agents were simulated by perturbing the conditional probabilities of the Alarm network. A testbed is constructed as follows.

1) *Build agents:* We first created ten agents using the Alarm networks so that they would have the same knowledge structure. By perturbing the CPTs in each network, we made the agents slightly different in their conditional probabilities, which would reflect similar but not exactly the same uncertainty about knowledge. We used a perturbation value to control the noise added in the conditional probabilities. For example, if the perturbation value is 0.1, the noise to be added is within ± 0.1.
2) *Create historical opinions and calculate correlations:* In order to calculate the correlations between agents, we need a sufficient number of historical inferencing processes. In each of the inferences, we feed all the agents with the same set of evidence, reason over the network, and record their posterior probabilities. This procedure was repeated a large number of times to simulate the historical opinions. The correlation value between the two agents was calculated using Pearson correlation. If the correlation value is close to one, it indicates a positive dependence. A negative dependence is denoted by -1. If it is close to zero, it means that the correlation between the two agents is weak.

   After the correlations were obtained, we tried to reproduce the training data through prediction using (3). The error between the predicted training data and the actual one is a reasonable estimation of normal decision error because the training data is assumed to be benign. We assume that the error of prediction follows a normal distribution so that its standard deviation can be used to check whether the error of predicting test data is beyond normal decision error.
3) *Simulate deception and evaluate detection performance:* In the testing process, agents are simulated as deceivers. After the inferencing was conducted, we rotated each agent's posterior probabilities in order to create deceptions. Then, we measure the distance between one's deceptive probabilities and predicted ones. If the error is more than four standard deviations, then we will identify the agent as a candidate deceiver and report a positive detection. We also determine the false activation rate by measuring the errors between each agent's predicted opinion and original opinion before creating deception. If we mistakenly identify any agent as a deceiver in this phase, then we will report a false activation.

TABLE I
Statistics on the Detection Rates of Alarm Network

Parameters	Agents = 100, Repeats = 1000, Perturbation = 0.1, Evidence = 1-10, No. of stdevs = 4				
Positive Detection Rate	Max	1.0	False Detection Rate	Max	0.2518
	Min	0.3770		Min	0.0
	Mean	0.8716		Mean	0.011
	Med	0.9627		Med	0.003

V. Experiments on Deception Detection

Santos and Johnson [1] presented a preliminary experiment evaluating the correlation values of the agents and the detection rate of the system. Here, we repeated the experiment with a modified parameter setting in order to verify the results and provide a more comprehensive analysis. In our experiment, 1000 repeats were conducted, each with a different set of ten pieces of evidence, in both training and testing processes. We perturb the conditional probabilities by ± 0.1. The error we allow for in normal decision deviation must be within four standard deviations. Table I shows the experiment result.

The result is similar to that in Santos and Johnson [1]. From the data, we can see that the mean detection rate is around 87%, which is much higher than the human detection rate (60%) [18], [19]. According to Ford [20], the most competent human detectors are poker players and secret service agents. However, poker players only detect successfully on opponents whom they are familiar with. They achieve a high detection rate by recording others' habits in detail. Secret service agents

TABLE II
STRUCTURES AND DETECTION RATES OF DIFFERENT NETWORKS

Parameters	Agents=10, Repeats = 100, Perturbation = 0.1, No. of stdevs = 4, Evidence = 30% of total nodes				
Network	no. of Nodes	no. of States	no. of Arcs	Mean positive detection rate	Mean false detection rate
Alarm	37	105	46	0.8884	0.0237
Hailfinder	56	223	66	0.8092	0.0226
Diabetes	413	4682	602	**0.4257**	**0.0110**
Munin	1041	5651	1397	0.6180	0.0178

are one of the few professionals who are skilled in detecting deception in the general population. However, only 12% of them can identify at most 80% of the deceivers. Therefore, both our maximum detection rate and mean detection rate are satisfactorily high compared with human detectors. The false-alarm rate is around 1%, which is also acceptably low.

In addition to validating the performance of the system based on Alarm networks, we further considered how the system performs using general BNs as testbeds. As such, we conducted the same experiment on several other BNs which are the *Hailfinder Network* [21], the *Diabetes Network* [22], and the *Munin Network* [23], with increasing number of nodes and increasing complexity of structure. Table II shows the detection rates together with each network's information.

Surprisingly, we observe from the table that the Diabetes network has the lowest detection rates, although its number of nodes, number of states, and number of arcs are not among the largest. By further studying the structure of the networks, we noticed that the height of the Diabetes network is more than 100 levels, while the other networks' heights are within 20 levels.

According to Yuan [3], detection rate is largely influenced by the network's intradependence. The intradependence index measures how dependent the states' probabilities are on the evidence. It can be calculated using [3]

$$I = \frac{\sum_{j=1}^{M} \sqrt{\sum_{i=1}^{N} (R_{i,j} - \overline{R}_i)^2}}{NM} \tag{4}$$

where $R_{i,j}$ denotes the posterior probability of random variable i in the jth test, $\overline{R_i}$ is the "neutral" value of random variable i, N is the number of variables in the network, and M is the total number of tests. The "neutral" value of a random variable is the average of all probabilities that the variable has obtained over all the test cases, which is calculated using

$$\overline{R}_i = \frac{\sum_{j=1}^{M} R_{i,j}}{M}. \tag{5}$$

Normally, the farther away a node is from the evidence, the less strongly it depends on the evidence. Since the nodes in the Diabetes network are highly separated from one another due to its larger height, we form the hypothesis that the nodes' dependence on the evidence is the weakest among all networks we tested on. To confirm our hypothesis, an experiment was conducted to measure the intradependence indexes of all the networks. Table III shows the test result. The result confirms our hypothesis that the Diabetes network has the lowest intradependence. Since detection rate is positively correlated to intradependence index, which means that the detection rate increases with the increase of the intradependence index; the low detection rate of Diabetes network is shown to be due to its great height. In conclusion, the detection method is valid on networks with moderate intradependences. If the height of the network is too large, then the network will be too weak to propagate the evidence to all the nodes, and thus, some deceptive information cannot be detected through reasoning.

TABLE III
INTRADEPENDENCE INDEX OF DIFFERENT NETWORKS

Network	Intra-Dependency Index
Alarm	0.023946267072262
Hailfinder	0.011272353508164927
Diabetes	0.001618689030060108
Munin	0.002419162273260489

In [3], parameters that influence the intradependence index were also studied. It demonstrates that the amount of evidence and the range of perturbation used in the multiagent experiments mainly determine the intradependence of the networks. This is due to the fact that the more evidence we possess, the more strongly the nodes depend on the evidence, but the dependence turns out to be weaker if the agents are perturbed more heavily. In addition to these two parameters, we showed that the structure of the network, specifically the height, also impacts the intradependence.

VI. EXPERIMENTS ON PARAMETER IMPACT

In our research, the goal is to evaluate the behavior of the deception-detection model more thoroughly by investigating what factors have an impact on the detection rate. Yuan [3] conducted a preliminary parametric study. The tested parameters include the number of agents used in the multiagent system, the perturbation value that determines the similarity between agents, the number of nodes that are set as evidence, and the number of repeats in each experiment. In addition to these parameters, we also focus on the level of standard deviations within which the difference between predicted and exact opinions can be accepted. Moreover, the amount of evidence has different impacts in the training and testing processes. Thus, we extended the parameters and conducted a more comprehensive experiment on all the testbeds. In our experiment, the following statistical data was calculated for analysis: Pearson-correlation value, standard deviation, positive detection rate, and false activation rate. For each item, we measured minimum, maximum, median, and average values. In this way, the impact of a parameter on various aspects of the system can be clearly recorded and then inspected. We now detail the results of our experiments for the Alarm network testbed.

1) *Results on the number of agents and the perturbation value:* First, we fixed the repeats, the amount of evidence, and the number of standard deviations while adjusting the perturbation values from ± 0.1 to ± 0.4 and the number

TABLE IV
DETECTION PERFORMANCE WITH THE NUMBER OF AGENTS. (a) MEANS OF PEARSON CORRELATION VALUES. (b) MEANS OF PREDICTION ERRORS STANDARD DEVIATION. (c) MEANS OF POSITIVE DETECTION RATE. (d) MEANS OF FALSE DETECTION RATE

(a)

Parameters	Repeats = 100, Evidence = 10, No. of stdevs = 4			
Pert.\ Agents	*3*	*10*	*30*	*100*
0.1	0.9030	0.9095	0.9022	0.9069
0.2	0.8144	0.8168	0.8187	0.8116
0.3	0.7595	0.7591	0.7669	0.7529
0.4	0.7175	0.6856	0.6713	0.7109

(b)

Parameters	Repeats = 100, Evidence = 10, No. of stdevs = 4			
Pert.\ Agents	*3*	*10*	*30*	*100*
0.1	0.0611	0.0562	0.0570	0.0568
0.2	0.0799	0.0741	0.0711	0.0729
0.3	0.0837	0.0807	0.0816	0.0821
0.4	0.0909	0.0887	0.0859	0.0858

(c)

Parameters	Repeats = 100, Evidence = 10, No. of stdevs = 4			
Pert.\ Agents	*3*	*10*	*30*	*100*
0.1	0.8585	0.8724	0.8835	0.8775
0.2	0.6996	0.6880	0.7455	0.7044
0.3	0.5946	0.6051	0.5691	0.5854
0.4	0.4808	0.4896	0.5229	0.5162

(d)

Parameters	Repeats = 100, Evidence = 10, No. of stdevs = 4			
Pert.\ Agents	*3*	*10*	*30*	*100*
0.1	0.0139	0.0163	0.0111	0.0122
0.2	0.0121	0.0085	0.0083	0.0103
0.3	0.0102	0.0108	0.0086	0.0088
0.4	0.0112	0.0086	0.0077	0.0116

TABLE V
DETECTION PERFORMANCE WITH THE NUMBER OF REPEATS. (a) MEANS OF PEARSON CORRELATION VALUES. (b) MEANS OF PREDICTION ERRORS STANDARD DEVIATION. (c) MEANS OF POSITIVE DETECTION RATE. (d) MEANS OF FALSE DETECTION RATE

(a)

Parameters	Agents=10, Evidence=10, No. of stdevs=4			
Pert.\ Repeats	*10*	*100*	*1000*	*10000*
0.1	0.8900	0.9091	0.9087	0.9026
0.2	0.8246	0.8234	0.8191	0.8208
0.3	0.6931	0.7550	0.7620	0.7466
0.4	0.6445	0.6674	0.6967	0.7048

(b)

Parameters	Agents=10, Evidence=10, No. of stdevs=4			
Pert.\ Repeats	*10*	*100*	*1000*	*10000*
0.1	0.0574	0.0546	0.0575	0.0567
0.2	0.0697	0.0713	0.0744	0.0727
0.3	0.0726	0.0824	0.0831	0.0833
0.4	0.0831	0.0901	0.0888	0.0869

(c)

Parameters	Agents=10, Evidence=10, No. of stdevs=4			
Pert.\ Repeats	*10*	*100*	*1000*	*10000*
0.1	0.9360	0.9022	0.8902	0.8923
0.2	0.7843	0.7433	0.7455	0.7470
0.3	0.7285	0.6261	0.6396	0.6370
0.4	0.6088	0.5878	0.5692	0.5792

(d)

Parameters	Agents=10, Evidence=10, No. of stdevs=4			
Pert.\ Repeats	*10*	*100*	*1000*	*10000*
0.1	0.0572	0.0185	0.0170	0.0148
0.2	0.0725	0.0193	0.0155	0.0170
0.3	0.1079	0.0115	0.0149	0.0159
0.4	0.0772	0.0258	0.0181	0.0197

of agents from 3 to 100. Since the detection method is based on the assumption that agents are highly correlated, by changing the perturbation value we can observe how sensitive the system is to this assumption under different environmental settings. Therefore, we will adjust the perturbation value while also adjusting the target parameter in each of the following experiments. Table IV(a) shows the means of Pearson-correlation values of all states. As we can see, the Pearson-correlation values are only determined by perturbation values. This is because the more heavily we perturb the agents, the less correlated the agents are. Table IV(b) shows the means of the standard deviations of the prediction error. It seems that the standard deviation has a slightly negative correlation with the number of agents. This can be explained by the fact that having more agents increases the number of correlation values for each agent and thus, increases the precision of predicting opinions. On the contrary, the perturbation value has a significant influence on the standard deviation because the less correlated the agents are, the more difficult it is to predict their opinions. Table IV(c) displays the means of positive detection rates. The number of agents still does not seem to have a strong impact on the detection rate, but the perturbation value does because the more correlated the agents are, the more obvious the inconsistency appears to be. From Table IV(d), it can be seen that only perturbation has a slight influence on the false detection rate. Since a high correlation leads to a high detection rate, it will also cause a high false-alarm rate.

2) *Results on the number of repeats:* Next, we fixed the number of agents, the amount of evidence, and the number of standard deviations but adjusted the repeats. Table V shows the experiment results. The results demonstrate that the number of repeats slightly influences the positive and false detection rates because the more questions that are asked, the easier for the deceiver to expose weakness and thus, less demanding to detect deception.
3) *Results on the amount of evidence in the testing process:* We proposed that evidence in the training process and in the testing process have a different impact on the performance. Thus, we first evaluated the impact of evidence on the test data. Since deception only occurs in the testing process, our hypothesis is that the more evidence is available, the higher is the detection rate that the system will achieve. The hypothesis can be explained intuitively by the fact that the more information we have about the environment, the easier for us to identify any

TABLE VI
DETECTION PERFORMANCE WITH THE NUMBER OF PIECES OF EVIDENCE IN THE TESTING PROCESS. (a) MEANS OF PEARSON CORRELATION VALUES. (b) MEANS OF PREDICTION ERRORS STANDARD DEVIATION. (c) MEANS OF POSITIVE DETECTION RATE. (d) MEANS OF FALSE DETECTION RATE

(a)

Parameters	Repeats = 100, Agents = 10, Training Evidence = 1-5, No. of stdevs = 4						
Pert.\ Test. evi.	*1-5*	*6-10*	*11-15*	*16-20*	*21-25*	*26-30*	*31-35*
0.1	0.9055	0.8943	0.8982	0.8912	0.9051	0.9050	0.8983
0.2	0.8251	0.8124	0.8039	0.8244	0.8124	0.8159	0.8189
0.3	0.7568	0.7477	0.7473	0.7444	0.7268	0.7473	0.7284
0.4	0.6890	0.6882	0.6888	0.6754	0.6776	0.6519	0.6785

(b)

Parameters	Repeats = 100, Agents = 10, Training Evidence = 1-5, No. of stdevs = 4						
Pert.\ Test. evi.	*1-5*	*6-10*	*11-15*	*16-20*	*21-25*	*26-30*	*31-35*
0.1	0.0546	0.0517	0.0545	0.0543	0.0554	0.0522	0.0539
0.2	0.0704	0.0720	0.0712	0.0701	0.0677	0.0678	0.0691
0.3	0.0780	0.0800	0.0788	0.0782	0.0796	0.0777	0.0761
0.4	0.0824	0.0816	0.0824	0.0849	0.0829	0.0827	0.0814

(c)

Parameters	Repeats = 100, Agents = 10, Training Evidence = 1-5, No. of stdevs = 4						
Pert.\ Test. evi.	*1-5*	*6-10*	*11-15*	*16-20*	*21-25*	*26-30*	*31-35*
0.1	0.8837	0.9171	0.9576	0.9617	0.9749	0.9852	0.9915
0.2	0.7253	0.77054	0.87145	0.9080	0.9438	0.9680	0.9865
0.3	0.6107	0.6679	0.8044	0.8555	0.9077	0.9520	0.9804
0.4	0.5628	0.6172	0.7518	0.8116	0.8779	0.9344	0.9656

(d)

Parameters	Repeats = 100, Agents = 10, Training Evidence = 1-5, No. of stdevs = 4						
Pert.\ Test. evi.	*1-5*	*6-10*	*11-15*	*16-20*	*21-25*	*26-30*	*31-35*
0.1	0.0172	0.0377	0.0573	0.0908	0.0878	0.0899	0.0584
0.2	0.0171	0.0266	0.0601	0.0917	0.1164	0.1024	0.1118
0.3	0.0144	0.0282	0.0762	0.0833	0.1118	0.1242	0.1039
0.4	0.0175	0.0329	0.0866	0.1334	0.1322	0.1445	0.1531

abnormal phenomenon. The results in Table VI support our hypothesis.

4) *Results on the amount of evidence in the training process:* We next fixed the amount of evidence in the testing process but adjusted it in the training process. Table VII shows that in contrast to the impact of evidence in the testing process, the lowest detection rate does not co-occur with the least amount of evidence in the training process, but with six to ten pieces of evidence. This may be because six to ten pieces of evidence is the crossover point around which the prediction will produce the most variable error. Crossover point is a terminology used in three satisfiability (3-SAT) problems [24]. Normally, 3-SAT problems with a large number of constraints and a small number of constraints are easy to solve. However, the problems with the number of constraints in between appear to be much harder. This critical number of constraints is called the crossover point in 3-SAT problems. Likewise, we also found the critical number of pieces of evidence that determines the standard deviation of the prediction error in the Alarm network. If we provide a small amount of evidence, the prediction is very hard, and thus, the prediction errors over the states are always very large. While given a large amount of evidence, the prediction errors over all states will become small. However, with an amount of evidence in between, prediction over some states is precise but over others is not, which results in a large standard deviation. Because of this unstable prediction, the normal decision error cannot be determined easily, and, thus, detection in the testing process turns out to be imprecise.

This finding leads us to the question of whether the crossover point exists in BNs in general. Therefore, we performed the same test on the other three networks. We used ten agents, 30% of all nodes as evidence in the testing process, and four standard deviations on all networks while adjusting the amount of evidence in the training process from 10% to 90% of the total nodes. The result is shown in Fig. 2, from which we can see that although located slightly differently, there is a crossover point in each network. For example, the crossover point of the Diabetes network is around 40% while that of Munin network appears at 20%. In general, the locations of crossover points float between 20% and 50%.

5) *Results on the number of standard deviations:* Lastly, we tested the number of standard deviations by fixing the number of agents, repeats, and the amount of evidence. The results shown in Table VIII indicate that if we relax the number of standard deviations, we will get fewer positive and negative alarms. This is very intuitive to understand since the more forgiving we are, the fewer inconsistencies we will care about.

From Tables IV–VIII, we can also see that when the perturbation value is kept below 0.2, the detection rate is always above 60% (higher than human detection rate), but when the opinions are perturbed by 0.3 to 0.4, the detection rate strongly depends on other parameters. Therefore, to ensure a good detection

TABLE VII
DETECTION PERFORMANCE WITH THE NUMBER OF PIECES OF EVIDENCE IN THE TRAINING PROCESS. (a) MEANS OF PEARSON CORRELATION VALUES. (b) MEANS OF PREDICTION ERRORS STANDARD DEVIATION. (c) MEANS OF POSITIVE DETECTION RATE. (d) MEANS OF FALSE DETECTION RATE

(a)

Parameters	Repeats = 100, Agents = 10, Testing Evidence = 1-5, No. of stdevs = 4						
Pert.\ Test. evi.	*1-5*	*6-10*	*11-15*	*16-20*	*21-25*	*26-30*	*31-35*
0.1	0.8951	0.9199	0.9096	0.9056	0.9065	0.9141	0.8914
0.2	0.8068	0.8382	0.8443	0.8549	0.8518	0.8594	0.8349
0.3	0.7317	0.7612	0.7745	0.7979	0.7931	0.8008	0.7538
0.4	0.6618	0.7078	0.7468	0.7315	0.7411	0.7564	0.7323

(b)

Parameters	Repeats = 100, Agents = 10, Testing Evidence = 1-5, No. of stdevs = 4						
Pert.\ Test. evi.	*1-5*	*6-10*	*11-15*	*16-20*	*21-25*	*26-30*	*31-35*
0.1	0.0509	0.0597	0.0566	0.0530	0.0459	0.0351	0.0201
0.2	0.0706	0.0752	0.0711	0.0629	0.0554	0.0427	0.0273
0.3	0.0775	0.0858	0.0804	0.0717	0.0622	0.0486	0.0293
0.4	0.0810	0.0895	0.0854	0.0775	0.0655	0.0536	0.0330

(c)

Parameters	Repeats = 100, Agents = 10, Testing Evidence = 1-5, No. of stdevs = 4						
Pert.\ Test. evi.	*1-5*	*6-10*	*11-15*	*16-20*	*21-25*	*26-30*	*31-35*
0.1	0.8964	0.8565	0.8758	0.8944	0.9092	0.9459	0.9858
0.2	0.7356	0.6718	0.6695	0.7555	0.7823	0.8579	0.9585
0.3	0.6305	0.5620	0.5754	0.6029	0.6503	0.7482	0.8974
0.4	0.5402	0.5082	0.5141	0.5427	0.5605	0.6920	0.8189

(d)

Parameters	Repeats = 100, Agents = 10, Testing Evidence = 1-5, No. of stdevs = 4						
Pert.\ Test. evi.	*1-5*	*6-10*	*11-15*	*16-20*	*21-25*	*26-30*	*31-35*
0.1	0.0172	0.0377	0.0573	0.0908	0.0878	0.0899	0.0584
0.2	0.0171	0.0266	0.0601	0.0917	0.1164	0.1024	0.1118
0.3	0.0144	0.0282	0.0762	0.0833	0.1118	0.1242	0.1039
0.4	0.0175	0.0329	0.0866	0.1334	0.1322	0.1445	0.1531

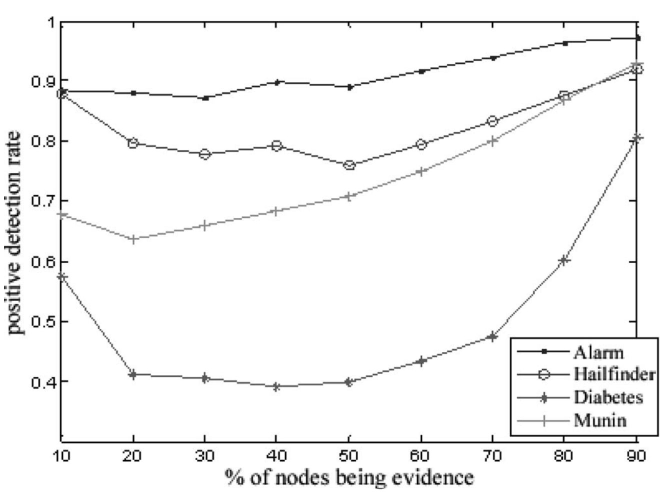

Fig. 2. Plot of positive detection rate against proportion of nodes being training evidence for Alarm network, Hailfinder network, Diabetes network, and Munin network.

performance which is robust to environmental change, it is necessary to assume that agents are highly correlated in giving opinions.

In order to get a more concrete idea of the differences caused by each parameter, we carried out a statistical significance test called analysis of variance (ANOVA). ANOVA is used to test the differences between two or more groups. We applied one-way ANOVA on the positive detection rates from tests performed on individual parameters to test the null hypothesis that the detection rates generated by using different values of a parameter are equal. The result of an ANOVA is an F-critical-value and an F-value. If the F-value is higher than the F-critical-value, then the null hypothesis is rejected. Table IX displays the ANOVA of the aforementioned six parameters.

The ANOVA shows that four out of six parameters: perturbation value, amount of training evidence, amount of testing evidence, and number of standard deviations significantly influence the detection rate. The other two parameters which are the number of agents and the number of repeats only slightly impact it. The results are consistent with our explanations of the parametric experiments.

1) The perturbation value determines how similar and how correlated the agents are with each other. The deceiving agent's abnormal opinion will be more distinct if the benevolent agents always agree or disagree with each other than if the benevolent agents have no clue about how the other agents will conclude.
2) The amount of evidence in the testing process indicates how much information we know in the current tasks. The more we know about the problem, the easier to detect if anyone is deceiving.
3) The amount of evidence in the training process indicates how much information we know in past tasks. If we have much information or little information in the past, we are quite sure about the normal decision error, which results in easier detection in the future. However, if we have learned 20% to 50% of the facts in the past, the normal

TABLE VIII
DETECTION PERFORMANCE WITH THE NUMBER OF STANDARD DEVIATION. (a) MEANS OF PEARSON CORRELATION VALUES. (b) MEANS OF PREDICTION ERRORS STANDARD DEVIATION. (c) MEANS OF POSITIVE DETECTION RATE. (d) MEANS OF FALSE DETECTION RATE

(a)

Parameters	Agents=10, Repeats=100, Evidence=10			
Pert.\ No. of stdevs	*4*	*3*	*2*	*1*
0.1	0.9342	0.9354	0.9362	0.9384
0.2	0.8703	0.8750	0.8661	0.8701
0.3	0.8105	0.8333	0.8278	0.8377
0.4	0.7926	0.7908	0.7940	0.7996

(b)

Parameters	Agents=10, Repeats=100, Evidence=10			
Pert.\ No. of stdevs	*4*	*3*	*2*	*1*
0.1	0.0630	0.0609	0.0653	0.0593
0.2	0.0822	0.0801	0.0820	0.0825
0.3	0.0920	0.0933	0.0942	0.0906
0.4	0.0986	0.0998	0.0985	0.0987

(c)

Parameters	Agents=10, Repeats=100, Evidence=10			
Pert.\ No. of stdevs	*4*	*3*	*2*	*1*
0.1	0.8496	0.9037	0.9383	0.9885
0.2	0.6403	0.7748	0.8529	0.9556
0.3	0.5195	0.6163	0.7445	0.9181
0.4	0.4349	0.5327	0.6858	0.8591

(d)

Parameters	Agents=10, Repeats=100, Evidence=10			
Pert\ No. of stdevs	*4*	*3*	*2*	*1*
0.1	0.0087	0.0213	0.0369	0.2127
0.2	0.0067	0.0181	0.0386	0.1657
0.3	0.0060	0.0096	0.0329	0.1668
0.4	0.0039	0.0111	0.0372	0.1450

TABLE IX
ANOVA OF PARAMETER IMPACT

Parameter	*Pert.*	*Agent*	*Repeat*
F value	3575.061	6.581	16.677
F Critical	2.683	3.101	2.683
P-level	<0.0001	0.002	4.490
Significant	Yes	Slight	Slight
Parameter	*Train. evi.*	*Test. evi.*	*No. of stdevs*
F value	527.163	493.583	1689.460
F Critical	3.101	3.101	2.683
P-level	<0.0001	<0.0001	<0.0001
Significant	Yes	Yes	Yes

decision error will be so variable that we are not confident enough to identify deceivers.

4) The number of standard deviations determines how much error between the actual and the predicted opinions we accept as a normal decision error. Normally the more forgiving we are, the larger error we can accept, and thus, the fewer deceivers can be caught no matter whether it is a positive detection or false activation.

To test the robustness of the model, we conducted the complete parametric experiment on other networks including Hailfinder network, Diabetes network, and Munin network. The

TABLE X
STATISTICS ON THE DETECTION RATES OF ALARM NETWORK

Parameters	Agents = 10, Repeats = 1000, Perturbation = 0.1, Evidence = 1-10, No. of stdevs = 4				
Positive Detection Rate	Max	1.0	False Detection Rate	Max	0.3349
	Min	0.2267		Min	0.0
	Mean	0.8734		Mean	0.0116
	Med	0.9427		Med	0.0035

TABLE XI
ANOVA ON THE DIFFERENCE BETWEEN DISINFORMATION AND MISINFORMATION

Parameter	Conditional / Posterior
F value	0.9209
F Critical	4.0069
P-level	0.3412
Significant	No

result shows that although the detection rates vary from network to network, the influence of the parameters are basically the same. This means that the methodology is robust to different structures and sizes of BNs as long as the network is ensured to have a moderate intradependence.

To summarize, the effectiveness in capturing deception is determined by how correlated the parties' knowledge is with each other, how much information is available in both the past experience and the current tasks, and how forgiving we are about mistakes.

VII. ON MISINFORMATION AND MULTIPLE DECEIVERS

The motivation in providing wrong information may be intentional or unintentional. The deception we intend to capture is intentional disinformation. Different from disinformation, misinformation is defined as mistakenly providing the wrong information. It is very hard to distinguish disinformation and misinformation because their effects are very similar. However, disinformation will probably bring more severe and long-term damage to the receiver, while misinformation can be corrected shortly and is not likely to happen frequently. In this paper, we present our initial extension of the Santos and Johnson's approach to misinformation detection. To simulate the features of misinformation, we first examine the features of disinformation as defined by Burgoon [5].

1) *The information is false from the sender's point of view.*
2) *The act is intentional.*
3) *The purpose is to take advantage.*

These features clearly differentiate disinformation from misinformation. It emphasizes that intent is the main factor in deception. Since our model focuses on modeling the human reasoning process rather than capturing human intent, we simulate misinformation in the way that the experts may misunderstand the information as true. If the information is true in the expert's mind, then his inherent knowledge, which is represented by the BN, contains the wrong information. Since the agents differ in their conditional probabilities, instead of rotating the posterior probabilities, we rotate the conditional probabilities in the CPT to create misinformation. Tables X and XI show the positive and

TABLE XII
MEANS OF DETECTION RATE OF ADJUSTING THE NUMBER OF AGENTS TOGETHER WITH THAT OF DECEIVERS

No. of agents\proportion of agents being deceiver	10%	30%	50%	70%	90%
3	NA	0.8538	NA	0.6642	NA
10	0.8728	0.8362	0.6783	**0.4680**	**0.1935**
30	0.8654	0.8045	0.6979	**0.4880**	**0.1815**
100	0.8502	0.7864	0.6668	**0.4515**	**0.1396**

false detection rates of this evaluation and the ANOVA testing whether positive detection rate of disinformation and that of misinformation are significantly different.

The result from Table X shows that we still have a high positive detection rate (87%) and an acceptably low false activation rate (1%) in identifying misinformation. After comparing the results in capturing disinformation with those in capturing misinformation using ANOVA, we find that the results are surprisingly similar. The test validates the null hypothesis that their detection rates are equal. As such, the model seems to perform equally well in detecting disinformation and misinformation.

The methodology we propose to distinguish between disinformation and misinformation is to reason back from the agent's opinion after activation has launched. Since the reasoning process of each agent is available, an agent's opinion can be explained by extending the inference back from the opinion to the hypothesis, and the explanation is expected to be consistent with the known evidence. In particular, first assume that after a candidate deceiver has been identified, we suspect that his opinion on random variable A is wrong. Next, we set the states of A as evidence, each one at a time, and reason back toward the original evidence. We assert that if the posterior probability of a state of A in his original opinion is large, we would also expect most of the original evidence in the resultant reasoning, and vice versa. If this is confirmed, it implies that the agent is correct in his reasoning but wrong in terms of his inherent knowledge. Otherwise, it implies that the agent is aware that his opinion is wrong with respect to his knowledge. Yet, he intentionally submits the wrong opinion. This implementation will be evaluated in the near future.

Finally, up to this point, all the experiments we conducted contained only one deceiver no matter how many agents are in the group. However in reality, we may face the situation that more than one deceiver is working or even cooperating together to mislead the decision maker. Taking this into consideration, we studied the performance of the model in detecting multiple deceivers. In this experiment, we adjusted the proportion of agents being deceivers while changing the total number of agents at the same time. The positive detection rates of the experiment are shown in Table XII.

As we can see from Table XII, when half or more of the experts are honest, the detection rates are above 67%, which is still relatively high. However, as soon as the majority of the experts become deceivers, our detection rates drop rapidly. This is intuitive since in real life, if the majority of people are lying, it is hard for the listener to distinguish out the truth. Fig. 3 shows the plotted detection rate against the proportion

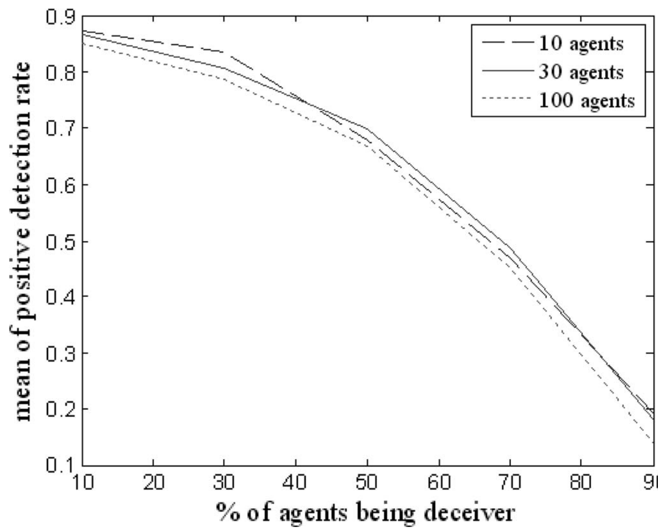

Fig. 3. Plot of detection rate against the proportion of agents being deceivers.

of agents being deceivers. The three lines represent systems with different numbers of agents. We observe from the figure that the detection rate is inversely proportional to both the proportion of agents being deceivers and the total number of agents. However, the impact from the number of agents is relatively small. Therefore, it is more critical to make sure that the proportion of benevolent agents is high rather than to have a large number of benevolent agents for the purpose of detecting deception successfully.

VIII. CONCLUSION AND FUTURE WORK

Catching deception from different parties with common or conflicting interests is important but challenging. In this paper, we have introduced a deception-detection model using a multiagent system framework. This model makes reasonable predictions on agents' opinions based on their relations with others. Then, it evaluates whether the agents' actual opinions are consistent with predicted ones. We first reevaluated the performance of the model from earlier work [1] and then tested the model using new testbeds. We showed that the model can achieve a mean detection rate ranging from 63% to 87% if the BN testbed has a moderate intradependence index [3]. This performance is significantly better than human face-to-face detection. However, if a network is of large height, which results in a small intradependence index, the detection rate will severely decrease. Next, we extended the parametric study conducted in [3]. We found out the following: 1) If the agents' opinions are more correlated to each other, the deceiver will be more distinguishable; 2) if we have more information about the environment, it is easier to identify any inconsistent opinion; 3) if we had little or much information about the environment in the past, we will be more confident in determining how much deviation from the expected opinion is considered to be normal; and 4) the more receptive we are of diverse opinions, the less likely we are to be suspicious about inconsistent opinions.

Different from disinformation, misinformation is providing wrong information unintentionally. We investigated the system's performance on misinformation detection and found that the detection rate is similar to that of disinformation. We proposed that to distinguish between them, we need to reason back through the network from the suspect opinion. If his opinion is consistent with the amount of evidence that can be inferred back, then he is only guilty of misinformation. In our future work, we will incorporate this method within our detection model.

Our last study was focused on simulating multiple deceivers. The test demonstrates the effectiveness of the system when more than half of the agents are benevolent and suggests that the proportion of deceivers in the agents is more important than the exact number of deceivers in improving the detection performance.

Although the effectiveness of our deception-detection method has been verified, there are still several shortcomings. First, the simulation of the experts' knowledge is still not realistic enough. In order to evaluate the performance of the model, we simply simulate all experts using the same network structure. The variance of knowledge is only represented by some noise in the conditional probabilities. However, in reality, the levels of knowledge of different experts may not be the same. Some experts may be more authoritative, while others may not specialize in the task domain. Thus, to simulate this in a more realistic manner, the structure of the network should also be altered for different experts. Likewise, we should also use a threshold to control the similarity between the agents.

Another concern lies in the simple way we simulate deceptions. Currently, we simulate deceptions by rotating the posterior probability of each state. In reality, deceivers are honest in most of their story in order to convince the listener. The strategies they take can be categorized into simulative deception (creating false) and dissimulative deceptions (hiding truth) [25]. Simulative deception is further divided into mimicking, inventing, and decoying. On the other hand, dissimulative deception is separated into masking, repackaging, and dazzling [26]. Therefore, instead of rotating all posterior probabilities, we will need to simulate different kinds of deception strategies. For example, simulative deception can be simulated by inserting nodes and dissimulative deception by removing nodes.

Finally, the Santos and Johnson's model [1] focuses on the activation stage of deception detection. After the activation, we must proceed to categorize the suspected deceptions into one of the six categories aforementioned. The categorization of deception is important to detectors because each kind of deception has its unique way of reasoning, and their different natures will determine the observables we can obtain, and thus, may influence the detection strategy.

Acknowledgment

An earlier version of this paper can be found in [27].

References

[1] E. Santos, Jr. and G. Johnson, "Towards detecting deception in intelligent systems," in *Proc. SPIE: Defense Security Symp.*, Orlando, FL, 2004, vol. 5423, pp. 131–140.

[2] G. Johnson and E. Santos, Jr., "Deception detection in information systems. I: Activation of deception detection tactics," in *Proc. AI*, London, ON, Canada, 2004, pp. 339–354.

[3] X. Q. Yuan, "Deception detection in multi-agent systems and wargaming," M.S. thesis, Thayer School Eng., Dartmouth College, Hanover, NH, 2007.

[4] B. Whaley, "Toward a general theory of deception," in *Military Deception and Strategic Surprise*, J. Gooch and A. Perlmutter, Eds. London, U.K.: Frank Cass, 1982.

[5] J. Burgoon and D. Buller, "Interpersonal deception: III. Effects of deceit on perceived communication and nonverbal behavior dynamics," *J. Nonverbal Behav.*, vol. 18, no. 2, pp. 155–184, Jun. 1994.

[6] J. F. George and J. R. Carlson, "Group support systems and deceptive communication," in *Proc. 32nd Hawaii Int. Conf. Syst. Sci.*, Maui, HI, Jan. 1999, vol. 1, p. 1038.

[7] S. Grazioli and S. L. Jarvenpaa, "Perils of Internet fraud: An empirical investigation of deception and trust with experienced Internet consumers," *IEEE Trans. Syst., Man, Cybern. A, Syst., Humans*, vol. 30, no. 4, pp. 395–410, Jul. 2000.

[8] P. E. Johnson, S. Grazioli, K. Jamal, and R. G. Berryman, "Detecting deception: Adversarial problem solving in a low base-rate world," *Cogn. Sci.*, vol. 25, no. 3, pp. 355–392, Jun. 2001.

[9] J. Pearl, *Probabilistic Reasoning in Intelligent Systems*. San Francisco, CA: Morgan Kaufmann, 1988.

[10] M. Schillo, P. Funk, and M. Rovatsos, "Using trust for detecting deceitful agents in artificial societies," *Appl. Artif. Intell.*, vol. 14, no. 8, pp. 825–848, Sep. 2000.

[11] A. Vyas and L. Zhou, "On detecting deception in agent societies," in *Proc. IEEE/WIC/ACM Int. Conf. IAT*, 2005, pp. 491–494.

[12] N. C. Rowe, "Automatic detection of fake file systems," in *Proc. Int. Conf. Intell. Anal. Methods Tools*, May 2005.

[13] G. A. Wang, H. Chen, J. J. Xu, and H. Atabakhsh, "Automatically detecting criminal identity deception: An adaptive detection algorithm," *IEEE Trans. Syst., Man, Cybern. A, Syst., Humans*, vol. 36, no. 5, pp. 988–999, Sep. 2006.

[14] F. J. Stech and C. Elässer, "Midway revisited: Deception by analysis of competing hypothesis," MITRE Corp., Bedford, MA, Tech. Rep., 2004.

[15] E. Santos, Jr., "On the generation of alternative explanations with implications for belief revision," in *Proc. 7th Conf. Uncertainty Artif. Intell.*, 1994, pp. 339–347.

[16] P. Resnick, N. Iacovou, M. Suchak, P. Bergstrom, and J. Riedl, "GroupLens: An open architecture for collaborative filtering of Netnews, in *Proc. ACM Conf. Comput. Supported Cooperative Work*, 1994, pp. 175–186.

[17] I. Beinlich, G. Suermondt, R. Chavez, and G. Cooper, "The alarm monitoring system," in *Proc. 2nd Eur. Conf. AI Med.*, J. Hunter, Ed., 1989, pp. 247–256.

[18] M. G. Millar and K. U. Millar, "The effects of cognitive capacity and suspicion on truth bias," *Commun. Res.*, vol. 24, no. 5, pp. 556–570, Oct. 1997.

[19] B. M. Depaulo, S. E. Kirkendol, J. Tang, and T. P. O'Brien, "The motivational impairment effect in the communication of deception: Replications and extensions," *J. Nonverbal Behav.*, vol. 12, no. 3, pp. 177–202, Sep. 1988.

[20] C. V. Ford, *Lies! Lies! Lies! The Psychology of Deceit*. Washington, DC: Amer. Psychiatric Press, 1996.

[21] B. Abramson, J. Brown, W. Edwards, A. Murphy, and R. L. Winkler, "Hailfinder: A Bayesian system for forecasting severe weather," *Int. J. Forecast.*, vol. 12, no. 1, pp. 57–71, Mar. 1996.

[22] S. Andreassen, R. Hovorka, J. Benn, K. G. Olesen, and E. R. Carson, "A model-based approach to insulin adjustment," in *Proc. 3rd Conf. Artif. Intell. Med.*, M. Stefanelli, A. Hasman, M. Fieschi, and J. Talmon, Eds., 1991, pp. 239–248.

[23] S. Andreassen, F. V. Jensen, S. K. Andersen, B. Falck, U. Kjærulff, M. Woldbye, A. R. Sørensen, A. Rosenfalck, and F. Jensen, "MUNIN—An expert EMG assistant," in *Computer-Aided Electromyography and Expert Systems*, vol. 2, J. E. Desmedt, Ed. Amsterdam, The Netherlands: Elsevier, 1989, pp. 255–277.

[24] J. M. Crawford and L. D. Auton, "Experimental results on the crossover point in random 3-SAT," *Artif. Intell.*, vol. 81, no. 1/2, pp. 31–57, Mar. 1996.

[25] J. B. Bell and B. Whaley, *Cheating and Deception*. Piscataway, NJ: Trans. Publ., 1991.

[26] J. B. Bowyer, *Cheating*. New York: St. Martin's Press, 1982.

[27] E. Santos, Jr., D. Li, and X. Yuan, "On deception detection in multi-agent systems and deception intent," *Proc. SPIE*, vol. 6965, pp. 696 502-1–696 502-11, Mar. 2008.

Eugene Santos, Jr. (M'93–SM'04) received the B.S. degree in mathematics and computer science and the M.S. degree in mathematics (specializing in numerical analysis) from Youngstown State University, Youngstown, OH, in 1985 and 1986, respectively, and the M.Sc. and Ph.D. degrees in computer science from Brown University, Providence, RI, in 1988 and 1992, respectively.

He was with the Department of Electrical and Computer Engineering, Air Force Institute of Technology, Wright-Patterson AFB, OH. He was also a Professor with the School of Engineering, University of Connecticut (UConn), Storrs, where during his time at UConn, he also served as Interim Director with the Booth Engineering Center for Advanced Technology—an interdisciplinary center supporting computer and communications research and education. He is currently a Professor of engineering with the Thayer School of Engineering, Dartmouth College, Hanover, NH and the Director of the Distributed Information and Intelligence Analysis Group ($\mathrm{DI^2AG}$). His research interest specializes in modern statistical and probabilistic methods with applications to intelligent systems, uncertain reasoning, and decision science. He is a leader in the growing community of artificial intelligence researchers who are advancing techniques for constructing models and software systems that can effectively and efficiently reason in the presence of noise and uncertainty. Most recently, he has pioneered new research on user and adversarial behavioral modeling and intent inferencing. Outside of academia, he is a leader, adviser, and active participant in defense-related working groups on decision science and information technology. He has over 100 refereed technical publications.

Prof. Santos is currently the Editor-in-Chief of the IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS: PART B, an Associate Editor of the *International Journal of Image and Graphics*, and is also on the editorial advisor board for *System and Information Sciences Notes* and on the editorial boards for the *Journal of Intelligent Information Systems* and *Journal of Experimental and Theoretical Artificial Intelligence*. He has chaired or served on numerous major conferences and professional society meetings.

Deqing Li (M'07) received the B.S. degree in electronic and information engineering from the Hong Kong Polytechnic University, Kowloon, Hong Kong. She is currently working toward the Ph.D. degree in computer engineering at the Thayer School of Engineering, Dartmouth College, Hanover, NH.

She has published papers in the *Proceedings of SPIE* and *Proceedings of the IEEE Conference on Systems Man and Cybernetics*, and a book chapter in the *E-Government Diffusion, Policy, and Impact: Advanced Issues and Practices*. Her research interest is in decision theory, intent modeling, and related applications.