Trust and Independence Aware Decision Fusion in Distributed Networks

Xinlei (Oscar) Wang†, Jin-Hee Cho‡, Kevin Chan†, MoonJeong Chang†, Ananthram Swami† and Prasant Mohapatra‡

†Computer Science Department, UC Davis
‡Computational and Information Sciences Directorate, U.S. Army Research Laboratory

Abstract—In distributed network environments, decisions must often be made based on incomplete or uncertain evidence whose sources may be dependent. Properly fusing potentially unreliable and dependent information from multiple sources is critical to effective decision making. The Transferable Belief Model (TBM), an extension of Dempster-Shafer Theory (DST), is a well known information fusion framework that can cope with conflicting evidence. However, neither DST nor TBM deals with misbehaving data sources and dependence of fusion data, which are often observed in dynamic multi-hop network environments. In this work, we propose a decision fusion framework that considers multi-dimensional trust and independence of information using a provenance technique, to enhance fusion reliability. We consider three information trust dimensions: correctness, completeness, and timeliness. Our simulation results show that the proposed framework yields a higher correct decision ratio, compared with the baseline counterparts.

Index Terms—Information trust, provenance, information fusion, decision making.

I. INTRODUCTION

In dynamic and distributed information sharing networks (e.g., tactical networks, sensor networks, vehicular networks), information is generated, shared and processed by different entities in the network. Entities need to have correct perception of the conditions in the network, in order to make right decisions. Let us consider the following multi-hop network scenario: a set of mobile or stationary nodes trying to monitor certain targets (objects, people, events, environmental factors, etc.) in the operational area; nodes share their observation data with their neighbors; multiple pieces of data which may have gone through a sequence of entities are finally received by a decision maker. It is crucial for the decision maker to be able to combine the evidence accurately to arrive at a correct perception about the tactical environment. The problem is compounded by the misbehaving nodes who supply false data. This work aims at enhancing the accuracy of the information fusion and thus the overall reliability of decision-making in such a hostile dynamic network environment.

Information fusion techniques have been studied extensively [1], [2]. Most of the techniques developed models for a cooperative environment where all the information being aggregated are reliable. In a network where entities may supply incorrect data, this leads to inaccurate decision making due to the use of untrustworthy evidence. Therefore, assessing the trustworthiness of information becomes important. In the field of information trust research, Raya et al. [3] studied data-centric trust to deal with hostile entities in ad hoc networks. Wang et al. [4] proposed information trust frameworks based on provenance techniques. Arunkumar et. al. [5] built a trust assessment framework between the “observe” and “orient” phases of multi-source decision making. However, these works only measure the “correctness” of information. In addition to correctness, other properties of information may affect its trustworthiness. First, information may become incomplete due to entities’ lack of capability or unwillingness to provide complete information, or information loss in the network. Moreover, timeliness of information often has a huge impact on decision making. A trusted piece of information may become untrusted as time passes because the target attribute may have changed. Bisidikian et. al. [6] and Bar-Noy et. al. [7] have advocated the need for multi-dimensional information quality metrics. They also emphasized the importance of provenance for information quality assessment. Our paper proposes a provenance model in detail and specifies how the properties of a multi-dimensional information trust, embracing correctness, completeness and timeliness, can be captured based on the provenance model.

In our targeted network environment, uncertainty in the information is often introduced by the following reasons: (1) a direct observer may not be able to observe a target accurately; (2) received information may contain untrustworthy content; and (3) information from different sources may be conflicting. In order to deal with uncertainty, we adopt the Transferable Belief Model (TBM) [8], an extension of Dempster-Shafer Theory (DST) [9], as the underlying information fusion framework. DST is a well known algorithm to deal with uncertain and incomplete information for data fusion [2], [10]. TBM is more robust in the presence of highly conflicting information than the original DST [8], [10]. However, neither scheme is able to correctly fuse dependent information. In multi-hop networks, multiple pieces of information may often go through the same set of nodes, thus leading to dependence among information. To tackle this problem, we introduce independence-awareness based on analyzing the overlapping provenance between information items; this enhances the robustness of information fusion under the scenario that an attacker or multiple colluding attackers provide a large amount of similar false information.

The contributions of this work are summarized as follows: First, a detailed provenance model is proposed and a multi-dimensional information trust metric is developed based on the provenance model to capture correctness, timeliness, and completeness of information. Second, a trust-aware and independence-aware decision fusion protocol is designed on top of the existing TBM framework. Third, node-level trust is maintained and updated at each node in order to facilitate information-level trust assessment. A dynamic node trust update algorithm is presented. Lastly, our simulation results show that the proposed framework outperforms counterparts that do not consider trust and/or independence in terms of decision accuracy.

The rest of the paper is organized as follows: Section II introduces our system model, adversary model and details our proposed provenance model. Section III discusses our multi-
dimensional information trust model. Section IV explains the information fusion framework and decision making protocols. Section V presents the node trust update algorithm. Section VI provides our experimental results and physical interpretation of the results. Section VII concludes our paper and suggests future work.

II. SYSTEM MODEL

A. Preliminaries

We consider a heterogeneous network consisting of a set of nodes, \( \{v_1, v_2, \ldots, v_N\} \), which can be stationary sensors, human or vehicles carrying devices/sensors, etc. Nodes that observed a relevant target will generate a report which is a description about an attribute of the target. Table I shows two exemplary reports generated by a source node based on its observation of a vehicle. The two reports illustrate the target vehicle’s type and location.

\[
\begin{array}{|c|c|}
\hline
\text{Vehicle type} & \text{BBA} \\
\hline
\text{Tank (T)} & 0.4 \\
\text{Armored car (AC)} & 0.2 \\
\text{Utility vehicle (UV)} & 0.1 \\
\text{T or AC } \{T, AC\} & 0.2 \\
\text{T or UV } \{T, UV\} & 0 \\
\text{AC or UV } \{AC, UV\} & 0 \\
\text{Ignorance } \{T, AC, UV\} & 0.1 \\
\text{Null (0)} & 0 \\
\hline
\end{array}
\]

Table I: EXAMPLES OF REPORTS

\[
\begin{array}{|c|c|}
\hline
\text{Location} & \text{BBA} \\
\hline
\text{District 1 (D1)} & 0.8 \\
\text{District 2 (D2)} & 0.2 \\
\text{Ignorance } \{D1, D2\} & 0 \\
\text{Null (0)} & 0 \\
\hline
\end{array}
\]

We use DST [9] to model reports. In DST, a Frame of Discernment (denoted as \( \Theta \)) represents a set of mutually exclusive hypotheses. In our scenario, \( \Theta \) is a set of non-overlapping alternatives of a particular target attribute, e.g., \( \{T, AC, UV\} \) in Report 1 or \( \{D1, D2\} \) in Report 2 of Table I. \( 2^\Theta \) denotes the power set of \( \Theta \). A basic belief assignment (BBA) is an assignment of mass (denoted as \( m \)) to each subset of \( 2^\Theta \). A mass is the amount of belief based on a node’s observation, which directly supports a given subset of \( 2^\Theta \). We denote the BBA of a report as \( m \), which is a vector of the individual masses ( \( m(\cdot) \)). Notice that an uncertain observation may lead to an assignment of mass to subsets which contain more than one alternative, e.g., \( \{T, AC\} \) and \( \{T, AC, UV\} \) in Report 1. Any mass assigned to the \( \{T, AC, UV\} \) subset (i.e., \( \Theta \)) does not help us to choose any of the alternatives, and therefore the \( \Theta \) subset in a report represents total ignorance. We include a null set in each report because TBM [8], handles conflicting evidence by allowing a non-zero mass of the null set. The amount of conflict among the fusion inputs is transferred to the null set \( m(0) \) after the fusion (the fusion process is elaborated in Section IV). Though conflict is not meaningfully quantifiable, the mass of the null set serves as an alarm signal for the existence of conflict and the level of conflict. The sum of the masses in a report should be unity. If not, the amount of missing mass (1 - sum of the masses) is assigned to ignorance (i.e., \( \Theta \)).

We deal with two types of nodes: regular nodes (RN) and decision maker (DM). An RN may generate, process and share observations (i.e., reports). A DM may make decisions based on received report(s). In this work, we consider one DM and multiple RNs. A node may share reports with its 1-hop neighbors. When an RN receives report(s), it may choose one operation among the following: (1) forward report(s) to one or more 1-hop neighbors without any modifications; (2) modify an individual report and share the updated report; (3) collect multiple reports, fuse them into one report, and share the fused report. A report shared by node \( v_i \) is denoted as \( r_j \).

Fig. 1 gives an overview of a node’s (an RN or a DM) operations in our system. The shaded actions in Fig. 1 are the key components of our decision fusion framework. Each of these shaded actions is elaborated in this paper. We define Node Trust as: node \( v_{mn} \)’s trust towards node \( v_{i} \), denoted as \( T_{mn} \), where \( T_{mn} \in [0,1] \), is \( v_{mn} \)’s subjective perception of \( v_{i} \)’s reliability in terms of sharing correct reports, based on the past reports received from \( v_{i} \). Every node maintains a node trust table locally which stores its subjective trust for other nodes. Node trust is initialized to 0.5, i.e., uncertainty. A node updates its local trust table whenever the node combines multiple received reports and obtains a fused report. The details of the node trust update process are described in Section V.

B. Provenance Model

Each report consists of meta-data and content. The content contains the BBA for a target attribute and the meta-data contains the provenance. Each source or intermediate node needs to generate a provenance record (denoted as \( P \)). The provenance of the entire report \( r_6 \) (denoted as \( P_6 \)) is represented as a tree of time-ordered provenance records \( p_1[p_2[\ldots p_i]] \). Fig. 2 shows an example scenario of report sharing and the structure of the final report and its provenance. In this scenario, \( v_1, v_2, v_3, v_4, v_5, v_6, v_7, v_8 \) are source nodes that make observations; \( v_4 \) simply forwards \( v_1 \)’s report; \( v_5, v_7 \) and \( v_8 \) receive multiple reports and fuse them before sending to the next hop. Finally, \( v_9 \) receives a report \( r_8 \) from \( v_8 \). The structure of \( r_8 \) is shown in Fig. 2 (b).

A report’s provenance \( P \) provides information about every node that has generated, forwarded or processed the content described by \( P \). A provenance record \( p_j \) of \( P \) consists of the following elements:

1) Node ID (\( v_j \))
2) Report generation time (\( RT_j \))
3) Performed action(s) (\( ACT_j \))
4) ID’s of previous 1-hop neighbors (\( \{v_{j1}, v_{j2}, \ldots v_{jk}\} \))
5) Trust recommendations for previous 1-hop neighbors (\( \{T_{j1}, T_{j2}, \ldots T_{jk}\} \))

where \( j_1, j_2, \ldots, j_k \) in items 4 and 5 represent the 1-hop neighbors of node \( v_j \) that passed reports to \( v_i \).

We define two required basic actions of \( ACT \): forwarding and processing. Forwarding means that the node only forwards the report without modification while processing means that

---

The 5th International Workshop on Information Quality and Quality of Service for Pervasive Computing 2013, San Diego (22 March 2013)
where $T^k_m$ is the node $v_m$’s direct trust towards node $v_k$, and $v_{k^+}$ represents $v_k$’s next-hop neighbor which has a trust recommendation $T^k_{k^+}$ for $v_k$ in its provenance record ($p_{k^+}$). The integrated node trust for the previous-hop sender ($v_m$) is nothing but the direct trust ($v^m_n$) in the node trust table because there is no trust recommendations for $v_n$ on the provenance. Otherwise, the integrated node trust is calculated based on both direct trust ($T^F_m$) and trust recommendation ($T^k_{k^+}$); the latter is multiplied by node $v_m$’s integrated node trust towards the recommender $v_{k^+}$, which requires recursive calculation. The other factors involving 0.5 are due to the fact trust values range from 0 to 1, with 0.5 representing ignorance.

The less a recommender is trusted, the more we want to push its trust recommendation to 0.5. $\rho^k_m$ controls the weight of the direct trust ($T^F_m$) and the adjusted trust recommendation ($0.5 + T^F_m (T^k_{k^+} - 0.5)$), and is defined as:

$$\rho^k_m = \begin{cases} T^F_m & \text{if } T^F_m \geq 0.5 \\ 1 - T^F_m & \text{otherwise} \end{cases}$$

The rationale behind Equation 3 is: The higher or lower the direct trust $v_m$ has for $v_k$, the more confident $v_m$ should be about this trust, assuming that this direct trust has been obtained after a large amount of interactions, thereby the trust recommendation from other nodes is weighted less. On the contrary, a direct trust around 0.5 means $v_m$ is uncertain about $v_k$’s trustworthiness, and thereby $v_m$ will rely equally on the trust recommendation from another node.

### B. Completeness

**Completeness** ($O$) is critical in deriving accurate information. We determine $O$ of a report based on two sub-properties:

1. **Content Completeness** ($CC$): This is a real number in [0, 1] that indicates the degree of completeness of the report. $CC$ is defined as the sum of masses in the report. We consider a report complete if the sum of the masses is unity; otherwise it is incomplete and we assign the missing mass to the null set.

2. **Provenance Completeness** ($PC$): This is a real number in the range of [0, 1], which indicates the degree of completeness of the report’s provenance. A weight $\epsilon_{pe}$ is given to each provenance element ($pe$). A higher $\epsilon_{pe}$ is assigned to a more important $pe$ based on the needs of the DM. The completeness of a single provenance record $p_i$ (denoted as $PC_i$) is computed by:

$$PC_i = \prod (1 - \epsilon_{pe}) \quad \text{for all } pe \text{'s missing on } p_i$$

$$PC_i = 0 \quad \text{if no } pe \text{'s missing}$$

$PC$ is the completeness of an entire report computed by averaging all of its $PC_i$’s.

We compute $O$ based on the product of $CC$ and $PC$:

$$O = CC \cdot PC$$

### C. Timeliness

**Timeliness** ($T$) refers to how fresh a received report is. High timeliness is desirable to capture recent information on a target attribute. $T$ is calculated by:

$$T = \begin{cases} 2 - 2 \frac{\tau_e - \tau_o}{\tau_C} & \text{if } \tau_e - \tau_o \leq TS \cdot \tau_C \\ 0 & \text{otherwise} \end{cases}$$

where $\tau_e - \tau_o$ is the time gap between evaluation time $\tau_e$ (i.e., the current time that $T$ is being evaluated) and observation time $\tau_o$ (i.e., when the first observation is made by an original source). $\tau_C$ is a scaling constant based on the network needs. $TS$ is the Target Stability, which is a parameter associated with the target being tracked. A more dynamic target is more time-sensitive and thereby a lower $TS$ value should be assigned.

Equation 6 implies that the decay of $T$ is exponential wrt time.
IV. REPORT FUSION AND DECISION MAKING

A. Discounting of Evidence

(1) Trust-aware Discounting: Reports that are considered untrustworthy should be discarded. More formally, we accept the BBA of a report if its trust level (defined as the product of correctness \( C \), completeness \( O \) and timeliness \( T \)) is one. Otherwise, we transfer a certain amount of mass of its BBA to \( \Theta \) (i.e., total ignorance) based on the trust level. The resulting BBA (denoted as \( \tilde{m} \)) is given by:

\[
\tilde{m}(A) = \begin{cases} 
\tilde{m}(A) \cdot C \cdot O \cdot T & \text{if } A \neq \Theta \\
1 - \sum_{B \neq \Theta} m(B) \cdot C \cdot O \cdot T & \text{if } A = \Theta
\end{cases}
\]

where \( A \) and \( B \) denotes any subset of \( \Theta \).

(2) Independence-aware Discounting: In a multi-hop network, reports may travel through independent paths or may traverse common subsets of nodes. We measure the independence of a report by looking at the amount of common provenance the report has with other reports. Let’s consider two reports \( r_i \) and \( r_j \) with processing provenance \( P_i^p \) and \( P_j^p \). We denote the number of nodes by \( |P_i^p| \) and \( |P_j^p| \) and the number of common nodes by \( |P_i^p \cap P_j^p| \). Then, we define the processing path difference as:

\[
\Delta(P_i^p, P_j^p) = \frac{\max\{|P_i^p|, |P_j^p|\} - |P_i^p \cap P_j^p|}{\max\{|P_i^p|, |P_j^p|\}}
\]

A newly received report’s independence is determined by the minimum processing path difference of the report compared with all other previously received reports. Hence, we calculate the independence (\( I \)) of a received report \( r_i \) by:

\[
I_i = \min_{r_j \in R} \Delta(P_i^p, P_j^p)
\]

where \( R \) denotes the set of previously received reports that need to be fused with \( r_i \).

Similar to trust-aware discounting, \( \tilde{m} \) is again discounted based on the corresponding report’s \( I \) value, and the resulting BBA (denoted as \( \tilde{\tilde{m}} \)) is given by:

\[
\tilde{\tilde{m}}(A) = \begin{cases} 
\tilde{m}(A) \cdot I & \text{if } A \neq \Theta \\
1 - \sum_{B \neq \Theta} \tilde{m}(B) \cdot I & \text{if } A = \Theta
\end{cases}
\]

B. Decision Making

(1) Report Fusion: After the discounting phase, the BBA of each report is weighed based on its importance level. The result of the fusion of two BBAs (\( \tilde{\tilde{m}}_1 \) and \( \tilde{\tilde{m}}_2 \)) for the mass of the Tank alternative is calculated by:

\[
\tilde{\tilde{m}}(T) = \tilde{\tilde{m}}_1(T)\tilde{\tilde{m}}_2(T) + \tilde{\tilde{m}}_1(T)\tilde{\tilde{m}}_2(T, AC) + \tilde{\tilde{m}}_1(T)\tilde{\tilde{m}}_2(T, UV) + \tilde{\tilde{m}}_1(T, AC)\tilde{\tilde{m}}_2(T) + \tilde{\tilde{m}}_1(T, AC, UV)\tilde{\tilde{m}}_2(T)
\]

Based on TBM [8], after fusing two BBAs, if the sum of masses for the non-empty subsets is not one, then the missing mass is caused by conflict and is transferred to \( \tilde{m}(\emptyset) \) which means that beliefs are assigned to subsets of \( 2^\Theta \) where some subsets contain more than one alternative (e.g., \( \Theta \)). When a decision must be made, the beliefs must be transformed to pignistic level, which means we have to re-distribute the masses of those subsets containing more than one alternative to the single alternatives \( \theta \in \Theta \) in order to see which alternative is the best to bet on. The result of such a transformation is called the pignistic probability function (denoted as \( \text{BetP} \)) [8]. Applied to our context, the resulting \( \text{BetP} \) is given by:

\[
\text{BetP}(\theta) = (1 - \tilde{m}(\emptyset)) \sum_{\theta \in A, A \in \Theta} \tilde{m}(A) \frac{|A|}{|A|}
\]

where \( \tilde{m}(\emptyset) \neq 1 \) and \( |A| \) is the cardinality of subset \( A \). The alternative \( \theta \) with the largest \( \text{BetP}(\theta) \) is selected.

(3) Decision Confidence: The \( \text{BetP} \) result helps the DM to select one from the alternatives, but it does not indicate the confidence level involved in the decision choice. We define the confidence of choosing \( \theta \) as:

\[
\text{Conf} = \text{BetP}(\theta) \cdot (1 - \tilde{m}(\emptyset) - \tilde{m}(\Theta)) \cdot e^{-\lambda T}
\]

Other than \( \text{BetP} \) of the decision choice, the confidence for a decision should also reflect the level of ignorance (\( \tilde{m}(\emptyset) \)) and conflict (\( \tilde{m}(\emptyset) \)) in the fused BBA. Take Report 2 in Table I as an example, let us compare two fused BBAs: (1) \( \tilde{m}(D_1) = 0.2, \tilde{m}(D_2) = 0.1, \tilde{m}(\emptyset) = 0.2, \tilde{m}(\Theta) = 0.5 \); (2) \( \tilde{m}(D_1) = 0.5625, \tilde{m}(D_2) = 0.4375, \tilde{m}(\emptyset) = 0, \tilde{m}(\Theta) = 0 \). The resulting \( \text{BetP} \) for both BBAs are the same: \( \text{BetP}(D_1) = 0.5625, \text{BetP}(D_2) = 0.4375 \), and hence \( D_1 \) is the decision choice for both cases. Assuming both BBAs are resulted from fusing the same number of reports, our confidence metric yields a higher \( \text{Conf} \) for the second case because of its lower ignorance and conflict. The \( e^{-\lambda T} \) term in the confidence metric is based on the form of reliability. The intuition is that one should be more confident in a fusion result of more reports, and this increment in confidence with the increment of fusion inputs should be negative exponential instead of linear.

(4) Decision Timing: In our context, reports are collected over time. To make a timely decision, the DM cannot keep waiting for reports, so a decision deadline must be set. However, the deadline should not be a fixed one for all targets with different levels of stability (TS), because more urgent decisions are often needed for targets with low TS while targets with high TS allow the DM to have more time collecting reports before making a decision. Hence, we set a dynamic decision deadline by defining a timeliness threshold \( T \). When the average timeliness of all the received reports reaches \( T \), a decision must be made. Based on the definition of \( T \) (Equation 6), once a \( T \) is chosen, the decision deadline is automatically adjusted based on TS of the target.

V. NODE TRUST UPDATE

After a node fuses multiple reports, the final fused report is assumed to be trustworthy. This is the time for the fusion node to perform an update on its own node trust table, based on the distance between its final fused report and each of its corresponding received reports. Notice that the fusion node could be either the DM or an intermediate RN. Though intermediate RNs do not make decisions, they also perform node trust update after fusing multiple reports. We adopt the Manhattan distance, which generates results that are more intuitively acceptable than other commonly used measures [12], to compute the distance between a received report’s BBA (\( \tilde{m} \)) and the fused BBA \( \tilde{\tilde{m}} \) based on their \( \text{BetP} \)’s:

\[
D(m, \tilde{m}) = \sum_{\theta \in \Theta} |\text{BetP}_m(\theta) - \text{BetP}_{\tilde{\tilde{m}}}(\theta)|
\]

484
The trust update is applied by the fusion node to the nodes which processed the received reports according to the provenance. We define a distance threshold $D \in (0, 1)$, which should be fine-tuned based on the network operations. If $D_i$ is smaller than $D$, we consider report $r_i$ supports the fused report and thus a reward (i.e., trust increment) is given to the nodes which processed $r_i$. A penalty (i.e., trust decrement) is given otherwise. The algorithm for local node trust update is given by Algorithm 1. The node trust update only changes the report and thus a reward (i.e., trust increment) is given to the remaining mass is randomly assigned to the other subsets.

Algorithm 1 Local node trust update

1. $v_{m} \leftarrow$ Report receiver and trust updating node
2. $R \leftarrow$ Received reports
3. for all $r_i \in R$ do
4. for all $p_k \in P_i$ do
5. $T_k = T_k^{m} + \left(\frac{|D(m, r_i)|}{|P_i|}\right) \cdot \gamma$
6. $\triangledown$ where $\gamma$ is a scaling parameter
7. $T_m = \min(1, \max(0, T_k^{m}))$
8. $\triangledown$ make sure $T_m \in [0, 1]$
9. end for
10. end for

VI. EXPERIMENTAL EVALUATION

A. Metrics

We use the following metrics to measure the performance of our proposed information fusion framework:

- Correct Decision Ratio (CDR): This is the ratio of correct decisions over the total number of decisions made.
- Average Decision Confidence (ADC): This is the average confidence level ($Conf$), over all decisions made. Similar false reports from colluding attackers may cause the DM to make wrong decisions with high confidence. ADC is cross-referenced with CDR to assess the reliability of the system.

B. Experiment Settings

We simulate a network with 100 RNs with a random mobility model. A stationary DM is set at the center of the operational area and every RN moves around the DM within a maximum allowed distance. We choose a random location for each target and the 10 nodes (good nodes or attackers) that are closest to the target are selected to make observations. Reports are continuously generated at these nodes. Each node shares its reports with its 1-hop neighbors that may change over time due to mobility. Similar to Report 1 of Table I, each report includes 3 non-overlapping alternatives and thus 8 subsets. One alternative is set as the ground truth.

We simulate a good node as one that generates reports with a high mass (drawn uniformly from [0.8, 1.0]) assigned to the ground-truth alternative, representing the node's certainty level for that alternative. The amount of uncertainty (i.e., the remaining mass) is randomly assigned among the other subsets. A good node always provides complete provenance elements (as defined in Section II-B) and follows the proposed protocols for fusing reports. An attacker node is simulated as one that generates false reports by assigning a high mass (drawn uniformly from [0.8, 1.0]) to a wrong alternative. The remaining mass is randomly assigned to the other subsets. In the same way as generating false reports, attackers always tamper with reports received from others. All attackers collude by choosing the same wrong alternative. An attacker also

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Default simulation parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of attackers</td>
<td>10</td>
</tr>
<tr>
<td>Target stability ($T/S$)</td>
<td>0.9</td>
</tr>
<tr>
<td>Timeliness threshold ($\tau$)</td>
<td>0.80</td>
</tr>
<tr>
<td>Report distance threshold ($D$)</td>
<td>0.35</td>
</tr>
<tr>
<td>Timeliness scaling parameter ($\lambda$)</td>
<td>0.40</td>
</tr>
<tr>
<td>Confidence scaling parameter ($\gamma$)</td>
<td>0.5</td>
</tr>
<tr>
<td>Node trust update scaling parameter ($\gamma$)</td>
<td>5.0</td>
</tr>
</tbody>
</table>

Fig. 3. Performance Comparison of TIA vs. TA vs. TBM in CDR and ADC

C. Results

We tested three different schemes: (1) our trust-aware and independence-aware scheme (TIA); (2) trust-aware only scheme (TA); and (3) basic TBM without trust-awareness or independence-awareness (TBM).

Fig. 3 (a) shows the impact of the number of attackers on CDR when these three different schemes are used. Recall that attackers perform fake information dissemination and colluding attacks. We observe that TIA outperforms the other two schemes, meaning that TIA is more resilient against attackers. The performance of TIA is more pronounced as more attackers exist in the network. This figure indicates that both trust-awareness and independence-awareness are essential in enhancing the CDR. Fig. 3 (b) shows the ADC results for the three schemes. An interesting observation is that the TA and TBM curves drop initially as the number of attackers increases, but start to rise in the end. However, Fig. 3 (a) tells us all the three systems are more prone to mistakes when there are more colluding attackers. A desired system property is to yield a low confidence for a wrong decision, so that the DM could be warned about the likelihood of a wrong decision before taking any actions. Obviously, we can claim that this property is broken for TA and TBM when the number of colluding attackers reaches a certain number. However, the TIA curve does not show such a problem even when 25 of the 100 nodes were colluding attackers. The reason is, when a large number of attackers collude, their similar false reports are likely to become the majority of all the reports received by the DM, which causes the DM to make wrong decisions with high confidence. However, the trust-awareness and independence-awareness of the TIA scheme could filter out those similar false reports and lower their influence over the decision making as well as the confidence analysis. The standard deviations of the ADC curves are high because of the fact that the ratio of correct/false reports received by the DM

485
Fig. 4. Impact of Target Stability on CDR and ADC

(a) CDR vs. Number of attackers
(b) ADC vs. Number of attackers

Fig. 5. Impact of Decision Timing on CDR and ADC

(a) CDR vs. Timeliness Threshold
(b) ADC vs. Timeliness Threshold

0.8
0.84
0.88
0.92
0.96
1

5
10
15
20
25

ADC
Number of A.ackers

TS = 0.9
TS = 0.7
TS = 0.5

CDR

0.85*
0.84
0.88
0.96

1%

0.4*
0.6*
0.8*
1%

0.2%
0.3%
0.5%
0.7%

VII. CONCLUSION

This paper proposed a trust-aware and independence-aware decision fusion protocol, which is built on top of Transferable Belief Model. In addition to the traditional “correctness” property, we take “completeness” and “timeliness” into account for assessing information trust, based on a provenance model we proposed. Node-level trust is also maintained and updated at each node. Both direct node trust and indirect node trust recommendations are used for information trust evaluation. In addition, provenance is also used to analyze the independence of received information. The weight of each information item is adjusted based on its trust and independence before the fusion process. Simulation results confirmed that our scheme enhances the reliability of the decision-making process when there are unreliable information sources. In the future, we will introduce a confidence-based decision timing model to handle the attribute changes, and study their impact on decision making.

REFERENCES