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
Email: {*xlwang, pmohapatra*}*@ucdavis.edu*
†Computational and Information Sciences Directorate, U.S. Army Research Laboratory
Email: {*jin-hee.cho.civ, kevin.s.chan.civ, ananthram.swami.civ*}*@mail.mil*
‡Computer Science Department, Virginia Tech
Email: *mjjang@vt.edu*

***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. Transferable Belief Model (TBM), an extension of Dempster-Shafer Theory (DST), is a well known information fusion framework to combine multiple evidence in order to derive a unified belief where conflicting evidence exists. 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 the reliability of fusion. 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 (non-trust or non-independence) 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 a correct perception of the situation in the network, in order to make right decisions for handling the situation. 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 observation 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 target. The problem is compounded by the misbehaving nodes who supply false data in the network. This work aims at enhancing the accuracy of the information fusion and thus the overall reliability of decision-making in such a dynamic network environment in the presence of malicious entities.

Information fusion techniques have been studied extensively [1], [2]. However, 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 an inaccurate decision making due to the use of untrustworthy evidence. Therefore, assessing the trust of information becomes important to obtain accurate fusion results. 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, and thus leading to the dependence among information. To tackle this problem, we introduce independence-awareness based on analyzing the overlapping provenance between information items, which 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 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. Physical interpretation of the results is also provided. Section VII concludes our paper and suggests future work.

II. System Model

A. Network Model

We consider a heterogeneous network consisting of a set of nodes, $\{v_1, v_2, ..., v_N\}$, 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 about a vehicle. The two reports illustrate the target vehicle's type and location.

TABLE I
Examples of report

(a) Report 1: target type

Vehicle type	BBA
Tank (T)	0.4
Armored car (AC)	0.2
Utility vehicle (UV)	0.1
T or AC ({T, AC})	0.2
T or UV ({T, UV})	0
AC or UV ({AC, UV})	0
Ignorance ({T, AC, UV})	0.1
Null ($\emptyset$)	0

(b) Report 2: target location

Location	BBA
District 1 (D1)	0.8
District 2 (D2)	0.2
Ignorance ({D1, D2})	0
Null ($\emptyset$)	0

We use DST [9] to model reports. In DST, a *Frame of Discernment* (denoted as Θ) represents a set of mutually exclusive hypotheses. In our scenario, Θ 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^Θ is the power set of Θ. A *basic belief assignment* (BBA) is an assignment of *mass* (denoted as m) to each subset of 2^Θ. A mass is the amount of belief based on a node's observation, which directly supports a given subset of 2^Θ. We denote the BBA of a report as $\boldsymbol{m}$, which is a vector of the individual masses ($m(.)$). Notice that an uncertain observation may lead to an assignment of mass to subsets which contain more than one alternatives, e.g., {*T, AC*} and {*T, AC, UV*} in Report 1. Any mass assigned to the {*T, AC, UV*} subset (i.e., Θ) does not help to choose any of the alternatives, and therefore the Θ subset in a report represents total ignorance. We include a null set in each report because TBM [8], which we use for report fusion, 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 ($m(\emptyset)$) 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., Θ).

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 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_i.

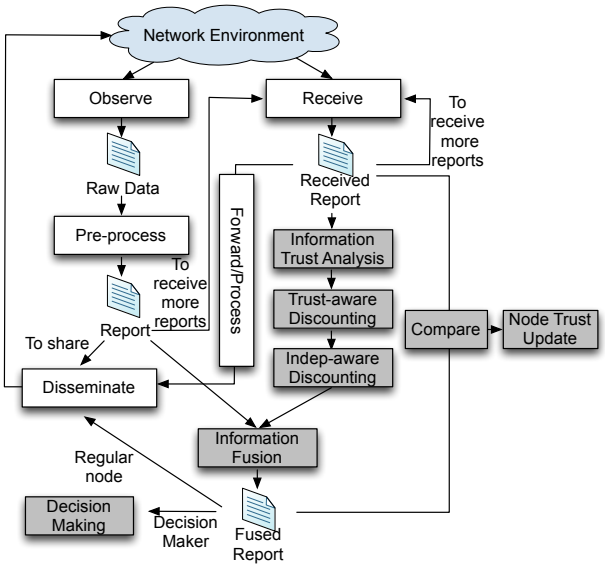


Fig. 1. Node operations for decision making

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_m's trust towards node v_n, denoted as T_m^n where $T_m^n \in [0, 1]$, is v_m's subjective perception of v_n's reliability in terms of *sharing correct reports*, based on the past reports received from v_n. 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.

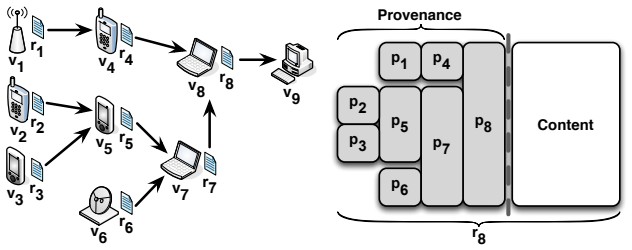
B. Provenance Model

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_i (denoted as P_i) is represented as a chain of time-ordered provenance records $p_1|p_2|\cdots|p_i$. 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 and v_6 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 ($\{v_{j_1}, v_{j_2}, \cdots v_{j_k}\}$)
5) Trust recommendations for previous 1-hop neighbors ($\{T_j^{j_1}, T_j^{j_2}, \cdots T_j^{j_k}\}$)

where $j_1, j_2, \cdots, j_k$ in items 4 and 5 represent the incoming 1-hop neighbors of node v_j.



(a) A scenario of report sharing (b) Structure of report r_8

Fig. 2. An example of report sharing and the structure of the final report

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 node modified the report (including fusing multiple reports). We define the part of the provenance chain P_i with only $ACT = processing$ nodes as the *processing provenance* of r_i, denoted as P_i^p. That is, P_i^p ignores the nodes who only forwarded r_i without modification.

C. Adversary Model

We assume that the DM is trustworthy. An RN may be trustworthy (i.e., a good node) or compromised (i.e., an attacker). A good node complies with the given protocol. An attacker may provide false report(s) or tamper with other's reports in arbitrary ways before re-sharing (i.e., fake information dissemination attack) in order to disrupt the decision making process. We expect an intelligent attacker would prefer tampering with over dropping reports. Thus, we do not specifically deal with report dropping attack. Multiple attackers may collude to enlarge the impact of false information dissemination by generating similar false reports.

Since nodes give trust recommendations based on their subjective trust for other nodes, an attacker may include false trust recommendations in its provenance records (i.e., good/bad mouthing attacks). We assume that the other provenance elements are generated by trustworthy middleware. Entities are allowed to choose not to supply some provenance elements. However, they are not able to lie about the elements they choose to supply. In addition, we assume provenance chains are securely protected by using approaches proposed in [11] while they are traveling in the network. Denial of Service attacks (e.g., traffic jamming) are out of the scope of this paper.

III. Information Trust Metric

Our information trust metric assesses information from three dimensions: *correctness*, *completeness* and *timeliness*. The details of each dimension are described below.

A. Correctness

Correctness ($\mathcal{C}$) is a real number in the range of [0, 1] and is based on a report's provenance and node trust values. $\mathcal{C}$ synthesizes both direct node trust and trust recommendations on the provenance. Suppose node v_m receives a report r_n from node v_n, $\mathcal{C}$ for r_n is given by:

$$\mathcal{C}_n = \frac{\sum_{p_k \in P_n^p} \hat{T}_m^k}{|P_n^p|} \tag{1}$$

where $\hat{T}_m^k \in [0, 1]$ is called v_m's *Integrated Node Trust* for a node v_k which processed r_n. P_n^p is the *processing provenance* of r_n and $|P_n^p|$ is the number of nodes in P_n^p. The calculation of $\hat{T}_m^k$ is given by:

$$\hat{T}_m^k = \begin{cases} T_m^k & \text{if } k = n \\ \rho_m^k T_m^k + (1 - \rho_m^k)[0.5 + \hat{T}_m^{k+}(T_{k+}^k - 0.5)] & \text{otherwise} \end{cases} \tag{2}$$

where T_m^k 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_n) 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_m^k) 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. ρ_m^k controls the weight of the direct trust (T_m^k) and the adjusted trust recommendation ($0.5 + \hat{T}_m^{k+}(T_{k+}^k - 0.5)$), and is defined as:

$$\rho_m^k = \begin{cases} T_m^k & \text{if } T_m^k \geqslant 0.5 \\ 1 - T_m^k & \text{otherwise} \end{cases} \tag{3}$$

The rationale behind Equation 3 is: The higher or lower a direct trust v_m has for v_k, the more confident v_m should be about this trust, assuming that a high or low level of direct trust is only obtained after a large amount of interactions, thereby the trust recommendation from another node 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 ($\mathcal{O}$) is critical in deriving accurate information. We determine $\mathcal{O}$ 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 its sum of the masses is unity; otherwise it is incomplete.

(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 ϵ_{pe} is given to each provenance element (pe). A higher ϵ_{pe} is assigned to a more important pe based on the needs of the network. The completeness of a single provenance record p_i (denoted as PC_i) is computed by:

$$PC_i = \begin{cases} \prod(1 - \epsilon_{pe}) & \text{for all } pe\text{'s missing on } p_i \\ 1 & \text{if no } pe \text{ missing} \end{cases} \tag{4}$$

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

We compute $\mathcal{O}$ based on the product of CC and PC:

$$\mathcal{O} = CC \cdot PC \tag{5}$$

C. Timeliness

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

$$\mathcal{T} = \begin{cases} 2 - 2^{\frac{\tau_e - \tau_o}{TS \cdot \tau_C}} & \text{if } \tau_e - \tau_o \leqslant TS \cdot \tau_C \\ 0 & \text{otherwise} \end{cases} \tag{6}$$

where $\tau_e - \tau_o$ is the time gap between evaluation time τ_e (i.e., the current time that $\mathcal{T}$ is being evaluated) and observation time τ_o (i.e., when the first observation is made by an original source). τ_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. The way we define Equation 6 implies that the decay of $\mathcal{T}$ should be exponential over time because a target attribute is not likely to change much within a short period of time since the creation of the report.

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 original BBA of a report if its trust level (defined as the product of correctness $\mathcal{C}$, completeness $\mathcal{O}$ and timeliness $\mathcal{T}$) is one. Otherwise, we transfer a certain amount of mass of its BBA to Θ (i.e., total ignorance) based on the trust level. The resulting BBA (denoted as $\dot{m}$) is given by:

$$\dot{m}(A) = \begin{cases} m(A) \cdot \mathcal{C} \cdot \mathcal{O} \cdot \mathcal{T} & \text{if } A \neq \Theta \\ 1 - \sum\limits_{B \neq A} m(B) \cdot \mathcal{C} \cdot \mathcal{O} \cdot \mathcal{T} & \text{if } A = \Theta \end{cases} \tag{7}$$

where A and B denotes any subset.

(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 of the report has compared 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:

$$\Delta(P_i^p, P_j^p) = \frac{max\{|P_i^p|, |P_j^p|\} - |P_i^p \cap P_j^p|}{max\{|P_i^p|, |P_j^p|\}} \tag{8}$$

A newly received report's independence should be determined by the minimum processing path difference of the report compared with all other previously received reports. Hence, we calculate the *independence* ($\mathcal{I}$) of a received report r_i by:

$$\mathcal{I}_i = \begin{cases} 1 & \text{if } R = \emptyset \\ \min\limits_{r_j \in R} \left(\Delta(P_i^p, P_j^p)\right) & \text{otherwise} \end{cases} \tag{9}$$

where R denotes the set of previously received reports to be fused with r_i.

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

$$\ddot{m}(A) = \begin{cases} \dot{m}(A) \cdot \mathcal{I} & \text{if } A \neq \Theta \\ 1 - \sum\limits_{B \neq \Theta} \dot{m}(B) \cdot \mathcal{I} & \text{if } A = \Theta \end{cases} \tag{10}$$

B. Decision Making

(1) Report Fusion: After the discounting phase, the BBA of each report is weighed based on its importance level. The final unified BBA (denoted as ($\hat{m}$) can be obtained by fusing all the discounted BBAs ($\ddot{m}$). We use the *TBM combination rule* [8] to fuse two BBAs, which is given by:

$$\hat{m}(A) = \sum_{B \cap C = A \neq \emptyset} \ddot{m}_1(B) \ddot{m}_2(C) \tag{11}$$

Take Report 1 of Table I as an example, the fusion result of two discounted BBAs ($\ddot{m}_1$ and $\ddot{m}_2$) for the mass of the *Tank* alternative ($\hat{m}(T)$) is calculated by:

$$\begin{aligned} \hat{m}(T) = {} & \ddot{m}_1(T)\ddot{m}_2(T) + \ddot{m}_1(T)\ddot{m}_2(\{T, AC\}) \\ & + \ddot{m}_1(T)\ddot{m}_2(\{T, UV\}) + \ddot{m}_1(T)\ddot{m}_2(\{T, AC, UV\}) \\ & + \ddot{m}_1(\{T, AC\})\ddot{m}_2(T) + \ddot{m}_1(\{T, UV\})\ddot{m}_2(T) \\ & + \ddot{m}_1(\{T, AC, UV\})\ddot{m}_2(T) \end{aligned} \tag{12}$$

The TBM combination rule has the properties of *associativity* and *commutativity*, so the order of reports being combined does not affect the final fusion result. 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 transfered to $m(\emptyset)$ which represents the amount of conflict between the fusion inputs.

(2) Alternative Selection: According to TBM [8], the fused BBA is at *credal level*, which means that beliefs are assigned to subsets of 2^Θ where some subsets contain more than one alternative (e.g., Θ). 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 $BetP$) [8]. Applied to our context, the resulting $BetP$ is given by:

$$BetP(\theta) = (1 - \hat{m}(\emptyset)) \cdot \sum_{\theta \in A, A \subseteq \Theta} \frac{\hat{m}(A)}{|A|} \tag{13}$$

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

(3) Decision Confidence: The $BetP$ result helps the DM to select one from the alternatives, but it fails to indicate the confidence level involved in the decision choice. We define the *confidence* of choosing θ as:

$$Conf = BetP(\theta) \cdot (1 - \hat{m}(\emptyset) - \hat{m}(\Theta)) \cdot e^{-\frac{\lambda}{F}} \tag{14}$$

where F is the number of fused reports and λ is a scaling parameter.

Other than $BetP$ of the decision choice, the confidence for a decision should also reflect the level of ignorance ($\hat{m}(\Theta)$) and conflict ($\hat{m}(\emptyset)$) in the fused BBA. Take Report 2 in Table I as an example, let us compare two fused BBAs: (1) $\hat{m}(\text{D1}) = 0.2, \hat{m}(\text{D2}) = 0.1, \hat{m}(\emptyset) = 0.2, \hat{m}(\Theta) = 0.5$; (2) $\hat{m}(\text{D1}) = 0.5625$, $\hat{m}(\text{D2}) = 0.4375, \hat{m}(\emptyset) = 0, \hat{m}(\Theta) = 0$. The resulting $BetP$ for both BBAs are the same: $BetP(\text{D1}) = 0.5625, BetP(\text{D2}) = 0.4375$, and hence D1 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 $Conf$ for the second case because of its lower ignorance and conflict. The $e^{-\frac{\lambda}{F}}$ 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* $\overline{\mathcal{T}}$. When the average timeliness of all the received reports reaches $\overline{\mathcal{T}}$, a decision must be made. Based on the definition of $\mathcal{T}$ (Equation 6), once a $\overline{\mathcal{T}}$ 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

$(\boldsymbol{m})$ and the fused BBA $\hat{\boldsymbol{m}}$ based on their $BetP$'s:

$$D(\boldsymbol{m}, \hat{\boldsymbol{m}}) = (\frac{1}{2} \sum_{\theta \in \Theta} |BetP_m(\theta) - BetP_{\hat{m}}(\theta)|) \quad (15)$$

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* $\overline{D} \in (0, 1)$, which should be fine-tuned based on the network operations. If D_i is smaller than $\overline{D}$, 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 fusion node's local trust table. The updated trust table is then used by the fusion node for correctness ($\mathcal{C}$) evaluation of its subsequently received reports.

Algorithm 1 Local node trust update

```
v_m ← Report receiver and trust updating node
R ← Received reports
for all r_i ∈ R do
    for all p_k ∈ P_i^p do
        T_m^k = T_m^k + ((D̄ − D(m_i, m̂)) / |P_i^p|) · γ
        // where γ is a scaling parameter
        T_m^k ← min(1, max(0, T_m^k))
        // make sure T_m^k ∈ [0, 1]
    end for
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 (i.e., the correct alternative is chosen) 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 see the reliability of the system in terms of confidence analysis.

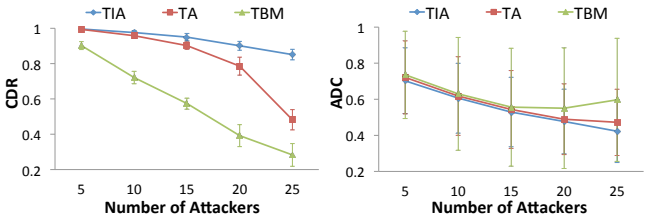
B. Experiment Settings

We simulate 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. 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 (randomly chosen in the range of [0.8, 1.0]) assigned to the ground-truth alternative, representing the node's certainty level for the 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 (randomly chosen in the range of [0.9, 1.0]) to a wrong alternative. The remaining mass is randomly assigned to the other subsets. Attackers always tamper with reports received from others in the same way as they generate false reports. All attackers collude by choosing the same wrong alternative. An attacker also inserts random trust recommendations and/or randomly hides some provenance elements in its provenance records. We vary the number of attackers from 5 to 25 among the 100 nodes. These attackers are randomly distributed in the network.

TABLE II
Default Simulation Parameters

Parameter	Value
Number of attackers	10
Target stability (TS)	0.9
Timeliness threshold ($\overline{T}$)	0.80
Report distance threshold ($\overline{D}$)	0.35
Timeliness scaling parameter (τ_C)	40.0
Confidence scaling parameter (λ)	0.5
Node trust update scaling parameter (γ)	5.0



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

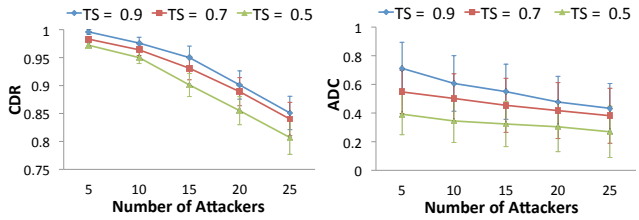
Fig. 3. Performance Comparison of TIA vs. TA vs. TBM in CDR and ADC

Table II shows the default parameter values we used. We vary the key parameters (e.g., number of attackers, target stability) to demonstrate their impacts on our performance metrics. Our results are based on observations over 20 runs of the simulation and 1000 decisions for each run.

C. Results

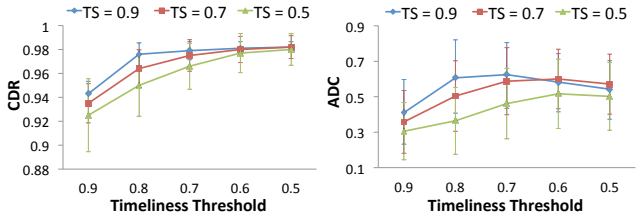
We tested three different schemes: (1) our trust-aware and independence-aware scheme (TIA); (2) trust-aware only scheme (TA); and (3) basic TBM without trust-awareness or independence-awareness (TBM).

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 as far as when 25 colluding attackers exist out of 100 nodes. 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



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

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



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

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

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 varies a lot for different decisions because of the randomness of the network settings. Different correct/false reports ratios may lead to very difference confidence levels.

Fig. 4 shows how Target Stability (TS) affects CDR and ADC of the TIA scheme. Recall that higher TS means that a target does not change its status fast while a target with lower TS is more unstable. Fig. 4 (a) shows that higher TS generates higher CDR. As expected, as more attackers are in the network, the adverse impact of low TS is more pronounced. Fig. 4 (b) also shows a similar trend as 4 (a) in that higher TS introduces high ADC while lower TS lowers down ADC. The reason is that if TS is low, the DM needs to make a decision within a shorter period of time, which means decisions must be made based on a smaller number of reports. Therefore, false reports provided by the attackers are likely to have more influence, resulting a lower CDR. Making decisions based on fewer reports leads to a lower ADC.

Fig. 5 shows the impact of the decision timing on our performance metrics. Decision timing is mainly affected by two factors: target stability (TS) and timeliness threshold ($\overline{\mathcal{T}}$). For a particular target, TS is known and fixed. Therefore, $\overline{\mathcal{T}}$ determines the decision delay. Lower $\overline{\mathcal{T}}$ allows the DM to wait a longer time before making a decision. Longer waiting time means more reports to be received by the DM with a higher communication cost because of the continuous report sharing in the network. Fig. 5 (a) and (b) show there is an optimal point at which we should make a decision. From Fig. 5 (a), we observe that CDR increases as $\overline{\mathcal{T}}$ decreases, due to the increment of received reports. However, the increment of CDR is only significant at the beginning part of the curves. This is because when the number of received reports reaches a certain level, any additional reports are likely to have large dependence with the previously received reports, since only 10 nodes are making observations. Therefore, the additional reports do not give much valuable information. Fig. 5 (b) shows that the ADC also increases at the beginning part of the curves due to the increment of received reports. However, the ADC curves start to drop when the $\overline{\mathcal{T}}$ becomes lower than a certain level. This is because the confidence metric reflects the level of uncertainty causes by the decrement of timeliness. Based on our definition of confidence (Equation 14), after the DM already received certain reports, any additional reports do not increase the confidence much. However, the continuous decrement of overall timeliness makes the mass of ignorance ($\hat{m}(\Theta)$) keep increasing, due to the trust-aware discounting of the fusion inputs. Therefore, if the DM waits too long, its confidence starts to drop. From these observations, we can see that a proper $\overline{\mathcal{T}}$ should be set to get high CDR and ADC without incurring too much delay and communication cost.

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 the provenance model we proposed. Node level trust is also maintained. 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 plan to introduce a confidence-based decision timing, model the changes of target attributes, and study their impact on decision making.

Acknowledgment

This research was partially supported by a grant from DDR&E Cyber Security Science and Technology Program, and was partially supported by the Army Research Laboratory under Cooperative Agreement Number W911NF-09-2-0053. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official polices, either expressed or implied, of the Army Research Laboratory or the U.S. Government. The U.S Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copy right notation here on.

References

[1] D. Smith and S. Singh, "Approaches to multisensor data fusion in target tracking: A survey," *IEEE Transactions on Knowledge and Data Engineering*, vol. 18, no. 12, pp. 1696–1710, 2006.

[2] E. Nakamura, A. Loureiro, and A. Frery, "Information fusion for wireless sensor networks: Methods, models, and classifications," *ACM Computing Surveys*, vol. 39, no. 9, 2007.

[3] M. Raya, P. Papadimitratos, V. Gligor, and J. Hubaux, "On data-centric trust establishment in ephemeral ad hoc networks," in *Proc. IEEE INFOCOM 2008*, pp. 1238–1246.

[4] X. Wang, K. Govindan, and P. Mohapatra, "Collusion-resilient quality of information evaluation based on information provenance," in *Proc. IEEE SECON 2011*.

[5] S. Arunkumar, M. Srivatsa, C. Bisdikian, and M. Sensoy, "Trust assessment when observing and orienting with uncertain, multi-source streaming information," in *Proc. ACITA 2012*.

[6] C. Bisdikian, L. Kaplan, M. Srivastava, D. Thornley, D. Verma, and R. Young, "Building principles for a quality of information specification for sensor information," in *Proc. IEEE FUSION*, 2009.

[7] A. Bar-Noy, G. Cirincione, R. Govindan, S. Krishnamurthy, T. LaPorta, P. Mohapatra, M. Neely, and A. Yener, "Quality-of-information aware networking for tactical military networks," in *Proc. IEEE PERCOM Workshops 2011*, pp. 2–7.

[8] P. Smets and R. Kennes, "The transferable belief model," *Artificial Intelligence*, vol. 66, no. 2, pp. 191–234, 1994.

[9] G. Shafer, *A Mathematical Theory of Evidence*. Princeton university press, 1976.

[10] T. Denœux, "Conjunctive and disjunctive combination of belief functions induced by nondistinct bodies of evidence," *Artificial Intelligence*, vol. 172, no. 2-3, pp. 234–264, 2008.

[11] X. Wang, K. Zeng, K. Govindan, and P. Mohapatra, "Chaining for securing data provenance in distributed information networks," in *Proc. IEEE MILCOM 2012*.

[12] Z. Liu, J. Dezert, Q. Pan *et al.*, "A new measure of dissimilarity between two basic belief assignments," in *Proc. IEEE FUSION 2010*.