Robust Aggregation of Inconsistent Information: Concepts and Research Directions

Aleksandar Ignjatovic, Mohsen Rezvani, Mohammad Allahbakhsh, and Elisa Bertino

1 Introduction

Today, more than ever, there is a critical need for organizations to share data within and across the organizations so that analysts, decision makers and control systems can make effective decisions. However, in order for analysts and decision makers to produce an accurate analysis and make effective decisions and take actions, data must be trustworthy. Therefore, it is critical that data trustworthiness issues, which also include data quality, provenance and lineage, be investigated for organizational data sharing, situation assessment, multi-sensor data integration and numerous other functions to support decision makers and analysts. Almost all application domains that we may think of require the ability to assess data trustworthiness; notable examples include: sensor networks (Lim, Moon, & Bertino, 2010; Lim, Ghinita, Bertino, & Kantarcioglu, 2012), social networks (Dai, Rao, Truta, & Bertino, 2012), location-based applications (Dai, Rao, Ghinita, & Bertino, 2011) critical infrastructures, e-health, and peer marking for massive open online courses (MOOCs).

The problem of providing trustworthy data to users and applications is an inherently difficult problem that requires articulated solutions combining different methods and techniques, ranging from iterative filtering (IF) algorithms (Laureti, Moret, Zhang, & Yu, 2006) to semantic integrity and ontology-based reasoning to digital signature techniques—just to name a few. It is however important to notice that technology has made possible to collect data from many different, possibly independent, sources. The advent of the Internet of Things (IoT) will further push

A. Ignjatovic • M. Rezvani • M. Allahbakhsh
University of New South Wales, Sydney, NSW, Australia
e-mail: aleksignjat@gmail.com

E. Bertino
Purdue University, West Lafayette, IN, USA

S.A. Matei et al. (eds.), *Transparency in Social Media*,
Computational Social Sciences, DOI 10.1007/978-3-319-18552-1_3

such capabilities. The availability of multiple observations and data pertaining to the same event or phenomenon in both the cyber space and the physical space represents an important opportunity for methodologies, referred to as data aggregation methodologies, aiming at assessing data trustworthiness by comparing and aggregating such multiple observations. Such methodologies can also include the use of IF algorithms resulting in iterative data aggregation methodologies. However, a major problem of data aggregation methodologies is that data items representing such observations are often inconsistent. Such inconsistencies arise because of errors, such as human and application errors or sensor calibration errors, or may be a result of deliberate attacks by malicious parties aiming at injecting deceiving information.

The use of provenance techniques may help in addressing such a problem. Provenance tracing makes it possible to trace back the source of a data item and the path that the data item followed in a given system in order to reach the intended recipient. Such provenance information can be used as a factor for assessing data trustworthiness in that it allows one to assign different weights to data items based on the source. An approach that combines IF with provenance has been proposed by Lim et al. (2010) in the context of sensor networks. Such approach is efficient and effective and has been widely extended. However, a major drawback of such approach is that it is not robust against collusion attacks. A collusion attack is one by which multiple malicious parties cooperate in order to inject deceiving information. Under such an attack, the data aggregation methodology will assess data as trustworthy whereas the data is not.

The problem of designing data aggregation methodologies that are robust against collusion attacks has been recently addressed by a novel IF methodology by Rezvani, Ignjatovic, Bertino, and Jha (2015). Such methodology is applicable to both numerical and non-numerical data, and, compared with the "classical" IF algorithms of Laureti et al. (2006), Yu, Zhang, Laureti, and Moret (2006) and De Kerchove and Van Dooren (2007, 2008, 2010) greatly improve the numerical stability of data aggregation as well as robustness against the collusion attacks.

In this paper we provide a survey of IF methodologies for assessing data trustworthiness and introduce a research roadmap to guide future research. In what follows, we first survey the methodology by Lim et al. (2010), Laureti et al. (2006), Yu et al. (2006) and De Kerchove and Van Dooren (2007, 2008, 2010) to introduce the basic concepts and IF with provenance. We then show a collusion attack against such methodology and survey the IF methodology by Rezvani et al. (2015). Experimental results show that this methodology is highly effective against collusion attacks. We then discuss relevant research directions and finally outline a few conclusions.

2 Provenance-Based Data Trustworthiness Assessment

A cyclic and provenance-aware trust computation framework was proposed by Lim et al. (2010) in the context of sensor networks. The proposed framework is based on a heuristic that the more trustworthy data a sensor node reports, the higher the node's trust score is. Moreover, the trustworthiness of a data item depends on the trust scores of the nodes which passed it towards the server node. The nodes through which a data item has been passed in the sensor network represent the *provenance* of such data item. By taking into account such interdependency relationship between the trustworthiness of data items and sensor nodes, a cyclic trust computation has been proposed in which the trust scores evolve gradually. This framework which we briefly review now can be employed as an online trust computation method. In what follows, we first introduce the network model underlying this framework, and the relevant notions of provenance. We then describe the cyclic framework, and finally report results from the experimental evaluation in Lim et al. (2010).

2.1 Background Notions

A sensor network is represented by m sensor nodes n_i, $i = 1, \ldots, m$ with identifier i for node n_i. In such a network, all sensor nodes are responsible for monitoring one event (i.e. nodes report multiple independent observations for one event). The sensor network is modeled as a graph $G(N, E)$, where $N = \{n_1, n_2, \ldots, n_m\}$ is the set of nodes and $E\{e_{i,j}\}$ denotes the set of edges, with $e_{i,j}$ an edge connecting nodes n_i and n_j. Figure 1a shows an example of a sensor network. As one can see in

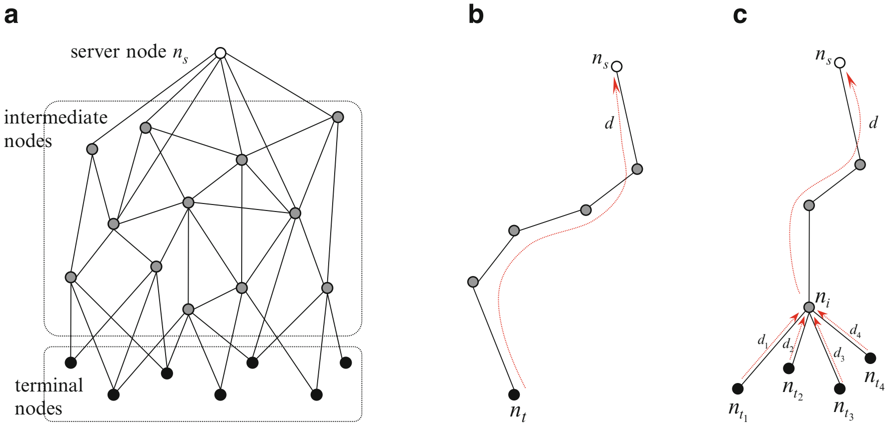

Fig. 1 Sensor network and data provenance examples. (**a**) Sensor network example. (**b**) Simple path example. (**c**) Tree path example

this figure, network nodes in N can be categorized into three types according to their roles in the network: a *terminal*, an *intermediate*, or a *server* node.

Definition 1 (Lim et al. (2010)).
A *terminal node* is a sensing node which generates a data item and sends it to one or more intermediate or server nodes (black filled nodes in Fig. 1a). An *intermediate node* receives data items from one or more terminal or intermediate nodes and passes them to another intermediate or a server node; it may also perform an aggregation function over the received data items and send the aggregate value to an intermediate or a server node (gray filled nodes in Fig. 1a). A *server node* (or base station) receives data items and evaluates continuous queries based on those items (white nodes in Fig. 1a).

Without loss of generality, it is assumed that there is only one server node in the network, denoted by n_s. Moreover, a data item d is represented by a single numeric value v_d.

In data management, the provenance concept represents the path of provisioning a data item. The provenance of a data item d, denoted by p_d, records where and how the data item d has been generated and how it has been passed through the sensor network towards the server n_s.

Definition 2 (Lim et al. (2010)).
The *provenance* p_d of a data item d is a rooted tree satisfying the following properties: (1) p_d is a subgraph of the sensor network $G(N,E)$; (2) the root node of p_d is the server node n_s; and (3) for two nodes n_i and n_j of p_d, n_i is a child of n_j if and only if n_i has passes the data item d to n_j through a direct link.

According to the tree nature of the data provenance, intermediate nodes are categorized into two categories: simple and aggregate.

- A *simple node* is an intermediate node having only one child. For example, in Fig. 1b every intermediate node is a simple node. Accordingly, a data provenance with only simple nodes can be represented by a simple path and this type of provenance is called a *simple provenance*.
- An *aggregate node* is an intermediate node with more than one child nodes. Figure 1c shows an intermediate node n_i which is an aggregate node and generates a new data item d by aggregating multiple data items $[d_1, d_2, d_3, d_4]$ received from nodes $[n_1, n_2, n_3, n_4]$ and passes d to the server n_s. A data provenance with at least one aggregate node is represented as a tree rather than a simple path and this provenance is called an *aggregate provenance*.

As an example of the sensor network, we can assume that a number of different sensors are distributed in a battlefield to collect the enemy locations (Tang et al., 2010). The sensors continuously watch the areas day and night to detect approaching enemies and send alarms to a server node. Moreover, the sensors are using a multihop routing scheme where each sensor may pass through the data of other sensors towards a server node.

2.2 Cyclic Trust Computation Framework

The main idea behind the trust computation approach by Lim et al. (2010) is to model the interdependency relationship between the trustworthiness of data items and their corresponding network nodes (as shown in Fig. 2). As one can see in this figure, the trust scores are assigned to both data items and network nodes, in an interdependent manner. The trust score of a data item is partially measured by the trust scores of the network nodes within its provenance. On the other hand, the trust score of a network node depends on the trustworthiness of data items that are generated by or passed through the node.

Figure 3 shows how the cyclic framework proposed in Lim et al. (2010) uses this interdependency to compute the trust scores of data items and network nodes. As shown in the figure, there are three different types of trust scores, *current*, *intermediate*, and *next*, for every data item and network node. The dashed line has separated the trust computation modules for data items and network nodes; the solid lines are traversed from one computation module to the next one.

For a set of data items received for a same event in the current window, the methodology by Lim et al. (2010) computes the current and intermediate trust scores for each data item in the first and second steps, respectively. The current trust score for a data item depends on the current trust scores of the nodes in its provenance, while its intermediate trust score is computed based on the latest set of

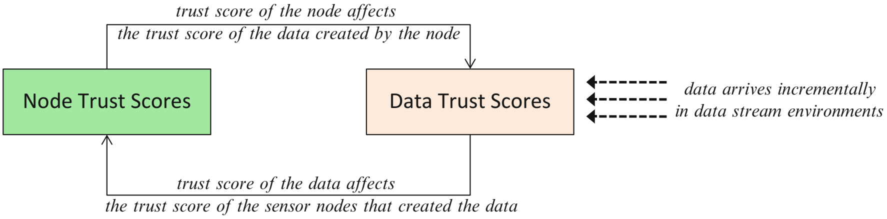
trust score of the node affects
the trust score of the data created by the node
Node Trust Scores
Data Trust Scores
data arrives incrementally
in data stream environments
trust score of the data affects
the trust score of the sensor nodes that created the data

Fig. 2 Interdependency between data and node trust scores

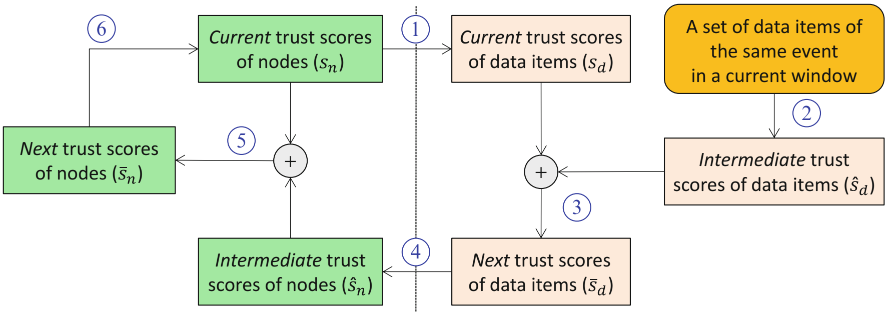
6
Current trust scores of nodes (s_n)
1
Current trust scores of data items (s_d)
A set of data items of the same event in a current window
2
Next trust scores of nodes ($\bar{s}_n$)
5
+
+
Intermediate trust scores of data items ($\hat{s}_d$)
3
Intermediate trust scores of nodes ($\hat{s}_n$)
4
Next trust scores of data items ($\bar{s}_d$)

Fig. 3 An cyclic framework for computing trust scores

data items reported for a same event in the current streaming window. In the third step, the next trust score for each data item is computed by aggregating the current and intermediate trust scores of data items.

As shown in left side of Fig. 3, the intermediate trust score for each network node is calculated based on the trust scores of its related data items (step 4). After that, the next trust score for a network node is obtained by combination of its current and intermediate trust scores. Finally, the next trust scores in the current streaming window are copied to the current scores in the next window (step 6). Note that the cyclic trust computation process needs initial trust scores for sensor nodes which are set to one for all nodes at a very beginning of the process.

Computing Node Trustworthiness As we described, the current trust score of a network node n, denoted by s_n, is equal to the next trust score obtained in the previous streaming window for that node. Thus, one needs to compute its intermediate and next trust score in the current window, denoted by $\hat{s}_n$ and $\bar{s}_n$, respectively.

The intermediate trust score of a network node n is computed based on the trustworthiness of its corresponding data items, which is a set of data items that are generated or passed through such a node during the current streaming window, denoted by D_n. The intermediate trust score $\hat{s}_n$ is simply computed as the average of the trustworthiness of its related data items, as follows:

$$\hat{s}_n = \frac{\sum\limits_{d \in D_n} \bar{s}_d}{|D_n|}, \tag{1}$$

where $|D_n|$ is the number of nodes in the set D_n, and the $\bar{s}_d$ indicates the current trust score of data item d obtained in the first step of the proposed trust computation framework (see ① in Fig. 3).

As we described, the next trust score of a network node is computed by the aggregation of its current and intermediate trust scores (see ⑤ in Fig. 3). These trust scores are aggregated using a weighted sum as follows:

$$\bar{s}_n = c_n s_n + (1 - c_n)\hat{s}_n \tag{2}$$

where c_n, $0 \leq c_n \leq 1$ is a constant which represents the relative impacts of trustworthiness from the current streaming window versus the previous one. In other words, if c_n is small, the trust scores of network nodes can change fast; if c_n is large, the trust scores will change more slowly from one window to the next.

Computing Data Trustworthiness The trustworthiness of a data item d depends on its value v_d and provenance p_d. Moreover, there are three trust scores for a data item d: the current, the intermediate, and the next scores, denoted by s_d, $\hat{s}_d$, and $\bar{s}_d$, respectively.

Current Trust Score s_d The current trust score of a data item d is obtained by aggregating the current trust scores of nodes within its provenance. In the proposed approach, the minimum of the current scores of the nodes in p_d is used as the current trust score. This can be explained by the fact that the trustworthiness of a data item can be dominated by the minimum trustworthy node among all nodes which such a data item has passed through.

If the data item d has a simple provenance, the current trust score s_d is simply computed using the minimum value of current trust scores of nodes in p_d. However, when the data item has an aggregate provenance, it is needed to take into account the nodes with more than one child in p_d. To address this problem, the average of the current trust scores of child nodes is used as their aggregate score. Therefore, these child nodes can be considered as a single child node with a trust score equal to the average of the original child nodes. Using this method, an aggregate provenance is formed as a simple provenance for the trust computation.

Intermediate Trust Score $\hat{s}_d$ An intermediate trust score of data item d, denoted by $\hat{s}_d$ is computed based on the data value similarities and its provenance similarities with other data items reported for the same event. it is assumed that D is the set of data items reported for the same event with d.

In order to compute the value similarity for a data item d with value v_d, the proposed approach uses the assumption that the data values in D are normally distributed and the mean and variance are μ and σ^2, respectively. Therefore, the cumulative probability of the normal distribution is employed to compute the similarity of data value v_d with other values within D. Basically, the computation gives high trust scores to the values close to the mean. Thus, the initial $\hat{s}_d$ is computed as follows:

$$\hat{s}_d = 2 \int_{v_d}^{\infty} f(x)dx \tag{3}$$

As shown in Fig. 4a, the shaded area represents the trust score $\hat{s}_d$ obtained from Eq. (3). Clearly, the intermediate trust score is obtained by considering only the data value similarity. Thus, it is needed to adjust the computation to reflect the provenance similarity of the data item as well. The impact of provenance similarity on the trust score computation is computed based on some intuitive observations, listed in Table 1. For example, it is clear that different provenances

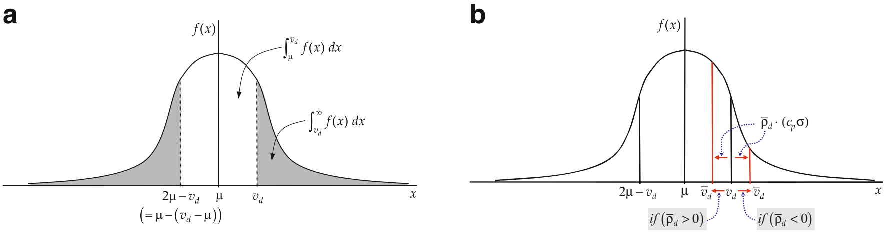

Fig. 4 Computing the intermediate trust score $\hat{s}_d$. (**a**) Intermediate trust score. (**b**) Intermediate trust score adjusted with provenance

Table 1 Impact of provenance similarity on adjusting $\hat{s}_d$

	Similar Data Value	Different Data Value
Similar Provenance	score ↑	score ↓↓↓
	small positive effect	large negative effect
Different Provenance	score ↑↑↑	score ↓
	large positive effect	small negative effect

of similar data values may increase the trustworthiness of data items. Accordingly, a normalized adjustable similarity value is defined for the similarities of the provenance of a data item d with all other data items in D, denoted by $\bar{\rho}_d$. More details can be found in a previous work on provenance-based trustworthiness assessment (Lim et al., 2010).

The adjusted similarity value $\bar{\rho}_d$ reflects the impact of the provenance p_d on the trust computation of the data item d. Thus, it is used to adjust the data value v_d to a new value $\bar{v}_d$ as follows:

$$\bar{v}_d = \min\{v_d - \bar{\rho}_d(c_p.\sigma), \mu\} \tag{4}$$

where c_p is a constant value greater than 0.

Now, the data value v_d in the Eq. (3) is replaced by the $\bar{v}_d$ to adjust the intermediate trust computation for data item d. Thus,

$$\hat{s}_d = 2\int_{\bar{v}_d}^{\infty} f(x)dx = 1 - \int_{2\mu-\bar{v}_d}^{\bar{v}_d} f(x)dx \tag{5}$$

Figure 4b shows how the adjusted similarity value $\bar{\rho}_d$ reflects the value similarity computation.

Next Trust Score $\bar{s}_d$ After computing the current and intermediate trust scores for a data item d, a weighted summation of these two trust values is used to compute the next trust score of data items, denoted by $\bar{s}_d$ (see ② in Fig. 3), Thus,

$$\bar{s}_d = c_d s_d + (1 - c_d)\hat{s}_d \tag{6}$$

where c_d is a constant, $0 \leq c_d \leq 1$, which defines how fast the data trustworthiness evolves as the cycle is repeated.

2.3 Experimental Evaluation

In this section, we briefly summarize the evaluation results from Lim et al. (2010) concerning the effectiveness of the proposed trust computation approach. The experiments were conducted by simulating the sensor networks and generating

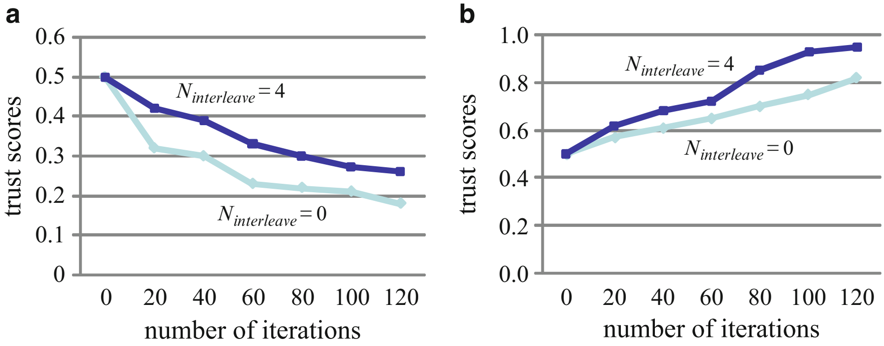

Fig. 5 Change of the trust scores for false data items. (**a**) With false data items, (**b**) with trustworthy data items

synthetic data. For observing the impact of provenance similarity, an interleaving factor was defined which means the interval between the assigned leaf nodes for generating data items in the simulated sensor network. In order to evaluate the effectiveness of the proposed solution, Lim et al. (2010) simulated the injection of false data items into the network and investigated how the proposed cyclic approach reflects this situation in the computation of the trust scores.

Figure 5a (from Lim et al., 2010) shows that when the false data items are injected, the trust scores change rapidly for smaller interleaving factors. This can be explained by the principle that different values with similar provenances rapidly reduce the trust scores (see Table 1). On the other hand, one can see in Fig. 5b that when the correct data items are injected again, the trust scores are increased more rapidly for larger interleaving factors. The reason is that similar values with different provenances result in a large positive effect (see Table 1).

2.4 Summary

This concludes our brief summary of the cyclic trust computation framework proposed in Lim et al. (2010). In Lim et al. (2012) the authors have proposed a game-theoretical defence strategy to protect sensor nodes from attacks and to guarantee a higher level of trustworthiness for sensed data. However, such approach can be compromised with collusive (collaborative) attacks which target the sample mean and variance of the data. In Sect. 4 we demonstrate this and then propose a safer solution based on Iterative Filtering algorithms.

3 IF Algorithms of Laureti et al. and De Kerchove et al.

A relevant class of algorithms for the assessment of information trustworthiness is presented by the *iterative filtering (IF) algorithms*. Pioneering algorithms of such kind were first proposed by Laureti, Moret, Zhang and Yu in their papers appearing in 2006 (Laureti et al., 2006; Yu et al., 2006). Their work was a motivation for the subsequent work of C. De Kerchove and P. Van Dooren in 2007 de Kerchove and Van Dooren (2007) and later in De Kerchove and Van Dooren (2008, 2010). Independently Ignjatovic rediscovered IF algorithms in 2007 (published in 2008, Ignjatovic, Foo, & Lee, 2008) and later introduced other novel algorithms in Lee et al. (2009), Lee, Rodrigues, Kazai, Ignjatovic, and Milic-Frayling (2009), Ignjatovic, Lee, Compton, Cutay, and Guo (2009), Chou, Ignjatovic, and Hu (2013).

The aims of IF-based data aggregation methodologies should be

1. to provide an aggregate value with a provably minimal variance due to stochastic errors of the sources;
2. to insure robustness against non-stochastic errors ranging from hardware faults to collusion attacks from some of the sources, with provable estimates of the level of robustness in terms of the fraction of misbehaving sources.

Moreover, such methodologies should be applicable to both numerical and non-numerical data.

We now explain the essence of IF algorithms using an example of a conference Chair. While such a problem is clearly not among the most pressing ones in the area of data aggregation, its familiarity to the reader makes it a very convenient example to explain both the challenges and our methods.

Let us assume that you are the Chair of a conference, and your referees have done their job: each paper has been reviewed by several referees and every referee has reviewed several papers and you got the scores. However, you suspect that some of the referees might have been unreasonably harsh with their marks; some others might have been sloppy, barely having looked at the papers and thus likely to have made large random errors. Worse, you are worried that some of your referees might have colluded in order to promote the papers of their friends and trash the papers of those against whom they might hold grudges. How should you aggregate the referee's scores and decide which papers to accept in the fairest possible way?

To analyze such a problem, let us assume that there are R referees marking P submitted papers, and, for the sake of simplicity of formulate, let us assume an unusual situation in which each referee marks every single paper. We denote by $M(r,p)$ the mark given by a referee r to a paper p. The main feature shared by most of IF algorithms is that they simultaneously produce approximations of the final aggregate values $\vec{\mu} = \langle \mu(p) : 1 \leq p \leq P \rangle$ (in the present case marks of papers) as well as trustworthiness ranks for the sources $\vec{\tau} = \langle \tau(r) : 1 \leq r \leq R \rangle$ (in this case referees), in a single iterative procedure.

An IF algorithm would typically start by giving all referees the same initial trustworthiness $\tau^{(0)}(r) = 1$ and obtain the initial approximation of the aggregate mark for each paper p as the simple mean of the marks of all referees, $\mu^{(0)}(p) = \sum_{r=1}^{R} M(r, p)/R$. Now, in turn, each referee can be judged on how accurate her marks are, by computing how close her marks are to such an initial approximation of the aggregate marks $\overset{\rightarrow}{\mu}^{(0)}$. Thus, we compute for each referee r the Euclidean distance $d^{(0)}(r) = \sqrt{\sum_{p=1}^{P} (M(r, p) - \mu^{(0)}(p))^2}$ between her marks $\langle M(r, p) \,:\, 1 \leq p \leq P\rangle$ and the aggregate values $\overset{\rightarrow}{\mu}^{(0)} = \langle \mu^{(0)}(p) : 1 \leq p \leq P\rangle$.

Since the trustworthiness of each referee should be inversely related to her distance (or deviation) $d^{(0)}(r)$, we pick a monotonically decreasing *penalty function* $F(d)$ and define the new estimate of trustworthiness of referee r as $\tau^{(1)}(r) = F(d^{(0)}(r))$. In the next round of iteration we obtain a new estimate $\overset{\rightarrow}{\mu}^{(1)}$ of the marks of papers as a weighted average of the marks of all referees, with the marks of a referee r taken with a weight $w^{(1)}(r)$ proportional to a referee's trustworthiness $\tau^{(1)}(r)$. In this way the outliers will be penalized, because their distance to the coarse, initial approximation $\overset{\rightarrow}{\mu}^{(0)}$ of the aggregate marks will be the largest and thus their trustworthiness and corresponding weight the smallest (but no outlier is ever completely excluded!). This process is iterated until it has, hopefully, converged, i.e., for a given precision threshold ε,

while $\sqrt{\sum_{1 \leq p \leq P} (\mu^{(n+1)}(p) - \mu^{(n)}(p))^2} > \varepsilon$ **repeat:**

$$d^{(n)}(r) = \sqrt{\sum_{1 \leq p \leq P} (M(r, p) - \mu^{(n)}(p))^2}; \quad \text{—computing the distance between } r's \text{ marks and estimate } \overset{\rightarrow}{\mu}^{(n)} \tag{7}$$

$$\tau^{(n+1)}(r) = F(d^{(n)}(r)); \qquad \text{— computing the new trustworthiness of r} \tag{8}$$

$$w^{(n+1)}(r) = \frac{\tau^{(n+1)}(r)}{\sum_{1 \leq r' \leq R} \tau^{(n+1)}(r')}; \quad \text{— computing } r's \text{ weight by normalising } r's \text{ trustworthiness} \tag{9}$$

$$\mu^{(n+1)}(p) = \sum_{1 \le r \le R} w^{(n+1)}(r)\, M(r,p), \quad - \text{ computing new estimate of the marks } \vec{\mu} \tag{10}$$

When such iteration terminates after, say, t many rounds of iteration, we get not only the aggregate values of marks of papers $\mu(p) = \mu^{(t)}(p)$ but also an estimate of the trustworthiness of the referees $\tau(p) = \tau^{(t)}(r)$. As we will see, choosing "the best" function $F(x)$ which provides an inverse relationship between distances and trustworthiness ranks is a tricky problem; the most commonly used functions are:

$$(i) \quad F(d(r)) = \frac{1}{d^2(r)}; \quad (ii) \quad F(d(r)) = e^{-d(r)}; \quad (iii) \quad F(d(r)) = 1 - k \cdot d(r),$$

where k appearing in the third function is allowed to be different for each round of iteration, and is chosen so that if r' is the referee with the largest (square of the) distance $d^{(n)}(r')$, then $F(d^{(n)}(r')) = 0$. We now briefly discuss the performance of the above algorithm with the first, reciprocal penalty function; other choices suffer from their own problems.

If (in a simulation experiment) each referee produces true marks plus some independent Gaussian noise with no bias and with variance v_r, then the performance of the above algorithm depends on the distribution of the variances v_r of the referees. For some distributions the algorithm produces an unbiased estimate of the true values with a variance which is remarkably low and essentially equal to the lowest possible variance as dictated by Information Theory, reaching the Cramer-Rao lower bound (CRLB). Note that in such a case the Maximum Likelihood Estimator (MLE) also reaches the CRLB; however, unlike the MLE, the above algorithm **does not** require prior knowledge of the variances of the referees; in fact, this particular form of the algorithm with the reciprocal function can be seen as alternating between estimations of variances of the referees (step 7) and applications of MLE with such estimated approximate variances (step 10).

4 Collusion Attacks

Although the above IF algorithm exhibits better robustness compared to the simple averaging techniques, for some distributions of variances the performance of this algorithm is very bad, with the algorithm producing an estimate of the true marks equal to the marks assigned by one of the referees. The reason for such a behavior is that the penalty function $F(d) = 1/d^2$ has a pole at $d = 0$, and thus the marks of referees act as *attractors* for the iterative procedure: if in the process of iteration the estimated marks get sufficiently close to the marks of any particular referee, the iterative procedure converges in only a few additional steps to the marks provided by that particular referee.

Worse, we have shown Rezvani, Ignjatovic, Bertino, and Jha (2013), Rezvani et al. (2015), such behavior makes the algorithm extremely vulnerable to a collusion attack. Assume that there are R referees among whom C are colluders. The colluders first do their best to estimate the true marks t_p; then $C-1$ of them report heavily skewed marks s_p while the last colluder reports values $((R-C+1)t_p + (C-1)\ s_p)/(R-1)$ as his marks. In such a case the first iteration of the procedure, which takes the mean of all marks, is very likely to produce aggregate marks very close to the marks proposed by the last attacker, causing the algorithm to quickly converge to the marks of the last attacker whose marks are still considerably skewed.

5 Data Aggregation with Protection from Collusions

In order to overcome such instability of the above IF algorithm and make it applicable to compressive sensing in wireless sensor networks in the presence of sensor faults, Chou et al. proposed Chou et al. (2013) to modify the penalty function by adding a small regularisation constant $a>0$ and define $F(d) = 1/(d^2+a)$. While this does make the algorithm more robust, it also has a serious drawback: if a is sufficiently large to make the algorithm stable, then the values returned by the algorithm might not differ significantly from the simple mean of the marks of all sources.

In trying to solve this problem in a more satisfactory manner, Rezvani et al. have proposed Rezvani et al. (2015) a better way to provide an initial approximation $\vec{\mu}^{(0)}$. Clearly, without knowing the true values, the algorithm cannot determine the error of each source; however, denoting again the true value of item p (in our example the true mark of a paper p) as t_p, we have that for every pair of sources r_1, r_2 (in the above example referees),

$$\begin{aligned}\sum_{1\le p\le P}\frac{(M(r_1,p)-M(r_2,p))^2}{P} &= \sum_{1\le p\le P}\frac{((M(r_1,p)-t_p)-(M(r_2,p)-t_p))^2}{P}\\ &= \sum_{1\le p\le P}\frac{(M(r_1,p)-t_p)^2}{P} + \sum_{1\le p\le P}\frac{(M(r_2,p)-t_p)^2}{P}\\ &\quad + 2\sum_{1\le p\le P}\frac{(M(r_1,p)-t_p)(M(r_2,p)-t_p)}{P}.\end{aligned} \tag{11}$$

The first two terms on the second line are estimators for the variances v_{r_1} and v_{r_2}, and, assuming that the errors of the sources are reasonably uncorrelated, the last term on the second line should be small. In this way we obtain $\sum_{1\leq p\leq P}(M(r_1,p)-M(r_2,p))^2 \approx v_{r_1}+v_{r_2}$, which results in $R(R-1)/2$ equations in R variables $v_1, v_2, \ldots, v_R$, that can be solved in the sense of the Least Squares. We can now take as the initial approximation $\vec{\mu}^{(0)}$ of the marks the MLE estimation with the obtained approximations of the variances v_r, i.e.,

$$\mu^{(0)}(p) = \frac{\sum\limits_{1\leq r\leq R} \frac{M(r,p)}{v_r}}{\sum\limits_{1\leq r\leq R} \frac{1}{v_r}}. \tag{12}$$

Remarkably, experiments have demonstrated that, even when the errors are significantly correlated, such initial value dramatically improves the stability of the algorithm without any sacrifice in performance. It also improves its robustness against a collusion attack, because the attackers have no way of estimating the variances of other referees (Rezvani et al., 2015). However, in general, the above algorithm can have several fixed points (de Kerchove & Van Dooren, 2010); for that reason, since it does not provide a unique solution, it is not suitable for a real life deployment. Moreover, the algorithm has another serious drawback: it is not applicable to non-numerical data because it crucially depends on using a distance function, $d(r)$.

For that reason the present authors have looked for IF algorithms which are both provably convergent and also applicable to non-numerical data. This was partly addressed by Allahbakhsh and Ignjatovic (2015), Allahbakhsh et al. (2015), Allahbakhsh, Ignjatovic, Benatallah, and Motahari-Nezhad (2013) by altering the main feature of the previously introduced IF algorithms, namely by separating the process of assessment of the trustworthiness of the sources from the actual data aggregation process. We explain the main idea using a Q&A website example.

At a typical *Q & A* website each question is open for new answers for a certain period of time, say 30 days, before the question is closed; users are allowed to vote for the best answer to a particular question for an additional period of time, say 10 days, before the votes are counted and the best answer is declared. In general, there are other, concurrently open questions on the same topic and, as it can be easily observed on such websites, users with the same interest tend to vote for the best answer to a number of questions in the same field, open during the past 30 days or so. For that reason, the following policy of such a social website would not be very restrictive: only the votes of members who are "active" at the time are taken into account, and a member is considered active if he or she has cast her vote for the best answer to a certain number of questions $Q > 1$ which were recently closed. This gives an opportunity to make vote aggregation significantly more robust by deciding **simultaneously** which are the best answers to all questions which have been recently closed, using the following algorithm proposed in

Allahbakhsh and Ignjatovic (2015), Allahbakhsh et al. (2015), Allahbakhsh et al. (2013) by the present CI and his student.

Assume that there are Q recently closed questions; for each question q_i we have a corresponding list Λ_i of n_i answers, $\Lambda_i = \langle a(i,1), a(i,2), \ldots, a(i,n_i)\rangle$. We also assume that there are V voters $v_1, v_2, \ldots, v_V$. Again, for the simplicity of presentation, we assume that each voter has chosen her best answer for every question; for a sparse pattern of votes all quantities involved can be appropriately normalized, according to the total number of questions each voter has participated in choosing the best answer for, see Allahbakhsh and Ignjatovic (2015), Allahbakhsh et al. (2013), Allahbakhsh et al. (2015). The algorithm for vote aggregation is again iterative, and it simultaneously evaluates the ratings $\rho(i,k)$ of all answers to each question posed in the given interval of time as well as the trustworthiness $\tau(m)$ of each voter v_m who participated in voting during that period of time, in the following manner:

Let p be a real number, $p \geq 1$, and let us denote by $m \rightarrow i,k$ the fact that voter v_m has voted for the answer $a(i,k)$ as the best answer to question q_i. In the initial round of iteration, for each question q_i and all of its answers $a(i,k), \quad 1 \leq k \leq n_i,$ we simply count the number $\nu(i,k)$ of votes which $a(i,k)$ has received. We now obtain the initial ranks of answers as the normalized number of votes, $\rho^{(0)}(i,k) = \nu(i,k)/\sqrt{\sum_{1\leq j\leq n_i} \nu(i,j)^2}$; thus, for all answers $a(i,k)$ to a question q_i we have $\sum_{1\leq k\leq n_i} \rho^{(0)}(i,k)^2 = 1$. We are now again in a position to judge for every voter v_m how good his choices are, namely, to what degree their voting is in agreement with the community sentiment, and assign to them his initial trustworthiness $\tau^{(0)}(m) = \sum_{i=1}^{Q} \{\rho^{(0)}(i,k) : m \rightarrow i,k\},$ which is simply a sum of the normalized number of votes received by all the answers which he voted for. Clearly, a voter v_m will get a large initial trustworthiness only if he has chosen answers which many other community members have also chosen. In the next round of iteration of our vote aggregation procedure not every vote has an equal value, but its value depends on the trustworthiness of the voter. Thus, at every consecutive stage of iteration $n + 1$ we have:

$$\tau^{(n+1)}(m) = \sum_{1\leq i\leq Q} \{\rho^{(n)}(i,k) : m \rightarrow i,k\}; - \text{ computing the trustworthiness of voter } v_m$$

$$\rho^{(n+1)}(i,k) = \frac{\sum_{m\,:\,m\rightarrow ik} \left(\tau^{(n+1)}(m)\right)^p}{\sqrt{\sum_{1\leq j\leq n_i} \left(\sum_{m\,:\,m\rightarrow ik} \left(\tau^{(n+1)}(m)\right)^p\right)^2}};$$

$$- \text{ computing the new rank of answer } a(i,k) \text{ to question } q_i$$

iterating until $\sum_{1\leq m\leq V} (\tau^{(n+1)}(m) - \tau^{(n)}(m))^2 < \varepsilon$. We note that the purpose of the denominator in the expression for $\rho^{(n+1)}(i,k)$ is a normalization which keeps the iteration stable and allows an elegant convergence proof by ensuring that at every stage of iteration $\sum_{1\leq k\leq n_i}\rho^{(n)}(i,k)^2 = 1$, see Allahbakhsh and Ignjatovic (2015). The parameter p controls filtering; the larger the value of p the more the algorithm is robust against collusion attacks, but larger values also increasingly marginalize honest voters who do not vote entirely in accordance with the prevailing sentiment of the community.

With such a vote aggregation procedure the colluding voters must vote for the best answer for a significant number of other questions posed during the same period of time, and they cannot vote randomly, but must vote in accordance with the prevailing sentiment of the community, in order to receive sufficient trustworthiness. Only then can they vote differently from other voters for the answer to the question they are attacking, and hope that they can prevail over the honest voters. While this does not preclude entirely collusion attacks, it obviously makes them harder to execute.

Also note that in this case the data (the choice of the best answer) is not only non-numerical but also does not have any natural ordering. However, the same algorithm is applicable to numerical choices with values which are integers in a limited range as well as ordered choices. For example, customer feedback is usually in the range of one to five "stars" and the same applies to movie ranking. Market analyst's recommendations are an example of non-numerical but ordered choices (strong_buy < buy < neutral < sell < strong_sell). After such an iterative procedure has converged and ranks $\rho(i,k)$ of all choices have been determined, in case of numerical data one can form a weighted average of such numerical choices, with weights obtained from the ranks; in case of ordered choices it can be left to the user to choose the particular numerical values for the ordered alternatives to reflect user's preferences, and then obtain the aggregate value as a corresponding weighted average.

Allahbakhsh at al. proved that the above algorithm always converges, and extensive tests not only on simulated data but also on real data, such as the publicly available movie rating dataset *MovieLens*, have shown that in terms of robustness against large collusion attacks such an algorithm outperforms the previous IF algorithms, see Allahbakhsh et al. (2015), Allahbakhsh et al. (2013).

Moreover, for cases where we can also rely on historic data, or in a case of a refereeing process where each referee can declare his level of competence for each paper, such additional information can be included into the iterative procedure of such an algorithm in a way that preserves the proof of convergence (Allahbakhsh et al., 2013).

The continuous case, such as aggregation of measurements of sensors, appears to be a significantly harder problem. An aggregation algorithm must be robust against collusion attacks without sacrificing its performance when the sources have only stochastic errors. In fact, even in the presence of a collusion attack, if the fraction of the colluding sources is reasonably small, the algorithms should provide output values which are close to the optimal, MLE estimate based on the data obtained from the sources with stochastic errors only. Rezvani et al. have designed an algorithm which, in extensive tests, appears to meet these requirements (Rezvani, Ignjatovic, Bertino, & Jha, 2014a). This algorithm is based on an idea of propagation of credibility $cr(r)$ of one source to another source. It again takes the simple mean as the initial approximation of the aggregate values $\mu^{(0)}(p)$ and assigns equal initial variance estimates $v^{(0)}(r) = \frac{1}{(P-1)R}\sum_{s=1}^{R}\sum_{p=1}^{P}(M(s,p)-\mu^{(0)}(p))^2$ to all sources; we then repeat until convergence:

$$\mathrm{cr}^{(n+1)}(r) = \left(\prod_{j=1}^{R}\frac{\exp\left(-\frac{\frac{1}{P-1}\sum_{1\le p\le P}(M(r,p)-\mu^{(n)}(p))^2}{2v^{(n)}(j)}\right)}{\sqrt{2\pi v^{(n)}(j)}}\right)^{\frac{1}{R}}; \tag{13}$$

– computing the credibility of source r

$$\mu^{(n+1)}(p) = \sum_{i=1}^{R}\frac{\mathrm{cr}^{(n+1)}(i)}{\sum_{k=1}^{R}\mathrm{cr}^{(n+1)}(k)}M(i,p); \quad \text{– computing the new aggregate values} \tag{14}$$

$$\mathrm{var}^{(n+1)}(r) = \frac{1}{P-1}\sum_{k=1}^{P}(M(i,k)-\mu^{(n+1)}(k))^2 \quad \text{– computing the new variance of source}r \tag{15}$$

Thus, at each stage of the iteration, the credibility of the values supplied by a source r is assessed by estimating the likelihood that the values supplied by r might have been obtained by every other source. The credibility is defined as the geometric mean of all of these likelihoods; see Eq. (13). The heuristic underlying such methodology is that the stability of such algorithm should come from the smoothing property of taking a mean of all of these likelihoods. The geometric mean was chosen with a hope that to be able to rigorously prove that, in case of purely stochastic normally distributed unbiased errors, the algorithm converges to the MLE estimation which could have been obtained if the non-colluding sources and their exact variances were a priori known; this would clearly ensure that our algorithm has the minimal possible variance, equal to the CRLB. Figure 6 shows a typical result obtained with 25 sources; 20 sources are “honest” providing the true mark t_p of item p plus a normally and independently distributed unbiased noise with randomly chosen variances between 1 and 5. The remaining 5 sources collude, with the first 4 sources reporting skewed values $s_p = 3t_p$ and the fifth colluder the mean $((R-C+1)t_p + (C-1)\,s_p)/(R-1)$.

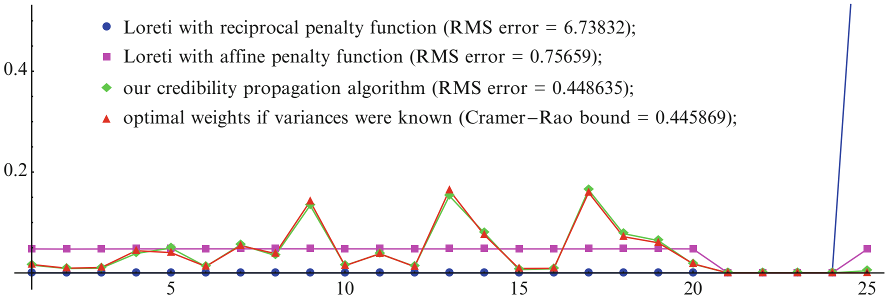

Fig. 6 Reciprocals of normalized variances of sources, estimated using: IF with $F(d) = 1d^2$ (*filled circle*), IF with $F(d) = 1 - k\ d$ (*filled square*), credibility propagation (*filled diamond*), normalized reciprocals of the true variances (*filled triangle*). Also shown are the corresponding RMS value of errors of the aggregate values (discrete values are joined by *lines* for better visual representation)

As it can be seen from Fig. 6, the weights obtained by the IF algorithm with the reciprocal penalty function $1/d^2$, (filled circle), are all essentially zero except for the weight of the last attacker which is 1 (out of range on the graph); the weights obtained by IF algorithm with the affine penalty function $F(d) = 1 - k\ d$, (filled square), are 0 for all attackers except the last one, but all other, non zero weights are essentially equal thus resulting in the simple mean of all honest sources and the last attacker. Finally, the weights produced by the algorithm based on the credibility propagation (filled diamond) are almost indistinguishable from the (normalized) reciprocals of the true variances of the “honest” sources (filled triangle), which in this case represent the optimal weights resulting in an estimator with the smallest possible variance. The RMS values of errors shown on the legend of Fig. 6 demonstrate the superiority of the credibility propagation algorithm. In fact, several IF algorithms—more than a dozen of them—were implemented and test and in all cases the algorithm by Rezvani et al. had the lowest RMS error, only slightly higher than the CRLB, even in the presence of a collusion attack. A *Mathematica* code which produced the above results is available online at http://www.cse.unsw.edu.au/~ignjat/IF.nb.

In addition, Rezvani et al. have applied ideas of the provenance of data (Lim et al., 2010) to design an iterative algorithm for computing the risk of flows and hosts in a computer network (Rezvani, Ignjatovic, Bertino, & Jha, 2014b; Rezvani, Ignjatovic, & Jha, 2013; Rezvani, Sekulic, Ignjatovic, Bertino, & Jha, 2014). For such iterative risk assessment algorithm as introduced in Rezvani et al. (2014b), Rezvani et al. were able to prove its convergence and also obtain sharp analytic estimates for its performance (Rezvani et al., 2014). Future research will aim to integrate the idea of provenance of data with IF algorithms in a single (possibly nested) iterative procedure. Such an integration should be done in a way which preserves the convergence proof of the resulting algorithm

6 Research Roadmap

In many real-life distributed systems such as social networks, rating system, participatory sensing networks and WSNs, the trustworthiness of participants has a significant role in the decision-making processes. While we believe that past results have demonstrated the potential of our IF algorithms as a robust trust framework for these distributed systems, achieving the objective requires much wider research efforts.

Most IF algorithms are still mostly "ad hoc" solutions which do not have a unified mathematical foundation. For example, in the discrete case we still lack an algorithm which, in case of domains which are integers (for example one to five star ratings) takes into account the proximity of votes, rather than just the coincidence of votes. This is clearly unsatisfactory: if a number of voters give a five star ranking to a movie, then a voter which gives it four stars should get some credit from them, and certainly more credit than a voter which gives the same movie only three stars. However, in algorithms by Allahbakhsh and Ignjatovic (2015), Allahbakhsh et al. (2015) both such dissent voters get no credit from the voters giving the movie five stars. Moreover, the degree of such credibility propagation from a voter to the voters who propose similar but not equal scores should depend on the estimated variances of the voters. It is also crucial that domain knowledge be incorporated into the data trustworthiness methodologies. For example, in a sensor network, a sensor that has been deployed for a long time may be considered less trustworthy than recently deployed sensors. Also metrics and methodologies from the area of data quality should be considered here (Reznik & Bertino, 2013).

In some distributed systems such as participatory sensing networks, preserving the privacy and anonymity of participants is mandatory (Wang, Cheng, Mohapatra, & Abdelzaher, 2013). Clearly, if the participatory networks fully anonymize the reported data, it is difficult to accurately estimate the trustworthiness of participants using the current state of our IF algorithms. Decentralization of our trust computation approach could improve the privacy of participant (Hasan, Brunie, Bertino, & Shang, 2013). Thus, proposing a decentralized privacy preserving IF algorithm for robust trust computation is an interesting open research area.

A tremendous volume of data generated by recent technological advances, referred to as *Big Data* can be used to provide data-driven decision-making. Moreover, the interconnected Big Data forms a large data redundancy which can be used to validate data trustworthiness (Labrinidis & Jagadish, 2012). An interesting research direction is to scale the IF algorithms to Big Data in order to extract hidden relationships within the data redundancy.

We will investigate applications of our IF algorithms other than just data aggregation or ranking. One such application was already implemented and tested as a part of an Honors Thesis project (D'Souza, 2011), where it was used to produce a novel recommender system. Taking as an example movie ranking, our algorithm aggregates ratings of movies provided by users, and, as we have explained, besides producing robust ratings of movies it also produces weights for users which reflect to what degree their ratings agree with the prevailing “community sentiment” ranks, as produced by our IF algorithm. We now use the observation that if two users have similar tastes, their weights must also be similar, because their movie ratings, being close to each other, must also be at a similar distance to the community sentiment ranks. Thus, to make recommendations for a particular user, we can restrict our attention only to users whose weights are close to the weight of that particular user.

In conclusion, we believe that the IF algorithms have demonstrated a promising potential for providing robust trust assessment methods for inconsistent information. Moreover, such algorithms provide a robust aggregate of such inconsistent information and can thus play a critical role in WSNs as a method of resolving a number of important problems, such as secure routing, fault tolerance, false data detection, compromised node detection, cluster head election, and outlier detection. They are also applicable to social networks, web services, and many other fields which involve inconsistent information.

References

Allahbakhsh, M., & Ignjatovic, A. (2015). An iterative method for calculating robust rating scores. *IEEE Transactions on Parallel and Distributed Systems, 26*(2), 340–350.

Allahbakhsh, M., Ignjatovic, A., Motahari-Nezhad H.R., Benatallah, B (2015). Robust evaluation of products and reviewers in social rating systems. *World Wide Web* 18(1):73–109

Allahbakhsh, M., Ignjatovic, A., Benatallah, B., & Motahari-Nezhad, H. R. (2013). Robust evaluation of products and reviewers in social rating systems. *WorldWide Web.*

Chou, C.-T., Ignjatovic, A., & Hu, W. (2013). Efficient computation of robust average of compressive sensing data in wireless sensor networks in the presence of sensor faults. *IEEE Transactions on Parallel and Distributed Systems, 24*(8), 1525–1534.

Dai, C., Rao, F.-Y., Ghinita, G., & Bertino, E. (2011). Privacy-preserving assessment of location data trustworthiness. In *Proceedings of the 19th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, GIS ’11 (pp. 231–240).

Dai, C., Rao, F.-Y., Truta, T. M., & Bertino, E. (2012). Privacy-preserving assessment of social network data trustworthiness. In *CollaborateCom* (pp. 97–106).

de Kerchove, C., & Van Dooren, P. (2007). Iterative filtering for a dynamical reputation system. *CoRR* [abs/0711.3964].

De Kerchove, C., & Van Dooren, P. (2008). Reputation systems and optimization. *Siam News, 41*(2), 2008.

de Kerchove, C., & Van Dooren, P. (2010). Iterative filtering in reputation systems. *The SIAM Journal on Matrix Analysis and Applications, 31*(4), 1812–1834.

D’Souza, N. (2011). Applications of adaptive averages: Recommendation systems. Honours Thesis, UNSW.

Hasan, O., Brunie, L., Bertino, E., & Shang, N. (2013). A decentralized privacy preserving reputation protocol for the malicious adversarial model. *IEEE Transactions on Information Forensics and Security, 8*(6), 949–962.

Ignjatovic, A., Foo, N., & Lee, C. T. (2008). An analytic approach to reputation ranking of participants in online transactions. In *IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology (WI).*

Ignjatovic, A., Lee, C. T., Compton, P., Cutay, C., & Guo, H. (2009). Computing marks from multiple assessors using adaptive averaging. In *International Conference on Engineering Education (ICEE)*.

Labrinidis, A., & Jagadish, H. V. (2012). Challenges and opportunities with big data. *The Proceedings of the VLDB Endowment, 5*(12), 2032–2033.

Laureti, P., Moret, L., Zhang, Y.-C., & Yu, Y.-K. (2006). Information filtering via Iterative Refinement. *EPL (Europhysics Letters), 75*, 1006–1012.

Lee, C.-T., Milic-Frayling, N., Vinay, V., Rodrigues, E.M., Ignjatovic, A., & Kazai, G. (2009). Measuring system performance and topic effectiveness using generalized means with adaptive weights. In *ACM Conference on Information and Knowledge Management (CIKM)* (pp. 2033–2036). New York, NY, USA: ACM.

Lee, C.-T., Rodrigues, E.M., Kazai, G., Ignjatovic, A., & Milic-Frayling, N. (2009). Model for voter scoring and best answer selection in community q & a services. In *IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology (WI)* (pp. 116–123). New York, NY, USA: ACM.

Lim, H.-S., Ghinita, G., Bertino, E., & Kantarcioglu, M. (2012). A game-theoretic approach for high-assurance of data trustworthiness in sensor networks. In *2012 I.E. 28th International Conference on Data Engineering (ICDE)* (pp. 1192–1203).

Lim, H.-S., Moon, Y.-S., & Bertino, E. (2010). Provenance-based trustworthiness assessment in sensor networks. In *Proceedings of the Seventh International Workshop on Data Management for Sensor Networks*, DMSN ’10 (pp. 2–7).

Reznik, L., & Bertino, E. (2013). Poster: Data quality evaluation: Integrating security and accuracy. In *Proceedings of the 2013 ACM SIGSAC Conference on Computer &#38; Communications Security*, CCS ’13 (pp. 1367–1370). New York, NY, USA: ACM.

Rezvani, M., Ignjatovic, A., Bertino, E., & Jha, S. (2013). Secure data aggregation technique for wireless sensor networks in the presence of collusion attacks. Technical Report UNSW-CSE-TR-201319, School of Computer Science and Engineering, UNSW.

Rezvani, M., Ignjatovic, A., Bertino, E., & Jha, S. (2015). Secure Data Aggregation Technique for Wireless Sensor Networks in the Presence of Collusion Attacks. IEEE Trans. Dependable Sec. Comput. 12(1):98–110.

Rezvani, M., Ignjatovic, A., Bertino, E., & Jha, S. (2014a). Credibility propagation for robust data aggregation in WSNs. Technical Report UNSW-CSE-TR-201414, School of Computer Science and Engineering, UNSW.

Rezvani, M., Ignjatovic, A., Bertino, E., & Jha, S. (2014b). Provenance-aware security risk analysis for hosts and network flows. In *IEEE/IFIP Network Operations and Management Symposium (NOMS)*.

Rezvani, M., Ignjatovic, A., & Jha, S. (2013). Iterative security risk analysis for network flows based on provenance and interdependency. In *IEEE International Conference on Distributed Computing in Sensor Systems (DCOSS 13)* (pp. 286–288).

Rezvani, M., Ignjatovic, A., Bertino, E., & Jha, S. (2015). Secure data aggregation technique for wireless sensor networks in the presence of collusion attacks. *IEEE Transactions on Dependable and Secure Computing, 12*(1), 98–110.

Tang, L.-A., Yu, X., Kim, S., Han, J., Hung, C.-C., & Peng, W.-C. (2010). Tru-Alarm: Trustworthiness analysis of sensor networks in cyber-physical systems. In *Proceedings of the 2010 I.E. International Conference on Data Mining*, ICDM ’10 (pp. 1079–1084).

Wang, X., Cheng, W., Mohapatra, P., & Abdelzaher, T. F. (2013). Artsense: Anonymous reputation and trust in participatory sensing. In *INFOCOM, 2013 Proceedings IEEE* (pp. 2517–2525).

Yu, Y.-K., Zhang, Y.-C., Laureti, P., & Moret, L. (2006). Decoding information from noisy, redundant, and intentionally distorted sources. *Physica A Statistical Mechanics and its Applications, 371*, 732–744.