On Secure Consensus Information Fusion over Sensor Networks

Mahdi Kefayati[†⋆], Mohammad S. Talebi[†], Hamid R. Rabiee[†⋆], Babak H. Khalaj[†]
[†]Sharif University of Technology, [⋆]Iran Telecommunication Research Center (ITRC)
kefayati@ce.sharif.edu, mstalebi@ee.sharif.edu, rabiee@sharif.edu, khalaj@sharif.edu

Abstract

In this work we have examined the problem of consensus information fusion from a novel point of view, challenging the fundamental assumption of mutual trust among the fusion parties. In quest for a method to make information fusion possible while preserving the mutual confidentiality and anonymity of the fused information even in case of collusion of the malicious nodes, we propose the Blind Information Fusion Framework (BIFF). *In BIFF, which is a secure information fusion framework, the nodes are not aware of the actual information they are processing, yet converging to the intended result(s). We formulate BIFF according to the* anonymization transform *and discuss its robustness against collusions for privacy violation. As an example, two secure consensus averaging methods are formulated according to BIFF.*

1 Introduction

Sensor networks have recently received much attention due to their high potential in formation of the next generation information gathering and processing systems. The flexible and scalable nature of them, boosted by agility in their deployment, proposes them as one of the major players of anytime anywhere computing idea.

Most of the past research in sensor networks is concentrated on power-aware networks for information gathering. Recent trends show high potential of sensor networks in formation of distributed information fusion networks [12]. The new paradigms transform sensor networks from mere data gathering communication networks to more intelligent distributed systems which are able to process the information to yield the intended result(s)[11, 14]. In this paper, by a sensor network we mean a general network of arbitrarily connected set of nodes which are sources of information for fusion. In fact, we broaden our view from mere power constrained data gathering networks. Nevertheless, we consider the challenges common in sensor networks including unreliable links, changing topology and power and computation constraints.

There are some fundamental differences between distributed information fusion over sensor networks and the classical information fusion systems and data gathering sensor networks. Sensor networks are loosely coupled, limited capability, infrastructure-less and sometimes untrusted. In contrast, centralized information fusion schemes are considered monolithic and structured with assumed inter-component trust. Data gathering sensor networks share many of the above mentioned challenges with distributed information fusion networks from communication point of view[4]; however, from computing point of view they exhibit different properties[5].

We divide distributed information fusion into two main categories: consensus information fusion or consensus fusion in short and, non-consensus information fusion. In distributed consensus fusion, the goal is to reach an agreement on a definite or estimated value or vector which depends on all members' information. Consensus fusion is one of the most important topics in distributed systems and algorithms because of its applications in coordination and agreement[9] as well as distributed estimation and filtering[12]. An example of consensus information fusion is consensus averaging. We will discuss consensus averaging as our main example throughout this paper due to its importance and applications.

Non-consensus information fusion is obviously more general. In fact, many of the common algorithms in distributed and networked systems can be modeled as a non-consensus distributed information fusion. Routing protocols can be considered an example of non-consensus information fusion in general case as the fusion result, which is the forwarding table, is different for each router.

We also note that information fusion is not always an application over a sensor network. In many cases, it is a mandatory part of the network and/or MAC layer although not vividly stated. Especially for consensus fusion which is our main focus, many of the routing protocols as well as cooperative MAC protocols can be analyzed as a consensus fusion function.

Security and reliability are two of the most challeng-

ing issues in sensor networks[4]. Although many information fusion schemes exhibit a good level of robustness to node failure and topological changes [10, 14], they lack certain features such as information confidentiality and privacy. Here, by confidentiality we mean preserving the information from attackers outside the network or intruders. In sensor fusion, confidentiality can be achieved through cryptographic measures such as using group keys. However, privacy, by which we mean keeping each nodes' information private in the course of the fusion, is a more challenging issue. The challenge mostly stems from the fact that the information must be communicated among the network members to make fusion possible. One may propose using a trusted third party which gathers the information, does the computation and communicates the results. We argue, however, that such solutions are not scalable and robust due to existence of the trusted third party. Also, the trusted third party may not be available in many scenarios since all nodes must have unconditional trust to it. This challenge is addressed in the cryptography literature as "secure multi-party computation (MPC)"[2]. The secure MPC methods are focused on secure consensus fusion and supported by rigorous mathematical bases. Yet, many of such solutions are not suitable for sensor networks since they require a fully connected topology and impose high computation and communication costs[2].

Most security solutions proposed so far for sensor networks mainly consider data gathering sensor networks. The main goal of such methods is maintaining confidentiality, integrity, availability and authenticity of the information in transit[4]. The most important assumption behind all these solutions is the implicit perfect trust between source and destination, or in general case among the fusion party. In other words, they assume that destination is authorized to be aware of the information sent by the source. This may not be the case in some situations. Generally, there may not be mutual trust among the fusion party, although they might be authorized to see the fusion result(s). The main difference of our point of view form the traditional approach in sensor networks is that all methods proposed so far has focused on *communication security*, that is, their goal is to protect the information in transit. In contrast, our focus is on maintaining security in the course of processing or namely *computation security*. These two topics have many overlaps in a distributed system, where communication among the components is inevitable. Nevertheless, there is a fundamental difference between the two since in many cases, perfect knowledge of the processed information and/or its source (the association between the source and the information) is not mandatory.

Secure consensus fusion can be used to form anonymity in some applications as well. Anonymity or more precisely ownership anonymity can be defined as making the association between the source of the information and the information undetectable. In context of secure consensus fusion, the information is anonymized in the sense that the effect is observable in the results but its source is not identifiable. One example can be analysis of the security logs and statistics of various organizations by each other. In log anonymization, the goal is to anonymize the security logs in a way that they can be processed by the security departments; yet, they do not reveal the exact vulnerabilities and incidents in the system. Currently, the common method is obfuscating sensitive parts of the logs like the IP addresses; however, such methods have shortcomings[8]. Another example is election, where, although all the voters are authorized to be aware of the final results, each voter saves its right in keeping his opinion private. In order to solve the problem in todays election and voting systems, anonymous, yet authenticated, ballots and ballot boxes are used which preserves voter privacy through anonymization of the votes. Patient anonymization is another application of secure consensus fusion. In patient anonymization a group of health organizations like hospitals decide to share their patient information for new findings by the analysis of the fused information. However, since patient privacy must be kept, the exact information can not be shared. A common practice is to anonymize the records by obfuscating their identification information. Yet, as a patient may have been recorded at more than one organization, the accuracy of the results might be affected. This is very critical especially for diseases like HIV.

The above examples are given to clarify the role of secure consensus fusion in general sense. In mobile computing and communication applications, there are many potential applications as well. The number of applications will grow substantially considering the trend towards more intelligent distributed information systems like intelligent sensor networks. Electronic voting can be considered as one of the promising applications of secure consensus fusion. Nevertheless, we believe that secure consensus fusion has many of its applications in secure formation of intelligent sensor networks like battlefield networks. Secure election of various roles like leaders is an example. Secure consensus fusion over sensor networks can make implementation of many cryptographic functions over sensor networks such as group digital signatures more scalable by removing the need for a trusted third party or predefined secure base stations for sensors. Calculating network wide parameters such as average remaining energy, number of neighbors and rate of energy consumption securely is another example.

Kefayati *et al.* have addressed the problem of secure information fusion as well as secure consensus averaging in [5], [6] and [7]. In [7], they have proposed a method for secure consensus averaging assuming the fused information are represented as continuous value numbers. Their

method, called *Random Offsets Method (ROM)*, though very light weight and efficient, suffers from steady-state error in fusion results. They have also proposed another method[6] for secure consensus averaging called *Random Projections Method (RPM)* which does not suffer from steady-state error; however, it requires secure channel establishment between neighbors. Both of the methods can be considered as special cases of the framework we present in this paper.

Our work especially addresses the problem of maintaining mutual confidentiality and anonymity in the course of fusion. This issue seems to be a paradox since its simple goal is to find a way to calculate the results while keeping the confidentiality of the processed information and/or their origin. This is why we call it, *blind information fusion*, i.e. the nodes must not see the real information they process. Our work introduces a general direction towards secure information fusion in multi-agent and distributed systems including various kinds of sensor networks. To the best of our knowledge, this is the first work which addresses consensus fusion security from this point of view.

The rest of this paper is organized as follows: In section 2 we will present some preliminaries and introduce a system model for distributed information fusion in sensor networks. Section 3 is dedicated to introduction and analysis of the *Blind Information Fusion Framework (BIFF)*. Finally, in section 4, we conclude the paper and present the future work.

2 Preliminaries and System Model

We model the network as a graph, $G(V, E)$, with $N = |V|$ nodes or analogously, its adjacency matrix, A. We also define N_i as the set of neighbors of the node i, i.e. $j \in N_i \Leftrightarrow (i, j) \in E$, and represent its size by n_i. Also N_i' is defined as $N_i \cup \{i\}$ and $|N_i'| = n_i'$.

Representing each nodes' information with a number, x_i, the ultimate goal of the fusion is calculating $\mathbf{r} = \mathbf{F}(\mathbf{X}) = \mathbf{F}(x_1, \ldots, x_N)$ as the fusion result. In consensus case, all nodes reach the same result, i.e. $\forall i : r_i = r$. For example, in consensus averaging we have $r = \frac{1}{N}\sum_i x_i$. Due to scalability issues, the fusion or goal function is usually implemented in a distributed manner, i.e. we have $r_i = f_i(x_k)$ where $k \in N_i'$, and convergence is achieved through multiple iterations of computation and communication of the intermediate results to the neighbors. One must note that $f_i(x_k)$ can have a totally different structure from $\mathbf{F}(\mathbf{X})$ elements. Also, even in case of consensus fusion, $f_i(.) \neq f_j(.)$ may hold as in adaptive methods[13]. Such forms of distributed fusion are robust and scalable since they replace multi-hop routing with information diffusion.

We focus on cooperative networks where the nodes behave according to a pre-defined protocol to achieve the fusion result(s). Fair election is considered as one of the real world examples of a cooperative systems: though all the voters are interested in a democracy, each voter saves its right for privacy of his/her own vote and therefore, the ballots shall be anonymous and filled privately[1]. Nevertheless, malicious members might be interested in other members' information. In other words, we have focused on methods for maintaining mutual confidentiality of network members' information in the course of the fusion process. Discussion of non-cooperative networks is out of the scope of the current work.

3 Secure Information Fusion

3.1 Problem Definition and Preconditions

We propose *Blind Information Fusion Framework (BIFF)* which gives a general approach to preserving mutual privacy and anonymity in information fusion networks considering an honest-but-curious adversary model. In other words, we propose a solution for the following problem:

> "Let us assume an arbitrarily connected cooperative set of nodes, each having a piece of information and interested in calculation of a function of all the information. How shall we calculate the result in a distributed and scalable manner such that the mutual privacy of all the members is maintained; even if some members collude?"

According to our prior discussion, in consensus fusion, preserving mutual privacy provides ownership anonymity. Privacy and anonymity are tied since a piece of information from all nodes is required for calculation of the goal function. Therefore, from members point of view, it is a matter of privacy while from the information and fusion point of view it is a matter of anonymity. Again, considering the voting problem example can clarify the case.

In a consensus fusion, some of the nodes may collude to reveal other nodes' information by, for example, sharing their information. Therefore, robustness against collusions is one of the most important goodness factors of a secure consensus fusion scheme. We define the *collusion resistance level* as *the upper bound of the nodes that their collusion can not reveal other node(s)' information* to capture robustness of a secure consensus fusion scheme against collusion. As shown in [5], this factor highly depends on the algorithm and may even be related to node parameters such as its connectivity degree.

[1] In general, voters may try to cheat in favor of their choice. This problem can be dealt with cheat proof protocols which can be implemented independently along with the fusion method. There exists primitives for cheater resistant MPC[3] as well.

Before going through our framework we shall discuss two primary conditions in our context without which, privacy and anonymity is meaningless:

- **Goal function anonymity:** The goal function shall be private and anonymous itself, that is, it must not give information about specific sources. This requires the goal function to be one-way with as equiprobable inverses as possible or one-to-one with P complexity in forward and NP complexity in inverse path. Summation is an example of the former and group digital signature based on RSA algorithm is an example of the latter case.
- **Low correlation among sources of information:** There must be low correlation among the sources of information because the information of one node can be guessed with high confidence from the other (colluding) node(s)' information when the correlation among the sources is high enough.

As we will see, the goal function anonymity condition is directly related to collusion resistance of the algorithms proposed for secure information fusion. The goal function anonymity condition can be extended to the *goal function anonymity order* concept which is directly related to the collusion resistance properties of the secure fusion algorithm.

3.2 The Blind Information Fusion Framework (BIFF)

In order to preserve information confidentiality and anonymity, we propose a pre-fusion transformation, $\mathbf{X}' = \mathbf{A}(\mathbf{X})$. The role of the pre-fusion transformation is to obfuscates the information in a way that calculation of the results is still possible from the transformed information:

$$\mathbf{A}: \quad \mathbf{X} \longrightarrow \mathbf{X}' \tag{1}$$

where $\mathbf{A}$ is called *pre-fusion transformation* or *anonymizer*. The anonymizer may require changes in the fusion function ($\mathbf{F}(\mathbf{X}) \rightarrow \mathbf{F}'(\mathbf{X}')$) and a transformation on the calculated results ($\mathbf{A}'(\mathbf{r}')$), called the *result inverse transform*:

$$\mathbf{A}': \quad \mathbf{r}' \longrightarrow \mathbf{r} \tag{2}$$

Therefore we shall have:

$$\mathbf{r} = \mathbf{F}(\mathbf{X}) = \mathbf{A}'(\mathbf{F}'(\mathbf{A}(\mathbf{X}))) \tag{3}$$

Although the above equation should hold for the secure information fusion system, in some cases, the system is designed in a way that the secure information algorithm approximates the intended result:

$$\mathbf{r} = \mathbf{F}(\mathbf{X}) \approx \mathbf{A}'(\mathbf{F}'(\mathbf{A}(\mathbf{X}))) \tag{4}$$

This result inaccuracy might be mandatory to improve collusion resistance of the system.

The most important role of the anonymizer is to transform the information from the *normal* or *definite space*, where values are represented in world readable or clear text format to the *anonymous space*, where node specific information can not be deduced. It is obvious that goal function anonymity condition is mandatory to make fusion in anonymous domain possible; however, it does not guarantee nor gives a way for calculation of the fusion function in the anonymous domain.

For distributed and scalable consensus fusion systems like sensor networks, the anonymizer should be implemented as a localized anonymization transformation as well. Hence we have: $x'_i = A_i(x_k)$ where $x_k \in N'_i$ for each node.

After the anonymization phase, information fusion takes place. According to the anonymous space properties, changes in fusion function might be required. A good heuristic for finding the proper fusion function over anonymous space is based on the mapping of the operators in definite space to the operators in the anonymous space. Obviously, the best anonymizer in terms of simplicity is the one that does not affect the fusion function. Such anonymization transforms would also eliminate the need for *result inverse transform function*. Figure 1-b illustrates BIFF phases and compares it with the classical information fusion model (figure 1-a).

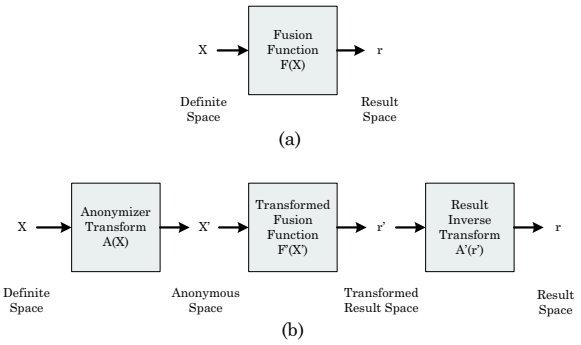

Figure 1. Information Fusion: a) Classical Information Fusion Model and b) Blind Information Fusion Framework (BIFF).

In a classical model which satisfies the conditions discussed before, each node has its own initial value at first, then it goes through the fusion process, modeled by $\mathbf{F}(\mathbf{X})$. This may incur calculating an iterative function of its own and neighbors' values. Finally each node converges to the correct results ($\mathbf{r}$). In BIFF, before going through the fusion process, there is an anonymization phase ($\mathbf{A}(\mathbf{X})$), where each node transforms its initial value to a new value which

might be a function of its own, or its own and other members' initial values.

The case in which the local anonymization transform, $A_i(x_k)$, is only a function of the node's initial value (i.e. $k \in \{i\}$) seems straight forward since no information exchange is required prior to fusion. Nevertheless, one may raise a question on how $A_i(x_k)$ may work when information from other members is required. In the latter case, obviously, the nodes do not communicate their initial value, but a variation of it or just provide each other random numbers. This variation may depend on the goal function, and taking the *transformed fusion function* ($\mathbf{F}'(\mathbf{X}')$) and *result inverse transform* ($\mathbf{A}'(\mathbf{r}')$) into consideration. The *Random Projections Method*[5] which will be discussed in the succeeding sections is an example of this model.

Even when the *goal function anonymity* condition holds, there is no guarantee for maximum robustness of the consensus fusion function. In fact the goal function anonymity condition is a primary condition and only guarantees robustness against only one adversary. The actual robustness of the system against collusions depends on the nature of the fusion function itself. For example, as we will see, the consensus averaging function is intrinsically robust to not more than $n-2$ colluding nodes. In order to clarify this point, let us model the set of possible values each node can have as an n dimensional space and call it information space. Theorem 1 and its succeeding corollary explain the relation of the fusion function properties with collusion resistance.

Theorem 1. *Assuming an n dimensional information space, any k dimensional consensus fusion function selects a $n-k-1$ dimensional sub-space of possible initial values for each node.*

Proof. Assuming each node has only a single real value as the input of the fusion function, let us arrange these values to an ordered n-tuple or a $1 \times n$ vector. The information space, which is the set of all possible values of this vector is an n dimensional space. Before the fusion, as we assumed, each node has only one value; hence, there are $n-1$ unknowns to each node if she is going to guess others' value.

A k dimensional consensus fusion function means that the results of the consensus fusion is a $1 \times k$ vector or analogously k one dimensional fusion functions. Generally, each one dimensional fusion function is a function of n variables. Assuming the result of the function is known, each function turns to an equation with n variables. Hence, a k dimensional fusion function defines a system of k equations of n variables after the fusion. Here we assume that all the equations are independent. This is a logical assumption since dependence among the equations leads to dependance among the results of the fusion function. This means that some of the fusion results can be obtained from the others and therefore, there is no reason to include them in the fusion process.

In order to guess the initial values of all the nodes, each node has $n-1$ unknowns and k equations. That is, the dimensions of the unknown space is reduced by k after the fusion for each node as the possible initial values must satisfy k equations. Consequently, a $n-k-1$ subspace of the n dimensional unknown space is left after the fusion for each node. □

Corollary 1. *Assuming an n dimensional information space and a k dimensional consensus fusion function over it, collusion of at least $n-k$ nodes is mandatory to exactly reveal all nodes information.*

It must be noted that corollary 1 states the upper bound since the complexity of solving the equations based on the fusion function is not considered in theorem 1. Also, theorem 1 gives the possible answer space and not the probability distribution function (PDF) over this space. This PDF depends on the PDF of the nodes' information as well as the fusion function. Consequently, the bound given by corollary 1 might be relaxed for estimation and statistical attacks.

As an example for corollary 1, let us consider n nodes and the simple multiplication fusion function: $r = \prod_i x_i$. The fusion function chooses an $n-2$ dimensional subspace of the n dimensional information space for each node. Hence, collusion of $n-1$ nodes or analogously knowing $n-1$ x_is will lead to exactly finding the unknown x_i.

The intrinsic collusion robustness of a consensus fusion over an n dimensional information space is therefore $n-k-1$, where k is the number of the bounds implicated by the fusion function. Consequently, for any consensus fusion function, the number of bounds put by the fusion function reduces the level of collusion resistance of the system.

Based on the above discussion, the goal function anonymity condition can be extended to the *goal function anonymity order* concept. We define the goal function anonymity order of a fusion function as *the number of independent variables the consensus goal function leaves over the information space minus one*. This parameter gives an upper bound for collusion resistance level in an exact secure consensus fusion scheme which can be met regardless of the nature of the fusion function and neglecting the complexity of finding the inverse fusion transform. For example, assuming the fusion function calculates k independent linear functions in a network of n members. Generally, knowledge of $n-k$ of the member information is enough for calculation of the other k.

The upper bound given by corollary 1 is for simple cases (such as linear ones) where reversing the fusion function has polynomial complexity; nevertheless, if the fusion function belongs to the P class while the reverse problem belong to the NP class, the node information might not be recoverable even in case of $n-1$ colluding nodes. Although not clearly

stated, some methods for distributed calculation of group digital signature over an ad hoc network are examples of such fusion functions[1].

Regardless of the complexity of the fusion function, the anonymizer can achieve anonymization in two fundamentally different ways:

- **Non-deterministic anonymization or Noisification:** Introduce new, possibly random, information to the system. This new information is regarded as noise since it is unwanted and may affect the accuracy of the results. The anonymizer must be designed in a way that the ratio of the effect of the introduced noise in the results to the effect perceived by the attacker is minimized. The effect of the noise can be interpreted as loosening the bounds put by the consensus fusion function. In this case, the upper bound of the collusion resistance can be increased. However, the final result will suffer from steady-state error as the effect of the noise can not be completely eliminated. The ratio of the perceived error by the attacker to the error in the results (steady-state error) is an important factor and modeled by the *Network Processing Gain (NPG)* in [7].

- **Information Shuffling or Decomposition:** Decompose the member information into shares and shuffle the shares between the fusion party in a way that the local aggregate of the shares can be fused for the results. This method just gives obfuscation, mutual privacy and anonymity but no increase in collusion resistance. Nevertheless, it can have no steady-state error, since no new information is introduced to the system.

From the above discussion, if the fusion function shows good collusion resistance properties the second method is preferred since it is steady-state error free. This decision can be made considering anonymity order of the fusion function as well as its complexity. For reversible fusion functions, if a collusion resistance level more than the bound given by the goal function anonymity order is required, the first method is the way.

Modeling the consensus fusion function as a filter over the information, the first method can be explained as adding noise to the input whose effect will be eliminated by the zeros of the fusion filter. Hence, we call this method the *noisification* method.

BIFF can be more understood considering common voting system where people cast their ballots to the ballot box. The desired property of the voting system is keeping each voters opinion private through anonymous ballots which only reflect the voters opinion; however, as the authenticity of the votes must be approved, each ballot shall be cast by a person and in fact, the real anonymization takes place in the ballot box. In BIFF, the anonymization transform plays the role of the ballot box.

BIFF can be adopted for secure consensus fusion over sensor networks as its suitable to be joined with locally implemented fusion functions. Implementing anonymizer based on the injected noise can specially fit computation constrained networks as random number generation is not a costly operation and most of the times readily available. For the locally implemented fusion functions algorithm 1 can be used as a general approach based on BIFF.

Algorithm Blind Fusion

Input: x_i (node information)
Output: r_i (fusion result)

Main Procedure
Step 1: Anonymization
Decompose x_i to x_{ij}s and/or Calculate side information or noise (ν_i)
Communicate the shares to the neighbors if required
Calculate the anonymized information:
$x'_i \leftarrow A_i(x_k) \equiv A_i(x_{ki}, \nu_i)$ for $k \in N'_i$

Step 2: Fusion
Go through the altered fusion function (possibly multiple rounds):
$r'_i \leftarrow \mathbf{f}'_\mathbf{i}(x'_k)$ for $k \in N'_i$

Step 3: Result Calculation
Calculate the results through the inverse anonymization transform:
$r_i \leftarrow A'(r'_i)$

End

Algorithm 1. Blind Fusion Algorithm for node i

3.3 Examples

In this section we formulate two secure consensus averaging methods proposed by Kefayati *et al.* [6, 7] according to BIFF. We briefly introduce each method and give the BIFF based notation, i.e. definition of $\mathbf{A}(\mathbf{X})$, $\mathbf{A}'(\mathbf{r}')$ as well as $A_i(x_k)$ and the others.

Since both methods are designed to be fusion method independent, that is, to work with any fusion function which realizes consensus averaging, they do not require any change in the fusion function. Therefore, $\mathbf{F}'(\mathbf{X}') = \mathbf{F}(\mathbf{X}')$ or simply:

$$r = \frac{1}{N}\sum_i x_i = \frac{1}{N}\sum_i x'_i \tag{5}$$

This means that for both methods $\mathbf{A}(\mathbf{X})$ is designed in a way that averaging function is the same over both definite and anonymous spaces. The advantage of such an approach is two fold: first it does not require any change in the fusion function which simplifies the implementation and increases flexibility of the system. Second, it eliminates the need for result inverse transform function, i.e. $\mathbf{r}' = \mathbf{r}$.

3.3.1 Secure Consensus Averaging based on Random Offsets Method:

The main idea behind *Random Offsets Method (ROM)*[7] is that, for large networks, the consensus averaging fusion function is the minimum variance unbiased estimator

(MVUBE) of the mean value of x_is, assuming they are i.i.d. random variables, or in the other words:

$$\lim_{n\to\infty} r = \lim_{n\to\infty} \bar{x} = \lim_{n\to\infty} \frac{1}{N}\sum_{i=0}^{n} x_i = E\{X\} \quad (6)$$

If each node adds a randomly chosen offset, o_i, to its initial value, and assuming that the offsets are chosen from a zero mean distribution we will have:

$$r = E\{X+O\} = E\{X\} + E\{O\} = E\{X\} \quad (7)$$

For finite number of nodes in the network, however, there will be an error in the final result. This error is inversely proportional to the number of nodes in the network[7].

From BIFF point of view anonymizer can be defined as:

$$\mathbf{X}' = \mathbf{A}(\mathbf{X}) = \mathbf{X} + \mathbf{O} \quad (8)$$

where $\mathbf{X} = [x_0, x_1, ..., x_n]^T$, $\mathbf{O} = [o_0, o_1, ..., o_n]^T$. Analogously we have:

$$x'_i = A_i(x_k) = x_i + o_i \quad (9)$$

It must be noted that the each offset is only known to the corresponding node itself. According to BIFF, ROM is based on the noisification methodology. ROM introduces new information to the system which helps it to obfuscate the original information and then exploits the properties of the fusion function to eliminate the effect of the noise. From another point of view and according to equation 4, by choosing an anonymous space for which the fusion results are approximated by the original fusion function much better than the original information, ROM achieves its goal in providing a means for mutual confidentiality.

According to BIFF, methods which introduce new information to the system can achieve a higher level of collusion resistance. Considering the linear bound by the consensus averaging fusion function, the intrinsic collusion resistance level of this fusion function is $n-2$. However, ROM achieves $n-1$ collusion resistance since it uses noisification methodology. This is not only more than the intrinsic collusion resistance level of consensus averaging function but also the maximum achievable level of collusion resistance for n nodes.

3.3.2 Secure Consensus Averaging based on Random Projections Method:

Random Projections Method (RPM)[6] is based on information decomposition idea presented in the previous section. According to RPM, before the fusion process, each node decomposes its initial value to a summation of n'_i randomly chosen numbers, namely x_{ij}s, in a way that $\sum_j x_{ij} = x_i$. These numbers or random projections are then communicated among the neighbors. It must be noted that there are n'_i random projection and $n'_i - 1$ neighbors, that is, each node keeps one random projection as self projection for itself denoted by $x_{in'_i}$. After all the nodes are done with the projection exchange process, each node calculates its transformed initial value according to the following equation:

$$x'_i = \sum_{j=0}^{n_i} x_{ji} + x_{in'_i} \quad (10)$$

Finally, all nodes go through the consensus averaging process using x'_i as their initial value. Obviously, the result is exactly the same as the intended results since the summation is done over all the random projections and divided by the number of the nodes in the network.

In RPM, formulation based on BIFF can also give a very simple proof of convergence. Before defining $\mathbf{A}(\mathbf{X})$ we first define an auxiliary matrix called *random projections matrix* denoted by $\mathbf{R}$ whose elements are the random projections communicated among the nodes. In other words, r_{ij} is the random projection sent from the node i to the node j. As projection exchange is done among the neighbors, for the nodes whom are not direct neighbors, we have $r_{ij} = 0$. Obviously, r_{ii} is the self projection of the node i. According to this notation we have:

$$\mathbf{X} = \mathbf{R}.\mathbf{1} \quad (11)$$

in which $\mathbf{1}$ is $|V|$ element column matrix of ones, i.e. $[1, 1, \ldots, 1]^T$. Equation 11 models the decomposition process. Aggregation can also be expressed in terms of $\mathbf{R}$ as:

$$\mathbf{X}' = \mathbf{A}(\mathbf{X}) = \mathbf{R}^{\mathbf{T}}.\mathbf{1} \quad (12)$$

which formulates $\mathbf{A}(\mathbf{X})$. Each row of $\mathbf{X}'$ in equation 12 gives the corresponding local anonymizer, $A(x_k)$, which is given in equation 10 as well. The convergence of the method can also be checked based on $\mathbf{R}$ matrix as:

$$r' = \frac{1}{N}\mathbf{1}^{\mathbf{T}}.\mathbf{X}' = \frac{1}{N}\mathbf{1}^{\mathbf{T}}.\mathbf{R}^{\mathbf{T}}.\mathbf{1} = \frac{1}{N}(\mathbf{1}^{\mathbf{T}}.\mathbf{X})^{\mathbf{T}} = r \quad (13)$$

As RPM is an exact method for secure consensus averaging, according to BIFF, its maximum degree of collusion resistance is at most $n-2$ which is confirmed by [6].

4 Conclusion and Future Work

In this paper we introduced a general framework for secure information fusion over sensor networks called *Blind Information Fusion Framework (BIFF)*. Our focus was mostly on consensus information fusion on cooperative networks. The most challenging issue addressed in BIFF is to maintain node information privacy while making the information fusion viable in an honest-but-curious adversary

model. Our solution to this problem is described as transformation of the information from the normal space to the anonymous space where the nodes' information can not be deduced and implementation of the fusion function in the anonymous space. The most important requirement of the anonymous space is its ability to hide node information while making the fusion possible. We also defined the collusion resistance level of a fusion function as "the maximum number of nodes whose collusion can not reveal other node(s)' information". After discussing some properties of BIFF and their relations with the fusion function, we analyzed collusion resistance of two major families of anonymization transforms. Also, two methods proposed for secure consensus averaging were formulated and discussed in BIFF as examples.

We believe that BIFF needs more work to mature. We are considering other properties of the anonymization transforms and their corresponding anonymous spaces as our major part of our future work. These properties include collusion resistance, flexibility and scalability for implementation in various kinds of sensor networks, especially the sensitivity of the anonymization to sparse connectivity. Our min focus is on finding a fusion function independent anonymizer suitable for implementation over capability limited sensor networks. Extension of BIFF to non-cooperative networks and the effect of non-cooperative nodes on performance and security of the fused information is left for our future studies as well.

References

[1] Aruna Balasubramanian, S. Mishra and R. Sridhar. "Analysis of a Hybrid Key Management Solution for Ad hoc Networks," *IEEE Wireless Communications and Networking Conference (WCNC 2005)*, Volume 4, pp. 2082–2087, Mar. 2005.

[2] Zuzana Beerliova-Trubiniova and Martin Hirt. "Efficient Multi-Party Computation with Dispute Control," *Theory of Cryptography (TCC'2006)*, LNCS, Springer-Verlag, vol. 3876, pp. 305–328, Mar. 2006.

[3] Matthias Fitzi, Martin Hirt, and Ueli Maurer. "General Adversaries in Unconditional Multi-Party Computation," *Advances in Cryptology (ASIACRYPT'99)*, LNCS, Springer-Verlag, vol. 1716, pp. 232–246, Nov. 1999.

[4] C. Karlof and D. Wagner. "Secure Routing in Wireless Sensor Networks: Attacks and Countermeasures," *Elsevier Journal of Ad hoc Networks*, vol. 1, no. 2-3, pp. 293–315, Sep. 2003.

[5] Mahdi Kefayati, Mohammad S. Talebi, Babak H. Khalaj and Hamid R. Rabiee. "Private and Anonymous Information Fusion over Sensor Networks, " *Technical Report*, Sharif University of Technology, February 2006.

[6] Mahdi Kefayati, Mohammad S. Talebi, Hamid R. Rabiee and Babak H. Khalaj. "Secure Consensus Averaging for Secure Information Fusion in Sensor Networks," *The 10th International Symposium on Signal Processing and its Applications (ISSPA 2007)*, Sharjah, UAE, Feb. 2007.

[7] Mahdi Kefayati, Mohammad S. Talebi, Babak H. Khalaj and Hamid R. Rabiee. "Secure Consensus Averaging in Sensor Networks Using Random Offsets," *Submitted to The 3rd International Conference on Networking and Services (ICNS 2007)*, Athens, Greece, Jun. 2007.

[8] Katherine Luo, Yifan Li, Charis Ermopoulos, William Yurcik and Adam Slagell. "Scrub-PA: A Multi-Level Multi-Dimensional Anonymization Tool for Process Accounting ," *ACM Computing Research Repository (CoRR)*, Technical Report cs.CR/0601079 , January 2006.

[9] Nancy Lynch. *"Distributed Algorithms,"* Morgan Kaufman Publishers, San Mateo, CA, 1996.

[10] R. Olfati-Saber and R.M. Murray. "Consensus Problems in Networks of Agents with Switching Topology and Time-delays," *Automatic Control, IEEE Transactions on* , vol. 49, no. 9, pp. 1520–1533, Sep. 2004.

[11] R. Olfati-Saber and Jeff S. Shamma. "Consensus Filters for Sensor Networks and Distributed Sensor Fusion," *The 4th IEEE Conference on Decision and Control and 2005 European Control Conference*, Dec. 2005.

[12] Wei Ren, Randal W. Beard, E. Atkins. "A Survey of Consensus Problems in Multi-agent Coordination," *American Control Conference*, Portland, OR, pp. 1859–1864, 2005.

[13] Mohammad S. Talebi, Mahdi Kefayati, Babak H. Khalaj and Hamid R. Rabiee. "Adaptive Consensus Averaging for Information Fusion over Sensor, " *The 3rd IEEE International Conference on Mobile Ad-hoc and Sensor Systems (MASS'06)*, 2006.

[14] L. Xiao, S. Boyd and S. Lall. "A scheme for robust distributed sensor fusion based on average consensus," *Proceedings of the International Conference on Information Processing in Sensor Networks (IPSN)*, pp. 63-70, 2005.