12th International Conference on Information Fusion
Seattle, WA, USA, July 6-9, 2009

Performance Improvements in Distributed Estimation and Fusion Induced by a Trusted Core

Kiran K. Somasundaram and John S. Baras
Institute for Systems Research
and Department of Electrical and Computer Engineering
University of Maryland
College Park, MD 20742
kirans@umd.edu, baras@umd.edu

Abstract – *We consider distributed state estimation and fusion in autonomic adhoc sensor networks when some nodes are untrusted, corrupted or malicious. We provide rigorous graph models to capture both trust and network functionality relations. We propose a novel hierarchical scheme, inspired from socio-cognitive dynamics in economics and sociology, that utilizes a trusted core. We provide a component-based architecture to show the interplay between the trust and estimation schemes. As an explanatory example we consider the problem of distributed Kalman filtering. We show that trust-based schemes, implemented by a trusted core, result in significant improvement in the state estimation procedure. We also find that the interplay between estimation and trust updates quickly isolates malicious nodes and helps the observation limited nodes (corrupted sensor nodes).*

Keywords: trust particles, Kalman filtering, trust monitoring, trust sensitive filtering.

1 Introduction

Recent advances in silicon technologies have provided us with low-cost, low-power electronics. This has instigated active research in large-scale networks of small, wireless, low-power sensors and actuators [17]. As a result pervasive sensing has been freed from the burden of infrastructure limitations. In the next phase of sensor networking technologies the onus has been shifted to what is called *pervasive computing*. Functions such as information fusion and collaborative decision-making are at the heart of this upcoming technology [13]. This has opened up an arena for new research directions, with problems concerning the stability and operation of large autonomous distributed systems being among the most critical ones. In this paper, we consider the key problem of state estimation in an untrusted sensor network.

Initially state tracking in sensor networks was considered as a centralized decision making process. Though useful, this approach had many fundamental limitations in power limited sensor networks [12]. In recent years, the problem of information fusion for estimation has been studied from a distributed multi-agent decision making perspective. But this approach can have poor convergence properties in an untrusted sensor network. This is because there is no notion of global trust and hence imprecise or false information can easily corrupt the state estimation algorithms. In this paper, we present a novel hierarchical trust architecture called the *trusted core*, which is inspired from practices in sociology and economics, to tackle the aforementioned problem.

In the work described here, we introduce the notion of trust in the context of estimation. We provide rigorous graph models to capture both trust and network functionality relations. We provide a component architecture to show the interplay between the trust and estimation schemes. As an explanatory example we consider the problem of distributed Kalman filtering (DKF). We present a scheme by which the trust methods can be used to accelerate the state estimation procedures. It should be mentioned that our architecture is not limited to the Kalman filtering approach and can be extended to other distributed likelihood calculations across the network.

The paper is organized as follows. In section 2 we introduce distributed Kalman filtering. Then in section 3 we present our trusted core architecture. In section 4 we introduce the mathematical notations prevalent in the trust literature. In sections 5, 6 and 7 we present our system model and algorithms for trusted Kalman filtering. Finally in section 7, we discuss some simulation results, which validate our approach.

2 Distributed Kalman Filtering in Sensor Networks

There have been many distributed versions of the estimation problems using Kalman filtering approaches addressed in recent literature [13], [2], [19], [15], [16]. In addition to the sensor networks community, distributed estimation has been investigated in the information fusion community too [4]. The algorithms proposed for these purposes can be classified into

1. Fusion-centric methods

978-0-9824438-0-4 ©2009 ISIF

2. Estimation propagation methods.

The *fusion-centric* methods rely on the existence of a fusion center that performs some form of network-wide aggregation and broadcasts this estimate to the individual nodes [12]. Though these algorithms have good convergence properties, they incur high costs in terms of the communications requirements. Schemes, which are based on *estimation propagation* methods employ only local message passing algorithms [12]. But as expected, these algorithms have slower convergence properties. In this paper, we provide a compromise solution to both approaches.

In this paper, we discuss a hierarchical scheme which is inspired from socio-cognitive dynamics found in several practical scenarios in economics and sociology [11], [18], [6]. The basic idea in hierarchical trust schemes is to provide a global trust on a particular context without requiring direct trust on the same context between all agents in the system [1]. To the best of our knowledge, using trust-based approaches to solve distributed estimation or fusion problems, have not been addressed in the literature.

3 Hierarchical Estimation

In this section we present our hierarchical estimation architecture. We consider a heterogeneous sensor network deployment. In addition to sensors which are deployed to carry out the infrastructure functions of the traditional sensor network, a dedicated class of nodes, called the *trusted core*, is also deployed at a much lower density. These special class nodes are assumed to have a broader observation of the system, due for example to aggregation of a larger set of observations in space and time (history), which the estimation algorithm tracks. The fundamental idea in this paper, is to construct a two layer estimation process. We establish a hybrid method that embodies the techniques used in *fusion-centric*, *collaborative filtering* and *estimation propagation methods*. In the rest of this section, we discuss the properties of the trusted core and its functionalities.

3.1 Trusted Core (TC)

Trust technologies have been used successfully in e-business and e-services with significant success in the recent past [3]. Existing computation models on trust technology that are used in practice usually work with direct trust measures. However these direct trust methods are myopic and hence do not provide scalable solutions to large sensor networks. There are also other trust models which work with indirect (transitive) trust (c.f. reputation, recommender and referral systems). However it has been found (after investigation) that these indirect (transitive) trust methods may be expensive to deploy in real networks [3]. In the present paper we work with only direct trust methods. We circumvent the aforementioned problem of myopia by invoking a special construction called the *trusted core*.

The *trusted core* is a special class of nodes, which are installed with higher levels of security. We refer to the individual nodes in the trusted core as the *trust particles*. These nodes are deployed at a critical density such that every sensor node has the ability to talk with one or more trust particles at an admissible cost. This would mean that the cryptographic primitives and multi-hop communications must be limited to the capabilities of the participating nodes. Such cores can be shown to exist using probabilistic methods from *Geometric Random Graphs* [8] . We also assume that before deployment there is a security association between the trusted core and the other sensor nodes in the system. Such associations can be installed using a light-weight *public-key* infrastructure [14]. Such a security association is necessary because our algorithms require that we need the *non-repudiation* property to hold for the messages from the trust particles. Furthermore, in our recent work we have shown that such security associations, with the trusted core being equipped with higher degree of security, are both essential and natural for establishing trust and security in autonomic networks, c.f. MANET.

3.2 Properties of the Trusted Core

The trusted core is deployed for the purpose of increasing the sensing or observing capabilities of the sensor network system. The trust particles can thus be dedicated sensor nodes whose primary function is to observe the dynamics of interest. In such a networked system, the communication channel between the trust particles should support *confidentiality* and *integrity*. Thus in a typical scenario, these communications should be supported by multi-path (i.e. multiple physical paths exist between the two communicating nodes) communications between the trust particles. It is clear that if we want to realize a small trusted core, these communication channels must be supported by the existing sensor node technologies (c.f. motes).

The trusted core can achieve a broader observation index either through collaborative filtering or by providing it (i.e. the trusted particles) with additional sensing capabilities.

In a sensor network with a trusted core subsystem, each sensor node requests a trustworthy estimate from the trusted core. It is assumed that this request is made and processed through a channel which provides *confidentiality* and *integrity*. Again this demands the use of multi-path channel establishment with one or more trust particles. There is an evident cost associated with such a secure communication infrastructure because of the cryptographic primitives employed and multi-hop-multi-path communications. Thus sensors should access the trusted core *frugally* to improve the performance of their estimation procedures.

3.3 Distributed Kalman Filter Particles

The sensor nodes in the system exchange estimates in their local neighborhood and trusted measurements from the trusted core to estimate the state dynamics. The resulting algorithm can be viewed, from an implementation perspective, as a component of particles executing Kalman filtering while distributed across the network. From henceforth we refer to these sensor nodes as *Distributed Kalman Filter particles* (DKF particles). Thus there would be message passing of the estimates among the local particles. Such approaches have been discussed in great detail in the current literature [12], [13], [2], [16]. The trusted estimates from the trusted particles can be used as a reference performance measure to characterize these local estimates.

It should be mentioned that unlike the typical formulation in traditional distributed Kalman filtering approaches, attempting to construct a Bayesian inference framework would not be apt. This is because, unlike the situation in Bayesian statistics, there is a notion of context hidden in these estimations. This notion of *context* in trusted transactions is discussed in the forthcoming sections. Unlike the setting in traditional distributed Kalman filters, we cannot make the *utopic* assumption that all nodes conform to the protocol. There might be *malicious* DKF particles which might report incorrect estimates. Trust systems are often governed by subjective measures and developing a rigorous Bayesian estimation will not be fruitful. Even if the DKF particles conform to the protocol, communications can be hacked (e.g. attacked, stolen, altered) by intelligent adversaries and the system under Bayesian inference can be driven into unacceptable states. In the next section we develop a rigorous model to account for these trust and estimation dynamics.

4 Mathematical model

To develop a model for the problem at hand, we need more than one simple graph to model the algorithms. Let us consider a sensor network organization model as shown in Figure 1. The nodes indicated by circles correspond to the DKF particles, while those represented by the squares correspond to the trust particles.

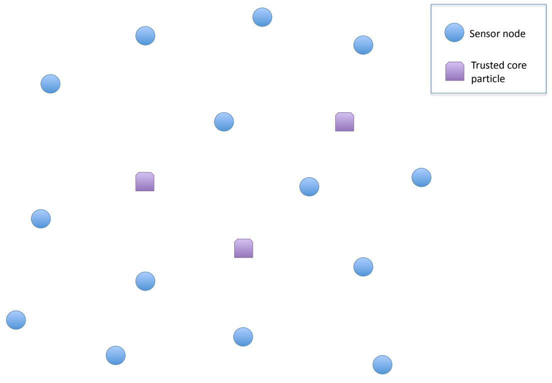


Figure 1: Sensor network organization model

There is an undirected communication graph $G_c(V, E_c)$ which is induced by the commonly employed *disc physical communication model*. That is, there is an edge $(i, j) \in E_c$, if and only if, $||p_i - p_j|| \leq \rho$ where p_i and p_j are the positions of nodes $i, j \in V$. Thus nodes i, j communicate (here ρ is the communication range, and in general can depend on the location of the nodes, as well as on other parameters such as power, modulation, etc.), when the link $(i, j) \in E_c$. Such a communication graph is shown in Figure 2.

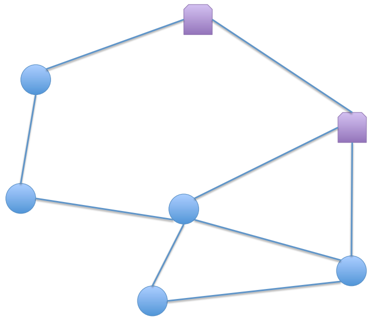


Figure 2: Communication graph G_c

In trust technology [20], [3], trust relations are modelled as binary relations between the participating agents. In our case, the context of such a trust relation corresponds to *providing a trustworthy estimate of the state dynamics*. An example is illustrated in Figure 3. In this case, node i *trusts* node j, in the above context, with a trust value of $t(i, j)[n]$ at time slot n.

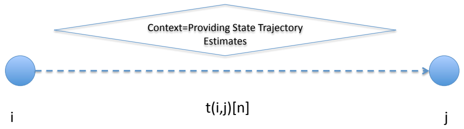


Figure 3: Illustrating the trust relation used

Such a binary relation would induce a *weighted directed dynamic trust graph* $G_t(V, A_t)$ in a network of agents/nodes. The weights in the arcs indicate the trust value of that relation. The trust particles form a subset $V_{tc} \subset V$. The trusted core assumption mandates that for this dynamic graph

$$t(i, tc)[n] = max_T \quad , \forall i \in V, \forall tc \in V_{tc} \text{ and } \forall n \geq 0$$

The parameter max_T in the above relation depends on the trust technology/scheme used. For the same communication graph shown in Figure 2, the corresponding trust graph would appear as the one shown in Figure 4.

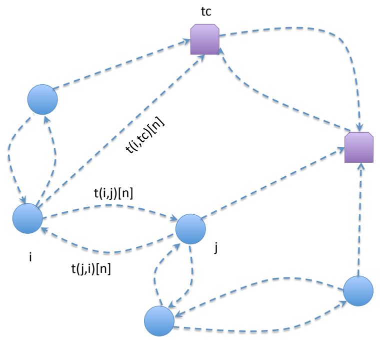


Figure 4: Trust Graph G_t

Both G_c and G_t induce another directed weighted graph $G(V, A)$. If there exists a well-defined trust relation between nodes i and $j \in V$ in the trust graph *and* a path between i and j in the communication graph, then an arc $(i, j) \in A$ is induced in G. Mathematically, this is the intersection of G_t and the transitive closure of G_c. The arcs of A are indexed by multi-valued (vector) weights. That is with the arc $(i, j) \in A$, there is an associated dynamic vector weight $w(i, j)[n] = (c(i, j), t(i, j)[n])$. Here $c(i, j)$ is used to model the cost of the cryptographic (i.e. secure) and multi-hop communication scheme to reach j from i. $t(i, j)[n]$ represents the trust that node i has on node j at time n. This is illustrated in the example shown in Figure 5. For instance, node i to reach the trusted particle tc has to incur a cost of two units, because of the two hop reachability between the two nodes.

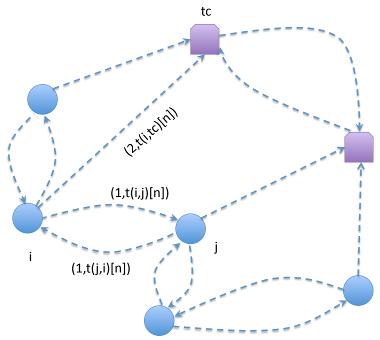


Figure 5: Induced Graph G

Henceforth the graph quantities, that we refer to, would correspond to those of the induced graph G. For any node $i \in V$, we denote its *local neighborhood* (i.e. one hop neighbors) by $\mathcal{N}(i)$. We denote the *inclusive neighborhood* by $\mathcal{N}^+(i) = \{i\} \cup \mathcal{N}(i)$. Any form of (local) message passing happens in this $\mathcal{N}^+(i)$. Using this graph model we present the algorithms to perform trusted Kalman filtering over a network in the forthcoming sections. In the next section we explain what we mean by a *Trusted Kalman Filter* (TKF) in the context of reliable tracking.

5 Goals of our Trusted System

We design the trusted system based on the following requirements for our design.

1. All the sensors which abide by the protocols of sensing and message passing, should be able to track the trajectories.

2. This implies that those nodes which have poor sensing capabilities, *nodes with corrupted sensors*, should be aided by their neighbors in tracking. We do mention here that this altruistic approach can lead to selfish behavior by the nodes which can exploit the coalition. Our goal is not to mitigate this selfishness.

3. Those nodes which are *malicious* and pass false estimates, should be quickly detected by the trust mechanism and their estimates should be discarded.

6 System model

In this paper, we consider a linear state space model for the system which needs to be tracked,

$$\begin{aligned} \underline{x}[n+1] &= \mathbf{A}\underline{x}[n] + \mathbf{B}\underline{w}[n] \\ \underline{z}_i[n] &= \mathbf{H_i}[n]\underline{x}[n] + \underline{v}_i[n]. \end{aligned}$$

The suffix i is specific to a node $i \in V$. Then $\underline{z}_i$ is the observation seen at particle i. The driving noise has covariance structure $\mathbf{Q}$ and the observation noise at every DKF particle is assumed to have a covariance structure $\mathbf{R_i}$.

6.1 Trusted core observations

The particles of the trusted core can be modelled as a single observation system because we assume completely trusted communications among the trust particles. Thus the observation at any trusted particle is modelled as

$$\underline{z}_{tc}[n] = \mathbf{H_{tc}}[n]\underline{x}(n) + \underline{v}_{tc}[n]$$

We assume that $(\mathbf{A}, \mathbf{H_{tc}})$ is completely observable and the observation noise's covariance structure $\mathbf{R_{tc}}$ is well bounded below the $\mathbf{R_i}$ for the other filter particles (i.e. non trusted particles).

7 Trusted Distributed Kalman Filter

Our trusted Kalman filter is best described using the component model shown in Figure 6. The DKF particle consists of a *Kalman Filtering component*, which computes the likelihood using the local information estimates. The *trust update component* is any valid trust system which achieves the goals of our design described in Section 5. Many protocols such as [9] and [10] satisfy our simple requirements. The component architecture clarifies the fact that trusted estimation is not necessarily restricted to the Kalman filtering problem or approach. Any distributed *sequential MMSE* scheme can be used as a component instead of the Kalman filter. In addition the designer also has the freedom to choose any good trust update system. The interface where these two components interact is through the trust weights $w(i, j)$. This is explained in this paper for the specific example of the DKF.

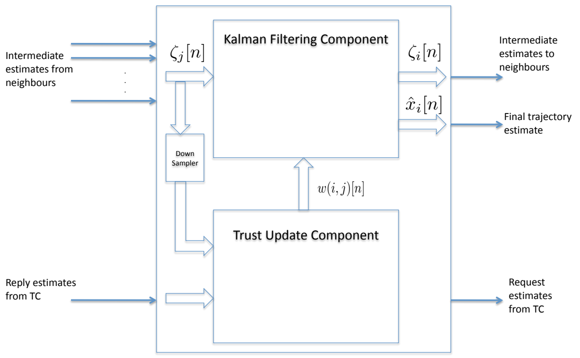

Intermediate estimates from neighbours
$\zeta_j[n]$
Kalman Filtering Component
$\zeta_i[n]$
Intermediate estimates to neighbours
$\hat{x}_i[n]$
Final trajectory estimate
Down Sampler
$w(i,j)[n]$
Trust Update Component
Reply estimates from TC
Request estimates from TC

Figure 6: Component Architecture for the DKF particle

Algorithm 1 Trusted Kalman Filter

```
Init $M[0], \underline{\hat{x}}_i = \underline{x}(0), n = 0$
**repeat**
  $n \leftarrow n + 1;$
  **Prediction MSE**
  $\mathbf{P}[n] = \mathbf{A}\mathbf{M}[n-1]\mathbf{A}^T + \mathbf{B}\mathbf{Q}\mathbf{B}^T$
  **Kalman Gain**
  $\mathbf{K}[n] = \mathbf{P}[n]\mathbf{H_i}^T(\mathbf{R_i} + \mathbf{H_i}\mathbf{P}[n]\mathbf{H_i}^T)^{-1}$
  **Local correction**
  $\underline{\zeta}_i[n] = \mathbf{A}\underline{\hat{x}}_i[n-1] + \mathbf{K}[n](\underline{z}_i[n] - \mathbf{H_i}\mathbf{A}\underline{\hat{x}}_i[n-1])$
  **The nodes exchanges the local estimates** $\underline{\hat{x}}_j, \forall j \in \mathcal{N}^+(i)$
  **Trust sensitive filtering**
  $\underline{\hat{x}}_i[n] = \sum_{jin\mathcal{N}^+(i)} w_{ij} \times \underline{\zeta}_j[n]$
  Estimation MSE
  $\mathbf{M}[n] = (\mathbf{I} - \mathbf{K}[n]\mathbf{H_i}[n])\mathbf{P}[n]$
**until** Forever
```

Algorithm 2 Trust Update for the inclusive neighborhood

```
Init $t(i,j)[0] = \frac{1}{|\mathcal{N}^+(i)|}, \quad \forall j \in \mathcal{N}^+(i)$, and $k = 0$
**repeat**
  Wait for Exponential time $\tau$
  $k \leftarrow k + \tau$
  Request Estimate update from the TC
  The TC replies with its trustworthy estimate $\underline{\hat{x}}_{tc}$
  **for all** $j \in \mathcal{N}^+(i)$ **do**
    $dev(j) = ||\underline{\zeta}_j - \underline{\zeta}_{tc}||_2$
    $t(i,j)[k] = \begin{cases} \min(max_T, t(i,j)[k-1] + \delta) & dev(j) \leq Dev_T \\ t(i,j)[k-1]/2 & dev(j) > Dev_T \end{cases}$
  **end for**
**until** Forever
```

Algorithm 1 is a modified form of the Kalman filter. The filtering is modified at the correction phase where instead of the sequential MMSE estimate, the update at node i is based on local estimates from the particles in $\mathcal{N}^+(i)$. The local estimates are filtered through a time varying trust-sensitive filter. The normalized time-varying trust values $w(i, j)[n]$ for this filter are obtained from the interface of the trust update component. Every node $i \in V$ carries out Algorithm 1 and passes messages across to its neighbors.

The trust update mechanism that we employ for our example system is a linear credit and exponential penalty scheme that is being used in many reputation technologies. For a detailed analysis of such dynamics we refer the reader to [5]. Each of the nodes in the network carries out Algorithm 2 to compute the trust value for the estimation capability of the local inclusive neighborhood. The trust monitoring algorithm compares the local estimates and trusted estimates. If the deviation is within a threshold Dev_T, the trusted agent is credited with an additive increment δ in the trust value. Otherwise, the trusted agent is penalized with an exponential decrease in the trust value. For the trust monitoring system, the reference estimates are obtained from trust particles. To obtain a realistic model for the trusted core access, we use a Poissonized access model ([7]). Each node accesses the trusted core with a Poisson point process, with the inter-access period exponentially distributed with mean $c(i, tc) \quad \forall tc \in V_{tc}$.

8 Simulation Results

Figure 7 shows a geometric random graph model of the sensor network. The nodes are deployed at the Poisson density of $\lambda = 100$. The communication range $\rho = \frac{1}{1/5}$ induces a communication graph G_c. The points denoted by the circles, pluses and crosses correspond to the set of DKF particles deployed. For this simulation we assume that there are 10 nodes with corrupted sensors and 5 malicious nodes which do not conform to the message passing and update protocols.

Additionally, another set of sensors which form the trusted core are deployed at a Poisson density $\lambda = 10$. These are denoted by the squares in Figure 7.

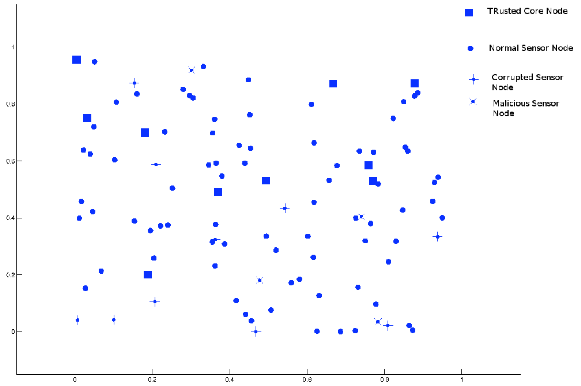


Figure 7: Sensor network realization

8.1 System Behavior

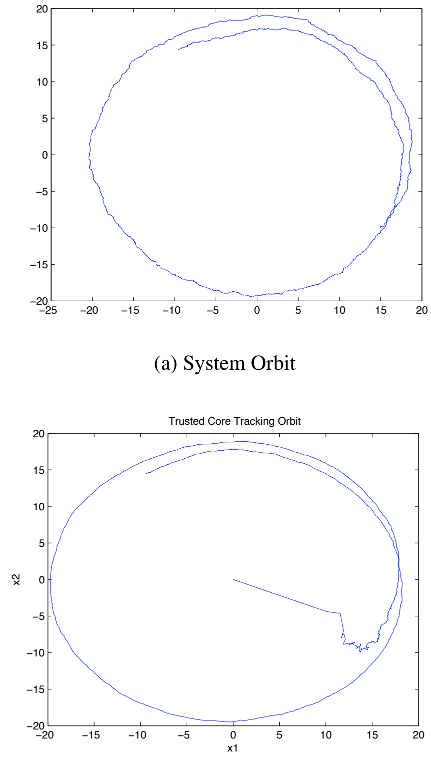
A sample path of the system orbit and the trusted core tracking trajectory is shown in Figures 8a and 8b. These correspond to the system model and trusted core observation model described in Section 6. For our simulations we use the linear system used by *Olfati-Saber-CDC 2007* [16] where

$$\mathbf{A} = 2\begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}$$

$$\mathbf{Q} = 25\mathbf{I_2}, \quad \underline{x}(0) = (15, -10)^T$$

$$\mathbf{H_{tc}} = I_2, \quad \mathbf{R_{tc}} = 30\mathbf{I_2}$$

The DKF particles either have an observation matrix $[1 \quad 0]$ or $[0 \quad 1]$. We set $R_i = 30$ for the reliable nodes and $R_i = 300$ for the corrupted nodes. As illustrated in Figure 8 the trusted core tracks the state trajectory with a very high level of accuracy. Thus indeed, this would be a trustworthy reliable reference for the DKF filter particles.



(a) System Orbit

(b) Trusted Core Tracking

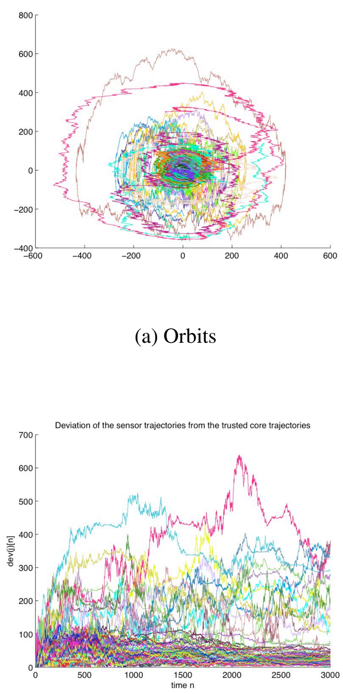
Figure 8: System Performance

8.2 Open loop performance

In this subsection we discuss the open loop performance of the distributed Kalman filter. By open loop we mean that the trust values are not fed back into the Kalman estimation component in Figure 6. Such an algorithm accepts blindly the estimates from its neighbors. We see a poor performance for this blind algorithm in terms of the tracking capability as depicted in figure 9. Figure 9a shows orbits of all sensor nodes in the system. The system is unable to discern between the reliable and unreliable estimates and hence we observe that the deviation from the trusted reference is large for almost all the estimated trajectories (Figure 9b).

8.3 Closed loop performance

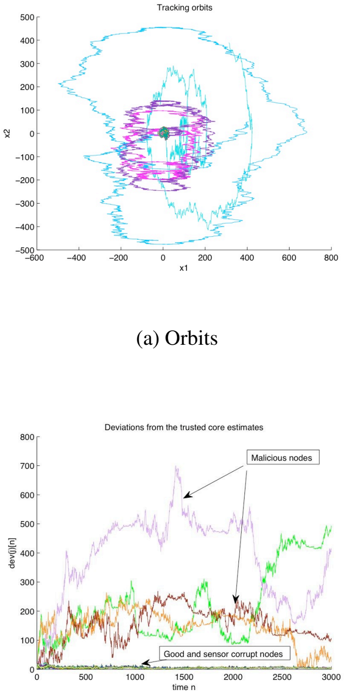
When the loop is closed by feeding back the trust values $w(i, j)$ to filter the estimates, we obtain a very good tracking performance as expected. This is shown in figure 10. The deviation from the trusted value has been significantly reduced for both good and corrupted sensor nodes. The nodes, which have high deviations in Figure 10b are the malicious nodes.



(a) Orbits

(b) Deviations

Figure 9: Open Loop Performance



(a) Orbits

(b) Deviations

Figure 10: Closed Loop Performance

We also observe that the trust system converges to values that we expect. The malicious nodes are segregated from the corrupted sensor nodes. The corrupted sensor and good nodes are treated equally in terms of trustworthiness. This is because the corrupted sensor nodes by following the message passing protocol, act as relay nodes only, thereby providing good estimates. These performance patterns are observed in Figure 11. We see from Figure 11a that the malicious nodes are quickly discovered and are isolated from participation in the filtering algorithm. From Figures 11b and 11c we observe that nodes which conform to the algorithm are treated equally. Their trust value in the system keeps increasing till it reaches T_{max}.

9 Conclusions

We have presented a generalized component model to address the problem of trusted distributed Kalman filtering in a large distributed sensor network. We find that trust methods implemented by a *trusted core* give significant improvement in the state estimation procedure. We also find that the interplay between estimation and trust updates quickly isolates malicious nodes and helps the observation limited nodes (corrupted sensor nodes).

Acknowledgment

This research was supported by award MURI W911-NF-0710287 from the Army Research Office.

References

[1] Needham R. Birell A., Lampson R. and Schroeder M. Global authentication without a global trust. In *IEEE Symposium on Security and Privacy*, May 1986.

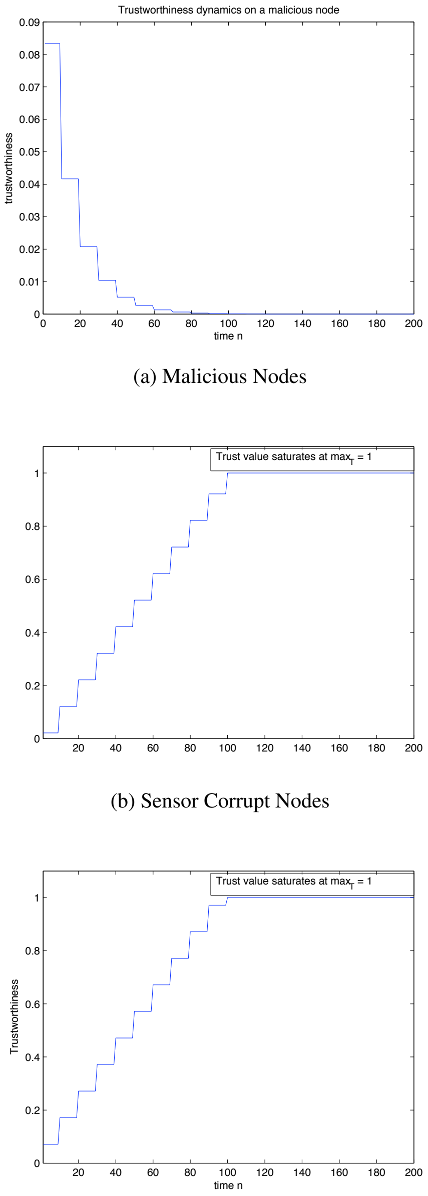
[2] Schenato L Carli R., Chiuso A. and Zampieri Sandro. Distributed kalman filtering based on consesus strategies. *IEEE Journal on Selected Areas in Communication*, 26(4):622–633, May 2008.

[3] Hussain F. K. Chang E., Dillon T. *Trust and reputation for service-oriented environments*. John Wiley and Sons, Ltd, 2006.

[4] Mori S. Chang K.C., Chong C-Y. On scalable distributed sensor fusion. In *11th International Conference on Information Fusion*, pages 1–8, 2008.

[5] Almeida J. Costa C. Reputation systems for fighting pollution in peer to peer file sharing systems. In *Peer-to-Peer Computing*, pages 53–60, 2007.

[6] Gambetta D. Can we trust trust. In *Trust:Making and Breaking Cooperative Relations*, pages 213–237. Basil Blackwell, 1988.



(a) Malicious Nodes

(b) Sensor Corrupt Nodes

(c) Good Nodes

Figure 11: Trust System performance

[7] Daley D.J. and Vere-Jones D. *An Introduction to the Theory of Point Processes*, volume 1. Springer, 2 edition, 2002.

[8] Meester R. Franceschetti M. *Random networks for communication*. Cambridge University Press, 2007.

[9] Burmester M. Hu J. Lars: a locally aware reputation system for mobile ad hoc networks. In *44th ACM Annual Southeast regional Conference*, pages 119–123, 2006.

[10] Buchegger S.m Le Boudec J-Y. Performance analysis of confidant protocol: Cooperation of nodes - fairness in dynamic ad-hoc networks. In *Proceedings MobiHoc*, 2002.

[11] Teles P. Marimon R., Nicolini J.P. Competition and reputation. http://www.utdt.edu/departamentos/economia/pdf-wp/WP002.pdf, 2000.

[12] Gnanapandithan N. Data detection and fusion in decentralized sensor networks. Master's thesis, K-State, Dept of Electrical and Computer Engineering, 2005.

[13] Sajid H. Pratik K.B. Special issue on information fusion in distributed sensor networks. *Information Fusion*, 9(3):330–331, 2008.

[14] Housely R. Public key infrastructure certificatte and crl profile. RFC2459, January 1999.

[15] Olfati-Saber R. Distributed kalman with embedded consensus filters. In *IEEE Conference on Decision and Control*, 2005.

[16] Olfati-Saber R. Distributed kalman filtering for sensor networks. In *IEEE Conference on Decision and Control*, pages 5492–5498, 2007.

[17] Znati T.F. Raghavendra C.S., Sivalingam K.M. *Wireless Sensor Networks*. Springer, 2004.

[18] Bhattacharjee B. Sherwood R., Lee S. Cooperative peer groups in nice. In *INFOCOM*, volume 2, pages 1272–1282, 2003.

[19] Johansson K.H Speranzon A., Fischione G. and Sangiovanni-Vinncentelli A. A distributed minimum variance estimator for sensor network. *IEEE Journal on Selected Areas in Communication*, 26(4):609–621, May 2008.

[20] Baras J.S. Theodorakopoulos G. On trust models and trust evaluation metrics for ad hoc networks. *IEEE Journal on Selected Areas in Communication*, pages 318–328, 2006.