

Secure Consensus Averaging in Sensor Networks Using Random Offsets

Mahdi Kefayati*[†], Mohammad S. Talebi*, Babak H. Khalaj* and Hamid R. Rabiee*[†]

*Sharif University of Technology and [†]Iran Telecommunication Research Center (ITRC)

Tehran, Iran

Email: kefayati@ce.sharif.edu, mstalebi@ee.sharif.edu, khalaj@sharif.edu, rabiee@sharif.edu

Abstract—In this work, we have examined the distributed consensus averaging problem from a novel point of view considering the need for privacy and anonymity. We have proposed a method for incorporating security into the scalable average consensus mechanisms proposed in the literature. Random Offsets Method (ROM) is lightweight, transparent and flexible since it is not based on cryptography, does not require any change in the fusion system and can be used optionally by some nodes who care about their privacy. In this method, which is based on noisification of nodes' information, we achieve robustness against $n - 1$ colluding adversaries in a network of n nodes, which is maximum level of robustness against collusions. Convergence and collusion robustness of ROM are analyzed mathematically and through simulation.

I. INTRODUCTION

While most of the past research in sensor networks is concentrated on power-aware networks for information gathering, recent trends show high potentials of sensor networks in formation of distributed information fusion networks [8] [13]. 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) [6]. Here we broaden our view from mere power constrained data gathering networks by considering a sensor network as a set of information sources (sensors) which are connected arbitrarily through wireless links. The goal of the network is not only gathering the information but also doing some processing. This processing is not limited to aggregation and may incur calculation of network wide parameters in a distributed and scalable manner. The result(s) of the processing must usually be available on all nodes. Such operations are known as consensus information fusion since all the network members reach an agreement on the calculated parameter. Averages and min/max parameters such as average remaining power, average number of neighbors and the minimum remaining energy are common examples of such calculations [1]. The application of consensus fusion, however, is not limited to MAC, routing or transport layer algorithms [11] and consensus fusion is usually the main application running on the network. Examples of such networks are employed in decentralized detection [12], distributed unknown parameter estimation [13] and distributed filtering [8] with applications in vehicle formation, flocking and robot position synchronization.

In an average consensus, the goal is to calculate $\bar{x} =$

$\frac{1}{n} \sum_i x_i$ in a distributed and scalable manner on all nodes assuming each node has an initial value, x_i . The consensus averaging problem is one of the fundamental problems of distributed sensor fusion; nevertheless, it is not limited to sensor fusion and as reflected by the literature has many applications and discussed in many disciplines [5] [12] [13]. This problem is recently explored by various researchers [7] [10]. Consensus averaging is even extended to form distributed Kalman filters (DKF) over sensor networks [6] and has many potential applications in distributed tracking and decision making in noisy environments [8].

In [7], Olfati-saber *et al.* have introduced a novel way for distributed consensus averaging in sensor networks based on local computation and exchange of information to direct neighbors iteratively. This method, which will be discussed in the succeeding section, converges even when the topology changes during the fusion process. Similar methods have also been proposed for distributed sensor fusion by Xiao and Boyd [13] and Talebi *et al.* [10]. Chen *et al.* [1] have also proposed another method for distributed computation of aggregates over sensor networks through distributed randomized algorithms. Their proposed method, *Distributed Random Grouping (DRG)*, exploits probabilistic random grouping over the sensors to calculate averages, minimums and maximums. Both of these methods are scalable and robust to topological changes which is a result of applying distributed local processing and single hop communication. In other words, these methods replace direct information routing with an information diffusion model.

The proposed methods, however, do not consider privacy of the information exchanged over the links nor the anonymity of nodes' information in the course of consensus averaging. Here, by privacy we mean preserving the secrecy of the nodes' information from its peers. This is a challenging issue since to process the information, the peers must become aware of it. In order to achieve privacy while making fusion possible, we actually have to obfuscate the relation of the sources with the information. This is also some form of anonymity. In other words, in order to keep the information anonymous, the association between the source and the information or the information itself must be obfuscated. Although the information can be protected over the air using cryptographic techniques, there is no cryptographic method suitable to address this issue efficiently. The most important shortcoming of the proposed methods stems from the fact that at least

some of the nodes must get aware of the information of some others to make information processing possible. Therefore, classical information fusion methods can not support member privacy intrinsically. The methods proposed for sensor network security [2] can not be employed to satisfy the requirements either since they are focused on communication security. Considering the energy and computation limitations and intermittent connectivity of common sensor networks, the overhead of any method employed for maintaining privacy and anonymity of the information should be considered as well.

In this paper, after going through some preliminaries and presentation of the system model in section II, we introduce and analyze *Random Offsets Method (ROM)*, a lightweight method for secure consensus averaging in section III. Section IV is dedicated to discussion about our simulation results and verification of the proposed properties. Finally, in section V, we conclude the paper and discuss our future work.

II. PRELIMINARIES, SYSTEM MODEL AND ASSUMPTIONS

We model the network as a graph, $G(V, E)$, which can be represented by its adjacency matrix, A_G . The number of the network members is represented as $n = |V|$. We also define N_i and N'_i as the set of neighbors of the node i and $N_i \cup \{i\}$ respectively. The *graph laplacian matrix* is defined as $L = D - A_G$, where D is the diagonal node degree matrix, i.e. $d_{ii} = |N_i|$ and $d_{ij} = 0$ for $i \neq j$.

In consensus averaging problems the goal is calculation of the average of nodes' information represented as x_i s. Without loss of generality, we focus on distributed and scalable consensus averaging methods. According to [7], each node keeps a temporary estimate of the average, z_i , which iteratively converges to the exact mean, $r = \bar{x} = \frac{1}{n} \sum_i x_i$. The temporary estimate is calculated based on the following equation:

$$z_i[m+1] = z_i[m] + \delta \sum_{j \in N_i} (z_i[m] - z_j[m]) \quad (1)$$

assuming $z_i[0] = x_i$. The whole system is formulated as:

$$\mathbf{z}[m+1] = \mathbf{z}[m] - \delta \mathbf{L} \mathbf{z}[m] \quad (2)$$

$$= (\mathbf{I} - \delta \mathbf{L}) \mathbf{z}[m] \quad (3)$$

where \mathbf{L} is the graph laplacian matrix and δ is the step size. The thorough analysis of the above and similar models is discussed in [7], [10], and [13] including their convergence behavior and stability.

Our focus is on cooperative networks where the nodes behave according to the consensus averaging protocol to achieve the fusion result. Nevertheless, adversaries may be interested in other nodes' information. In other words, we assume an honest-but-curious adversary model though the adversaries may exploit gathered information to launch attacks against other mechanisms in the network. Therefore, we have focused on methods for maintaining mutual privacy of network members' information in the course of consensus averaging process. In this scheme, none of the network members can get aware of the exact peer information even if it colludes with other peers. Discussion about non-cooperative networks

and related methods such as outlier detection and intrusion detection is out of the scope of the current work.

Privacy and anonymity, though conceptually different, are closely tied in distributed fusion systems especially for consensus goal functions. Information confidentiality is a requirement in many applications such as distributed voting and distributed tracking, and as perceived by the network members can be expressed as limiting nodes from getting aware of each others' information. Nonetheless, as information must be exchanged among members and considering the limitations of sensor networks, to make the fusion possible, classical cryptography and secure multiparty computation can not be exploited. Anonymity is a requirement in some applications like distributed voting and anonymous decision making as well. From another point of view, however, anonymity of the exchanged information leads into node privacy in a fusion system, as no one should be able to guess whose information is being processed.

III. PRIVATE AND ANONYMOUS CONSENSUS AVERAGING BASED ON RANDOM OFFSETS

A. Our Approach

Addressing the need for mutual privacy in information fusion networks seems to be a dilemma at the first glance. Our approach towards solving this problem is via introduction of a pre-fusion transformation, called anonymizer or anonymization transform denoted by $\mathbf{A}(\mathbf{X}) = \mathbf{X}'$, which obfuscates information in a way that the actual value of the nodes' information can not be derived while fusion is still possible with the transformed information. For distributed fusion schemes, anonymizer is implemented in a distributed manner as well. The local anonymizer can be modeled as $A_i(x_k)$ where $k \in N'_i$. We have also proposed a generalization of this model as *Blind Information Fusion Framework (BIFF)* in [3].

B. Random Offsets Method (ROM)

In random offsets method, we propose a novel anonymization transform which stochastically obfuscates information and does not require any change in the fusion function. The other major advantage of this method is that it does not depend on any cryptographic algorithm, the only additional functionality required is random number generation which can be readily available on the nodes as it has much more uses. This property is very useful specially in energy and computation bounded sensor networks which can not afford strong cryptography due to its high computational and energy costs. ROM is also transparent in the sense that no change in fusion system nor any additional protocol is required and can be optionally used by some nodes who care about their privacy while not used by others without any inaccuracy in the results.

Assuming the member values (x_i s) are identically distributed random variables (RV) represented as X , the average consensus result calculates $\bar{x} = \frac{1}{n} \sum_i x_i$, which is an unbiased minimum variance (UMV) estimate of the mean value of X , denoted by $E\{X\}$. The estimation error in MSE sense will

decay with the number of samples or analogously the number of network nodes. Therefore we have:

$$\lim_{n \rightarrow \infty} r = \lim_{n \rightarrow \infty} \bar{x} = \lim_{n \rightarrow \infty} \frac{1}{n} \sum_{i=0}^n x_i = E\{X\} \quad (4)$$

This exhibits the low-pass property of the consensus averaging which satisfies the *goal function anonymity* condition [3]. This property is also discussed from another point of view in [8]. We shall exploit this property to define our anonymization transform which not only keeps the fusion function intact, but also is robust against $n - 1$ colluding adversaries.

In order to form such an anonymization transform, we noisify the information of each node at anonymization phase with a high-pass noise. To accomplish this, each node chooses a random offset, o_i , at pre-fusion phase from a finite-variance zero-mean probability distribution function agreed by all nodes. This random offset is chosen once for each fusion process. Then each node adds its offset to its information. Therefore, o_i s are i.i.d. random variables which can be represented by O and $A_i(x_i) = x'_i = x_i + o_i$. In other words we shall define our anonymization transformation as:

$$\mathbf{X}' = \mathbf{A}(\mathbf{X}) = \mathbf{X} + \mathbf{O} \quad (5)$$

where $\mathbf{X} = [x_0, x_1, \dots, x_n]^T$, $\mathbf{O} = [o_0, o_1, \dots, o_n]^T$ and o_i s are the randomly chosen offsets known only by the corresponding node. For $n \rightarrow \infty$, according to equation 4, we have:

$$\lim_{n \rightarrow \infty} r' = E\{X + O\} = E\{X\} = \lim_{n \rightarrow \infty} r \quad (6)$$

since o_i s are chosen from a zero-mean probability distribution function. The above equation assumes offsets are independently chosen from information, which is a valid assumption in almost all cases. The low-pass property of the fusion function eliminates the effect of high-pass zero mean offsets added to the original information.

C. Accuracy

For real networks, the number of nodes is finite and therefore, we have a steady state error which does not decay as the number of consensus averaging iterations grows. For large values of n according to the central limit theorem or in case X and O are Gaussian RVs, the average steady state error, as perceived by each node, can be calculated as:

$$E_{SS} = E \left\{ \left(\frac{\sum_i (x_i + o_i)}{n} - \bar{x} \right)^2 \right\} = \frac{\sigma_o^2}{n} \quad (7)$$

The average initial error in \mathbf{X}' can be calculated as:

$$E_{Init} = E \left\{ \frac{\sum_i ((x_i + o_i) - \bar{x})^2}{n} \right\} = \sigma_x^2 + \sigma_o^2 \quad (8)$$

We define the *error reduction gain* or the *network processing gain (NPG)* as the ratio of the initial error to the steady state error. This parameter exhibits the ability of the network to omit the effect of the noise introduced to the network at pre-fusion phase. NPG can be calculated as:

$$NPG \triangleq \frac{E_{Init}}{E_{SS}} = n \frac{\sigma_x^2 + \sigma_o^2}{\sigma_o^2} = n(1 + a) \quad (9)$$

where $a \triangleq \frac{\sigma_x^2}{\sigma_o^2}$, can be considered as *noisification SNR*.

Equation 9, shows that error linearly decays with n as approved by our simulations as well. There is a clear trade-off on a : the less the information is noisified, the higher will be the gain as there will be less error in the system and for no obfuscation we have the perfect result; however, this parameter directly affects the ability of the network to hide member information: a higher a means lower obfuscation and analogously, higher confidence in deduction of the information of nodes.

D. Security

ROM has maximum robustness against collusion since even $n - 1$ colluding adversaries can not recover the exact information of a specific node. In case of continuous x_i s, the random offset is only known to the node itself and therefore, exact the x_i can not be guessed. This is a very interesting property of ROM due to the fact that *collusion robustness* [3] of consensus averaging goal function can not be more than $n - 2$ and $n - 1$ colluding adversaries can recover the single victim's information according to the following equation:

$$x_i = n\bar{x} - \sum_{j \neq i} x_j \quad (10)$$

This result is due to the stochastic approach taken to solve the secure consensus averaging problem. For higher security, nodes may even discard x_i s and offsets from their memory after the addition is done and replace x_i with $x'_i = x_i + o_i$. This way, the attackers can not find the real value stored in node memory through other ways like tampering the node. For discrete signals or cases in which the attackers try to run interval estimation, the normalized accuracy of the attack result is proportional to a , that is, the higher the information is noisified the lower will be the confidence of the estimation by the attackers or the wider will be the interval estimated for the victims' information assuming a fixed confidence bound is aimed by the attacker(s). The through analysis of the interval estimation attack by $n - 1$ colluding adversaries is given in the appendix.

Another main advantage of ROM is that it can work completely transparently from the fusion system. This property lets each node to decide about its privacy, that is, the ones who do not care about their information to be known by others or ones unable to generate the random offset can skip the anonymization phase at the cost of losing their privacy. This heterogeneity even improves the accuracy of the result as it reduces the amount of noise introduced to the system.

In case the fusion process is run multiple times or periodically and the value offered by the nodes are relatively constant over some of the runs, adversaries can try to improve their estimation by averaging over the values proposed by the victim node. In order to guard against such attacks, nodes can choose a new random offset only when their information changes. On the other hand, choosing a new random offset at each fusion process can help hiding changes in the input value from the adversaries.

IV. SIMULATION RESULTS

We verified our results by running a large set of simulations. We considered 100 uniformly distributed sensors over a $100m \times 100m$ area and nodes up to d meters apart were considered neighbors (i.e. free space propagation model with range of sight d).

Figure 1 shows the mean square error (MSE) of the average consensus result defined as $(z_i[m] - \bar{x})^2$ on nodes with maximum (solid) and minimum (dotted) connectivity degree as well as average MSE (dashed lines) for both normal and ROM based average consensus vs. number of iterations. According to this figure, steady state MSE of the normal consensus model converges to zero with number of iterations; however, as previously discussed, ROM reaches a steady state MSE, which is inversely proportional to the number of members in the network and directly related to the noisification power (σ_o^2) according to equation 7. For the sample case presented in figure 1, $n = 100$ and $a = \frac{\sigma_x^2}{\sigma_o^2} = 1 \equiv 0 \text{ dB}$, -28 dB average steady state error is measured. It must be noted that even for normal average consensus, bounded number of iterations lead to a steady state error, which is normally set to tolerable error margin of the system. On the other hand, the number of iterations might be bounded by the desired lifetime of the network since the number of iterations affects the amount of messages exchanged and analogously the energy consumed by the nodes.

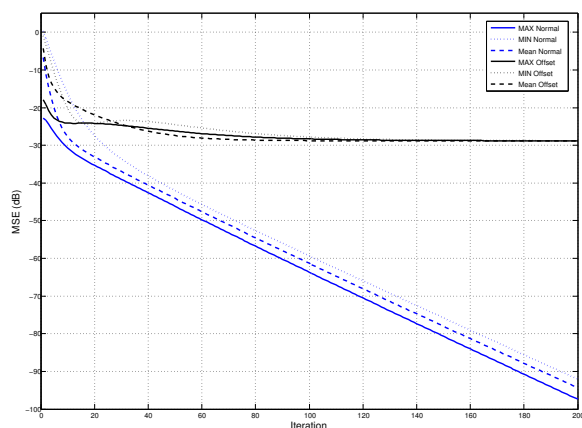


Fig. 1. MSE vs. number of iterations; dotted lines: minimum degree nodes, solid lines: maximum degree nodes, dashed lines: average. $\frac{\sigma_x^2}{\sigma_o^2} = 0 \text{ dB}$, range of sight = $40m$.

The linear relation of NPG with number of network members is verified through extensive simulation and data analysis. The results are presented in figure 2, which depicts NPG vs. number of network nodes. Each point represents average NPG of 1000 randomly generated networks with the respective n . The lines are fit to the measured points to calculate the slope and verify equation 9. The linear fit results are presented in table I. Here we only present the results for normally distributed sensor values. Nonetheless, our simulations showed that as n increases, the results for sensor values chosen from

other probability distributions become more similar to the results of the normally distributed sensor values.

For discrete and bounded node values, a similar approach can be taken with the values rounded to the nearest discrete value in the system. The results are the same due to the similar nature of the method employed in both cases. Depending on the application, for continuous results, NPG can be a good measure of the performance of the method while for discrete case, it is more meaningful to calculate the relation of the probability of error in guessing a sensor value before fusion (perceived by malicious guesser) to the probability of error in the consensus result. The detail analysis of such and related topics is left for future work.

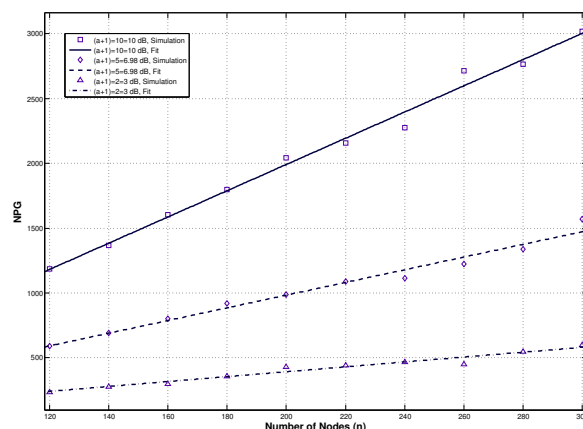


Fig. 2. Network Processing Gain (NPG) vs. Number of Nodes (n).

TABLE I
RESULTS OF LINEAR FITS FOR FIGURE 2.

$(a + 1)$	fit slope	95% confidence bounds	r^2
10	10.11	9.285, 10.94	0.99
5	4.89	4.258, 5.525	0.97
2	1.885	1.556, 2.214	0.95

V. CONCLUSION AND FUTURE WORK

In this paper, we have analyzed consensus averaging problem from a security perspective, considering the need for mutual privacy in the course of fusion process. We proposed *Random Offsets Method (ROM)* as a solution to the problem of *blind consensus averaging* which exploits the properties of consensus averaging fusion function in elimination of high-pass noise and obfuscates the nodes' information through noisification before consensus averaging takes place. We characterized this property and its relation to network size by introducing *network processing gain (NPG)* which exhibits the ability of the network to eliminate obfuscation noise. We also analyzed collusion resistance of ROM, reduced it to an estimation problem, and showed that ROM is robust to up to $n - 1$ colluding honest-but-curious adversaries, which is more than the intrinsically possible limit for consensus averaging ($n - 2$). This interesting property is due to the stochastic

approach taken towards hiding node information. Last, but not least, we derived the relation between NPG and the power of the attack lunched by $n - 1$ colluding adversaries. ROM suffers from steady state error in final result. The amount of this steady state error is inversely proportional to number of nodes or analogously NPG and shown to be negligible in real-world applications. We consider the methods for eliminating this error through various ways including proper choice of the noise p.d.f. for future work along with extension of this method to problems beyond consensus averaging. Dealing with non-cooperative adversaries is to be investigated in future as well.

APPENDIX

ACCURACY OF INTERVAL ESTIMATION ATTACKS ON ROM

Without loss of generality, we assume that victim is the 1st node, i.e. the goal of the attackers is to find x_1 . For exact fusion result, as depicted in 10, x_1 can be easily recovered by $n - 1$ colluding nodes.

However, in ROM, as presented before, $r' = \bar{x}'$ is a random variable with mean value r and variance $\frac{\sigma_o}{n}$. If the attackers try to estimate x_1 based on a modification of 10 by replacing x'_i s with x_i s known to the attackers we have:

$$\tilde{x}_1 = n\bar{x}' - \sum_{j=2}^n x_j \quad (11)$$

where \tilde{x}_1 is the estimated value of x_1 . In order to evaluate the accuracy of this estimate, we calculate $\sigma_{\tilde{x}_1}^2$, the variance of \tilde{x}_1 :

$$\sigma_{\tilde{x}_1}^2 = n^2 \sigma_{x'}^2 = n \sigma_o^2 \quad (12)$$

Therefore, if the attackers exercise this method for interval estimation on x_1 , though $E\{\tilde{x}_1\} = x_1$, the variance of the attack will be intensified by n . This is a single sample estimation and therefore, for normal offsets and assuming ζ_u is the standard normal u percentile of \tilde{x}_1 , γ confidence interval of x_1 will be $\tilde{x}_1 \pm \zeta_u \sigma_o \sqrt{n}$ [9]. This means that x_1 will be in $[-\zeta_u \sigma_o \sqrt{n}, \zeta_u \sigma_o \sqrt{n}]$ interval round \tilde{x}_1 with probability $\gamma = 1 - 2(1 - u)$. Assuming normal x_i s, the normalized estimated interval using the above method will be $1 \pm \zeta_u \sqrt{\frac{n}{a}}$.

The attackers can also use another method to estimate x_1 , which is based on directly attacking the anonymization transform. Based on 5, we have:

$$x'_1 = x_1 + o_1 \quad (13)$$

consequently, we have to solve a single sample detection problem for finding x_1 from x'_1 , assuming both x_i s and o_i s are normal random variables, the γ confidence interval for estimation of x_1 from single sample, x'_1 , is $x'_1 \pm \zeta_u \sigma_o$, which is more precise than the attack based on equation 10. Therefore, normalized γ confidence interval of estimated x_1 will be $1 \pm \frac{\zeta_u}{\sqrt{a}}$.

From the above discussion we can see the role of NPG and the trade-off on it very clearly: higher a will improve the attackers confidence interval while remedying the steady state error of consensus averaging. In other words, changing a can improve the accuracy of estimation from the average and the

accuracy of the attackers from node information. Therefore, to improve the accuracy of the estimation of the average while keeping the success probability of the attackers constant, the number of the network members must be increased.

ACKNOWLEDGMENTS

The authors would like to thank Reza Olfati-Saber for our discussion on sensor networks and fusion mechanisms, Mohammad H. Falaki for our discussion on the need for distributed security and members of Sharif Digital Media Lab. (DML) for their invaluable cooperation.

This work was supported by Iran Telecommunication Research Center (ITRC) and Sharif Advanced Information and Communication Technology Center (AICTC).

REFERENCES

- [1] J. Chen, G. Pandurangan and D. Xu. "Robust Aggregates Computation in Wireless Sensor Networks: Distributed Randomized Algorithms and Analysis", in *IEEE Transactions on Parallel and Distributed Systems*, Vol. 17, No. 9, 2006.
- [2] C. Karlof and D. Wagner. "Secure Routing in Wireless Sensor Networks: Attacks and Countermeasures," *Elsevier's AdHoc Networks Journal, Special Issue on Sensor Network Applications and Protocols*, Volume 1, No. 2-3, pages 293-315, September 2003.
- [3] Mahdi Kefayati, Mohammad S. Talebi, Hamid R. Rabiee and Babak H. Khalaj. "On Secure Consensus Information Fusion over Sensor Networks," *Proceedings of 2007 ACS/IEEE International Conference on Computer Systems and Applications (AICCSA 2007)*, Amman, Jordan, May 2007.
- [4] Mahdi Kefayati, Mohammad S. Talebi, Hamid R. Rabiee and Babak H. Khalaj. "Secure Consensus Averaging for Secure Information Fusion in Sensor Networks," *Proceedings of The 10th International Symposium on Signal Processing and its Applications (ISSPA 2007)*, Sharjah, UAE, Feb. 2007.
- [5] Nancy Lynch. "*Distributed Algorithms*," Morgan Kaufman Publishers, San Mateo, CA, 1996.
- [6] R. Olfati-Saber. "Distributed Kalman Filter with Embedded Consensus Filters," *Proceedings of 44th IEEE Conference on Decision and Control and 2005 European Control Conference*, Dec. 2005.
- [7] 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, Sept. 2004.
- [8] R. Olfati-Saber and Jeff S. Shamma. "Consensus Filters for Sensor Networks and Distributed Sensor Fusion," *Proceedings of 44th IEEE Conference on Decision and Control and 2005 European Control Conference*, Dec. 2005.
- [9] Athanasios Papoulis and S. Unnikrishna Pillai. "Probability, Random Variables and Stochastic Processes, 4th ed.," McGraw-Hill, 2002.
- [10] Mohammad S. Talebi, Mahdi Kefayati, Babak H. Khalaj and Hamid R. Rabiee. "Adaptive Consensus Averaging over Sensor Networks," *Proceedings of the 3rd IEEE Conference on Mobile Ad hoc and Sensor Systems (MASS 2006)*, 2006.
- [11] Mohammad S. Talebi and Babak H. Khalaj. "Distributed Power Allocation for OFDM Wireless Ad hoc Networks Based on Average Consensus," *10th IEEE International Conference on Communication Systems 2006 (IEEE ICCS'06)*, Singapore, October 2006.
- [12] J. N. Tsitsiklis. "Decentralized Detection;" in *Advances in Signal Processing*, Vol. 2, H. V. Poor and J. B. Thomas, editors, JAI Press, pp. 297-344, 1993.
- [13] Lin Xiao, Stephen Boyd and Sanjay Lall. "A space-time diffusion scheme for peer-to-peer least-squares estimation," *Proceedings of Fifth International Conference on Information Processing in Sensor Networks (IPSN 2006)*, pp. 168-176, Nashville, TN, April 2006.