A GOSSIPING APPROACH TO SAMPLING CLOCK SYNCHRONIZATION IN WIRELESS ACOUSTIC SENSOR NETWORKS

Joerg Schmalenstroeer, Patrick Jebramcik, Reinhold Haeb-Umbach

Department of Communications Engineering, University of Paderborn, Germany
{schmalen,haeb}@nt.uni-paderborn.de

ABSTRACT

In this paper we present an approach for synchronizing the sampling clocks of distributed microphones over a wireless network. The proposed system uses a two stage procedure. It first employs a two-way message exchange algorithm to estimate the clock phase and frequency difference between two nodes and then uses a gossiping algorithm to estimate a virtual master clock, to which all sensor nodes synchronize. Simulation results are presented for networks of different topology and size, showing the effectiveness of our approach.

Index Terms— Gossip algorithm, clock synchronization, wide area sensor networks

1. INTRODUCTION

While wide area sensor networks (WASN) have been a research topic for many years [1], acoustic sensor networks have gained increased interest only recently [2]. If audio signal processing algorithms, such as beamforming or source localization, are realized with distributed microphones, a tight sampling rate synchronization is required. For example, if the time difference of arrival of an impinging audio signal is to be estimated in the presence of a sampling frequency mismatch between two microphones, it cannot be decided whether an observed delay change is caused by a moving speaker or is just due to differences in the sampling frequencies.

The estimation of clock frequency offsets can either be accomplished at the baseband audio signal processing side by analyzing the sampled signals [3, 4] or by the exchange of time stamps via a communication link between the sensors [5]. In [3] the coherence function between the microphone signals was evaluated in times of speech absence to blindly estimate the clock frequency offset. Assuming a spatially stationary source, the frequency offset was estimated as a linear phase shift in the short-time Fourier domain in [4]. For both approaches, the assumptions about the input signals are quite restrictive, and it is unclear how these approaches can be scaled to large sensor networks with many nodes.

The exchange of time stamps, which can be realized as either a one or two-way message exchange, is a well established method for the synchronization of communication networks (e.g. network time protocol [6]). Chaudhari proposed a simple and robust two-way exchange algorithm [7, 8] which we successfully used in our previous work for the synchronization of the sampling rate of a microphone to a master clock via a wireless link [9].

Once the clock frequency and phase offsets have been estimated the signals need to be adjusted for the subsequent audio processing. This can be either accomplished by resampling or in hardware by adjusting the oscillators of the A/D devices. While the first can be carried out in software, the latter does not incur any additional computational effort, however requires an adjustable frequency synthesizer. In our prior work we adopted the latter approach by employing a Direct Digital Synthesis (DDS) device by which the sampling rate can be adjusted in mHz precision [9].

In this paper we extend our approach to the synchronization of large networks comprising many sensor nodes. We adopt a gossiping strategy [10] to achieve a network wide clock synchronization by only local message passing between neighboring nodes, similarly to the consensus approaches of [11] and [12]. A distributed clock synchronization has also been attempted in [13], where a distributed Kalman filter is developed to estimate a virtual master clock. In contrast to our approach it requires a network in beacon-enabled mode with dedicated time-slots to work properly.

The paper is organized as follows: First we briefly review the time stamp exchange protocol in Sec. 2. Then we discuss a model for the estimation error in Sec. 3 and derive the Kalman Filter in Sec. 4. The gossiping algorithm is described in Sec. 5 and after presenting simulation results in Sec. 6 we end up with some conclusions drawn in Sec. 7.

2. TIME STAMP EXCHANGE

We use the two-way message exchange algorithm of Chaudari [8] to estimate the clock frequency and phase offset between two sensor nodes, see Fig. 1.

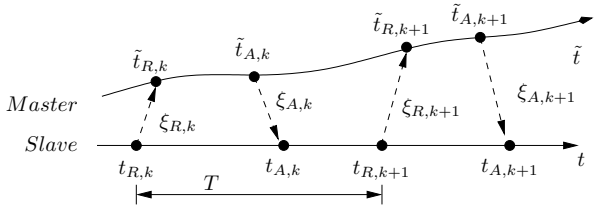

Fig. 1. Two-way time stamp message exchange

The exchange is initiated by the slave node by sending a time stamp $t_{R,k}$ to the master. The packet arrives after a time $\xi_{R,k}$ (measured in the time basis of the slave) at the master node, whose local time is $\tilde{t}_{R,k}$. The master node combines the time stamps $t_{R,k}$, $\tilde{t}_{R,k}$ and $\tilde{t}_{A,k}$ in a packet and sends it back to the slave node at time $\tilde{t}_{A,k}$. After a network transmission time of $\xi_{A,k}$ the slave node receives the packet.

If we denote the ratio of the oscillator frequency of the slave to that of the master by $(1+\epsilon)$, the following relationship between the time stamps can be obtained [8]:

$$\tilde{t}_{R,k} = (t_{R,k} + \xi_{R,k}) \cdot (1+\epsilon) + \varphi \quad (1)$$

$$\tilde{t}_{A,k} = (t_{A,k} - \xi_{A,k}) \cdot (1+\epsilon) + \varphi \quad (2)$$

and similarly for the $(k+1)$-st time stamp exchange. Here, φ is the phase offset between the two time bases. With $\Delta\tilde{t}^+ = (\tilde{t}_{R,k+1} - \tilde{t}_{A,k})$, $\Delta\tilde{t}^- = (\tilde{t}_{R,k} - \tilde{t}_{A,k+1})$ and the corresponding definition for Δt^+ and Δt^- we can obtain an estimate for ϵ as follows (see [8, 9] for details)

$$\begin{aligned}\tilde{\epsilon} &= \frac{\Delta\tilde{t}^+ - \Delta\tilde{t}^-}{\Delta t^+ - \Delta t^-} - 1 \\ &\approx \epsilon + \frac{(\xi_{R,k+1} - \xi_{R,k}) + (\xi_{A,k} - \xi_{A,k+1})}{(t_{R,k+1} - t_{R,k}) + (t_{A,k+1} - t_{A,k})}. \end{aligned} \quad (3)$$

3. OBSERVATION ERROR DISTRIBUTION

The unknown transmission times ξ of eq. (3) can be modeled as the sum of three contributions: A constant minimum delay T_c, multiples of a delay T_d, which are caused by medium access (MAC) wait times, and an exponentially distributed random component T_r, e.g., due to hardware delays. With $\xi = T_c + k \cdot T_d + T_r$, eq. (3) becomes

$$\tilde{\epsilon} \approx \epsilon + \left(\frac{k' \cdot T_d}{2T}\right) + \left(\frac{T_r'}{2T}\right) =: \epsilon + v_\epsilon. \quad (4)$$

Here, $k'T_d$ summarizes the delays caused by medium access, while T_r' captures the exponentially distributed delays. Further, the denominator in eq. (3) has been approximated by $2T$, where T is the time delay between two time stamp exchanges, see Fig. 1.

Fig. 2 displays a measured histogram of the observation error term v_ϵ of eq. (4). The measurement has been obtained by connecting two sensor nodes to the same physical crystal oscillator, thus ensuring that ϵ=0, and estimating the frequency offset by the time stamp exchange according to eq. (3). Here, the two error terms can be clearly distinguished: the large errors caused by the medium access are those which are multiples of some delay T_d, while the random component T_r' is much smaller and causes the (hardly visible) widening of the peaks.

The error histogram can be approximated by a Gaussian mixture model (GMM) $p(v_\epsilon) = \sum_{m=0}^{M-1} \gamma_m \mathcal{N}(v_\epsilon; \mu_{m,\epsilon}, \sigma_\epsilon^2)$, where the $\mu_{m,\epsilon}$ model the medium access delays (large-scale errors) and the variations around the $\mu_{m,\epsilon}$ of variance σ_ϵ^2 are caused by T_r' (small-scale errors). Indeed, the sum of four exponentially distributed random variables, two of them with a negative sign, can be well approximated by a zero-mean Gaussian.

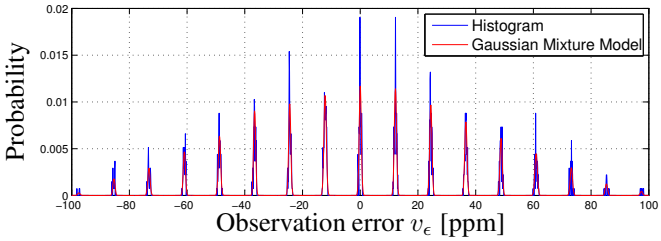

Fig. 2. Histogram of observation error v_ϵ (obtained from approximately 6 hours of data) and GMM approximation

From eq. (4) we can conclude that increasing the time T between two time stamp exchanges reduces both the means $\mu_{m,\epsilon}$ and the variance σ_ϵ^2 of the observation error. Of course, T cannot be made arbitrarily large, since then one would not be able to track the fluctuations in ϵ.

Note that an estimate $\tilde{\varphi}$ for φ can also be derived from eq. (1) and eq. (2) [8]. Its measurement error has a similar GMM-like shape as the one above, and its mixture specific variance is denoted by σ_φ^2 in the following.

4. REDUCTION OF OBSERVATION ERROR

In the following we show how the estimates $\tilde{\epsilon}$ and $\tilde{\varphi}$ obtained from the time stamp exchange protocol can be improved. Due to the nature of the error term v_ϵ, the large-scale errors $\mu_{m,\epsilon}$ need special attention, while the small-scale errors are taken care of by a Kalman filter.

4.1. Kalman Filter

The task of the Kalman filter is to track the frequency offset $\epsilon(n)$, its temporal drift $\Delta\epsilon(n)$, and the phase offset $\varphi(n)$ using the state vector $\boldsymbol{x}(n) = [\varphi(n), \epsilon(n), \Delta\epsilon(n)]^T$. The state and measurement equations are given by

$$\boldsymbol{x}(n+1) = \boldsymbol{F} \cdot \boldsymbol{x}(n) + \boldsymbol{G} \cdot \boldsymbol{v}_S \quad (5)$$

$$\boldsymbol{z}(n) = \boldsymbol{H}^T \cdot \boldsymbol{x}(n) + \boldsymbol{v} \quad (6)$$

with the matrices

$$\boldsymbol{F} = \begin{pmatrix} 1 & T \cdot f_0 & 0 \\ 0 & 1 & T \\ 0 & 0 & 1 \end{pmatrix} \quad \boldsymbol{G} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad \boldsymbol{H} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \end{pmatrix}$$

$$E[\boldsymbol{v}_S \boldsymbol{v}_S^T] = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & \sigma_{\Delta\epsilon}^2 \end{pmatrix} \quad E[\boldsymbol{v}\boldsymbol{v}^T] = \begin{pmatrix} \sigma_\varphi^2 & 0 \\ 0 & \sigma_\epsilon^2 \end{pmatrix} \quad (7)$$

where f_0 is the targeted frequency and $\sigma_{\Delta\epsilon}^2$ models the fluctuations of the crystal oscillator, e.g., due to environmental effects. The input to the Kalman filter are the estimates of the message exchange protocol: $\boldsymbol{z}(n) = (z_1(n)\ z_2(n))^T$, where $z_1(n) = \tilde{\varphi}(n)$ and $z_2(n) = \tilde{\epsilon}(n)$.

If we assume that the prediction of the Kalman filter $\hat{\epsilon}(n|n-1)$ is close to the true value $\epsilon(n)$:

$$|\hat{\epsilon}(n|n-1) - \epsilon(n)| \ll \frac{1}{2}\frac{T_d}{2T}, \quad (8)$$

then the large-scale observation error can be uniquely determined by finding that mixture component that is closest to the prediction

$$\hat{m} = \underset{m}{\operatorname{argmin}} \, |\hat{\epsilon}(n|n-1) - (\tilde{\epsilon}(n) - \mu_{m,\epsilon})| \qquad (9)$$

and removed by subtracting the mean of the identified mixture component from the observation: $z_2'(n) = z_2(n) - \mu_{\hat{m},\epsilon}$. Then $z_2'(n)$ is presented to the Kalman filter instead of $z_2(n)$. The analogue procedure is also applied to the phase offset observation $z_1(n)$.

4.2. Initialization

In the beginning the Kalman filter will not deliver sufficiently exact estimates to identify the large-scale error $\mu_{m,\epsilon}$. If this error term is wrongly identified the Kalman filter may get locked at a poor estimate which differs from the true value by a multiple of the GMM inter mean distance. However, it is easy to detect this situation by the following two criteria. Firstly, it can be observed that, although the deviation from the master clock is small, the offset keeps increasing rapidly over time. Secondly, the mean of all observations is not zero.

These two criteria are checked constantly, not only during initialization, and if they indicate a false lock, the Kalman filter estimate is corrected by the detected mean error.

5. GOSSIPING ALGORITHM

So far we have considered the synchronization among two nodes. However, an acoustic sensor network may consist of many more nodes. To synchronize all nodes to a common sampling frequency we will employ a gossiping strategy [10], where nodes only exchange messages between their immediate neighbors with the goal to synchronize to a virtual master node whose clock frequency is given by the average of the frequencies of the nodes.

This approach has several advantages compared to synchronizing to a predefined physical master node. Firstly, it avoids the problem of introducing a single point of failure. Secondly, on average all nodes only have to adapt their sampling rate by a small factor. Thirdly, network topology changes, e.g., nodes entering or leaving the network, do not interrupt the synchronization process. However, exchanging information only between neighboring nodes will increase the time until a steady state is reached.

5.1. Information exchange

The task of the gossiping algorithm is to iteratively estimate, for each node, the sampling rate deviation to a virtual master node, whose sampling rate equals the average sampling rate of all nodes of the network, by only local information exchange between neighboring nodes. Let $G = (\mathcal{N}, \mathcal{C})$ denote the graph with $\mathcal{N} = \{N_1, \dots N_k, \dots N_P\}$ being the set of nodes and $\mathcal{C}$ the set of edges between the nodes. Now consider the message exchange of some node N_k with its neighbors (see Fig. 3).

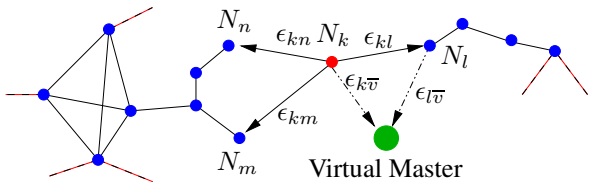

Fig. 3. Example network topology with highlighted node N_k and virtual master node.

After conducting the time stamp exchange protocol node N_k knows the frequency deviations $\{\epsilon_{kl}, \epsilon_{km}, \epsilon_{kn}\}$ to its neighbor nodes. From the last iteration of the gossip algorithm it further knows the deviations $\{\epsilon_{l\overline{v}}, \epsilon_{m\overline{v}}, \epsilon_{n\overline{v}}\}$ of the neighboring nodes from the virtual master clock. Note, that the deviation towards the virtual master node equals the value the node will change its oscillator frequency during the adjustment phase (after the gossiping) since the goal is to synchronize all nodes to the virtual master clock.

For the next gossiping iteration the node has to select an adjacent node to exchange data with. We did exhaustive simulations of several variants of gossip algorithms (e.g. [14]) to determine the optimal decision rule for choosing which neighboring node to exchange data with. The fastest synchronization was achieved by selecting the neighbor with the largest deviation to be expected after performing the sampling rate adjustment, i.e., that node $N_\alpha \in \mathcal{N}_k$, for which $(\epsilon_{k\alpha} - \epsilon_{k\overline{v}} + \epsilon_{\alpha\overline{v}})$ assumes the largest value, where $\mathcal{N}_k$ is the set of nodes adjacent to node N_k. For the example of Fig. 3 assume that node N_l has been selected. Node N_k then sends the information $[\epsilon_{kl}, \epsilon_{k\overline{v}}]$ to this node. Node N_l computes the average

$$\overline{\epsilon}_{lk} = \left(\frac{\epsilon_{lk}}{2} + \frac{-\epsilon_{kl}}{2} \right) \qquad (10)$$

and calculates the deviation after adjustment with

$$\overline{\epsilon}_{lk}^{(A)} = \left(\overline{\epsilon}_{lk} - \epsilon_{l\overline{v}} + \epsilon_{k\overline{v}} \right). \qquad (11)$$

Subsequently, the deviation towards the virtual master (i.e., the adjustment value for the clock frequency synthesizer of node N_l) is updated to

$$\epsilon_{l\overline{v}} \leftarrow \epsilon_{l\overline{v}} + \frac{\overline{\epsilon}_{lk}^{(A)}}{2} \qquad (12)$$

and the values $\left[\overline{\epsilon}_{lk}^{(A)}, \epsilon_{l\overline{v}}\right]$ are transmitted to node k, such that it can update $\epsilon_{k\overline{v}}$.

This procedure is carried out for all nodes of the network and iterated until convergence or until a predefined number of iterations is reached.

The same gossiping strategy is also employed for the phase offset estimates, such that in the end of the gossiping iterations all nodes have estimates of their frequency deviation $\epsilon_{k\overline{v}}$ and their phase offset $\varphi_{k\overline{v}}$ to the virtual master node.

Then nodes N_k, k=1,...P, will adjust their oscillator fre-

quencies according to $\epsilon_{k\overline{v}}$. Note that a controller is employed to compute the oscillator adjustment from $\epsilon_{l\overline{v}}$ and $\varphi_{l\overline{v}}$. The discussion of the controller is, however, beyond the scope of this paper.

6. SIMULATION RESULTS

In prior work a sensor node has been realized in hardware to test the clock synchronization to a master clock via a wireless communication link [9]. However, the test of the gossiping algorithm for large networks could only be carried out by simulation. Wherever possible, the parameters used in the simulations had been derived from measurements on our hardware. The time between two message exchanges $(t_{A,k+1} - t_{A,k})$ is set to $T{=}10\,\mathrm{s}$, which is also the available processing time for the gossiping. Each node tries to access the medium approximately every $100\,\mathrm{ms}$, which equals a maximum of 100 synchronization attempts per node shared across the set of connections the node has. Since the medium access control can only provide one node at a time the privilege to use the medium and the time basis of the nodes are not synchronized, this kind of communication is more like an asynchronous time model than a synchronous one.

Figure 4 illustrates the synchronization of a network with 25 nodes in terms of the clock frequency deviation ϵ from the master, measured in ppm. The network topology has been generated at random with a node having on average three neighbors. Starting with random deviations between $\pm 50\,\mathrm{ppm}$ our algorithm synchronizes the network within $3\,\mathrm{min}$ such that the RMS deviation to the master equals $0.04\,\mathrm{ppm}$ (maximum deviation $\leq 0.19\,\mathrm{ppm}$).

The top figure of Fig. 5 shows the steady-state RMS error of the phase where for each network size (in terms of nodes) 100 randomly generated topologies have been simulated. The RMS error is expressed as the percentage of a sampling interval. The red curve indicates the average over the different topologies, and the bars show the largest and smallest value, respectively. Clearly, the synchronization performance depends on the network topology: A network with clusters of nodes and only few edges between the clusters will need more time for settling on a common sampling rate than a network without bottleneck nodes.

The bottom figure of Fig. 5 displays the largest phase error observed during $60\,\mathrm{min}$ of simulation time, averaged over the 100 randomly generated network topologies for different

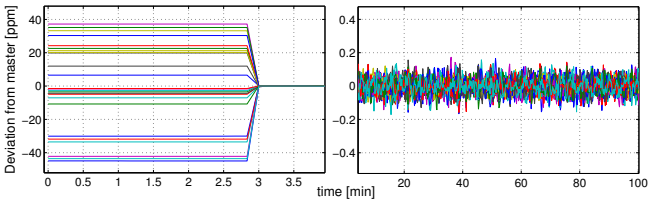

Fig. 4. Development of the frequency deviation ϵ of 25 sensor nodes to the virtual master node over time

Maximum phase error [%]	0	5	10	35	50	60
RMS Position error [m]	0.06	0.06	0.07	0.10	0.19	0.42

Table 1. Influence of offset errors on position estimates.

numbers of nodes. The bars indicate the values for the best and worst performing topology, respectively. Overall, Fig. 5 seems to indicate that the network topology has a larger impact on the synchronization performance than the network size.

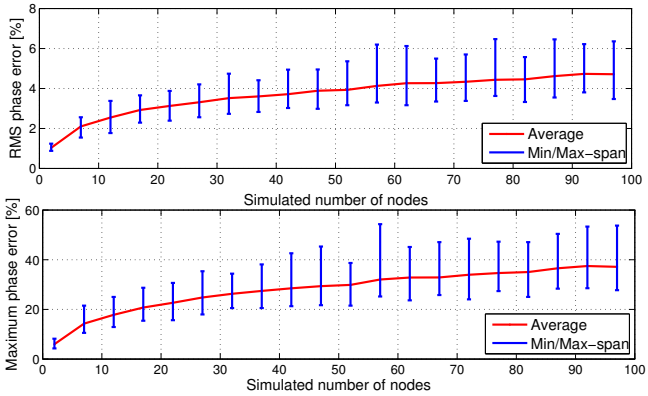

Fig. 5. Simulation results of random graphs in terms of the RMS phase error (top) and maximum phase error (bottom)

6.1. Impact on position estimation

We simulated the microphone signals of a 5-element circular microphone array of $2\,\mathrm{m}$ diameter, which are caused by a moving speaker in a reverberation-free room of size $10 \times 10\,\mathrm{m}$. The speaker position was estimated with an interpolated GCC-PHAT approach [15] with a global coherence field analysis [16] (grid distance $1\,\mathrm{cm}$). In Tab. 6.1 the root-mean-square positioning error is given for different synchronization precisions. For example, if the maximum phase error remains below $35\ \%$ of a sampling interval the sampling time jitter causes a positioning error of $10\,\mathrm{cm}$. According to Fig. 5 the proposed synchronization scheme is able to keep the maximum phase offset below $35\ \%$ on average for networks comprising as many as 80 nodes.

7. CONCLUSIONS & CONTRIBUTIONS

In this paper we extended our ideas about synchronizing two acoustic sensor nodes by time stamp exchange over a wireless link [9] towards the application in large sensor networks. Compared to [9] an improved Kalman filter is derived for clock phase and frequency offset estimation between two sensor nodes. Further, a gossiping algorithm is presented to synchronize a sensor network to a virtual master clock. Networks of different size and topology have been simulated to evaluate the synchronization performance. Even for networks as large as 100 nodes the RMS clock phase error could be kept below 6% of a sampling period.

8. REFERENCES

[1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless sensor networks: a survey," *Computer Networks*, vol. 38, pp. 393–422, 2002.

[2] R. Heusdens, G. Zhang, R.C. Hendriks, Y. Zeng, and W.B. Kleijn, "Distributed MVDR beamforming for (wireless) microphone networks using message passing," in *Proc. International Workshop on Acoustic Signal Enhancement (IWAENC)*, Aachen, 2012.

[3] S. Markovich-Golan, S. Gannot, and I. Cohen, "Blind sampling rate offset estimation and compensation in wireless acoustic sensor networks with application to beamforming," in *Proc. International Workshop on Acoustic Signal Enhancement (IWAENC)*, Aachen, 2012.

[4] S. Miyabe, N. Ono, and S. Makino, "Blind compensation of inter-channel sampling frequency mismatch with maximum likelihood estimation in the STFT domain," in *Proc. ICASSP*, Vancouver, May 2013.

[5] Jeremy Elson and Kay Römer, "Wireless sensor networks: a new regime for time synchronization," *SIGCOMM Comput. Commun. Rev.*, vol. 33, no. 1, pp. 149–154, Jan. 2003.

[6] David L. Mills, "Internet time synchronization: the network time protocol," *IEEE Transactions on Communications*, vol. 39, pp. 1482–1493, 1991.

[7] Q.M. Chaudhari, E. Serpedin, and K. Qaraqe, "On maximum likelihood estimation of clock offset and skew in networks with exponential delays," *IEEE Transactions on Signal Processing*, vol. 56, no. 4, pp. 1685–1697, April 2008.

[8] Q.M. Chaudhari, "A simple and robust clock synchronization scheme," *IEEE Transactions on Communications*, vol. 60, no. 2, pp. 328–332, February 2012.

[9] Joerg Schmalenstroeer and Reinhold Haeb-Umbach, "Sampling rate synchronisation in acoustic sensor networks with a pre-trained clock skew error model," in *Proc. 21th European Signal Processing Conference, EUSIPCO'13*, 2013.

[10] Stephen Boyd, Arpita Ghosh, Balaji Prabhakar, and Devavrat Shah, "Randomized gossip algorithms," *IEEE Transactions on Information Theory*, vol. 14, no. SI, pp. 2508–2530, June 2006.

[11] L. Schenato and G. Gamba, "A distributed consensus protocol for clock synchronization in wireless sensor network," in *46th IEEE Conference on Decision and Control*, Dec 2007, pp. 2289–2294.

[12] M.K. Maggs, S.G. O'Keefe, and D.V. Thiel, "Consensus clock synchronization for wireless sensor networks," *IEEE Sensors Journal*, vol. 12, no. 6, pp. 2269–2277, June 2012.

[13] Fabian Kirsch and Martin Vossiek, "Distributed Kalman filter for precise and robust clock synchronization in wireless networks," in *Proceedings of the 4th international conference on Radio and wireless symposium*, Piscataway, NJ, USA, 2009, RWS'09, pp. 455–458, IEEE Press.

[14] D. Ustebay, B.N. Oreshkin, M.J. Coates, and M.G. Rabbat, "Greedy gossip with eavesdropping," *IEEE Transactions on Signal Processing*, vol. 58, no. 7, pp. 3765–3776, 2010.

[15] C. Knapp and G. Carter, "The generalized correlation method for estimation of time delay," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 24, no. 4, pp. 320–327, Aug. 1976.

[16] M. Omologo, P. Svaizer, A. Brutti, and L. Cristoforetti, "Speaker localization in CHIL lectures: Evaluation criteria and results," *Lecture Notes in Computer Science: Machine Learning for Multimodal Interaction*, vol. 3869, pp. 476–487, 2006.