



and the current measurement  $\mathbf{z}_n$  is conditionally independent of all previous measurements  $\mathbf{z}_{1:n-1}$  given the current state  $\mathbf{x}_n$ , i.e.,

$$f(\mathbf{z}_n|\mathbf{x}_n, \mathbf{z}_{1:n-1}) = f(\mathbf{z}_n|\mathbf{x}_n). \quad (4)$$

### 3. TIME-SPACE-SEQUENTIAL DISTRIBUTED BAYESIAN STATE ESTIMATION

Estimating the state  $\mathbf{x}_n$  from  $\mathbf{z}_{1:n}$ , i.e., all measurements up to the current time  $n$ , essentially amounts to calculating the *posterior pdf*  $f(\mathbf{x}_n|\mathbf{z}_{1:n})$ . Indeed, from  $f(\mathbf{x}_n|\mathbf{z}_{1:n})$ , optimal Bayesian estimators like the minimum mean-square error (MMSE) and maximum a posteriori (MAP) estimators, as well as quantities characterizing estimation performance, can be derived [9]. We will now develop a time-space sequential, distributed algorithm for calculating  $f(\mathbf{x}_n|\mathbf{z}_{1:n})$ .

#### 3.1. Time recursion

Using  $\mathbf{z}_{1:n} = (\mathbf{z}_{1:n-1}^T \mathbf{z}_n^T)^T$ , Bayes' rule, and (3) and (4), it is shown e.g. in [10] that the current posterior  $f(\mathbf{x}_n|\mathbf{z}_{1:n})$  can be calculated time-sequentially from the previous posterior  $f(\mathbf{x}_{n-1}|\mathbf{z}_{1:n-1})$  and the current measurement  $\mathbf{z}_n$ . This recursion consists of two steps. In the *prediction* step, the ‘‘predicted posterior’’  $f(\mathbf{x}_n|\mathbf{z}_{1:n-1})$  is calculated from the previous posterior  $f(\mathbf{x}_{n-1}|\mathbf{z}_{1:n-1})$  according to

$$f(\mathbf{x}_n|\mathbf{z}_{1:n-1}) = \int_{\mathbf{x}_{n-1}} f(\mathbf{x}_n|\mathbf{x}_{n-1}) f(\mathbf{x}_{n-1}|\mathbf{z}_{1:n-1}) d\mathbf{x}_{n-1}. \quad (5)$$

In the *update* step, the current measurement  $\mathbf{z}_n$  is used to convert  $f(\mathbf{x}_n|\mathbf{z}_{1:n-1})$  into the current posterior  $f(\mathbf{x}_n|\mathbf{z}_{1:n})$  according to

$$f(\mathbf{x}_n|\mathbf{z}_{1:n}) = \frac{f(\mathbf{z}_n|\mathbf{x}_n)f(\mathbf{x}_n|\mathbf{z}_{1:n-1})}{f(\mathbf{z}_n|\mathbf{z}_{1:n-1})}, \quad (6)$$

with

$$f(\mathbf{z}_n|\mathbf{z}_{1:n-1}) = \int_{\mathbf{x}_n} f(\mathbf{z}_n|\mathbf{x}_n) f(\mathbf{x}_n|\mathbf{z}_{1:n-1}) d\mathbf{x}_n.$$

#### 3.2. Space recursion

The time update (6) requires centralized processing of the total (all sensors) measurement vector  $\mathbf{z}_n$ . However, we will now show that (6) can be reformulated in a *space-sequential, distributed* manner in which sensor  $k$  receives a ‘‘partial posterior’’ from sensor  $k-1$ , updates it using the *local* measurement  $\mathbf{z}_n^{(k)}$ , and transmits the updated partial posterior to sensor  $k+1$ . This space-sequential update is analogous to that presented in [6], with the difference that [6] does not consider a temporal state transition.

Let  $\mathbf{z}_n^{(1:k)} \triangleq (\mathbf{z}_n^{(1)T} \dots \mathbf{z}_n^{(k)T})^T$  denote the current measurements of all sensors up to sensor  $k$ , where  $\mathbf{z}_n^{(1:K_n)} = \mathbf{z}_n$  and  $\mathbf{z}_n^{(1:0)}$  is the empty vector. Because of (2) and our statistical assumptions,  $\mathbf{z}_n^{(k)}$  is conditionally independent of the current measurements of all spatially previous sensors  $\mathbf{z}_n^{(1:k-1)}$  and all (temporally) previous measurements  $\mathbf{z}_{1:n-1}$  given the current state  $\mathbf{x}_n$ , i.e.,

$$f(\mathbf{z}_n^{(k)}|\mathbf{x}_n, \mathbf{z}_n^{(1:k-1)}, \mathbf{z}_{1:n-1}) = f(\mathbf{z}_n^{(k)}|\mathbf{x}_n). \quad (7)$$

Note that this is a refinement of (4). Using (7), we are able to apply the recursion of [6], which we will here rederive in our setting. Suppose that at time  $n$ , sensor  $k$  receives from sensor  $k-1$  the ‘‘partial posterior’’  $f(\mathbf{x}_n|\mathbf{z}_n^{(1:k-1)}, \mathbf{z}_{1:n-1})$  that incorporates all previous measurements  $\mathbf{z}_{1:n-1}$  and all current measurements up to sensor  $k-1$ ,  $\mathbf{z}_n^{(1:k-1)}$ . Sensor  $k$  now has to calculate the updated partial posterior  $f(\mathbf{x}_n|\mathbf{z}_n^{(1:k)}, \mathbf{z}_{1:n-1})$  that incorporates also the local measurement  $\mathbf{z}_n^{(k)}$ . Using  $\mathbf{z}_n^{(1:k)} = (\mathbf{z}_n^{(1:k-1)T} \mathbf{z}_n^{(k)T})^T$  and Bayes' rule (with background knowledge  $\mathbf{z}_n^{(1:k-1)}, \mathbf{z}_{1:n-1}$ ), we have

$$\begin{aligned} f(\mathbf{x}_n|\mathbf{z}_n^{(1:k)}, \mathbf{z}_{1:n-1}) &= f(\mathbf{x}_n|\mathbf{z}_n^{(k)}, \mathbf{z}_n^{(1:k-1)}, \mathbf{z}_{1:n-1}) \\ &= \frac{f(\mathbf{z}_n^{(k)}|\mathbf{x}_n, \mathbf{z}_n^{(1:k-1)}, \mathbf{z}_{1:n-1}) f(\mathbf{x}_n|\mathbf{z}_n^{(1:k-1)}, \mathbf{z}_{1:n-1})}{f(\mathbf{z}_n^{(k)}|\mathbf{z}_n^{(1:k-1)}, \mathbf{z}_{1:n-1})} \\ &= \frac{f(\mathbf{z}_n^{(k)}|\mathbf{x}_n) f(\mathbf{x}_n|\mathbf{z}_n^{(1:k-1)}, \mathbf{z}_{1:n-1})}{f(\mathbf{z}_n^{(k)}|\mathbf{z}_n^{(1:k-1)}, \mathbf{z}_{1:n-1})}, \end{aligned} \quad (8)$$

where (7) was used in the last step. The denominator is calculated as

$$\begin{aligned} f(\mathbf{z}_n^{(k)}|\mathbf{z}_n^{(1:k-1)}, \mathbf{z}_{1:n-1}) &= \int_{\mathbf{x}_n} f(\mathbf{z}_n^{(k)}|\mathbf{x}_n, \mathbf{z}_n^{(1:k-1)}, \mathbf{z}_{1:n-1}) f(\mathbf{x}_n|\mathbf{z}_n^{(1:k-1)}, \mathbf{z}_{1:n-1}) d\mathbf{x}_n \\ &= \int_{\mathbf{x}_n} f(\mathbf{z}_n^{(k)}|\mathbf{x}_n) f(\mathbf{x}_n|\mathbf{z}_n^{(1:k-1)}, \mathbf{z}_{1:n-1}) d\mathbf{x}_n. \end{aligned} \quad (9)$$

Equation (8) provides the desired space recursion that converts  $f(\mathbf{x}_n|\mathbf{z}_n^{(1:k-1)}, \mathbf{z}_{1:n-1})$  into  $f(\mathbf{x}_n|\mathbf{z}_n^{(1:k)}, \mathbf{z}_{1:n-1})$ , using the local measurement  $\mathbf{z}_n^{(k)}$ . At the last sensor ( $k = K_n$ ), we obtain the ‘‘total posterior’’  $f(\mathbf{x}_n|\mathbf{z}_{1:n})$  since  $f(\mathbf{x}_n|\mathbf{z}_n^{(1:K_n)}, \mathbf{z}_{1:n-1}) = f(\mathbf{x}_n|\mathbf{z}_n, \mathbf{z}_{1:n-1}) = f(\mathbf{x}_n|\mathbf{z}_{1:n})$ . From  $f(\mathbf{x}_n|\mathbf{z}_{1:n})$ , an estimate of  $\mathbf{x}_n$  and quantities characterizing its performance can be calculated.

The space recursion is initialized by  $f(\mathbf{x}_n|\mathbf{z}_n^{(1:k-1)}, \mathbf{z}_{1:n-1})|_{k=1} = f(\mathbf{x}_n|\mathbf{z}_n^{(1:0)}, \mathbf{z}_{1:n-1}) = f(\mathbf{x}_n|\mathbf{z}_{1:n-1})$ . This is the predicted posterior, which is calculated according to (5), using the known  $f(\mathbf{x}_n|\mathbf{x}_{n-1})$  and the previous posterior  $f(\mathbf{x}_{n-1}|\mathbf{z}_{1:n-1})$ .

#### 3.3. Time-space sequential algorithm

The space-sequential formulation (8) of the time update (6) yields a *time-space-sequential algorithm* for calculating the sequence of posteriors  $f(\mathbf{x}_n|\mathbf{z}_{1:n})$ ,  $n = 1, 2, \dots$ . This algorithm is fully distributed and uses only communication between neighboring sensors. Its operation at time  $n$  can be summarized as follows (cf. Fig. 1).

**1) Preparation.** The last sensor ( $k = K_{n-1}$ ) of the previous time recursion (time  $n-1$ ) transmits the previously calculated posterior  $f(\mathbf{x}_{n-1}|\mathbf{z}_{1:n-1})$  to the current first sensor ( $k = 1$ ). (It is assumed that these two sensors are spatially close; in many cases, they will be identical, so that no transmission is required.) *Initialization:* This preparation step is not performed for  $n = 1$ .

**2) Prediction.** The first sensor ( $k = 1$ ) converts  $f(\mathbf{x}_{n-1}|\mathbf{z}_{1:n-1})$  into the predicted posterior  $f(\mathbf{x}_n|\mathbf{z}_{1:n-1})$  according to (5). Note that this prediction step does not require the current measurement  $\mathbf{z}_n$ . *Initialization:* For  $n = 1$ ,  $\mathbf{z}_{1:n-1}|_{n=1} = \mathbf{z}_{1:0}$  is empty, and thus (5) degenerates to the following relation involving the initial statistics  $f(\mathbf{x}_0)$ :

$$f(\mathbf{x}_1) = \int_{\mathbf{x}_0} f(\mathbf{x}_1|\mathbf{x}_0) f(\mathbf{x}_0) d\mathbf{x}_0. \quad (10)$$

**3) Space updates.** The sensors  $k = 1, \dots, K_n$  sequentially carry out the space update (8), using the local measurements  $\mathbf{z}_n^{(k)}$  and communication between neighboring sensors as described in Section 3.2. At the last sensor ( $k = K_n$ ), the total posterior  $f(\mathbf{x}_n|\mathbf{z}_{1:n})$  is obtained. *Initialization:* The space updates are initialized at the first sensor ( $k = 1$ ) by the predicted posterior  $f(\mathbf{x}_n|\mathbf{z}_{1:n-1})$ , which was calculated by the first sensor in Step 2. For  $n = 1$ , the space update (8) takes on the following degenerate form:

$$f(\mathbf{x}_1|\mathbf{z}_1^{(1:k)}) = \frac{f(\mathbf{z}_1^{(k)}|\mathbf{x}_1) f(\mathbf{x}_1|\mathbf{z}_1^{(1:k-1)})}{f(\mathbf{z}_1^{(k)}|\mathbf{z}_1^{(1:k-1)})},$$

with (see (9))

$$f(\mathbf{z}_1^{(k)}|\mathbf{z}_1^{(1:k-1)}) = \int_{\mathbf{x}_1} f(\mathbf{z}_1^{(k)}|\mathbf{x}_1) f(\mathbf{x}_1|\mathbf{z}_1^{(1:k-1)}) d\mathbf{x}_1,$$

for  $k = 1, \dots, K_1$ . This is initialized by  $f(\mathbf{x}_1|\mathbf{z}_1^{(1:k-1)})|_{k=1} = f(\mathbf{x}_1|\mathbf{z}_1^{(1:0)}) = f(\mathbf{x}_1)$ , which was calculated in Step 2 (see (10)).

#### 4. TIME-SPACE-SEQUENTIAL DISTRIBUTED KALMAN FILTER

Next, we study the special case of a linear/Gaussian state-space model. Consider a linear state-transition equation (1)

$$\mathbf{x}_n = \mathbf{g}_n(\mathbf{x}_{n-1}, \mathbf{u}_n) = \mathbf{A}_n \mathbf{x}_{n-1} + \mathbf{u}_n$$

and linear measurement equations (2)

$$\mathbf{z}_n^{(k)} = \mathbf{h}_n^{(k)}(\mathbf{x}_n, \mathbf{w}_n^{(k)}) = \mathbf{H}_n^{(k)} \mathbf{x}_n + \mathbf{w}_n^{(k)}, \quad k = 1, 2, \dots, K_n \quad (11)$$

with Gaussian driving and measurement noises  $\mathbf{u}_n \sim \mathcal{N}(\mathbf{0}, \mathbf{C}_{\mathbf{u}_n})$ ,  $\mathbf{w}_n^{(k)} \sim \mathcal{N}(\mathbf{0}, \mathbf{C}_{\mathbf{w}_n^{(k)}})$  and Gaussian initial state  $\mathbf{x}_0 \sim \mathcal{N}(\boldsymbol{\mu}_{\mathbf{x}_0}, \mathbf{C}_{\mathbf{x}_0})$ . As before, all random vectors  $\mathbf{u}_n, \mathbf{w}_n^{(k)}, \mathbf{x}_0$  are independent. The model parameters  $\mathbf{A}_n, \mathbf{H}_n^{(k)}, \mathbf{C}_{\mathbf{u}_n}, \mathbf{C}_{\mathbf{w}_n^{(k)}}, \boldsymbol{\mu}_{\mathbf{x}_0}$ , and  $\mathbf{C}_{\mathbf{x}_0}$  are known. The measurement equations (11) can be combined as  $\mathbf{z}_n = \mathbf{H}_n \mathbf{x}_n + \mathbf{w}_n$ , with  $\mathbf{H}_n \triangleq (\mathbf{H}_n^{(1)T} \dots \mathbf{H}_n^{(K_n)T})^T$  and the block-diagonal covariance matrix  $\mathbf{C}_{\mathbf{w}_n} \triangleq \text{diag}\{\mathbf{C}_{\mathbf{w}_n^{(1)}}, \dots, \mathbf{C}_{\mathbf{w}_n^{(K_n)}}\}$ .

The pdf's  $f(\mathbf{x}_n|\mathbf{z}_{1:n})$ ,  $f(\mathbf{x}_n|\mathbf{z}_{1:n-1})$ ,  $f(\mathbf{z}_n|\mathbf{x}_n)$ , and  $f(\mathbf{x}_n|\mathbf{x}_{n-1})$  now are Gaussian and hence fully described by their means and covariances. In particular, let  $\hat{\mathbf{x}}_{n|n} \triangleq \text{E}\{\mathbf{x}_n|\mathbf{z}_{1:n}\}$  (the MMSE estimate of  $\mathbf{x}_n$  from  $\mathbf{z}_{1:n}$ ),  $\mathbf{C}_{n|n} \triangleq \text{cov}\{\mathbf{x}_n|\mathbf{z}_{1:n}\}$ ,  $\hat{\mathbf{x}}_{n|n-1} \triangleq \text{E}\{\mathbf{x}_n|\mathbf{z}_{1:n-1}\}$ , and  $\mathbf{C}_{n|n-1} \triangleq \text{cov}\{\mathbf{x}_n|\mathbf{z}_{1:n-1}\}$ . Then, it is well known [11] that the time recursion (5), (6) reduces to the Kalman filter (KF) recursion, which consists of the *prediction* step (cf. (5))

$$\hat{\mathbf{x}}_{n|n-1} = \mathbf{A}_n \hat{\mathbf{x}}_{n-1|n-1} \quad (12)$$

$$\mathbf{C}_{n|n-1} = \mathbf{A}_n \mathbf{C}_{n-1|n-1} \mathbf{A}_n^T + \mathbf{C}_{\mathbf{u}_n} \quad (13)$$

and the *update* step (cf. (6))

$$\hat{\mathbf{x}}_n = \hat{\mathbf{x}}_{n|n-1} + \mathbf{K}_n (\mathbf{z}_n - \mathbf{H}_n \hat{\mathbf{x}}_{n|n-1}) \quad (14)$$

$$\mathbf{C}_n = \mathbf{C}_{n|n-1} - \mathbf{K}_n \mathbf{H}_n \mathbf{C}_{n|n-1}, \quad (15)$$

with  $\mathbf{K}_n \triangleq \mathbf{C}_{n|n-1} \mathbf{H}_n^T (\mathbf{H}_n \mathbf{C}_{n|n-1} \mathbf{H}_n^T + \mathbf{C}_{\mathbf{w}_n})^{-1}$ . This recursion is initialized by  $\hat{\mathbf{x}}_{n-1|n-1}|_{n=1} = \hat{\mathbf{x}}_{0|0} = \boldsymbol{\mu}_{\mathbf{x}_0}$  and  $\mathbf{C}_{n-1|n-1}|_{n=1} = \mathbf{C}_{0|0} = \mathbf{C}_{\mathbf{x}_0}$ .

We will now demonstrate that the time-update relations (14) and (15) can be calculated in a space-sequential, distributed manner using another KF-type recursion. This recursion specializes the general space recursion (8) to the linear/Gaussian case. Instead of the partial posteriors  $f(\mathbf{x}_n|\mathbf{z}_n^{(1:k)}, \mathbf{z}_{1:n-1})$ , partial means and covariances are now transmitted from one sensor to the next.

Indeed, the partial posterior at sensor  $k$ ,  $f(\mathbf{x}_n|\mathbf{z}_n^{(1:k)}, \mathbf{z}_{1:n-1})$ , is Gaussian and hence described by the partial mean  $\hat{\mathbf{x}}_{n|n-1}^{(1:k)} \triangleq \text{E}\{\mathbf{x}_n|\mathbf{z}_n^{(1:k)}, \mathbf{z}_{1:n-1}\}$  (the MMSE estimate of  $\mathbf{x}_n$  from  $\mathbf{z}_n^{(1:k)}, \mathbf{z}_{1:n-1}$ ) and the partial covariance  $\mathbf{C}_{n|n-1}^{(1:k)} \triangleq \text{cov}\{\mathbf{x}_n|\mathbf{z}_n^{(1:k)}, \mathbf{z}_{1:n-1}\}$ . The space recursion (8) can then be shown to correspond to the mean/covariance update relations [12]

$$\hat{\mathbf{x}}_{n|n-1}^{(1:k)} = \hat{\mathbf{x}}_{n|n-1}^{(1:k-1)} + \mathbf{K}_n^{(k)} (\mathbf{z}_n^{(k)} - \mathbf{H}_n^{(k)} \hat{\mathbf{x}}_{n|n-1}^{(1:k-1)}) \quad (16)$$

$$\mathbf{C}_{n|n-1}^{(1:k)} = \mathbf{C}_{n|n-1}^{(1:k-1)} - \mathbf{K}_n^{(k)} \mathbf{H}_n^{(k)} \mathbf{C}_{n|n-1}^{(1:k-1)}, \quad (17)$$

with  $\mathbf{K}_n^{(k)} \triangleq \mathbf{C}_{n|n-1}^{(1:k-1)} \mathbf{H}_n^{(k)T} (\mathbf{H}_n^{(k)} \mathbf{C}_{n|n-1}^{(1:k-1)} \mathbf{H}_n^{(k)T} + \mathbf{C}_{\mathbf{w}_n^{(k)}})^{-1}$ .

At the last sensor ( $k = K_n$ ), this recursion yields  $\hat{\mathbf{x}}_{n|n-1}^{(1:K_n)} = \text{E}\{\mathbf{x}_n|\mathbf{z}_n^{(1:K_n)}, \mathbf{z}_{1:n-1}\} = \text{E}\{\mathbf{x}_n|\mathbf{z}_{1:n}\} = \hat{\mathbf{x}}_{n|n}$  and  $\mathbf{C}_{n|n-1}^{(1:K_n)} = \text{cov}\{\mathbf{x}_n|\mathbf{z}_n^{(1:K_n)}, \mathbf{z}_{1:n-1}\} = \text{cov}\{\mathbf{x}_n|\mathbf{z}_{1:n}\} = \mathbf{C}_{n|n}$ , i.e., the result of (14), (15) as desired. The recursion is initialized by  $\hat{\mathbf{x}}_{n|n-1}^{(1:k-1)}|_{k=1} = \hat{\mathbf{x}}_{n|n-1}^{(1:0)} = \text{E}\{\mathbf{x}_n|\mathbf{z}_n^{(1:0)}, \mathbf{z}_{1:n-1}\} = \text{E}\{\mathbf{x}_n|\mathbf{z}_{1:n-1}\} = \hat{\mathbf{x}}_{n|n-1}$  and  $\mathbf{C}_{n|n-1}^{(1:k-1)}|_{k=1} = \mathbf{C}_{n|n-1}^{(1:0)} = \text{cov}\{\mathbf{x}_n|\mathbf{z}_n^{(1:0)}, \mathbf{z}_{1:n-1}\} = \text{cov}\{\mathbf{x}_n|\mathbf{z}_{1:n-1}\} = \mathbf{C}_{n|n-1}$ , which are provided by the prediction step (12), (13). This type of sequential processing was previously derived (however, not in the context of distributed sensor processing) in [12]; it is enabled by the block-diagonal structure of  $\mathbf{C}_{\mathbf{w}_n}$ .

The space-sequential formulation (16), (17) of the time update (14), (15) yields a *time-space-sequential*, distributed form of the KF that uses only communication between neighboring sensors. The operation of this algorithm at time  $n$  can be summarized as follows.

**1) Preparation.** The last sensor of the previous time recursion (time  $n-1$ ) transmits  $\hat{\mathbf{x}}_{n-1|n-1}$  and  $\mathbf{C}_{n-1|n-1}$  to the current first sensor ( $k=1$ ). *Initialization:* This step is not performed for  $n=1$ .

**2) Prediction.** The first sensor ( $k=1$ ) converts  $\hat{\mathbf{x}}_{n-1|n-1}$  and  $\mathbf{C}_{n-1|n-1}$  into the predicted mean  $\hat{\mathbf{x}}_{n|n-1}$  and predicted covariance  $\mathbf{C}_{n|n-1}$  according to (12), (13). *Initialization:* For  $n=1$ , the prediction step (12), (13) is initialized by the initial statistics, i.e.,  $\hat{\mathbf{x}}_{n-1|n-1}|_{n=1} = \hat{\mathbf{x}}_{0|0} = \boldsymbol{\mu}_{\mathbf{x}_0}$  and  $\mathbf{C}_{n-1|n-1}|_{n=1} = \mathbf{C}_{0|0} = \mathbf{C}_{\mathbf{x}_0}$ .

**3) Space updates.** Each sensor  $k$  receives from sensor  $k-1$  the partial mean and covariance  $\hat{\mathbf{x}}_{n|n-1}^{(1:k-1)}, \mathbf{C}_{n|n-1}^{(1:k-1)}$ ; uses (16), (17) to calculate the updated partial mean and covariance  $\hat{\mathbf{x}}_{n|n-1}^{(1:k)}, \mathbf{C}_{n|n-1}^{(1:k)}$  incorporating the local measurement  $\mathbf{z}_n^{(k)}$ ; and transmits the updated quantities to sensor  $k+1$ . At the last sensor ( $k=K_n$ ), the total mean (MMSE estimate)  $\hat{\mathbf{x}}_{n|n}$  and total covariance  $\mathbf{C}_{n|n}$  are obtained. *Initialization:* These space updates are initialized at the first sensor ( $k=1$ ) by the predicted mean  $\hat{\mathbf{x}}_{n|n-1}$  and predicted covariance  $\mathbf{C}_{n|n-1}$ , which were calculated by the first sensor in Step 2.

#### 5. APPLICATION TO TARGET TRACKING

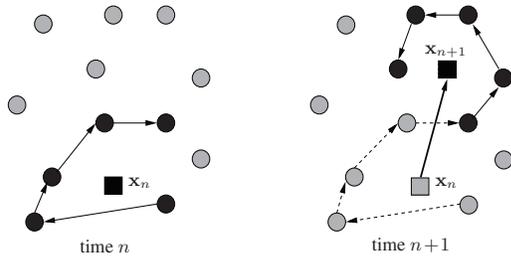
We now apply the distributed estimation algorithm of Section 3 to the tracking of a target moving in a plane. The target's 2-D Cartesian coordinate vector  $\mathbf{x}_n$  is to be estimated from the sound emitted by the target. We assume that the target's speed and heading are uniformly distributed in  $[0, v_{\max}]$  and in  $[0, 2\pi)$ , respectively. It follows that the state-transition pdf  $f(\mathbf{x}_n|\mathbf{x}_{n-1})$  is uniform inside a disk of radius  $T v_{\max}$  centered at  $\mathbf{x}_{n-1}$  (here,  $T$  is the sampling period, i.e., the time duration of the discrete-time interval  $[n-1, n]$ ) [7].

The target is sensed by acoustic amplitude sensors [1] deployed in the plane. At time  $n$ ,  $K_n$  sensors that are close to the current target position  $\mathbf{x}_n$  form a "local sensor chain" (LSC) emulating the serial sensor network of Fig. 1. In Fig. 2, we depict the LSCs at two consecutive times. Let  $\boldsymbol{\xi}_n^{(k)}$  denote the position of the  $k$ th sensor of the LSC at time  $n$ . The measurement  $z_n^{(k)}$  of the  $k$ th sensor depends on the distance from the target (sound source),  $r_n^{(k)} \triangleq \|\mathbf{x}_n - \boldsymbol{\xi}_n^{(k)}\|$ , via the nonlinear measurement equation [7]

$$z_n^{(k)} = \frac{a}{r_n^{(k)}} + w_n^{(k)}. \quad (18)$$

Here,  $a$  (the amplitude of the sound source) is modeled as uniformly distributed in  $[a_1, a_2]$ , and  $w_n^{(k)} \sim \mathcal{N}(0, \sigma^2)$  is white Gaussian noise. For this measurement model, the likelihood function is given by [8]

$$f(z_n^{(k)}|\mathbf{x}_n) = \frac{r_n^{(k)}}{\Delta a} \left[ \Phi\left(\frac{a_2 - r_n^{(k)} z_n^{(k)}}{r_n^{(k)} \sigma}\right) - \Phi\left(\frac{a_1 - r_n^{(k)} z_n^{(k)}}{r_n^{(k)} \sigma}\right) \right],$$



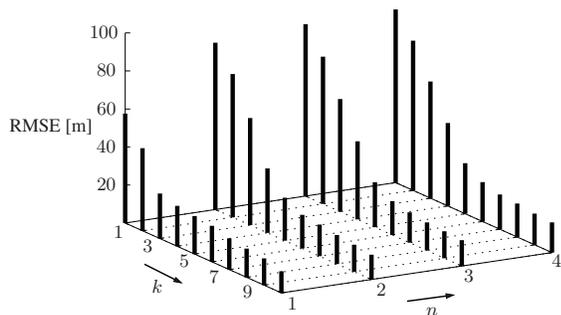
**Fig. 2.** Target position (black square) and LSC sensors (black disks) at two consecutive time instants.

with  $\Delta a \triangleq a_2 - a_1$  and  $\Phi(y) \triangleq \frac{1}{2\pi} \int_{-\infty}^y e^{-\eta^2/2} d\eta$ .

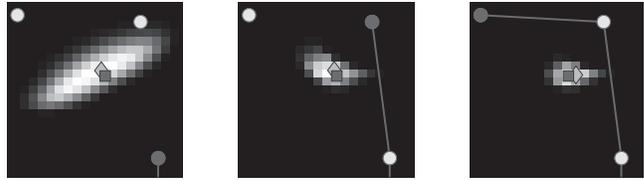
Within the LSC, the space-sequential update of the posterior proceeds as described in Section 3. The total posterior is available at the last ( $K_n$ th) sensor, which is also the first sensor of the LSC at the next time instant  $n+1$  (this new LSC is located around target position  $\mathbf{x}_{n+1}$ , see Fig. 2). This first sensor performs the prediction step, and the predicted posterior initializes the space-sequential update procedure performed by the new LSC.

At any given time, the current LSC is formed dynamically using the following decentralized scheduling algorithm. The sensor currently active selects as the next sensor of the LSC the neighboring sensor with minimum distance to (the local MMSE estimate of) the target position. (The local MMSE estimate of the target position is the mean of the local partial posterior. We assume that each sensor knows the positions of its neighbors.) From (18), it then follows that sensors with a higher signal-to-noise ratio are more likely to be included in the LSC. While a sensor may appear only once in a given LSC, it may be reused in a subsequent LSC.

We simulated this target tracking method with 64 sensors randomly deployed in a rectangular region of size  $1000\text{m} \times 1000\text{m}$ , using target parameters  $v_{\max} = 180\text{m}/T$ ,  $a_1 = 90$ , and  $a_2 = 130$ , noise variance  $\sigma^2 = 0.0042$ , and constant LSC size  $K_n \equiv 10$ . The posteriors and likelihoods were discretized using a 2-D sampling grid of  $100 \times 100$  samples. (This simple discretization results in high-rate communications and has to be improved in practical applications—see, e.g., [7].) In Fig. 3, we show the empirical root mean-square error (RMSE)  $\sqrt{\mathbb{E}\{\|\hat{\mathbf{x}}_{n|n-1}^{(1:k)} - \mathbf{x}_n\|^2\}}$  of the local MMSE state estimate  $\hat{\mathbf{x}}_{n|n-1}^{(1:k)}$  (i.e., of the mean of the partial posterior  $f(\mathbf{x}_n | \mathbf{z}_n^{(1:k)}, \mathbf{z}_{1:n-1})$ ) as a function of time  $n$  and sensor index  $k$ . The RMSE was estimated by averaging  $\|\hat{\mathbf{x}}_{n|n-1}^{(1:k)} - \mathbf{x}_n\|^2$  over 1000 simulation runs. It is seen that at fixed time  $n$ , the RMSE decreases with the number  $k$  of sensor measurements included in the state estimate. However, when passing on to the next time, the RMSE again increases because the state has evolved to a new value. For



**Fig. 3.** Empirical RMSE of the local MMSE state estimate versus time  $n$  and sensor index  $k$ .



**Fig. 4.** Discretized partial posterior  $f(\mathbf{x}_n | \mathbf{z}_n^{(1:k)}, \mathbf{z}_{1:n-1})$  for three consecutive sensors  $k$  (disks) of an LSC along with the estimated target position (diamond) and the true target position (square). Lighter shading indicates a larger posterior.

one realization, we show in Fig. 4 the discretized partial posterior  $f(\mathbf{x}_n | \mathbf{z}_n^{(1:k)}, \mathbf{z}_{1:n-1})$  and estimated target position for three consecutive sensor indices  $k$  of an LSC. It is seen that the partial posterior becomes narrower for increasing  $k$ . (This behavior is not observed for all realizations, even though it is true on average, cf. Fig. 3).

## 6. CONCLUSION

We presented a distributed algorithm for nonlinear/non-Gaussian state estimation for the case where the current state is sensed by a serial sensor network. Our algorithm is sequential both temporally and across sensors, and it requires only local communication between neighboring sensors. In the linear/Gaussian case, it reduces to a time-space-sequential, distributed form of the Kalman filter. As a simple application, we considered target tracking by means of acoustic amplitude sensors, using a decentralized, dynamic choice of a “local sensor chain” around the current target position.

## 7. REFERENCES

- [1] F. Zhao and L. J. Guibas, *Wireless Sensor Networks: An Information Processing Approach*. Amsterdam, The Netherlands: Morgan Kaufmann, 2004.
- [2] T. Vercauteren and X. Wang, “Decentralized sigma-point information filters for target tracking in collaborative sensor networks,” *IEEE Trans. Signal Process.*, vol. 53, pp. 2997–3009, Aug. 2005.
- [3] A. Ribeiro, G. B. Giannakis, and S. I. Roumeliotis, “SOI-KF: Distributed Kalman filtering with low-cost communications using the sign of innovations,” *IEEE Trans. Signal Process.*, vol. 54, pp. 4782–4795, Dec. 2006.
- [4] R. G. Baraniuk, V. Delouille, and R. Neelamani, “Robust distributed estimation using the embedded subgraphs algorithm,” *IEEE Trans. Signal Process.*, vol. 54, pp. 2998–3010, Aug. 2006.
- [5] T. Zhao and A. Nehorai, “Information-driven distributed maximum likelihood estimation based on Gauss-Newton method in wireless sensor networks,” *IEEE Trans. Signal Process.*, vol. 55, pp. 4669–4682, Sep. 2007.
- [6] T. Zhao and A. Nehorai, “Distributed sequential Bayesian estimation of a diffusive source in wireless sensor networks,” *IEEE Trans. Signal Process.*, vol. 55, pp. 1511–1524, Apr. 2007.
- [7] J. Liu, J. Reich, and F. Zhao, “Collaborative in-network processing for target tracking,” *EURASIP J. Appl. Sig. Process.*, vol. 2003, pp. 378–391, Jan. 2003.
- [8] M. Chu, H. Haussecker, and F. Zhao, “Scalable information-driven sensor querying and routing for ad hoc heterogeneous sensor networks,” *Int. J. High Perf. Comp. Appl.*, vol. 16, pp. 293–314, Aug. 2002.
- [9] H. V. Poor, *An Introduction to Signal Detection and Estimation*. New York, NY: Springer, 1994.
- [10] B. Ristic and S. Arulampalam, *Beyond the Kalman Filter: Particle Filters for Tracking Applications*. Boston, MA: Artech House, 2004.
- [11] Y. Ho and R. Lee, “A Bayesian approach to problems in stochastic estimation and control,” *IEEE Trans. Autom. Contr.*, vol. 9, pp. 333–339, Oct. 1964.
- [12] B. D. O. Anderson and J. B. Moore, *Optimal Filtering*. Englewood Cliffs, NJ: Prentice Hall, 1979.