Self-Synchronization Based Distributed Localization of Wireless Transmitters

Evert I. Pocoma Copa*, François Quitin*, Luc Vandendorpe †,
Philippe De Doncker*, François Horlin*

*Université Libre de Bruxelles (ULB)
Av. Rooselvelt 50, 1050 Brussels, Belgium
Emails: {epocomac,fquitin,philippe.dedoncker,francois.horlin,}@ulb.be

†Université Catholique de Louvain (UCLouvain)
Place de l'Université 1, 1348 Ottignies-Louvain-la-Neuve, Belgium
Emails: {luc.vandendorpe}@uclouvain.be

***Abstract*—In this paper, we introduce a fully distributed localization algorithm based on self-synchronization mechanism. The proposed algorithm reaches consensus for the posterior distribution of the transmitter position at each base station. To reduce the communication overhead at each iteration, we propose to represent the state variable matrices of the self-synchronization mechanism with only four parameters (radial and angular means and variances). The performance of the algorithms is numerically assessed by the mean distance error and mean Kullback–Leibler divergence. Finally, we show through Monte-Carlo simulations that our approach gets very close to the direct-centralized-localization performance after a few iterations.**

***Index Terms*—Distributed localization, decentralized localization, self-synchronization, average consensus, iterative positioning.**

I. Introduction

Cellular networks have evolved towards increasingly accurate geo-location services. Positioning reference signals (PRS) are included in the protocols to support device localization based on the estimation of the signal time-of-arrival (ToA) [1]. The direct position estimation (DPE) algorithm estimates the transmitter position using grid search, i.e., it compares the signals received by each base station (BS) to the signals that would theoretically be received if the transmitter was at a particular position [2], [3]. Therefore, all the baseband signals are communicated to a processing node known as fusion centre (FC). To reduce the amount of communicated data a two-step approach can be used. It consists in a range estimation step at the BSs followed by a multi-lateration step at the FC [4]. Nonetheless, some information is lost in the first step, since only local information is available at the BS. Moreover, it has been analytically demonstrated in [5] that the DPE algorithm always outperforms any two-step approach.

Different approaches have been studied to achieve distributed localization [6]–[9]. On the one hand, a semi-distributed version of DPE algorithm is proposed in [9]. Nonetheless, such approach is still centralized in the local neighborhood, i.e., the received baseband signals are shared between direct neighbors and the information among non-direct neighbors is approximated using interpolation. On the other hand, in [6], [7] and [8] the distributed localization is approached as an optimization problem, which is solved by different methods such as: alternating direction method of multipliers (ADMM), primal-dual method of multipliers (PDMM) or second-order cone programming (SOCP) combined with standard solvers. Nevertheless, in these studies the non-convex optimization problem is solved after the range estimation step; hence its performance is bounded by the two-step localization approach. For such a reason, several algorithms were proposed to improve the performance after the multi-lateration step [10]–[12]. The iterative position estimation (IPE) algorithm proposed in [10] iterates over the two steps with the goal of refining the range estimation in the first step. Nevertheless, IPE is still a centralized algorithm.

Average consensus can be seen as a particular case of the self-synchronization mechanism introduced in [13], which computes the average of a parameter in a distributed fashion. It has already been used, in the context of distributed localization, to improve the accuracy in a network [11], [12]. For instance, in [11] the final estimations are averaged among all BSs using the average consensus. In [12] the localization is done by means of an iterative process such as the Gauss-Newton algorithm, where the self-synchronization mechanism is used to average the intermediate estimates at each iteration.

Self-synchronization can be used to compute an optimal decision in a distributed manner [13]. The convergence is guaranteed for very relaxed conditions. In fact, the only requirement for convergence is to have a strongly connected network. The convergence is guaranteed even for noisy and/or unreliable links, requiring a strongly connected network only on average over time [14], [15]. The self-synchronization mechanism relies on the exchange of state variables among the sensor nodes. Such state variables need to be compressed so that the number of communicated parameters is reduced.

Coupling noise is introduced to the system, when such a compression involves approximation errors; thus, having an impact on the speed and convergence values, i.e., the self-synchronization mechanism becomes biased. On the one hand, zero order consensus converges directly on the state variables. On the other hand, first order consensus converges on the first order derivative of the state variables. The main advantage of the first order over the zero order consensus is that it has a better rejection to coupling noise [14].

The contributions presented in this work are threefold. First, we introduce a fully distributed localization algorithm based on self-synchronization mechanism. Our approach works at the received signal level, i.e., we average the received signal log-likelihoods. Such an approach reaches consensus for the posterior distribution of the transmitter position rather than directly on the final position estimates. It also means that we are able to achieve the performance of the DPE algorithm in a distributed fashion. Second, we propose to compress the state variables that will be exchanged among BSs, with the first two radial and angular moments. Therefore, only four parameters are exchanged between neighboring BSs. Such a compression leads to a degradation of performance that is reduced through out extra iterations. Lastly, we assess the performance of the proposed algorithm by means of a Monte-Carlo simulation. We assess the performance in terms of mean distance error (MDE) and the average of Kullback–Leibler (KL) divergence.

The paper is organized as follows. Section II introduces the signal model. Section III describes the centralized DPE algorithm and the proposed distributed self-synchronization position estimation (SSPE) algorithm. Afterwards, it describes the proposed compression and reconstruction for the state variable matrices and its effects. Section IV assesses numerically the performance of our algorithm. Finally, Section V concludes the paper.

II. System Model

We consider a cellular network operating with the orthogonal frequency-division multiplexing (OFDM) modulation. We assume a static transmitter that is simultaneously connected to N time-synchronized BSs in its neighbourhood. The OFDM modulation splits the communication bandwidth into orthogonal sub-carriers that are allocated to data or pilot symbols. A cyclic prefix (CP) is added to each block of transmitted symbols to maintain the orthogonality between the sub-carriers even in the presence of channel time dispersion.

We consider that P equispaced pilot sub-carriers are allocated over the communication bandwidth, i.e., the frequency difference between two consecutive pilot sub-carriers is constant and referred to as Δf. For simplicity, the channel is considered to be a single propagation delay τ_i between the transmitter position (x, y) and BS-i. The propagation delay is related to the distance δ_i between transmitter and BS-i by the expression: $\tau_i = \delta_i/c$, where c is the propagation velocity. If τ_i is shorter than the CP duration (which is a reasonable assumption for typical system parameters), the signal received on pilot-p at BS-i is

$$r_{i,p} = s_p e^{-j\frac{2\pi p \Delta f}{c}\delta_i} + w_{i,p}, \tag{1}$$

where $r_{i,p}$ and $w_{i,p}$ are the received signal and corrupting noise respectively, for pilot sub-carrier-p at BS-i, and s_p is the symbol at pilot sub-carrier-p. The noise is assumed to be independent zero mean circularly symmetric complex Gaussian of variance $\sigma^2_{w_i}$, which at the same time, is assumed to be known at BS-i. Finally, similarly to the system model adopted in [10], a vector model is constructed at each BS-i by stacking all the received pilot symbols as

$$\mathbf{r}_i = \mathbf{s}(\delta_i) + \mathbf{w}_i, \tag{2}$$

with

$$\mathbf{r}_i = [r_{i,1}, ..., r_{i,P}]^T \qquad ; \qquad \mathbf{w}_i = [w_{i,1}, ..., w_{i,P}]^T, \tag{3}$$

$$\mathbf{s}(\delta_i) = [s_1 e^{-j1\zeta\delta_i}, ..., s_P e^{-jP\zeta\delta_i}]^T, \tag{4}$$

where the constant ζ is used to simplify the notation and it is defined as $\zeta = \frac{2\pi\Delta f}{c}$.

III. Self-Synchronization Based Localization

A. Centralized Direct Position Estimation

The DPE algorithm estimates the transmitter position based on all the received signals. Hence, each BS-i should communicate its received baseband signal $\mathbf{r}_i$ to a central node denoted as FC. We assume that the transmitter is located inside a scene $\mathcal{S}$ delimited for $x \in [x_{min}, x_{max}]$ and $y \in [y_{min}, y_{max}]$. The posterior distribution of the transmitter position is given by (5), where C_p is a normalization factor ensuring that the integral of the posterior distribution in the scene $\mathcal{S}$ is 1. Lastly, $p(x, y)$ is the prior probabilistic density function (PDF) of the transmitter position.

$$p(x, y|\mathbf{r}_1, ...\mathbf{r}_N) = C_p \prod_{i=1}^{N} p_i(\mathbf{r}_i|x, y)p(x, y). \tag{5}$$

We assume that x and y are two mutually independent uniformly distributed random variables. Therefore, the prior PDF is expressed as $p(x, y){=}p(x)p(y)$, where $p(x)$ and $p(y)$ are uniformly distributed in the scene $\mathcal{S}$ and 0 elsewhere. Finally, based on (2), the likelihood of the received signal at BS-i can be modeled as a Gaussian PDF as in (6), where $C_\mathbf{r}$ is a normalization factor and $(.)^H$ represents the Hermitian transpose. Notice that the right hand side of (6) is dependent on the transmitter position (x, y) through the range δ_i, i.e. $\delta_i = \delta_i(x, y)$. Finally, we can compute the estimates of the transmitter position $(\hat{x}, \hat{y}$ as expected values with respect to the PDF defined in (6).

$$p_i(\mathbf{r}_i|x, y) = C_\mathbf{r} e^{-\frac{1}{\sigma^2_{w_i}}(\mathbf{r}_i - \mathbf{s}(\delta_i))^H(\mathbf{r}_i - \mathbf{s}(\delta_i))} \tag{6}$$

B. SSPE Algorithm

Taking the logarithm of (5) yields the log-posterior distribution written as

$$\log(p(x, y|\mathbf{r}_1, ...\mathbf{r}_N)) = \sum_{i=1}^{N} \mathcal{L}_i(\mathbf{r}_i|x, y) + b, \quad (7)$$

where $b{=}\log(p(x, y)){-}\log(C_p)$ is a constant that considers the normalization factor and the uniformly distributed prior. $\mathcal{L}_i(\mathbf{r}_i|x, y)$ is the log-likelihood of the received signal $\mathbf{r}_i$ at BS-i, which can be computed by taking the logarithm of (6). Equation (7) was derived from a centralized localization case, where all the signals are collected in one node. Hence, if (7) is approximated in a distributed manner, the performance of the approximation will get close to the centralized approach. We propose to use the self-synchronization to perform such an approximation described as follows.

We can express (7) in terms of the average of the received signal log-likelihoods as

$$\log(p(x, y|\mathbf{r}_1, ...\mathbf{r}_N)) = N\Big(\frac{1}{N}\sum_{i=1}^{N} \mathcal{L}_i(\mathbf{r}_i|x, y)\Big) + b. \quad (8)$$

By analyzing the right hand side of (8), two main remarks can be done. First, the constant term b can be omitted if the posterior distribution is normalized again after taking the exponential to compute the corresponding PDF. Second, the average of the received signal log-likelihoods can be computed in a fully distributed fashion with the help of the self-synchronization mechanism introduced in [13]. Following the notation used in [13], we consider that at BS-i, the measured parameter is equivalent to the received signal $\mathbf{r}_i$ and that the measurement function is the log-likelihood of the received signal $\mathcal{L}_i(\mathbf{r}_i|x, y)$. Therefore, each BS-$i$ have state variables $\mathbf{v}_i$ that will evolve as

$$\Delta\mathbf{v}_i[k] = \mathcal{L}_i(\mathbf{r}_i|x, y) + \beta\sum_{j=1}^{N} a_{i,j}(\mathbf{v}_j[k] - \mathbf{v}_i[k]), \quad (9)$$

where:

- $a_{i,j}$ is 1 if node-i and node-j are connected and 0 otherwise.
- $\mathbf{v}_j[k]$ is the state variable of node-j communicated to node-i at iteration-k. Notice that at iteration $k = 0$ the initial value is $\mathbf{v}_i[0] = \mathbf{0}$ for all nodes.
- β is a constant defined as the control loop gain.

It has been well studied in [13]–[16] that the dynamical system shown in (9) surely converges when the network is strongly connected, i.e., there exist at least one path that connects any node-i to any other node-j. Furthermore, it can be mathematically proven that the dynamical system given in (9) will asymptotically converge to the average of the log-likelihoods as in (10), if and only if, the network is strongly connected and balanced, i.e., $a_{i,j} = a_{j,i}$.

$$\Delta\mathbf{v}_i[k \to \infty] = \frac{1}{N}\sum_{j=1}^{N} \mathcal{L}_j(\mathbf{r}_j|x, y). \quad (10)$$

Algorithm 1 SSPE Algorithm

Each BS-i constructs $\mathcal{L}_i(\mathbf{r}_i|x, y)$ for each point in the search grid based on the received signal $\mathbf{r}_i$.
Each BS-i initializes $\mathbf{v}_i = 0$
for iteration $k = 1, 2, \ldots$ **do**
    for BS-$i = 1, 2, \ldots$ in parallel **do**
        Compute $\Delta\mathbf{v}_i[k]$ using (9)
        Estimate transmitter position using (12) and (13)
        Update $\mathbf{v}_i[k+1] = \mathbf{v}_i[k] + \Delta\mathbf{v}_i[k]$
        Communicate $\mathbf{v}_i[k+1]$ to neighboring BSs
        Receive $\mathbf{v}_j[k+1]$ from neighboring BSs
    end for
end for

Finally, based on (8) and (10), we can approximate the posterior distribution at each iteration-k for BS-i as

$$\hat{p}_i(x, y|\mathbf{r}_1, ...\mathbf{r}_N)[k] = C_{\Delta\mathbf{v}_i}\ e^{N\Delta\mathbf{v}_i[k]}, \quad (11)$$

where the constant $C_{\Delta\mathbf{v}_i}$ is a normalization factor ensuring that the integral of $\hat{p}_i(x, y|\mathbf{r}_1, ...\mathbf{r}_N)$ in the scene $\mathcal{S}$ is 1. Notice that the argument of the exponential, i.e., $N\Delta\mathbf{v}_i$ will asymptotically converge to the sum of the received signal log-likelihoods as shown in (10). Hence, the expression in (11) asymptotically converges to the posterior distribution defined in (5). In addition, based on (11), it is possible to numerically compute the transmitter position estimate $(\hat{x}, \hat{y})$ following the minimum mean squared error (MMSE) estimator (as it was done in [10]):

$$\hat{x}[k] = \mathbb{E}[x|\mathbf{r}_1, ...\mathbf{r}_N] = \iint_{\mathcal{S}} x\,\hat{p}_i(x, y|\mathbf{r}_1, ...\mathbf{r}_N)[k]dxdy \quad (12)$$

$$\hat{y}[k] = \mathbb{E}[y|\mathbf{r}_1, ...\mathbf{r}_N] = \iint_{\mathcal{S}} y\,\hat{p}_i(x, y|\mathbf{r}_1, ...\mathbf{r}_N)[k]\,dxdy \quad (13)$$

Finally, the SSPE algorithm can be summarized in the pseudo-code Algorithm 1.

C. Compression of State Variables

In order for the dynamical system defined in (9) to evolve, each BS-i should communicate its state variable $\mathbf{v}_i$ to the neighboring BSs at each iteration. First, the state variable $\mathbf{v}_i$ is a matrix of the same dimensions as the search grid, since all terms in (9) are defined for each single point. Therefore, the matrix $\mathbf{v}_i$ needs to be compressed since it is impractical to send all the elements of the state variable matrix. Second, the exponential of the state variable $\mathbf{v}_i$ can be seen as a probability distribution defined inside the scene, as

$$p_{\mathbf{v}_i}[k] = C_{\mathbf{v}_i}\ e^{\mathbf{v}_i[k]}, \quad (14)$$

where $C_{\mathbf{v}_i}$ is a normalization factor. It was observed in simulations that $p_{\mathbf{v}_i}$ can be modeled as independent normal distributions in the radial and angular domain. Therefore, we propose to communicate only the first two radial and angular order moments of $p_{\mathbf{v}_i}$. Such radial and angle means and variances can be computed using the law of the unconscious statistician (LOTUS) rule as

$$\mathbb{E}_{p_{\mathbf{v}_i}}\big[f(x,y)\big] = \iint_{\mathcal{S}} f(x,y)\, p_{\mathbf{v}_i}[k]\, dxdy, \tag{15}$$

where $f(x,y)$ is a function defined in the xy-domain, i.e., defined in the scene $\mathcal{S}$. We denote the radial parameter δ_i in terms of x and y as:

$$\delta_i(x,y) = \sqrt{(x-x_i)^2 + (y-y_i)^2}, \tag{16}$$

where, x_i and y_i are the coordinates of the BS-i. Therefore, the radial mean $\hat{\delta}_i$ and variance $\sigma^2_{\hat{\delta}_i}$ can be computed as:

$$\hat{\delta}_i[k] = \mathbb{E}_{p_{\mathbf{v}_i}}\big[\delta_i\big] \quad ; \quad \sigma^2_{\hat{\delta}_i}[k] = \mathbb{E}_{p_{\mathbf{v}_i}}\big[\big(\delta_i - \hat{\delta}_i[k]\big)^2\big]. \tag{17}$$

Similarly, we denote the angular parameter θ_i in terms of x and y and compute its mean $\hat{\theta}_i$ and variance $\sigma^2_{\hat{\theta}_i}$ as:

$$\theta_i(x,y) = \arctan\big(\frac{y-y_i}{x-x_i}\big), \tag{18}$$

$$\hat{\theta}_i[k] = \mathbb{E}_{p_{\mathbf{v}_i}}\big[\theta_i\big] \quad ; \quad \sigma^2_{\hat{\theta}_i}[k] = \mathbb{E}_{p_{\mathbf{v}_i}}\big[\big(\theta_i - \hat{\theta}_i[k]\big)^2\big]. \tag{19}$$

The selection of such parameters is linked to the nature of the position information carried out by the baseband signal and it is further discussed in Section IV-B. Lastly, we communicate four parameters $(\hat{\delta}_i, \sigma^2_{\hat{\delta}_i}, \hat{\theta}_i, \sigma^2_{\hat{\theta}_i})$ instead of communicating all the elements of the state variable matrix $\mathbf{v}_i$ at each iteration.

D. Reconstruction of State Variables

Based on the communicated parameters, the reconstruction of the state variable matrix $\hat{\mathbf{v}}_i$ is done easily as:

$$\hat{\mathbf{v}}_i[k] = -\frac{1}{2\sigma^2_{\hat{\delta}_i}[k]}\big(\delta - \hat{\delta}_i[k]\big)^2 - \frac{1}{2\sigma^2_{\hat{\theta}_i}[k]}\big(\theta - \hat{\theta}_i[k]\big)^2. \tag{20}$$

An additional term is introduced in (9) by using (20) to reconstruct the state variable matrix as

$$\Delta\mathbf{v}_i[k] = \mathcal{L}_i(\mathbf{r}_i|x,y) + \beta \sum_{j=1}^{N} a_{i,j}(\hat{\mathbf{v}}_j[k] + \eta_j[k] - \mathbf{v}_i[k]) \tag{21}$$

The additional term $\eta_j[k]$ is known as coupling noise and has been well studied in the self-synchronization literature [13]–[15]. The effects on the consensus convergence are mainly twofold. First, it slows down the speed of convergence meaning that more iterations are necessary to achieve consensus. Second, it introduces bias to the convergence value. Such bias can be reduced by reducing the value of the control loop gain β, leading again to the use of more iterations.

Finally, taking into account compression of state variables matrices $\mathbf{v}_i$, the SSPE algorithm can be summarized in the pseudo-code given in Algorithm 2.

Algorithm 2 SSPE - Compression of $\mathbf{v}_i$

```
Each BS-i constructs L_i(r_i|x,y) for each point in the search grid based on the received signal r_i.
Each BS-i initializes v_i = 0
for iteration k = 1, 2, ... do
    for BS-i = 1, 2, ... in parallel do
        Compute Δv_i[k] using (9)
        Estimate transmitter position using (12) and (13)
        Update v_i[k+1] = v_i[k] + Δv_i[k]
        Compute (δ̂_i, σ²_δ̂_i, θ̂_i, σ²_θ̂_i) based on v_i[k+1] using (17) - (19) and send them to neighboring BSs
        Receive (δ̂_i, σ²_δ̂_i, θ̂_i, σ²_θ̂_i) sent by neighboring BSs and reconstruct v̂_j[k+1] using (20)
    end for
end for
```

IV. Simulation Results

We consider a scene consisting of N=4 BSs located on the corners of a $100m$-sided square. We study a fully connected network only for convenience and without loss of generality. In fact, the self-synchronization mechanism ensures convergence for any network configuration as long as the network is strongly connected and balanced. The transmitter lies at arbitrary positions inside the rectangular scene and communicates with the BSs over a bandwidth of 20 MHz. At each BS, the processing is done using a single OFDM symbol containing P=64 equispaced pilots with Δf=312.5kHz. The signal-to-noise-ratio (SNR) is assumed to be equal at the four base stations and it is defined as

$$SNR = \frac{1}{P\sigma_w^2} \sum_p |s_p|^2 \tag{22}$$

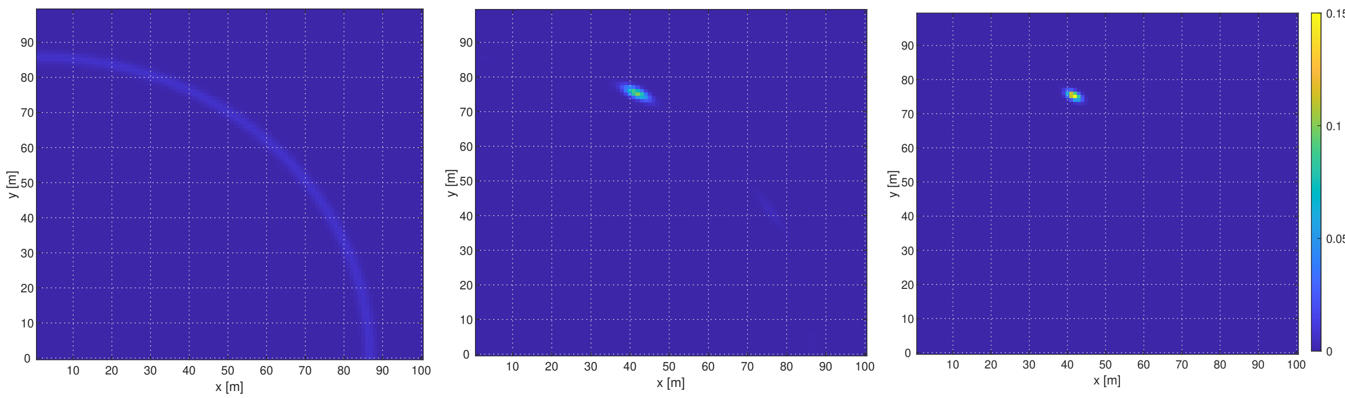
The performance of the proposed algorithm is investigated by assessing the MDE denoted as e_d, which is defined for a particular SNR and iteration values as:

$$e_d = \frac{1}{N_{sim}} \sum_{n=1}^{N_{sim}} \sqrt{(\hat{x}_n - x_n)^2 + (\hat{y}_n - y_n)^2} \tag{23}$$

where N_{sim} is the total number of realizations. The coordinates $(\hat{x}_n, \hat{y}_n)$ and (x_n, y_n) are the estimated and true transmitter position respectively for realization n.

In addition, we also assess the mean value of the KL divergence for every SNR and iteration values as shown in (24). The KL divergence is adopted as a measure of difference between the centralized posterior distribution p (defined in (5)) and the distributed approximation of the posterior $\hat{p}$ (defined in (11)). Notice that the argument $(x, y|\mathbf{r}_1, ...\mathbf{r}_N)$ of the distributions are omitted to simplify the notation.

$$\bar{D}_{KL}(\hat{p}||p) = \frac{1}{N_{sim}} \sum_{n=1}^{N_{sim}} \big(\sum_{\mathcal{S}} \hat{p}_n \log\big(\frac{\hat{p}_n}{p_n}\big)\big) \tag{24}$$



(a) k=1 (b) k=2 (c) k=3

Fig. 1: Evolution of $p_{\mathbf{v}_i}$, as defined in (14), for the BS located at the origin when all the values of $\mathbf{v}_i$ are communicated (No $\mathbf{v}_i$ compression). Four BS are located at the corners of a 100m squared scene and the transmitter is located at $x{=}41m, y{=}76m$.

Finally, the MDE and KL divergence are averaged over $N_{sim}{=}10000$ transmitter positions, channel and noise realizations for each single SNR value.

A. No Compression of State Variables

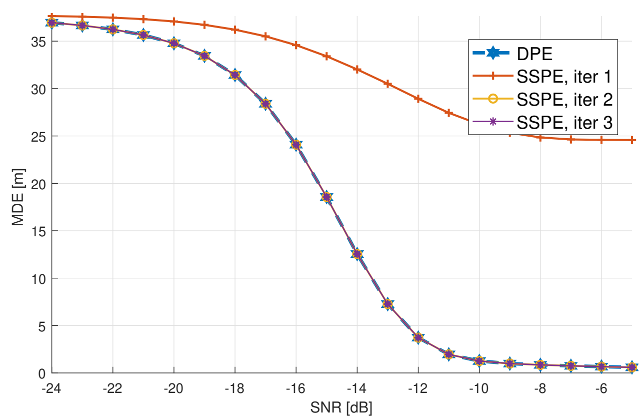
In this section, we discuss the performance of the proposed algorithm summarized in Algorithm 1, i.e., the state variable matrix $\mathbf{v}_i$ is communicated entirely; hence there is no coupling noise introduced to the self-synchronization mechanism. In addition, the value of the control loop gain $\beta{=}0.25$ is chosen for the fastest convergence. Such a value has been obtained based on the eigenvalues of the Laplacian matrix that depends on the logical configuration of the network as explained in [16].

Figure 2 illustrates the MDE and the KL divergence only for BS one, since all the BSs have similar curves when using the proposed algorithm. Figure 2(a) shows the MDE of the proposed algorithm and the centralized DPE algorithm used as a reference (as in [10]).

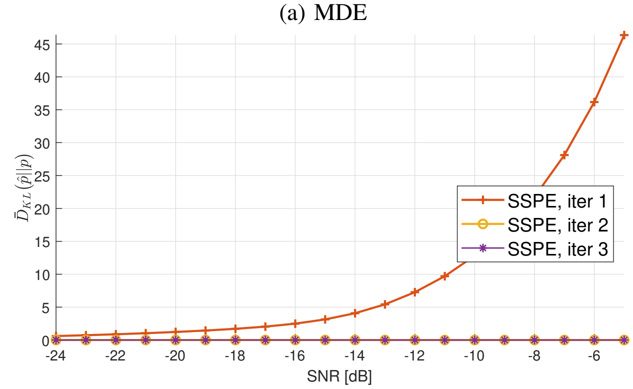
At iteration $k{=}1$, only the local information is available at the BS; hence the MDE is large for all the SNR values. The same effect can be seen for the KL divergence in Figure 2(b). At iteration $k{=}2$, the MDE of the proposed algorithm converges to the one of the centralized DPE algorithm. Such fast convergence is mainly due to the fact that the network is fully connected. Therefore, all information is available at each BS at the second iteration, since there is no compression of the state variable matrix $\mathbf{v}_i$. Moreover, the same effect can be seen in Figure 2(b) where the KL divergence is zero for all SNR values from iteration $k{=}2$ onwards.

B. Compression of State Variables

In this section, we discuss the choice of parameters for representing the state variable matrix $\mathbf{v}_i$ and the performance of the proposed algorithm summarized in Algorithm 2, i.e., when the state variable matrix $\mathbf{v}_i$ is compressed.



(a) MDE

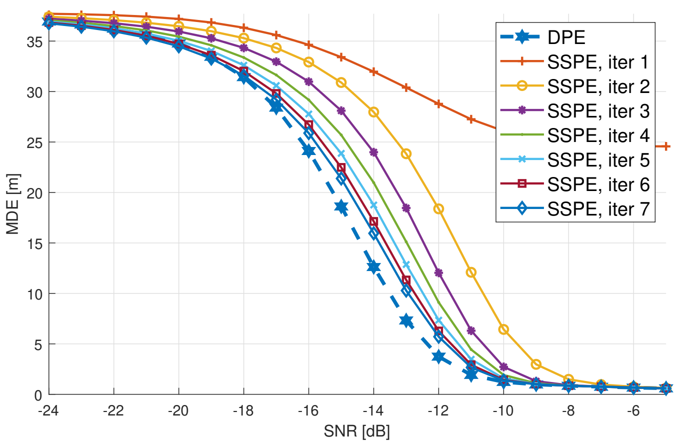


(b) Mean KL divergence

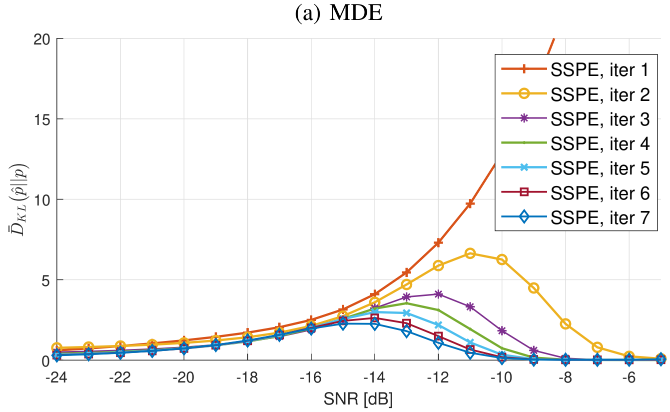
Fig. 2: MDE and mean KL divergence for the case of complete communication of state variable matrix $\mathbf{v}_i$

Figure 1 shows the probability distribution $p_{\mathbf{v}_i}$ for different iterations, for the case where the complete state variables matrix $\mathbf{v}_i$ is communicated. The choice of the radial and angular first two order moments is explained as follows: at iteration $k{=}1$, the state variable $\mathbf{v}_i$ is equal to the log-likelihood of the received signal $\mathcal{L}_i(\mathbf{r}_i|x,y)$; hence, it only contains radial information, since the received signal is delayed due to the distance between transmitter and receiver, as can be seen in Figure 1(a). For iterations $k \geq 2$, the state variable $\mathbf{v}_i[k \geq 2]$ contains radial and angular information. The angular information is obtained from the information communicated by the other BSs, as observed in Figures 1(b) and 1(c).

Due to compression of state variables, there is coupling noise introduced to the self-synchronization mechanism as explained in Section III-D. As observed in Figure 3(a), we need more iterations to reduce such bias. Furthermore, Figure 3(b) shows clearly that the KL divergence is no longer zero for low SNR values. In addition, it can be seen that the KL divergence decreases with more iterations, meaning that the distributed approximated posterior distribution $\hat{p}$ gets closer and closer to the original centralized posterior distribution p.



(a) MDE



(b) Mean KL divergence

Fig. 3: MDE and mean KL divergence for the case of compression of state variable matrix $\mathbf{v}_i$

V. Conclusion

In this work, we introduced a fully distributed localization algorithm based on the self-synchronization mechanism. We also proposed a compression approach for the state variables to reduce the amount of communication overhead between the nodes. Therefore, the localization is done by means of an iterative process, in which each BS shares just a few parameters between all other BSs, hence the transmitter position is available at each BS at the end of each iteration. Numerical results show that the performance of the final algorithm gets close to the performance of a direct localization.

VI. Acknowledgment

The authors acknowledge the financial support of the Walloon Region through the WIN2Wal/2018/1/DI/34 LUMINET project.

References

[1] "Evolved universal terrestrial radio access (e-utra); physical channels and modulation (release 13)," in *Technical Specification 36.211 V13.13.0*, January 2020.

[2] P. Closas, C. Fernandez-Prades, and J. A. Fernandez-Rubio, "Maximum likelihood estimation of position in gnss," *IEEE Signal Processing Letters*, vol. 14, no. 5, pp. 359–362, 2007.

[3] A. J. Weiss, "Direct position determination of narrowband radio frequency transmitters," *IEEE Signal Processing Letters*, vol. 11, no. 5, pp. 513–516, 2004.

[4] R. Zekavat and R. M. Buehrer, *Source Localization: Algorithms and Analysis*, 2019, pp. 59–106.

[5] A. Amar and A. J. Weiss, "New asymptotic results on two fundamental approaches to mobile terminal location," in *2008 3rd International Symposium on Communications, Control and Signal Processing*, 2008, pp. 1320–1323.

[6] I. D. Schizas, A. Ribeiro, and G. B. Giannakis, "Consensus in ad hoc wsns with noisy links—part i: Distributed estimation of deterministic signals," *IEEE Transactions on Signal Processing*, vol. 56, no. 1, pp. 350–364, 2008.

[7] W. Yu, N. D. Gaubitch, and R. Heusdens, "Distributed tdoa-based indoor source localisation," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 6887–6891.

[8] S. Salari, I.-M. Kim, and F. Chan, "Distributed cooperative localization for mobile wireless sensor networks," *IEEE Wireless Communications Letters*, vol. 7, no. 1, pp. 18–21, 2018.

[9] F. Ma, Z.-M. Liu, and F. Guo, "Distributed direct position determination," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 11, pp. 14 007–14 012, 2020.

[10] F. Horlin, M. Van Eeckhaute, T. Van der Vorst, A. Bourdoux, F. Quitin, and P. De Doncker, "Iterative toa-based terminal positioning in emerging cellular systems," in *2017 IEEE International Conference on Communications (ICC)*, 2017, pp. 1–5.

[11] W. Cui, S. Wu, Y. Wang, and Y. Shan, "A gossip-based tdoa distributed localization algorithm for wireless sensor networks," in *2013 2nd International Symposium on Instrumentation and Measurement, Sensor Network and Automation (IMSNA)*, 2013, pp. 783–788.

[12] G. Soatti, M. Nicoli, A. Matera, S. Schiaroli, and U. Spagnolini, "Weighted consensus algorithms for distributed localization in cooperative wireless networks," in *2014 11th International Symposium on Wireless Communications Systems (ISWCS)*, 2014, pp. 116–120.

[13] S. Barbarossa and G. Scutari, "Bio-inspired sensor network design," *IEEE Signal Processing Magazine*, vol. 24, no. 3, pp. 26–35, 2007.

[14] A. Fasano and G. Scutari, "The effect of additive noise on consensus achievement in wireless sensor networks," in *2008 IEEE International Conference on Acoustics, Speech and Signal Processing*, 2008, pp. 2277–2280.

[15] S. Kar and J. M. F. Moura, "Distributed consensus algorithms in sensor networks with imperfect communication: Link failures and channel noise," *IEEE Transactions on Signal Processing*, vol. 57, no. 1, pp. 355–369, 2009.

[16] S. Dhuli, K. Gaurav, and Y. N. Singh, "Convergence analysis for regular wireless consensus networks," *IEEE Sensors Journal*, vol. 15, no. 8, pp. 4522–4531, 2015.