Ranging in IR-UWB Networks II
Modelling of Synchronization Error and Applications to Maximum Likelihood Localization

Robin Scheibler

MICS summer internship
under supervision of
Dr. Ruben Merz
Prof. J.-Y. Le Boudec

LCA2, EPFL

November 10, 2008

Abstract

Based on the results of a previous semester project, an accurate probabilistic model of the synchronization offset of an Impulse-Radio Ultra-Wide Band (IR-UWB) energy receiver is developped. This model is optimized for different Signal-to-Noise Ratio (SNR) points using data sets obtained using the UWB receiver simulator developped at the LCA2. Then, Single Sided Two-Way Ranging as well as Double Sided Two-Way Ranging transactions are studied in detail incoporating both the drift of the nodes clock and the synchronization offset model. A link budget is computed for the system and the transmit power is computed according to the regulations of the FCC. Based on the synchronization offset model and this link budget, a localization simulator is built and the performance of different algorithms based on Maximum Likelihood Estimation is evaluated.

Contents

1 Synchronization Error Modelling

1.1 Review of semester project

As this summer internship is the continuation of a semester project [1], the results obtained during the semester project will first be summarized. During the semester project, the simulator of an energy-detection UWB receiver developed at the LCA2 was used to generate samples of the synchronization offset. The synchronization offset is defined as the difference between the time of arrival of the first path and the time at which the beginning of the training sequence is detected.

A statistical analysis of the collected data was done in order to find a model for the probability distribution of the synchronization offset. It was noticed that the empirical distribution of the offset presented three distinct parts with different behavior. A central peak located approximately between $\alpha \approx -1.5$ and $\beta \approx 1.2$ nanoseconds with the two other parts located on its left and right respectively called nose and tail. The tail is limited to the left by the depth of the backsearch algorithm $\tau_{bs} = 32$ ns. During the analysis, the data set was split into three sets corresponding to those three parts and each set was analysed separately. The goal being to come up with a model of the form :

$$f_{offset}(x) = \begin{cases} p_{nose} f_{nose}(x) & \text{if } -\tau_{bs} \leq x < \alpha \\ p_{peak} f_{peak}(x) & \text{if } \alpha \leq x \leq \beta \\ p_{tail} f_{tail}(x) & \text{if } x > \beta \\ 0 & \text{if } x < -\tau_{bs} \end{cases} \tag{1}$$

where $f_{nose}(x)$, $f_{peak}(x)$ and $f_{tail}(x)$ are the individual probability distribution functions (pdf) over the different sets while p_{nose}, p_{peak} and p_{tail} are the probabilities that a sample belongs to respectively the nose, peak or tail of the distribution.

Through the analysis it was determined that the tail can be accurately modelled with an exponential distribution. On the other hand, the nose contained so few samples that it was enough to model it with a uniform distribution. A truncated Gaussian model was used for the peak, but didn't fit it very well. The parameters of those different distributions were computed using traditionnal maximum likelihood estimators of the parameters on the considered datasets (either nose, peak or tail). Finally the p_{nose}, p_{peak} and p_{tail} were computed. Both the parameters and the probabilities to belong to nose, peak or tail were computed for different values of the signal-to-noise ration (SNR). Fig. 1 shows the empirical distribution along with the pdf

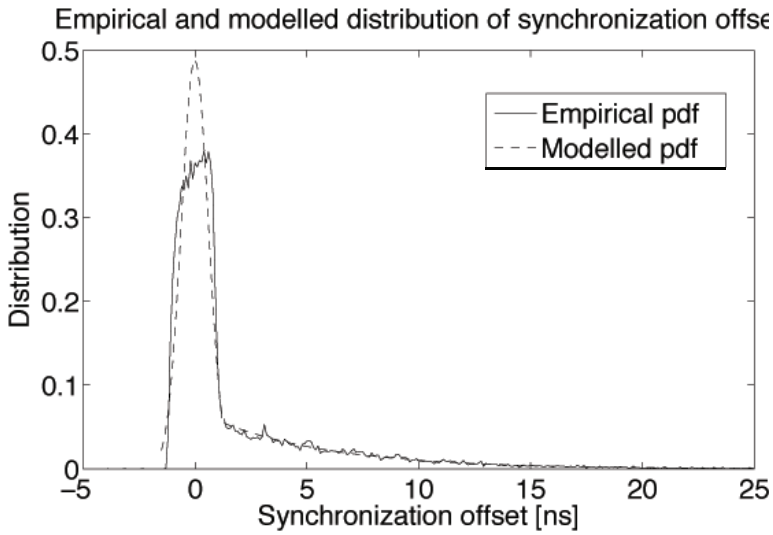


Figure 1: The model developped during the semester project. This is for an SNR of 12 dB. It can be observed that the central peak is too narrow.

model. It can be observed that for the central peak, the model is not accurate at all. Fig. 2 shows the evolution of p_{nose}, p_{peak} and p_{tail} with the SNR.

1.2 Mixture Model

The previous error model developped during the semester project showed some serious limitations. In addition to being cumbersome, it lacks continuity and the center of distribution was too loosely modelled while it is a crucial point of the distribution (up to 90% of the samples at high SNR falls at the center). For those reasons, a better model was developped. The underlying characteristics of the system were taken into account to produce a model that fits the data very tightly.

These underlying characteristics are :

- The channel
- The synchronization algorithm
- The sampling frequency

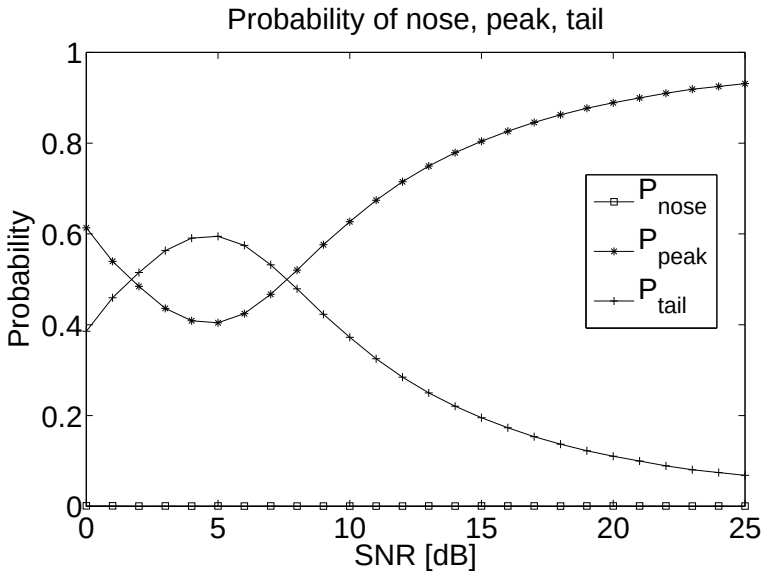


Figure 2: The evolution of p_{nose}, p_{peak} and p_{tail} with the SNR.

- The backsearch algorithm[1]

The model is then constructed in three steps. Firstly, the offset between the time of arrival of the first path and the first observable path is modelled as a random variable (r.v.) T. The exponential behavior of the tail that was observed previously can be attributed to the channel model used in the simulator which use an exponential decay for the power of the different paths. The central peak of the distribution is due partly to the synchronization algorithm and the backsearch algorithm which improve the observability of the first path. Finally, the backsearch fails sometimes and synchronization happens before the true time of arrival (*i.e.* the offset is negative). In this case, the error due to the backsearch is located between $-\tau_{bs}$ and 0 ns, where $\tau_{bs} = 32$ ns in our case is the depth of the backsearch. Therefore a mixture distribution with an exponential component accounting for the channel, a centered Gaussian component for the path that are correctly detected and a uniform component for the premature synchronization is chosen as the

[1]The backsearch algorithm was introduced to cope with situations whereby the first path is not the strongest path. It operates by going back from the detected path and try to detect some weaker paths that were missed. The search region goes from 0 to $-\tau_{bs}$ ns. In our case, $\tau_{bs} = 32$ ns was chosen.

probability density function (pdf) of T :

$$f_T(t) = w_1 f_g(t) + w_2 f_e(t) + w_3 f_u(t) \tag{2}$$

where $f_g(t)$ and $f_e(t)$ are the pdf of respectively a centered Gaussian r.v. with variance σ^2 and an exponential r.v. with decay parameter λ :

$$f_g(t) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{t^2}{2\sigma^2}} \qquad f_e(t) = \begin{cases} \lambda e^{-\lambda t} & \text{if } t \geq 0 \\ 0 & \text{if } t < 0 \end{cases}.$$

$$f_u(x) = \begin{cases} \frac{1}{\tau_{bs}} & \text{if } x \in [-\tau_{bs}; 0] \\ 0 & \text{o.w.} \end{cases}$$

The second step is to add the effect of the sampling. The sampling is modelled as an additive r.v. $S \sim \mathcal{U}[-\rho/2; \rho/2]$ where ρ is the sampling period. The error becomes now :

$$E = T + S$$

and the pdf of S is :

$$f_S(s) = \begin{cases} \frac{1}{\rho} & \text{if } -\frac{\rho}{2} \leq s \leq \frac{\rho}{2} \\ 0 & \text{o.w.} \end{cases}, \tag{3}$$

The pdf of the sum of two r.v. is given by the convolution of their respective pdf.

$$f_E(x) = f_T(x) * f_S(x)$$

Let's first compute the convolution of a given pdf $f(t)$ with the pdf of a r.v. uniformely distributed in $[a; b]$, $f_U(t)$:

$$\begin{aligned} f(t) * f_U(t) &= \int_{-\infty}^{\infty} f(x) f_U(t-x) dx = \frac{1}{b-a} \int_{t-b}^{t-a} f(x) dx \\ &= \frac{1}{b-a} \left[\int_{-\infty}^{t-a} f(x) dx - \int_{-\infty}^{t-b} f(x) dx \right] \\ &= \frac{1}{b-a} \left[F(t-a) - F(t-b) \right] \end{aligned}$$

where $F(t)$ is the cumulative distribution function (cdf) corresponding to the pdf $f(t)$. Using the linarity of the convolution, we can apply this result

separately on the components of the mixture. However, since the support of $f_S(t)$ is much shorter than the support of $f_u(t)$, when convolved together, the result is essentially similar to $f_u(t)$ with small border effects that can be neglected in the model. Therefore, we use the approximation :

$$f_S(x) * f_u(x) \approx f_u(x).$$

Hence, the pdf of E can be written :

$$f_E(x) = w_1 \frac{\left[F_g\left(x+\frac{\rho}{2}\right) - F_g\left(x-\frac{\rho}{2}\right)\right]}{\rho} + w_2 \frac{\left[F_e\left(x+\frac{\rho}{2}\right) - F_e\left(x-\frac{\rho}{2}\right)\right]}{\rho} + w_3 f_u(x)$$

where $F_g(x)$ and $F_e(x)$ are the cdf of respectively the Gaussian and Exponential distributions. In the system considered, the sampling period is $\rho = 2$ ns. So our error model has the following pdf :

$$f_E(x) = w_1 \frac{\left[F_g\left(x+1\right) - F_g\left(x-1\right)\right]}{2} + w_2 \frac{\left[F_e\left(x+1\right) - F_e\left(x-1\right)\right]}{2} + w_3 f_u(x)$$

where :

$$F_g(x) = \frac{1}{\sqrt{2\pi}\sigma} \int_{-\infty}^{x} e^{-\frac{t^2}{2\sigma^2}} dt \qquad F_e(x) = \begin{cases} 1 - e^{-\lambda t} & \text{if } x \geq 0 \\ 0 & \text{if } x < 0 \end{cases}$$

and since $f_E(x)$ is a pdf we have the following counstraint :

$$w_1 + w_2 + w_3 = 1$$

During the optimization of the parameters it was noticed that the distribution is not always centered. Therefore, non-centrality parameters where added to the Gaussian and exponential components :

$$\begin{aligned} f_E(x) &= w_1 \frac{\left[F_g\left(x-\mu_1+1\right) - F_g\left(x-\mu_1-1\right)\right]}{2} \\ &\quad + w_2 \frac{\left[F_e\left(x-\mu_2+1\right) - F_e\left(x-\mu_2-1\right)\right]}{2} \\ &\quad + w_3 f_u(t) \end{aligned}$$

The cdf of E can also be derived and we find :

$$\begin{aligned} F_E(x) &= \mathbb{P}\left\{E \leq x\right\} \\ &= \frac{w_1}{2}\left[(x-\mu_1+1)F_g(x-\mu_1+1) - (x-\mu_1-1)F_g(x-\mu_1-1)\right. \\ &\qquad \left. + \sigma^2\left(f_g(x-\mu_1+1) - f_g(x-\mu_1-1)\right)\right] \\ &\quad + \frac{w_2}{2}\left[g(x-\mu_2+1) - g(x-\mu_2-1)\right] \\ &\quad + w_3 F_u(x) \end{aligned}$$

where :

$$g(x) = \int_{-\infty}^{x} F_e(t)dt = \begin{cases} x + \frac{1}{\lambda}\left(e^{-\lambda x} - 1\right) & \text{if } x \geq 0 \\ 0 & \text{o.w.} \end{cases}$$

and $F_u(x)$ is the cdf of $f_u(x)$:

$$F_u(x) = \begin{cases} 0 & \text{if } x < -\tau_{bs} \\ \frac{x+\tau_{bs}}{\tau_{bs}} & \text{if } -\tau_{bs} \leq x \leq 0 \\ 1 & \text{if } x > 0 \end{cases}$$

1.3 Optimization of the Model

Once the model derived in the previous section was obtained, the next logical step was to optimize the parameters to fit the data. The model has seven parameters, out of which six are independent :

- w_1, w_2 and w_3, the weights of the mixture with constraint :

$$w_1 + w_2 + w_3 = 1$$

- μ_1 and μ_2, the non-centrality parameters of respectively the Normal and exponential components of the mixture
- σ, the standard deviation of the Normal component
- λ, the decay parameter of the exponential component

When it comes to optimize mixture models, the standard solution is the EM algorithm. It solves elegantly the constrained optimization of the weights. However, for our model there is no closed form of the Maximization step for μ_1, μ_2, σ and λ. Then, instead of the EM step, we do one step in the direction of the gradient of the log-likelihood function :

$$l(\theta, \bar{x}) = \sum_{n=1}^{N} \log f_{E|\Theta}(x_n|\theta) \quad , \quad \theta = \{w_1, w_2, w_3, \mu_1, \mu_2, \sigma, \lambda\}$$

where $\bar{x} = [x_1, \ldots, x_N]$ is the dataset. The iterative optimization procedure is completely described in Algo. 1. ν is the gradient ascent step size. The derivation of the gradient of $l(\theta, \bar{x})$ is given in Appendix B. The initial conditions for w_1, w_2 and w_3 are the probabilities to belong to respectively the peak, tail and nose, as found in Section 1.1. The initial condition of λ is set to the parameter for the distribution of the tail also found in Section 1.1.

Algorithm 1 Model parameter optimization

Require: $\bar{x}$, $\theta^{(0)} = \{w_1^{(0)}, w_2^{(0)}, w_3^{(0)}, \mu_1^{(0)}, \mu_2^{(0)}, \sigma^{(0)}, \lambda^{(0)}\}$
Ensure: Optimal θ
  /* Initialization */
  $\theta \leftarrow \theta^{(0)}$
  repeat
    /* EM step */
    for $n = 1, \ldots, N$ **do**
      /* Posterior probability */
      $\hat{w}_{1,n} = \frac{w_1[F_g(x_n-\mu_1+1,\sigma)-F_g(x_n-\mu_1-1,\sigma)]}{2f_E(x_n,\theta)}$
      $\hat{w}_{2,n} = \frac{w_2[F_e(x_n-\mu_2+1,\lambda)-F_e(x_n-\mu_2-1,\lambda)]}{2f_E(x_n,\theta)}$
      $\hat{w}_{3,n} = \frac{w_3 f_u(x_n)}{f_E(x_n,\theta)}$
    end for
    for $i = 1, 2, 3$ **do**
      /* Weight update */
      $w_i = \frac{1}{N} \sum_{j=1}^{N} \hat{w}_{i,j}$
    end for
    /* Gradient ascent step */
    $\mu \leftarrow \mu + \nu \frac{\partial}{\partial \mu} l(\theta, \bar{x})$
    $\sigma \leftarrow \sigma + \nu \frac{\partial}{\partial \sigma} l(\theta, \bar{x})$
    $\lambda \leftarrow \lambda + \nu \frac{\partial}{\partial \lambda} l(\theta, \bar{x})$
  until convergence reached
  return θ

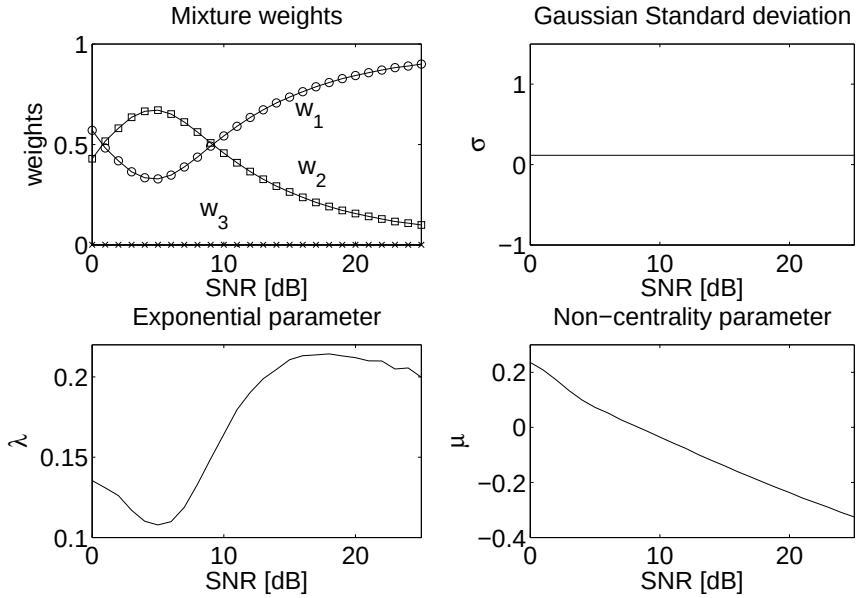


Figure 3: The optimal parameters of the model for single user communication.

μ_1 and μ_2 were set to zero and it was found experimentally that 0.1 is a good inital value for σ.

The optimization procedure was run for SNR points 0 to 25 dB for which data was collected through simulations as described in Section 1.1. However, the values found for σ were not stable but oscillated around some average value. Therefore we set $\sigma = 0.115$, its approximate average value. On the other hand it was observed that μ_1 and μ_2 are not independent parameters and they were thus reduced to a single parameter $\mu = \mu_1 = \mu_2$. Then the optimization algorithm was run again to get the new values of the parameters.

The result is shown in Fig. 3 for single user communication and in Fig. 4 when there is an interferer with 10 times superior transmit power. The numerical values of the optimized parameters can be found in Appendix C.

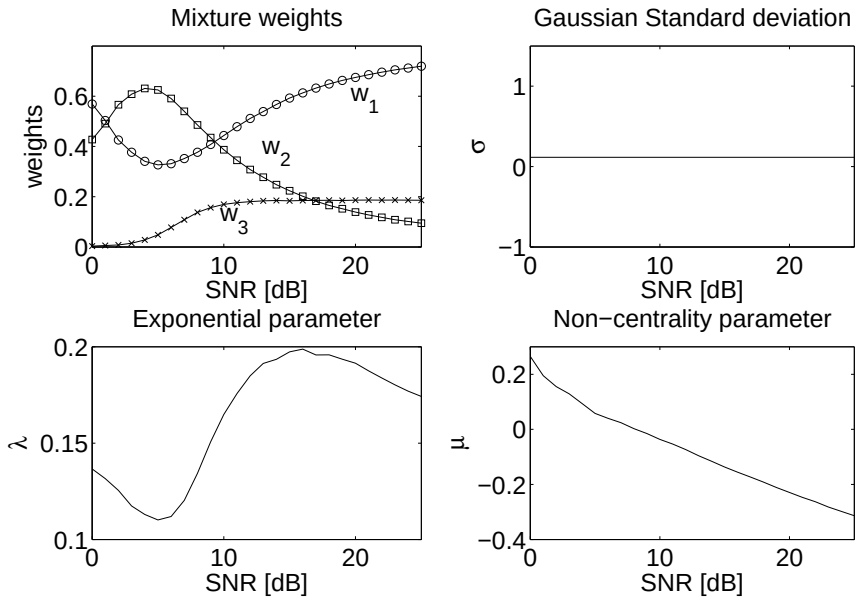


Figure 4: The optimal parameters of the model for one user and one interferer with 10 times transmit power.

The final model we obtain is :

$$\begin{aligned} f_E(x) &= w_1 \frac{[F_g(x-\mu+1) - F_g(x-\mu-1)]}{2} \\ &\quad + w_2 \frac{[F_e(x-\mu+1) - F_e(x-\mu-1)]}{2} \\ &\quad + w_3 f_u(t) \end{aligned}$$

The optimized model is plotted along the empirical pdf for some SNR points in Fig. 5, Fig. 6 and Fig. 7. In Fig. 8 we can see QQ-plots of the empirical data against samples generated according to the model. It can be seen that for middle to high SNR, the model fits very tightly the data. However when the SNR is low, the central peak tends to be narrower than the model. This is probably due to the fact that when the noise level is high, the sampling error becomes critical, *i.e.* when the sampling error is too high, the synchronization algorithm fails. The model could be corrected by letting the sampling error width variable. However, the gain might not be worth the extra complexity introduced in the model.

2 Simulation of localization scenarios

2.1 Realistic Two-Way Ranging transactions

The classical technique to make Time of Arrival (ToA) measurement without synchronization of the nodes' clock is to use Two-Way Ranging (TWR) transactions. In this case, the measurement is made on the round-trip time of a packet, taking into account some turnaround and processing time at the second node. These measurements are then used to estimate the propagation time t_p between the two nodes. The effect of clock drifts on TWR was studied in a another previous semester project [2]. Two flavours of TWR were analyzed, Single Sided TWR (SS-TWR) and Double Sided TWR (DS-TWR). In order to get an idea of the measurement error, we added in addition the error due to the synchronization offset.

Before describing in details the two techniques a quick glossary of the variables used is given :

Node A : node making the measurement

Node B : node with which the measurement is done

t_p : the propagation time

$\hat{t}_p^{(ss)}$: The SS-TWR estimate of the propagation time

$\hat{t}_p^{(ds)}$: The DS-TWR estimate of the propagation time

t_{rd} : the round-trip time measurement

r : the total time spent at node B before the reply is transmitted including the turnaround time of the transciever t_{rxtx}, the processing time t_{proc} and the preamble length L.

ϵ_A : clock drift of node A

ϵ_B : clock drift of node B

t_{rxtx} : turnaround time of the UWB transceiver

t_{proc} : processing time

L : preamble duration

T : random offset between the first path and first observable path as described in the synchronization error model in Section 1.2 with distribution given in Eq. (2).

S : random sampling offset with pdf given in Eq. (3).

SS-TWR As illustrated in Fig. 9, only A measures the round-trip time. We have :

$$t_{rd}^{A} = (1+\epsilon_A)\left[2t_p + \frac{r}{1+\epsilon_B} + T_B + S_B + T_A + S_A\right]$$

and the propagation time is estimated as :

$$\hat{t}_p^{(ss)} = \frac{t_{rd}^{A} - r}{2} = \underbrace{(1+\epsilon_A)t_p}_{\mathbf{A}} + \underbrace{\frac{r}{2}\left[\frac{1+\epsilon_A}{1+\epsilon_B} - 1\right]}_{\mathbf{B}} + \underbrace{\frac{1+\epsilon_A}{2}(T_B + S_B + T_A + S_A)}_{\mathbf{C}}$$

Now to get a simpler model, we remove the errors which are not significant (*i.e.* $\ll t_p$). Note that t_p, T_B, T_A, S_B and S_A are in the order of nanoseconds (10^{-9}), while ϵ_A and ϵ_B are in the tens of microseconds (10^{-5}) and r in the hundreds of microseconds (10^{-4}). Therfore, the terms in ϵ_A in **A** and **C** can be neglected. On the other hand, **B** needs some attention. Let's first use the Taylor expansion of the middle terme :

$$\frac{1+\epsilon_A}{1+\epsilon_B} - 1 = (1+\epsilon_A)(1-\epsilon_B + O(\epsilon_B^2)) - 1 = \epsilon_A - \epsilon_B + O(10^{-10})$$

and **B** becomes :

$$\frac{r}{2}\left[\frac{1+\epsilon_A}{1+\epsilon_B} - 1\right] = \frac{r}{2}(\epsilon_A - \epsilon_B) + O(10^{-14})$$

where the terms in $O(10^{-14})$ can be neglected since it is 5 orders of magnitude smaller than t_p.

We can now write the error model for SS-TWR :

$$E^{(ss)} = 2(\hat{t}_p - t_p) = T_A + T_B + S_A + S_B + r(\epsilon_A - \epsilon_B) \tag{4}$$

DS-TWR In DS-TWR, the round-trip time is measured by both node A and B, and the results are combined to estimate the propagation time. The measuring method is illustrated in Fig. 10. As it will be shown, this technique allows to elegantly remove the effect of the clock drifts at the cost of some communication overhead.

We have two SS-TWR measurements :

$$t_{rd}^{A} = (1+\epsilon_A)\left[2t_p + \frac{r}{1+\epsilon_B} + T_B^{(1)} + S_B^{(1)} + T_A + S_A\right]$$

and

$$t_{rd}^{B} = (1+\epsilon_B)\left[2t_p + \frac{r}{1+\epsilon_A} + T_A + S_A + T_B^{(2)} + S_B^{(2)}\right]$$

The propagation time estimate is now taken as :

$$\hat{t}_p^{(ds)} = \frac{t_{rd}^{A} + t_{rd}^{B} - 2r}{4}$$

which can be expanded into :

$$\begin{aligned}\hat{t}_p^{(ds)} &= \underbrace{\left(1 + \frac{\epsilon_A + \epsilon_B}{2}\right) t_p}_{\mathbf{A}} + \underbrace{\frac{r}{4}\left[\frac{1+\epsilon_a}{1+\epsilon_b} + \frac{1+\epsilon_b}{1+\epsilon_a} - 2\right]}_{\mathbf{B}} \\ &\quad + \underbrace{\left[(1+\epsilon_A)\left(T_B^{(1)} + S_B^{(1)} + T_A + S_A\right) + (1+\epsilon_B)\left(T_A + S_A + T_B^{(2)} + S_B^{(2)}\right)\right]}_{\mathbf{C}}\end{aligned}$$

As for SS-TWR, the terms in ϵ_A and ϵ_B in **A** and **C** can be neglected. And using taylor expansions in **B** yields :

$$\begin{aligned}\mathbf{B} &= \frac{r}{4}\left[\frac{1+\epsilon_a}{1+\epsilon_b} + \frac{1+\epsilon_b}{1+\epsilon_a} - 2\right] \\ &= \frac{r}{4}((1+\epsilon_A)(1-\epsilon_B+O(\epsilon_B^2)) + (1+\epsilon_B)(1-\epsilon_A+O(\epsilon_A^2)) - 2) \\ &= \frac{r}{4}(-2\epsilon_A\epsilon_B + O(\epsilon_A^2) + O(\epsilon_B^2)) \\ &= O(10^{-14})\end{aligned}$$

which is much smaller than t_p and can thus be neglected.

The remaining error is completely determined by the synchronization and can be written :

$$E^{(ds)} = 4\left(\hat{t}_p^{(ds)} - t_p\right) = 2T_A + 2S_A + T_B^{(1)} + S_B^{(1)} + T_B^{(2)} + S_B^{(2)}$$

2.2 Maximum Likelihood Localization

In maximum likelihood localization, we try to estimate the coordinate $\theta = [x, y]$ using N TWR measurement :

$$\hat{\boldsymbol{t}}_p = \left[\hat{t}_p^{(1)}, \ldots, \hat{t}_p^{(N)}\right]$$

between the target and N reference nodes with coordinates $[x_i, y_i]$, $i = 1, \ldots, N$. The propagation time between the i^{th} reference node and the target can be expressed as a function of the coordinates of the target node :

$$t_p^{(i)}(x,y) = \frac{\sqrt{(x-x_i)^2 + (y-y_i)^2}}{c} \quad ; \quad i = 1, \ldots, N$$

where $c = 3 \times 10^8$ m/s is the speed of light, and the TWR measurement errors can be expressed as :

$$E_i^{(twr)} = k\left(\hat{t}_p^{(i)} - t_p^{(i)}(x,y)\right) \quad ; \quad i = 1, \ldots, N$$

where $k = 2$ for SS-TWR and $k = 4$ for DS-TWR.

Now, the negative log-likelihood function of the coordinates θ of the target is :

$$l(\theta, \hat{\boldsymbol{t}}_p) = -\sum_{i=1}^{N} \log f_{E_i^{(twr)}}\left(k\left(\hat{t}_p^{(i)} - t_p^{(i)}(x,y)\right)\right)$$

and the Maximum Likelihood Estimator (MLE) of θ is :

$$\hat{\theta} = \underset{\theta \in \mathbb{R}^2}{\arg\min}\, l(\theta, \hat{\boldsymbol{t}}_p)$$

Depending on the probabilistic model we use for the measurement error $E^{(twr)}$, different error functions can be obtained.

Quadratic Error function The quadratic error function is obtained by assuming the measurement errors to be independent and identically distributed (iid) Normal random variables. Then, the MLE is obtained by minimizing :

$$l(\theta, \hat{\boldsymbol{t}}_p) = \sum_{i=1}^{N}\left(\hat{t}_p^{(i)} - t_p^{(i)}(x,y)\right)^2$$

Although this model is far from representing the reality, it is widely used in the literature (for example in [3, 4]) and is used here as a benchmark to compare to the performance of the estimator developped using the model derived in Section 1.2.

Double-sided Exponential Error function According to the model of Section 1.2, the measurement error has an exponential tail behavior. If we reduce this model to a double-sided exponential and consider all the errors to be iid with pdf :

$$f(t) = \lambda e^{-\lambda|t|}$$

then the MLE can be found by minimizing the sum of the absolute value of the measurement errors :

$$l(\theta, \hat{\boldsymbol{t}}_p) = \sum_{i=1}^{N}\left|\hat{t}_p^{(i)} - t_p^{(i)}(x,y)\right|$$

SS-TWR Error function The two previous error functions presented don't consider the clock drifts at all. As the error due to the drift is very significative this leads to poor performance. To try to enhance this, a new error functions based on the error of SS-TWR shown in Section 2.1, Eq. (4). Since the drift ϵ_t of the target node is common to all the measurements, it is considered as a parameter. The drifts of the reference nodes are assumed uniform iid random variables :

$$\epsilon_i \overset{iid}{\sim} \mathcal{U}[-\alpha/2; \alpha/2] \quad ; \quad i = 1, \ldots, N$$

and we have now the following error model :

$$\begin{aligned} E_i &= 2\left(\hat{t}_p^{(i)} - t_p^{(i)}(x,y)\right) - r\epsilon_t \\ &= T_i^{(t)} + T_i + S_i^{(t)} + S_i - r\epsilon_i \qquad i = 1, \ldots, N \\ &\approx T_t^{(t)} + T_i - r\epsilon_i \end{aligned}$$

where $T_i^{(t)}$ and T_i are the synchronization errors at respectively the target and the i^{th} reference node, $S_i^{(t)}$ and S_i are similarly the sampling errors in the synchronization. The approximation is made because the support of $S_i^{(t)}$ and S_i is much smaller than the support of $r\epsilon_i$. Note that the E_i are iid. Their pdf $f_{E_i}(t)$ was computed and is given in Appendix A. Using this pdf we find the MLE by minimizing :

$$l(\theta, \epsilon_t, \hat{\boldsymbol{t}}_p) = -\sum_{i=1}^{N} \log f_{E_i}\left(2\left(\hat{t}_p^{(i)} - t_p^{(i)}(x,y)\right) - r\epsilon_t\right)$$

If the drifts can somehow be approximed, then the following measurement error model can be used and the sampling errors $S_i^{(t)}$ and S_i are replaced by a single uniform random variable U_i :

$$E_i = \hat{t}_p^{(i)} - t_p^{(i)}(x,y) + r(\epsilon_i - \epsilon_t) = T_i^{(t)} + T_i + U_i \quad ; \quad i = 1, \ldots, N$$

and the derivation for the pdf are the same as in Appendix A with different bounds for the uniform random variable.

2.3 Link Budget

In order to draw samples of the synchronization error, we need to establish a link budget to compute the signal-to-noise (SNR) ratio per link. The SNR is given by :

$$SNR = \frac{P_{rx}}{N_0 B}$$

where P_{rx} is the received power, N_0 the thermal noise power per Herz and B the bandwidth of the signal. In order to get the SNR as a function of the transmit power P_{tx}, we choose the free space propagation model. In free space environnement, the relation between P_{rx} and the transmit power P_{tx} is given by the Friis formula [5] :

$$\frac{P_{rx}}{P_{tx}} = G_{tx}G_{rx}\left(\frac{\lambda}{4\pi R}\right)^2$$

where G_{tx} and G_{rx} are respectively the gains of the transmit and receive antennas, λ is the wavelength and R the distance between transmitter and receiver. The thermal noise power per Herz is only function of the environnement and is given by :

$$N_0 = k_B T$$

where k_B is Boltzmann's constant and T the temperature in Kelvin degrees. Then the SNR can be rewritten as :

$$SNR = \frac{\lambda^2 G_{tx} G_{rx} P_{tx} R^{-\alpha}}{(4\pi)^2 k_B T B}$$

where α is a path loss exponent. The path loss exponent was introduced to account for other environnements than free space. If we set $\alpha = 2$, then we are considering the free space propagation, but for example for indoor environnements the path loss exponent is closer to $\alpha = 4$.

Therefore the SNR is a direct function of the transmit power P_{tx}. However, the transmit power level is severly restricted by rules and regulations edicted by national organisms like the FCC in the USA or the OFCOM in Switzerland. To set P_{tx} we refer to the FCC regulations described in [6]. The limitations on P_{tx} are :

- The average transmit power must be less -41.3 dBm / MHz.
- The peak signal strength over a 50 MHz bandwidth is limited to 20 dB with respect to the maximum permitted average emission level[2].

[2] A second rule precise that the peak power over the entire bandwidth is variable based on the amount by which the -10 dB exceeds 50 MHz according to the following formula :

$$20 + 20\log_{10}(\text{-10 dB bandwidth of the emission in Hz}/50\text{ MHz})$$

with the further stipulation that the absolute peak emission level not be permitted to exceed the average limit by more than 60 dB.

The UWB system we are considering is bandlimited between 4.15 GHz and 4.65 GHz and has hence a bandwidth of $B = 500$ MHz with a center frequency $f_c = 4.4$ GHz. The preamble for synchronization in 802.15.4a is composed of $L_p = 31$ symbols from a ternary code with symbols {-1, 0, 1} out of which 16 are 0. This means there is $N_p = 15$ pulses in the preamble. The duration of one symbol is $T_s = 128$ ns while each pulse has a duration of $T_p = 2$ ns. Firstly, we compute the average power per MHz :

$$P_{1MHz} = 10^{\frac{-41.3}{10}} \approx 8 \times 10^{-5}\,\text{mW/MHz}$$

Then, the total power that can be transmitted is :

$$P_{tot} = P_{1MHz} \times B$$

We can now compute the power for a single pulse :

$$P_p = \frac{L_p}{N_p} \times \frac{P_{tot} \times T_s}{T_p} \approx 5\,\text{mW/pulse} \approx 7\,\text{dBm}$$

Now we need to check the peak power according to the second rule :

$$P_{50MHz} = \frac{P_p}{B/50} = 0.5\,\text{mW}/50\text{ MHz} = -3\,\text{dBm}/50\text{ MHz}$$

Since :

$$10\log_{10}\left(\frac{P_{50MHz}}{P_{tot}}\right) \approx 11\,\text{dB} < 20\,\text{dB}$$

the second regulation is respected and we set :

$$P_{tx} = 5\,\text{mW} \approx 7\,\text{dBm}$$

2.4 Simulation Results

Finally, simulations were used in order to evaluate the performance of the different maximum likelihood algorithms presented in Section 2.2. Before giving the performance results, the simulator will be briefly presented.

The scenario considered is the following. $N + 1$ nodes are uniformely distributed on a plane square of surface L^2. N nodes n_i, $i = 1, \ldots, N$, are chosen as reference nodes, i.e. their coordinates are known, and one node n_t is the target node, i.e. the goal is to estimate the coordinates of n_t based on distance measurements between the reference nodes and the target node. It is assumed that no special synchronization protocol is used. The simulation goes as follows :

1. The coordinates $\theta_i = [x_i, y_i]^T$, $i = 1, \ldots, N$ of the reference nodes and the coordinates $\theta_0 = [x_0, y_0]^T$ of the target nodes are drawn uniformely in the squared surface.

$$\theta_i \overset{i.i.d.}{\sim} \mathcal{U}[0; S]^2 \qquad ; \qquad i = 0, \ldots, N$$

2. The distances between the reference nodes and the target are computed :

$$d_i = \|\theta_i - \theta_0\|_2 \qquad ; \qquad i = 1, \ldots, N$$

3. The SNR is computed for every link based on the link budget of Section 2.3 :

$$\psi_i = \frac{\lambda^2 G_{tx} G_{rx} P_{tx} d_i^{-\alpha}}{(4\pi)^2 k_B T B}$$

where :

- The gains of the antennas is chosen to be $G_{tx} = G_{rx} = 2$ dB, according to the litterature [7, 8, 9].
- The bandwidth B of our system is 500 MHz.
- The wavelength λ is chosen with respect to the center frequency of the system $f_c = 4.4$ GHz.

$$\lambda = \frac{c}{f_c} \approx 0.0682 \text{ m}$$

- $k_B \approx 1.38 \times 10^{-38}$ J/K is Boltzmann constant.
- The temperature is chosen to be $T = 298.15$ K (25 °C).
- The propagation exponent is set to $\alpha = 4$ to model indoor environnement.
- The transmit power is chosen to be $P_{tx} = 5$mW according to the link budget of Section 2.3.

However, we need to restrict those values to $\{0, \ldots, 25\}$ in dB since we only optimized the model parameters for those SNR points.

$$\Psi_i = \max\{0 \ ; \ \min\{[10 \log_{10}(\psi_i)] \ ; \ 25\}\}$$

4. The drifts ϵ_i, $i = 0, \ldots, N$ of the nodes (where ϵ_0 corresponds to the target node) are drawn independently from a zero-mean uniform distribution with width α.

$$\epsilon_i \overset{i.i.d.}{\sim} \mathcal{U}\left[-\frac{\alpha}{2}; \frac{\alpha}{2}\right] \qquad ; \qquad i = 0, \ldots, N$$

5. The propagation time estimate $\hat{t}_p^{(i)}$ corresponding to d_i is calculated using the models described in Section 2.1. The processing time at the node is limited to the turnaround time, which is set to $t_{rxtx} = 400$ μs, value which can be found in [10]. In the calculations we use the exact propagation time $t_p^{(i)} = \frac{d_i}{c}$, where c is the speed of light. Three cases are explored :

   - SS-TWR with both drifts and synchronization errors.
   - DS-TWR with both drifts and synchronization errors.
   - SS-TWR when the drifts are zero.

   Moreover, two values for the range drift were simulated :

   - $\alpha = 40$ ppm
   - $\alpha = 80$ ppm

6. Then, θ_0 is estimated based on $\left\{\hat{t}_p^{(i)}\right\}_{i=1}^{N}$ and $\{\theta_i\}_{i=1}^{N}$ by minimizing one of the error functions described in Section 2.2.

$$\hat{\theta}_0 = \underset{\theta_0 \in \mathbb{R}}{\arg\min} l\left(\theta, \hat{\boldsymbol{t}}_p\right)$$

   The error function is optimized using the Matlab function *fminsearch*. The initial value of θ_0 is calculated using a direct estimation method described in [4].

7. Finally the squared distance between the estimated position and the true position of the target is recorded :

$$e = \left\|\theta_0 - \hat{\theta}_0\right\|_2^2$$

The performance metric is then defined as :

$$E = \sqrt{\frac{1}{N}\sum_{k=0}^{K-1} e_i}$$

where k is the simulation index and the total number of runs is $K = 10000$. The simulations were repeated with different number of reference nodes. We expect the localization error to reach a minimum when the number of reference nodes is large, i.e. adding more reference nodes brings no increase in performance. However, the best algorithm should not only reach the

smallest error possible, but should also converge to this value quite fast because in a practical system there might not be a large number of reference nodes available.

The results of the simulations are shown in Fig. 11 for SS-TWR and in Fig. 12 for DS-TWR. Finally Fig. 13 shows the performance of the algorithm when there is no drift, only synchronization errors. For SS-TWR without drifts, the uniform variables modelling the drift is replaced by another uniform variable that accounts for all the sampling errors and others. Its support is empirically set to : $[-1.5; 1.5]$ ns.

Overall, we can see that the use of an accurate error model in the maximum likelihood localization brings an increase in performance compared to methods with less complexity like the quadratic or Laplacian approximation. The exception is for SS-TWR when there is no drift. Then even if they have the same asymptotical performance, the Laplacian error function has a better performance than the SS-TWR error function when the number of reference nodes is low. In the case of DS-TWR, even though the SS-TWR error function doesn't describe accurately the error its performance is still the best one. Thus we can reasonnably presume that an accurate model would give even better performance.

However, the overall performance is rather poor. Even with a large number of nodes, the performance doesn't go far below half a meter for any of the considered scenario.

3 Conclusion and Future Work

A very accurate probabilistic model of the synchronization offset was developped based on the statistical analysis of simulation results. Its parameters were optimized to fit the data set of the simulation results. Then, SS-TWR and DS-TWR were reviewed taking into accounts both the drifts of the nodes clock and the synchronization offset model. Simplified error models for SS-TWR and DS-TWR were derived and in the case of SS-TWR its probability density function was derived as well.

To perform the localization itself, the performance of the maximum likelihood estimation with three different error functions was analyzed. A quadratic error function, a distance (Laplacian) error function and an error function based on the SS-TWR error model. In order to do the performance evaluation, a detailed link budget was established based on the system characteristics as well as the official rules and regulations governing the use of UWB. Based on this link budget and the optimized synchronization offset

model, a localization simulator was created to test the different algorithms. This simulator can be reused in the future to test new algorithms.

The considered scenario was composed of a number of reference nodes with known coordinates trying to localize a node with unknown coordinates using only imprecise distance measurements. The results showed that :

1. The use of an accurate error model for the maximum likelihood brings non-negligible increase in performance over simpler model at the cost of a more difficult optimization problem.

2. However, the nature itself of the drifts makes the localization an ill-posed problem and even an accurate model cannot compensate for this. Thus, something must be done about the drifts before performing the localization.

3. When there is no drift of the clocks, simpler algorithms like the distance error function based maximum likelihood can achieve a better performance than the accurate SS-TWR error function at low SNR while being asymptotically equal. This could come from the better properties of the distance error function (convex).

4. Even so, the number of reference nodes required to achieve an acceptable performance is too high. Using a synchronization protocol dedicated to ranging could alleviate this problem.

Therefore, before doing any future work, it might be useful to define a precise framework with regard to a concrete problem. This could allow to use some custom solutions depending on the problem considered. In addition to this the following work could be done :

- Try to use the synchronization protocol dedicated to ranging that is defined in the 802.15.4a standard.

- Replace the accurate probability density function in the maximum likelihood error model by an approximation with better optimization properties. Good candidates for this could be the distance error function or a mixture of Gaussian.

- Study more in depth the optimization problem. It might turn out that an equivalent problem can be solved with better performance.

- Study more precisely the behavior of DS-TWR.

- Investigate the use of drift correction.

References

[1] R. Scheibler, "Ranging in IR-UWB Networks : Survey and Modelling of Synchronization Error." Sem. Proj., June 2008.

[2] S. Krishnan, "Clock inaccuracies in UWB Ranging." Sem. Proj., June 2008.

[3] Y.-T. Chan, H. Y. C. Hang, and P.-C. Ching, "Exact and Approximate Maximum Likelihood Localization Algorithms," *IEEE Transactions on Vehicular Technology*, vol. 55, pp. 10–16, January 2006.

[4] J. Caffery Jr., "A new approach to the geometry of TOA location," *Vehicular Technology Conference, 2000. IEEE VTS-Fall VTC 2000. 52nd*, vol. 4, pp. 1943–1949, 2000.

[5] F. Gardiol, *Électromagnétisme*, vol. 3 of *Traité d'Électricité*. Lausanne: PPUR, 2002.

[6] Federal Communications Commission, *Revision of Part 15 of the commission's rules regarding ultra-wideband transmission systems*, April 2002. ET-Docket 98-153, pp. 64, 73.

[7] K. Kiminami, A. Hirata, and T. Shiozawa, "Double-sided printed bow-tie antenna for uwb communications," *Antennas and Wireless Propagation Letters, IEEE*, vol. 3, pp. 152–153, 2004.

[8] Z. Low, J. Cheong, and C. Law, "Low-cost pcb antenna for uwb applications," *Antennas and Wireless Propagation Letters, IEEE*, vol. 4, pp. 237–239, 2005.

[9] S. H. Choi, J. K. Park, S. K. Kim, and J. Y. Park, "A new ultra-wideband antenna for uwb applications," *Microwave and Optical Technology Letters*, vol. 40, pp. 399–401, March 2004.

[10] M. Shen, T. Koivisto, T. Peltonen, L.-R. Zheng, E. Tjukanoff, and H. Tenhunerf, "Uwb radio module design for wireless sensor networks," *NORCHIP Conference, 2005. 23rd*, pp. 184–187, Nov. 2005.

[11] N. Patwari, J. N. Ash, S. Kyperountas, A. O. Hero III, R. L. Moses, and N. S. Correal, "Locating the Nodes : Cooperative localization in wireless sensor networks," *Signal Processing Magazine, IEEE*, vol. 22, pp. 54–69, July 2005.

[12] F. Gustafsson and F. Gunnarsson, "Mobile positioning using wireless networks: possibilities and fundamental limitations based on available wireless network measurements," *Signal Processing Magazine, IEEE*, vol. 22, pp. 41–53, July 2005.

[13] A. Sayed, A. Tarighat, and N. Khajehnouri, "Network-based wireless location: challenges faced in developing techniques for accurate wireless location information," *Signal Processing Magazine, IEEE*, vol. 22, pp. 24–40, July 2005.

[14] B. Fang, "Simple solutions for hyperbolic and related position fixes," *Aerospace and Electronic Systems, IEEE Transactions on*, vol. 26, pp. 748–753, Sep 1990.

[15] J. Abel, "A divide and conquer approach to least-squares estimation," *Aerospace and Electronic Systems, IEEE Transactions on*, vol. 26, pp. 423–427, Mar 1990.

[16] F. Sivrikaya and B. Yener, "Time synchronization in sensor networks: a survey," *Network, IEEE*, vol. 18, pp. 45–50, July-Aug. 2004.

[17] D. Moore, J. Leonard, D. Rus, and S. Teller, "Robust distributed network localization with noisy range measurements," in *Proceedings of the Second ACM Conference on Embedded Networked Sensor Systems (SenSys '04), Baltimore, MD*, pp. 50–61, November 2004.

[18] N. B. Priyantha, H. Balakrishnan, E. Demaine, and S. Teller, "Anchor-Free Distributed Localization in Sensor Networks." LCS Tech. Report #892, April 2003.

[19] H. Fan and C. Yan, "Asynchronous Differential TDOA for Sensor Self-Localization," *Acoustics, Speech and Signal Processing, 2007. ICASSP 2007. IEEE International Conference on*, vol. 2, pp. II–1109–II–1112, April 2007.

[20] B. Denis, J.-B. Pierrot, and C. Abou-Rjeily, "Joint Distributed Synchronization and Positioning in UWB Ad Hoc Networks Using TOA," *IEEE Transactions on Microwave Theory and Techniques*, vol. 54, pp. 1896–1911, April 2006.

A SS-TWR measurement error approximate pdf

As explained in Section 2.2, the SS-TWR measurement error can be approximed as :

$$E = T_1 + T_2 + U$$

where T_1 and T_2 are two independent variables drawn from the mixture distribution with pdf :

$$f_T(t) = w_1 f_g(t) + w_2 f_e(t) + w_3 f_u(t) \tag{5}$$

with :

$$f_g(t) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{t^2}{2\sigma^2}} \qquad f_e(t) = \begin{cases} \lambda e^{-\lambda t} & \text{if } t \geq 0 \\ 0 & \text{if } t < 0 \end{cases}.$$

$$f_u(x) = \begin{cases} \frac{1}{\tau_{bs}} & \text{if } x \in [-\tau_{bs}; 0] \\ 0 & \text{o.w.} \end{cases}$$

and $U \sim \mathcal{U}[a; b]$ and is independent from T_1 and T_2. Since we have a sum of random variables, the result is given by the convolution of the pdf :

$$f_E(x) = f_T(x) * f_T(x) * f_U(x) = \frac{1}{b-a} \int_{x-b}^{x-a} (f_T(t) * f_T(t))\, dt$$

The details of the computations is not given here, but the resulting pdf is also a mixture model with six components.

$$c_1(x) = \frac{1}{b-a} \left(F_g(x-a) - F_g(x-b)\right)$$

$$\begin{aligned} c_2(x) = \frac{1}{b-a} \Big[& \left(1 + \max\{x-a; 0\}\right) e^{-\lambda \max\{x-a;0\}} \\ & - \left(1 + \max\{x-b; 0\}\right) e^{-\lambda \max\{x-b;0\}} \Big] \end{aligned}$$

$$
c_3(x) = \begin{cases}
\frac{1}{b-a}\frac{1}{\tau_{bs}^2}\left[\frac{1}{2}\left((x-a)^2 - \max\{x-b,-2\tau_{bs}\}^2\right)\right. & \\
\quad \left. + 2\tau_{bs}\left(x-a-\max\{x-b;-2\tau_{bs}\}\right)\right] & \text{if } a-2\tau_{bs} \le x \le a-\tau_{bs} \\
\frac{1}{b-a}\left(1-\frac{1}{\tau_{bs}^2}\left[\frac{1}{2}\left(\max\{x-b;-2\tau_{bs}\}^2 + \min\{x-a;0\}^2\right)\right.\right. & \\
\quad \left.\left. + 2\tau_{bs}\left(\tau_{bs}+\max\{x-b;-2\tau_{bs}\}\right)\right]\right) & \text{if } a-\tau_{bs} \le x \le b-\tau_{bs} \\
\frac{1}{b-a}\frac{1}{\tau_{bs}^2}\frac{1}{2}\left((x-b)^2 - \min\{x-a;0\}^2\right) & \text{if } b-\tau_{bs} \le x \le b \\
0 & \text{o.w.}
\end{cases}
$$

$$
\begin{aligned}
c_4(x) &= \frac{1}{b-a}\left[e^{-\lambda(x-\sigma^2\lambda-b)}F_g(x-\sigma^2-b) - e^{-\lambda(x-\sigma^2\lambda-a)}F_g(x-\sigma^2-a)\right] \\
&\quad + \frac{1}{b-a}e^{\frac{\sigma^2\lambda^2}{2}}\left(F_g(x-a) - F_g(x-b)\right)
\end{aligned}
$$

$$
\begin{aligned}
c_5(x) &= \frac{1}{\tau_{bs}}\frac{1}{b-a}\left[(x+\tau_{bs})F_g(x+\tau_{bs}-a) - (x+\tau_{bs}-b)F_g(x+\tau_{bs}-b)\right. \\
&\quad \left. - \sigma^2\left(f_g(x+\tau_{bs}-b) - f_g(x+\tau_{bs}-a)\right)\right] \\
&\quad - \frac{1}{\tau_{bs}}\frac{1}{b-a}\left[(x-a)F_g(x-a) - (x-b)F_g(x-b)\right. \\
&\quad \left. - \sigma^2\left(f_g(x-b) - f_g(x-a)\right)\right]
\end{aligned}
$$

$$
\begin{aligned}
c_6(x) &= \frac{1}{\tau_{bs}}\frac{1}{b-a}\left(\max\{x+\tau_{bs}-a;0\} - \max\{x+\tau_{bs}-b;0\}\right) \\
&\quad + \frac{1}{\tau_{bs}}\frac{1}{b-a}\frac{1}{\lambda}\left(e^{-\lambda\max\{x+\tau_{bs}-a;0\}} - e^{-\lambda\max\{x+\tau_{bs}-b;0\}}\right) \\
&\quad - \frac{1}{\tau_{bs}}\frac{1}{b-a}\left(\max\{x-a;0\} - \max\{x-b;0\}\right) \\
&\quad - \frac{1}{\tau_{bs}}\frac{1}{b-a}\frac{1}{\lambda}\left(e^{-\lambda\max\{x-a;0\}} - e^{-\lambda\max\{x-b;0\}}\right)
\end{aligned}
$$

And finally we sum up everything :

$$
f_E(x) = w_1^2c_1(x) + w_2^2c_2(x) + w_3^2c_3(x) + 2w_1w_2c_4(x) + 2w_1w_3c_5(x) + w_2w_3c_5(x)
$$

B Optimization algorithm derivation

After creating the mathematical model for the synchronization offset, the problem was to optimize its parameters according to the data sets corresponding to different SNR. The model is a mixture of three distributions :

$$\begin{aligned} f_E(x) &= w_1 \frac{[F_g(x-\mu_1+1) - F_g(x-\mu_1-1)]}{2} \\ &\quad + w_2 \frac{[F_e(x-\mu_2+1) - F_e(x-\mu_2-1)]}{2} \\ &\quad + w_3 f_u(t) \end{aligned}$$

where :

$$F_g(x) = \frac{1}{\sqrt{2\pi}\sigma} \int_{-\infty}^{x} e^{-\frac{t^2}{2\sigma^2}} dt \qquad F_e(x) = \begin{cases} 1 - e^{-\lambda t} & \text{if } x \geq 0 \\ 0 & \text{if } x < 0 \end{cases}$$

and

$$f_u(x) = \begin{cases} \frac{1}{\tau_{bs}} & \text{if } x \in [-\tau_{bs}; 0] \\ 0 & \text{o.w.} \end{cases}$$

and since $f_E(x)$ is a pdf we have the following counstraint :

$$w_1 + w_2 + w_3 = 1$$

The standard way of optimizing the parameters of a mixture model is the Expectation Maximization (EM) algorithm. However, in this case, the standard EM algorithm for mixtures could not be applied for all the parameters because of the different natures of the components in the mixture. Therefore, the EM algorithm was used to optimize the weights in conjunction with a gradient descent to optimize the other parameters.

Let's begin by describing the notation :

- $\boldsymbol{\theta}$ is a the parameters vector containing the weights, σ, λ, μ_1 and μ_2.
- The data set is $\boldsymbol{y} = \{y_i\}_{i=1}^{N}$.
- We denote by $f_g(x)$ and $f_e(x)$ the pdf of respectively a Gaussian and an exponential random variable.

EM step The EM update equations for a mixture model are well-known. First we need to compute the posterior probability that the i^{th} sample was generated by the j^{th} component of the mixture :

$$\hat{w}_{i,j} = p\left(z_j = i \mid y_j, \boldsymbol{\theta}\right) = \frac{p\left(y_j \mid z_j = i, \boldsymbol{\theta}\right) p\left(z_j = i \mid \boldsymbol{\theta}\right)}{p\left(y_j \mid \boldsymbol{\theta}\right)}$$

where z_j is a random variable representing the component of the mixture by which sample y_j was generated. We can now write this for our distribution :

$$\begin{aligned}
\hat{w}_{1,j} &= \frac{w_1 \frac{1}{2}\left[F_g\left(y_j - \mu_1 + 1\right) - F_g\left(y_j - \mu_1 - 1\right)\right]}{f_E(y_j|\boldsymbol{\theta})} \\
\hat{w}_{2,j} &= \frac{w_2 \frac{1}{2}\left[F_e\left(y_j - \mu_1 + 1\right) - F_e\left(y_j - \mu_1 - 1\right)\right]}{f_E(y_j|\boldsymbol{\theta})} \\
\hat{w}_{3,j} &= \frac{w_3 f_u(y_j)}{f_E(y_j|\boldsymbol{\theta})}
\end{aligned}$$

And finally, the updated weight is given by :

$$w_i^{(new)} = \frac{1}{N}\sum_{j=1}^{N} \hat{w}_{i,j} \qquad ; \qquad i = 1, 2, 3$$

Gradient ascent step At the same time we optimize the weights using EM we want to optimize the other parameters with respect to the log-likelihood :

$$l\left(\boldsymbol{\theta}, \boldsymbol{y}\right) = \sum_{i=1}^{N} \log f_E(y_i|\boldsymbol{\theta})$$

In order to achieve this, a simple gradient ascent method is used. Every time we update the waits, we do a step in the direction of the positive gradient :

$$\boldsymbol{\theta}^{(new)} = \boldsymbol{\theta} + \nu \vec{\nabla}_{\boldsymbol{\theta}} l\left(\boldsymbol{\theta}, \boldsymbol{y}\right)$$

where ν is the step size. The derivate with respect to a single parameter can be expressed as follows :

$$\frac{\partial}{\partial \theta_k} l\left(\boldsymbol{\theta}, \boldsymbol{y}\right) = \sum_{i=0}^{N} \frac{\frac{\partial}{\partial \theta_k} f_E(y_i|\boldsymbol{\theta})}{f_E(y_i|\boldsymbol{\theta})}$$

First some intermediate results are derived.

$$\frac{\partial F_g(x)}{\partial \sigma} = \frac{\partial}{\partial \sigma}\left\{\frac{1}{\sqrt{2\pi}\sigma}\int_{-\infty}^{x} e^{-\frac{(t-\mu)^2}{2\sigma^2}}\, dt\right\}$$

$$\overset{(a)}{=} -\frac{1}{\sigma}\frac{1}{\sqrt{2\pi}\sigma}\int_{-\infty}^{x} e^{-\frac{(t-\mu)^2}{2\sigma^2}}\,dt + \frac{1}{\sqrt{2\pi}\sigma}\int_{-\infty}^{x}\frac{(t-\mu_1)^2}{\sigma^3}e^{-\frac{(t-\mu)^2}{2\sigma^2}}\,dt$$

$$= -\frac{1}{\sigma}F_g(x) + \frac{1}{\sqrt{2\pi}\sigma}\int_{-\infty}^{x}\left(-\frac{t-\mu}{\sigma}\right)\left(-\frac{t-\mu}{\sigma^2}\right)e^{-\frac{(t-\mu)^2}{2\sigma^2}}\,dt$$

$$\overset{(b)}{=} -\frac{1}{\sigma}F_g(x) + \frac{1}{\sqrt{2\pi}\sigma}\left[\frac{\mu-t}{\sigma}e^{-\frac{(t-\mu)^2}{2\sigma^2}}\Big|_{-\infty}^{x} + \frac{1}{\sigma}\int_{-\infty}^{x} e^{-\frac{(t-\mu)^2}{2\sigma^2}}\,dt\right]$$

$$= -\frac{1}{\sigma}F_g(x) - \frac{x-\mu}{\sigma}f_g(x) + \frac{1}{\sigma}F_g(x) = -\frac{x-\mu}{\sigma}f_g(x)$$

where the product derivation formula was used at (a) and integration by part at (b). Similarly, we can derive the gradient of $F_g(x)$ with respect to μ.

$$\frac{\partial F_g(x)}{\partial \mu} = \frac{\partial}{\partial \mu}\left\{\frac{1}{\sqrt{2\pi}\sigma}\int_{-\infty}^{x} e^{-\frac{(t-\mu)^2}{2\sigma^2}}\,dt\right\}$$

$$= \frac{1}{\sqrt{2\pi}\sigma}\int_{-\infty}^{x}\left(\frac{t-\mu}{\sigma^2}\right)e^{-\frac{(t-\mu)^2}{2\sigma^2}}\,dt$$

$$= \frac{1}{\sqrt{2\pi}\sigma}\int_{-\infty}^{x}\left(-\frac{\partial}{\partial t}\left\{e^{-\frac{(t-\mu)^2}{2\sigma^2}}\right\}\right)dt$$

$$= -\frac{1}{\sqrt{2\pi}\sigma}e^{-\frac{(t-\mu)^2}{2\sigma^2}} = -f_g(x)$$

Finally, we easily compute the derivative of $F_e(x)$ with respect to μ and λ :

$$\frac{\partial F_e(x)}{\partial \mu} = -f_e(x)$$

where $f_e(x)$ is the pdf of an exponential random variable. Finally :

$$g(x) = \frac{\partial F_e(x)}{\partial \lambda} = \begin{cases}(x-\mu)e^{-\lambda(x-\mu)} & \text{if } x \geq \mu \\ 0 & \text{if } x < \mu\end{cases}$$

Now using all the results derived before, we easily get the gradient of $l(\boldsymbol{\theta}, \boldsymbol{y})$.

$$\frac{\partial}{\partial \mu_1}l(\boldsymbol{\theta}, \boldsymbol{y}) = \sum_{i=1}^{N}\frac{w_1\frac{1}{2}(-f_g(y_i+1) + f_g(y_i-1))}{f_E(y_i)}$$

$$\frac{\partial}{\partial\sigma}l(\boldsymbol{\theta},\boldsymbol{y}) = \sum_{i=1}^{N} \frac{w_1 \frac{1}{2}\left(-\frac{y_i+1-\mu_1}{\sigma} f_g(y_i+1) + \frac{y_i-1-\mu_1}{\sigma} f_g(y_i-1)\right)}{f_E(y_i)}$$

$$\frac{\partial}{\partial\mu_2}l(\boldsymbol{\theta},\boldsymbol{y}) = \sum_{i=1}^{N} \frac{w_2 \frac{1}{2}(-f_e(y_i+1) + f_e(y_i-1))}{f_E(y_i)}$$

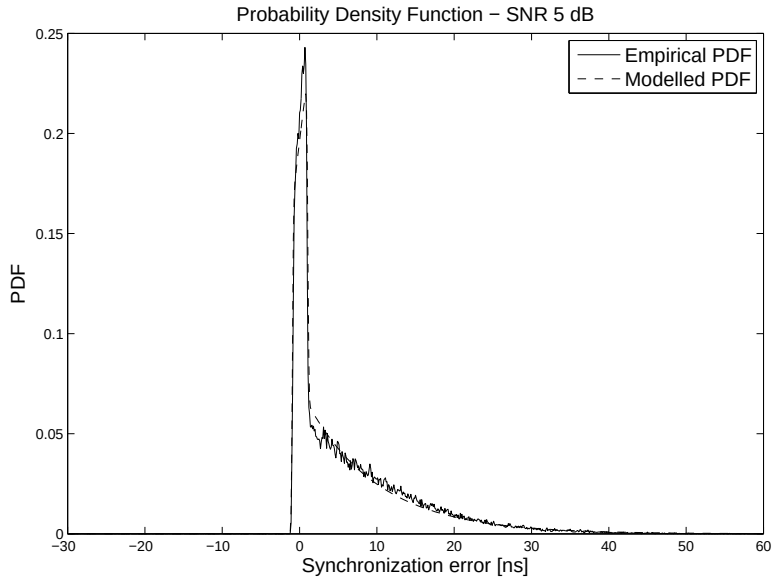
$$\frac{\partial}{\partial\lambda}l(\boldsymbol{\theta},\boldsymbol{y}) = \sum_{i=1}^{N} \frac{w_2 \frac{1}{2}\left(g(y_i+1) - g(y_i-1)\right)}{f_E(y_i)}$$

If we reduce μ_1 and μ_2 to a single parameter $\mu = \mu_1 = \mu_2$, then we have :

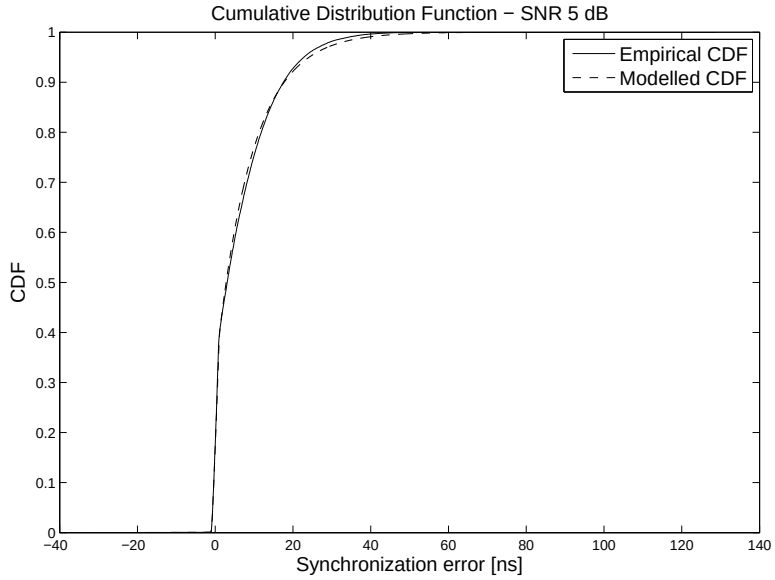
$$\frac{\partial}{\partial\mu}l(\boldsymbol{\theta},\boldsymbol{y}) = \sum_{i=1}^{N} \frac{w_1 \frac{1}{2}(-f_g(y_i+1) + f_g(y_i-1)) + w_2 \frac{1}{2}(-f_e(y_i+1) + f_e(y_i-1))}{f_E(y_i)}$$

C Model parameters values

The standard deviation of the Gaussian is set to $\sigma = 0.115$ and only a single non-centrality parameter was kept (i.e. $\mu = \mu_1 = \mu_2$). The value of the other parameters for single user communication can be found in Table 1 while for the case where a powerful interferer is present they can be found in Table 2. Those parameters corresponds to a synchronization offset in nanoseconds. Samples produced using the model with those parameters should be scaled by 10^{-9} to be in seconds.

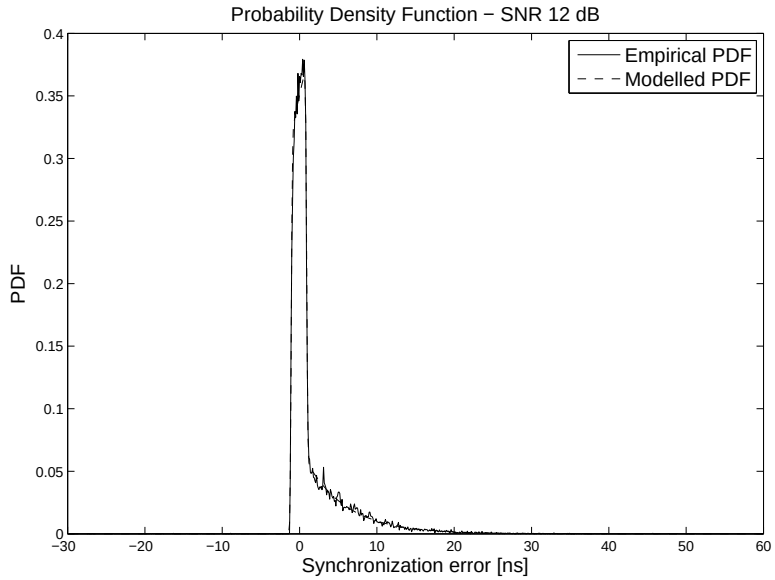


(a)

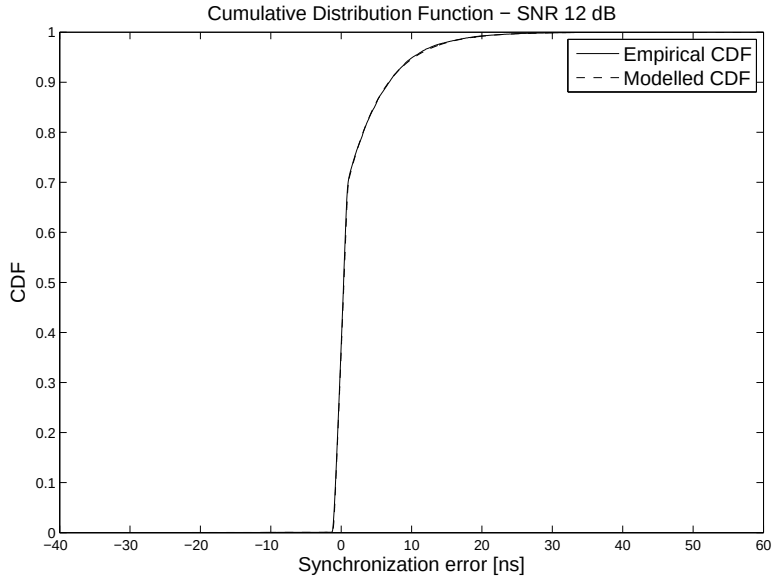


(b)

Figure 5: The optimized model (dashed line) plotted along its empirical counterpart (plain line) at an SNR of 5 dB. (a) pdf (b) cdf. Single user communication.

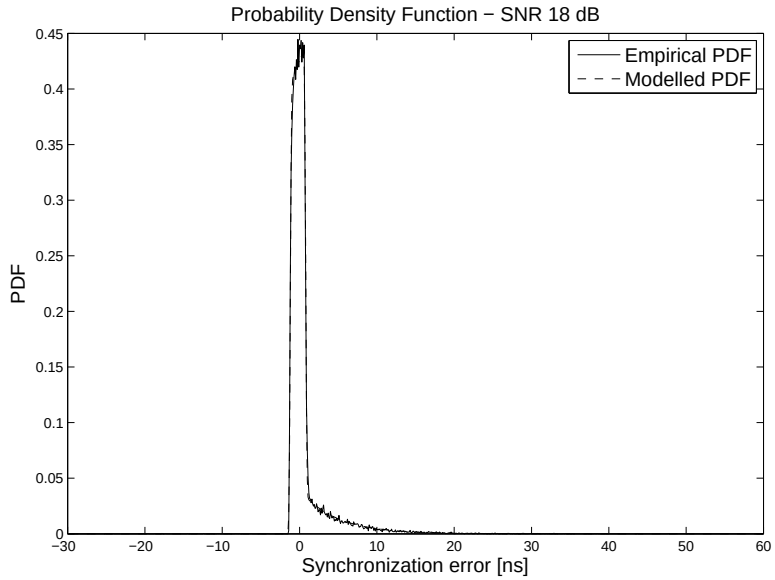


(a)

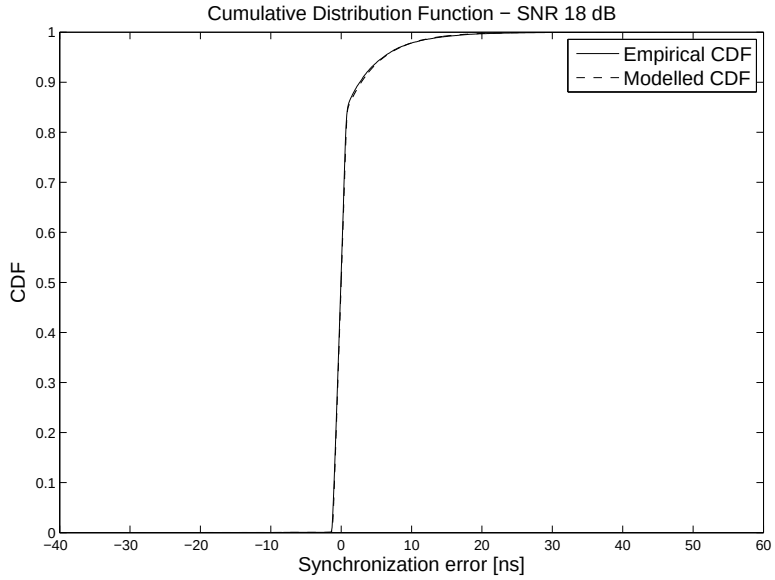


(b)

Figure 6: The optimized model (dashed line) plotted along its empirical counterpart (plain line) at an SNR of 12 dB. (a) pdf (b) cdf. Single user communication.



(a)



(b)

Figure 7: The optimized model (dashed line) plotted along its empirical counterpart (plain line) at an SNR of 18 dB. (a) pdf (b) cdf. Single user communication.

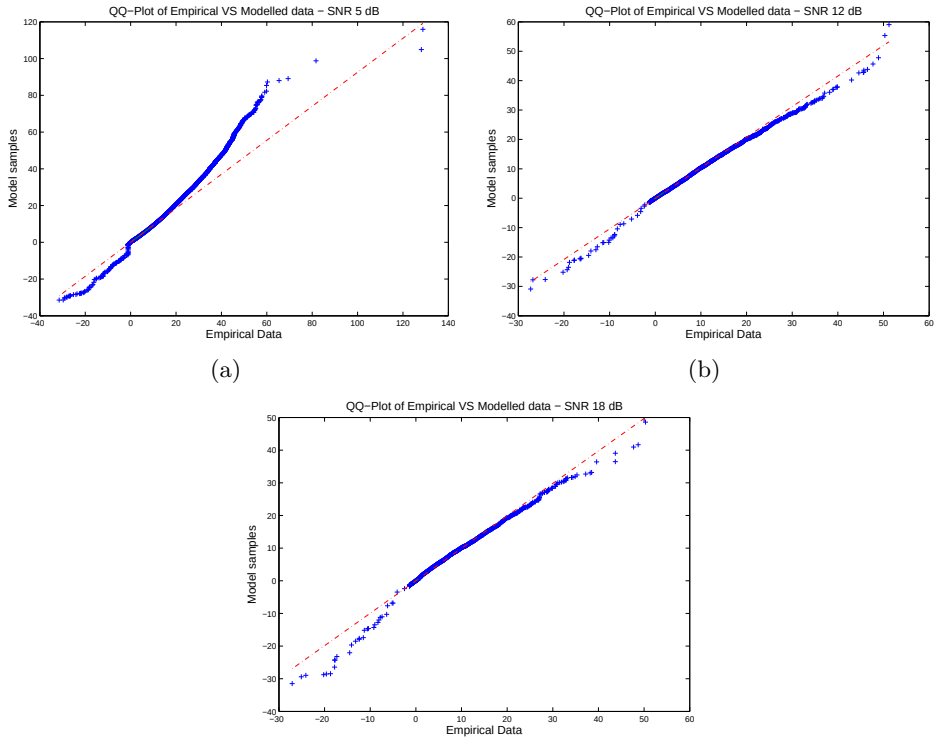


Figure 8: QQplots with empirical samples on the x-axis and samples generated from the model on the y-axis at SNR (a) 5 dB (b) 12 dB and (c) 18 dB. Single user communication.

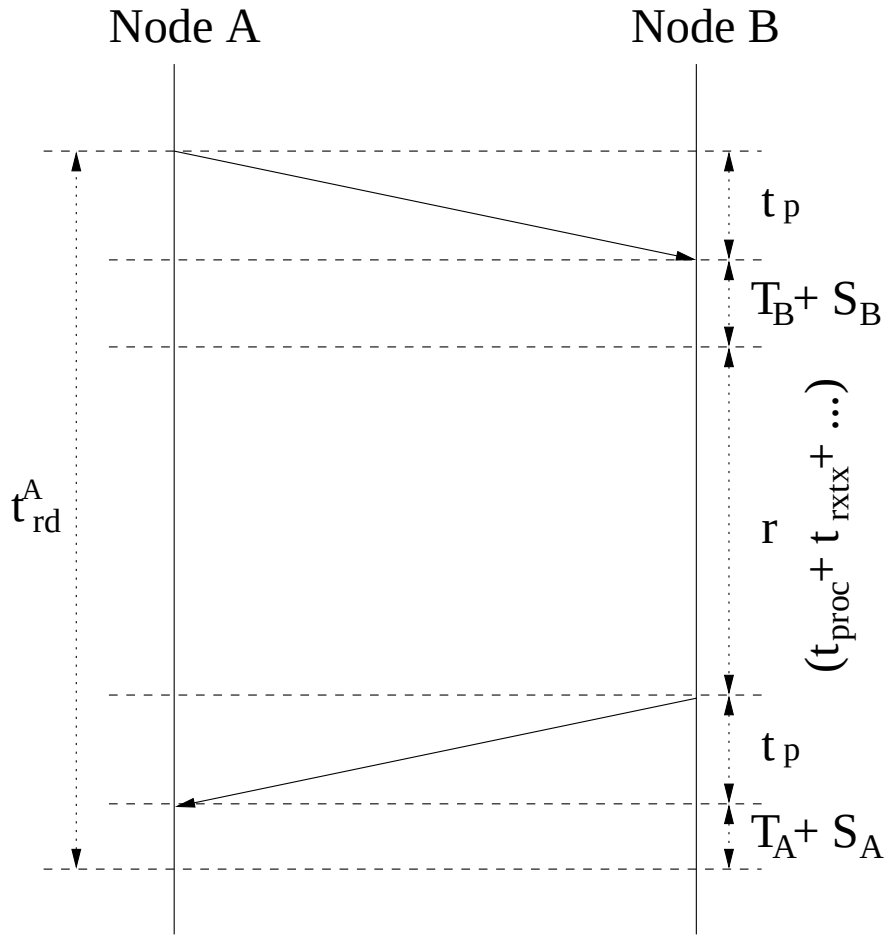


Figure 9: Illustration of SS-TWR transactions.

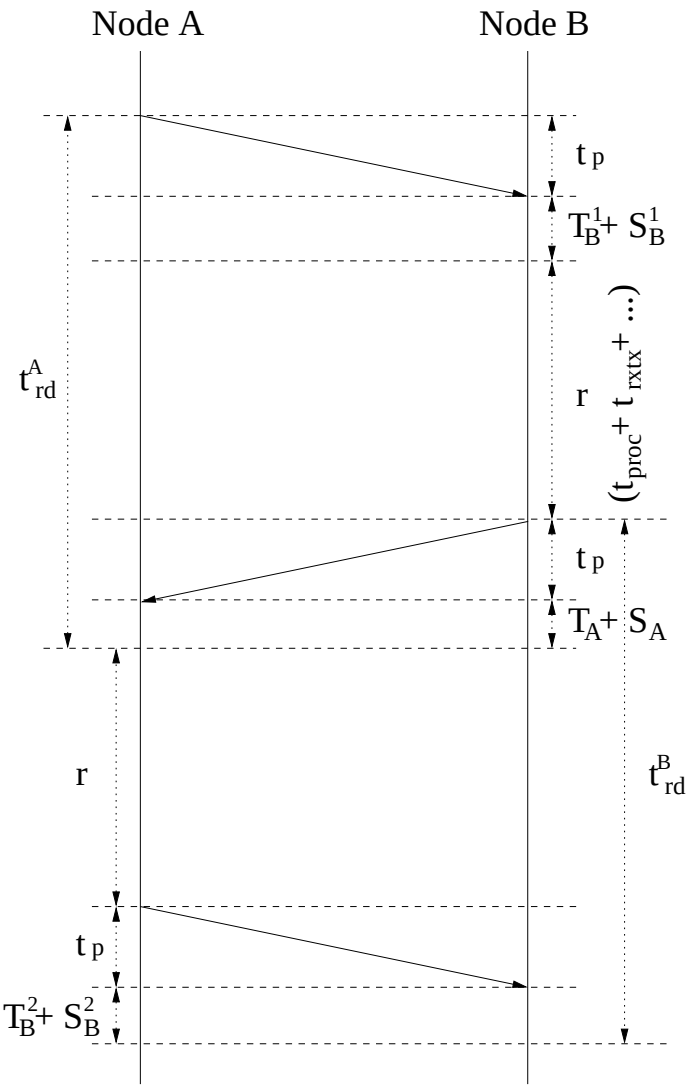
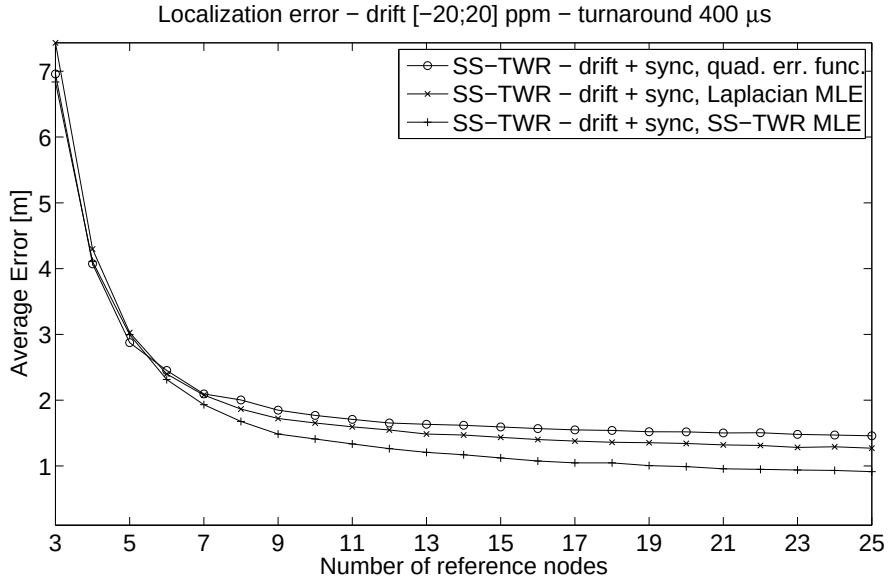
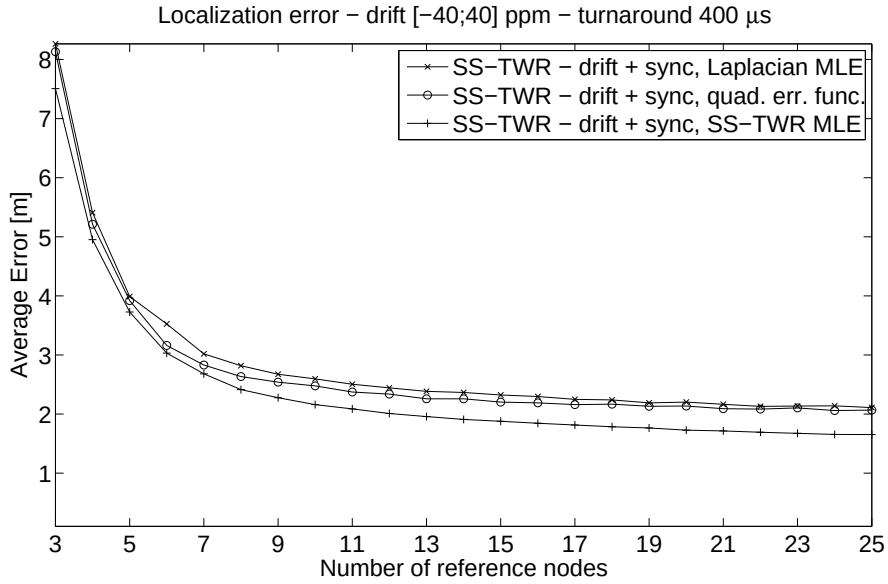


Figure 10: Illustration of DS-TWR transactions.

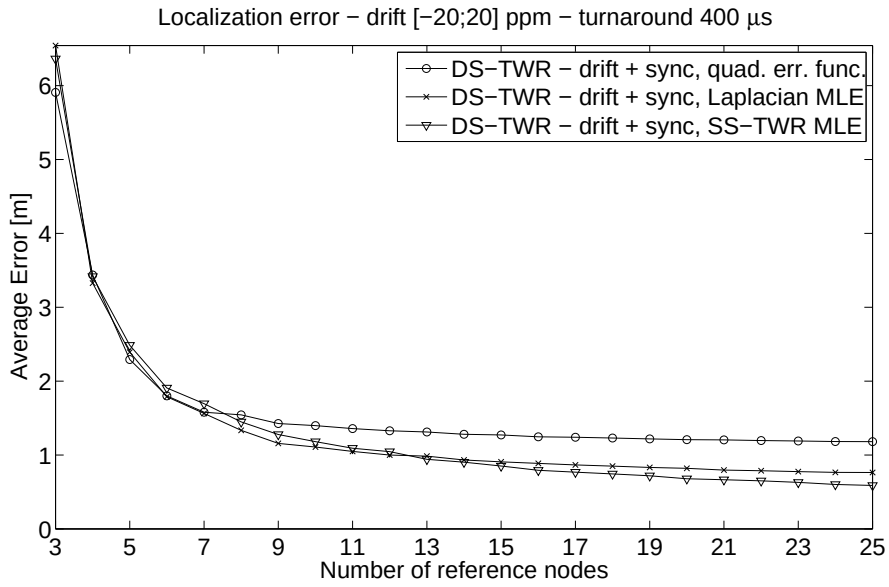


(a)

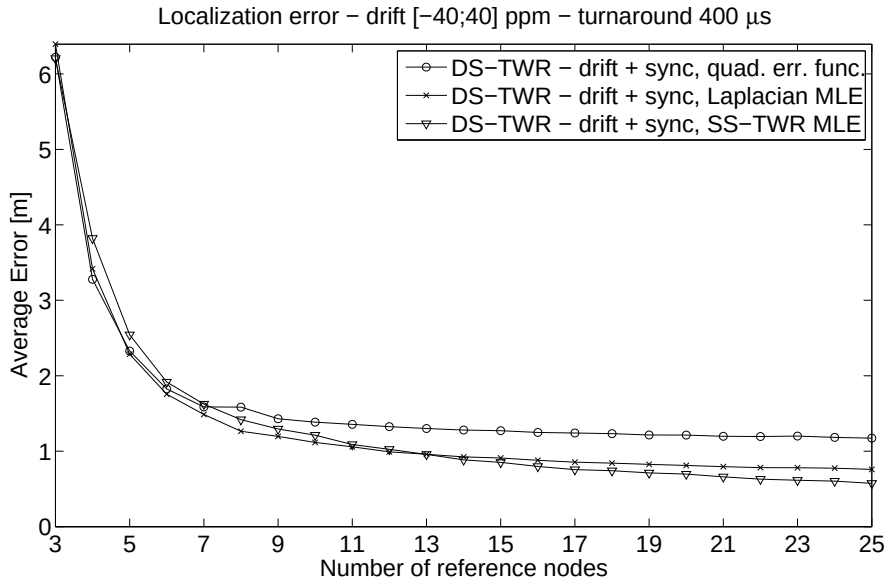


(b)

Figure 11: Simulation results for SS-TWR transactions with the errors due to the drifts and the synchronization.



(a)



(b)

Figure 12: Simulation results for DS-TWR transactions with the errors due to the drifts and the synchronization.

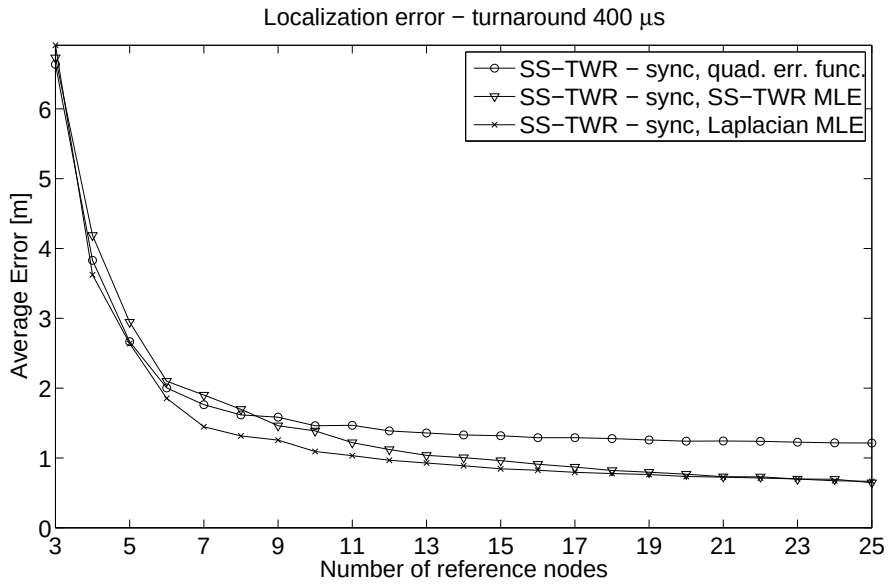


Figure 13: Simulation results for SS-TWR transactions with the errors due to the synchronization only.

SNR [dB]	w_1	w_2	w_3	λ	μ
0	0.5702	0.4288	0.0011	0.1354	0.2357
1	0.4823	0.5165	0.0012	0.1309	0.2083
2	0.4185	0.5809	0.0006	0.1261	0.1724
3	0.3639	0.6352	0.0008	0.1169	0.1334
4	0.3343	0.6651	0.0006	0.1101	0.0994
5	0.3285	0.6707	0.0008	0.1078	0.0725
6	0.3477	0.6515	0.0008	0.1099	0.0520
7	0.3878	0.6113	0.0009	0.1188	0.0275
8	0.4373	0.5620	0.0007	0.1331	0.0071
9	0.4917	0.5076	0.0007	0.1489	-0.0135
10	0.5422	0.4571	0.0007	0.1643	-0.0351
11	0.5904	0.4090	0.0006	0.1796	-0.0565
12	0.6342	0.3651	0.0008	0.1903	-0.0762
13	0.6723	0.3268	0.0009	0.1989	-0.0998
14	0.7064	0.2929	0.0007	0.2047	-0.1201
15	0.7357	0.2635	0.0008	0.2107	-0.1395
16	0.7623	0.2368	0.0009	0.2132	-0.1608
17	0.7873	0.2118	0.0009	0.2137	-0.1800
18	0.8083	0.1910	0.0007	0.2143	-0.1995
19	0.8273	0.1719	0.0008	0.2130	-0.2188
20	0.8429	0.1563	0.0008	0.2120	-0.2366
21	0.8570	0.1422	0.0008	0.2100	-0.2567
22	0.8703	0.1289	0.0008	0.2099	-0.2731
23	0.8825	0.1168	0.0007	0.2049	-0.2903
24	0.8904	0.1088	0.0008	0.2055	-0.3091
25	0.9000	0.0993	0.0007	0.1998	-0.3255

Table 1: Optimal values of the parameters of the model for different SNR points with a single user.

SNR [dB]	w_1	w_2	w_3	λ	μ
0	0.5688	0.4277	0.0035	0.1366	0.2650
1	0.5031	0.4910	0.0059	0.1316	0.1948
2	0.4258	0.5664	0.0079	0.1255	0.1555
3	0.3769	0.6083	0.0149	0.1174	0.1291
4	0.3409	0.6309	0.0282	0.1130	0.0939
5	0.3268	0.6250	0.0482	0.1101	0.0582
6	0.3314	0.5908	0.0779	0.1119	0.0403
7	0.3519	0.5396	0.1085	0.1203	0.0244
8	0.3772	0.4855	0.1373	0.1344	0.0025
9	0.4079	0.4353	0.1568	0.1507	-0.0151
10	0.4435	0.3869	0.1696	0.1649	-0.0366
11	0.4787	0.3451	0.1762	0.1757	-0.0537
12	0.5116	0.3080	0.1804	0.1849	-0.0734
13	0.5389	0.2777	0.1834	0.1914	-0.0959
14	0.5674	0.2478	0.1849	0.1935	-0.1158
15	0.5928	0.2238	0.1834	0.1974	-0.1367
16	0.6131	0.2018	0.1851	0.1988	-0.1554
17	0.6325	0.1821	0.1854	0.1957	-0.1736
18	0.6488	0.1654	0.1859	0.1958	-0.1918
19	0.6630	0.1522	0.1848	0.1936	-0.2114
20	0.6748	0.1388	0.1863	0.1915	-0.2293
21	0.6859	0.1273	0.1868	0.1875	-0.2469
22	0.6960	0.1179	0.1862	0.1838	-0.2631
23	0.7050	0.1082	0.1868	0.1802	-0.2821
24	0.7122	0.1015	0.1864	0.1769	-0.2979
25	0.7197	0.0945	0.1858	0.1742	-0.3137

Table 2: Optimal values of the parameters of the model for different SNR points with two users.