# Adaptive time mismatches identification and correction in polar transmitter architecture

J.-F. Bercher and C. Berland

ESYCOM-ESIEE, Université de Paris-Est Cité Descartes, BP99, 93162 Noisy-le-Grand Cedex France { jf.bercher,c.berland}@esiee.fr

*Abstract*— This paper deals with synchronization in EER type of transmitter architecture, illustrated in the context of 3GPP standard. Firstly, we point out the sensitivity of such architecture to delay mismatch between the two signals path. This sensitivity can be observed on EVM and output spectrum performances. To correct this default, two algorithms are studied. Both are based on the comparison between the emitted signal and the digitally generated one but with the difference that the first algorithm needs I/Q demodulation whereas the second one only needs the inphase component of the transmitted signal. We demonstrate that the quality of the interpolator used is crucial. Implementation details are given. We present simulation results realised with HPADS and draw attention to the enhancement of the transmitter performances using such algorithm.

# I. INTRODUCTION

Polar transmitter is a solution to achieve a multi mode, multi standard efficient transmitter. This architecture based on EER Kahn transmitter was improved so that actual solution tends toward a quasi fully digital transmitter [1]. Instead of using both analog limiter and envelope detector to generate envelope and phase modulated signals, a CORDIC processor (Coordinate Rotation Digital Computer) is implemented [2] as shown Fig. 1. The phase-modulated signal is commonly translated to RF frequency using a modulated phase locked loop, either digital or analog. The envelope is still restored at the output of the transmitter while varying the supply of the power amplifier (PA) through a switching power supply.



Fig. 1. Polar transmitter using a CORDIC processor

An important drawback of this architecture is its sensibility to the difference of propagation time between amplitude and phase signals paths [4]. This mismatch becomes more and more limiting as data rate increases.

# II. IMPACT OF DELAY MISMATCHES ON A 16QAM MODULATION

In this article, we focus on 16QAM modulation with a data rate of 3.84Mcps. This corresponds to the 3GPP standard [3]. ACLR for this modulation has to remain below -33dBc for adjacent channel frequencies at  $\pm$ 5MHz from the assigned channel frequency, and -43dBc for adjacent channel frequencies at  $\pm$ 10MHz from the assigned channel frequency. The output spectrum is specified as relative level from the power of the transmitted signal at different frequency offset from the carrier. Specification are as follows:

- at 2.5MHz, -35dBc in 30kHz bandwidth,
- at 3.485MHz, -49.8dBc in 30kHz bandwidth,
- at 4MHz, -35.5dBc in 1MHz bandwidth,
- for F>8MHz, -49dBc in 1MHz bandwidth.

The more limiting point is here -49.8dBc in 30kHz bandwidth at 3.485MHz. The EVM has to remain lower than 17.5%. The modelisation of the transmitter is realised on HPADS. Envelope and phase signals are generated from the output of the classical 16QAM modulator. In order to simulate non integer delay mismatches, two sample and hold were added, followed by two baseband filters. The delay mismatch implies spectral regrowths [4], as shown Fig 2.



Fig. 2. Relative Output spectrum for different delays, with a 30kHz bandwidth resolution

A limiting point of this specification is the measurement at 3.5MHz from the carrier. According to the specifications, an acceptable delay shall be of the order of  $T_s/30$ , that is to say lower than 9ns. This implies that a synchronisation algorithm is mandatory. In this paper we devise and analyse two algorithms based on a LMS principle for identification and correction of this delay.

# **III. ALGORITHMS AND STRUCTURES**

Two algorithms are presented here. The first one adaptively corrects the two delays induced by paths mismatches, while the second one only identifies these delays, but with a lower complexity. A preliminary version of the first algorithm was presented in [6]. The present version is more elaborated, detailed and enhanced in several aspects particularly concerning implementation issues and the analysis of performances.

# A. Envelope and phase alignment

For complex modulation schemes, the emitted signal x(t) can often be modelled as a complex gaussian process. In the case of a complex circular gaussian process it is well known that envelope and phase are independent and respectively distributed according to Rayleigh and uniform distributions. The case of delayed envelope and phase is less known. In fact, it appears that  $\rho(t)$  and  $\phi(t - \Delta)$  are also Rayleigh and uniform independent variables, with no reference to the correlation coefficient, for any mismatch  $\Delta$ , see [4]. The proof is as follows. Let  $x_1 = \rho_1 cos\phi_1$  and  $x_2 = \rho_1 cos\phi_2$  two gaussian random variables with common variance  $\sigma^2$  and correlation coefficient R, with  $R^2 = E[x_1x_2]/2\sigma^2$ . Then, the joint distribution  $p(\rho_1, \phi_1, \rho_2, \phi_2)$  between envelopes and phases is

$$\frac{\rho_1 \rho_2}{2\pi \sigma^2} \exp\left(-\frac{\rho_1^2 + \rho_2^2 + R\rho_1 \rho_2 \cos(\phi_1 - \phi_2)}{2\sigma^2 (1 - R^2)}\right).$$
(1)

The joint distribution between  $\rho_1$  and  $\phi_2$ , that is the joint distribution between envelope and phase at two distinct instants, say t and  $t - \Delta$ , is obtained by marginalization of (1) with respect to  $\rho_2$  and  $\phi_1: p(\rho_1, \phi_2) = \int \int p(\rho_1, \phi_1, \rho_2, \phi_2) d\rho_2 d\phi_1$ . Using the integral representation of modified Bessel functions of first kind, we first obtain  $p(\rho_1, \rho_2, \phi_2)$  as

$$\frac{\rho_1 \rho_2}{2\pi \sigma^2} \exp\left(-\frac{\rho_1^2 + \rho_2^2}{2\sigma^2 (1 - R^2)}\right) I_0\left(\frac{R\rho_1 \rho_2}{2\sigma^2 (1 - R^2)}\right).$$

Then, using the integration formula [11-4-29] in [5], together the relation between regular and modified Bessel functions, we arrive at

$$p(\rho_1, \phi_2) = \frac{1}{2\pi} \frac{\rho_1}{2\pi\sigma^2} \exp\left(-\frac{\rho_1^2}{2\sigma^2}\right) = p(\phi_2)p(\rho_1).$$

that does not depend on the correlation coefficient anymore and shows that  $\rho(t)$  and  $\phi(t - \Delta)$  are independent with Rayleigh and uniform distributions respectively. This result shows that for gaussian processes, envelope and phase are *always* independent with no reference to the correlation coefficient, whatever the delay between envelope and phase components.

Hence the output do not convey any information on the time alignment or mismatch between envelope and phase components. Therefore, it is *not* possible to correct the time mismatch from the sole observation of the system output. Any method will have to rely on a feedback loop. Such methods are devised in the following.<sup>1</sup>

# B. A correction algorithm

This first algorithm relies on a simple idea, illustrated in Fig. 3.



Fig. 3. Principle of delays correction.

Let x(t) denote the input of the system and z(t) its output. With  $\Delta_1$  the delay that affects the envelope path and  $\Delta_2$ the delay for the phase term, the output is  $z(t) = \rho(t - \Delta_1) \cos(\phi(t - \Delta_2))$ . Therefore, the simple principle is to introduce two advances  $\mu_1$  and  $\mu_2$  in order to precompensate the delays. In such a situation, the output becomes

$$z(t) = \rho(t + \mu_1 - \Delta_1) \cos(\phi(t + \mu_2 - \Delta_2)),$$

and one shall design a procedure to adjust  $\mu_1$  and  $\mu_2$  so that in fine  $\mu_1 = \Delta_1$ ,  $\mu_2 = \Delta_2$  and z(t) = x(t). For that purpose, we have to minimize some (statistical) distance between z(t)and x(t). A natural criterion is simply the quadratic distance between z(t) and  $x(t)^2$ :

$$J(\mu_1, \mu_2) = \mathbf{E} \left[ |x(t) - \rho(t_1) \cos(\phi(t_2))|^2 \right],$$

where we noted  $t_1 = t + \mu_1 - \Delta_1$  and  $t_2 = t + \mu_2 - \Delta_2$  in order to simplify expressions and where E[] is the statistical expectation operator. Taking into account the independance between  $\rho(t_1)$  and  $\phi(t_2)$ , it reduces to

$$J(\mu_1, \mu_2) = 4R(0, 0) - 4R(\mu_1 - \Delta_1, \mu_2 - \Delta_2)$$

where  $R(\tau_1, \tau_2) = E[\rho(t) \cos(\phi(t))\rho(t - \tau_1) \cos(\phi(t - \tau_2))]$ is a kind of 'correlation function' <sup>3</sup> The criterion  $J(\mu_1, \mu_2)$ was evaluated numerically by Monte-Carlo simulations in the case of a 16QAM modulation, with a square-root Nyquist filter (raised cosine with 0.22 roll-off). It is presented in Fig. 4 for delays lower than 1.5 symbol period. For more important delays (data not shown), other local minima appear. However, Fig. 4 clearly indicates that a descent algorithm will avoid local minima for delays  $\Delta_1, \Delta_2 \leq T_s$ , with  $T_s$  the symbol

<sup>3</sup>By Schwarz inequality, one can check that  $R(0,0) \ge |R(\tau_1,\tau_2)|$ .

<sup>&</sup>lt;sup>1</sup>However let us insist on the fact that this is true for gaussian processes, that are corrrect and convenient models for complex modulations. But one can also devise special non-gaussian training sequences in order to align envelope and phase components, by using only properties of the output.

 $<sup>^{2}</sup>$ Note that a similar criterion can also be built for the quadrature component, or for both components together.

period. It is possible to show that the behaviour of the error function is linked to the shaping filter h.



Fig. 4. Shape of criterion for a 16QAM

Since we do not have a closed-form for the criterion nor a direct explicit solution for its global minimizers, we need to exhibit the solution using a descent algorithm. We use a gradient algorithm that consists in iterating

$$\mu_1(n+1) = \mu_1(n) - \gamma_1(n) \left. \frac{\partial J}{\partial \mu_1} \right|_{\mu_1 = \mu_1(n)}$$

$$\mu_2(n+1) = \mu_2(n) - \gamma_2(n) \left. \frac{\partial J}{\partial \mu_2} \right|_{\mu_2 = \mu_2(n)}$$
(2)

where  $\gamma_1$  and  $\gamma_2$  are adaptation steps. The gradients are

$$\frac{\partial J(\mu_1,\mu_2)}{\partial \mu_1} = -2E \left[ \frac{d\rho(u)}{du} \Big|_{u=t_1} \cos\left(\phi(t_2)\right) e(t) \right]$$
  
$$\frac{\partial J(\mu_1,\mu_2)}{\partial \mu_2} = 2E \left[ \frac{d\phi(u)}{du} \Big|_{u=t_2} \rho(t_1) \sin\left(\phi(t_2)\right) e(t) \right]$$
(3)

with

$$e(t) = x(t) - \rho(t + \mu_1 - \Delta_1) \cos(\phi(t + \mu_2 - \Delta_2)).$$
 (4)

The update equations are obtained using these gradients in (2). But in practice, we have to resort to an approximation of these theoretical recursions, since we do not have analytical expressions of the statistical expectations involved. A solution is to adopt a stochastic algorithm. The LMS algorithm consists in using the *instantaneous gradient* rather than the (correct) statistical average, and in updating the equations at each new sample. This gives

$$\mu_{1}(n+1) = \mu_{1}(n) + \gamma_{1}(n) \frac{d\rho(u)}{du} \Big|_{t_{1}(n)} \cos\left(\phi(t_{2}(n))\right) e(t)$$
  

$$\mu_{2}(n+1) = \mu_{2}(n) + \gamma_{2}(n)\rho(t_{1}(n)) \frac{d\cos\phi(u)}{du} \Big|_{t_{2}(n)} e(t)$$
(5)

Practical implementation of formulas (5) involves

- computation of the error e(t) defined in (4) that is simply the comparison of the system input and output,
- computation of derivatives in (5) that can be simply approximated by finite differences of  $\rho(t)$  and  $\cos(\phi(t))$

For formulas (5), the computational load is of about 6 real multiplications per iteration. However, since the derivatives must be computed at time  $t_1$  and  $t_2$ , that is at the output of the corrected system,  $\rho(t)$  and  $\cos(\phi(t))$  must be available separately. This implies a quadrature demodulation before the feedback loop.

# C. An identification structure

An alternative approach, that consists in estimating the delays, relax this constraint. The structure is given in Fig. 5. The idea is to delay the envelope and phase of the input x(t) by  $\tau_1$  and  $\tau_2$ , compare this delayed version to the output, and then adjust the delays so as to minimize the quadratic error.



# Fig. 5. Identification architecture

Along the same lines as before, we end up with

$$\tau_1(n+1) = \tau_1(n) - 2\gamma_1 \mathbf{E} \left[ \left. \frac{d\rho(u)}{du} \right|_{t-\tau_1(n)} \cos \phi(t-\tau_2) e(t) \right]$$

with  $e(t) = z(t) - \rho(t - \tau_1) \cos \phi(t - \tau_2)$  and accordingly for  $\tau_2(n+1)$ . The points are that envelope and phase of the input are already available separately and that formulas only involve derivatives of the (delayed) input, so that we do not need anymore to quad-demodulate the output. However, the input will have to be corrected afterwards. Compared to the first algorithm, the computational complexity is unchanged. Note that for the two algorithms the output sampling clock do not need to be synchronous to the input: it may be a divided version of the input clock, and propagation delays will be absorbed in the correction procedure. In these two approaches, we need to adjust two delays  $\tau_1, \tau_2$  or advances  $\mu_1, \mu_2$  and apply them to the input signal. Adopting very high sampling frequencies (in order to get the required precision) may not be an efficient solution. A key for adjusting these delays/advances from available data samples is digital interpolation.

# D. Interpolation

By Shannon-Nyquist sampling theorem, we know [7] that any band-limited signal x(t) can be recovered exactly from its samples  $x(n) = x(nT_s)$  taken at the sampling frequency  $1/T_s$ , according to

$$x(t) = \sum_{n} x(n) \operatorname{sinc}(\pi(t - nT_s)/T_s),$$

This indicates that, in principle, the samples convey enough information to reconstruct the original signal at any desired time. In particular, it is possible to reconstruct  $x(t - \tau)$ , for any  $\tau$ , and therefore new shifted samples  $x(kT_s - \tau)$  from the original samples  $x(kT_s)$ , according to

$$x(kT_s - \tau) = \sum_n x(nT_s)\operatorname{sinc}(\pi(kT_s - \tau - nT_s)/T_s),$$

where sinc is the cardinal sine. This last expression is in the form of a digital convolution and can be implemented as a filtering operation. However the underlying filter has infinite (sinc) impulse response, and is non causal. Practical implementation thus introduces truncation and delay. Another possibility is to use a convenient approximation of the ideal interpolator above. For instance, the MMSE FIR interpolator [8] is the minimum mean square error approximation of the ideal filter with finite impulse response. Although there exist optimum implementations of these interpolators [9], the coefficients shall be pre-computed and tabulated for each possible fractional delay and these kind structures shall be reserved to the interpolation of fixed (finals) delays.

Another class of interpolators rely on polynomial approximation. Indeed, the Weierstrass approximation theorem states that every continuous function defined on an interval can be uniformly approximated as closely as desired by a polynomial function. One can then use a polynomial (wrt the delay) to approximate the value of the function, given a series of samples, at the desired delay. In numerical analysis this task is solved by Lagrange interpolation, and the solution is unique. However, some other (improper) interpolation polynomials may present a lower residual. The interest of poynomial interpolators is that they can be implemented very efficiently in hardware, and that coefficients can be computed in real time rather than taken from a table. The efficient structure was devised by Farrow [10] and is described below in our context. An improvement was recently presented in [11].

In our experiments, we considered the following interpolators: Lagrange interpolation [5, formula 25.2.15] (5 points), Bessel and modified Bessel interpolation [5, 25.2.47] (4 and 6 points), Newton interpolation [5, 25.2.28] (3 points) and Linear (2 points) interpolation. From these experiments, we find that both Lagrange and Bessel interpolators give interesting results with a low residual.

# E. Farrow implementation of a Bessel interpolator on HPADS

Calculation of the two interpolated signals, in HPADS, are realised with two steps. Integer values of delays are realised by simple delay blocs as non integer values are realised with the following interpolator :

$$y(n,\mu) = x(n) + \mu (x(n+1) - x(n)) + \frac{\mu (\mu - 1)}{4}$$
(6)  
  $\times (x(n+2) - x(n+1) - x(n) + x(n-1)),$ 

where  $y(n, \mu)$  represent the interpolated value of the input at the time  $(n + \mu)Ts$ , with  $0 < \mu < 1$ . One can notice that this interpolator need to sample value after the current point. This implies the reference signal would have to be delayed by 2 samples before being sending at the transmitter output. This kind of interpolator can be efficiency implemented using a Farrow structure [10]. This structure relies on the parallelisation of N FIR filters which can be expressed as:

$$y_i(n) = \sum_{k=-M}^{M-1} h_i(k) x(n-k)$$

Output of these filters are combined so that the output signal can be expressed as

$$y(n,\mu) = \sum_{i=0}^{N-1} \mu^{i} y_{i}(n) = \sum_{i=0}^{N-1} \mu^{i} \sum_{k=-M}^{M-1} h_{i}(k) x(n-k)$$

Rewriting equation (7) in the desired form gives

$$y_0(n) = x(n),$$
  

$$y_1(n) = (-0.25x(n+2) + 1.25x(n+1)) -0.75x(n) - 0.25x(n-1)),$$
  

$$y_2(n) = (0.25x(n+2) - 0.25x(n+1)) -0.25x(n) + 0.25x(n-1))$$

and can be simply simulated on HPADS as presented on Fig. 6.



Fig. 6. Farrow structure for the Bessel interpolator.

# IV. RESULTS

We examined the behavior of the algorithms with respect to the choice of interpolators and with respect to the choice of the adaptation steps. Although certainly not optimum, we choosed to take the same adaptation  $\gamma = \gamma_1 = \gamma_2$  step for the two adaptations. Furthermore, in order to be independent of signals scales, we normalized the adaptation step by the power  $P_x$  of the input signal x(t) We studied both EVM and convergence time. These results are reported on figs 7 and 8, in the case of initial delays  $\Delta_1 = 0.36T_S$  and  $\Delta_1 = 0.12T_S$ . As far as the EVM is concerned, Fig. 7, we obtain an EVM as low as 0.5%, and it is very satisfactory to note that this is true for a large range of values of the adaptation step. Concerning the settling time, we noted Fig. 8 that all interpolators have the same behaviour. As far as the adaptation step increases, the settling time decreases. It can be easily selected to be lower than 50 symbols ( $\alpha \ge 0.1$ ).



Fig. 7. EVM obtained for several interpolators, as a function of parameter  $\alpha = \gamma P_x$ , with  $\gamma$  the adaptation step and  $P_x$  the signal power.



Fig. 8. Convergence performances for several interpolators, as a function of parameter  $\alpha = \gamma P_x$ , with  $\gamma$  the adaptation step and  $P_x$  the signal power.

The algorithms are implemented in Matlab and HPADS. ADS simulations include a more detailed modelization of radio aspects, including sample and hold and baseband filtering. Cosimulation with Matlab is used for updating equations (5). Typical results, given here for the first algorithm, are shown in Fig. 9 in the case of  $0.6T_s$  and  $0.3T_s$  delays for envelope and phase respectively plus baseband analog delays. The parameter  $\alpha$  was fixed to 0.1. The measured EVM after convergence is 0.5% against 16% without correction. The corresponding spectrum is reported in Fig. 10. It shows very interesting performances: the spectrum is highly improved by 30dB compared to the uncorrected case. The remaining noise floor at -50dBc corresponds to interpolation errors.

### V. CONCLUSIONS AND FUTURE WORK

We demonstrate the efficiency of a synchronisation algorithm for polar architecture. Future work include account for further mismatches (loop gain, phase noise), study of finite precision effects. It might also be possible to couple the implementation advantages of the identification algorithm to the correction capabilities of the first algorithm.

### References

- R.B. Staszewski, J.L Wallberg and al, "All-digital PLL and transmitter for Mobile phones," IEEE journal of Solid-State circuits, vol.40, pp2469-2481, Dec 2005.
- [2] K.Sarigeorgidis and J Rabaey, "Ultra low power CORDIC processor for wireless communication algorithms," Journal of VLSI Signal processing, vol. 38, pp115-130, Sept. 2004



Fig. 9. Convergence of  $\mu_1$  and  $\mu_2$  to the true delays  $\Delta_1=0.6T_s$  and  $\Delta_2=0.3T_s.$ 



Fig. 10. Comparison of ideal, uncorrected and corrected spectra, in the case of delays  $\Delta_1 = 0.6T_s$  and  $\Delta_2 = 0.3T_s$ .

- [3] 3GPP TS 125 213, version 7.0.0, release 7 and 3GPP TS 125 101, version 7.6.0, release 7
- [4] J.-F. Bercher, A. Diet, C. Berland, G. Baudoin, M. Villegas, "Monte carlo Estimation of Time mismatch Effect in an OFDM EER Architecture", IEEE Radio & Wireless Conference RAWCON 2004, pp. 283- 286, sept 2004.
- [5] M. Abramowitz, I. A. Stegun, "Handbook of Mathematical Functions: with Formulas, Graphs, and Mathematical Tables", Dover Publications, 1965.
- [6] J.-F. Bercher and C. Berland, « Envelope/phase delays correction in an EER radio architecture », Proceedings of the 13th IEEE International Conference on Electronics, Circuits, and Systems, ICECS2006, pp. 443-446, december 2006.
- [7] J. M. Whittaker, "Interpolatory Function Theory", Cambridge Univ. Press, Cambridge, England,
- [8] G. Oetken, "A New Approach for the Design of Digital Interpolating Filters", IEEE Trans. Acoust., Speech, Signal Processing, vol. ASSP-27, pp. 637-643, Dec. 1979.
- [9] R.E. Crochiere and L.R. Rabiner, "Optimum FIR digital filter implementations for decimation, Interpolation, and Narrow-Band Filtering", IEEE Trans. ASSP, VOL.ASSP-23, NO.5,pp444-456, October 1975.
- [10] C. W. Farrow, "A continuous variable digital delay element", in Proc. IEEE Int. Symp. Circuits Systems, 1988, pp. 2641-2645.
- [11] C. Candan, "An Efficient Filtering Structure For Lagrange Interpolation", IEEE Signal Processing Letters, Vol. 14, No.1, p. 17-19, Jan. 2007.