EFFICIENT SYMBOL SYNCHRONIZATION TECHNIQUES USING VARIABLE FIR OR IIR INTERPOLATION FILTERS

Martin Makundi and Timo I. Laakso

Helsinki University of Technology, Signal Processing Laboratory, FIN-02015 HUT, Espoo, Finland
email: {martin.makundi, timo.laakso}@hut.fi

ABSTRACT

Maximum Likelihood estimation theory can be used to develop optimal timing recovery schemes for digital communication systems. Variable digital interpolation filters are commonly used for symbol timing adjustment. The so-called gathering structure offers an efficient and flexible realization structure for such an interpolation task. In this paper we propose a feedforward timing estimation scheme which uses the efficient variable gathering filter structure for FIR filters leveraging an algorithm previously developed for the so-called Farrow structure. The scheme is based on a low-order polynomial approximation of the log likelihood function. Furthermore, the scheme is extended to an IIR gathering structure with transient suppression. The performance of IIR and FIR implementations and their computational complexity are compared in example simulations, showing that polynomially approximated low-order Thiran IIR fractional-delay filters offer more efficient implementation than their Lagrange FIR counterparts with similar performance.

1. INTRODUCTION

The new trend is to use digital receivers where the sampling of the demodulated (baseband) signal is performed by a fixed sampling rate oscillator. This new design approach reduces the number of required analog components as most of the receiver functions are performed digitally. Using digital signal processing instead of analog signal processing allows increased flexibility, configurability, and integrability of the receiver. From these properties arises the software radio concept [1] which is the natural progression of digital radio receivers towards multimode, multistandard terminals in which most of the functionalities are defined by software.

In order to evaluate the received symbols at optimum sampling instants we must operate in synchronism with the symbol stream. In a software radio synchronization can be performed digitally using so-called fractional-delay (FD) filters. Previously published articles have been utilizing FIR FD filters [2]-[5]. IIR filters, however, often provide more efficient implementation than their FIR counterparts. In this paper we compare their performance.

This paper proposes novel IIR and FIR implementations for Maximum Likelihood (ML) symbol timing estimation and synchronization, utilizing the so-called gathering structures [6] for efficient implementation of variable IIR and FIR filters. Furthermore, we investigate whether an IIR implementation for synchronization can compete with an FIR implementation in terms of performance and computational complexity. In particular, we compare the performance of low-order FIR and IIR FD filters, both having the maximally flat group delay property, namely the Lagrange and the Thiran FD approximations [7].

2. FEEDFORWARD ML TIMING ESTIMATION SCHEME

Fig. 1 shows a receiver structure where symbol synchronization is performed digitally using an FD filter. The received signal $r(k)$ is first digitally sampled at a fixed sampling rate $F_s = 1/T_s$. After sampling it is passed through a matched filter $h_c(k)$ and a channel equalizer (EQ). Before symbol detection the timing offset is corrected using an FD interpolation filter and ML feed-forward timing estimation.

$\hat{a}_k$ = $\hat{m}(k)$

Figure 1: Digital receiver using interpolation for symbol synchronization.

The ML estimate of the log likelihood function (LLF) for symbol timing estimation, assuming a gaussian noise channel and successful equalization, is given by [2, 8]

$$\Lambda(d) = \sum_{j=1}^{M} \hat{a}_j \hat{m}(k, d)$$

(1)

Here $\{\hat{a}_j\}$ are the correct or estimated symbol values, $\hat{m}(k, d)$ the fractionally delayed output samples of the matched filter, $d$ a fractional delay, and $M$ the number of used past symbols. The delay is assumed to remain constant within the block of $M$ symbols. In our discussion we will assume a training signal and thus the symbols in (1) are known, i.e., $\{\hat{a}_j\} \rightarrow \{\hat{a}_k\}$.

The ML feed-forward fractional delay estimate $\hat{d}$ is defined as

$$\hat{d} = \frac{\hat{\tau}}{T_s} = \arg \max_d \{\Lambda(d)\}$$

(2)

where $\hat{\tau}$ is the timing error estimate in seconds and $T_s$ the sampling interval. In this paper $\Lambda(d)$ is approximated by a polynomial

$$\Lambda(d) \approx A_0 d^p + A_{p-1} d^{p-1} + \cdots + A_0$$

(3)
of order $P$. The coefficients $\{A_0, A_1, \ldots, A_P\}$ can be obtained by solving the system of linear equations

\[
\begin{bmatrix}
d_0^P & d_1^{P-1} & \cdots & d_1^0 \\
d_1^P & d_1^{P-1} & \cdots & d_1^0 \\
\vdots & \vdots & \ddots & \vdots \\
d_p^P & d_p^{P-1} & \cdots & d_p^0
\end{bmatrix}
\begin{bmatrix}
A_0 \\
A_1 \\
\vdots \\
A_P
\end{bmatrix}
= \begin{bmatrix}
\Lambda(d_0) \\
\Lambda(d_1) \\
\vdots \\
\Lambda(d_P)
\end{bmatrix}
\tag{4}
\]

provided we have an interpolation filter that can evaluate $\Lambda(d_k)$ for the chosen set of delay values $\{d_k, d_1, \ldots, d_P\}$.

The maximum-point estimate is then obtained in closed form by solving for the peak point of the polynomial (we assume there is only a single maximum within the range $d \in [-0.5, 0.5]$) [4]. The LLF (3) is very smooth and thus $P$ can be small, e.g., 3. The structures presented next in this paper can, however, be used with any $P$.

### 3. Timing Adjustment Using Gathering Structures

In this section we propose an implementation of variable FD filters using the IIR and FIR gathering structures [6]. The implementation allows to efficiently obtain also the values of the LLF in (1) and in (4) at the desired values of $d$.

#### 3.1. IIR Allpass Structure

IIR filters have not yet been used for interpolation in synchronization, even though they often provide more efficient implementation than their FIR counterparts [7]. This subsection proposes a novel approximation of the LLF in (1), utilizing IIR allpass gathering structures [6] in which the fractionally delayed symbol estimate is obtained as

\[
\hat{m}(k, d_k) = m(k - N) + \sum_{l=0}^{L} F_{ld_k} d_l^{-1} \tag{5}
\]

where the variable

\[
F_{ld_k} = \sum_{n=1}^{N} [m(k - N + n) - \hat{m}(k - n, d_k)] e_{ln} \tag{6}
\]

is introduced merely for the purpose of conveniently pinpointing the signal paths in Fig. 2. The subscripts $d_k$ express the effect of feedback in variable IIR FD filters — changing the coefficients of a recursive filter on-line causes transients. In order to avoid transients the filter output is multiplexed and the different constant delay values, $\{d_k\}$, processed in separate feedback loops. The symbol estimates, however, at the estimated delay values $d$ will suffer from transients, because only a single feedback loop is used for the output signal $\hat{m}(k, d)$. If necessary, transient suppression [9] can be used to enhance quality of the symbol estimates at the estimated delays $d$.

#### 3.2. FIR Structure

According to [6] we have

\[
\hat{m}(k, d_k) = \sum_{l=0}^{L} F_{ld_k} d_l^{-1} = \sum_{l=0}^{L} \left[ \sum_{n=0}^{N} m(k - n) e_{ln} \right] d_l^{-1} \tag{7}
\]

where the variable $F_{ld_k}$ is again introduced for conveniency. The signal paths in question can be pointed out from FIR gathering structures in a manner similar to what is shown for IIR gathering structures in Fig. 2.

#### 3.3. Oversampling

If we oversample the input signal by an integer factor $U$ we end up with $U$ sets of candidate symbols (polyphase components). Because of integer-ratio oversampling and fractional delaying being limited to $[-0.5, \ldots, 0.5]$, one of the $U$ polyphase components will maximize $\Lambda(d)$ in Eq. (2) for a certain symbol. This means that we must evaluate

\[
\Lambda_\lambda(d_k) = \left| \sum_{l=0}^{M} \hat{a}_l \hat{m}(Uk - \lambda, d_l) \right|, \quad \lambda = 0, 1, 2, \ldots, U-1 \tag{8}
\]

for all the different polyphase components, choose the polyphase component for which $\max_\lambda \{\Lambda_\lambda(d_k)\}$ is largest, and finally evaluate (2) for it to obtain the ML delay estimate.

### 4. Simulations

In this Section the performance of the proposed IIR and FIR FD filters is compared. The FD filters used in the simulations were “Lagrange” FIR FD filters [7] using the gathering structure, “Thiran g” IIR FD filters as explained in [10], and “Thiran (poly.)” IIR FD filters whose constant coefficients are a polynomial approximation (in $d$) of the Thiran allpass filter model coefficients [7]. Furthermore, the performance of IIR FD filters with and without transient suppression was compared.
The block size used for timing estimation was \( M = 64 \) symbols. The total number of QAM-64 symbols (root-raised cosine with roll-off factor \( \alpha = 0.35 \)) transmitted over an additive white gaussian noise (AWGN) channel was 256. The simulation results were obtained by evaluating 128 symbols after the initial synchronization transient. The signal-to-noise ratio (SNR) used on the AWGN channel was 20 dB. The symbol oversampling factor, \( U \), used at the receiver was 2. Finally, the order of approximating the LLF in (3) was \( P = 3 \), as in [2].

Table 1 shows the implementation complexities of the different FD filters in terms of delay elements, variable multipliers, and constant coefficients. For handling the block of \( M \) previously received symbols a variety of implementations is available, but in order to save computations we chose a reusability approach in which the previously computed symbol estimates are stored in a bank of \( (M - 1)(P + 1) \) delay elements. Using this approach the extra implementation complexity coming from block-processing the \( M \) symbols is the same for both FIR and IIR and is, therefore, not shown in Table 1.

### 4.1. Results

The signal path delay values (relative to the symbol interval) used in our simulations were \( d \in \{-0.5, -0.475, \ldots, 0.5 \} \). For each delay value we transmitted the same 256 symbols and measured the timing jitter variance \( \text{Var}(\hat{d} - d) \), the timing jitter mean \( \text{Avg}(\hat{d} - d) \), and the excess mean-squared error (EMSE) caused by interpolation (relative to ideal synchronization) as

\[
\text{EMSE} = 10 \log_{10}(\text{Avg}(\Delta^2)/P_n),
\]

where \( \Delta = m(k, \hat{d}) - m(k, d) \) and \( P_n = \text{Avg}(m(k, d)^2) \). We consider this a realistic scenario, because in reality both the transmitter and the receiver have approximately the same symbol rate. The timing error then consists of an offset that can be considered constant for the received block of \( M \) symbols.

Peak and average EMSE measurements show that the second-order Lagrange FD filter has performance comparable to a first-order Thiran FD filter (see Table 1). Performance measurement results for different orders of the filters are plotted in Figs. 3-5. The EMSE of the allpass filter varies less for different values of \( d \) as compared to the EMSE of the Lagrange FD filter.

In order to simplify the IIR implementation we compared the Thiran \( g \) performance against a polynomially approximated Thiran. As it turns out, the performance of this Thiran (polyn.) is similar to the Thiran \( g \) (see Fig. 5) while its implementation complexity is lower with \( I = N + 1 \) less multiplications by constant coefficients.

The extra computational complexity that would result from implementing transient suppression in IIR filters is of the order \( O(N^2) \) [9]. Therefore, IIR implementations with and without transient suppression are compared in Fig. 6. The steady state results of block-processing the \( M \) symbols show that using IIR filters without transient suppression results in only a minor degradation in synchronization performance. Most importantly, the implementation complexity of a first-order Thiran (polyn.) without transient suppression is even less than that of a second-order Lagrange FIR FD filter having similar performance.

We note, however, that if \( d \) were to vary a lot between consecutive symbols the transient energy in IIR filters would grow. This might render the transient-suppressed IIR FD filters too costly for such applications.

### 5. CONCLUSIONS

This paper derived novel IIR and FIR gathering structures for maximum-likelihood symbol synchronization. Especially, variable IIR filters have not been used for synchronization applications before. While the performance of the FIR gathering structures is identical to the so-called Farrow structures, the latter implementation structure requires \( NL \) delay elements more. For IIR structures, the Farrow structure is not applicable at all [6].

The synchronization performance was measured for both IIR and FIR FD implementations with different computational complexities. The performance of the IIR FD filters was shown both with and without transient suppression. This indicated that when the incoming signal is oversampled by an integer ratio, relative to the symbol rate, it is not necessary to complicate the IIR FD synchronization filter with extra circuitry for transient suppression. Furthermore, the polynomially approximated Thiran IIR FD filter having sufficient performance it allows low-order IIR FD implementation with complexity less than that of a comparable FIR FD filter. In conclusion, IIR FD filters seem a promising choice for synchronization applications employing integer-ratio oversampling.

### 6. REFERENCES


Figure 3: Timing jitter variance, timing jitter mean, and EMSE of the Lagrange FIR gathering FD filter. The dash-dot lines represent a first-order ($N = 1$), the dashed lines a second-order ($N = 2$), and the solid line a third-order ($N = 3$) implementation, respectively.

Figure 4: Timing jitter variance, timing jitter mean, and EMSE of the Thiran g IIR gathering FD filter. The dash-dot lines represent a first-order ($N = 1$), the dashed lines a second-order ($N = 2$), and the solid line a third-order ($N = 3$) implementation, respectively.

Figure 5: Timing jitter variance, timing jitter mean, and EMSE of the Thiran (polyn.) IIR gathering FD filter. The dash-dot lines represent a first-order ($N = 1$), the dashed lines a second-order ($N = 2$), and the solid line a third-order ($N = 3$) implementation, respectively.

Figure 6: EMSE of different orders of the IIR FD filters with and without transient suppression; IIR filter order $N = 1$ on top, below it $N = 2$ and $N = 3$. Solid and dashed lines represent the Thiran g with and without transient suppression, respectively. Dash-dotted and dotted lines represent the Thiran (polyn.) with and without transient suppression, respectively.


