The HKU Scholars Hub    The University of Hong Kong    香港大學學術庫

| | |
|---|---|
| **Title** | **Time-Interleaved Analog-to-Digital Converter (TIADC) Compensation Using Multichannel Filters** |
| **Author(s)** | **Lim, YC; Zou, YX; Chan, JW; Chan, SC** |
| **Citation** | **IEEE Transactions on Circuits and Systems Part 1: Regular Papers, 2009, v. 56 n. 10, p. 2234-2247** |
| **Issued Date** | **2009** |
| **URL** | **http://hdl.handle.net/10722/73667** |
| **Rights** | **IEEE Transactions on Circuits and Systems Part 1: Regular Papers. Copyright © IEEE.** |

# Time-Interleaved Analog-to-Digital-Converter Compensation Using Multichannel Filters

Yong Ching Lim, *Fellow, IEEE*, Yue-Xian Zou, *Senior Member, IEEE*, Jun Wei Lee, *Member, IEEE*, and Shing-Chow Chan, *Senior Member, IEEE*

*Abstract*—Published methods that employ a filter bank for compensating the timing and bandwidth mismatches of an $M$-channel time-interleaved analog-to-digital converter (TIADC) were developed based on the fact that each sub-ADC channel is a downsampled version of the analog input. The output of each sub-ADC is filtered in such a way that, when all the filter outputs are summed, the aliasing components are minimized. If each channel of the filter bank has $N$ coefficients, the optimization of the coefficients requires computing the inverse of an $MN \times MN$ matrix if the weighted least squares (WLS) technique is used as the optimization tool. In this paper, we present a multichannel filtering approach for TIADC mismatch compensation. We apply the generalized sampling theorem to directly estimate the ideal output of each sub-ADC using the outputs of all the sub-ADCs. If the WLS technique is used as the optimization tool, the dimension of the matrix to be inversed is $N \times N$. For the same number of coefficients (and also the same spurious component performance given sufficient arithmetic precision), our technique is computationally less complex and more robust than the filter-bank approach. If mixed integer linear programming is used as the optimization tool to produce filters with coefficient values that are integer powers of two, our technique produces a saving in computing resources by a factor of approximately $(10^{0.2N(M-1)})/(M - 1)$ in the TIADC filter design.

*Index Terms*—Time-interleaved analog-to-digital converter (TIADC), TIADC mismatch compensation, multichannel filter.

## I. INTRODUCTION

IN MANY signal processing applications, the highest sampling speed is limited by the speed of the analog-to-digital converter (ADC). In order to achieve an analog-to-digital (AD) conversion time that is much shorter than what can be achieved with a single ADC, a bank of properly sequenced sub-ADCs is used. Each sub-ADC samples and converts at a time displaced from the others at a regular interval, as shown in Fig. 1. Such an AD conversion system is called a time-interleaved ADC (TIADC) [1]–[3]. The advantages of a TIADC are well known [1] but TIADC exhibits several major problems [4]–[6].
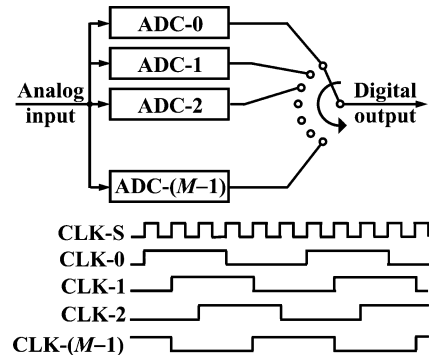
Fig. 1. TIADC employing $M$ sub-ADCs.

Table I shows a list of the symbols for easy reference.

In Fig. 1, each of the $M$ sub-ADCs of the TIADC system is assumed to be linear but a different sub-ADC may have a slightly different constant of proportionality leading to a "gain mismatch" and a different bandwidth leading to a bandwidth mismatch. Each sub-ADC may also have a different dc offset leading to an offset mismatch. Detailed discussions on offset and gain mismatches may be found in [7]–[13]. In Fig. 1, ADC-$m$ is driven by clock signal CLK-$m$, $m = 0, 1, 2, \ldots, M - 1$, generated from a stable system clock CLK-S. Let the frequency of CLK-S be $f_s$. The sampling interval for the TIADC system is $T$, given by

$$T = 1/f_s. \qquad (1)$$

We shall assume that the highest frequency component of the input signal is band limited to $0.5f_s$.

Ideally, the sampling instant of each sub-ADC is spaced apart by $T$ unit time. The conversion speed of each sub-ADC is $f_s/M$ but the TIADC system conversion speed is $f_s$. The circuits generating CLK-$m$, $m = 0, 1, 2, \ldots, M - 1$, (including routing) introduce delay between the edges of CLK-S and CLK-$m$; the delay may be different for different clock signals and may drift with temperature. The difference in delay contributes to timing mismatch.

Many techniques have been published in literature for compensating bandwidth and timing mismatch [14]–[28]. Among these techniques, [14]–[25] use a bank of filters where each filter of the filter bank filters the output of each sub-ADC; the outputs of the filters are summed to form the final output, as shown in Fig. 2. A single-channel filter is a filter with one input port. Each filter in the filter bank is a single-channel filter producing output at a rate of $f_s$; the compensated output is obtained by summing all the outputs of the filters. In Fig. 2, $H_{A,m}(\omega)$ represents the

| Symbol | Definitions and comments |
|---|---|
| $M$ | Number of sub-ADCs. |
| $f_s, T$ | $f_s$ = TIADC sampling rate. $T = 1/f_s$. ($T = 1$ on and after (11b).) |
| $H_{A,m}(\omega)$ | See Fig. 2 and Fig. 5. |
| $v(n)$ | $n^{\text{th}}$ ideal sample of analog input. |
| $x_m(n)$ | $n^{\text{th}}$ output sample of ADC-$m$. |
| $\ddot{x}_m(n)$ | Ideal value of $x_m(n)$. |
| $\hat{x}_m(n)$ | Estimate of $\ddot{x}_m(n - \eta)$. |
| $\eta$ | Delay between $\ddot{x}_m(n)$ and estimate of $\ddot{x}_m(n)$. |
| $\bar{x}_m(n)$ | Vector of consecutive sub-ADC data for computing $\hat{x}_m(n)$. See (6). |
| $\boldsymbol{h}_m$ | Vector of coefficients for $m^{\text{th}}$-channel filter. See (6). |
| $h_{i,j}(n)$ | $n^{\text{th}}$ coefficient for estimating the desired output of ADC-$i$ using data from ADC-$j$. |
| $\hat{X}_m(z^M)$ | $z$-transform of $\hat{x}_m(n)$. See (8). |
| $\boldsymbol{H}_m(z^M)$ | $z$-transform of $\boldsymbol{h}_m$. See (9). |
| $\hat{X}_m(e^{jM\omega})$ etc. | $\hat{X}_m(z^M)$ etc. evaluated at $z = e^{j\omega}$. |
| $V(\omega)$ | Frequency spectrum of $v(n)$. |
| $E_m(e^{j\omega})$ | Difference between $\hat{X}_m(e^{jM\omega})$ and $V(\omega)e^{j(m-MD)\omega}$. See (14b). |
| $\xi_m(j\omega)$ | See (15), (16), and the explanations between (15) and (16) as well as two lines below (16). |
| $N_m$ | The number of coefficients in $\boldsymbol{h}_m$. |
| $N$ | Number of coefficients in $\boldsymbol{h}_m$. Number of coefficients in FILTER-$m$. |
| $L$ | Number of evaluation frequencies. |
| $[\bullet]^{\text{T}}$ | Transpose of $[\bullet]$. |
| $[\bullet]^{\text{H}}$ | Complex conjugate transpose of $[\bullet]$. |
| $\boldsymbol{E}_m$ | $\boldsymbol{E}_m = \left[E_m(e^{j\omega_1}), E_m(e^{j\omega_2}), ..., E_m(e^{j\omega_L})\right]^{\text{T}}$. See(19a). |
| $\boldsymbol{\Lambda}_m$ | $\boldsymbol{\Lambda}_m = \left[e^{j(m-M\eta)\omega_1}, e^{j(m-M\eta)\omega_2}, ..., e^{j(m-M\eta)\omega_L}\right]^{\text{T}}$. See (19b). |
| $\boldsymbol{\Xi}_m$ | $\boldsymbol{\Xi}_m = \left[\xi_m(j\omega_1), \xi_m(j\omega_2), ..., \xi_m(j\omega_L)\right]^{\text{T}}$. See (19c). |
| $\mathbf{W}$ | See (19d). |
| $w(\omega)$ | Frequency response error weight used in MILP. |
| $\delta$ | Peak response error used in MILP. |
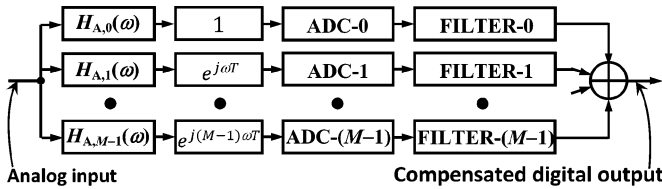


Fig. 2. Schematic of the filter-bank approach.

frequency response of the $m$th sub-ADC channel. It includes the frequency responses of the sub-ADC, the waveguide transmitting the signal to the sub-ADC, and the photonic front end if the analog signal is sampled using a laser. It also includes timing mismatch, i.e., if the timing mismatch is $\Delta_m T$, $e^{j\omega\Delta_m T}$ will be a factor in $H_{A,m}(\omega)$. It does not include dc bias, harmonic distortion, and noise. The $e^{jm\omega T}$ term in Fig. 2 represents the sampling clock displacement between sub-ADCs.

The detailed development for the methods reported in [14]–[25] may differ from one another. However, the basic principles are the same: 1) TIADC is a collection of many channels of under-sampled analog input, and 2) the goal is to design a filter bank in which each filter shapes the spectral characteristics of each sub-ADC output in such a way that, when all the filter outputs are summed, the aliasing terms are

removed as much as possible and the "signal term" is as close as possible to a linear phase term. In order to facilitate comparison between our technique and existing techniques, we shall briefly describe the technique presented in [18].

The frequency spectrum of the compensated TIADC output is $\tilde{Y}(f)$, where $f = \omega/(2\pi)$. It is given in terms of the frequency spectrum of the analog input $X(f)$ by (after [18, eq. (6), (7)])

$$\tilde{Y}(f) = \sum \tilde{c}_m\left(f - \frac{m}{M}f_s\right)X\left(f - \frac{m}{M}f_s\right) \quad (2)$$

where

$$\tilde{c}_m(f) = \frac{1}{M}\sum_{k=0}^{M-1}\sum_{n=0}^{N-1}e^{-j2\pi k\frac{m}{M}}e^{-j2\pi n(\frac{f}{f_s}+\frac{m}{M})}H_{A,k}(f)\hat{h}_k(n),$$
$$m = 0, 1, \ldots, M - 1. \quad (3)$$

In (3), $\hat{h}_k(n)$ is the $n$th impulse response of FILTER-$k$. The impulse response $\hat{h}_k(n)$ is optimized so that the aliasing gain $\tilde{c}_m(f)$ for $m \neq 0$ is as close as possible to zero and $\tilde{c}_m(f)$ for $m = 0$ is as close as possible to a linear phase term. The weighted least squares (WLS) solution is given by (after [18, eq. (10)])

$$\hat{H}_{\text{WLS}} = (\mathbf{A}^{\text{H}}\mathbf{W}\mathbf{A})^{-1}\mathbf{A}^{\text{H}}\mathbf{W}D. \quad (4)$$

The derivation for (4) can be found in [18]. Let $L$ be the number of evaluation frequencies and $N$ be the length of FILTER-$k$. The matrix $\mathbf{A}$ is $LM \times NM$, $\mathbf{W}$ is $LM \times LM$, $\hat{H}_{\text{WLS}}$ is $NM \times 1$, and $D$ is $LM \times 1$ (after [18]). The solution requires the inversion of $\mathbf{A}^{\text{H}}\mathbf{W}\mathbf{A}$, which is an $NM \times NM$ matrix.

In Section II, we present a new approach with reduced computational complexity in the design process. Our approach is a multichannel filtering approach. A multichannel filter is a filter with more than one input port. In our approach, each filter is an $M$-channel filter producing output at a rate of $f_s/M$; the compensated output is obtained by selecting the outputs of the filters one at a time. The derivation of the coefficient values of the multichannel filter is cast into a WLS optimization problem in Section III. In our approach, the corresponding matrices in (4) have dimensions $L \times N$, $L \times L$, $N \times 1$, and $L \times 1$, respectively (instead of $LM \times NM$, $LM \times LM$, $NM \times 1$, and $LM \times 1$). The corresponding matrix to be inverted has a dimension of $N \times N$ (instead of $NM \times NM$). As can be seen in the computational complexity analysis presented in Section IV, the computational complexity of our approach is much lower than that of the filter-bank approach. As a result of the much smaller matrices, our approach is also numerically more robust. Another advantage of our approach is that the filters need not have the same number of coefficients. The optimization problem is cast into a mixed integer linear programming (MILP) problem in Section VII for the design of TIADC compensation filters whose coefficients are integer powers of two. An actual 14-bit $4 \times 400$ MS/s (megasamples per second) TIADC implemented with our technique is presented in Section VIII. A summary of the differences between our approach and the filter-bank approach and the advantages of our approach over the filter-bank approach is presented in Section IX.

Several examples are presented in Sections V–VII. In the examples, offset mismatch is not considered. Offset mismatch can neither be compensated by the filter-bank approach nor the

multichannel approach. Nevertheless, it can be easily estimated from the sub-ADC outputs with their inputs grounded. A constant bias may then be added to each sub-ADC output to remove the offset. Both the filter-bank and multichannel approaches are able to compensate for gain mismatch although they are not included in the examples presented in Sections V–VII. Gain mismatch can also be compensated easily by connecting all the sub-ADCs to a full-scale dc input and then scaling each of the sub-ADC outputs using a scaling amplifier such that all sub-ADCs give the same full-scale output. In *Example-I* and *Example-IV*, both timing and bandwidth mismatches are considered. In *Example-II*, only bandwidth mismatch is considered. In *Example-III* only timing mismatch is considered. Not both timing and bandwidth mismatches are considered in *Example-II* and *Example-III* simply to provide a variety of examples; timing and bandwidth mismatches can be added if desired. *Example-V* is an actual TIADC example with offset, gain, timing, bandwidth, and frequency-response mismatches.

Closely related to the techniques reported in [14]–[28] is another class of multiple sub-ADC systems which uses a bank of analog filters to split the input signal in the frequency domain into $M$ channels [29]–[31]. Each sub-ADC digitizes the output of each frequency channel at $1/M$ the TIADC sampling rate. A bank of digital filters is designed to combine all the frequency interleaved sub-ADC outputs to form the final system output in such a way that the aliasing terms are minimized and the output is a close approximation to a delayed version of the input.

Several authors have proposed blind equalization techniques [9], [32]–[36]. Blind techniques do not require using a known signal to estimate the mismatch parameters. In [9], the gain and timing mismatches are estimated using the fact that, if the mismatch compensation is perfect, the TIADC output would be wide sense stationary. When compared to offline calibration techniques, blind calibration techniques are, in general, less accurate and have much higher computational complexity.

## II. TIADC MISMATCH COMPENSATION WITH REDUCED DESIGN COMPLEXITY

In this section, we present a new approach to TIADC compensation. This approach results in a reduction in the arithmetic complexity in the design process. In our approach, we make use of the fact that neighboring sub-ADCs sample a filtered version of the analog input signal nonuniformly at a time interval whose average value is $T$. Fig. 3 shows an example of such a scenario for $M = 3$. In Fig. 3, $x_m(n)$ is the $n$th output sample of ADC-$m$, and $v(n)$ is the $n$th sample of the analog input if it was sampled by an ideal ADC. It has been shown in [37] that a signal may be represented by a weighted combination of any irregularly spaced samples provided that the highest frequency component of the signal is less than 0.5 times the average sampling rate. (The result presented in [37] is known as the *generalized sampling theorem*. It is the general form of the sampling theorem in [38]–[40].) Let $\ddot{x}_m(n)$ be the ideal value of $x_m(n)$. The ideal value of $x_m(n)$ is the value of $x_m(n)$ if ADC-$m$ is a perfect ADC, free from timing error and free from all forms of mismatch. Thus

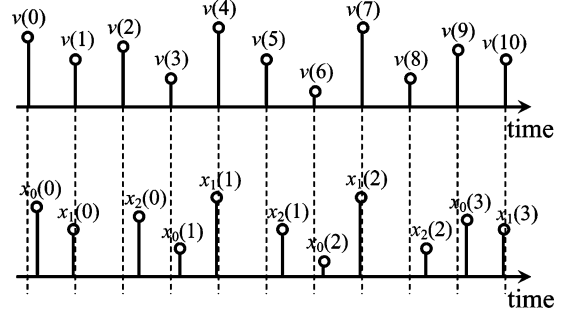$$v(nM + m) = \ddot{x}_m(n). \tag{5}$$



Fig. 3. $x_m(n)$ is the $n$th output of ADC-$m$. $v(n)$ is the $n$th sample of the analog input if it was sampled by an ideal ADC.

Our approach is to estimate the value of $v(n)$ using a weighted combination of the outputs from all the sub-ADCs sampled consecutively in the neighborhood of $v(n)$. The weights also provide a filtering effect to compensate for frequency-response mismatch (including bandwidth mismatch) between the sub-ADCs. For example, $v(5)$ (or $\ddot{x}_2(1)$) in Fig. 3 may be estimated using the seven consecutive samples $x_2(0)$, $x_0(1)$, $x_1(1)$, $x_2(1)$, $x_0(2)$, $x_1(2)$, and $x_2(2)$; a better estimate may be obtained by using more consecutive samples. As a result of the sampling structure of the sub-ADCs, the sampled data appear in $M$ channels of a multichannel time series [41]. Thus, we may write "estimated value of $\ddot{x}_2(1)$" $= h_2(0)x_2(2) + h_2(1)x_1(2) + h_2(2)x_0(2) + h_2(3)x_2(1) + h_2(4)x_1(1) + h_2(5)x_0(1) + h_2(6)x_2(0)$ where $h_2(n), n = 0, 1, \ldots, 6$, is a constant. We may also write "estimated value of $\ddot{x}_2(1)$" $= h_{2,2}(0)x_2(2) + h_{2,1}(0)x_1(2) + h_{2,0}(0)x_0(2) + h_{2,2}(1)x_2(1) + h_{2,1}(1)x_1(1) + h_{2,0}(1)x_0(1) + h_{2,2}(2)x_2(0)$ where $h_{m,k}(n)$ means the $n$th partial coefficient in estimating the ideal outputs of ADC-$m$ using outputs from ADC-$k$. Obviously, $h_{2,2}(0) = h_2(0), h_{2,1}(0) = h_2(1), \ldots$, etc. The equation defines the input–output relationship of a multichannel filter with three input ports. We note from the equation that $x_2(2)$ is used to estimate $x_2(1)$. However, $x_2(2)$ is "future" to $x_2(1)$. Thus, in order to make the estimation process causal, we shall aim at estimating a past output of a sub-ADC instead of estimating its immediate output. Let $\hat{x}_m(n)$ denote the estimated value of $\ddot{x}_m(n-\eta)$ where $\eta$ represents a time delay. In general, $\eta$ need not be an integer. In order to simplify the notation, we shall write $\hat{x}_m(\eta)$ instead of "estimated value of $\ddot{x}_m(n-\eta)$." Thus, we write

$$\hat{x}_m(n) = [\bar{\boldsymbol{x}}_m(n)]^{\mathrm{T}} \boldsymbol{h}_m. \tag{6}$$

In (6), $\hat{\boldsymbol{x}}_m(n)$ is the signal vector in the filter, and $\boldsymbol{h}_m$ is the coefficient vector. We shall illustrate the notation using $\hat{x}_2(2)$ as an example. In the aforementioned example, $\eta = 1$, $\hat{x}_2(2)$ is the estimated value of $\ddot{x}_2(1)$, $[\bar{\boldsymbol{x}}_2(2)]^{\mathrm{T}} = [x_2(2), x_1(2), x_0(2), x_2(1), x_1(1), x_0(1), x_2(0)]$, and $\boldsymbol{h}_2 = [h_{2,2}(0), h_{2,1}(0), h_{2,0}(0), h_{2,2}(1), h_{2,1}(1), h_{2,0}(1), h_{2,2}(2)]^{\mathrm{T}}$. The symbol $[\bullet]^{\mathrm{T}}$ denotes the transpose of $[\bullet]$. In order to fix the idea, the computation structure for $\hat{x}_2(n)$ for this particular example is shown in Fig. 4. Note that, in Fig. 4, the sub-ADC outputs $x_0(n), x_1(n)$, and $x_2(n)$ form a multichannel time series [41]. The filter shown in Fig. 4 is a multichannel filter since it has three input ports.
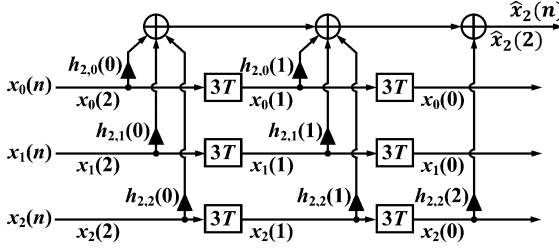
Fig. 4. Computation structure for $\hat{x}_2(n)$ for the example where $\hat{x}_2(n) = h_{2,2}(0)x_2(n) + h_{2,1}(0)x_1(n) + h_{2,0}(0)x_0(n) + h_{2,2}(1)x_2(n-1) + h_{2,1}(1)x_1(n-1) + h_{2,0}(1)x_0(n-1) + h_{2,2}(2)x_2(n-2)$. Signals for $n = 2$ are marked on the signal lines.



Fig. 5. Schematic of our multichannel filtering approach.



Fig. 6. Implementation structure for $\boldsymbol{H}_2(z)$ with $M = 5$ and $N_2 = 15$.

Let $N_m$ denote the number of coefficients in $\boldsymbol{h}_m$. In general, $N_i$ may be different from $N_j$ for $i \neq j$. If ADC-$i$ has a more serious mismatch than ADC-$j$, $N_i$ may be chosen to be larger than $N_j$. In order to facilitate a comparison with the filter-bank approach, we shall let $N = N_i = N_j$ for all $i \neq j$, i.e., all $N_m$'s are equal, if no specific value is given to each $N_m$. Therefore, we have used the same notation "$N$" to denote the number of coefficients for each output channel in our multichannel approach and the number of coefficients for each filter in the filter-bank approach. For the same $M$ and $N$, both our multichannel approach and the filter-bank approach will have the same total number of coefficients $MN$. Hence, we shall be making a comparison based on the same total number of coefficients.

Let $\hat{X}_m(z^M)$ denote the $z$ transform of $\hat{x}_m(n)$. The $z^M$ factor in $\hat{X}_m(z^M)$ signifies that $\hat{x}_m(n)$ is clocked with sampling interval $MT$. We have

$$\hat{X}_m(z^M) = \sum_{n=0}^{\infty} \hat{x}_m(n)z^{-Mn}. \tag{7}$$

Let $\bar{\boldsymbol{X}}_m(z^M)$ and $\boldsymbol{H}_m(z^M)$ denote the multichannel $z$ transforms of $\bar{\boldsymbol{x}}_m(n)$ and $\boldsymbol{h}_m$, respectively. Similarly, the $z^M$ factors in $\bar{\boldsymbol{X}}_m(z^M)$ and $\boldsymbol{H}_m(z^M)$ signify that the clock period is $MT$. We have

$$\bar{\boldsymbol{X}}_m(z^M) = \sum_{n=0}^{\infty} [x_{M-1}(n), \ldots, x_1(n), x_0(n)]^{\mathrm{T}} z^{-Mn} \tag{8}$$

$$\boldsymbol{H}_m(z^M) = \sum_{n=0}^{\bullet} [h_{m,M-1}(n), \ldots, h_{m,1}(n), h_{m,0}(n)]^{\mathrm{T}} z^{-Mn}. \tag{9}$$

In (9), the summation $\sum_{n=0}^{\bullet}$ is summed over all $n$, where at least one of the elements in $[h_{m,M-1}(n), \ldots, h_{m,1}(n), h_{m,0}(n)]$ is nonzero. (The $z$ transform of a multichannel system is the matrix/vector formed from the $z$ transform of each of the separate responses [41].) For the purpose of clarifying the concept, in the aforementioned example, $[\bar{\boldsymbol{X}}_2(z^3)]^{\mathrm{T}} = \sum_{n=0}^{\infty}[x_2(n), x_1(n), x_0(n)]z^{-3n}$ and $\boldsymbol{H}_2(z^3) = [h_{2,2}(0), h_{2,1}(0), h_{2,0}(0)]^{\mathrm{T}} + [h_{2,2}(1), h_{2,1}(1), h_{2,0}(1)]^{\mathrm{T}}z^{-3} + [h_{2,2}(2), 0, 0]^{\mathrm{T}}z^{-6}$. Taking the $z$ transform of (6), we have

$$\hat{X}_m(z^M) = \left[\bar{\boldsymbol{X}}_m(z^M)\right]^{\mathrm{T}} \boldsymbol{H}_m(z^M). \tag{10}$$

Fig. 5 shows a schematic of our approach for TIADC compensation.

Fig. 6 shows an implementation structure for $\boldsymbol{H}_2(z)$ with $M = 5$ and $N_2 = 15$. Figs. 7 and 8 show two complete imple-
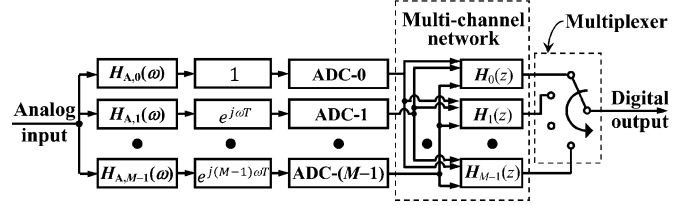
mentation structures for the case where $M = 3$ and $N = 4$. The differences in Figs. 7 and 8 lead to differences in the complexity of the optimization process and in the mismatch compensation performance. In Fig. 7, $\hat{x}_0(n) = h_{0,2}(0)x_2(n) + h_{0,1}(0)x_1(n) + h_{0,0}(0)x_0(n) + h_{0,2}(1)x_2(n-1)$. $\hat{x}_1(n) = h_{1,2}(0)x_2(n) + h_{1,1}(0)x_1(n) + h_{1,0}(0)x_0(n) + h_{1,2}(1)x_2(n-1)$. $\hat{x}_2(n) = h_{2,2}(0)x_2(n) + h_{2,1}(0)x_1(n) + h_{2,0}(0)x_0(n) + h_{2,2}(1)x_2(n-1)$. Specifically, $\hat{x}_0(n)$, $\hat{x}_1(n)$, and $\hat{x}_2(n)$ are computed in one batch using the same set of raw data $x_2(n)$, $x_1(n)$, $x_0(n)$, and $x_2(n-1)$. We shall call the structure in Fig. 7 the *batch processing* structure.
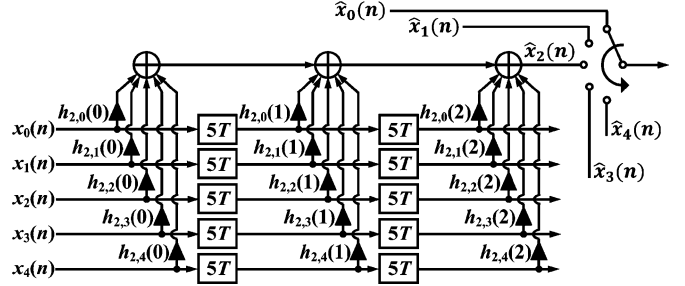
In Fig. 8, $\hat{x}_0(n) = h_{0,0}(0)x_0(n) + h_{0,2}(1)x_2(n-1) + h_{0,1}(1)x_1(n-1) + h_{0,0}(1)x_0(n-1)$. $\hat{x}_1(n) = h_{1,1}(0)x_1(n) + h_{1,0}(0)x_0(n) + h_{1,2}(1)x_2(n-1) + h_{1,1}(1)x_1(n-1)$. $\hat{x}_2(n) = h_{2,2}(0)x_2(n) + h_{2,1}(0)x_1(n) + h_{2,0}(0)x_0(n) + h_{2,2}(1)x_2(n-1)$. The raw data used to compute $\hat{x}_0(n)$, $\hat{x}_1(n)$, and $\hat{x}_2(n)$ are $x_2(n)$, $x_1(n)$, $x_0(n)$, $x_2(n-1)$, $x_1(n-1)$, and $x_0(n-1)$. Since $x_2(n), x_1(n)$, and $x_0(n)$ are available sequentially one after the other, $\hat{x}_2(n)$, $\hat{x}_1(n)$, and $\hat{x}_0(n)$ are computed sequentially one after the other, i.e., the computation of $\hat{x}_2(n)$, $\hat{x}_1(n)$, and $\hat{x}_0(n)$ can be initiated once their respective data are available. We shall call the structure in Fig. 8 the *online processing* structure.

For the same number of coefficients, in general, Fig. 8 computes $\hat{x}_j(n)$ using a longer span of raw data $x_i(n)$. For example, to compute $\hat{x}_0(n)$, $\hat{x}_1(n)$, and $\hat{x}_2(n)$ in Fig. 8, the raw data used are $x_2(n)$, $x_1(n)$, $x_0(n)$, $x_2(n-1)$, $x_1(n-1)$, and $x_0(n-1)$, whereas, in Fig. 7, the raw data used are $x_2(n)$, $x_1(n)$, $x_0(n)$, and $x_2(n-1)$. As a consequence, Fig. 8 has a better mismatch compensation performance than Fig. 7 when $N$ is not significantly larger than $M$.

We have shown that the *online processing* structure (Fig. 8) uses more raw data than the *batch processing* structure (Fig. 7) does for computing $\hat{x}_0(n)$, $\hat{x}_1(n)$, and $\hat{x}_2(n)$ if both the *online processing* and *batch processing* structures have the same number of coefficients. If the *online processing* structure is restricted to using the same number of data for computing
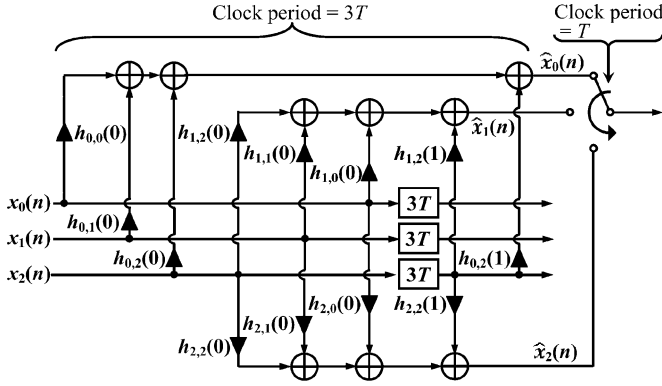
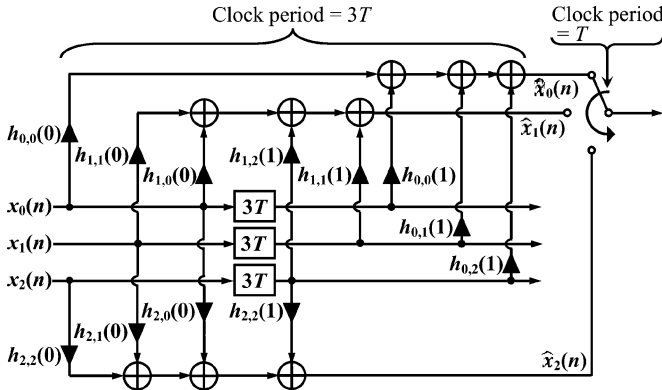Fig. 7. Complete implementation structure for $M = 3$ and $N = 4$ (*batch processing*).

Fig. 8. Complete implementation structure for $M = 3$ and $N = 4$ (*online processing*).

Fig. 9. Complete implementation structure for $M = 3$ and $N = 2$ (*online processing*).

$\hat{x}_0(n)$, $\hat{x}_1(n)$, and $\hat{x}_2(n)$ as the *batch processing* structure does, the *online processing* structure will have less coefficients. For example, if the *online processing* structure is restricted to using only $x_2(n)$, $x_1(n)$, $x_0(n)$, and $x_2(n-1)$ for computing $\hat{x}_0(n)$, $\hat{x}_1(n)$, and $\hat{x}_2(n)$, $N$ will be equal to two, and the structure is shown in Fig. 9. Fig. 9 has six coefficients but Fig. 7 has 12 coefficients. In this case, in general, Fig. 7 has a better mismatch compensation performance than Fig. 9. Nevertheless, the difference in performance is negligibly small if $N$ is significantly larger than $M$. We shall illustrate this by using an example in Section VI.

Let $\bar{\boldsymbol{X}}_m(\mathrm{e}^{jM\omega})$ be the frequency spectrum of the signal vector whose $z$ transform is $\bar{\boldsymbol{X}}_m(z^M)$. Let $V(\omega)$ be the frequency spectrum of the band-limited analog input signal shown in Fig. 5. From Fig. 5, we have

$$\bar{\boldsymbol{X}}_m(\mathrm{e}^{jM\omega}) = V(\omega)\left[\mathrm{e}^{j\omega(M-1)\mathrm{T}}H_{A,M-1}(\omega), \ldots, \mathrm{e}^{j\omega\mathrm{T}}\right.$$
$$\left. \times H_{A,1}(\omega), H_{A,0}(\omega)\right]^{\mathrm{T}}. \quad (11a)$$

In order to simplify the notation, we shall normalize the sampling interval such that $T = 1$. The frequency axis for $V(\omega)$, $\bar{\boldsymbol{X}}_m(\mathrm{e}^{jM\omega})$, and $H_{A,m}(\omega)$, $m = 0, \ldots, M-1$, will also be appropriately scaled by $f_s$. Thus, we write

$$\bar{\boldsymbol{X}}_m(\mathrm{e}^{jM\omega}) = V(\omega)\left[\mathrm{e}^{j\omega(M-1)}H_{A,M-1}(\omega), \ldots, \mathrm{e}^{j\omega}\right.$$
$$\left. \times H_{A,1}(\omega), H_{A,0}(\omega)\right]^{\mathrm{T}}. \quad (11b)$$

Let the frequency spectrum of $\hat{x}_m(n)$ (whose $z$ transform is $\hat{X}_m(z^M)$) be $\hat{X}_m(\mathrm{e}^{jM\omega})$ and let the frequency response of $\boldsymbol{H}_m(z^M)$ be $\boldsymbol{H}_m(\mathrm{e}^{jM\omega})$. Replacing $z$ by $\mathrm{e}^{j\omega}$ in (10), we have

$$\hat{X}_m(\mathrm{e}^{jM\omega}) = \left[\bar{\boldsymbol{X}}_m(\mathrm{e}^{jM\omega})\right]^{\mathrm{T}} \boldsymbol{H}_m(\mathrm{e}^{jM\omega}). \quad (12)$$

From (11b) and (12), we have

$$\hat{X}_m(\mathrm{e}^{jM\omega}) = V(\omega)\left[\mathrm{e}^{j\omega(M-1)}H_{A,M-1}(\omega), \ldots, H_{A,0}(\omega)\right]$$
$$\times \boldsymbol{H}_m(\mathrm{e}^{jM\omega}). \quad (13)$$

We have defined $\hat{x}_m(n)$ to be an estimate of a delayed version of $v(nM + m)$, i.e., $\ddot{x}_m(n - \eta)$. Thus, we wish to find a set of coefficients for $\boldsymbol{H}_m(\mathrm{e}^{jM\omega})$ such that

$$\hat{X}_m(\mathrm{e}^{jM\omega}) \approx V(\omega)\mathrm{e}^{j(m-M\eta)\omega} \quad (14a)$$
$$\hat{X}_m(\mathrm{e}^{jM\omega}) \approx V(\omega)\mathrm{e}^{j(m-M\eta)\omega} + E_m(\mathrm{e}^{jM\omega}). \quad (14b)$$

In (14b), $E_m(\mathrm{e}^{jM\omega})$ is the difference between $\hat{X}_m(\mathrm{e}^{jM\omega})$ and $V(\omega)\mathrm{e}^{j(m-M\eta)\omega}$. From (13) and (14b), we have

$$E_m(\mathrm{e}^{jM\omega}) = \hat{X}_m(\mathrm{e}^{jM\omega}) - V(\omega)\mathrm{e}^{j(m-M\eta)\omega}$$
$$= V(\omega)\left\{\left[\mathrm{e}^{j\omega(M-1)}H_{A,M-1}(\omega), \ldots, H_{A,0}(\omega)\right]\right.$$
$$\left. \times \boldsymbol{H}_m(\mathrm{e}^{jM\omega}) - \mathrm{e}^{j(m-M\eta)\omega}\right\}. \quad (15)$$

In order to clarify the concept, we shall illustrate the term $\left[\mathrm{e}^{j\omega(M-1)}H_{A,M-1}(\omega), \ldots, H_{A,0}(\omega)\right]\boldsymbol{H}_m(\mathrm{e}^{jM\omega})$ using the example shown in Fig. 7 for $m = 0$. In this particular example, $\boldsymbol{H}_0(\mathrm{e}^{j3\omega}) = [h_{0,2}(0), h_{0,1}(0), h_{0,0}(0)]^{\mathrm{T}} + [h_{0,2}(1), 0, 0]^{\mathrm{T}}\mathrm{e}^{-j3\omega}$. Using the definition $\boldsymbol{h}_0 = [h_{0,2}(0), h_{0,1}(0), h_{0,0}(0), h_{0,2}(1)]^{\mathrm{T}}$, $[\mathrm{e}^{j2\omega}H_{A,2}(\omega), \mathrm{e}^{j\omega}H_{A,1}(\omega), H_{A,0}(\omega)]\boldsymbol{H}_0(\mathrm{e}^{j3\omega})$ may be written as $[\mathrm{e}^{j2\omega}H_{A,2}(\omega), \mathrm{e}^{j\omega}H_{A,1}(\omega), H_{A,0}(\omega), \mathrm{e}^{-j\omega}H_{A,2}(\omega)]\boldsymbol{h}_0$. Hence, (15) can be rewritten in terms of $\boldsymbol{h}_m$ as

$$E_m(\mathrm{e}^{jM\omega}) = V(\omega)\left\{[\boldsymbol{\xi}_m(j\omega)]^{\mathrm{T}}\boldsymbol{h}_m - \mathrm{e}^{j(m-M\eta)\omega}\right\}. \quad (16)$$

In the example for Fig. 7, $[\boldsymbol{\xi}_0(j\omega)]^{\mathrm{T}} = [\mathrm{e}^{j2\omega}H_{A,2}(\omega), \mathrm{e}^{j\omega}H_{A,1}(\omega), H_{A,0}(\omega), \mathrm{e}^{-j\omega}H_{A,2}(\omega)]$.

## III. WLS OPTIMIZATION APPROACH

In (16), the factor $V(\omega)$ (which is the frequency spectrum of $v(n)$) corresponds to a weighting function on the optimization

of $E_m(\mathrm{e}^{jM\omega})$. The phase of $V(\omega)$ is unimportant but its magnitude may be of interest. For example, if the highest frequency component of the input signal is $\omega_c$, we may let $V(\omega) = 1$ for $\omega < \omega_c$ and $V(\omega) = 0$ for $\omega > \omega_c$. Let

$$w_{\mathrm{LS}}(\omega) = |V(\omega)|^2. \tag{17}$$

Substituting (17) into (16), we have

$$E_m(\mathrm{e}^{jM\omega}) = \left( [\boldsymbol{\xi}_m(j\omega)]^{\mathrm{T}} \boldsymbol{h}_m - \mathrm{e}^{j(m-M\eta)\omega} \right) | \sqrt{W_{\mathrm{LS}}(\omega)}|. \tag{18}$$

In the WLS approach, (18) is evaluated on a dense grid of frequencies $\omega_i, i = 1, 2, 3, \ldots, L$. Define the complex column vectors $\boldsymbol{E}_m$ and $\boldsymbol{\Lambda}_m$, the complex matrix $\boldsymbol{\Xi}_m$, and the diagonal real matrix $\mathbf{W}$ as follows:

$$\boldsymbol{E}_m = \left[ E_m(\mathrm{e}^{jM\omega_1}), E_m(\mathrm{e}^{jM\omega_2}), \ldots, E_m(\mathrm{e}^{jM\omega_L}) \right]^{\mathrm{T}} \tag{19a}$$

$$\boldsymbol{\Lambda}_m = \left[ \mathrm{e}^{j(m-M\eta)\omega_1}, \mathrm{e}^{j(m-M\eta)\omega_2}, \ldots, \mathrm{e}^{j(m-M\eta)\omega_L} \right]^{\mathrm{T}} \tag{19b}$$

$$\boldsymbol{\Xi}_m = [\boldsymbol{\xi}_m(j\omega_1), \boldsymbol{\xi}_m(j\omega_2), \ldots, \boldsymbol{\xi}_m(j\omega_L)]^{\mathrm{T}} \tag{19c}$$

$$\mathbf{W} = \begin{bmatrix} W_{\mathrm{LS}}(\omega_1) & 0 & \ldots & 0 \\ 0 & W_{\mathrm{LS}}(\omega_2) & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & W_{\mathrm{LS}}(\omega_L) \end{bmatrix}. \tag{19d}$$

Evaluating (18) on a dense grid of $\omega_i, i = 1, 2, 3, \ldots, L$ and using the notation of (19), we have

$$\boldsymbol{E}_m = \sqrt{\mathbf{W}}(\boldsymbol{\Xi}_m \boldsymbol{h}_m - \boldsymbol{\Lambda}_m). \tag{20}$$

$\boldsymbol{E}_m$ and $\boldsymbol{\Lambda}_m$ are $L \times 1$ vectors. $\boldsymbol{\Xi}_m$ and $\mathbf{W}$ are $L \times N$ and $L \times L$ matrices, respectively. The WLS optimum solution of $\boldsymbol{h}_m$ that minimizes $\sum_{i=1}^{L} |E_m(\mathrm{e}^{jM\omega_i})|^2$ is given by [42]

$$\boldsymbol{h}_m = \left( \boldsymbol{\Xi}_m^{\mathrm{H}} \mathbf{W} \boldsymbol{\Xi}_m \right)^{-1} \left( \boldsymbol{\Xi}_m^{\mathrm{H}} \mathbf{W} \boldsymbol{\Lambda}_m \right). \tag{21}$$

In (21), $\boldsymbol{\Xi}_m^{\mathrm{H}}$ denotes the complex conjugate transpose of $\boldsymbol{\Xi}_m$. Solving (21) requires the computation of the inverse of $\boldsymbol{\Xi}_m^{\mathrm{H}} \mathbf{W} \boldsymbol{\Xi}_m$, an $N \times N$ matrix. The filter-bank approach requires the inversion of an $MN \times MN$ matrix [18]. The difference in the size of the matrix to be inversed is significant when $M$ is large. Interestingly, the compensated digital outputs of the filter-bank approach and the multichannel approach are the same.

Note that, for the *batch processing* structure, $\boldsymbol{\xi}_m(j\omega)$ is not a function of $m$. This can be easily verified by evaluating $[\boldsymbol{\xi}_m(j\omega)]$ in Fig. 7 for $m = 0, 1,$ and 2. For Fig. 7, $\boldsymbol{h}_0 = [h_{0,2}(0), h_{0,1}(0), h_{0,0}(0), h_{0,2}(1)]^{\mathrm{T}}, \boldsymbol{h}_1 = [h_{1,2}(0), h_{1,1}(0), h_{1,0}(0), h_{1,2}(1)]^{\mathrm{T}},$ and $\boldsymbol{h}_2 = [h_{2,2}(0), h_{2,1}(0), h_{2,0}(0), h_{2,2}(1)]^{\mathrm{T}}.$ Thus, we have $[\boldsymbol{\xi}_0(j\omega)]^{\mathrm{T}} = [\boldsymbol{\xi}_1(j\omega)]^{\mathrm{T}} = [\boldsymbol{\xi}_2(j\omega)]^{\mathrm{T}} = [\mathrm{e}^{j2\omega}H_{A,2}(\omega), \mathrm{e}^{j\omega}H_{A,1}(\omega), H_{A,0}(\omega), \mathrm{e}^{-j\omega}H_{A,2}(\omega)].$ Since $\boldsymbol{\xi}_m(j\omega)$ is not a function of $m, \boldsymbol{\Xi}_m$ is also not a function of $m$. This means that $(\boldsymbol{\Xi}_m^{\mathrm{H}} \mathbf{W} \boldsymbol{\Xi}_m)^{-1}$ is the same for all $m$. Thus, for the *batch processing* structure, evaluating $\boldsymbol{h}_m$ for all $m$ using (21) requires only one matrix inversion.

For the *online processing* structure, $\boldsymbol{\xi}_m(j\omega)$ is a function of $m$. This can be easily verified by evaluating $\boldsymbol{\xi}_m(j\omega)$ in Fig. 8 for $m = 0, 1,$ and 2. For Fig. 8, $\boldsymbol{h}_0 = [h_{0,0}(0), h_{0,2}(1), h_{0,1}(1), h_{0,0}(1)]^{\mathrm{T}}, \boldsymbol{h}_1 =$

### TABLE II
#### COMPUTATIONAL COMPLEXITY OF FILTER DESIGN

| Our multichannel approach | | Filter-bank approach | |
| --- | --- | --- | --- |
| Matrices | Number of multiplications | Matrices | Number of multiplications |
| $\boldsymbol{\Xi}_m^{\mathrm{H}} \mathbf{W} \boldsymbol{\Xi}_m$ | $2(LN^2 + LN)$ | $\mathbf{A}^{\mathrm{H}} \mathbf{W} \mathbf{A}$ | $2(LN^2M^3 + LNM^2)$ |
| $\boldsymbol{\Xi}_m^{\mathrm{H}} \mathbf{W} \boldsymbol{\Lambda}_m$ | $2(LN + L)$ | $\mathbf{A}^{\mathrm{H}} \mathbf{W} \mathbf{D}$ | $2(LNM^2 + LM)$ |
| $\left( \boldsymbol{\Xi}_m^{\mathrm{H}} \mathbf{W} \boldsymbol{\Xi}_m \right)^{-1}$ | $O(N^2)$ | $\left( \mathbf{A}^{\mathrm{H}} \mathbf{W} \mathbf{A} \right)^{-1}$ | $O((NM)^2)$ |

$[h_{1,1}(0), h_{1,0}(0), h_{1,2}(1), h_{1,1}(1)]^{\mathrm{T}},$ and $\boldsymbol{h}_2 = [h_{2,2}(0), h_{2,1}(0), h_{2,0}(0), h_{2,2}(1)]^{\mathrm{T}}.$ Thus, we have $[\boldsymbol{\xi}_0(j\omega)]^{\mathrm{T}} = [H_{A,0}(\omega), \mathrm{e}^{-j\omega}H_{A,2}(\omega), \mathrm{e}^{-2j\omega}H_{A,1}(\omega), \mathrm{e}^{-3j\omega}H_{A,0}(\omega)]$, $[\boldsymbol{\xi}_1(j\omega)]^{\mathrm{T}} = [\mathrm{e}^{j\omega}H_{A,1}(\omega), H_{A,0}(\omega), \mathrm{e}^{-j\omega}H_{A,2}(\omega), \mathrm{e}^{-2j\omega}H_{A,1}(\omega)],$ and $[\boldsymbol{\xi}_2(j\omega)]^{\mathrm{T}} = [\mathrm{e}^{j2\omega}H_{A,2}(\omega), \mathrm{e}^{j\omega}H_{A,1}(\omega), H_{A,0}(\omega), \mathrm{e}^{-j\omega}h_{A,2}(\omega)].$ Since $\boldsymbol{\xi}_m(j\omega)$ is a function of $m, \boldsymbol{\Xi}_m$ is also a function of $m$. This means that $(\boldsymbol{\Xi}_m^{\mathrm{H}} \mathbf{W} \boldsymbol{\Xi}_m)^{-1}$ is different for different $m$'s. Thus, for the *online processing* structure, evaluating $\boldsymbol{h}_m$ for all $m$ using (21) requires $M$ matrix inversions.

## IV. COMPUTATIONAL COMPLEXITY OF FILTER DESIGN

We shall compare the computational complexity for evaluating (21) in our approach and that for evaluating (4) in the filter-bank approach. The comparison is based on the same total number of coefficients $NM$. Simulation results show that the two approaches gives the same TIADC spurious frequency performance when both (21) and (4) are evaluated with sufficient arithmetic precision.

In (21), $\boldsymbol{\Xi}_m$ is $L \times N$. The corresponding matrix in (4) is $\mathbf{A}$ which is $LM \times NM$. The computation of $\boldsymbol{\Xi}_m^{\mathrm{H}} \mathbf{W} \boldsymbol{\Xi}_m$ in (21) requires $2(LN^2 + LN)$ multiplications. $\boldsymbol{\Xi}_m^{\mathrm{H}} \mathbf{W} \boldsymbol{\Xi}_m$ in our multichannel approach corresponds to $\mathbf{A}^{\mathrm{H}} \mathbf{W} \mathbf{A}$ in the filter-bank approach. The computation of the $\mathbf{A}^{\mathrm{H}} \mathbf{W} \mathbf{A}$ matrix in (4) requires $2(LN^2M^3 + LNM^2)$ multiplications. The computation of $\boldsymbol{\Xi}_m^{\mathrm{H}} \mathbf{W} \boldsymbol{\Lambda}_m$ in (21) requires $2(LN + L)$ multiplications. The corresponding computation in the filter-bank approach is the computation of $\mathbf{A}^{\mathrm{H}} \mathbf{W} \mathbf{D}$ in (4) and requires $2(LNM^2 + LM)$ multiplications. $\boldsymbol{\Xi}_m^{\mathrm{H}} \mathbf{W} \boldsymbol{\Xi}_m$ in (21) is an $N \times N$ matrix whose inverse can be computed with complexity $O(N^2)$ (order $N^2$). The $\mathbf{A}^{\mathrm{H}} \mathbf{W} \mathbf{A}$ matrix in (4) is an $NM \times NM$ matrix whose inverse can be computed with complexity $O((NM)^2)$. If the implementation structure is that in Fig. 7, $\boldsymbol{\Xi}_m$ is not a function of $m$, and therefore, it is necessary to compute the inverse $(\boldsymbol{\Xi}_m^{\mathrm{H}} \mathbf{W} \boldsymbol{\Xi}_m)^{-1}$ only once; otherwise, it is necessary to compute the inverse $M$ times. The computational complexity is summarized in Table II. From the aforementioned analysis, it is clear that solving (21) is significantly less complex than solving (4).

## V. COMPUTATIONAL ROBUSTNESS

We have shown the computational advantage of our multichannel technique over that of the filter-bank technique in terms of computational complexity in Section IV. In this section, we shall demonstrate, using an example, that the computational robustness of solving (21) is greater than the computational robustness of solving (4).
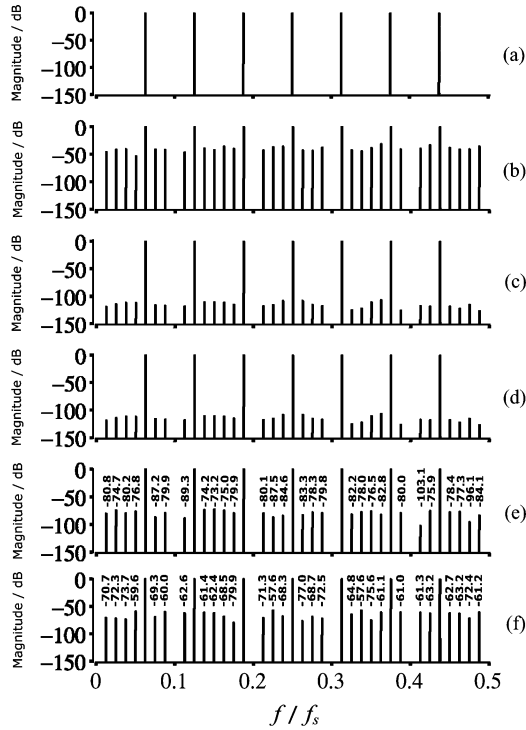
Fig. 10. (a) Input. (b) Uncompensated. (c) Multichannel filter (64-bit arithmetic). (d) Filter bank (64-bit arithmetic). (e) Multichannel filter (32-bit arithmetic). (f) Filter bank (32-bit arithmetic).



Fig. 11. (a) Input. (b) Uncompensated. (c) Filter of Table V. (d) Filter of Table VI.

*Example-I:* Consider an example with $M = 10$ and $N = 41$. The mismatch parameters $H_{A,m}(\omega)$, $m = 0, 1, \ldots, 9$, are tabulated in Table III. In Table III, the $e^{j\omega\bullet}$ factor corresponds to timing mismatch, and the $(1 + j\omega\bullet)$ factor corresponds to bandwidth mismatch. The multichannel filters and the filter bank were optimized in the WLS sense with uniform weight to compensate the mismatches for frequencies ranging from 0 to $0.45f_s$. The optimization processes were done using 32- and 64-bit floating point arithmetic for comparison. The timing of ADC-0 was taken as timing reference. Fig. 10 shows the simulation results when the TIADC was excited by seven sinusoids with frequencies $0.0625mf_s$, $m = 1, 2, \ldots, 7$. Fig. 10(a) shows the spectral lines of the input sinusoids. Fig. 10(b) shows the spectral lines of the uncompensated TIADC output. Fig. 10(c) and (d) shows the compensated output using multichannel filters and filter banks, respectively, both optimized using 64-bit floating point arithmetic. It can be seen from Fig. 10(c) and (d) that both methods produce filters that have the same performance; the largest spurious component has a magnitude of $-107.0$ dB occurring at $0.3625f_s$. Fig. 10(e) shows the spectral lines for the multichannel approach when the filters were optimized using 32-bit floating point arithmetic; the magnitudes of the spurious components (in decibels) are shown directly above the components. The largest spurious component has a magnitude of $-73.2$ dB. Comparing Fig. 10(c) and (e), it can be seen that there is significant degradation in performance when the filters are optimized using 32-bit floating point arithmetic. Fig. 10(f) shows the spectral lines for the filter-bank approach when the filter bank was optimized using 32-bit floating point arithmetic; the magnitudes of the spurious components (in decibels) are shown directly above the
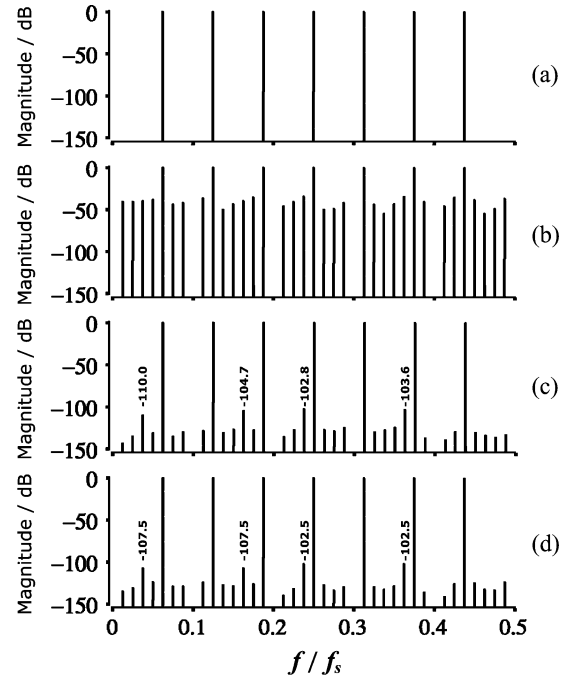
TABLE III
SUB-ADC MISMATCH PARAMETERS FOR *EXAMPLE-I*

| $H_{A,0}(\omega) = \dfrac{1}{1 + j\dfrac{\omega}{2\pi f_s}}$ | $H_{A,1}(\omega) = \dfrac{e^{j0.03\omega/f_s}}{1 + j\dfrac{1.2\omega}{2\pi f_s}}$ | $H_{A,2}(\omega) = \dfrac{e^{j0.05\omega/f_s}}{1 + j\dfrac{1.3\omega}{2\pi f_s}}$ |
|---|---|---|
| $H_{A,3}(\omega) = \dfrac{e^{-j0.04\omega/f_s}}{1 + j\dfrac{0.8\omega}{2\pi f_s}}$ | $H_{A,4}(\omega) = \dfrac{e^{-j0.05\omega/f_s}}{1 + j\dfrac{0.7\omega}{2\pi f_s}}$ | $H_{A,5}(\omega) = \dfrac{e^{j0.02\omega/f_s}}{1 + j\dfrac{1.05\omega}{2\pi f_s}}$ |
| $H_{A,6}(\omega) = \dfrac{e^{j0.03\omega/f_s}}{1 + j\dfrac{1.15\omega}{2\pi f_s}}$ | $H_{A,7}(\omega) = \dfrac{e^{j0.04\omega/f_s}}{1 + j\dfrac{1.2\omega}{2\pi f_s}}$ | $H_{A,8}(\omega) = \dfrac{e^{-j0.03\omega/f_s}}{1 + j\dfrac{0.85\omega}{2\pi f_s}}$ |
| $H_{A,9}(\omega) = \dfrac{e^{-j0.02\omega/f_s}}{1 + j\dfrac{0.75\omega}{2\pi f_s}}$ | | |

components. The largest spurious component has a magnitude of $-57.6$ dB. Comparing Fig. 10(e) and (f), it can be seen that the performance of the multichannel approach is, on average, better than the performance of the filter-bank approach. This is expected because the optimization of the filter-bank requires the inversion of a $410 \times 410$ matrix whereas the optimization of the multichannel filter requires the inversion of a $41 \times 41$ matrix; a larger matrix not only takes a longer time to invert but the computation is also more demanding in terms of arithmetic precision.

## VI. EXAMPLES

*Example-II:* We shall compare the performance of the batch processing structure in Fig. 7 and that of the on-line processing structure in Fig. 8 using an example with $M = 5$ and $N = 41$. The bandwidth mismatch parameters $H_{A,m}(\omega)$, $m = 0, 1, \ldots, 4$, are shown in Table IV. The filters are optimized in the WLS sense with uniform weight over the frequency range from 0 to $0.4375f_s$. For the structure in Fig. 7, the optimization involves the inversion of one $41 \times 41$

TABLE IV
SUB-ADC BANDWIDTH MISMATCH PARAMETERS FOR *EXAMPLE-II*

| | | |
|---|---|---|
| $H_{A,0}(\omega) = \dfrac{1}{1+j\frac{\omega}{2\pi f_s}}$ | $H_{A,1}(\omega) = \dfrac{1}{1+j\frac{1.07\omega}{2\pi f_s}}$ | $H_{A,2}(\omega) = \dfrac{1}{1+j\frac{1.15\omega}{2\pi f_s}}$ |
| $H_{A,3}(\omega) = \dfrac{1}{1+j\frac{0.93\omega}{2\pi f_s}}$ | $H_{A,4}(\omega) = \dfrac{1}{1+j\frac{0.95\omega}{2\pi f_s}}$ | |

TABLE V
COEFFICIENT VALUES FOR *EXAMPLE-II* SYNTHESIZED USING THE STRUCTURE IN FIG. 7

| | $k=0$ | $k=1$ | $k=2$ | $k=3$ | $k=4$ |
|---|---|---|---|---|---|
| $h_{k,2}(0)$ | 0 | 0.00000158 | -0.00000345 | -0.00000181 | 0.00000132 |
| $h_{k,1}(0)$ | 0 | -0.00000440 | 0.00000966 | 0.00000511 | -0.00000373 |
| $h_{k,0}(0)$ | 0 | 0.00000935 | -0.00002062 | -0.00001097 | 0.00000804 |
| $h_{k,4}(1)$ | 0 | -0.00001722 | 0.00003816 | 0.00002040 | -0.00001503 |
| $h_{k,3}(1)$ | 0 | 0.00002888 | -0.00006429 | -0.00003454 | 0.00002559 |
| $h_{k,2}(1)$ | 0 | -0.00004811 | 0.00010763 | 0.00005815 | -0.00004335 |
| $h_{k,1}(1)$ | 0 | 0.00007595 | -0.00017077 | -0.00009280 | 0.00006964 |
| $h_{k,0}(1)$ | 0 | -0.00011122 | 0.00025152 | 0.00013757 | -0.00010402 |
| $h_{k,4}(2)$ | 0 | 0.00015580 | -0.00035460 | -0.00019540 | 0.00014907 |
| $h_{k,3}(2)$ | 0 | -0.00021079 | 0.00048332 | 0.00026867 | -0.00020712 |
| $h_{k,2}(2)$ | 0 | 0.00029436 | -0.00068088 | -0.00038252 | 0.00029873 |
| $h_{k,1}(2)$ | 0 | -0.00040448 | 0.00094498 | 0.00053743 | -0.00042622 |
| $h_{k,0}(2)$ | 0 | 0.00052713 | -0.00124663 | -0.00071997 | 0.00058250 |
| $h_{k,4}(3)$ | 0 | -0.00067070 | 0.00161055 | 0.00094877 | -0.00078849 |
| $h_{k,3}(3)$ | 0 | 0.00084079 | -0.00205915 | -0.00124587 | 0.00107567 |
| $h_{k,2}(3)$ | 0 | -0.00111041 | 0.00279400 | 0.00175732 | -0.00161127 |
| $h_{k,1}(3)$ | 0 | 0.00148279 | -0.00387038 | -0.00257735 | 0.00261401 |
| $h_{k,0}(3)$ | 0 | -0.00193856 | 0.00535572 | 0.00394337 | -0.00502143 |
| $h_{k,4}(4)$ | 0 | 0.00259969 | -0.00793349 | -0.00732129 | 0.99628141 |
| $h_{k,3}(4)$ | 0 | -0.00375605 | 0.01433994 | 0.99521835 | 0.00938062 |
| $h_{k,2}(4)$ | 0 | 0.00712824 | 1.01215186 | 0.01324679 | -0.00371944 |
| $h_{k,1}(4)$ | 0 | 1.00501356 | -0.02851757 | -0.00561884 | 0.00246461 |
| $h_{k,0}(4)$ | 1 | -0.01290611 | 0.01065066 | 0.00343283 | -0.00175014 |
| $h_{k,4}(5)$ | 0 | 0.00466108 | -0.00628853 | -0.00234741 | 0.00129291 |
| $h_{k,3}(5)$ | 0 | -0.00267230 | 0.00417064 | 0.00168032 | -0.00097054 |
| $h_{k,2}(5)$ | 0 | 0.00181013 | -0.00304988 | -0.00128874 | 0.00076922 |
| $h_{k,1}(5)$ | 0 | -0.00134440 | 0.00236771 | 0.00103150 | -0.00062969 |
| $h_{k,0}(5)$ | 0 | 0.00097422 | -0.00176986 | -0.00078883 | 0.00049010 |
| $h_{k,4}(6)$ | 0 | -0.00070083 | 0.00130251 | 0.00059081 | -0.00037225 |
| $h_{k,3}(6)$ | 0 | 0.00049807 | -0.00094201 | -0.00043330 | 0.00027616 |
| $h_{k,2}(6)$ | 0 | -0.00036570 | 0.00070150 | 0.00032645 | -0.00021011 |
| $h_{k,1}(6)$ | 0 | 0.00027411 | -0.00053156 | -0.00024964 | 0.00016194 |
| $h_{k,0}(6)$ | 0 | -0.00019218 | 0.00037617 | 0.00017808 | -0.00011633 |
| $h_{k,4}(7)$ | 0 | 0.00012935 | -0.00025522 | -0.00012167 | 0.00007997 |
| $h_{k,3}(7)$ | 0 | -0.00008343 | 0.00016578 | 0.00007953 | -0.00005256 |
| $h_{k,2}(7)$ | 0 | 0.00005374 | -0.00010747 | -0.00005185 | 0.00003445 |
| $h_{k,1}(7)$ | 0 | -0.00003440 | 0.00006917 | 0.00003354 | -0.00002238 |
| $h_{k,0}(7)$ | 0 | 0.00001967 | -0.00003975 | -0.00001936 | 0.00001298 |
| $h_{k,4}(8)$ | 0 | -0.00001012 | 0.00002055 | 0.00001006 | -0.00000677 |
| $h_{k,3}(8)$ | 0 | 0.00000448 | -0.00000915 | -0.00000450 | 0.00000304 |
| $h_{k,2}(8)$ | 0 | -0.00000155 | 0.00000319 | 0.00000158 | -0.00000107 |

TABLE VI
COEFFICIENT VALUES FOR *EXAMPLE-II* SYNTHESIZED USING THE STRUCTURE IN FIG. 8

| | $k=0$ | $k=1$ | $k=2$ | $k=3$ | $k=4$ |
|---|---|---|---|---|---|
| $h_{k,[k+0]}(0)$ | 0 | -0.00000158 | -0.00000345 | 0.00000163 | 0.00000114 |
| $h_{k,[k+4]}(1-I(k+4))$ | 0 | 0.00000438 | 0.00000966 | -0.00000475 | -0.00000311 |
| $h_{k,[k+3]}(1-I(k+3))$ | 0 | -0.00000931 | -0.00002062 | 0.00001047 | 0.00000699 |
| $h_{k,[k+2]}(1-I(k+2))$ | 0 | 0.00001716 | 0.00003816 | -0.00001960 | -0.00001352 |
| $h_{k,[k+1]}(1-I(k+1))$ | 0 | -0.00003071 | -0.00006429 | 0.00003326 | 0.00002319 |
| $h_{k,[k+0]}(1)$ | 0 | 0.00005116 | 0.00010763 | -0.00005260 | -0.00003692 |
| $h_{k,[k+4]}(2-I(k+4))$ | 0 | -0.00007840 | -0.00017077 | 0.00008381 | 0.00005558 |
| $h_{k,[k+3]}(2-I(k+3))$ | 0 | 0.00011419 | 0.00025152 | -0.00012805 | -0.00008509 |
| $h_{k,[k+2]}(2-I(k+2))$ | 0 | -0.00015991 | -0.00035460 | 0.00018278 | 0.00012600 |
| $h_{k,[k+1]}(2-I(k+1))$ | 0 | 0.00023044 | 0.00048332 | -0.00025108 | -0.00017521 |
| $h_{k,[k+0]}(2)$ | 0 | -0.00032543 | -0.00068088 | 0.00033500 | 0.00023553 |
| $h_{k,[k+4]}(3-I(k+4))$ | 0 | 0.00043510 | 0.00094498 | -0.00046382 | -0.00030882 |
| $h_{k,[k+3]}(3-I(k+3))$ | 0 | -0.00056709 | -0.00124663 | 0.00063581 | 0.00042192 |
| $h_{k,[k+2]}(3-I(k+2))$ | 0 | 0.00072734 | 0.00161055 | -0.00083213 | -0.00057374 |
| $h_{k,[k+1]}(3-I(k+1))$ | 0 | -0.00098216 | -0.00205915 | 0.00107256 | 0.00074900 |
| $h_{k,[k+0]}(3)$ | 0 | 0.00133856 | 0.00279400 | -0.00137884 | -0.00097048 |
| $h_{k,[k+4]}(4-I(k+4))$ | 0 | -0.00178548 | -0.00387038 | 0.00190481 | 0.00126938 |
| $h_{k,[k+3]}(4-I(k+3))$ | 0 | 0.00244208 | 0.00535572 | -0.00274035 | -0.00182418 |
| $h_{k,[k+2]}(4-I(k+2))$ | 0 | -0.00359786 | -0.00793349 | 0.00411341 | 0.00284736 |
| $h_{k,[k+1]}(4-I(k+1))$ | 0 | 0.00696431 | 0.01433994 | -0.00749339 | -0.00526285 |
| $h_{k,[k+0]}(4)$ | 1 | 1.00518355 | 1.01215186 | 0.99538816 | 0.99652376 |
| $h_{k,[k+4]}(5-I(k+4))$ | 0 | -0.01307242 | -0.02851757 | 0.01307384 | 0.00914343 |
| $h_{k,[k+3]}(5-I(k+3))$ | 0 | 0.00481892 | 0.01065066 | -0.00544242 | -0.00347984 |
| $h_{k,[k+2]}(5-I(k+2))$ | 0 | -0.00281823 | -0.00628853 | 0.00326312 | 0.00222212 |
| $h_{k,[k+1]}(5-I(k+1))$ | 0 | 0.00194914 | 0.00417064 | -0.00218908 | -0.00151872 |
| $h_{k,[k+0]}(5)$ | 0 | -0.00147731 | -0.00304988 | 0.00153648 | 0.00107876 |
| $h_{k,[k+4]}(6-I(k+4))$ | 0 | 0.00109378 | 0.00236771 | -0.00115423 | -0.00077759 |
| $h_{k,[k+3]}(6-I(k+3))$ | 0 | -0.00080488 | -0.00176986 | 0.00090518 | 0.00059033 |
| $h_{k,[k+2]}(6-I(k+2))$ | 0 | 0.00058600 | 0.00130251 | -0.00067734 | -0.00046312 |
| $h_{k,[k+1]}(6-I(k+1))$ | 0 | -0.00044182 | -0.00094201 | 0.00049570 | 0.00034438 |
| $h_{k,[k+0]}(6)$ | 0 | 0.00034028 | 0.00070150 | -0.00035460 | -0.00024910 |
| $h_{k,[k+4]}(7-I(k+4))$ | 0 | -0.00024592 | -0.00053156 | 0.00025988 | 0.00017527 |
| $h_{k,[k+3]}(7-I(k+3))$ | 0 | 0.00017125 | 0.00037617 | -0.00019312 | -0.00012568 |
| $h_{k,[k+2]}(7-I(k+2))$ | 0 | -0.00011486 | -0.00025522 | 0.00013335 | 0.00009104 |
| $h_{k,[k+1]}(7-I(k+1))$ | 0 | 0.00007753 | 0.00016578 | -0.00008779 | -0.00006090 |
| $h_{k,[k+0]}(7)$ | 0 | -0.00005236 | -0.00010747 | 0.00005493 | 0.00003855 |
| $h_{k,[k+4]}(8-I(k+4))$ | 0 | 0.00003206 | 0.00006917 | -0.00003393 | -0.00002295 |
| $h_{k,[k+3]}(8-I(k+3))$ | 0 | -0.00001810 | -0.00003975 | 0.00002058 | 0.00001326 |
| $h_{k,[k+2]}(8-I(k+2))$ | 0 | 0.00000923 | 0.00002055 | -0.00001088 | -0.00000737 |
| $h_{k,[k+1]}(8-I(k+1))$ | 0 | -0.00000421 | -0.00000915 | 0.00000496 | 0.00000340 |
| $h_{k,[k+0]}(8)$ | 0 | 0.00000158 | 0.00000319 | -0.00000174 | -0.00000120 |

TABLE VII
SUB-ADC TIMING MISMATCH PARAMETERS FOR *EXAMPLE-III*

| | | |
|---|---|---|
| $H_{A,0}(\omega) = 1$ | $H_{A,1}(\omega) = e^{j0.02\omega/f_s}$ | $H_{A,2}(\omega) = e^{j0.0001\omega/f_s}$ |
| $H_{A,3}(\omega) = e^{-j0.0001\omega/f_s}$ | $H_{A,4}(\omega) = e^{j0.0001\omega/f_s}$ | |

processing structure of Fig. 7 are shown in Table V. The coefficient values for the online processing structure in Fig. 8 are shown in Table VI. For $h_{[k+i]}(j - I(k+r))$ in Table VI (and also Table VIII), the subscript $[k+i]$ is $(k+i)$ modulo 5. $I(k+r)$ is the integer part of $(k+r)/5$. The rms values of $|E_m(e^{jM\omega})|$ for the filters of Tables V and VI are $1.23 \times 10^{-6}$ and $1.22 \times 10^{-6}$, respectively; the difference is negligible. In general, for $N \gg M$, the difference is small. Note the difference in the subscripts for $h_{k,i}(n)$ and the location of the coefficient whose magnitude is approximately equal to unity in Tables V and VI. Fig. 11 shows the simulation results when the TIADC was excited by seven sinusoids with frequencies

matrix since $\Xi_m$ is independent of $m$. For the structure in Fig. 8, the optimization involves the inversion of four $41 \times 41$ matrices since $\Xi_m$ is different for each $m$. The coefficient values for the multichannel filters synthesized using the batch
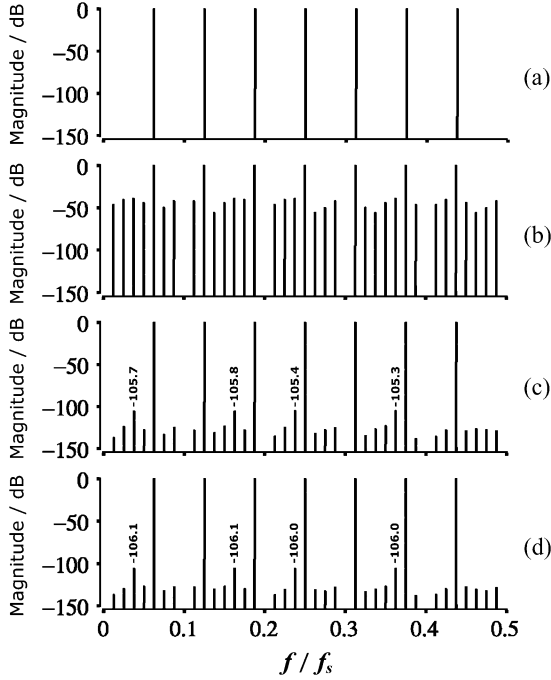
Fig. 12. (a) Input. (b) Uncompensated. (c) Compensation with $N_0 = 1$, $N_1 = 41$, $N_2 = N_3 = N_4 = 21$. (d) Compensation with $N_0 = 1$, $N_1 = N_2 = N_3 = N_4 = 41$.

TABLE VIII
COEFFICIENT VALUES FOR *EXAMPLE-III*

| | $k = 0$ | $k = 1$ | $k = 2$ | $k = 3$ | $k = 4$ |
|---|---|---|---|---|---|
| $h_{k,|k+0|}(0)$ | 0 | 0.00000362 | 0 | 0 | 0 |
| $h_{k,|k+4|}(1\text{-}I(k+4))$ | 0 | -0.00001008 | 0 | 0 | 0 |
| $h_{k,|k+3|}(1\text{-}I(k+3))$ | 0 | 0.00002205 | 0 | 0 | 0 |
| $h_{k,|k+2|}(1\text{-}I(k+2))$ | 0 | -0.00004200 | 0 | 0 | 0 |
| $h_{k,|k+1|}(1\text{-}I(k+1))$ | 0 | 0.00007366 | 0 | 0 | 0 |
| $h_{k,|k+0|}(1)$ | 0 | -0.00011656 | 0 | 0 | 0 |
| $h_{k,|k+4|}(2\text{-}I(k+4))$ | 0 | 0.00017877 | 0 | 0 | 0 |
| $h_{k,|k+3|}(2\text{-}I(k+3))$ | 0 | -0.00026939 | 0 | 0 | 0 |
| $h_{k,|k+2|}(2\text{-}I(k+2))$ | 0 | 0.00039143 | 0 | 0 | 0 |
| $h_{k,|k+1|}(2\text{-}I(k+1))$ | 0 | -0.00055690 | 0 | 0 | 0 |
| $h_{k,|k+0|}(2)$ | 0 | 0.00074780 | 0.00000094 | -0.00000095 | 0.00000095 |
| $h_{k,|k+4|}(3\text{-}I(k+4))$ | 0 | -0.00099897 | -0.00000188 | 0.00000197 | -0.00000196 |
| $h_{k,|k+3|}(3\text{-}I(k+3))$ | 0 | 0.00135053 | 0.00000330 | -0.00000338 | 0.00000350 |
| $h_{k,|k+2|}(3\text{-}I(k+2))$ | 0 | -0.00180385 | -0.00000542 | 0.00000543 | -0.00000552 |
| $h_{k,|k+1|}(3\text{-}I(k+1))$ | 0 | 0.00241848 | 0.00000839 | -0.00000844 | 0.00000841 |
| $h_{k,|k+0|}(3)$ | 0 | -0.00315008 | -0.00001270 | 0.00001270 | -0.00001271 |
| $h_{k,|k+4|}(4\text{-}I(k+4))$ | 0 | 0.00423607 | 0.00001834 | -0.00001906 | 0.00001897 |
| $h_{k,|k+3|}(4\text{-}I(k+3))$ | 0 | -0.00610797 | -0.00002750 | 0.00002813 | -0.00002910 |
| $h_{k,|k+2|}(4\text{-}I(k+2))$ | 0 | 0.00967727 | 0.00004566 | -0.00004569 | 0.00004635 |
| $h_{k,|k+1|}(4\text{-}I(k+1))$ | 0 | -0.02016488 | -0.00009701 | 0.00009750 | -0.00009705 |
| $h_{k,|k+0|}(4)$ | 1 | 1.00059755 | 0.99999772 | 1.00000088 | 0.99999918 |
| $h_{k,|k+4|}(5\text{-}I(k+4))$ | 0 | 0.01948480 | 0.00009864 | -0.00009959 | 0.00009906 |
| $h_{k,|k+3|}(5\text{-}I(k+3))$ | 0 | -0.00960188 | -0.00004502 | 0.00004697 | -0.00004763 |
| $h_{k,|k+2|}(5\text{-}I(k+2))$ | 0 | 0.00616233 | 0.00002782 | -0.00002783 | 0.00002881 |
| $h_{k,|k+1|}(5\text{-}I(k+1))$ | 0 | -0.00438910 | -0.00001851 | 0.00001861 | -0.00001852 |
| $h_{k,|k+0|}(5)$ | 0 | 0.00319081 | 0.00001266 | -0.00001266 | 0.00001266 |
| $h_{k,|k+4|}(6\text{-}I(k+4))$ | 0 | -0.00233930 | -0.00000840 | 0.00000859 | -0.00000855 |
| $h_{k,|k+3|}(6\text{-}I(k+3))$ | 0 | 0.00179182 | 0.00000532 | -0.00000554 | 0.00000564 |
| $h_{k,|k+2|}(6\text{-}I(k+2))$ | 0 | -0.00136416 | -0.00000332 | 0.00000333 | -0.00000346 |
| $h_{k,|k+1|}(6\text{-}I(k+1))$ | 0 | 0.00103635 | 0.00000189 | -0.00000191 | 0.00000191 |
| $h_{k,|k+0|}(6)$ | 0 | -0.00075891 | -0.00000093 | 0.00000094 | -0.00000094 |
| $h_{k,|k+4|}(7\text{-}I(k+4))$ | 0 | 0.00053945 | 0 | 0 | 0 |
| $h_{k,|k+3|}(7\text{-}I(k+3))$ | 0 | -0.00038944 | 0 | 0 | 0 |
| $h_{k,|k+2|}(7\text{-}I(k+2))$ | 0 | 0.00027260 | 0 | 0 | 0 |
| $h_{k,|k+1|}(7\text{-}I(k+1))$ | 0 | -0.00018584 | 0 | 0 | 0 |
| $h_{k,|k+0|}(7)$ | 0 | 0.00011928 | 0 | 0 | 0 |
| $h_{k,|k+4|}(8\text{-}I(k+4))$ | 0 | -0.00007156 | 0 | 0 | 0 |
| $h_{k,|k+3|}(8\text{-}I(k+3))$ | 0 | 0.00004194 | 0 | 0 | 0 |
| $h_{k,|k+2|}(8\text{-}I(k+2))$ | 0 | -0.00002242 | 0 | 0 | 0 |
| $h_{k,|k+1|}(8\text{-}I(k+1))$ | 0 | 0.00001056 | 0 | 0 | 0 |
| $h_{k,|k+0|}(8)$ | 0 | -0.00000388 | 0 | 0 | 0 |

$0.0625mf_s$, $m = 1, 2, \ldots, 7$. Fig. 11(a) and (b) shows the spectral lines of the input sinusoids and the uncompensated TIADC output, respectively. Fig. 11(c) and (d) shows the spectral lines of the compensated outputs for the filter of Table V and that of Table VI, respectively. The magnitudes (in decibels) of the larger spurious components are also marked above them; the difference is insignificant. This is usually so for $N \gg M$.

*Example-III:* The multichannel approach has an advantage that $N_m$ need not be the same for all $m$. If the mismatch of ADC-$i$ is less serious than that of ADC-$j$, $N_i$ may be less than $N_j$. For example, if ADC-0 is taken as reference, it is assumed to have no mismatch. Thus, we choose $N_0 = 1$. This can be seen in the coefficient values under the column $k = 0$ tabulated in Tables V and VI.

For the filter-bank approach, the length of FILTER-0 cannot be reduced even though ADC-0 is assumed to be perfect. Although dividing the transfer functions of FILTER-$i$ by the transfer function of FILTER-0 will make the transfer function of FILTER-0 unity, it will make FILTER-$i$, $i \neq 0$, become a recursive filter; the total number of distinct coefficients remains unchanged, and the filter becomes unstable if FILTER-0 has a zero outside the unit circle.

We shall illustrate this advantage of the multichannel approach using an example with timing mismatch parameters as shown in Table VII. It can be seen from Table VII that the mismatch of ADC-1 is significantly more serious than the others. Thus, we choose $N_1 = 41$, $N_2 = N_3 = N_4 = 21$. We choose $N_0 = 1$ since ADC-0 is taken as reference. The coefficient values are shown in Table VIII, and the spectral plots are shown in Fig. 12. For comparison, the spectral plot for a design with $N_1 = N_2 = N_3 = N_4 = 41$ is shown in Fig. 12(d). Comparing Fig. 12(c) and (d), it can be seen that the degradation in performance is negligible for a large saving in the number of nontrivial coefficient values.

## VII. MULTIPLIERLESS TIADC DESIGN USING MILP

A TIADC with photonic front end is able to sample at a very high sampling rate well above 100 GS/s (gigasamples per second) [43]. The analog input may be sampled using a mode-locked laser and stretched in an optical fiber until it is slow enough for resampling using a high-speed electronic device. In the implementation of very high-sampling-rate TIADCs, it is necessary to reduce the computational complexity of the compensation filters. The building blocks of a compensation filter are the coefficient multiplier, adder, and delay. Among these building blocks, the coefficient multiplier is the one that consumes the most power, occupies the largest silicon area, and is slowest in speed. Thus, it is essential to reduce the complexity of the coefficient multiplier.

In binary arithmetic, multiplying a number by an integer power of two is a trivial process; it can be realized by wiring without using any active device in a full custom implementation in silicon. Consequently, if each coefficient value of a compensation filter is expressed as a sum of a limited number of signed power of two (SPT) terms [44]–[50], the resulting filter is essentially "multiplierless" from an implementation point of view. Various methods have been developed for optimizing the coefficient values in the SPT coefficient space for a digital filter. Among these techniques, MILP [51] is the only method that can guarantee the obtaining of the optimum solution for a given number of SPT terms allocated to each coefficient. Discrete space optimization is a time-consuming process. The computing time required increases exponentially with the number of discrete variables. From the statistics given in [49], the computer time required is approximately proportional to $10^{0.2P}$, where $P$ is the number of filter coefficients to be optimized in the discrete space. Note that the expression $10^{0.2P}$ gives an estimate on the order of magnitude of the computer time required. Individual runs may differ by over an order of magnitude from what is predicted by $10^{0.2P}$.

When using MILP, the error magnitude $|E_m(e^{jM\omega})|$ is constrained to be not more than $w(\omega)\delta$, where $w(\omega)$ is a weighting function and $\delta$ is a variable to be minimized. Unfortunately, $E_m(e^{jM\omega})$ is complex, and $|E_m(e^{jM\omega})| = \sqrt{\mathrm{Re}\{E_m(e^{jM\omega})\} + \mathrm{Im}\{E_m(e^{jM\omega})\}}$ is not a linear function of $\boldsymbol{h_m}$ where $\mathrm{Re}\{x\}$ and $\mathrm{Im}\{x\}$ denote the real and imaginary parts of $x$, respectively. The optimum cannot be found using linear programming by directly minimizing $\sqrt{\mathrm{Re}\{E_m(e^{jM\omega})\} + \mathrm{Im}\{E_m(e^{jM\omega})\}}$. However, the difficulty can be overcome as follows. By multiplying both sides of (16) by $e^{j\theta}$, we have

$$
\begin{aligned}
&e^{j\theta} E_m(e^{jM\omega}) \\
&= V(\omega)\left\{ e^{j\theta}[\boldsymbol{\xi}_m(jM\omega)]^{\mathrm{T}} \boldsymbol{h}_m - e^{j\theta} e^{j(m-M\eta)\omega} \right\}. \quad (22)
\end{aligned}
$$

Our aim is to constrain the maximum value of $|E_m(e^{jM\omega})|$ such that $|E_m(e^{jM\omega})| \leq w(\omega)\delta$. Note that $|E_m(e^{jM\omega})| = |e^{j\theta} E_m(e^{jM\omega})| = |\mathrm{Re}\{e^{j\theta} E_m(e^{jM\omega})\}|$ if $\theta$ is equal to the negative phase angle of $E_m(e^{jM\omega})$. We do not know the phase angle of $E_m(e^{jM\omega})$ but we can optimize $E_m(e^{jM\omega})$ over a dense grid of possible phase angles. Thus, the constraint $|E_m(e^{jM\omega})| \leq w(\omega)\delta$ can be imposed by imposing the set of constraints $|\mathrm{Re}\{e^{j\theta} E_m(e^{jM\omega})\}| \leq w(\omega)\delta$ on a dense grid of $\theta$ where $0 \leq \theta \leq \pi$. Thus, we have the MILP problem

$$
\begin{aligned}
&\mathrm{Re}\{e^{j\theta}[\boldsymbol{\xi}_m(j\omega)]^{\mathrm{T}} \boldsymbol{h}_m V(\omega)\} - w(\omega)\delta \\
&\leq \mathrm{Re}\{e^{j\theta} e^{j(m-M\eta)\omega} V(\omega)\} \\
&\hspace{5cm} \text{over all } \omega \text{ and } \theta \quad (23a) \\
&\mathrm{Re}\{e^{j\theta}[\boldsymbol{\xi}_m(j\omega)]^{\mathrm{T}} \boldsymbol{h_m} V(\omega)\} + w(\omega)\delta \\
&\geq \mathrm{Re}\left\{ e^{j\theta} e^{j(m-M\eta)\omega} V(\omega) \right\} \\
&\hspace{5cm} \text{over all } \omega \text{ and } \theta \quad (23b)
\end{aligned}
$$

$$
\text{Objective:} \quad \text{minimize } \delta \quad (23c)
$$

$$
\begin{aligned}
\text{Subject to:} \quad &\text{elements of } \boldsymbol{h}_m \text{ are sum of a specified} \\
&\text{number of SPT terms.} \quad (23d)
\end{aligned}
$$

TABLE IX
SUB-ADC MISMATCH PARAMETERS FOR *EXAMPLE-IV*

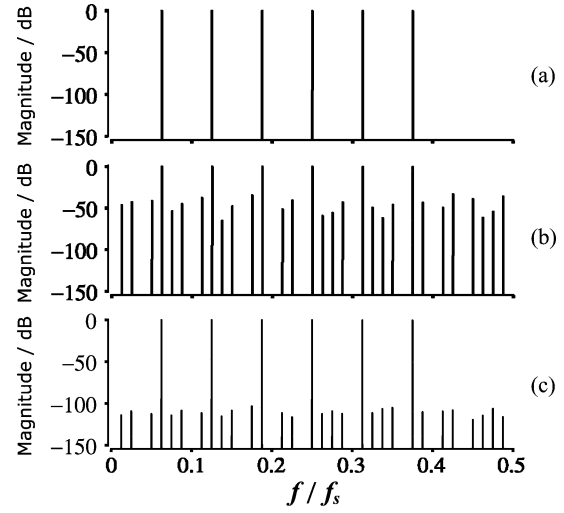| | | |
|---|---|---|
| $H_{\mathrm{A},0}(\omega) = \dfrac{1}{1 + j\frac{\omega}{2\pi f_s}}$ | $H_{\mathrm{A},1}(\omega) = \dfrac{e^{j0.03\omega/f_s}}{1 + j\frac{1.2\omega}{2\pi f_s}}$ | $H_{\mathrm{A},2}(\omega) = \dfrac{e^{j0.05\omega/f_s}}{1 + j\frac{1.3\omega}{2\pi f_s}}$ |
| $H_{\mathrm{A},3}(\omega) = \dfrac{e^{-j0.04\omega/f_s}}{1 + j\frac{0.8\omega}{2\pi f_s}}$ | $H_{\mathrm{A},4}(\omega) = \dfrac{e^{-j0.05\omega/f_s}}{1 + j\frac{0.7\omega}{2\pi f_s}}$ | |



Fig. 13. (a) Input. (b) Uncompensated. (c) Compensated.

*Example-IV:* We shall illustrate a multiplierless multichannel TIADC compensation filter using the mismatch parameters as shown in Table IX. The multichannel filters are optimized using MILP over the frequency ranging from 0 to $0.375f_s$ with $\delta$ constrained to be less than 0.00001. Our objective is to minimize the total number of SPT terms. In the optimization, each coefficient is given a fixed number of SPT terms. The number of coefficients for the multichannel filters are $N_0 = 1$, $N_1 = N_4 = 20$, and $N_2 = N_3 = 21$. The coefficient values of our multichannel filters are shown in Table X. The last row in Table X shows the computer time required on an HPxw4400 PC. The spectrum plots are shown in Fig. 13. Fig. 13(a) shows the input spectral lines at $0.625mf_s$, $m = 1, 2, \ldots, 6$. Fig. 13(b) and (c) shows the spectral lines of the uncompensated and compensated TIADC, respectively.

On the average, the computer time required to optimize each of the multichannel filters is $(42 + 412 + 80 + 122)/4 \text{ s} \approx 3 \text{ min}$. If the filter-bank approach is used, the number of variables to be optimized in the discrete space would be about 100; using the estimation formula that the computer time required is proportional to $10^{0.2P}$, the computer time required would be approximately $3 \times 10^{16}$ min $\approx 60$ billion years.

## VIII. 14-BIT 1.6-GS/s TIADC

The spectral plots shown in the previous examples were simulated results where the sub-ADCs were assumed to be perfect. In an actual TIADC system, the sub-ADCs are not perfect, and the TIADC spectral plot contains spurs other than those due to sub-ADC mismatch.

TABLE X
COEFFICIENT VALUES FOR *EXAMPLE-IV*

| | $k=0$ | $k=1$ | $k=2$ | $k=3$ | $k=4$ |
|---|---|---|---|---|---|
| $h_{k,[k+0]}(0)$ | 0 | $2^{-20}$ | $2^{-17}$ | $-2^{-17}+2^{-20}$ | $-2^{-16}-2^{-18}$ |
| $h_{k,[k+4]}(1-I(k+4))$ | 0 | $-2^{-17}$ | $-2^{-15}-2^{-17}$ | $2^{-15}+2^{-18}$ | $2^{-14}+2^{-16}$ |
| $h_{k,[k+3]}(1-I(k+3))$ | 0 | $2^{-15}+2^{-18}$ | $2^{-13}-2^{-16}+2^{-20}$ | $-2^{-13}+2^{-15}-2^{-18}$ | $-2^{-12}+2^{-14}-2^{-17}$ |
| $h_{k,[k+2]}(1-I(k+2))$ | 0 | $-2^{-13}+2^{-16}$ | $-2^{-12}-2^{-16}+2^{-20}$ | $2^{-12}-2^{-16}-2^{-19}$ | $2^{-11}-2^{-13}+2^{-15}$ |
| $h_{k,[k+1]}(1-I(k+1))$ | 0 | $2^{-12}$ | $2^{-11}+2^{-14}+2^{-15}$ | $-2^{-11}+2^{-18}$ | $-2^{-11}-2^{-12}+2^{-19}$ |
| $h_{k,[k+0]}(1)$ | 0 | $-2^{-11}+2^{-18}$ | $-2^{-10}-2^{-13}-2^{-15}$ | $2^{-10}+2^{-16}+2^{-18}$ | $2^{-10}+2^{-12}+2^{-14}$ |
| $h_{k,[k+4]}(2-I(k+4))$ | 0 | $2^{-10}-2^{-14}-2^{-16}$ | $2^{-9}+2^{-14}+2^{-16}$ | $-2^{-9}+2^{-14}+2^{-16}$ | $-2^{-9}-2^{-12}-2^{-14}$ |
| $h_{k,[k+3]}(2-I(k+3))$ | 0 | $-2^{-9}+2^{-12}+2^{-18}$ | $-2^{-8}+2^{-12}+2^{-14}$ | $2^{-8}-2^{-11}-2^{-14}$ | $2^{-8}-2^{-14}+2^{-17}$ |
| $h_{k,[k+2]}(2-I(k+2))$ | 0 | $2^{-8}-2^{-12}+2^{-19}$ | $2^{-7}-2^{-10}+2^{-12}$ | $-2^{-7}+2^{-10}+2^{-11}-2^{-15}$ | $-2^{-7}+2^{-10}-2^{-17}-2^{-19}$ |
| $h_{k,[k+1]}(2-I(k+1))$ | 0 | $-2^{-7}-2^{-9}-2^{-11}$ | $-2^{-6}-2^{-8}-2^{-11}-2^{-13}$ | $2^{-6}+2^{-10}+2^{-14}+2^{-17}$ | $2^{-6}+2^{-10}+2^{-12}-2^{-15}$ |
| $h_{k,[k+0]}(2)$ | 1 | $2^{0}+2^{-6}-2^{-13}+2^{-19}$ | $2^{0}+2^{-5}-2^{-7}+2^{-11}-2^{-17}$ | $2^{-0}-2^{-6}+2^{-8}-2^{-11}$ | $2^{0}-2^{-6}-2^{-8}+2^{-13}-2^{-16}$ |
| $h_{k,[k+4]}(3-I(k+4))$ | 0 | $-2^{-7}+2^{-12}+2^{-18}$ | $-2^{-7}+2^{-10}+2^{-12}-2^{-16}$ | $-2^{-8}+2^{-10}-2^{-12}+2^{-17}$ | $2^{-7}-2^{-10}-2^{-12}-2^{-15}$ |
| $h_{k,[k+3]}(3-I(k+3))$ | 0 | $-2^{-11}-2^{-13}+2^{-16}$ | $-2^{-8}+2^{-11}+2^{-16}$ | $2^{-8}+2^{-10}-2^{-19}$ | $2^{-10}+2^{-15}+2^{-17}$ |
| $h_{k,[k+2]}(3-I(k+2))$ | 0 | $2^{-11}+2^{-13}+2^{-15}$ | $2^{-9}+2^{-11}-2^{-15}$ | $-2^{-8}+2^{-11}+2^{-12}$ | $-2^{-10}+2^{-13}-2^{-16}$ |
| $h_{k,[k+1]}(3-I(k+1))$ | 0 | $-2^{-11}+2^{-15}+2^{-18}$ | $-2^{-9}+2^{-12}+2^{-13}$ | $2^{-9}+2^{-16}+2^{-19}$ | $2^{-11}+2^{-17}+2^{-18}$ |
| $h_{k,[k+0]}(3)$ | 0 | $2^{-12}+2^{-16}$ | $2^{-10}+2^{-15}-2^{-18}$ | $-2^{-10}-2^{-12}$ | $-2^{-12}+2^{-18}$ |
| $h_{k,[k+4]}(4-I(k+4))$ | 0 | $-2^{-13}-2^{-18}$ | $-2^{-11}-2^{-14}-2^{-16}$ | $2^{-11}+2^{-12}-2^{-19}$ | $2^{-13}-2^{-15}+2^{-17}$ |
| $h_{k,[k+3]}(4-I(k+3))$ | 0 | $2^{-14}-2^{-17}$ | $2^{-12}+2^{-15}-2^{-19}$ | $-2^{-11}+2^{-13}-2^{-16}$ | $-2^{-15}-2^{-19}$ |
| $h_{k,[k+2]}(4-I(k+2))$ | 0 | $-2^{-16}$ | $-2^{-13}+2^{-17}$ | $2^{-13}+2^{-15}+2^{-16}$ | $2^{-19}$ |
| $h_{k,[k+1]}(4-I(k+1))$ | 0 | $2^{-19}$ | $2^{-15}+2^{-17}$ | $-2^{-14}+2^{-19}$ | $2^{-19}$ |
| $h_{k,[k+0]}(4)$ | 0 | 0 | $-2^{-17}$ | $2^{-16}$ | 0 |
| **HPxw4400 time** | **0** | **42 seconds** | **412 seconds** | **80 seconds** | **122 seconds** |
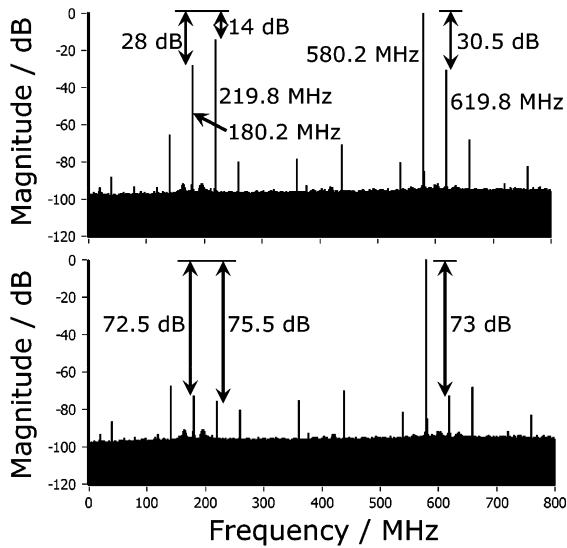


Fig. 14. Spectral plot for the output of a 14-bit 1.6 GS/s TIADC. The input is a 580.2-MHz sinusoid. The top plot shows the uncompensated output spectrum. The bottom plot shows the compensated output spectrum. The filter length was 7.

*Example-V:* Our mismatch compensation technique was implemented on a system that employs four 14-bit 400-MS/s sub-ADCs to construct a 14-bit 1.6-GS/s TIADC. The mismatch between sub-ADCs was characterized by injecting sinusoids spaced 1 MHz apart one at a time. The phase and amplitude of

the fundamental frequency of the sinusoid at the output of the sub-ADCs were used to determine the phase and amplitude of the mismatch.

Fig. 14 shows a spectral plot of the test result when the input frequency was 580.2 MHz. The spurious frequencies due to sub-ADC mismatch were 180.2, 219.8, and 619.8 MHz, respectively. The magnitude of the input sinusoid was about 82% of full scale and was normalized to 0 dB. Each spectral line in each of the plots was the root-mean-square (rms) value of 50 test runs; 8000 data samples were collected for computing the spectral lines in each test run. Since the ADC had 14 bits, if the TIADC was perfect, the expected value of each spectral line (other than those corresponding to the input frequencies) would be $-10\log_{10}(0.5 \times (0.82 \times 2^{13})^2 \times 12 \times 8000) = -123.4$ dB. It can be seen from Fig. 14 that the noise floor was significantly higher than $-123.4$ dB. Actual measurements showed that the noise floor was $-98.2$ dB; this corresponds to a degradation of 25.2 dB (or 4.2 bits). Thus, the effective precision of the ADC was about 9.8 bits (instead of 14 bits) at 580.2 MHz.

The top plot of Fig. 14 shows the spectral plot for the output of the uncompensated TIADC. The magnitudes of the spurious lines due to mismatch were $-28$, $-14$, and $-30.5$ dB, respectively. The other spurs were due to the nonlinearity of the sub-ADCs. The bottom plot of Fig. 14 shows the spectral lines for the output of the WLS-compensated TIADC where the length of each of the compensation filters was 7. It can be seen that the spurs due to sub-ADC mismatch were significantly reduced and the magnitudes were less than those of the spurs due to the non-linearity of the sub-ADC. Thus, for this particular case, unless
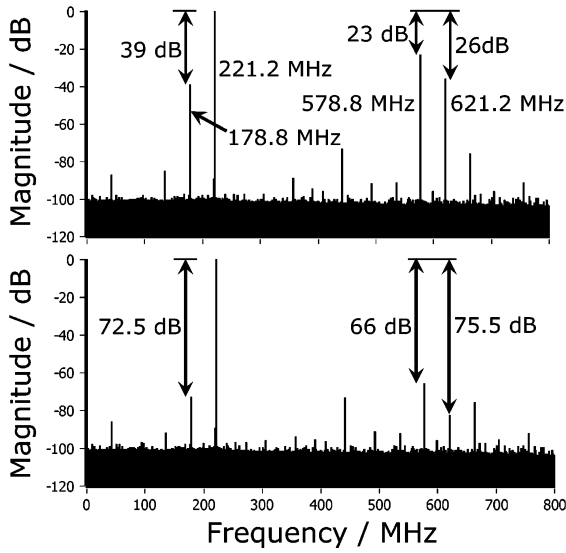
Fig. 15. Similar plots to that in Fig. 14 but the 580.2-MHz sinusoid was replaced by a 221.2-MHz sinusoid.



Fig. 16. Similar plots to that in Fig. 14 but the 580.2-MHz sinusoid was replaced by a 69.8-MHz sinusoid.

the linearity of the individual sub-ADC was improved, there was little advantage in using compensation filters longer than 7.

Figs. 15 and 16 show the results when the 580.2-MHz sinusoid of Fig. 14 was replaced by 221.2- and 69.8-MHz sinusoids, respectively. In Fig. 15, the magnitude of the 221.2-MHz sinusoid was 85% of the full scale, and after normalizing the magnitude of the 221.2-MHz signal to 0 dB, the noise floor was $-105.2$ dB. Thus, the effective word length of the ADC at 221.2 MHz was

$$ 1 + \frac{\log_{10}\left(\left(\sqrt{\frac{10^{105.2/10}}{0.5 \times 12 \times 8000}}\right) \middle/ 0.85\right)}{\log_{10}2} \text{bits} = 10.9 \text{ bits}. $$

In Fig. 16, the magnitude of the 69.8-MHz sinusoid was 89% of the full scale, and after normalizing the magnitude of the 69.8-MHz signal to 0 dB, the noise floor was $-107.5$ dB. Thus, the effective word length of the ADC at 69.8 MHz was

$$ 1 + \frac{\log_{10}\left(\left(\sqrt{\frac{10^{107.5/10}}{0.5 \times 12 \times 8000}}\right) \middle/ 0.89\right)}{\log_{10}2} \text{bits} = 11.2 \text{ bits}. $$

## IX. CONCLUSION

In this paper, we have presented a new multichannel approach for TIADC mismatch compensation. Comparing with the filter-bank approach, for the same number of coefficients $NM$, the two approaches produce filters with the same performance if the filters were designed with sufficient arithmetic precision.

Our new multichannel approach has the following advantages over the filter-bank approach. First, for the same total number of coefficients $NM$, each dimension of the matrices in our approach is a factor of $M$ smaller than in the filter-bank approach. Second, in our approach, the matrix to be inverted has a dimension of $N \times N$, whereas the matrix to be inverted in the filter-bank case has a dimension of $NM \times NM$. This means
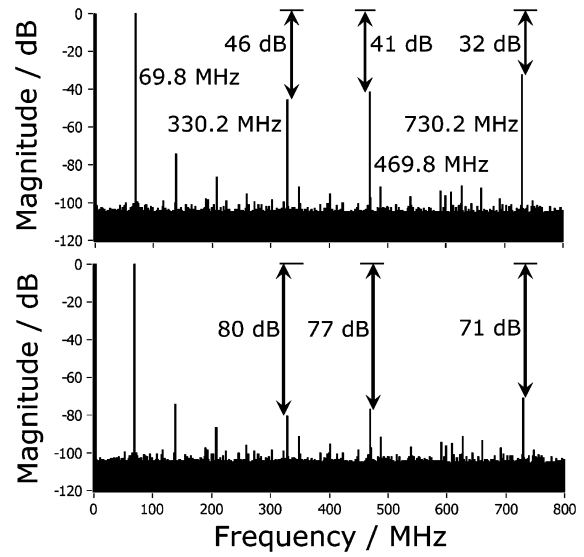
that the computational complexity of the filter design in our approach is lower. Third, as a result of the smaller matrices, the filter design in our approach is computationally more robust than the filter design in the filter-bank approach, as illustrated in *Example-I*. Lastly, in our approach, each compensation filter may have a different number of coefficients; a less seriously mismatched sub-ADC may be compensated by using a filter with less number of coefficients.

Our multichannel approach and the filter-bank approach are developed based on different principles. In our approach, we make use of the fact that a signal may be represented by a weighted combination of any irregularly spaced samples provided that the highest frequency component of the signal is less than 0.5 times the average sampling speed (the generalized sampling theorem). We use a multichannel filter to estimate the ideal output of each sub-ADC. In the filter-bank approach, the frequency spectrum of each sub-ADC output is shaped by a filter in such a way that, when all the filter outputs are summed, the aliasing terms cancel each other, and the baseband term becomes the best approximation to the input analog signal.

The response of a hardware 1.6-GS/s TIADC implemented using four 400-MS/s sub-ADCs has also been presented. It can be seen from the spurious frequency components before and after mismatch compensation that mismatch is not the only source of spurious frequency components. Spurious frequency components that are not caused by mismatch cannot be removed using the mismatch compensation technique.

## X. FUTURE WORK

In some systems, the TIADC operates in a tightly controlled environment, and mismatch parameter drift is not a problem. In some systems, the luxury of a tightly controlled environment is not available. In such cases, the TIADC mismatch parameters may drift with environmental variables such as temperature, humidity, proximity to ferromagnetic materials, etc., depending on implementation. The design of a TIADC compensation filter subject to environmental variable drift is part of our work that will be reported in the future.

REFERENCES

[1] W. C. Black and D. A. Hodges, "Time interleaved converter arrays," *IEEE J. Solid State Circuits*, vol. SSC-15, no. 6, pp. 1022–1029, Dec. 1980.

[2] S. Limotyrakis, S. D. Kulchycki, D. K. Su, and B. A. Wooley, "A 150-MS/s 8-b 71-mW CMOS time-interleaved ADC," *IEEE J. Solid State Circuits*, vol. 40, no. 5, pp. 1057–1067, May 2005.

[3] Z. M. Lee, C. Y. Wang, and J. T. Wu, "A CMOS 15-bit 125-MS/s time-interleaved ADC with digital background calibration," *IEEE J. Solid State Circuits*, vol. 42, no. 10, pp. 2149–2160, Oct. 2007.

[4] J. Elbornsson, F. Gustafsson, and J. E. Eklund, "Analysis of mismatch effects in a randomly interleaved A/D converter system," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 52, no. 3, pp. 465–476, Mar. 2005.

[5] G. Leger, E. J. Peralias, A. Rueda, and J. L. Huertas, "Impact of random channel mismatch on the SNR and SFDR of time-interleaved ADCs," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 51, no. 1, pp. 140–150, Jan. 2004.

[6] C. Vogel and H. Johansson, "Time-interleaved analog-to-digital converters: Status and future directions," in *Proc. Int. Symp. Circuits Syst.*, May 2006, pp. 3386–3389.

[7] K. Dyer, D. H. Fu, P. Hurst, and S. Lewis, "A comparison of monolithic background calibration in two time-interleaved analog-to-digital converters," in *Proc. Int. Symp. Circuits Syst.*, May/Jun. 1998, vol. 1, pp. 13–16.

[8] J. Elbornsson, F. Gustafsson, and J. E. Eklund, "Amplitude and gain error influence on time error estimation algorithm for time interleaved A/D converter system," in *Proc. Int. Conf. Acoust., Speech, Signal Process.*, May 2002, vol. 2, pp. 1281–1284.

[9] M. Seo, M. J. W. Rodwell, and U. Madhow, "Blind correction of gain and timing mismatches for a two-channel time-interleaved analog-to-digital converter," in *Proc. 39th Asilomar Conf. Signals, Syst., Comput.*, Oct. 2005, pp. 1121–1124.

[10] A. Cabrini, F. Maloberti, R. Rovatti, and G. Setti, "On-line calibration of offset and gain mismatch in time-interleaved ADC using a sampled-data chaotic bit-stream," in *Proc. Int. Symp. Circuits Syst.*, May 2006, pp. 3398–3401.

[11] A. Haftbaradaran and K. W. Martin, "Mismatch compensation techniques using random data for time-interleaved A/D converters," in *Proc. Int. Symp. Circuits Syst.*, May 2006, pp. 3402–3405.

[12] V. Hakkarainen, L. Sumanen, M. Aho, M. Waltari, and K. Halonen, "A self-calibration technique for time-interleaved pipelined ADCs," in *Proc. Int. Symp. Circuits Syst.*, May 2003, vol. I, pp. 825–828.

[13] E. El-Sankary and M. Sawam, "A background calibration technique for multibit/stage pipelined and time-interleaved ADCs," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 53, no. 6, pp. 448–452, Jun. 2006.

[14] S. M. Jamal, D. Fu, M. P. Singh, P. J. Hurst, and S. H. Lewis, "Calibration of sample-time error in a two-channel time-interleaved analog-to-digital converter," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 51, no. 1, pp. 130–139, Jan. 2004.

[15] T. H. Tsai, P. J. Hurst, and S. H. Lewis, "Bandwidth mismatch and its correction in time-interleaved analog-to-digital converters," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 53, no. 10, pp. 1133–1137, Oct. 2006.

[16] K. Asami, "Technique to improve the performance of time-interleaved A–D converters," in *Proc. Int. Test Conf.*, 2005, Paper 34.1.

[17] S. Mendel and C. Vogel, "A compensation method for magnitude response mismatches in two-channel time-interleaved analog-to-digital converters," in *Proc. 13th IEEE Int. Conf. Electron., Circuits Syst.*, Nice, France, Dec. 2006, pp. 712–715.

[18] M. Seo, M. J. W. Rodwell, and U. Madhow, "Comprehensive digital correction of mismatch errors for a 400-Msamples/s 80-dB SFDR time-interleaved analog-to-digital converter," *IEEE Trans. Microw. Theory Tech.*, vol. 53, no. 3, pp. 1072–1082, Mar. 2005.

[19] P. Satarzadeh, B. C. Levy, and P. J. Hurst, "Bandwidth mismatch correction for a two-channel time-interleaved A/D converter," in *Proc. IEEE Int. Symp. Circuits Syst.*, May 2007, pp. 1705–1708.

[20] S. Mendel and C. Vogel, "On the compensation of magnitude response mismatches in $M$-channel time-interleaved ADCs," in *Proc. IEEE Int. Symp. Circuits Syst.*, May 2007, pp. 3375–3378.

[21] H. Johansson, P. Löwenborg, and K. Vengattaramane, "Least-squares and minimax design of polynomial impulse response FIR filters for reconstruction of two-periodic nonuniformly sampled signals," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 54, no. 4, pp. 877–888, Apr. 2007.

[22] H. Johansson and P. Löwenborg, "Reconstruction of nonuniformly sampled bandlimited signals by means of digital fractional delay filters," *IEEE Trans. Signal Process.*, vol. 50, no. 11, pp. 2757–2767, Nov. 2002.

[23] R. S. Prendergast, B. C. Levy, and P. J. Hurst, "Reconstruction of band-limited periodic nonuniformly sampled signals through multirate filter banks," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 51, no. 8, pp. 1612–1622, Aug. 2004.

[24] H. Johansson and P. Löwenborg, "Reconstruction of nonuniformly sampled bandlimited signals by means of time-varying discrete-time FIR filters," *EURASIP J. Appl. Signal Process.*, vol. 2006, pp. 1–18, 2006, Article ID 64185.

[25] S. Tertinek and C. Vogel, "Reconstruction of two-periodic nonuniformly sampled band-limited signals using a discrete-time differentiator and a time-varying multiplier," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 54, no. 7, pp. 616–620, Jul. 2007.

[26] Y. C. Jenq, "Perfect reconstruction of digital spectrum from nonuniformly sampled signals," *IEEE Trans. Instrum. Meas.*, vol. 46, no. 3, pp. 649–652, Jun. 1997.

[27] H. Jin and E. K. F. Lee, "A digital-background calibration technique for minimizing timing-error effects in time-interleaved ADC's," *IEEE Trans. Circuits Syst. II, Analog Digit. Signal Process.*, vol. 47, no. 7, pp. 603–613, Jul. 2000.

[28] C. Y. Wang and J. T. Wu, "A background timing-skew calibration technique for time-interleaved analog-to-digital converters," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 53, no. 4, pp. 299–303, Apr. 2006.

[29] A. Petraglia and S. K. Mitra, "High speed A/D conversion using QMF banks," in *Proc. IEEE Int. Symp. Circuits Syst.*, 1990, pp. 2797–2800.

[30] H. Shu, T. Chen, and B. A. Francis, "Minimax design of hybrid multirate filter banks," *IEEE Trans. Circuits Syst. II, Analog Digit. Signal Process.*, vol. 44, no. 2, pp. 120–128, Feb. 1997.

[31] S. R. Velazquez, T. Q. Nguyen, and S. R. Broadstone, "Design of hybrid filter banks for analog/digital conversion," *IEEE Trans. Signal Process.*, vol. 46, no. 4, pp. 956–967, Apr. 1998.

[32] V. Divi and G. Wornell, "Scalable blind calibration of timing skew in high-resolution time-interleaved ADCs," in *Proc. Int. Symp. Circuits Syst.*, May 2006, pp. 3390–3393.

[33] M. Seo, M. J. W. Rodwell, and G. Madhow, "Blind correction of gain and timing mismatches for a two-channel time-interleaved analog-to-digital converter: Experimental verification," in *Proc. Int. Symp. Circuits Syst.*, May 2006, pp. 3394–3397.

[34] J. Elbornsson, F. Gustafsson, and J. E. Eklund, "Blind equalization of time errors in a time-interleaved ADC system," *IEEE Trans. Signal Process.*, vol. 53, no. 4, pp. 1413–1424, Apr. 2005.

[35] S. Huang and B. C. Levy, "Blind calibration of timing offsets for four-channel time-interleaved ADCs," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 54, no. 4, pp. 863–876, Apr. 2007.

[36] S. Huang and B. C. Levy, "Adaptive blind calibration of timing offset and gain mismatch for two-channel time-interleaved ADCs," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 53, no. 6, pp. 1278–1288, Jun. 2006.

[37] F. J. Beutler, "Error-free recovery of signals from irregularly spaced samples," *SIAM Rev.*, vol. 8, no. 3, pp. 328–335, Jul. 1966.

[38] H. Nyquist, "Certain topics in telegraph transmission theory," *Trans. A.I.E.E.*, vol. 47, pp. 617–644, Feb. 1928.

[39] C. E. Shannon, "A mathematical theory of communication," *Bell Syst. Tech. J.*, vol. 27, pp. 379–423, 623–656, Jul., Oct. 1948.

[40] C. E. Shannon, "Communication in the presence of noise," *Proc. IRE*, vol. 37, no. 1, pp. 10–21, Jan. 1949.

[41] E. A. Robinson, *Multichannel Time Series Analysis With Digital Computer Programs*. San Francisco, CA: Holden-Day, Inc., 1967.

[42] T. C. Hsia, *System Identification: Least-Squares Methods*. Lexington, MA: Lexington Books, 1997.

[43] A. S. Bhushan, P. V. Kelkar, B. Jalali, O. Boyraz, and M. Islam, "130-GSa/s photonic analog-to-digital converter with time stretch preprocessor," *IEEE Photon. Technol. Lett.*, vol. 14, no. 5, pp. 684–686, May 2002.

[44] Y. C. Lim and A. G. Constantinides, "Linear phase FIR digital filter without multipliers," in *Proc. IEEE Int. Symp. Circuits Syst.*, 1979, pp. 185–188.

[45] Y. C. Lim and A. G. Constantinides, "New integer programming scheme for nonrecursive digital filter design," *Electron. Lett.*, vol. 15, no. 25, pp. 812–813, Dec. 6, 1979.

[46] Y. C. Lim, S. R. Parker, and A. G. Constantinides, "Finite wordlength FIR filter design using integer programming over a discrete coefficient space," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-30, no. 4, pp. 661–664, Aug. 1982.

[47] Y. C. Lim and S. R. Parker, "FIR filter design over a discrete powers-of-two coefficient space," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-31, no. 3, pp. 583–591, Jun. 1983.

[48] Y. C. Lim and S. R. Parker, "Discrete coefficient FIR digital filter design based upon an LMS criteria," *IEEE Trans. Circuits Syst.*, vol. CAS-30, no. 10, pp. 723–739, Oct. 1983.

[49] Y. C. Lim, "Design of discrete-coefficient-value linear phase FIR filters with optimum normalized peak ripple magnitude," *IEEE Trans. Circuits Syst.*, vol. 37, no. 12, pp. 1480–1486, Dec. 1990.

[50] Y. C. Lim, R. Yang, D. Li, and J. Song, "Signed power-of-two term allocation scheme for the design of digital filters," *IEEE Trans. Circuits Syst. II, Analog Digit. Signal Process.*, vol. 46, no. 5, pp. 577–584, May 1999.

[51] A. Land and S. Powell, *Fortran Codes for Mathematical Programming*. Hoboken, NJ: Wiley, 1973.

**Yong Ching Lim** (S'79–M'82–SM'92–F'00) received the A.C.G.I. and B.Sc. degrees in electrical engineering in 1977 and the D.I.C. and Ph.D. degrees in electrical engineering in 1980 from Imperial College, University of London, London, U.K.

From 1982 to 2003, he was with the Department of Electrical Engineering, National University of Singapore, Singapore, Singapore. From 1980 to 1982, he was a National Research Council Research Associate with the Naval Postgraduate School, Monterey, CA. Since 2003, he has been with the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore, where he is currently a Professor. He has also served as an Associate Editor for *Circuits, Systems and Signal Processing* from 1993 to 2000. His research interests include digital signal processing and VLSI circuits and system design.

Dr. Lim was a recipient of the 1996 IEEE Circuits and Systems Society's Guillemin–Cauer Award, the 1990 IREE (Australia) Norman Hayes Memorial Award, 1977 IEE (U.K.) Prize, and the 1974–1977 Siemens Memorial (Imperial College) Award. He served as a Lecturer for the IEEE Circuits and Systems Society under the distinguished lecturer program from 2001 to 2002 and as an Associate Editor for the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS from 1991 to 1993 and from 1999 to 2001. He served as the Chairman of the DSP Technical Committee of the IEEE Circuits and Systems Society from 1998 to 2000. He served in the Technical Program Committee's DSP Track as the Chairman in the IEEE International Symposium on Circuits and Systems (ISCAS) 1997 and 2000 and as a Cochairman in the IEEE ISCAS 1999. He is the General Chairman for the IEEE Asia Pacific Conference on Circuits and Systems 2006 and a General Cochair for IEEE ISCAS 2009. He is a member of Eta Kappa Nu.

**Yue-Xian Zou** received the Ph.D. degree from The University of Hong Kong, Hong Kong, Hong Kong, in 2000.

She is an Associate Professor with Peking University, Beijing, China, and is the Director of the Advanced Digital Signal Processing Laboratory, Shenzhen Graduate School, Peking University, Shenzhen, China. She is currently working on a high-speed and high-resolution analog-to-digital converter project. Her research work includes digital filter design, adaptive signal processing, array signal processing, and video signal processing.

**Jun Wei Lee** (S'02–M'06) was born in Singapore on October 26, 1976. He received the B.Eng. degree in electrical engineering from the National University of Singapore, Singapore, Singapore, in 2001 and the Ph.D. degree in electrical engineering from both the National University of Singapore and the Technische Universiteit Eindhoven, Eindhoven, The Netherlands, in 2007.

Since 2005, he has been a Researcher with Temasek Laboratories, Nanyang Technological University, Singapore. His research interests include time-interleaved analog-to-digital conversion, design of digital filters and filter banks, and audio compression.

**Shing-Chow Chan** (S'87–M'92) received the B.Sc.(Eng.) and Ph.D. degrees in electrical engineering from The University of Hong Kong, Hong Kong, Hong Kong, in 1986 and 1992, respectively.

He joined the City Polytechnic of Hong Kong, Kowloon, Hong Kong, in 1990 as an Assistant Lecturer and, later, as a University Lecturer. Since 1994, he has been with the Department of Electrical and Electronic Engineering, The University of Hong Kong, where he is currently a Professor. He holds visiting positions in Microsoft Corporation, Redmond, U.S., Microsoft Research Asia, University of Texas, Arlington, and Nanyang Technological University, Singapore, Singapore. He has published more than 200 journal and conference articles and is the coauthor of a book entitled *Image-Based Rendering* (Springer, 2006). He is an Associate Editor of the *Journal of Signal Processing Systems*. His research interests include fast transform algorithms, filter design and realization, multirate signal processing, communications signal processing, and image-based rendering.

Dr. Chan is currently a member of the Digital Signal Processing Technical Committee of the IEEE Circuits and Systems Society and an Associate Editor for the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS—I: REGULAR PAPERS. He was the Chairman of the IEEE Hong Kong Chapter of Signal Processing from 2000 to 2002. He is the recipient of the Myril B. Reed Best Paper Award at the Midwest Symposium on Circuits and Systems 2004 and the Best Student Paper Award with Y. Zhou at the IEEE 2005 Workshop on Statistical Signal Processing. He was in the organizing committees of several international conferences including the special session as a Cochair of the IEEE International Conference on Acoustics, Speech, and Signal Processing in 2003, the Technical Program Cochair of IEEE International Conference on Field-Programmable Technology in 2002, and the Technical Program Committee of the IEEE Asia Pacific Conference on Circuits and Systems (APCCAS) 2006, IEEE APCCAS 2008, and IEEE Biomedical Circuits and Systems Conference 2006. He is also in the organizational committee of the IEEE International Conference on Image Processing in 2010.