Polyphase Nonlinear Equalization of Time-Interleaved Analog-to-Digital Converters

Citation: Goodman, J. et al. "Polyphase Nonlinear Equalization of Time-Interleaved Analog-to-Digital Converters." Selected Topics in Signal Processing, IEEE Journal of 3.3 (2009): 362-373. © 2009 Institute of Electrical and Electronics Engineers

As Published: http://dx.doi.org/10.1109/JSTSP.2009.2020243

Publisher: Institute of Electrical and Electronics Engineers

Persistent URL: http://hdl.handle.net/1721.1/52371

Version: Final published version: final published article, as it appeared in a journal, conference proceedings, or other formally published context

Polyphase Nonlinear Equalization of Time-Interleaved Analog-to-Digital Converters

Joel Goodman, *Member, IEEE*, Benjamin Miller, Matthew Herman, Gil Raz, *Member, IEEE*, and Jeffrey Jackson, *Member, IEEE*

***Abstract*—As the demand for higher data rates increases, commercial analog-to-digital converters (ADCs) are more commonly being implemented with multiple on-chip converters whose outputs are time-interleaved. The distortion generated by time-interleaved ADCs is now not only a function of the nonlinear behavior of the constituent circuitry, but also mismatches associated with interleaving multiple output streams. To mitigate distortion generated by time-interleaved ADCs, we have developed a polyphase NonLinear EQualizer (pNLEQ) which is capable of simultaneously mitigating distortion generated by both the on-chip circuitry and mismatches due to time interleaving. In this paper, we describe the pNLEQ architecture and present measurements of its performance.**

***Index Terms*—Compressed sensing, mismatch distortions, multidimensional filter, nonlinear compensation, nonlinear equalization, polynomial filter, time-interleaved analog-to-digital converter (ADC), Volterra.**

I. Introduction

DEMAND for high sampling rate ADCs is driven by the desire to process information from wideband signal sources, where much of the signal information processing has migrated from the analog front-end to the digital backend of the receiver [1]–[4]. Systems currently being developed to operate across wide bandwidths with high sensitivity requirements are limited by the inherent dynamic range of the receiver's analog and mixed-signal components. Among these components (e.g., LNA, mixer), the ADC commonly has the lowest dynamic range [1]. An ADC's deviation from its ideal "linear" performance is commonly characterized by its spurious- and/or intermodulation-free dynamic range (SFDR and IFDR), which is a frequency-domain measurement that determines the minimum signal level that can be distinguished from distortion components [5]. The SFDR and IFDR of an ADC are typically dominated by circuit-based (e.g., buffer amplifier, sample-and-hold) nonlinearities that are distinct from the nonlinear process of ideal quantization, which in principle can be circumvented with processing gain [6].

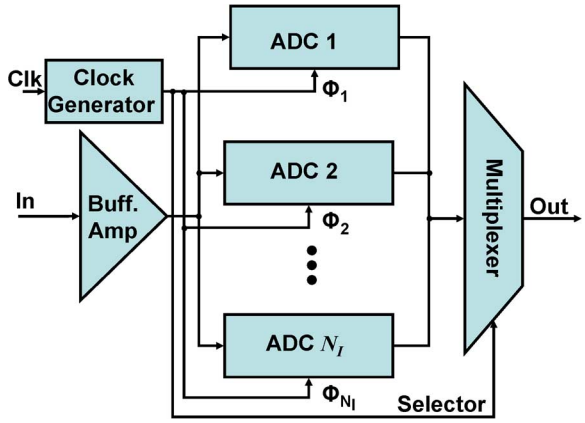

Fig. 1. Simplified block diagram of a time-interleaved analog-to-digital converter. Note that in most implementations the multiplexer is omitted, allowing for a system-level power reduction when the lower-rate data streams are processed off-chip.

Currently, the highest sampling rate ADCs are time interleaved, such that the outputs of multiple on-chip converters are aggregated [7] as depicted in Fig. 1. Time-interleaved ADCs are designed so that the timing phases of the sampling clocks to the constituent converters are *ideally* adjusted so that the digitized samples at the output of the composite ADC are evenly spaced in time. This enables a time-interleaved ADC to achieve an effective sample rate of N_I times the sample rate of any individual converter operating in isolation, where N_I is the number of on-chip converters. However, as we show, small gain and phase mismatches in the linear and nonlinear response of the ADC create unwanted spurs that can in some cases dominate the device's SFDR and IFDR. We refer to these distortions as *mismatch* distortions, which are nonlinear distortions that occur at frequencies that do not correspond to polynomial combinations of the input signal.

A. Previous Approaches

There are descriptions in the literature of how to mitigate distortion generated by gain, phase and offset errors in time-interleaved ADCs [8]–[11]. However, there are relatively few descriptions of realistic compensation for nonlinear *polynomial* distortion. In [12], single-channel static integral and differential memoryless nonlinearities are mitigated; however, this approach is ill-suited for wideband ADCs, where memory effects

Manuscript received June 01, 2008; revised March 05, 2009. Current version published May 15, 2009. This work is sponsored by DARPA under Air Force Contract FA8721-05-C-0002. Opinions, interpretations, conclusions, and recommendations are those of the author and are not necessarily endorsed by the United States Government. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Naofal Al-Dhahir.

J. Goodman, B. Miller, and M. Herman are with MIT Lincoln Laboratory, Lexington, MA 02420 USA (e-mail: jgoodman@ll.mit.edu; bamiller@ll.mit.edu; mherman@ll.mit.edu).

G. Raz and J. Jackson are with GMR Research and Technology, Concord, MA 01720 USA (e-mail: raz@gmrtech.com; jjackson@gmrtech.com).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/JSTSP.2009.2020243

are significant due to variations in the (multidimensional) frequency response of the converter [13]. In [14], the dynamic nature of time-interleaved nonlinearities is modeled, with a proposed compensation technique employing randomization of the channel ADCs, and imposing the constraint that the ADCs be designed with pairwise complimentary nonlinearities. This approach, however, does not address mitigation of distortions generated by existing interleaved ADCs, and imposes a difficult constraint on future chip designs.

In [15], a so-called "phase plane error compensation" approach is taken in which a look-up table following the ADC is used to compensate for errors. The corrected ADC output $x_c \in \mathbb{R}^M$ is formulated as

$$Wx_c = Wx - We(x,s) \tag{1}$$

where $x \in \mathbb{R}^M$ is the vector of ADC output code words, $e(x,s) \in \mathbb{R}^M$ is the vector of output conversion errors, $s \in \mathbb{R}^M$ are the slopes of the input signals, and $W \in \mathbb{C}^{L\times M}$ is the DFT matrix given by

$$W = \begin{pmatrix} 1 & e^{-j2\pi k_1/M} & \dots & e^{-j2\pi(M-1)k_1/M} \\ 1 & e^{-j2\pi k_2/M} & \dots & e^{-j2\pi(M-1)k_2/M} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & e^{-j2\pi k_L/M} & \dots & e^{-j2\pi(M-1)k_L/M} \end{pmatrix} \tag{2}$$

where $k_1,\dots,k_L$ are the L frequency locations where the distortions fall. Formulating the output conversion error as $e(x,s) = B\alpha$, where the columns of $B = [b_1(x,s)\dots b_M(x,s)]$ are the basis functions (i.e., a Gaussian basis), and $\alpha = [\alpha_1 \dots \alpha_M]^T$ are the coefficients, then the least squares solution for the coefficient vector α is given by

$$\alpha = \left((WB)^H(WB)\right)^{-1}(WB)^H x \tag{3}$$

where H denotes the Hermitian operator. Consider the case of an interleaved ADC with two constituent on-chip converters ($N_I = 2$), where the converters are driven by clocks that are 180° out of phase from one another. Then, if the matrix B in (3) is split along its even and odd time samples, we can construct a polyphase matrix expressed as

$$B_{\text{poly}_2} = \begin{pmatrix} (B)_1 & 0 \\ 0 & (B)_2 \\ \vdots & \vdots \\ (B)_{M-1} & 0 \\ 0 & (B)_M \end{pmatrix} \tag{4}$$

where $(B)_i$ corresponds to entries in the ith row of matrix B. The polyphase least squares solution has the same form as (3), with the exception that B is replaced by B_{poly_2}, and α is now twice as long. There are some drawbacks to the approach outlined in (1)–(4). First, even though the slope of the signal is being used in the adaptation, explicit signal state dependence (i.e., a nonlinearity with memory) is not being considered. Second, only a single tone at different frequencies was used during training, i.e., [16] does not address intermodulation distortion.

B. Original Contributions

The uniqueness of pNLEQ is that it *simultaneously* suppresses *not only* frequency-dependent linear mismatch distortions in time-interleaved ADCs, but also polynomial and polynomial mismatch distortions in a computationally efficient fashion. Existing approaches to achieving computationally efficient polynomial filter architectures for RF compensation, principally developed to mitigate distortions generated by power amplifiers in transmitters, limit the multidimensional signal space over which the architecture can suppress spectral regrowth and in-band spurs [17], [18]. In this paper, we develop a technique to construct a polynomial filter architecture that searches over an unrestricted multidimensional signal space to select polynomial components that yield the highest equalization performance for a given computational complexity. In particular, we develop a coordinate system representation for polynomial filters, and leverage compressed sensing techniques to identify a sparse polynomial representation of an inverse nonlinearity. As we demonstrate in Section IV using measured data from commercial time-interleaved ADCs, frequency-dependent polynomial and polynomial mismatch distortions can be on the same order as linear mismatch distortions, which, if not equalized, will limit an ADCs spur- and intermod-free dynamic range.

C. Organization of This Paper

The rest of this paper is organized as follows. In Section II, we formulate the effect of gain and phase mismatches between the linear responses of time-interleaved ADCs, and develop an efficient equalizer to mitigate distortion resulting from these mismatches. In Section III, we introduce the polynomial basis used to mitigate nonlinear distortion, develop a formulation for nonlinear equalization (NLEQ), and show how NLEQ can be augmented to address the polynomial mismatch distortion that arises in interleaved ADCs with a computationally efficient polyphase version of NLEQ (pNLEQ). In Section IV, we demonstrate the performance of pNLEQ on both Maxim and National Semiconductor interleaved ADCs and compare its performance to other interleaved ADC compensation approaches. In Section V, we provide a brief summary.

II. Mismatches in Time-Interleaved ADCs

Although there are papers that address gain, timing and offset mismatch errors in time-interleaved ADCs from a linear signal processing perspective [8], [19]–[24], for completeness we briefly review the effect of frequency-dependent interleaved ADC mismatch errors and derive an equalizer to compensate for these errors. Note that in many practical applications, particularly those that involve detecting weak signals in the presence of strong interference (e.g., near-far problems), the effect of weak distortions landing on top of strong interference are insignificant. However, distortions generated by strong signals may interfere with or obscure the detection and/or demodulation of weak signals, and in practice these distortions must be eliminated.

A. Effect of Frequency-Dependent Gain, Timing, and Offset Mismatch

Consider the case of two ADCs, each sampling at half the Nyquist rate of $f_s = 1/T_s$, whose outputs are interleaved to produce an aggregate set of samples that are evenly spaced in time, where the following analysis easily extends to more than two converters. Let Δ denote the sampling phase error that is offset from the ideal 180° phase difference between the two ADC sample clocks, let $h_{\mathrm{ADC}_1}(t)$ and $h_{\mathrm{ADC}_2}(t)$ denote the linear impulse responses (gain, phase and memory) of the individual ADCs, and let $q(t)$ represent the analog input signal. Then, if $q_i(n) \triangleq q_i(nT_s)$ denotes the convolution of $q(t)$ with $h_{\mathrm{ADC}_i}(t)$ at integer sampling index n plus a fixed offset ζ_i, i.e., $q_i(n) = (q \star h_{\mathrm{ADC}_i})(nT_s) + \zeta_i$ where the symbol $\star$ denotes the convolution operator, then the upsampled output of each of the ADCs before interleaving can be modeled by

$$\begin{aligned} x_1(n) &= \frac{1}{2}\left(q_1(n) + q_1(n)e^{j\pi n}\right) \\ &\overset{\mathcal{F}}{\Longleftrightarrow} \frac{1}{2}\left(Q_1(\omega) + Q_1(\omega-\pi)\right) \text{ and} \\ x_2(n) &= \frac{1}{2}\left(q_2(n+\Delta) - q_2(n+\Delta)e^{j\pi n}\right) \\ &\overset{\mathcal{F}}{\Longleftrightarrow} \frac{1}{2}\left(Q_2(\omega)e^{j\omega\Delta} - Q_2(\omega-\pi)e^{j(\omega-\pi)\Delta}\right) \end{aligned} \quad (5)$$

where $\overset{\mathcal{F}}{\Longleftrightarrow}$ represents the symmetric Fourier transform pair. The interleaved output of the ADC is then expressed as

$$\begin{aligned} y(n) &= x_1(n) + x_2(n) \overset{\mathcal{F}}{\Longleftrightarrow} \frac{1}{2}\left(Q_1(\omega) + Q_1(\omega-\pi)\right) \\ &\quad + \frac{1}{2}\left(Q_2(\omega)e^{j\omega\Delta} - Q_2(\omega-\pi)e^{j(\omega-\pi)\Delta}\right) \\ &= Y(\omega). \end{aligned} \quad (6)$$

To clearly see the impact of the sampling phase error, consider the case where the impulse responses of the two ADCs are roughly equal, i.e., $Q_1(\omega) \approx Q_2(\omega) \triangleq Q(\omega)$. Plugging this into (6) and rearranging terms we get

$$\begin{aligned} Y(\omega) &= e^{j\frac{\omega\Delta}{2}} Q(\omega)\cos\left(\frac{\omega\Delta}{2}\right) \\ &\quad + \underbrace{e^{j\left(\frac{(\omega-\pi)\Delta}{2} - \frac{\pi}{2}\right)} Q(\omega-\pi)\sin\left(\frac{(\omega-\pi)\Delta}{2}\right)}_{\text{residual distortion at the image frequency}}. \end{aligned} \quad (7)$$

The first term to the right of the equal sign in (7) corresponds to the Nyquist rate output of the interleaved ADC, scaled in amplitude by $\cos(\omega\Delta/2)$ so that as $\Delta \to 0$, the scale factor approaches 1. The second term corresponds to a *residual* distortion term at the image frequency $\omega - \pi$, and this residual distortion term goes to 0 as $\Delta \to 0$, and conversely, grows as Δ becomes large.

B. ADC Mismatch Distortion Compensation

In a time-interleaved ADC with N_I constituent converters, the distortions that are generated by the constant offsets ζ_i are located at the fixed frequencies $(\omega_s/N_I)k$, for $k = 1, \ldots, N_I - 1$. Suppressing these distortions simply requires estimation and subtraction without application of a stimulus.

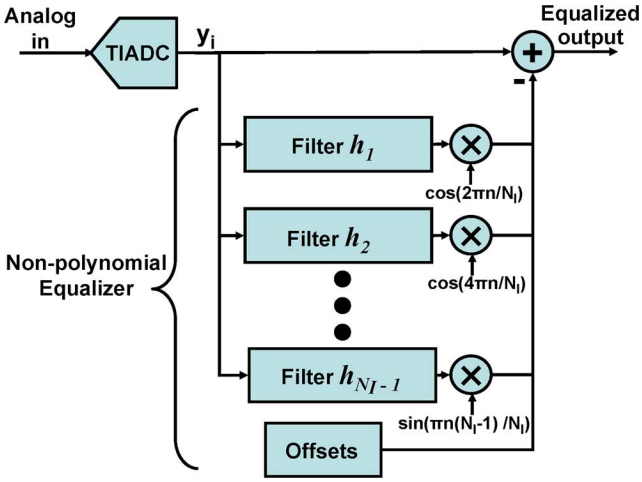

Fig. 2. Equalizer structure for frequency-dependent gain, timing, and offset mismatches in time-interleaved ADCs.

For frequency-dependent gain and timing mismatches between N_I], [constituent converters, (5) can be recast as

$$x_i(n) = q_i(n + \Delta_i) \sum_{k=0}^{N_I-1} e^{j\frac{2\pi k}{N_I}(n+i-1)} \quad (8)$$

where x_i, for $i = 1, \ldots, N_I$, corresponds to the output from converter i, and the distortions now land at $N_I - 1$ image frequencies. To suppress these distortions, we use the equalizer illustrated in Fig. 2. The idea is to model the distortions by filtering the input data, upconverting to the image frequencies, and subtracting the output of the equalizer from the input signal to eliminate the residual distortion. Let each filter have P taps with coefficient vectors $h_k \in \mathbb{R}^P$ for $k = 1, \ldots, N_I - 1$. We collect M ADC output samples from which we construct the Toeplitz matrix given by

$$Y = [y_{P-1}, y_{P-2}, \ldots, y_0] \in \mathbb{R}^{M\times P} \quad (9)$$

where

$$y_j = [y(j+i), y(j+i+1), \ldots, y(j+i+M-1)]^T \quad (10)$$

so that Yh_k is the output of the kth filter. The index i is used to center the data over the P taps, and the symbol T is the matrix transpose operator. The output of each filter is upconverted to its corresponding image frequency via the product $D_k Y h_k$, where $D_k \in \mathbb{R}^{M\times M}$ is the diagonal matrix given by

$$D_k = \mathrm{diag}\left[1 \quad \cos\left(\frac{2\pi k}{N_I}\right) \quad \cdots \quad \cos\left(\frac{2\pi(M-1)k}{N_I}\right)\right] \quad (11)$$

for $k = 1, \ldots, \lceil (N_I - 1)/2 \rceil$, and

$$D_k = \mathrm{diag}\left[0 \quad \sin\left(\frac{2\pi\left(k - \left\lceil \frac{N_I-1}{2} \right\rceil\right)}{N_I}\right) \quad \cdots \quad \sin\left(\frac{2\pi(M-1)\left(k - \left\lceil \frac{N_I-1}{2} \right\rceil\right)}{N_I}\right)\right] \quad (12)$$

for $k = \lceil (N_I - 1)/2 \rceil + 1, \ldots, N_I - 1$, where the symbol $\lceil \cdot \rceil$ represents rounding up to the nearest integer, and separate sine and cosine diagonal matrices are used to ensure that h_k is real. The equalizer filter coefficients used to model the residual distortions after upconversion are derived by solving

$$h_{res} = \arg\min_{h} J(h) \tag{13}$$

where

$$J(h) = \left\| W_{\text{image}} y - W_{\text{image}} \left[D_1 Y \right| \cdots \left| D_{N_I - 1} Y \right] h \right\|_2 \tag{14}$$

$h = [h_1^T \ \ldots \ h_{N_I-1}^T]^T$, $y = [y(i), \cdots, y(i + M - 1)]^T$ is the vector of ADC outputs, and W_{image} is a pruned DFT matrix that keeps only the frequencies where the nonlinear distortions fall. To isolate the signals at the image frequencies during training, a tonal excitation set q is used across a preselected grid of frequencies with

$$W_{\text{image}} = \begin{bmatrix} W_{\text{image}_1}^H & W_{\text{image}_2}^H & \cdots & W_{\text{image}_{N_{\text{sets}}}}^H \end{bmatrix}^H \tag{15}$$

where N_{sets} is the number of different frequency excitation sets, and W_{image_ℓ} is analogously defined in (2).

III. Polyphase Nonlinear Equalization

Although the analysis in Section II addresses *mismatch* distortions caused by frequency-dependent gain and phase mismatches in the linear impulse response of time-interleaved ADCs, it does not address the *polynomial* and polynomial mismatch distortions that are generated by both circuit-based nonlinearities and mismatches in the nonlinear responses of the constituent converters. In this section, we combine the equalizer developed in Section II with a nonlinear equalizer developed to mitigate additional nonlinearities that are compounded by time interleaving the outputs of multiple ADCs.

A. Polynomial Basis

The nonlinear system response of an ADC can often be described with the Pth-order (truncated) polynomial series expansion [25]–[27]

$$\begin{aligned} y_{\text{NL}}(n) &= \sum_{p=2}^{P} \overline{h_p}\left[x(n)\right], \quad \text{where} \\ \underbrace{\overline{h_p}\left[x(n)\right]}_{y_p(n)} &= \sum_{m_1=0}^{N_p-1} \cdots \sum_{m_p=0}^{N_p-1} h_p(m_1, \ldots, m_p) \\ &\quad \times \prod_{l=1}^{p} x(n - m_l) \end{aligned} \tag{16}$$

where N_p is the memory depth in each dimension of the pth-order Volterra kernel and $h_p(m_1, \ldots, m_p)$ are the pth-order kernel coefficients. The number of nonredundant terms in $y_{\text{NL}}(n)$ is $\sum_{p=2}^{P} \binom{N_p+p-1}{p}$. Using N samples of $x(n)$, (16) can be rewritten in matrix form as

$$y_p = X^p(x) h_p \tag{17}$$

with

$$y_p = [y_p(n) \quad \ldots \quad y_p(n - N + 1)]^T \tag{18}$$

where each of the $\binom{N_p+p-1}{p}$ columns of $X^p(x)$ can be represented by

$$\left[\prod_{l=1}^{p} x(n - m_l) \quad \ldots \quad \prod_{l=1}^{p} x(n - m_l - N + 1) \right]^T.$$

The full series can now be expressed in matrix form as

$$y_{\text{NL}} = X(x) h \tag{19}$$

with nonlinear convolution matrix $X(x) = [X^2(x) \ \ldots \ X^P(x)]$ and $h = [h_2^T \ \ldots \ h_P^T]^T$. It was shown in [28] that a large class of nonlinear systems can be approximated with arbitrarily small error using the polynomial representation in (16). However, this comes with the disadvantage that a relatively large number of parameters (factorial in N and p) are needed to represent systems with modest polynomial order and memory. To reduce the computational complexity when using (16) to model ADC nonlinearities, we develop an efficient representation of (16) in a new coordinate system.

In [27], the kernels in (16) were rewritten in terms of elements of a horizontal coordinate system (HCS) in which a pth-order processing element (PE) is formulated as

$$\begin{aligned} y_p^{HCS}(n, \alpha_2, \ldots, \alpha_p) &= \underbrace{\sum_{m=i_1}^{N_p+i_1-1} h(m, \alpha_2, \ldots, \alpha_p) x(n - m)}_{h(n;\alpha_2,\ldots,\alpha_p) \star x(n)} \\ &\quad \times \prod_{l=2}^{p} x(n - \alpha_l). \end{aligned} \tag{20}$$

Hence, it is possible to sum all the pth-order HCS PEs to obtain the pth-order kernel

$$y_p(n) = \sum_{\alpha_2=i_2}^{N_p+i_2-1} \cdots \sum_{\alpha_p=i_p}^{N_p+i_p-1} y_p^{HCS}(n, \alpha_2, \ldots, \alpha_p) \tag{21}$$

where the variables i_k in (20), like i in (10), are used to center the data over the taps of the *multidimensional* filter. Equation (20) geometrically corresponds to coefficients selected along a single horizontal (m) dimension while the other dimensions of the pth-order kernel (α_2 to α_p) remain fixed. The HCS representation has a very appealing interpretation, which is that we can represent (16) as the sum of one-dimensional convolutions, each multiplied by the product of time-delayed values of the input.

Let the data matrix associated with the jth HCS PE of order p_j be defined as X^{PE_j} with the cth column given by

$$\begin{aligned} \left(X^{PE_j}(x)\right)_c &= \overline{x}(n - c_j - i_{1,j}) \circ \overline{x}(n - \alpha_{j,2}) \circ \cdots \\ &\quad \cdots \circ \overline{x}\left(n - \alpha_{j,p_j}\right), \\ &\text{for } c_j = 0, 1, \ldots, N_{taps}^{PE_j} - 1 \end{aligned} \tag{22}$$

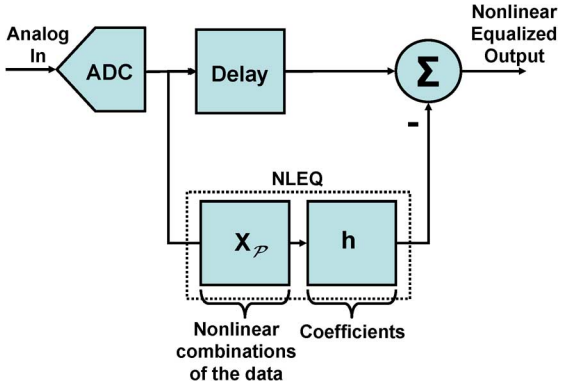

Fig. 3. Nonlinear equalization of ADC distortions by modeling and subtraction.

where $\overline{x}(n) = [x(n)\, x(n-1)\, \ldots\, x(n-M+1)]^T$, $\circ$ represents the Hadamard product, M is the number of samples, $c_j = 0, 1, \ldots, N_{taps}^{PE_j} - 1, N_{taps}^{PE_j}$ is the number of filter taps in HCS processing element j, and $p_j \in \{2, 3, \ldots, P\}$. Using (22), we can formulate an approximation to (16) in which

$$\widehat{y}(n) = [X^{PE_1} \quad X^{PE_2} \quad \ldots \quad X^{PE_K}] \begin{pmatrix} h^{PE_1} \\ h^{PE_2} \\ \vdots \\ h^{PE_K} \end{pmatrix} \tag{23}$$

where $X_{\mathcal{P}} = [X^{PE_1}, \ldots, X^{PE_K}]$ is the data matrix associated with processing element set $\mathcal{P}$. The construction in (23) both simplifies the computational burden of using sequential estimation in architecture identification, described next, and provides a regular structure for hardware implementation.

B. Nonlinear Equalizer Architecture Identification

Using (23), we can derive an equalizer architecture for mitigating harmonic and intermodulation (nonlinear) distortion generated by an ADC, as illustrated in Fig. 3, where the nonlinear response of the ADC is estimated and subtracted from an appropriately delayed version of the ADC output. The objective of this section is to present techniques that enable the construction of a (polyphase) polynomial equalizer using a minimum set of PEs by efficiently searching over the multidimensional signal space.

1) Forward–Backward Sequential Estimation in Architecture Identification: In this section, we develop a sequential algorithm for selecting PEs that minimize the mean square error, ε. Let $\varepsilon(y; \mathcal{P}) = \min_h \|y - X_{\mathcal{P}} h\|_2$ denote the modeling error, where $X_{\mathcal{P}} = [X^{PE_1}, \ldots, X^{PE_L}]$, $\mathcal{P}$ is the set of PEs in the architecture with cardinality $|\mathcal{P}| = L$, and y is the vector of ADC output samples. Let $\mathcal{P} \subset \mathcal{X}$, where $\mathcal{X}$ is the set of *user-defined* candidate processing elements with $|\mathcal{X}| \gg L$, then the pseudocode for sequentially selecting processing elements to construct an NLEQ architecture is shown in Fig. 4. The total number of processing elements $|\mathcal{X}|$ can be adjusted according to computational considerations; the PEs that comprise a set are unique in their polynomial order, delay values and filter coefficients. The parameter δ in the outer loop of the pseudocode is a threshold for the MMSE; alternatively a fixed number of iterations can be used. In either case, the parameters k and m, where $m \leq k$, are user defined and (can) change from iteration to iteration. In our experiments, m is typically set to constrain real-time computational complexity and k is set to be at most $2m$.

```
initialize P = ∅
while ε(y; P) ≥ δ   {
        // Add Processing Elements
        while |P| ≤ k   {
          p ← arg min_{p∈X} ε(y; P ∪ {p})
          P ← P ∪ {p}, X ← X\{p}   };
        // Remove Processing Elements
        while |P| ≥ m   {
          p ← arg min_{p∈P} ε(y; P\{p})
          P ← P\{p}, X ← X ∪ {p}   };
  };
```

Fig. 4. Pseudocode for forward–backward sequential estimation. Each new PE j from the set $\mathcal{X}$ that yields the best minimum mean square error (MMSE) performance in combination with the previous $j-1$ PEs is added to the set $\mathcal{P}$ in the forward stage. In the backward stage, a PE is removed one at a time from $\mathcal{P}$ and placed back into $\mathcal{X}$ such that the PE removed has the least impact on MMSE performance.

2) Global Estimation in Architecture Identification: An alternative to forward–backward sequential estimation is to formulate the problem of NLEQ architecture identification as a constrained optimization, in which the constraints are imposed to insure a computationally efficient solution. In [29], basis pursuit was used to find a sparse set of coefficients in a *linear* system identification problem. However, unlike linear systems, the size of the convolution matrix $X(x)$ representing the nonlinear combinations of the input data can grow prohibitively large. One approach to reducing the dimensionality leverages the following theorem.

Theorem 1: Let $X \in \mathbb{R}^{N \times M}$, with $N \gg M$, be a nonsingular matrix whose columns correspond to the p-fold products of the input, and let y correspond to some vector in $\mathbb{R}^N$. Then there exists a projection matrix $P \in \mathbb{R}^{M \times N}$ with $\widehat{X} = PX \in \mathbb{R}^{M \times M}$, and hence an orthogonal projection matrix $P_{\widehat{X}\widehat{X}}^{\perp} = (I - \widehat{X}(\widehat{X}^T \widehat{X})^{-1} \widehat{X}^T)$, such that $\widehat{y}^T P_{\widehat{X}\widehat{X}}^{\perp} \widehat{y} = y^T P_{XX}^{\perp} y$, where $\widehat{y} = Py$.

Proof: Consider a matrix X with full column rank having singular value decomposition

$$X = [U \quad \overline{U}] \begin{bmatrix} D \\ 0 \end{bmatrix} V^T \in \mathbb{R}^{N \times M},$$

with $U \in \mathbb{R}^{N \times M}$, and $N \gg M$. Further, consider the solutions to the least squares problems $h = \arg\min_h \|y - Xh\|$ and $h^* = \arg\min_h \|Py - PXh\|$, where P is a $M \times N$ matrix that projects any N-dimensional vector v down to an M-dimensional space. Then if $P = U^T$, $h^* = h$.

It is possible to use the left singular vectors to reduce the dimensionality without any loss in performance; however, the computational cost of computing the SVD (singular value decomposition) of a large matrix, coupled with the fact that Theorem 1 is only valid for $N \geq M$, is of very little practical value. We propose two techniques for reducing the dimensionality in a computationally efficient fashion. To reduce the dimensionality of the rows, we leverage the following lemma [30].

Lemma 1 (Johnson–Lindenstrauss): For any $0 < \epsilon < 1$ and any integer n, let K be a positive integer such that $K \geq$

$4(\epsilon^2/2-\epsilon^3/3)^{-1}\ln n$. Then for any set V of n points in $\mathbb{R}^N$, there is a map $f:\mathbb{R}^N\rightarrow\mathbb{R}^K$ such that for $u,v\in V$

$$(1-\epsilon)\|u-v\|\leq\|f(u)-f(v)\|\leq(1+\epsilon)\|u-v\|. \quad (24)$$

Extensions to the Johnson–Lindenstrauss lemma [31] have used concentration measure theory to show that if points in a vector space are projected onto a *randomly* selected subspace of suitably high dimension then the distances between the points (in a Euclidean sense) are approximately preserved. Therefore, we can construct a matrix $\Phi\in\mathbb{R}^{K\times N}$ whose elements are randomly drawn from a Gaussian distribution $(\mathcal{N}(0,(1/\sqrt{K})))$, so that the projected matrix given by $\Phi X\in\mathbb{R}^{K\times M}$ where $K<N$ does not impart a significant error during architecture identification.

To further reduce the size of the matrix X, its columns can be pruned by projection and subtraction. First, the M columns of ΦX are unit normalized so that each column can be considered as the coordinate of a point on the surface of a unit hypersphere. From the set of M vectors, L vectors are selected one at a time, such that the mth vector chosen has the highest correlation with $\Phi y^{(m-1)}$ after the $m-1$ projections of the previously selected columns have been subtracted off, that is

$$\begin{aligned}\Phi y^{(1)}&=\Phi y-\left\langle(\Phi X)_{i(1)},\Phi y\right\rangle(\Phi X)_{i(1)}\\ \Phi y^{(2)}&=\Phi y^{(1)}-\left\langle(\Phi X)_{i(2)},\Phi y^{(1)}\right\rangle(\Phi X)_{i(2)}\\ &\vdots\\ \Phi y^{(L)}&=\Phi y^{(L-1)}-\left\langle(\Phi X)_{i(L)},\Phi y^{(L-1)}\right\rangle(\Phi X)_{i(L)}.\end{aligned} \quad (25)$$

In (25), the symbol $(\Phi X)_{i(m)}$ is the column of ΦX that has the highest correlation with $\Phi y^{(m-1)}$, whose column index is given by $i(m)$, and $\langle x,y\rangle$ is the dot product of x and y. The goal of projection and subtraction is to find the points on the unit hypersphere with widest angular separation that have a significant projected component on the received data. We use projection and subtraction to reduce the dimensionality of the data so that the subsequent constrained optimization is computationally tractable. Defining the matrix $\Theta\in\mathbb{R}^{M\times L}$ such that $X\Theta$ keeps only the columns of X we get from projection and subtraction in (25), the constrained optimization to find a sparse NLEQ architecture is given by

$$\begin{aligned}&\hat{h}=\arg\min_h e^T h\\ s.t.\ &\|\Phi y-\Phi[X\Theta|-X\Theta]h\|_2\leq\epsilon\\ &h\geq 0\end{aligned} \quad (26)$$

where $e=[1\ 1\ \ldots\ 1]^T\in\mathbb{R}^{2L}$ and $\hat{h}_+,\hat{h}_-\in\mathbb{R}^L$, with $\hat{h}=[\hat{h}_+^T\ \hat{h}_-^T]^T$ and $h_{\rm opt}=\hat{h}_+-\hat{h}_-$. Equation (26) is easily solved using second order cone programming (SOCP) [32]. The basis pursuit (BP) cost function, along with the constraint that $h\geq 0$, in (26) is an ℓ_1 norm that favors sparse solutions [29]. The scalar regularization parameter ϵ in the constraint balances the residual ℓ_2 reconstruction error, that is, it ensures that the sparse solution $h_{\rm opt}$ whose nonzero entries span the columns of $\Phi X\Theta$ is in the cone of feasible solutions. Note that unlike forward–backward sequential estimation, the columns selected (nonzero entries of $h_{\rm opt}$) correspond to *single-tap* processing elements, where individual PEs are not connected with any specific coordinate system (e.g., horizontal, etc.).

C. pNLEQ Formulation

Like in the case of linear mismatches, there are also distortions at the (nonlinear) image frequencies due to mismatches in the nonlinear response between on-chip converters. Consider the case of a simple third-order nonlinearity that is present in the converters on a two-way interleaved ADC ($N_I=2$), but whose response (amplitude and phase) is somewhat different. Let

$$\begin{aligned}x_1^{\rm NL}(n)&=\gamma_1\left(\prod_{i=1}^{3}q(n-\alpha_i)\right)(1+e^{j\pi n})\\ &\overset{\mathcal{F}}{\Leftrightarrow}\gamma_1\underbrace{\left(Q(\omega)e^{-j\omega\alpha_1}\star Q(\omega)e^{-j\omega\alpha_2}\star Q(\omega)e^{-j\omega\alpha_3}\right)}_{Z_1(\omega)}\\ &\quad\star(\delta(\omega)+\delta(\omega-\pi))\\ &=\gamma_1\left(Z_1(\omega)+Z_1(\omega-\pi)\right),\end{aligned} \quad (27)$$

where $\gamma_1=h_3(\alpha_1,\alpha_2,\alpha_3)$ from (16) is a constant associated with the third-order nonlinear response of converter 1, and

$$\begin{aligned}x_2^{\rm NL}(n)&=\gamma_2\left(\prod_{i=1}^{3}q(n+\Delta-\beta_i)\right)(1-e^{j\pi n})\\ &\overset{\mathcal{F}}{\Longleftrightarrow}\gamma_2\left(Z_2(\omega)-Z_2(\omega-\pi)\right)\end{aligned} \quad (28)$$

such that the sampling phase error Δ is absorbed in the composite third-order nonlinear response $Z_2(\omega)$, and γ_2 is a constant associated with the third-order nonlinear response of converter 2. Equations (27) and (28) have the same form as (5), where there is a distortion term at both the frequency associated with the third-order polynomial nonlinearity, and a *mismatch* distortion at its image frequency.

To mitigate the distortions exemplified in (27) and (28), and those generated by linear mismatches, we now combine the linear mismatch equalizer in (13) with a polyphase version of the polynomial matrix $X\in\mathbb{R}^{N\times M}$ from (26) for N_I-way interleaving. Define

$$X_{{\rm poly}_{N_I}}(y)=\begin{bmatrix}(X(y))_1&\cdots&0\\ \vdots&\ddots&\vdots\\ 0&\cdots&(X(y))_{N_I}\\ \vdots&&\vdots\\ (X(y))_{N-N_I-1}&\cdots&0\\ \vdots&\ddots&\vdots\\ 0&\cdots&(X(y))_N\end{bmatrix} \quad (29)$$

where $(X)_j$ corresponds to the jth row of X. Then the ℓ_2 component used to formulate the polyphase nonlinear equalizer using sequential estimation in Fig. 4 or SOCP in (26) is given by

$$\left\|\Phi Wy-\Phi W\left[D_1Y|\cdots|D_{N_I-1}Y|X_{{\rm poly}_{N_I}}(y)\Theta\right]h\right\|_2. \quad (30)$$

with $h=[h_1^{lin^T}\ \ldots\ h_{N_I-1}^{lin^T},\ h_1^{NL^T}\ \ldots\ h_{N_I}^{NL^T}]^T$, where h_i^{lin} and $h_i^{\rm NL}$ corresponds to the coefficients associated with the ith linear and nonlinear equalizer, respectively, and D_k for $k=$

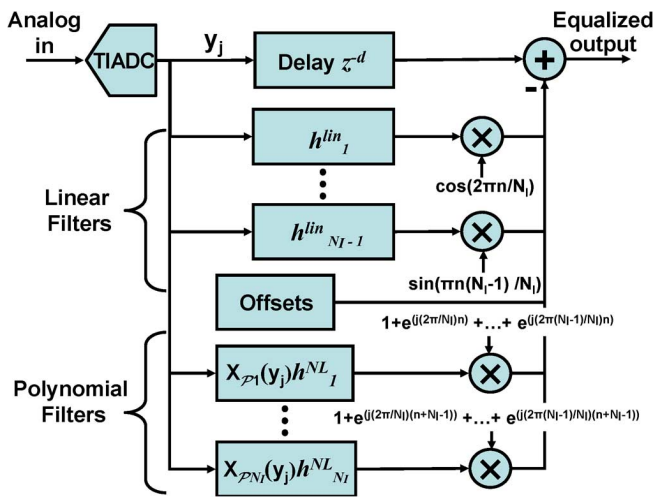

Fig. 5. The polyphase nonlinear equalization (pNLEQ) architecture for an N_I-way time-interleaved ADC. $N_I - 1$ linear filters are used to remove distortions generated by the linear frequency-dependent gain and phase mismatches, and N_I polynomial filters are used to remove distortions due to the nonlinear response of the constituent converters.

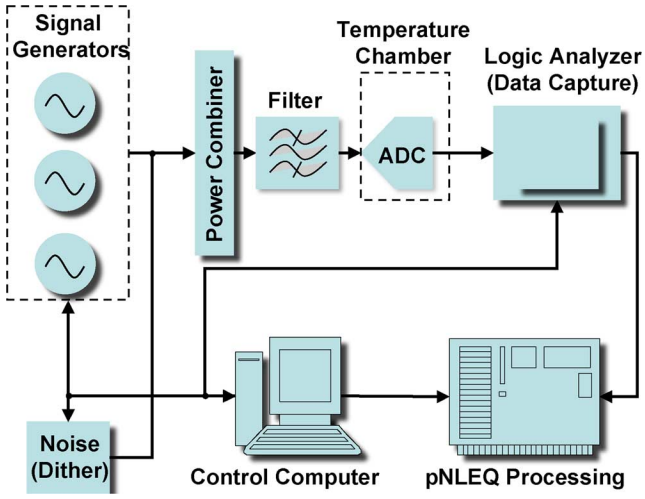

Fig. 6. Simplified view of the the Lincoln Laboratory NLEQ testbed. The testbed consists of three Agilent E8257D tone generators connected to an analog power combiner, an Agilent 16702B Logic Analyzer used for data capture, a temperature chamber where the time-interleaved ADCs are seated, and a Windows-based PC running Matlab to control the instrumentation for excitation and data capture.

$1, \ldots, N_I - 1$ and Y are defined in (13). The overall pNLEQ architecture is illustrated in Fig. 5, where there are $N_I - 1$ *linear* filters and offsets to equalize distortions generated by mismatches in the overall linear response of the constituent converters. N_I polynomial filters are used to model the nonlinear distortions and mismatches between the nonlinear responses of the constituent converters, which are also subtracted from the ADC output. In Fig. 5, $X_{\mathcal{P}i}(y_j)h_i^{\mathrm{NL}}$ corresponds to the set $\mathcal{P}_i$ of processing elements chosen using (30) that comprise the ith component of the polyphase nonlinear equalizer, where $i = 1, \ldots, N_I$, and y_j corresponds to the output of the ADC when stimulated by (analog) tone set $q_j(t)$. Each component i of the nonlinear equalizer operates on data at the full Nyquist rate, and outputs data at $1/N_I$ the full rate. This is illustrated in Fig. 5, which shows the output of each filter being selected in a round-robin fashion using $1 + e^{j(2\pi/N_I)(n+i-1)} + \ldots + e^{j(2\pi(N_I-1)/N_I)(n+i-1)}$, where i corresponds to ith equalization component of pNLEQ. If the ADC outputs data from each of the on-chip converters on a separate set of output pins in parallel at $1/N_I$ the clock rate, which we have found is often the case in practice, then synchronizing the selection of the ith nonlinear equalization component is trivial.

In summary, the polyphase nonlinear equalizer has the following advantages.

1) pNLEQ simultaneously mitigates both linear and nonlinear (harmonic and intermodulation) mismatch distortions in time-interleaved ADCs.
2) Using forward-backward sequential estimation or SOCP we can find a computationally efficient pNLEQ solution to mitigate distortions with state-dependent behavior, i.e., nonlinearities with memory.
3) The pNLEQ approach works with current interleaved ADC architectures.

In the next section, we demonstrate and compare the performance of pNLEQ to other techniques using measured results.

TABLE I
SOURCE EXCITATION AND VERIFICATION PARAMETERS

ADC	Excitation	Verification
Maxim MAX101	22 one-tone sets 60 two-tone sets	100 one-tone sets 280 two-tone sets
National Semiconductor ADC081000	39 one-tone sets 60 two-tone sets 262 three-tone sets	350 one-tone sets 280 two-tone sets 2340 three-tone sets

IV. MEASURED RESULTS

A. Test Setup

To evaluate pNLEQ performance, we used the MIT Lincoln Laboratory NLEQ testbed depicted in Fig. 6. The analog outputs of three Agilent E8257D tone generators were combined, filtered, and injected into a time-interleaved ADC that was seated in a temperature-controlled chamber set to 20 °C. A Windows PC running MATLAB scheduled the tone generators, controlled the Agilent 16702B logic analyzer that was used to capture data at the output of the ADC, and transferred data from the logic analyzer's memory back to the PC's hard drive. We tested two 8-bit time-interleaved ADCs: the Maxim MAX101, sampling at a rate of 500 MS/s and the National Semiconductor ADC081000 sampling at a rate of 983.04 MS/s.

B. Training and Verification

We trained both devices with a series of one-, two-, and three-tone sets. We spaced the tones across a 345-MHz band of interest for the ADC081000, and a 200-MHz band for the MAX101. The number and type of tone sets used to excite the linear and nonlinear modalities in the ADCs are listed in Table I. In all cases, the pNLEQ architecture's computational complexity was constrained so that it could be efficiently implemented in hardware.

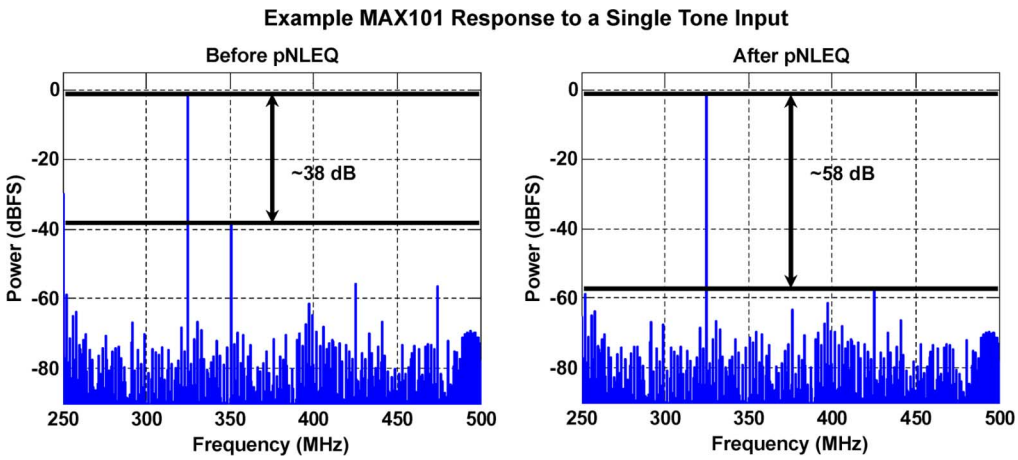

Fig. 7. Representative response of the MAX101 to a single-tone stimulus. The plot to the left represents the unequalized output of the ADC, with both harmonic and mismatch distortions present. The plot to the right represents the equalized response of the MAX101 after running pNLEQ, which demonstrates roughly 20-dB improvement in SFDR.

TABLE II
MAXIM MAX101 ONE-TONE EQUALIZATION RESULTS

Equalization Technique	MDR (dBFS)	MDR Improvement (dB)	Operations per Sample
None	-36.9	0	0
Linear Mismatch	-36.9	0	10
Linear Mismatch + Memoryless Polynomial	-39.7	2.8	22
pNLEQ	-55.8	18.9	198

TABLE III
MAXIM MAX101 TWO-TONE EQUALIZATION RESULTS

Equalization Technique	MDR (dBFS)	MDR Improvement (dB)	Operations per Sample
None	-42.4	0	0
Linear Mismatch	-42.4	0	10
Linear Mismatch + Memoryless Polynomial	-45.1	2.7	22
pNLEQ	-58.3	15.9	198

After identifying a pNLEQ architecture with the techniques presented in Section III, we cross-validated pNLEQ performance using a sequence of verification tone sets. These one-, two-, and three-tone verification sets were entirely different from the training sets used to derive the coefficients. We quantify dynamic range by taking the mean of the individual SFDR/IFDR measurements for each of the verification tone sets. This is the *mean dynamic range* (MDR) performance metric, which we measure in dBFS.

C. Results

In Tables II and III, we list the results for the MAX101. In this case, the MDR is computed over the one-tone and two-tone verification sets. For the one-tone data, pNLEQ gives −55.8-dBFS mean dynamic range, which is 18.9 dB greater than the performance without any equalization or with only linear mismatch compensation and 16.1 dB greater than the performance with linear mismatch compensation and memoryless polynomial equalization. Similarly, for two-tone data, pNLEQ provides −58.3-dBFS MDR, which is 15.9 dB greater than the performance without equalization and 13.2 dB greater than the improvement with linear mismatch compensation and memoryless polynomial equalization. We also report the complexity of each equalization technique in operations per sample, where the total number of operations is the sum of the number of additions and multiplications. Note that each polyphase HCS component requires $(M + p - 1)/N_I$ multiplications and M/N_I additions per sample, where M is the number of taps in the polyphase FIR filter, p is the polynomial order, and N_I is the number of interleaved ADCs. Here we use ten-tap polyphase HCS components up to fifth order, and pNLEQ requires 198 operations per sample. In Fig. 7, we show the response of the MAX101 to a single sinusoidal input (in the second Nyquist zone) both before and after pNLEQ. This example yields 20-dB improvement in SFDR. Similarly, Fig. 8 shows 23-dB improvement in IFDR for an example two-tone set. In addition to these examples, we plot the SFDR as a function of frequency in Fig. 9, and this graph demonstrates that pNLEQ outperforms the phase plane error compensation method [16] by roughly an order of magnitude.

We also derived a pNLEQ architecture for the ADC081000 and we present a summary of our results in Table IV. For this ADC, the MDR measurements are computed over the three-tone verification sets. In this case, pNLEQ provided a 15.4-dB improvement in MDR with a computational complexity of 220 operations per sample. This is 12.1 dB greater than the improvement using a memoryless polynomial equalizer with linear mismatch compensation. In Figs. 10 and 11, we show examples of the ADC response before and after pNLEQ in which we achieve ∼15-dB and ∼20-dB improvement in dynamic range. In this case, we also compare the performance of pNLEQ to a Volterra equalizer constructed over the space from which we selected 8-tap HCS processing elements using forward–backward sequential estimation. The space from which we selected HCS processing elements was comprised of a second-order kernel with process delays (i_ks in (21)) ranging from −2 to 6, a third-order kernel with process delays ranging from 0 to 3, and a fourth-order kernel with process delays of only 0. The Volterra approach required over an order of magnitude more operations per sample, while only achieving an additional 1.1 dB of dynamic range improvement over pNLEQ. We did not derive a

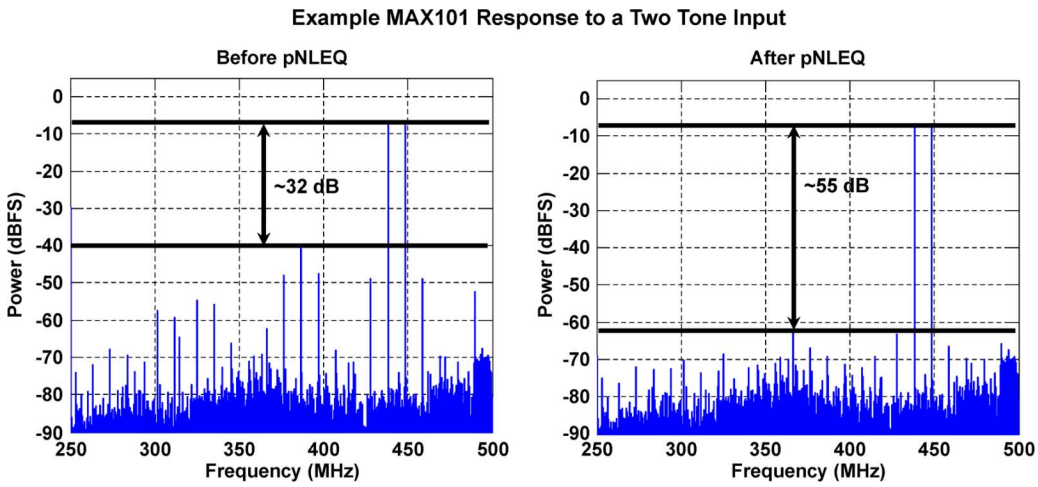

Fig. 8. Representative response of the MAX101 to a two-tone stimulus. The plot to the left represents the unequalized output of the ADC, with both intermodulation, harmonic and mismatch distortions present. The plot to the right represents the equalized response of the MAX101 after running pNLEQ, which demonstrates roughly 23-dB improvement in dynamic range.

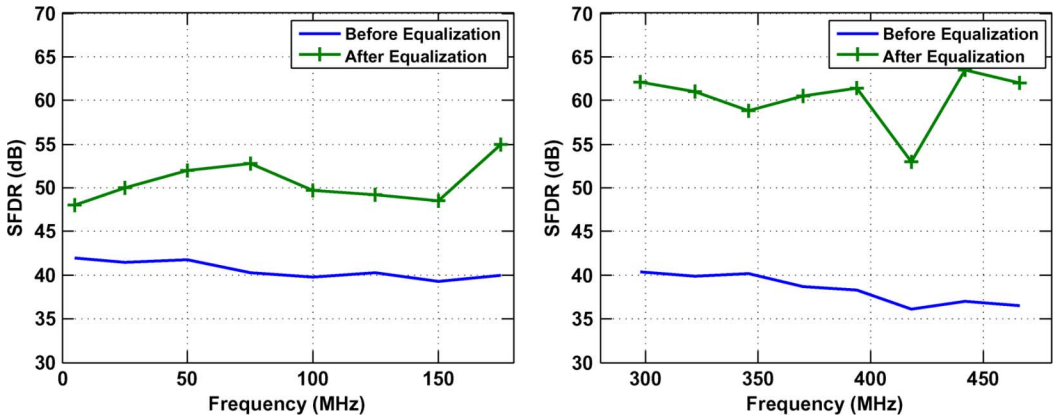

Fig. 9. SFDR as a function of frequency for one tone inputs to the MAX101. The plot on the left shows the results for phase plane error compensation method [16]. The plot on the right shows the results for pNLEQ. Note that the authors in [16] operated the MAX101 in the first Nyquist zone (baseband sampling) to obtain the results for the phase plane error compensation method. Due to the ready availability of analog filters in our lab, the MAX101 was operated in the second Nyquist zone (IF sampling) to obtain pNLEQ performance results.

TABLE IV
NATIONAL SEMICONDUCTOR ADC081000
THREE-TONE EQUALIZATION RESULTS

Equalization Technique	MDR (dBFS)	MDR Improvement (dB)	Operations per Sample
None	-55.5	0	0
Linear Mismatch	-55.5	0	10
Linear Mismatch + Memoryless Polynomial	-58.8	3.3	19
pNLEQ	-70.1	15.4	220
Volterra	-71.2	16.5	2242

Volterra equalizer for the MAX101 over the space from which we constructed pNLEQ using second- through fifth-order kernels, memory depth 10 and process delays ranging from -2 to 3, as this was computationally prohibitive.

D. Convergence

In Fig. 12, we plot the MDR in dBFS versus the number of operations per sample of pNLEQ operating on verification data. In both cases, pNLEQ was derived using forward–backward sequential estimation. For the MAX101, 11 polyphase HCS PEs achieve an MDR of -57 dBFS using both one- and two-tone verification sets. For the ADC081000, 17 polyphase HCS PEs achieve an MDR of roughly -70 dBFS. In all cases, we found that backward iterations helped only marginally ($\sim$1 dB), as the forward iteration of sequential estimation was able to identify HCS components on the first pass that yielded performance on par with a full Volterra equalizer as is shown in Table IV.

Fig. 13 is a plot of the convergence of basis pursuit (BP) against the convergence of sequential estimation during the training phase of pNLEQ. For BP, we pruned the number of columns of the matrix $X_{\text{poly}_2}(y)$ from 4745 to 1107 using (25), and we reduce the number of rows from from 60 522 to 2000 using random linear projections (*Lemma* 1). The 60 522 rows correspond to 262 3-tone sets with 231 samples per set; the 4475 rows correspond to 8-tap filters with six incremental process delays. A normalized mean square error metric

$$\text{MSE} = \frac{\left\|\Phi W y - \Phi W \left[D_1 Y \quad X_{\text{poly}_2}\Theta\right] h\right\|}{\|\Phi W y\|} \tag{31}$$

was used to assess performance on the training data. For BP, ϵ in (26) was set to achieve the desired level of MSE performance. After selecting 17 PEs, sequential estimation achieved an MSE of -13 dB, with each iteration (PE selection) taking roughly as

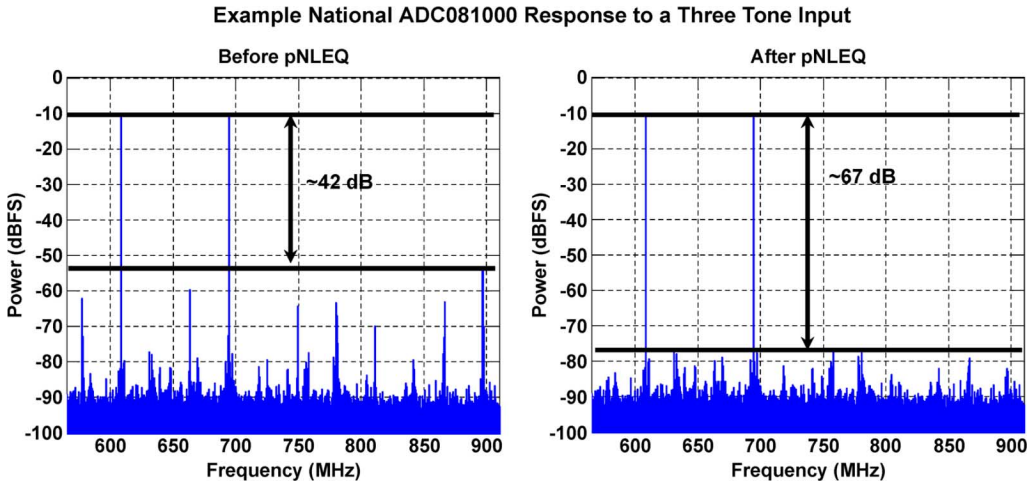

Fig. 10. Representative response of the National Semiconductor ADC081000 to a three-tone stimulus. (Two of the tones are very close together.) The plot to the left represents the unequalized output of the ADC, with both intermodulation, harmonic and mismatch distortions present. The plot to the right represents the equalized response of the ADC081000 after running pNLEQ, which demonstrates roughly 25-dB improvement in dynamic range.

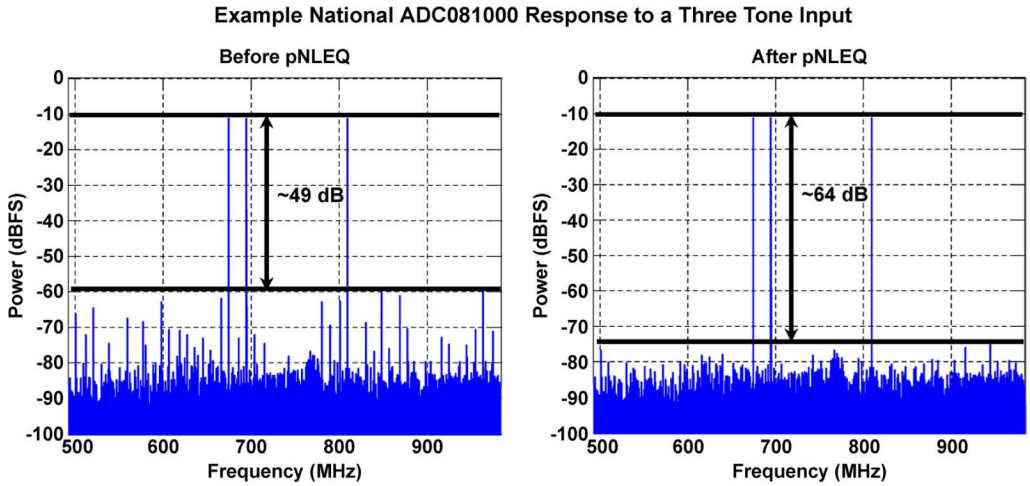

Fig. 11. Representative response of the National Semiconductor ADC081000 to a three-tone stimulus. The plot to the left represents the unequalized output of the ADC, with both intermodulation, harmonic and mismatch distortions present. The plot to the right represents the equalized response of the National Semiconductor ADC081000 after running pNLEQ, which demonstrates roughly 15 dB improvement in dynamic range.

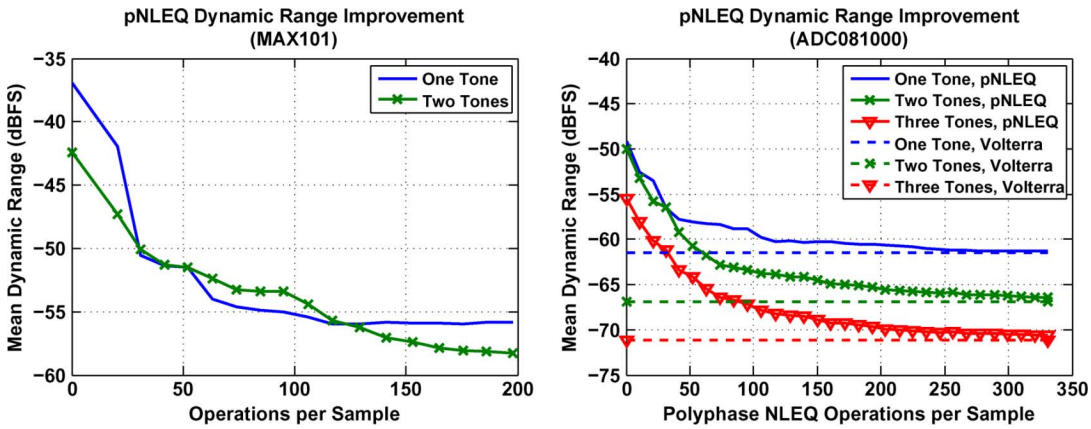

Fig. 12. Computational complexity of pNLEQ vs. mean dynamic range for sequential estimation with a polyphase HCS architecture for the MAX101 (left) and the National Semiconductor ADC081000. The dotted horizontal lines in the plot to the right indicate MDR performance of a Volterra equalizer for a fixed number (i.e., 2242) of operations per sample. The Volterra equalizer was composed of second- through fourth-order kernels with memory depth 8 and process delays (i_ks in (21)) ranging from -2 to 3, corresponding to the space from which (polyphase) HCS components were selected via sequential estimation.

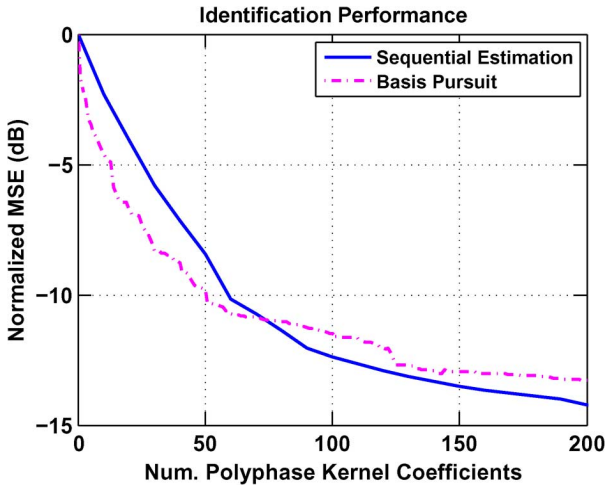

Fig. 13. Convergence of the sequential estimation and basis pursuit algorithms for pNLEQ identification for the National Semiconductor ADC081000. In the case of sequential estimation, each HCS processing element has eight taps, where each tap corresponds to a (kernel) coefficient multiplying a column in the nonlinear convolution matrix $X_{\mathrm{poly}_{N_I}}(y)$. Basis Pursuit selects a set of coefficients (*not* constrained to form processing elements) to satisfy a mean square error constraint and minimize a sparsity objective.

long as an iteration of (26). Although BP and sequential estimation performed nearly as well, we found that sequential estimation was able to achieve a slightly lower MSE than BP. The slight difference in performance was principally due to having to project the data down into a much lower dimensional subspace for BP computational tractability.

V. Summary

We presented a method for constructing a computationally efficient polyphase nonlinear equalizer (pNLEQ) for mitigating polynomial distortions as well as mismatches in the linear and nonlinear responses of the constituent converters of time-interleaved ADCs.

We developed a (polyphase) polynomial basis in a new coordinate space (HCS) that enabled us to use forward–backward sequential estimation to construct a computationally efficient nonlinear equalizer to mitigate nonlinear distortions in ADCs. In addition, we leveraged recent work in compressed sensing [29] and dimensionality reduction to formulate a constrained optimization problem for NLEQ architecture identification.

We combined the linear mismatch equalizer and NLEQ into a polyphase nonlinear equalizer that accounted for linear and nonlinear frequency-dependent timing and phase mismatches in time-interleaved ADCs. Using pNLEQ, we demonstrated 1 to 2 orders of magnitude improvement in dynamic range using data collected from the National Semiconductor ADC081000 and Maxim MAX101 time-interleaved ADCs on MIT Lincoln Laboratory's NLEQ testbed. The computational complexity of pNLEQ to achieve this result was on the order of 100–200 operations/sample, which was an order of magnitude less than a Volterra equalizer for roughly the same level of equalization performance.

Polyphase NLEQ outperformed the phase plane error lookup table method by nearly 10 dB on average. Polyphase NLEQ also outperformed a linear mismatch and memoryless (polyphase) polynomial equalizer by over an order magnitude (12–16 dB) on a multitone stimulus set, demonstrating the need to mitigate frequency-dependent linear *and* frequency-dependent polynomial mismatch distortions in time-interleaved ADCs.

Acknowledgment

The authors would like thank Dr. D. Healy of DARPA for his generous support and encouragement of our research in nonlinear signal processing. Special thanks go to Dr. J. Anderson, Dr. P. Monticciolo, and Dr. M. Vai for their helpful comments on various aspects of this paper. Finally, the authors would like to acknowledge T. Emberley for running the experiments that were instrumental in generating the results for this paper.

References

[1] A. Abidi, "The path to the software-defined radio receiver," *IEEE J. Solid State Circuits*, vol. 42, no. 5, pp. 954–966, May 2007.
[2] A. Salkintzis, H. Nie, and P. Mathiopoulos, "ADC and DSP challenges in the development of software radio base stations," *IEEE Personal Commun.*, vol. 6, no. 4, pp. 47–55, Aug. 1999.
[3] R. Walden, "Analog-to-digital converter survey and analysis," *IEEE J. Sel. Areas Commun.*, vol. 17, no. 4, pp. 539–550, Apr. 1999.
[4] R. Blazquez *et al.*, "A baseband processor for impulse ultra-wideband communications," *IEEE J. Solid State Circuits*, vol. 40, no. 9, pp. 1821–1828, Sep. 2005.
[5] D. Martinez, M. Vai, and R. Bond, *High Performance Embedded Computing Handbook: A Systems Perspective*. Boca Raton, FL: CRC, June 2008.
[6] F. Adamo *et al.*, "A/D converters nonlinearity measurement and correction by frequency analysis and dither," *IEEE Trans. Instrum. Meas.*, vol. 52, no. 4, pp. 1200–1205, Aug. 2003.
[7] W. Black and D. Hodges, "Time interleaved converter arrays," *IEEE J. Solid State Circuits*, vol. SSC-15, no. 6, pp. 1022–1029, Dec. 1980.
[8] J. Elbornsson *et al.*, "Blind adaptive equalization of mismatch errors in time-interleaved A/D converter systems," *IEEE Trans. Circuits Syst.*, vol. 51, no. 1, pp. 151–159, Jan. 2004.
[9] S. Huang and B. Levy, "Blind calibration of timing offsets for four-channel time-interleaved ADCs," *IEEE Trans. Circuits Syst.*, vol. 54, no. 4, pp. 863–866, Apr. 2007.
[10] C. Vogel, "Time-interleaved analog-to-digital converters: Status and future directions," in *Proc. IEEE Int. Symp. Circuits Syst.*, 2006, pp. 3386–3389.
[11] V. Divi and G. Wornell, "Signal recovery in time-interleaved analog-to-digital converters," in *Proc. IEEE ICASSP*, 2004, pp. 593–596.
[12] S. Velazquez, "Linearity error compensator," U.S. patent 6,198,416, Mar. 6, 2001.
[13] N. Bjorsella, P. Suchanek, P. Handel, and D. Ronnow, "Measuring Volterra kernels of analog-to-digital converters using a stepped three tone scan," *IEEE Trans. Instrum. Meas.*, vol. 57, no. 4, pp. 666–671, Apr. 2008.
[14] C. Vogel and G. Kubin, "Analysis and compensation of nonlinearity mismatches in time-interleaved analog-to-digital converter arrays," in *Proc. IEEE Int. Symp. Circuits Syst.*, 2004, pp. 593–596.
[15] G. Kelso, D. Hummels, and F. Irons, "Fast compensation of analog to digital converters," in *Proc. 16th IEEE Conf. Instrum. Meas. Technol.*, 1999, pp. 1295–1298.
[16] D. Hummels, J. McDonald, and F. Irons, "Distortion compensation for time-interleaved analog-to-digital converters," in *Proc. IEEE Conf. Instrum. Meas. Technol.*, 1996, pp. 728–731.
[17] D. R. Morgan *et al.*, "A generalized memory polynomial model for digital predistortion of RF power amplifiers," *IEEE Trans. Signal Process.*, vol. 54, no. 10, pp. 3852–3860, Oct. 2006.
[18] L. Ding *et al.*, "A robust digital baseband predistorter constructed using memory polynomials," *IEEE Trans. Commun.*, vol. 52, no. 1, pp. 159–165, Jan. 2004.
[19] J. P. Deyst, J. J. Vytal, P. R. Blasche, and W. M. Siebert, "Wideband distortion compensation for bipolar flash analog-to-digital converters," in *Proc. Instrum. Meas. Technol. Conf., IMTC'92*, 1992, pp. 290–294.
[20] J. Tsimbinos and K. V. Lever, "Error table and Volterra compensation compensation of A/D converter nonlinearities—A comparison," in *Proc. 4rth Int. Symp. Signal Process. Its Applicat., ISSPA 96*, 1996, vol. 2, pp. 861–864.

[21] 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.
[22] P. Lowenborg, H. Johansson, and L. Wanhammar, "A class of two-channel approximately perfect reconstruction hybrid analog/digital filter banks," in *Proc. 2000 IEEE Int. Symp. Circuits Syst.*, 2000, vol. 1, pp. 579–582.
[23] 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.
[24] 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: Express Briefs*, vol. 53, no. 10, pp. 1133–1137, Oct. 2006.
[25] V. Mathews, "Adaptive Volterra filters using orthogonal structures," *IEEE Signal Process. Lett.*, vol. 3, no. 12, pp. 307–309, Dec. 1996.
[26] G. M. Raz, "Preliminary report of nonlinear equalization of radar receivers," Lincoln Lab., Mass. Inst. Technol., Project Rep. NLEQ-1, Oct. 2002.
[27] C. Chan and G. Raz, "Nonlinear equalization preliminary design review: PHoCS and HCS signal processing," Lincoln Lab., Mass. Inst. of Technol., Project Rep. NLEQ-2,, Apr. 2005.
[28] G. Palm and T. Poggio, "The Volterra representation and Wiener expansion: Validity and pitfalls," *SIAM J. Appl. Math.*, pp. 195–216, 1977.
[29] E. Candes and T. Tao, "The Dantzig selector: Statistical estimation when p is much larger than n," *Ann. Statist.*, vol. 35, no. 6, pp. 2313–2351, 2007.
[30] W. Johnson and J. Lindenstrauss, "Extensions of Lipschitz mappings into a Hilbert space," in *Proc. Conf. Modern Anal. Probab.*, 1977, pp. 186–206.
[31] D. Achlioptas, "Database friendly random projections," in *Proc. ACM Symp. Principles of Database Syst.*, 2001, pp. 274–281.
[32] F. Alizadeh and D. Goldfarb, "Second order cone programming," RUTCOR Res. Rep., Nov. 2001.

Joel Goodman, Photograph and biography not available at the time of publication.

Benjamin Miller, Photograph and biography not available at the time of publication.

Matthew Herman, Photograph and biography not available at the time of publication.

Gil Raz, Photograph and biography not available at the time of publication.

Jeffrey Jackson, Photograph and biography not available at the time of publication.