A NEW CLASS OF FLANN FILTERS WITH APPLICATION TO NONLINEAR ACTIVE NOISE CONTROL

Alberto Carini

DiSBeF, University of Urbino
Piazza della Repubblica 13, 61029, Urbino, Italy
email: alberto.carini@uniurb.it

Giovanni L. Sicuranza

DI3, University of Trieste
Via A. Valerio 10, 34127, Trieste, Italy
email: sicuranza@univ.trieste.it

ABSTRACT

FLANN and generalized FLANN filters exploiting trigonometric functions are often used in active noise control. However, they cannot approximate arbitrarily well every causal, time-invariant, finite-memory, nonlinear system, i.e., they are not universal approximators as the Volterra filters. In this paper, we propose a novel class of FLANN filters, called Complete FLANN filters, which satisfy the Stone-Weierstrass theorem, and thus can arbitrarily well approximate any nonlinear, time-invariant, finite-memory, continuous system. CFLANN filters are members of the class of nonlinear filters characterized by the property that their output depends linearly on the filter coefficients. As a consequence, they can be efficiently implemented in the form of a filter bank and adapted using algorithms simply derived from those applied to linear filters. In the paper, we apply a nonlinearly Filtered-X NLMS algorithm for CFLANN filters and describe some useful applications in the area of nonlinear active noise control.

***Index Terms*—** FLANN filters, Complete FLANN filters, Nonlinear filters, Active noise control

1. INTRODUCTION

The Functional Link Artificial Neural Network (FLANN) has been first studied in the literature on neural networks as an effective alternative to the well-known Multilayer Artificial Neural Network (MLANN). In fact, it is able to offer a simple structure for hardware implementation, together with a reduced computational complexity [1]. While the MLANN exploits linear links and hidden layers, the FLANN is a net without hidden layers resorting to point-wise functional expansions of the current input sample that are then linearly combined to generate the output. The nonlinear expansions most frequently used are those based on orthogonal trigonometric polynomials, involving cosines and sines as basis functions [2, 3]. These filters have been used, in particular, to solve problems in nonlinear active noise control [3]. However, as pointed out in [4], the performance of a FLANN filter can be negatively affected since it does not use cross-terms, i.e., products of samples with different time shifts. To overcome this difficulty, the structure of the conventional FLANN filter has been modified in [5] to include appropriate cross-terms, and thus a generalized FLANN (GFLANN) filter has been proposed. It is worth noting that, from a signal processing point of view, the FLANN and GFLANN filters are both members of the class of causal, shift-invariant, finite-memory, nonlinear filters characterized by the property that their output depends linearly on the filter coefficients [6]. In addition to FLANN and GFLANN filters, this class includes most of the popular nonlinear filters commonly used to model nonlinear systems, as truncated Volterra [7] and generalized memory polynomial filters [8], to mention only a few. However, the FLANN and GFLANN filters cannot be considered universal approximators as the Volterra filters, since they are not able to approximate arbitrarily well every causal, time-invariant, finite-memory, nonlinear system. Indeed, FLANN and GFLANN filters do not satisfy the requirements of the Stone-Weierstrass theorem [9], which provides conditions for the approximation of any real continuous function on a compact set with arbitrary accuracy. In fact, since the product of two FLANN or GFLANN functions does not belong to the set of FLANN or GFLANN functions, respectively, they do not constitute an *algebra*. However, it is possible to "complete" the set of trigonometric FLANN functions so that they satisfy to the requests of the Stone-Weierstrass theorem. In this paper, we describe how this extension can be obtained and we define a novel class of filters, called "Complete FLANN" (CFLANN) filters, that have the remarkable property of being able to approximate any causal, time-invariant, finite-memory, continuous nonlinear system with arbitrary accuracy.

It is worth noting that the CFLANN filter is another member of the class of the linear-in-the-coefficients nonlinear filters mentioned above. As a consequence, it shares with the other members of the class the main characteristics described in [6], that are the efficient implementation in the form of a filter bank and the adaptive algorithms simply derived from those applied to linear filters. In particular, in this paper we apply a nonlinearly Filtered-X NLMS algorithm for CFLANN filters and describe some useful applications in the area of nonlinear ANC.

The paper is organized as follows: The CFLANN functions, satisfying the Stone-Weierstrass theorem, are introduced in Section 2. Their expressions for order $P = 1, 2, 3$ are given in the form of pseudocodes, used for their generation, that avoid any repetition or cancellation between terms. A nonlinearly Filtered-X adaptive algorithm for CFLANN filters is presented in Section 3. A couple of applications of CFLANN filters to nonlinear ANC are shown in Section 4, together with some comments on the obtained results. Concluding remarks follow in Section 5.

2. CFLANN FUNCTIONS AND FILTERS

In this paper we consider the problem of the identification or approximation of the input-output relationship of a time-invariant, finite-memory, continuous nonlinear system. Assuming for simplicity the system to be causal, its input-output relationship can be expressed in the following form,

$$y(n) = f[x(n), x(n-1), \ldots, x(n-N+1)] \qquad (1)$$

where $x(n)$ is the input signal, with

$$x(n) \in \mathbb{R}_1 = \{x \in \mathbb{R}, \text{ with } |x| \leq 1\},$$

$y(n) \in \mathbb{R}$ is the output signal, N is the system memory-length, $f[\ldots]$ is a continuous function from $\mathbb{R}_1^N$ to $\mathbb{R}$.

FLANN filters of order P approximate (1) with the following input-output relationship

$$\begin{aligned}
&y(n) = a_0 x(n) + \ldots + a_{N-1} x(n-N+1) + \\
&c_{1,0} \cos[\pi x(n)] + \ldots + c_{1,N-1} \cos[\pi x(n-N+1)] + \\
&s_{1,0} \sin[\pi x(n)] + \ldots + s_{1,N-1} \sin[\pi x(n-N+1)] + \ldots + \\
&c_{P,0} \cos[P\pi x(n)] + \ldots + c_{P,N-1} \cos[P\pi x(n-N+1)] + \\
&s_{P,0} \sin[P\pi x(n)] + \ldots + s_{P,N-1} \sin[P\pi x(n-N+1)].
\end{aligned} \qquad (2)$$

GFLANN filters add to the FLANN filters the cross products

$$x(n-1)\cos[\pi x(n)], \ldots, x(n-N+1)\cos[\pi x(n)],$$
$$x(n-1)\sin[\pi x(n)], \ldots, x(n-N+1)\sin[\pi x(n)],$$

and suitably selected delayed versions as shown in [5]. FLANN and GFLANN filters cannot approximate every nonlinear system in (1) with arbitrary accuracy. For example, FLANN and GFLANN filters cannot well approximate the system with input-output relationship $y(n) = x^3(n)x^3(n-1)$ because none of their basis functions has this cross-product in its Volterra series expansion.

In this section we introduce a novel class of filters, based on sine and cosine functions, that are able to arbitrarily well approximate any time-invariant, finite-memory, continuous nonlinear system (1). The demonstration is based on the well-known Stone-Weierstrass theorem:

Stone-Weierstrass Theorem : "Let $\mathcal{A}$ be an algebra of real continuous functions on a compact set K. If $\mathcal{A}$ separates points on K and if $\mathcal{A}$ vanishes at no point of K, then the uniform closure $\mathcal{B}$ of $\mathcal{A}$ consists of all real continuous functions on K" [9].

According to the Stone-Weierstrass Theorem every algebra of real continuous functions on the compact $\mathbb{R}_1^N$ which separates points and vanishes at no point is able to arbitrarily well approximate the continuous function $f[\ldots]$ in (1). A family $\mathcal{A}$ of real functions is said to be an algebra if $\mathcal{A}$ is closed under addition, multiplication, and scalar multiplication, i.e., if (i) $f + g \in \mathcal{A}$, (ii) $f \cdot g \in \mathcal{A}$, and (iii) $cf \in \mathcal{A}$, for all $f \in \mathcal{A}$, $g \in \mathcal{A}$ and for all real constants c. The basis functions of the FLANN and GFLANN filters are not an algebra because they are not closed under multiplication: the product of sine and cosine functions (e.g., $\sin[\pi x(n)] \cdot \cos[\pi x(n-1)]$) cannot be expressed as a linear combination of the basis functions. However, by using the prosthaphaeresis formulas, such products can be conveniently expressed as cosines and sines of sums and differences of the input samples,

$$\sin[\pi x(n)] \cdot \cos[\pi x(n-1)] = \frac{1}{2}\sin\{\pi[x(n) - x(n-1)]\} + \frac{1}{2}\sin\{\pi[x(n) + x(n-1)]\}.$$

Thus, it is easy to find a set of trigonometric functions that form an algebra. Such a set is composed by all cosine and sine functions of any order P having the form

$$\begin{aligned}
&\cos\{\pi[x(n-i_1+1) \pm \cdots \pm x(n-i_P+1)]\}, \\
&\sin\{\pi[x(n-i_1+1) \pm \cdots \pm x(n-i_P+1)]\},
\end{aligned} \qquad (3)$$

with $i_1 = 1, \ldots, N, i_2 = 1, \ldots N, \ldots, i_P = 1, \ldots, N$ and their linear combinations. For $P = 0$, the constant function equal to 1 is considered. The set of basis functions in (3) is redundant because it includes repeated terms and terms which may cancel one to each other when the minus sign is used. Nevertheless, the resulting set of functions on the compact $\mathbb{R}_1$ is closed under addition, multiplication (for the prosthaphaeresis formulas), and scalar multiplication, it separates points[1], and it vanishes at no point (since the constant function is also considered), i.e., it satisfies all the requests of the Stone-Weierstrass theorem. Thus, the input-output relationship of a P-th order CFLANN filter, defined by the linear combination of all the terms in (3) up to the order P, is able to approximate the nonlinear system in (1) with arbitrary accuracy for sufficiently large P. The pseudocodes used for the generation of the CFLANN basis functions of order $P = 1, 2, 3$, that avoid any repetition or cancellation between terms, are given in Table 1, Table 2 and Table 3, respectively.

[1] Two separate points must have at least one different coordinate x_i and $\sin(\pi x_i)$ separates these points.

Table 1. CFLANN basis functions of order $P = 1$.

```
for i = 1 : N_1
    g_1(i) = cos[πx(n − i + 1)]
    g_2(i) = sin[πx(n − i + 1)]
end
```

Table 2. CFLANN basis functions of order $P = 2$.

```
for  i = 1 : N_2
 for  j = i : N_2
   g_3(i, j) = cos{π[x(n − i + 1) + x(n − j + 1)]}
   g_4(i, j) = sin{π[x(n − i + 1) + x(n − j + 1)]}
 end
end
for  i = 1 : N_2 − 1,
 for  j = i + 1 : N_2,
   g_5(i, j) = cos{π[x(n − i + 1) − x(n − j + 1)]}
   g_6(i, j) = sin{π[x(n − i + 1) − x(n − j + 1)]}
 end
end
```

It is easily observed that the number of elements in each of the functions g_1 and g_2 is N_1, in g_3 and g_4 is $N_2(N_2+1)/2$, in g_5 and g_6 is $(N_2-1)N_2/2$, in g_7 and g_8 is $N_3(N_3+1)(N_3+2)/6$, in g_9, g_{10}, g_{11} and g_{12} is $(N_3 - 1)N_3(N_3 + 1)/6$, and in g_{13} and g_{14} is $(N_3 - 2)(N_3 - 1)N_3/6$.

3. NONLINEARLY FILTERED-X NLMS ALGORITHM

CFLANN functions and related filters belong to the class of nonlinear filters with finite memory whose output depends linearly on the filter coefficients. This class of filters has been studied in [6], and includes Hammerstein and Volterra filters, memory polynomial filters, FLANN and GFLANN filters, etc. The CFLANN filter can be added as a further element to this class, sharing notation, properties and algorithms described in [6]. Thus, its input-output relationship can be expressed as

$$y(n) = \mathbf{h}^T(n)\mathbf{x}_{\mathcal{F}}(n), \tag{4}$$

where $\mathbf{h}(n)$ is the vector containing L filter coefficients and $\mathbf{x}_{\mathcal{F}}(n)$ is a vector whose L entries are formed by the trigonometric combinations, corresponding to the CFLANN functions, of the N most recent samples of the input signal. In adaptive applications using zero-mean signals, a CFLANN filter does not include the constant term which corresponds to a bias that should be avoided, as done also for Volterra filters. Moreover, the CFLANN filter does not include linear terms that are implicitly contained in the higher-order terms. The non-homogeneous CFLANN filter of order $P = 2$ includes also the functions of order $P = 1$ and thus it contains $L = 2N_1 + 2N_2^2$ coefficients. Similarly, the non-homogeneous CFLANN filter of order $P = 3$ includes also

Table 3. CFLANN basis functions of order $P = 3$.

```
for  i = 1 : N_3
 for  j = i : N_3
  for  k = j : N_3
    g_7(i, j, k) = cos{π·
        [x(n − i + 1) + x(n − j + 1) + x(n − k + 1)]}
    g_8(i, j, k) = sin{π·
        [x(n − i + 1) + x(n − j + 1) + x(n − k + 1)]}
  end
 end
end
for  i = 1 : N_3 − 1
 for  j = i : N_3 − 1
  for  k = j + 1 : N_3
    g_9(i, j, k) = cos{π·
        [x(n − i + 1) + x(n − j + 1) − x(n − k + 1)]}
    g_10(i, j, k) = sin{π·
        [x(n − i + 1) + x(n − j + 1) − x(n − k + 1)]}
  end
 end
end
for  i = 1 : N_3 − 1
 for  j = i + 1 : N_3
  for  k = j : N_3
    g_11(i, j, k) = cos{π·
        [x(n − i + 1) − x(n − j + 1) − x(n − k + 1)]}
    g_12(i, j, k) = sin{π·
        [x(n − i + 1) − x(n − j + 1) − x(n − k + 1)]}
  end
 end
end
for  i = 1 : N_3 − 2
 for  j = i + 1 : N_3 − 1
  for  k = j + 1 : N_3
    g_13(i, j, k) = cos{π·
        [x(n − i + 1) − x(n − j + 1) + x(n − k + 1)]}
    g_14(i, j, k) = sin{π·
        [x(n − i + 1) − x(n − j + 1) + x(n − k + 1)]}
  end
 end
end
```

the functions of order $P = 1$ and $P = 2$, respectively, and thus it contains $L = 2N_1 + 2N_2^2 + (2/3)N_3 + (4/3)N_3^3$ coefficients.

Let us now consider the CFLANN filters in the framework of nonlinear active noise control (NANC). An active noise controller is based on the destructive interference in a given location between the primary noise, propagating through a primary path, and a secondary interfering noise, generated by an active controller and propagating through a secondary path. The principle of NANC is illustrated in Fig. 1, where the primary and secondary paths may contain some nonlinear-

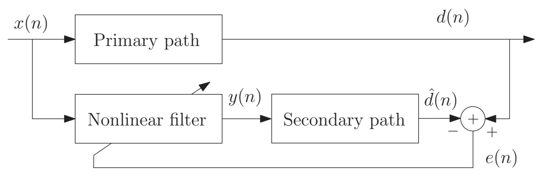

Fig. 1. Principle of nonlinear active noise control.

ities. When the nonlinearities are present in the primary path or in the input signal $x(n)$, the controllers belonging to the class of filters described in [6] can be adapted with Filtered-X algorithms that are similar to those used for linear filters. In contrast, when the nonlinearities affect the secondary path, it is necessary to resort to the concept of *virtual secondary path* $\tilde{s}(n)$, as shown in [4]. With reference to Fig. 1, $\hat{d}(n)$ is the interfering signal at the cancellation point, and $y(n)$ is the signal at the output of the nonlinear controller. The length of the virtual secondary path is denoted as N_s. In practice, in the adaptation rule, the input signal is filtered by a time-varying filter $\tilde{s}(n)$ whose coefficients

$$\left[\frac{\partial \hat{d}(n)}{\partial y(n)} \; \frac{\partial \hat{d}(n)}{\partial y(n-1)} \cdots \frac{\partial \hat{d}(n)}{\partial y(n-N_s+1)}\right], \tag{5}$$

replace the coefficients of the impulse response $s(n)$ of the linear case. Each element of the input signal must be filtered by $\tilde{s}(n)$, and this fact significantly increases the computational complexity with respect to conventional algorithms working in presence of a linear secondary path. The updating rule used in this paper is a normalized version of that derived in [4], i.e.,

$$\mathbf{h}(n+1) = \mathbf{h}(n) + \frac{\mu}{||\mathbf{u}(n)||^2} e(n)\mathbf{u}(n), \tag{6}$$

where μ is the step size, $\mathbf{u}(n) = \tilde{s}(n) * \mathbf{x}_{\mathcal{F}}(n)$, and the symbol $*$ means convolution. The squared Euclidean norm $||\mathbf{u}(n)||^2 = \mathbf{u}^T(n)\mathbf{u}(n)$ is used as a normalization factor. The computational complexity of the adaptive algorithm formulated in this section is on the order of $\mathcal{O}(LN_s + 3L)$ multiplications and additions, where L is the number of filter coefficients.

4. COMPUTER SIMULATIONS

In this section we consider the application of a CFLANN filter to NANC. The adaptive algorithm is the nonlinearly Filtered-X NLMS algorithm of (6).

4.1. Example 1

In this example, we refer to the situation described in [4, Simulation 2] and [5, Example 2]. The primary and secondary paths are described by Volterra models given as

$$d(n) = x(n) + 0.8x(n-1) + 0.3x(n-2) + 0.4x(n-3)$$

$$-0.8x(n)x(n-1) + 0.9x(n)x(n-2)$$
$$+0.7x(n)x(n-3) - 3.9x^2(n-1)x(n-2)$$
$$-2.6x^2(n-1)x(n-3) + 2.1x^2(n-2)x(n-3) \tag{7}$$

and

$$\hat{d}(n) = y(n) + 0.35y(n-1) + 0.09y(n-2)$$
$$-0.5y(n)y(n-1) + 0.4y(n)y(n-2) \tag{8}$$

respectively.

The reference signal is a random noise with a uniform distribution between -0.5 and $+0.5$, and the ensemble learning curves in Fig. 2 are computed over 200 independent runs of the adaptive algorithm. The five curves in Fig. 2 refer to: (a) the FLANN filter of order $P = 3$, with step sizes $\mu_1 = 0.8$ and $\mu_2 = 0.01$ for the linear and nonlinear parts, respectively; (b) the second-order Volterra filter, with step sizes $\mu_1 = 0.05$ and $\mu_2 = 0.01$ for the linear and quadratic parts, respectively; (c) the third-order Volterra filter, with step sizes $\mu_1 = 0.04$, $\mu_2 = 0.05$, and $\mu_3 = 0.1$ for the linear, quadratic, and cubic parts, respectively; (d) the GFLANN filter of order $P = 1$ using cross-terms with $N_d = 9$, as shown in [5, Example 2], with step sizes $\mu_1 = 0.09$, $\mu_2 = 0.05$, and $\mu_3 = 0.2$ for the linear part, the cosine and sine functions, and the cross-terms, respectively; and (e) the complete FLANN filter of order $P = 2$, with step sizes $\mu_1 = 0.16$, and $\mu_2 = 0.06$ for the cosine and sine functions of a single input sample, and the cosine and sine functions of sum/differences of input samples, respectively. The step sizes have been tuned for similar initial convergence speed to provide a fair comparison. The FLANN and Volterra filters have a memory of 10 samples and include a linear term with the same memory length. The CFLANN filter of order $P = 2$ does not include the linear part and has a memory of $N_1 = N_2 = 6$ samples. In this case, while the number of coefficients is 65 for the second-order Volterra filter, 70 for the FLANN filter, 120 for the GFLANN filter, and

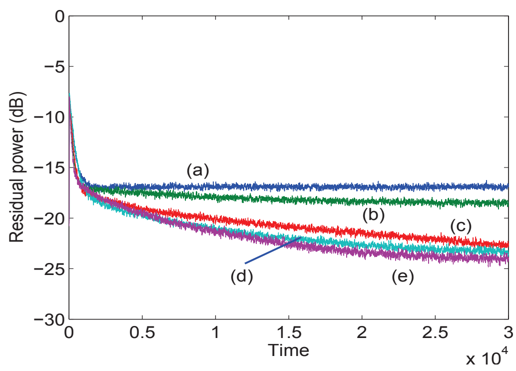

Fig. 2. Learning curves for Example 1: (a) FLANN filter of order $P = 3$, (b) second-order Volterra filter, (c) third-order Volterra filter, (d) GFLANN filter of order $P = 1$, and (e) CFLANN filter of order $P = 2$.

Table 4. Noise attenuation (dB) and number of coefficients for Example 2.

Filter	Noise attenuation	Number of coefficients
FLANN $P = 1$ $N = N_1 = 10$	12.7	30
FLANN $P = 3$ $N = N_1 = 10$	13.7	70
GFLANN $P = 1$ $N = N_1 = 10, N_\mathrm{d} = 3$	16.6	78
GFLANN $P = 3$ $N = N_1 = 10, N_\mathrm{d} = 3$	18.9	214
CFLANN $P = 2$ $N_1 = N_2 = 4$	16.9	40
CFLANN $P = 3$ $N_1 = N_2 = N_3 = 4$	19.4	128

285 for the third-order Volterra filter, the CFLANN filter has only 84 coefficients and clearly offers the best performance.

4.2. Example 2

We refer here to the same primary path of Example 1 and to a secondary path modeled as a Hammerstein filter with a memoryless nonlinearity given by $w(n) = \tanh[y(n)]$, followed by the linear filter

$$\hat{d}(n) = w(n) + 0.2w(n-1) + 0.05w(n-2). \quad (9)$$

This model efficiently describes the nonlinearities affecting the power amplifier and the loudspeaker at the controller. In the experiment, the reference signal is the noise generated by a fan, recorded at a sampling frequency of 44.1 kHz and then decimated by a factor 12, with 16 bits per sample. The length of the used signal is equal to 65 000 samples. Table 4 shows the noise attenuation and the number of coefficients of the controllers. The attenuation is defined as the ratio between the mean noise power values (calculated over the last 10 000 samples) at the cancellation point without and with the active noise control. As reported in [5, Example 2], the memories of the linear and nonlinear parts of the FLANN and GFLANN filters are $N = 10$ and $N_1 = 10$ samples, respectively. The GFLANN filters exploit 3 diagonals, as explained in [5]. The CFLANN filters of order $P = 2$ and $P = 3$ do not include a linear part and have a memory of $N_1 = N_2 = 4$ samples and $N_1 = N_2 = N_3 = 4$ samples, respectively. It is seen from Table 4 that the CFLANN filters improve the performance of the conventional and generalized FLANN filters with a reduced number of coefficients.

5. CONCLUSIONS

In this paper, a set of trigonometric functions, the so-called CFLANN functions, is presented. These functions permit to approximate any real continuous function on a compact set with arbitrary accuracy according to the Stone-Weierstrass theorem. Using the CFLANN functions, it is possible to define a new member of the class of linear-in-the-coefficients nonlinear filters, that is the CFLANN filter. Therefore, CFLANN filters can be efficiently implemented by means of filter banks and adapted using algorithms simply derived from those applied to linear filters. In this paper we show that the CFLANN filters, adapted with a nonlinearly Filtered-X NLMS algorithm, can be profitably employed to solve some nonlinear ANC problems. The details of the derivations and other useful considerations and comparisons, not included here due to space limitation, will be presented in a paper under preparation, together with the analysis of the properties of the CFLANN functions.

6. REFERENCES

[1] Y. H. Pao, *Adaptive Pattern Recognition and Neural Networks*. Addison-Wesley, Reading, Mass., 1989.

[2] J. C. Patra and R. N. Pal, "A functional link artificial neural network for adaptive channel equalization," *Signal Processing*, vol. 43, pp. 181-195, May 1995.

[3] D. P. Das and G. Panda, "Active mitigation of nonlinear noise processes using a novel filtered-s LMS algorithm," *IEEE Trans. Speech Audio Processing*, vol. 12, pp. 313-322, May 2004.

[4] D. Zhou and V. DeBrunner, "Efficient adaptive nonlinear filters for nonlinear active noise control," *IEEE Trans. Circuits Syst. I*, vol. 54, pp. 669-681, March 2007.

[5] G. L. Sicuranza and A. Carini, "A generalized FLANN filter for nonlinear active noise control," *IEEE Trans. Speech, Audio, and Language Processing*, vol. 19 , pp. 2412-2417 , Nov. 2011.

[6] G. L. Sicuranza and A. Carini, "On a class of nonlinear filters", in *Festschrift in Honor of Jaakko Astola on the Occasion of his 60th Birthday*, I. Tabus, K. Egiazarian and M. Gabbouj, Eds, TICSP series No. 47, Tampere 2009, pp. 115-144.

[7] V. J. Mathews and G. L. Sicuranza, *Polynomial Signal Processing*, New York: Wiley, 2000.

[8] D. R. Morgan, Z. Ma, J. Kim, M. G. Zierdt and J. Pastalan, "A generalized memory polynomial model for digital predistortion of RF power amplifiers," *IEEE Trans. Signal Processing*, vol. 54, pp. 3852-3860, Oct. 2006.

[9] W. Rudin, *Principles of Mathematical Analysis.* New York: McGraw-Hill, 1976.