Extended Principal Domain for Discrete Frequency-Domain Quadratic Volterra Models

이산 주파수 영역 2차 Volterra 모델의 확장된 주영역

Sungbin Im*, Won Chul Lee*, and Myung Jin Bae*

임 성 빈*, 이 원 철*, 배 명 진*

Abstract

In this paper we point out that if the classical principal domain for bispectra is utilized to determine a second-order Volterra model's output, such an output will be incomplete. This deficiency is associated with the periodic nature of the DFT. For this reason, the objective of this paper is to present an "extended" principal domain for Volterra kernels which leads to an improved estimate of the nonlinear system's response. In order to define the extended principal domain, we derive a new discrete frequency-domain Volterra model from a discrete time-domain Volterra model utilizing 2-dimensional DFT and the relationship between the quadratic component of the Volterra model and a square filter. The effect of the extended domain on the model output is interpreted in terms of the periodicity of DFT. Through computer simulations, we demonstrate the effects of the extended principal domain on the Volterra modeling. The simulation results indicate that the extended principal domain plays an important role in computing Volterra model outputs and estimating Volterra model coefficients.

요 약

본 논문에서는 bispectra를 위한 고전적 주영역 (classical principal domain)이 2차 Volterra 모델의 출력을 결정짓는데 사용되면 그 출력은 완전하지 못하게 될 것임을 지적한다. 이러한 불완전함은 DFT의 주기적 특성과 관련이 있다. 이런 이유로, 본 논문의 목적은 비선형 시스템의 응답의 추정을 향상시키는 Volterra 커널을 위한 확장된 주영역 (extended principal domain)을 제안하는데 있다. 확장된 주영역을 정의 내리기 위하여, 2차원 DFT와 Volterra 모델의 2계 요소와 정사각형 필터와의 관계를 사용하여 이산 시간 영역 Volterra 모델에서 새로운 이산 주파수 영역의 Volterra 모델을 유도하였다. 확장된 영역이 모델의 출력에 미치는 영향을 DFT의 주기성 측면에서 해석을 하였다. 컴퓨터 모의 실험을 통하여, Volterra 모델링에서 확장된 주영역의 영향을 살펴보았다. 모의 실험 결과에 의하면, Volterra 모델의 출력을 계산하는 과정과 Volterra 모델의 계수를 추정하는데 있어서 매우 중요한 역할을 함을 알 수 있었다.

I. Introduction

In recent years, as interest in nonlinear systems, particularly in nonlinear digital modeling steadily increased, application of Volterra models has played an ever increasing role in nonlinear system analysis, identification, and prediction, such as communication channel equalization [1], characterization of digital magnetic recording channels [2] and high-speed A/D converters used in digital radio systems [3], and speech prediction [4].

Approaches based on Volterra models have a firm mathematical foundation [5] and can describe a broad class of nonlinear phenomena. Furthermore, since the output of a Volterra model depends linearly on the linear, quadratic, and cubic model coefficients (but nonlinearly on the input) [6], many concepts originally developed for linear models can be extended to Volterra models. Moreover, through

*Department of Information and Telecommunication Engineering Soongsil Univesity

접수일자: 1995년 10월 25일

the use of Volterra models to model nonlinear systems, one often gains new insight into the physical mechanisms underlying such nonlinear systems [7].

Higher-order spectra have previously been shown to be very powerful in determining frequency-domain Volterra kernels [8] used to model a wide variety of nonlinear systems and nonlinear phenomena. The principal domains for auto and cross-bispectra have been described in the literature [6, 8, 9]. In this paper we point out that if the classical principal domain for bispectra is utilized to determine a second-order Volterra model's output, such an output will be incomplete. This deficiency is associated with the periodic nature of the DFT. For this reason, the objective of this paper is to present an "extended" principal domain for Volterra kernels which leads to an improved estimate of the nonlinear system's response.

In earlier literature [6, 8], the discrete frequency-domain second-order Volterra kernel has been considered in the principal domain *AOPCQR* in Figure 1. However, when dealing with discrete time and frequency-domain Volterra models, we often observe a discrepancy between the time and frequency-domain

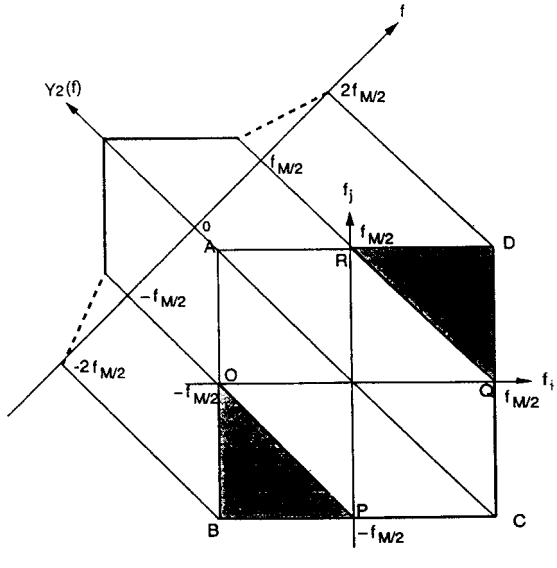


Fig 1. The area *AOPCQR* is the principal domain and the shaded areas *RQD* and *OBP* represent the extended principal domain for a second-order Volterra model. $Y_2(f)$ denotes the output of the second-order component of the Volterra model. The index set $\{0, 1,...,M-1\}$ is translated to a discrete frequency set $\{-f_{M/2},...,0,...,f_{M/2}\}$ by the M-periodicity of the DFT and sampling frequency.

Volterra model outputs. Our study presented in the following sections shows that the frequency-domain second-order Volterra kernel defined in the shaded regions *RQD* and *OBP* in Figure 1 also contributes to the quadratic output of the Volterra model and, therefore, should be included. For this reason we define an extended principal domain for the second-order Volterra kernel consisting of *AOPCQR*, *RQD*, and *OBP*.

In the following section, we present discrete time and frequency-domain second-order Volterra models, in which the discrete frequency-domain second-order Volterra model is different from those in earlier literature [6, 8]. In Section 3, we present the derivation and explanation of the extended principal domain. In Section 4, for a numerical example, we demonstrate the effects of the classical and extended principal domains on computing Volterra model outputs and estimating Volterra model coefficients.

Ⅱ. Volterra Models

If we assume that the nonlinear system to be represented by a Volterra model is stable and of finite memory and has nonlinearities up to second order, the second-order Volterra model can approximate the output of the nonlinear system by its sampled data form [10]. The model output can be represented as follows:

$$y(n)=\sum_{i=0}^{N-1} h_1(i)x(n-i)+\sum_{i=0}^{N-1}\sum_{j=0}^{N-1} h_2(i,j)x(n-i)x(n-j) \qquad (1)$$

where $x(n)$ and $y(n)$ denote the input sequence to the system and the system output sequence predicted by the second-order Volterra model, respectively. $h_1(i)$ and $h_2(i, j)$ represent the linear and quadratic Volterra model coefficients, respectively.

Since the Volterra model can be interpreted as a generalized Taylor series representation of a function with memory, the Volterra model can be interpreted as an extension of a linear model in that a quadratic model, a cubic model, and so on are appended in parallel to a linear model. For this reason, the Volterra models can describe a broad class of gently nonlinear systems. For practical reasons, primarily, this paper deals with a second-

order model but the same procedure can be extended, in principal, to higher orders.

In earlier literature [6, 8], the discrete frequency-domain version of (1) is considered as follows:

$$Y(m) = H_1(m)X(m) + \sum_{p=0}^{M-1}\sum_{q=0}^{M-1} H_2(p, q)X(p)X(q)\delta(m-p-q) \quad (2)$$

where

$$\delta(\eta) = \begin{cases} 1, \eta = 0 \\ 0, \eta \neq 0 \end{cases} \quad (3)$$

In (2), $Y(m)$, $X(m)$, $H_1(m)$, and $H_2(p, q)$ are DFT's of $y(n)$, $x(n)$, $h_1(i)$ abd $h_2(i, j)$ respectively. The delta function is a representation of the "frequency selection rule" [8], which determines discrete frequency component pairs (p, q) satisfying $m = p - q$. The last term of (2) is called the quadratic component of the Volterra model output. This term sums the contributions of all pairs of quadratically interacting spectral components that satisfy the selection rule $m = p + q$.

In this paper, however, we introduce a new second-order Volterra model based on an M-point DFT:

$$Y(m) = H_1(m)X(m) + \sum_{p=0}^{M-1}\sum_{q=0}^{M-1} H_2(p, q)X(p)X(q)\delta_M(m-p-q) \quad (4)$$

where

$$\delta_M(\eta) := \begin{cases} 1, (\eta \bmod \text{ulo } M) = 0 \\ 0, (\eta \bmod \text{ulo } M) \neq 0 \end{cases} \quad (5)$$

In (4) $Y(m)$, $X(m)$, $H_1(m)$, and $H_2(p, q)$ are DFT's of $y(n)$, $x(n)$, $h_1(i)$ and $h_2(i, j)$, respectively. Note that the frequency-domain Volterra model (4) is different from (2) in that $\delta_M(\cdot)$ is defined by the modulo function [11]. This is explained further in the next section.

Ⅲ. Extended Principal Domain

In this section, we investigate how the one-dimensional output of the Volterra model is relate to the two-dimensional frequency components generated by the product of the one-dimensional input sequence. This reveals that in the discrete frequency domain, the Volterra kernel defined outside the classical principal domain also contributes to the quadratic output of the model.

For this investigation, first, we need to look into the relationship between a quadratic model output of the second-order Volterra model (1) and a two-dimensional square filter output, when the two-dimensional filter coefficients are equal to the quadratic kernel $h_2(i, j)$. The two-dimensional square filter output $q(l, m)$ is represented by the following two-dimensional convolution.

$$q(l, m) = \sum_{i=0}^{N-1}\sum_{j=0}^{N-1} h_2(i, j)x_2(l-i, m-j) \quad (6)$$

where $x_2(\cdot, \cdot)$ represents a two-dimensional input sequence. According to (1), the quadratic model output $y_2(n)$ is given by

$$y_2(n) = \sum_{i=0}^{N-1}\sum_{j=0}^{N-1} h_2(i, j)x(n-i)x(n-j) \quad (7)$$

Let us assume that the two-dimensional input sequence $x_2(i, j)$ in (6) is equal to $x(i)x(j)$, which implies that $x_2(i, j)$ is separable [12]. Then, (6) becomes

$$q(l, m) = \sum_{i=0}^{N-1}\sum_{j=0}^{N-1} h_2(i, j)x(l-i)x(m-j) \quad (8)$$

By comparing (7) and (8), we arrive at the following expression,

$$y_2(n) = q(n, n) \quad (9)$$

That is, the output of the quadratic component of the Volterra model corresponds to the values on the diagonal of the two-dimensional filter outputs. Based on this observation and the separability of $x_2(i, j)$, we consider the process of computing $q(l, m)$ using the two-dimensional IDFT of $H_2(p, q)X(p)X(q)$,

$$q(l, m) = \sum_{p=0}^{M-1}\sum_{q=0}^{M-1} H_2(p, q)X(p)X(q)W_M^{pl}W_M^{qm} \quad (10)$$

where $W_M = e^{2\pi\sqrt{-1}/M}$. In the Volterra model, the output is one-dimensional, thus we are interested in only the diagonal terms $q(n, n)$ rather than all the two-dimensional outputs. For $l = n$ and $m = n$, (10) becomes

$$q(n, n) = \sum_{p=0}^{M-1}\sum_{q=0}^{M-1} H_2(p, q)X(p)X(q)W_M^{(p+q)n} \quad (11)$$

By substituting $x = p + q$ in (11) and utilizing $y_2(n) = y(n, n)$, (11) becomes

$$y_2(n) = \sum_{x=0}^{2M-2} W_M^{xn} \{ \sum_{(p,q)\in D(x)} H_2(p, q) X(p) X(q) \} \quad (12)$$

where $D(x) = \{(p, q) \mid p + q = x,\ p, q = 0, 1, \ldots, M-1\}$. Because of the periodicity of W_M, that is, $W_M^{(M+m)n} = W_M^{mn}$, we can rewrite (12) as follows:

$$y_2(n) = \sum_{x=0}^{M-1} W_M^{xn} \{ \sum_{(p,q)\in D_1(x)} H_2(p, q) X(p) X(q) + \sum_{(p,q)\in D_2(x)} H_2(p, q) X(p) X(q) \} \quad (13)$$

where

$$D_1(x) = \{(p, q) \mid p + q = x,\ p, q = 0, 1, \ldots, M-1\} \quad (14)$$

and

$$D_2(x) = \{(p, q) \mid p + q = M + x,\ p, q = 0, 1, \ldots, M-1\} \quad (15)$$

(13) shows that the quadratic output in the time domain is a one-dimensional IDFT of the summation of the product of the input frequency components weighted by the Volterra model coefficients.

Let us define the frequency-domain quadratic output $Y_2(m)$ from (13) as follows:

$$Y_2(m) = \sum_{(p,q)\in D_1(m)} H_2(p, q) X(p) X(q) + \sum_{(p,q)\in D_2(m)} H_2(p, q) X(p) X(q) \quad (16)$$

We use the delta function defined in (5) to simplify (16) as follows:

$$Y_2(m) = \sum_{p=0}^{M-1} \sum_{q=0}^{M-1} H_2(p, q) X(p) X(q) \delta_M(m-p-q) \quad (17)$$

Utilizing (17), we have the discrete frequency-domain second-order Volterra model (4) associated with an M-point DFT.

We need to carefully consider the meaning of the last term in (16). This term has not been previously considered under the assumptions that the quadratic transfer function is confined within the principal domains [6, 8] and/or that the input $X(p)$ is band-limited. For the purpose of a clear explanation, assume that M is an even number and let us consider the quadratic case where the quadratic transfer function $H_2(p, q)$ has nonzero values over the region $D_2(m)$. In Figure 1, the index set $\{0, 1, \ldots, M-1\}$ is translated to a discrete frequency set $\{-f_{M/2}, \ldots, 0, \ldots, f_{M/2}\}$ by the M-periodicity of the DFT and sampling frequency. The shaded areas *RQD* and *OBP* in Figure 1 represent the region $D_2(m)$. When the input frequency components are not zero in these areas, the last term in (16) has a nonzero value. As shown in Figure 2, for the positive output frequency components, due to the M-periodicity of the DFT, the area *OBP* gets reflected into the principal domain *AOPCQR* as the area denoted by *O'B'P'*. This phenomenon, where in effect the last term in (16) takes on the identity of frequency components in the classical principal domain, is similar to the aliasing effects in the linear filtering. In order to accurately compute the Volterra model output, these terms located outside the classical principal domains should be taken into account.

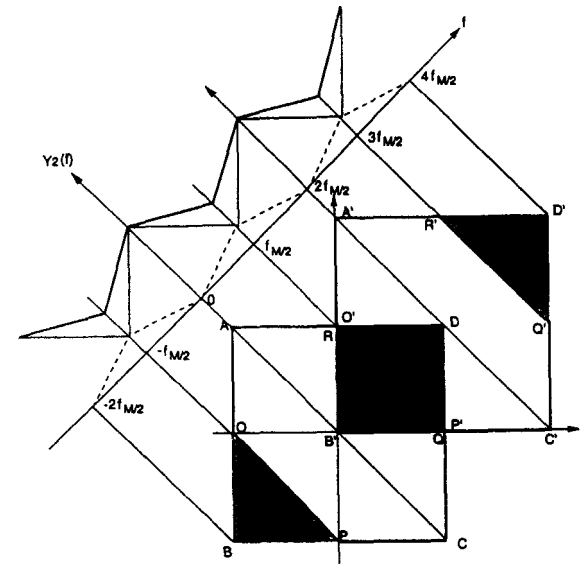


Fig 2. Effects of Periodicity of DFT: For the positive output frequency components, due to the M-periodicity of the DFT, the area *OBP* gets reflected into the principal domain *AOPCQR* as the area *O'B'P'*.

Ⅳ. Computer Simulation

In this section, through computer simulations, we investigate the validity of the extended principal domain in terms of computing Volterra model outputs and estimating Volterra model coefficients. In the case of computing Volterra model outputs, for a known time-domain Volterra system, frequency-domain Volterra model outputs are computed over

the extended and classical principal domains, respectively. They are inverse transformed into the time domain, and, then compared to the system outputs generated from the known time-domain Volterra system.

Next, we consider the identification of discrete frequency-domain second-order Volterra models associated with the extended principal domain. For a known Volterra system, we apply the identification methods with the classical and extended principal domains and compare the coefficient estimates to the known true coefficients.

For the two cases mentioned in the previous paragraphs, we choose the following second-order Volterra system [6]

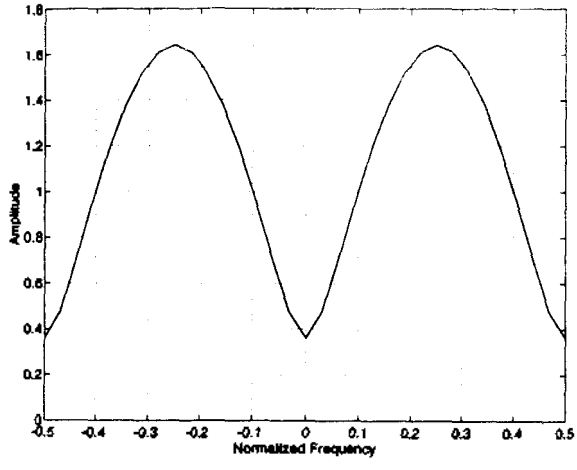
$$y(n) = -0.64x(n) + x(n-2) + 0.9x^2(n) + x^2(n-1) \quad (18)$$

where the first two terms represent the linear response and the remaining terms denote the quadratic response of the system. The transfer functions are given by

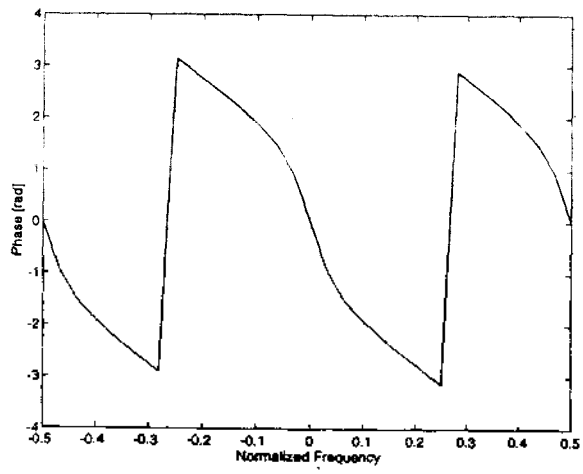
$$H_1(m) = -0.64 + \exp\{-j4\pi m\} \quad (19)$$

$$H_2(p, q) = 0.9 + \exp\{-j2\pi(p+q)\} \quad (20)$$

The linear transfer function $H_1(m)$ is shown in Figure 3 and Figures 4 and 5 show the three-dimensional plots and the contour plots of the magnitude and phase of the quadratic transfer function $H_2(p, q)$, respectively. As indicated in Figure 4 (b), the quadratic transfer function $H_2(p, q)$ has nonzero values in the region denoted by RQD and OBP. For this reason, the Volterra system (18) is adequate to illus-

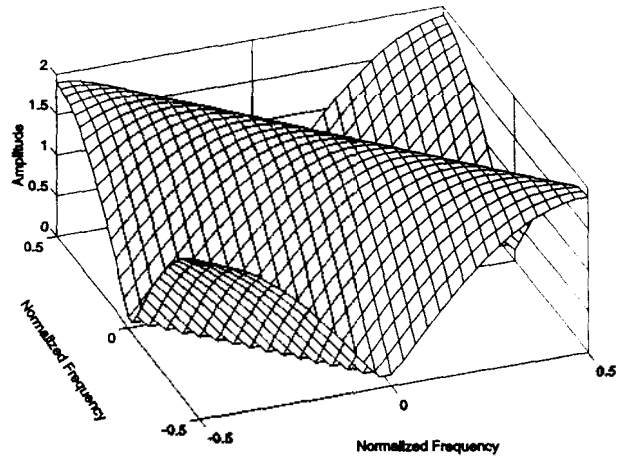


(a)

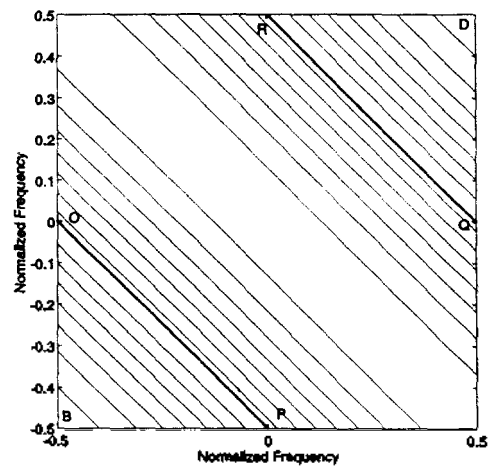


(b)

Fig 3. Linear Transfer Function of (19) : (a) Amplitude, (b) Phase.

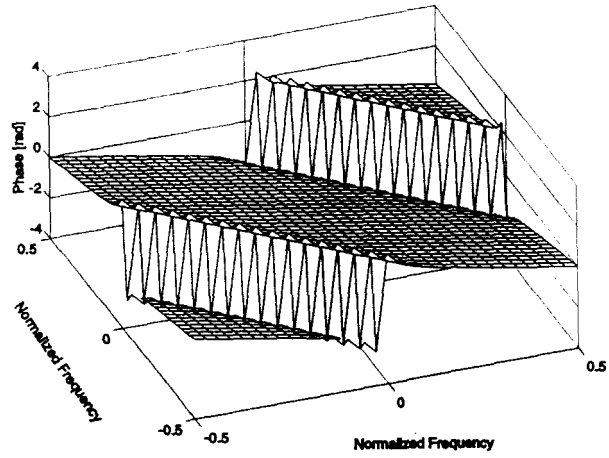


(a)

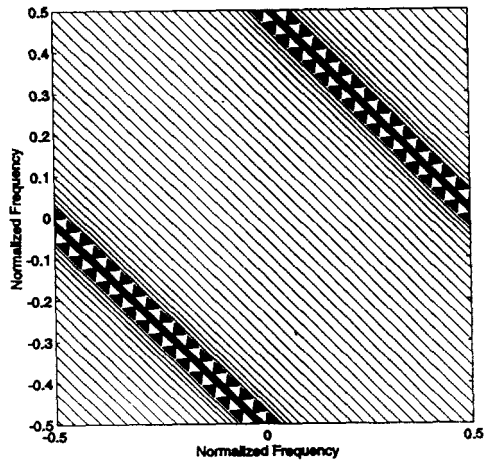


(b)

Fig 4. Amplitude of the Quadratic Transfer Function of (20) : (a)Three-Dimensional Plot, (b) Contour Plot. Note that the level step size is 0.25.



(a)



(b)

Fig 5. Phase of the Quadratic Transfer Function of (20): (a) Three-Dimensional Plot, (b) Contour Plot. The unit is radian. Note that the level step size is 0.1.

trate the effects of the terms outside the classical principal domain on the Volterra model output.

VI.A *Computing Volterra Model Outputs*

In order to demonstrate the effects of the extended and classical principal domains on computing Volterra model outputs, the input sequence $x(i)$ to the Volterra system (18) is generated using MATLAB RANDN and plotted in Figure 6. The length of the sequence is 64. Figure 7 displays the power spectrum of the input sequence $x(i)$. As mentioned in the previous section, the frequency components of the input sequence associated with the regions *RQD* and *OBP* have nonzero values. Figure 8 shows the Volterra model output sequence $y(n)$, which is computed using the time-domain Volterra model (18).

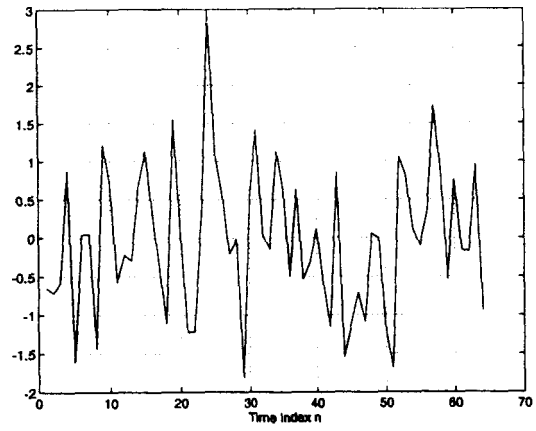


Fig 6. Volterra System Input $x(n)$ in (18).

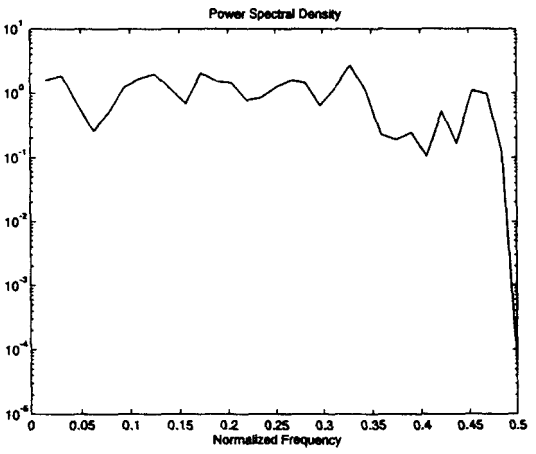


Fig 7 Power Spectrum of Volterra System Input $x(n)$ in (18).

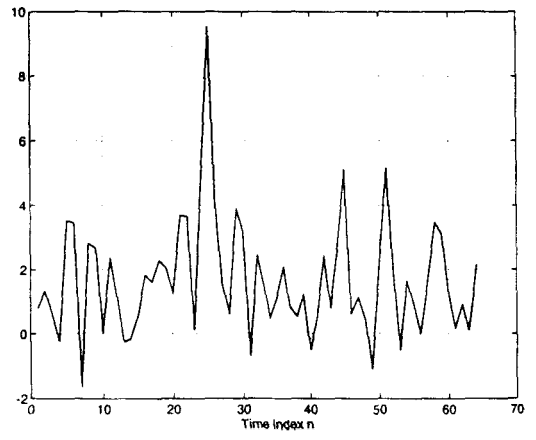


Fig 8. Volterra System Output $y(t)$ in (18).

In order to compute the Volterra model output in the frequency domain, the input sequence is sectioned into 4 segments of 32 samples each with 50% overlap. The reason for overlapping each sec-

tion is to avoid the effect of circular convolution [12]. Each segment is transformed via FFT. First, for each input section, we compute the frequency domain output section using (2), which is based on the classical principal domain. Then, the frequency-domain output section is inverse transformed to obtain the time-domain one. For each time-domain output section, we collect the last correct 16 samples that correspond to linear convolution results. Those samples are connected to generate the time-domain Volterra model outputs, which are plotted in Figure 9. Second, in order to compute the Volterra model outputs using (4), we apply the same procedure as mentioned previously. Figure 10 shows that result. Figure 11 (a) displays the absolute values of error sequence between the output sequence using the classical principal domain and the true output sequence plotted in Figure 8, while Figure 11 (b) shows the absolute values of error sequence between the output sequence using the extended principal domain and the true output sequence, where we note the error is of the order of 10^{-15}. In addition, the normalized mean square error (NMSE) for Figure 11 (a) is 0.03591, while the NMSE for Figure 11 (b) is 8.131×10^{-32}. This virtually perfect match with the true output sequence demonstrates the validity of the extended principal domain.

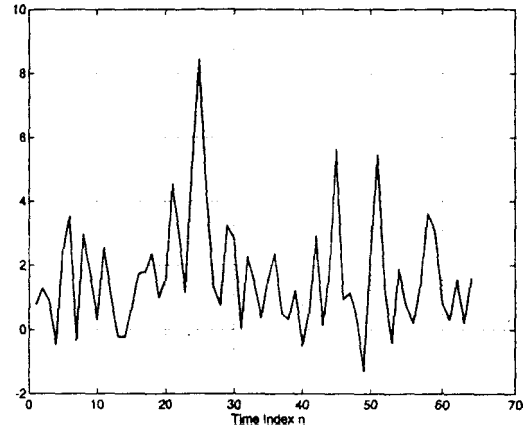


Fig 9. Model Output Computed Based on the Classical Principal Domain.

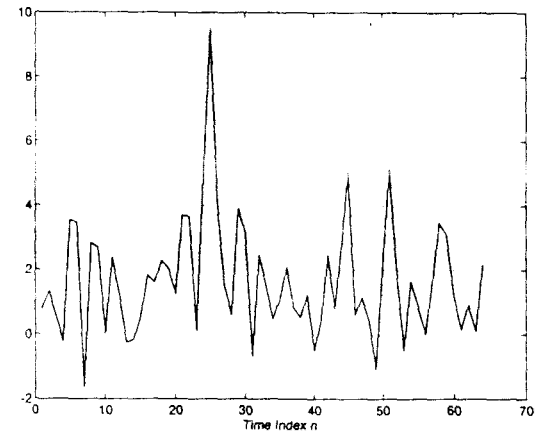
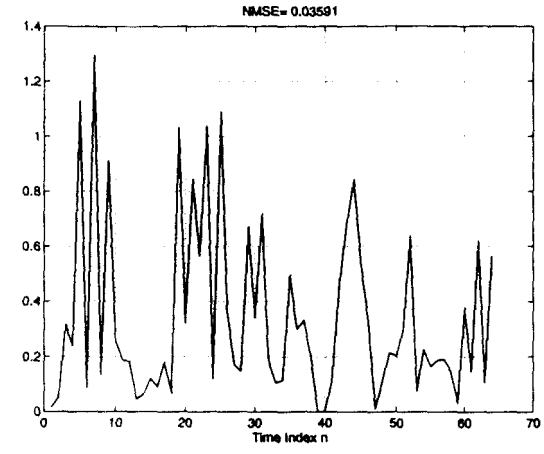
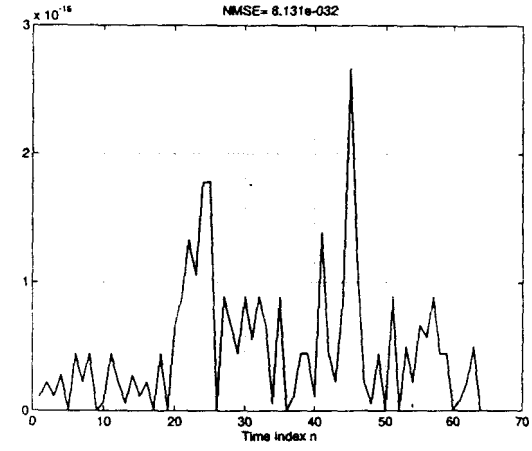


Fig 10. Model Output Computed Based on the Extended Principal Domain.



(a)



(b)

Fig 11. (a) Absolute Value of Error between System Output $y(n)$ and Output Based on the Classical Principal Domain, (b) Absolute Value of Error between System Output $y(n)$ and Output Based on the Extended Principal Domain. Note the scale change in (b).

V. B Identification of Volterra Models

In this subsection, we examine the validity of the extended principal domain in conjunction with the identification of second-order discrete frequency-domain Volterra models. The identification method that we utilize in this experiment is the frequency-domain Volterra model identification method presented in [6, 8]. However, for identifying the quadratic transfer function, the method rests on the extended principal domain proposed in this paper, which we will call the *extended domain method*. Furthermore, for the purpose of comparison, the same method with the *classical principal domain*, which will be called the *classical domain method*, is also applied. Then, we compare the coefficient estimates by the two methods to the known true coefficients. In the following, we briefly describe the classical and extended domain methods.

Classical Domain Method: (2) can be rewritten in a vector form

$$Y(m) = \boldsymbol{H}(m)\, \boldsymbol{X}(m) \tag{21}$$

where

$$\boldsymbol{H}(m) = [H_1(m)\ H_2(p_1, q_1)....H_2(p_n, q_n)...] \tag{22}$$

and

$$\boldsymbol{X}(m) = [X(m)\, X(p_1)\, X(q_1)...X(p_n)X(q_n)...]^T \tag{23}$$

with $p_n + q_n = m$. In (23), the superscript T represents the transpose of a vector. Multiplying both sides of (21) by $\boldsymbol{X}^*(m)$ and taking expected values of both sides, we arrive at

$$E\{Y(m)\, \boldsymbol{X}^*(m)\} = \boldsymbol{H}(m) E\{\boldsymbol{X}(m)\, \boldsymbol{X}^*(m)\} \tag{24}$$

where the superscript * denotes the complex conjugate and transpose of a vector. If $E\{\boldsymbol{X}(m)\, \boldsymbol{X}^*(m)\}$ is nonsingular, the transfer function vector $\boldsymbol{H}(m)$ is given by

$$\boldsymbol{H}(m) = E\{Y(m)\, \boldsymbol{X}^*(m)\}[E\{\boldsymbol{X}(m)\, \boldsymbol{X}^*(m)\}]^{-1} \tag{25}$$

where the superscript -1 represents the inverse of a matrix.

Extended Domain Method: The extended domain method utilizes the same approach as in the classical domain method. The difference between the classical and extended domain methods lies in constructing $\boldsymbol{X}(m)$ and $\boldsymbol{H}(m)$. In the extended domain method, $\boldsymbol{X}(m)$ and $\boldsymbol{H}(m)$ consist of the components satisfying $p_n + q_n = m$ or $p_n + q_n = M + m$. For this reason, the sizes of $\boldsymbol{X}(m)$ and $\boldsymbol{H}(m)$ are larger than those of $\boldsymbol{X}(m)$ and $\boldsymbol{H}(m)$ of the classical domain method.

The time-domain Volterra system (18) is employed to generate input and output data records to be utilized in this experiment. We apply a zero-mean white Gaussian sequence of unit variance, which is generated using MATLAB RANDN, to the second-order Volterra system (18). Since the focus of this simulation lies on demonstrating the effect of the extended principal domain on the frequency-domain Volterra model identification, we do not consider additive noise in the input-output data records. These noise-free data records consist of 32×500 sample points and are divided up into 500 segments of 32 data points each. Each segment is DFTed via the FFT algorithm to calculate various second-, third-, and fourth-order sample spectra, which are then ensemble averaged to obtain the spectral moment estimates.

To quantify the quality of the estimated frequency-domain linear and quadratic transfer functions, the normalized mean square errors are defined as follows:

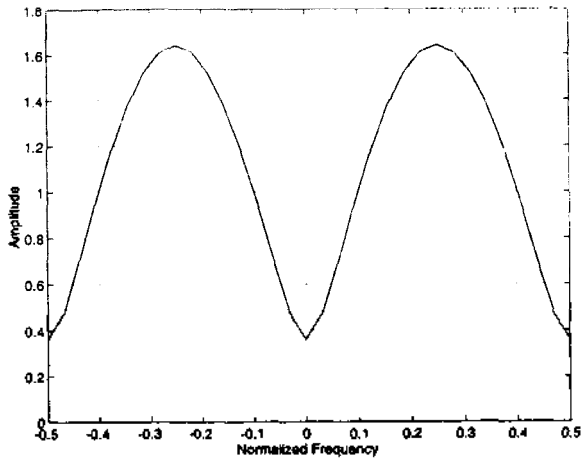
$$\mathrm{MSEL} = \frac{1}{M} \sum_{m=0}^{M-1} \frac{|H_1(m) - \hat{H}_1(m)|^2}{|H_1(m)|^2} \tag{26}$$

$$\mathrm{MSEQ} = \frac{1}{N_q} \sum_{(p,q)\in S} \frac{|H_2(p, q) - \hat{H}_2(p, q)|^2}{|H_2(p, q)|^2} \tag{27}$$

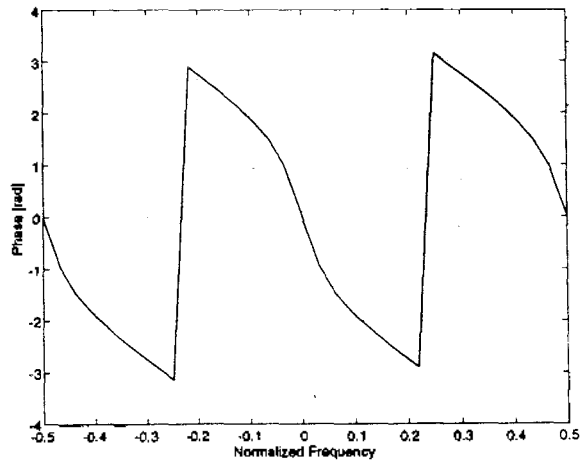
where the hat denotes an estimated quantity. For the extended domain method, S represents the extended principal domain and N_q is 1024, while for the classical domain method, S denotes the classical principal domain and N_q is 768. M signifies the number of points in the FFT used in the experiment.

Figures 12 and 13 show the amplitudes and phases of the linear and quadratic transfer functions estimated by the extended domain method, respectively. Figures 14 and 15 show the amplitudes and phases of the linear and quadratic transfer functions estimated by the classical domain method, respectively. Table 1 summarizes the estimation results by the two methods in terms of the mean squared error defined in (26) and (27). The virtually perfect match

demonstrates the validity of the extended principal domain, vs the classical principal domain in the identification problem.

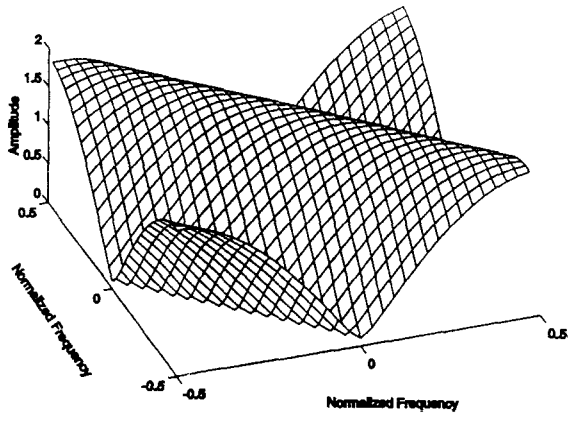


(a)

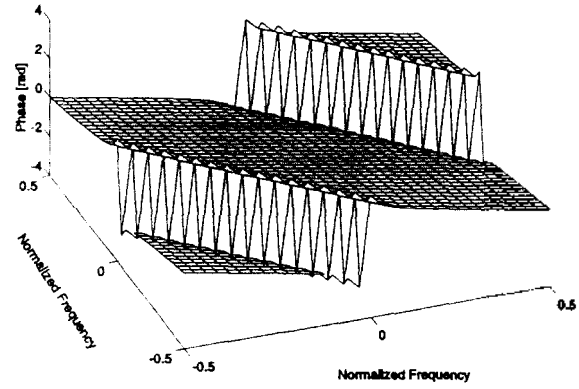


(b)

Fig 12. Estimated Linear Transfer Function by the Extended Domain Method: (a) Amplitude, (b) Phase.

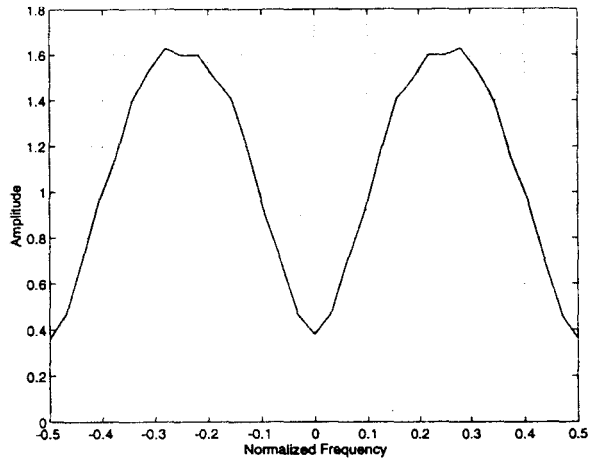


(a)

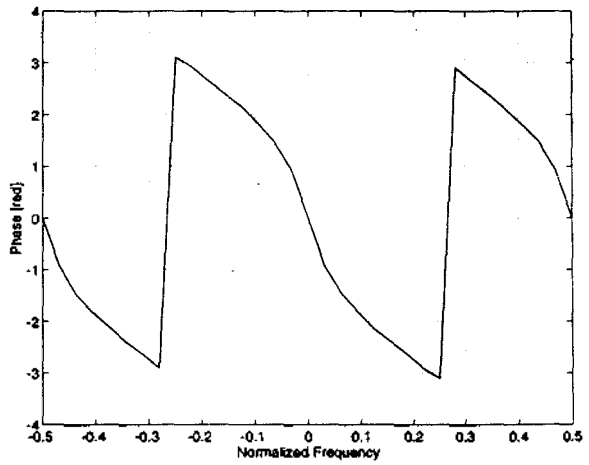


(b)

Fig 13. Estimated Quadratic Transfer Function by the Extended Domain Method: (a) Amplitude, (b) Phase.

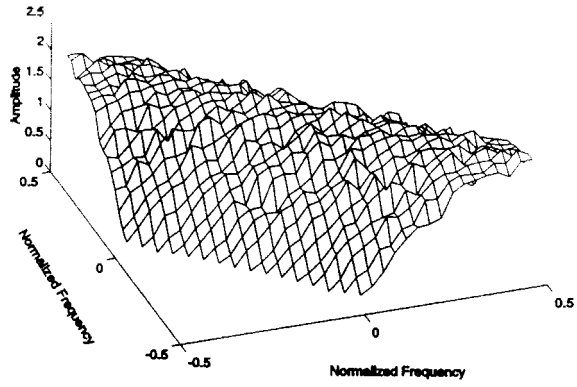


(a)

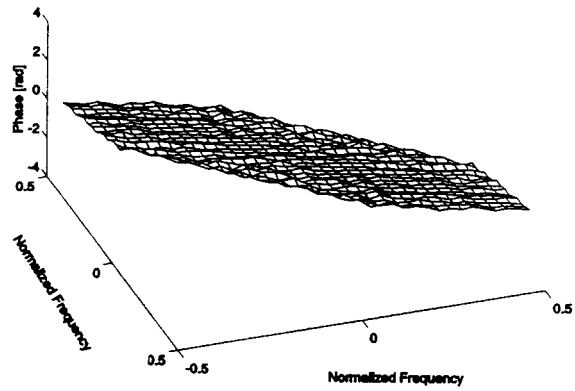


(b)

Fig 14. Estimated Linear Transfer Function by the Classical Domain Method: (a) Amplitude, (b) Phase.



(a)



(b)

Fig 15. Estimated Quadratic Transfer Function by the Classical Domain Method : (a) Amplitude, (b) Phase.

Table 1. Normalized Mean Square Errors of Linear and Quadratic Transfer Functions Estimated by Extended and Classical Domain Methods

Method	MSEL	MSEQ
Extended	1.5177e-031	2.6939e-031
Classical	9.3907e-004	9.8182e-003

V. Conclusion

In this paper, we presented the extended principal domain for Volterra kernels which leads to an improved estimate of the Volterra system's response. Using computer simulation, we investigated the validity of the extended principal domain by comparing Volterra model outputs based on the classical and extended principal domains. In addition, the improvement achieved by employing the extended principal domain for estimating Volterra transfer function was demonstrated using computer simulation. Especially, the extended domain identification method was demonstrated to be much better than the classical domain identification method. In this paper, our discussion on the extended principal domain is limited to the quadratic case. In principal, however, the same procedure can be extended to higher orders.

References

1. G. Karam and H. Sari, "Analysis of Predistortion, Equalization, and ISI Cancellation Techniques in Digital Radio Systems with Nonlinear Transmit Amplifiers," *IEEE Trans. on Communications*, vol. 37, no. 12, pp. 1245-1253, December 1989.
2. R. Hermann, "Volterra Modeling of Digital Magnetic Saturation Recording Channels," *IEEE Trans. Magnetics*, vol. 26, no. 5, pp. 2125-2127, September 1990.
3. J. Tsimbinos and K. V. Lever, "Application of Higher-Order Statistics to Modelling, Identification, and Cancellation of Nonlinear Distortion in High-Speed Samplers and A/D Converters using the Volterra and Wiener Models," *IEEE SP Workshop on Higher-Order Statistics*, pp. 379-383, South Lake Tahoe, CA, 1993.
4. J. Thyssen, H. Nielsen, and S. D. Hansen, "Non-linear Short Term Prediction in Speech Coding," *Proc. of ICASSP 94*, vol. 1, pp. 185-188, 1994.
5. I. W. Sandberg, "Uniform Approximation with Doubly Finite Volterra Series," *IEEE Trans. on Signal Processing*, vol. 40, no. 6, pp. 1438-1442, June 1992.
6. K. I. Kim and E. J. Powers, "A Digital Method of Modeling Quadratically Nonlinear Systems with a General Random Input," *IEEE Trans. ASSP*, vol. 36, no. 11, pp. 1758-1769, November 1988.
7. L. O. Chua and C. -Y. Ng, "Frequency-Domain Analysis of Nonlinear Systems : Formulation of Transfer Functions," *Electronic Circuits and Systems*, vol. 3, no. 6, pp. 257-269, November 1979.
8. C. L. Nikias and A. P. Petropuou, *Higher-Order Spectra Analysis : A Nonlinear Signal Processing Framework*, New Jersey : Prentice-Hall Inc., 1993.
9. D. R. Brillinger and M. Rosenblatt, "Computation and Interpretation of Kth-Order Spectra," in Spectral Analysis of Time Series, B. Harris, ed., Wiley, New York, NY, pp. 189-232, 1967.
10. W. J. Rugh, *Nonlinear System Theory-The Volterra/Wiener Approach*, Johns Hopkins University Press, Balimore, 1981.
11. H. J. Nussbaumer, *Fast Fourier Transform and Convolution Algorithms*, Berlin : Springer-Verlag, 1981.
12. A. V. Oppenheim and R. W. Schafer, *Digital Signal Processing*, New Jersey : Prentice-Hall Inc., 1975.

▲임 성 빈(SungBin Im)
현재 : 숭실대학교 정보통신공학과 교수

▲이 원 철(WonCheol Lee)
현재 : 숭실대학교 정보통신공학과 교수

▲배 명 진(MyungJin Bae)
현재 : 숭실대학교 정보통신공학과 교수