

## 디지털 신호처리의 개요 및 기술 동향

鄭 在 皓  
仁荷大學校 電子工學科

### I. 디지털 신호처리

디지털 신호처리(digital signal processing)는 주어진 신호(signal)를 일련의 숫자들(numbers) 또는 기호들(symbols)을 사용하여 나타내고, 그들(즉 숫자들 또는 기호들)을 처리하는 분야를 말한다. 이와 같은 디지털 신호처리의 목적은 주어진 신호를 분석하기에 또는 처리하기에 보다 효과적인 형태로 바꾸거나, 주어진 신호의 특징 파라메타들(characteristic parameters)의 추정이라 할 수 있다. 디지털 신호처리 분야는 지난 25년간 놀라운 속도로 발전되어 왔으며, 본 분야의 연구 개발은 앞으로 더욱 더 활성화 되어 질 것이다. 디지털 신호처리 기술은 현재 통신 시스템, telephony, radar와 sonar, 음성 및 화상 신호처리, 의용 생체 등 수 많은 분야에 효과적으로 응용되어지고 있으며 응용분야는 앞으로도 계속 늘어날 것으로 기대된다.

디지털 신호처리는 1960년대 중반에 신호처리(signal processing) 분야내에서 새로운 부문으로 도출되었다. 도출된 이유는 디지털 컴퓨터를 이용한 신호처리의 실시간 구현(real-time implementation)이라 할 수 있다. 초기 디지털 신호처리 분야의 발전에 가장 큰 영향을 준 것은 아마도 Kaiser의(당시 Bell 연구소에서) 디지털 필터의 설계 및 구현에 관한 연구 결과이다. Kaiser는 그의 연구를 통하여, 효과적인 디지털 필터의 설계방법을 제시 하였다. 디지털 신호처리 분야는 1965년 discrete Fourier transform을 효과적으로 계산할 수 있는 알고리즘의 출현과 함께 발전을 계속했다. 이 알고리즘은 Cooley와 Tukey에 의하여 제안 되었으며 fast Fourier

transform 또는 FFT로 불리어 지고 있다.<sup>(1)</sup> FFT 알고리즘은 아마도 디지털 신호처리 분야에서 가장 중요한 알고리즘이라고 할 수 있을 것이다. FFT 알고리즘은 discrete Fourier transform의 계산에 필요한 계산량을 현격하게 줄였다. FFT 알고리즘의 출현은 매우 복잡한 신호처리 알고리즘의 실시간 구현을 가능케 하였으며, 또한 구현이 어려웠던 복잡한 시스템의 실제적인 구현을 가능케 하였다.

디지털 신호처리는 그림 1과 같이 크게 세 부분으로 나누어 생각할 수 있다. 첫번째 부분은 아날로그-디지털 변환기(analog-to-digital converter, A/D)이다. 많은 경우에 디지털 신호처리를 적용하고자 하는 신호 자체가 디지털 형태인 경우도 있으나, 대부분의 경우는 아날로그 형태를 갖는다. 이러한 경우, 아날로그-디지털 변환기를 사용하여 주어진 아날로그 신호를 일련의 숫자들 또는 기호들로 바꾼다. 이 과정을 샘플링(sampling)이라 부른다. 디지털 신호처리를 구성하는 두번째 부분은 discrete-time 시스템이라 할 수 있다. 이 부분에서는 디지털 형태로 바꾼 입력 신호에 다양한 디지털 신호처리 과정들을 적용시키며(digital filtering), 적용시키는 과정들은 전체 시스템의 목적에 따라 결정된다. 예를 들어, 주어진 신호의 특징 파라메타들을 추정 하거나, 분석 처리하기에 편리한 형태로 주어진 신호를 변환시키는

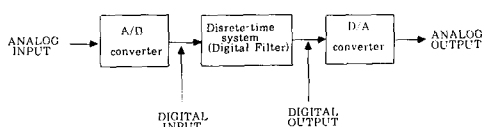


그림 1. 디지털 신호처리 블록 도표

등 다양한 목적을 생각할 수 있다. 디지털 신호처리 과정의 마지막 부분은 디지털-아날로그 변환(digital-to-analog converting, D/A)이다. 이 부분에서는 디지털 형태의 신호를 다시 아날로그 형태로 변환 시킨다.

## II. 샘플링

위에서 언급하였듯이, 아날로그 신호에 디지털 신호처리 기술을 적용하기 위해서는, 먼저 아날로그-디지털 변환기를 사용하여 아날로그 신호를 디지털 신호로 바꾸어야 한다. 변환을 위하여 이론상으로는 많은 방법들이 적용될 수 있으나, 실제적으로 가장 손쉬운 방법은 주어진 아날로그 신호를 균등한 시간 간격마다 샘플링(sampling) 하는 것이라 할 수 있다. 그림 2에 표시된 바와 같이, 주어진 아날로그 신호  $x(t)$ 를 균등한 시간 간격  $T$ 초 마다 샘플링을 하였다고 가정하자. 이 경우, 샘플링한 결과, 즉 디지털 신호  $x[n]$ 은 다음과 같이 나타낼 수 있다.

$$x[n] = x(nT), \quad -\infty < n < \infty \quad (1)$$

디지털 신호  $x[n]$ 과 아날로그 신호  $x[t]$  사이의 관계는, 샘플링을 통하여 초래되는 엘리어싱(aliasing)이라는 현상을 관찰하므로 쉽게 알 수 있다.

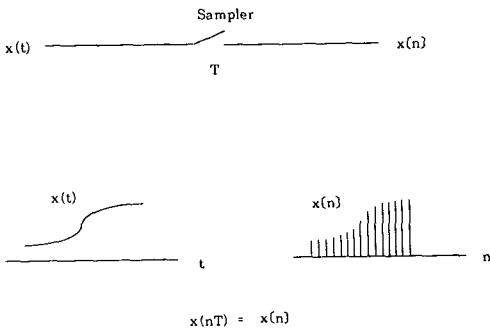


그림 2. 샘플링 블록도표

엘리어싱 현상은 주파수 영역 (frequency domain)에서 쉽게 설명되어 진다. [2,3,4,5] 디지털 신호  $x[n]$ 의  $z$ -변환( $z$ -transform)을  $X(z)$ 라 하고, 아날로그 신호  $x[t]$ 의 Fourier-변환 (Fourier-transform)을  $X_a(\omega)$

라 하자. 디지털 신호  $x[n]$ 의  $z$ -변환은 정의에 의하여 다음과 같이 쓸 수 있다.

$$X(z) = \sum_{n=-\infty}^{\infty} x[n]z^{-n} \quad (2)$$

식 (2)에서,  $z = e^{j\omega}$ 라 가정하면, 다음과 같은 식을 얻는다.

$$X(e^{j\omega}) = \sum_{n=-\infty}^{\infty} x(nT)e^{-j\omega n} \quad (3)$$

한편, 아날로그 신호  $x(t)$ 는 Fourier-역변환 (inverse Fourier-transform)을 사용하여 식 (4)와 같이 나타낼 수 있으므로, 즉,

$$x(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} X_a(\omega)e^{j\omega t} d\omega, \quad (4)$$

식 (3)은 다음과 같이 쓸 수 있다.

$$X(e^{j\omega}) = \sum_{n=-\infty}^{\infty} \frac{1}{2\pi} \int_{-\infty}^{\infty} X_a(\omega)e^{j\omega nT} du e^{-j\omega n} \quad (5)$$

$$= \frac{1}{2\pi} \int_{-\infty}^{\infty} X_a(\omega) \sum_{n=-\infty}^{\infty} e^{jn(\omega T - \omega)} du \quad (6)$$

$$= \frac{1}{T} \sum_{k=-\infty}^{\infty} X_a[(\omega + 2\pi k) / T] \quad (7)$$

식 (7)로 부터 디지털 스펙트럼(spectrum)  $X(e^{j\omega})$ 는 스케일링(scaling)되어 일정한 간격을 두고 존재하는 무한개의 아날로그 스펙트럼  $X_a(\omega)$ 들의 합으로 표시됨을 알 수 있다.

그림 3과 그림 4에 디지털 스펙트럼  $X(e^{j\omega})$ 와 아날로그 스펙트럼  $X_a(\omega)$ 사이의 관계가 설명 되어져 있다. 아날로그 신호  $x(t)$ 의 주파수 특성을 다음과 같이 가정하자.

$$X_a(\omega) = 0, \quad \text{for } |\omega| > \frac{\pi}{T} \quad (8)$$

이와같은 경우, 만약 샘플링의 간격  $T$ 가  $T'$ 의 값보다 작다면, 식 (7)에서 우변의 합을 구성하는 각각의 성분들은 겹쳐지지 않는다. 그림 3에 설명 되어져 있다. 반면에 샘플링의 간격  $T$ 값이 점점 커지면, 각각의 성분들 사이의 거리는 점점 좁혀진다. 결국  $T$ 의 값이  $T'$ 의 값보다 클 경우, 구성원들은 서로 겹쳐지게 된다. 이와같이 한 주파수 대역의 정보가 다른 주파수 대역으로 이동(shifting)되는 현상을 엘리어싱

이라 한다. 그림 4에 엘리어싱 현상이 설명 되어져 있다. 따라서 주어진 아날로그 신호가 주파수 폭이 제한된 특성을 가질 때, 엘리어싱 현상은 방지될 수 있으며, 그 방법은 샘플링의 간격인  $T$ 값을 충분히 작게 하는 것이다. 이와 같은 원리를 소위 샘플링의 정리 (sampling theorem)이라 하며, 샘플링의 정리는 디지털 신호처리의 가장 근본을 이루는 중요한 원리이다.

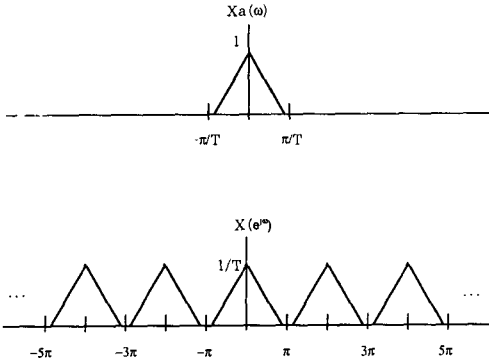


그림 3. 디지털 스펙트럼  $X(e^{j\omega})$ 와 아날로그 스펙트럼  $X_a(\omega)$ 사이의 관계 I

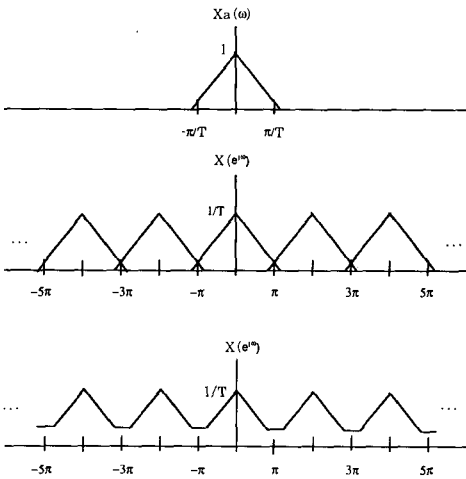


그림 4. 디지털 스펙트럼  $X(e^{j\omega})$ 와 아날로그 스펙트럼  $X_a(\omega)$ 사이의 관계 II

### III. Linear Time-Invariant System

Discrete-time 시스템은 주어진 입력신호  $x[n]$ 을

일정한 규칙에 따라 출력신호  $y[n]$ 으로 바꾸는 변환기라 할 수 있다. (그림 1 참조) 주어진 Discrete-time 시스템의 입력과 출력 사이의 관계는 다음과 같이 식을 사용하여 나타낼 수 있다.

$$y[n] = T\{x[n]\} \tag{9}$$

식 (9)에서 operator  $T$ 는 입력신호  $x[n]$ 과 출력신호  $y[n]$ 사이의 관계를 나타낸다. Discrete-time 신호처리 시스템은 아날로그 시스템에 비하여 많은 장점을 지니고 있다.

특히, Discrete-time 시스템은 아날로그 하드웨어로는 구현이 어려운 신호처리 (또는 신호변환)을 가능하게 한다는 큰 장점이 있다. 따라서, 많은 경우에 주어진 신호를 디지털 신호 처리 방법을 사용하여 처리하는 이유가 여기에 있다.

주어진 시스템이 다음과 같은 성질을 가질 때, 그 시스템을 linear 시스템이라 한다. 먼저, 두 신호  $x_1[n]$ 과  $x_2[n]$ 이 주어진 시스템의 입력들이고 두 신호  $y_1[n]$ 과  $y_2[n]$ 이 각각의 출력들이라 가정하자. 이러한 시스템에, 입력으로  $ax_1[n] + bx_2[n]$ 을 적용했다 가정하자. 이 때에, 시스템의 출력이  $ay_1[n] + by_2[n]$ 인 경우, 주어진 시스템을 linear 시스템이라 한다. 한편, 입력  $x[n]$ 이 출력  $y[n]$ 을 초래할 경우, 신호  $x[n-n_0]$ 은 출력으로  $y[n-n_0]$ 을 생산한다면, 주어진 시스템을 time-invariant 시스템이라 한다.

Discrete-time 시스템들 중에서도, 위에서 언급한 두가지의 성질, 즉 linearity와 time-invariance를 모두 만족하는 시스템들은 특히 중요하며, linear time-invariant (LTI) 시스템이라 부른다. LTI 시스템의 가장 중요한 특성은, 시스템의 임펄스 응답이 주어졌을 때, 입력신호  $x[n]$ 과 출력신호  $y[n]$ 의 관계가 컨볼루션(convolution)을 통하여 다음 식과 같이 표현되는 것이라 할 수 있다. <sup>[2.3.4]</sup>

$$y[n] = \sum_{k=-\infty}^{\infty} x[k] h[n-k] \tag{10}$$

따라서, LTI 시스템의 경우, 그림 5에 나타 내었듯이, 임펄스 응답  $h[n]$ 에 의하여 시스템의 성질이 결정된다. 보편적으로, 컨볼루션의 관계는 다음과 같은 기호를 사용하여 나타낸다.

$$y[n] = x[n] * h[n] \tag{11}$$

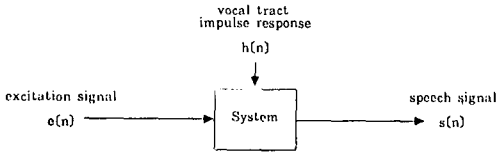


그림 5. 음성 신호의 모델링

LTI 시스템은 신호처리의 많은 응용분야에 실제적으로 사용되고 있다. 한 예로서, 디지털 음성 신호처리 (digital speech signal processing)을 들 수 있다. 음성신호는 음원신호가 성도를 통과하여 생성된다. 성도의 모양은 일반적으로 시간의 흐름에 따라 천천히 바뀐다. 따라서, 성도는 시간에 따라 그 성질이 천천히 바뀌는 discrete time-varying 시스템으로 모델링 할 수 있다. 또한, 음원신호와 성도의 모양이 서로 독립적이라 (independent) 가정한다면, 음성신호는 discrete time-varying 시스템의 출력이라 해석 할 수 있다. 한단계 더 나아가서, 성도의 모양이 천천히 바뀌므로, 성도를 모델링 하는 시스템의 성질이 매우 짧은 기간들 안에서는 (예를 들어 10 내지 30 ms) 일정하다고 가정 할 수 있다. 결과적으로 음성신호  $s(n)$ 은 짧은 시간들의 간격 내에서는, 다음식과 같이 콘볼루션을 통하여 표현될 수 있다.

$$s[n] = h[n] * e[n] \quad (12)$$

식 (12)에서,  $h[n]$ 은 성도의 임펄스 응답을 나타내며, 은 음원신호를 표현한다. 식 (12)는 LTI 시스템이 갖는 콘볼루션이라는 성질을 응용한 것이다. 음성신호의 많은 응용분야에서는,  $s(n)$  자체보다도  $s(n)$ 을 구성하는  $h(n)$ 과  $e(n)$ 을 처리하는 것이 더욱 효과적이다.<sup>[6,7]</sup> 이와 같은 분리를 위하여 많은 방법들이 제안되어 현재 사용 되어지고 있다. 하나의 예로서, homo-morphic filtering을 들 수 있다.<sup>[8,9,10]</sup>

#### IV. 디지털 필터<sup>[2,3,4,5]</sup>

대략적으로, 디지털 필터란 discrete LTI 시스템의 실현 (realization)이라 말할 수 있다. 디지털 필터들은 크게 두 그룹, 즉 유한 임펄스 응답 (finite impulse response, FIR) 필터와 무한 임펄스 응답

(infinite impulse response, IIR) 필터로 구분한다. 필터의 임펄스 응답이 유한한 샘플들로 표시되어질 때, 그 필터를 FIR 필터라 한다. 반면에 필터의 임펄스 응답이 무한개의 샘플들로 구성되어질 때, 그러한 필터를 IIR 필터라 부른다. 따라서, FIR 필터의  $z$ -변환,  $H_f(z)$ 은 변수  $z$ 의 다항식으로 다음 식과 같이 나타낼 수 있다.

$$H_f(z) = \sum_{n=0}^{N-1} h[n] z^{-n} \quad (13)$$

반면에, IIR 필터의  $z$ -변환,  $H(z)$ 는 다음과 같은 rational 형태로 표시된다.

$$H_i(z) = \frac{\sum_{m=0}^{M-1} b_m z^{-m}}{1 + \sum_{n=1}^{N-1} a_n z^{-n}} \quad (14)$$

필터의 설계를 위해서는 설계 명세 조건들이 주어져야 하며, 이러한 조건들은 일반적으로, 시간 대역 또는 주파수 대역에서의 조건들로 주어진다. 실제로 많은 경우, 어느 특정한 주파수 대역만을 통과시키고 그 외의 대역은 차단시키는 성질을 갖는 필터가 많이 사용되며, 이와같은 경우 설계 명세는 주파수 대역에서의 조건들로 표현된다. 주어진 설계 명세 조건들을 만족하는 디지털 필터의 설계과정에서 항상 염두에 두어야 할 것은, 구현하는 필터의 복잡도 (complexity)를 최소한으로 유지해야 한다는 것이다.

IIR 필터의 한 설계 방법은, 적합한 아날로그 필터를 설계한후 샘플링을 통하여 디지털 필터로 변형시키는 방법이다. 이 방법은 매우 널리 사용되어지고 있으며, 그 이유는 첫째, 아날로그 필터 설계 방법이 현재 많이 발전되어 있고, 둘째, 그와 같이 발전된 아날로그 필터 설계 기술을 디지털 필터 설계에 이용하고자 함이다. 이와같은 디지털 필터 설계 방법은, 특히, 저역통과(low pass), 고역통과(high pass), 대역통과(band pass), 그리고 대역저지(band stop) 필터등의 설계에 매우 효과적이다. 샘플링을 통한 아날로그 필터의 디지털 필터로의 변환 과정에서 유의 해야 할 것은, 안정성(stability)이다. 즉, stable한 아날로그 필터의 변환은 stable한 디지털 필터를 만들어야 한다. 많이 사용되는 변환 방법으로는, impulse invariant 변환 방법과 bilinear 변환

방법 등을 들 수 있다. Impulse invariant 변환 방법은 변환 과정에서 엘리어싱을 초래 한다. 따라서, impulse invariant 방법은 band limited 필터의 설계에만 적당하다 할 수 있다. Bilinear 변환 방식은 impulse invariant 변환 방식에서 초래되는 엘리어싱의 문제를 배제 시킨다. 허나, bilinear 변환 방식에서는 아날로그 필터로부터 디지털 필터로의 mapping이 비선형(non linear)이라는 단점이 있다.

IIR 필터 설계에 있어서, 주어진 설계 명세 조건이 위에서 언급한 변환 방식의 사용으로는 부적합할 경우, 컴퓨터를 사용하여 필터를 설계 한다. 이와 같이 필터 설계를 위하여, 선형 또는 비선형 방정식들을 컴퓨터로 푸는 방법을 computer-aided 설계 방법이라 한다. Computer-aided에 기초를 둔 몇가지의 IIR 필터 설계 방법이 필터 설계 분야에 소개되어 있다.

많은 IIR 필터 설계 방법중 어느 특정한 방법이 항상 우월하다고 할 수는 없다. 사용하고자 하는 용도에 따라 제일 적합한 필터가 다를 수 있는 것이다. 허나, 많은 경우, IIR 필터 설계는 bilinear 변환 방식을 사용하여 lowpass나 bandpass 필터의 설계에 사용된다.

FIR 필터는 몇가지의 장점을 갖고 있다. 예를 들어, FIR 필터는 항상 안정성(stability)과 실현성(realizability)이 보장된다. 그 이유는 FIR 필터의 임펄스 응답은 유한값을 갖는 유한개의 샘플들로 표시되기 때문이다. FIR 필터의 또다른 특성은, 필터의 위상(phase)을 선형(linear)으로 설계 할 수 있다는 것이다. 반면에, IIR 필터는 주어지는 진폭특성(magnitude characteristics) 조건을 매우 효과적으로 만족 시킬 수 있으나, 위상을 선형으로 설계하기에 어렵다는 단점이 있다.

FIR 필터 설계의 가장 대표적인 방법으로는 windowing 방식을 꼽을 수 있다. Windowing 방식에서는, 유한 길이를 갖는 window를 IIR sequence에 적용시켜 절단(truncation) 함으로, FIR 필터를 구현한다. 많이 사용되는 window로서는, rectangular, Bartlett, Hamming, Hanning, Blackman, 그리고 Kaiser window 등을 들 수 있다. Windowing 방식은 매우 간단하지만, IIR sequence를 절단하는 과정에서 생기는 Gibbs 현상 때문에 sharp cutoff filtering이 어렵다는 단점이 있다. 따라서, sharp cutoff filtering이 요구될 경우, 사용되는 window의 길이가 상대적으로 늘어나

며, 이러한 현상은 필터의 복잡도를 증가 시킨다.

대개의 경우, Windowing 방식을 사용하여 설계한 필터들은, 주어진 설계 조건을 만족하는 최적의 필터는 아니다. 어떠한 에러 기준(error criterion) 안에서, 주어진 설계 조건들을 최적으로 만족하는 FIR 필터를 설계하기 위해서는, computer-aided 설계방식을 택해야 한다. 허나, Computer-aided 방식은 설계 과정이 상대적으로 복잡하고 많은 시간이 요구되는 단점이 있다.

디지털 필터의 설계는 위에서 언급 하였듯이 참으로 다양하다 할 수 있다. 허나, 어느 특정한 필터가 다른 필터들보다, 또는 어느 종류의 필터들이 다른 필터들보다 (예를 들어, IIR이 FIR 보다) 낫다고 일률적으로 말 할 수는 없다. 어떠한 필터를 사용할 것인가 하는 문제는, 주어진 필터 설계 조건과 경제성 그리고 사용할 수 있는 장비 여건등에 따라서 결정되어야 할 것이다.

## V. 디지털 신호 처리 칩 (DSP Chip)

일반적으로, 디지털 신호 처리 알고리즘들은 많은 양의 계산을 요구한다. 특히, 이러한 알고리즘들을 실제로 응용 분야에 적용할 경우, 알고리즘들의 실시간 구현(real-time implementation)이 중요한 관건으로 대두 된다. 이와같이, 많은 양의 계산을 요하는 디지털 신호 처리 알고리즘들의 구현을 위하여 제작된 고속(high-speed) 마이크로 프로세서들을 디지털 신호 처리 칩(Digital Signal Processing Chip), 또는 DSP 칩이라 부른다. 이러한 DSP 칩들은 매초당 수 백만 내지 수 천만의 DSP 연산(operation)이 가능하며, VLSI 기술의 급격한 발전은 값싸고 크기가 작은 이와 같은 DSP 칩들의 실현을 가능케 하였다.

차세대의 DSP 칩들은 성능면에서 현재로서는 예측하기도 어려울 만큼의 지속적인 성능 향상이 기대된다. 더욱 다양한 연산 능력, 더욱 빠른 연산 속도, 더욱 확장된 기억 장치, 그리고 효과적인 연산의 다중 처리(multi-processing)등 이 모든 것이 차세대의 DSP 칩의 가치를 더욱 높여 줄 것이며, 따라서 디지털 신호 처리의 응용분야는 앞으로도 계속 확장 되어 질 것이다.

參考文獻

[1] J. W. Cooley and J. W. Tukey. "An Algorithm for the Machine computation of Complex Fourier Series." *Mathematics of Computation*, vol.19, pp.297-301, Apr. 1965.

[2] A. V. Oppenheim and R. W. Schaffer, *Discrete-Time Signal Processing*, Prentice-Hall, New Jersey, 1989.

[3] A. V. Oppenheim and R. W. Schaffer, *Digital Signal Processing*, Prentice-Hall, New Jersey, 1975.

[4] L. R. Rabiner and B. Gold, *Theory and Application of Digital Signal Processing*, Prentice-Hall, New Jersey, 1975.

[5] S. A. Tretter, *Introduction to Discrete-Time Signal Processing*, John Wiley & Sons, New York, 1976.

[6] L. R. Rabiner and R. W. Schaffer, *Digital Processing of Speech Signals*, Prentice-Hall, New Jersey, 1978.

[7] D. O'Shaughnessy, *Speech Communication*, Addison Wesley, Massachusetts, 1987.

[8] A. V. Oppenheim and R. W. Schaffer, "Homomorphic Analysis of Speech," *IEEE Trans. on Audio and Electroacoustics*, pp. 118-123, June, 1968.

[9] A. V. Oppenheim, "Speech Analysis-Synthesis System Based on Homomorphic Filtering," *Journal of Acoustical Society of America*, vol. 45, pp. 458-465, Feb., 1969.

[10] J. H. Chung and R. W. Schaffer, "A 4.8 Kbps Homomorphic Vocoder Using Analysis-by-Synthesis Excitation Analysis," *Proc. Int. Conf. Acoust., Speech, Signal Processing*, pp. 144-148, May, 1989. (C)

筆者紹介



鄭在皓

1982年 美國 University of Maryland (학사)  
 1984年 美國 University of Maryland (석사)  
 1990年 美國 Georgia Institute of Technology (박사)

1984年 ~ 1985年 美國 Naval Surface Warfare Center, Electronic Engr.  
 1991年 ~ 1992年 美國 AT&T Bell Laboratories, 연구원  
 1992年 ~ 인하대학교 공과대학 전자공학과 조교수

주관심 분야 : 신호처리, 음성통신