

## 통신 시스템의 시뮬레이션

李 聖 元, 趙 東 浩  
慶熙大學校 電子計算工學科

### I. 개 요

수년 동안, 통신 시스템을 분석하고 설계하기 위한 많은 도구들이 컴퓨터를 기반으로 하여 만들어져 왔다. 컴퓨터를 이용하는 이유는 현재의 통신 기술이 몇 십년 전의 과거에 비하여 급성장한 것 이외에 현재의 시스템이 과거의 시스템과 달리 단순한 가우시안(Gaussian) 잡음 모델로 나타내기가 어렵고, 대역폭 제한, 인접 신호 간섭, 다중 경로, 비선형성, 다중 호스트에 의한 영향 등의 여러 사항들을 고려해야 하기 때문이다. 따라서, 통신 시스템을 복잡한 시스템과 채널로 모델링할 때는 수학적 접근방법만으로 해결하기가 어렵고, 이를 설계하거나 분석하기는 사실상 불가능해진다. 따라서, 컴퓨터를 이용함으로써 이러한 환경들에 대한 접근을 용이하게 할 수 있는데, 이 글에서 통신 시스템을 시뮬레이션 할 때에 고려해야 할 기본적인 사항들에 대해서 논하고자 한다.

시뮬레이션에 대한 관심이 증가하고 있는 또 하나의 원인은 현재의 컴퓨터가 가지는 강력한 성능이다. 즉 통신 기술자들은 자신의 책상 앞에서 고도로 복잡한 시스템의 시뮬레이션을 수행할 수 있게 되었다. 이와 함께 컴퓨터가 제공하는 그래픽 화면과 레이저 프린터 등을 통하여 보다 효과적인 출력물을 볼 수 있게 되었다.

전통적인 수학적 분석방식과 컴퓨터를 이용한 분석방식은 모두가 블록 다이어그램으로 표시되는 여러개의 서브 시스템이 하나의 큰 시스템을 이루는 상호연결형 모델로 나타내어 진다는 공통점을 가진다. 서브 시스템이나 기능 블록은 시스템에 대한 입력과 출력간의 관계를 기술하는 단계를 나타낸다. 전통적인 수학적 접근과 컴퓨터를 이용한 방안 상호간의 정확성 문제는 시스템 모델의 정확성에 기인한다. 따라서, 시스템 모델을 만들 때에 사용된 모든 가정들은 잘 정리되고 파악되어야 한다.

시뮬레이션이 수학적 접근방식의 대리방안만은 아니다. 어느 정도의 분석은 시뮬레이션이 옳게 수행되어졌는 가하는 점과 결과가 신뢰할 만한가에

대한 지침으로 필요하다. 따라서, 시뮬레이션이 적절하게 사용된다면 전형적인 수학적 방안들과 함께 사용되어야 한다. 시뮬레이션 결과는 종종 시스템에서 가장 중요한 변수들을 추출해 내고, 그외의 변수들은 고려대상에서 제거하는 기준으로써 사용될 수 있다. 다시 말해서, 시뮬레이션 결과는 분석을 위한 자료로 이용되는데, 이는 적절하게 작성된 시뮬레이션이 시스템의 기능을 잘 보여주기 때문이다.

비록 시뮬레이션이 분석과 설계를 위하여 필요한 도구이지만, 몇 가지 문제를 갖고 있다. 예를 들어, 연속적인 파형도 시뮬레이션에서는 이산 신호 샘플로 나타내어져야 하므로, aliasing 에러를 최소화할 수 있도록 샘플링되어야 한다. 아무리 단순한 문제라 하더라도 aliasing 에러를 줄이기 위해서는 높은 샘플링 주파수가 요구된다. 이 경우 높은 샘플링 주파수는 시뮬레이션 수행시간에 영향을 끼친다.

서론에 이어 제2장에서는 신호생성, 선형 시스템의 모델링, 비선형시스템의 모델링에 대하여 알아보도록 한다. 또한, 제3장에서는 통신 시스템의 성능을 측정하기 위하여 사용되는 척도에 대해서 기술한다. 아울러, 제4장에서는 통신 시스템의 시뮬레이션 도구에 관해 기술하며, 마지막으로 제5장에서는 시뮬레이션 개발에 대해서 언급한다.

## II. 신호와 시스템 모델링

시스템 레벨의 시뮬레이션은 시간 영역이나 주파수 영역, 또는 이의 혼합적인 형태로 나타내어질 수 있다. 이 장에서는 통신 시스템의 디지털 시뮬레이션에서 시간 영역을 기반으로 하여 시스템을 모델링하는 방안을 다룬다.

### 1. 신호 생성

대부분의 통신 시스템에 있어서 결정적(deterministic) 신호와 랜덤 신호가 존재하며, 디지털 컴퓨터가 이러한 신호를 시뮬레이션하기 위해서는

각각의 신호 형태에 맞는 모델이 만들어져야 한다. 결정적 신호는 신호에 대한 정의식을 이용하여 생성할 수 있으며, 랜덤 신호는 일반적으로 PN 시퀀스 알고리즘이나 선형 congruential 알고리즘으로 생성한다. 비록, 두 가지 알고리즘에 대한 수식적인 기술은 서로 다르지만 근본적인 면에서는 서로 동일하다. 근본적으로 컴퓨터가 FSM(finite-state machine)이기 때문에 진정한 랜덤 신호는 만들어질 수 없다. 따라서 근원적으로는 주기적일 수밖에 없다. 이러한 이유로 궁극적으로 비교적 긴 주기를 가지는 결정적 신호를 이용하여 의사 랜덤 시퀀스(pseudo-random sequence)를 생성한다.

선형 congruential 알고리즘은 다음과 같이 표현된다.

$$x[n + 1] = (ax[n] + c) \bmod m \quad (1)$$

윗 식에서  $m$ 은 modulus,  $a$ 는 승수(multiplier),  $c$ 는 증가치를 의미한다. 샘플의 발생속도를 향상시키기 위하여 일반적으로  $c$ 는 0으로 한다. 시퀀스의 처음 값인  $x(0)$ 는 대개 프로세스에서의 시드(seed)라고 한다. 일단 시드가 정해지면 남은 수들은 식(1)에 의해서 발생한다. 이제 남은 문제점은 충분히 긴 주기를 가지도록 식(1)에서 정의된 발생함수의  $a, c$  그리고  $m$ 을 정하는 것이다. 비록 여러 가지 이론들이 이러한 문제를 해결하기 위하여 제안되었으나 실제로 좋은 성능을 발휘하는 랜덤 시퀀스 발생 및 이에 대한 알고리즘은 계속 연구중이다.

### 2. 선형 시스템을 위한 모델들

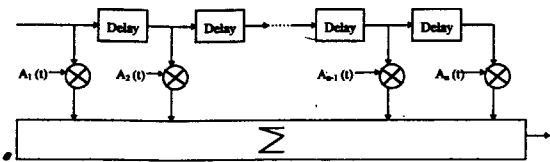
디지털 컴퓨터에서 구현되기에 적합한 선형 시스템 모델들은 시스템의 전달함수  $H(f)$ 나 단위 임펄스 응답함수  $h(t)$ 에 의해서 결정될 수 있다. 만약 전달함수  $H(f)$ 가 lowpass 형태의 시스템을 위한 것이라면 컴퓨터 모델은 표준 디지털 필터 합성 기술을 사용하여 전달함수를 디지털 필터로 매핑시킬 수 있다. 일반적으로 가장 많이 사용되는 합성 기술은 impulse-invariant, step-invariant, bilinear 속성을 만족시키는 z-transform 필터이다.

이러한 합성 방식들은 모두 추정을 전제로 하며 시뮬레이션 결과에 신뢰성을 부가하기 위해서는 이러한 추정에 대한 인식이 확실하여야 한다. 만약, 산술적인 진폭과 위상 응답을 가지는 필터를 시뮬레이션하고 싶다면 원하는 진폭과 위상 응답에 부합하는 주파수 샘플이 필요하다. 이러한 경우에 있어서 샘플들은 단위 펄스 응답  $h(n)$ 을 얻기 위하여 역푸리에 변환되어야 한다. 따라서, 입력 시퀀스들은  $h(n)$ 와 컨볼루션(convolution)되어 필터의 출력으로 나타난다. 다른 방안으로는 블럭화된 FFT 처리를 이용할 수 있는데 이 경우에는 입력 시퀀스를 적절한 크기의 블럭들로 나누어 FFT를 거친 후에 필터 주파수 전달함수와 곱한 값을 역푸리에 변환하여 출력 샘플을 생성한다.

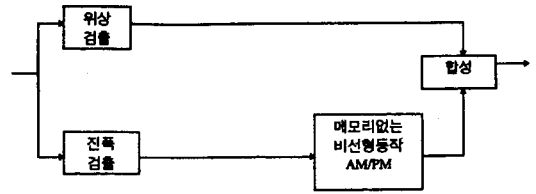
3. 비선형 시간 변형 시스템 모델

비선형이고 시간 변형적(time-varying)인 시스템을 나타내는데 있어서 bandpass 모델의 경우는 몇 가지 문제를 갖고 있다. 비록 선형이고 시간 변형인 시스템에서는 복소수형을 사용할 수 있었지만, 비선형이고 시간 변형인 시스템의 경우는 앞서의 복소수형과 같은 형태를 사용할 수 없다. 따라서, 이러한 시스템을 시뮬레이션하기 위해서는 적절한 추정형에 의지할 수 밖에 없다. 비선형이고 시간 변형인 시스템을 나타내기 위한 모델이 <그림 1>에 나타나 있다. 이는 실질적으로 시간 변형 계수를 가지는 트랜스버살(transversal) 필터이다.

비선형이고 시간 변형인 시스템 모델들은 많이 존재하며, Volterra Serie가 가장 잘 알려진 것 가운데 하나이다. 그러나 이 방안은 많은 계산량을 요구하기 때문에 잘 쓰이지 않는다. 여러 모델 가운데 메모리를 적게 필요로 하거나 아예 필요로 하지 않는 비선형 도구들이 존재하며, 메모리가 필요



<그림 1> 시간 변형 선형 시스템 모델



<그림 2> 비메모리 bandpass 비선형 시스템을 위한 AM/AM 모델

없는 도구들은 “제공 연산”과 같이 출력이 현재의 입력에만 관계되는 경우이다. 이러한 형태의 도구들은 복소수형 표현과 잘 부합한다. <그림 2>에 나타난 것처럼 복소수형을 진폭과 위상으로 나눈 뒤, 진폭을 비선형도구를 통하여 통과시키고 다시 변하지 않은 위상의 항과 합친다.

III. 성능 평가

앞서에서도 언급했듯이 컴퓨터 시뮬레이션의 주된 목적은 시스템의 성능을 측정하거나 예상하는 것이다. 여러 종류의 성능 측정 방안이 이 장에서 고려된다.

1. SNR 측정

아날로그 시스템의 성능 측정에 있어서 가장 널리 사용되는 것은 대개의 경우 복조 출력에서의 신호대 잡음비(SNR)이다. SNR의 계산은 일반적으로 대상이 되는 파형, 그리고 이와 비교될 “이상적인” 파형을 필요로 한다. 후자에 해당하는 파형은 종종 원래 파형의 진폭과 시간이 조정된 형태로 나타나 지는데, 이는 진폭 스케일링과 시간 지연이 파형의 왜곡에 영향을 끼치지 않기 때문이다. 대상이 되는 파형은 이상적인 파형과 비교가 되며 대상이 되는 파형에서 이상적인 파형과 직교성을 가지는 성분이 잡음으로 간주된다. 이 경우 SNR은 다음과 같다.

$$SNR = \frac{\rho^2}{1 - \rho^2} \quad (2)$$

윗 식에서  $\rho$ 는 대상이 되는 과형과 이상적인 과형간의 상관 계수를 의미한다.

2. 심볼 에러율 측정 및 Monte Carlo 시뮬레이션  
 디지털 통신 시스템에 있어서 복조 에러  $P_e$ 의 확률은 주요 성능 측정 요소에 속한다. 단순성을 위하여 이 글에서는 이진 통신 시스템만을 고려하여  $P_e$ 를 비트 에러율(BER)로 나타낸다.

Monte Carlo(MC) 방법은 통신 시스템의 BER을 측정하기 위한 방안 중 가장 널리 알려진 방안이다. 분석을 위한 기본 단계로서, 우선 현재 연구 중인 시스템의 기능을 묘사할 수 있도록 시뮬레이션 프로그램이 개발되는데, 시뮬레이션 프로그램은 시스템에 주어지는 과형을 다룰 장비의 모델과 함께 의사 랜덤 데이터와 잡음 데이터를 포함한다. 이때, 여러 심볼들이 시뮬레이션을 통하여 처리되고 발생된 에러를 심볼의 수로 나눔으로서 실험적인 BER이 계산된다. 대부분의 시스템에서 이렇게 얻어진 BER값은 실제 시스템의 BER값과 일관성을 유지하거나 그렇게 큰 차이를 나타내지 않으므로, 일반적으로 MC방안은 모든 통신 시스템에 대하여 사용될 수 있다. MC 방안은 시스템이 입력으로 받을 수 있는 값들과 유사한 신호를 생성할 수 있는데, 이를 통하여 시뮬레이션의 검증에 있어서 매우 중요한 이득을 가질 수 있다. 이와 같이 실험적으로 얻어지는 BER의 신뢰성에 대한 특성으로, MC 측정에서는 복조되는 심볼의 수가 무한대로 증가함에 따라 실험적인 BER이 실제의 BER에 수렴하는 것을 볼 수 있다. 그러나 시뮬레이션은 한정된 심볼 수만을 처리할 수 있으므로, 얼마나 많은 수의 심볼이 신뢰성 있는 BER을 계산하는데 필요한가라는 문제가 발생한다. 이 질문에 대답하기 위하여 신뢰성에 대한 정의가 필요하고, 이를 통한 신뢰도의 계산방법이 요구된다.

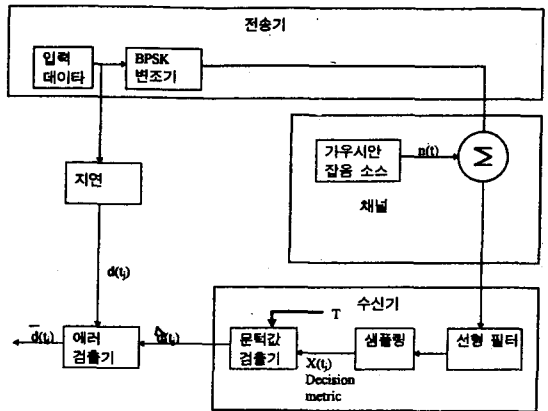
시뮬레이션의 수행시간과 관련하여, 낮은 에러율을 가지는 복잡한 시스템에 대하여 정확한 BER을 얻고자 한다면 MC 접근은 많은 수행시간을 요구

한다. 이를 해결하기 위하여 빠르게 BER을 구하고자 하는 많은 시도들이 제안되었다. 이러한 방안들은 일반적으로 분산 최소화(variance-reduction) 기법으로 불리며, 공통적인 특성은 시스템의 구조나 입력 신호 등에 대한 여러 가지 가정을 기반으로 정확성을 유지하면서 처리하여 심볼의 수를 줄이는 것이다. 일반적으로 시뮬레이션을 위한 가정의 수가 증가할수록 시뮬레이션의 수행시간은 감소한다. 그러나 이러한 방안은 시스템을 잘 이해하는 사람에게는 이득이 되지만 시스템을 잘 이해하지 못하거나 잘못 이해하는 경우에는 잘못된 시뮬레이션 결과를 양산할 수 있다.

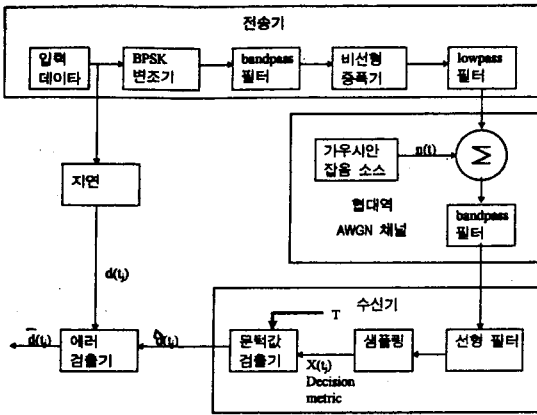
### 3. SA 분석방안

SA(Semi-Analytic) 방안은 시스템 구조와 분석에 대하여 몇 가지 근본적인 사항을 요구하지만, 시뮬레이션의 수행속도가 매우 빠르다는 장점을 제공한다. <그림 3>에 나타난 단순한 형태의 통신 시스템에서 Decision metric은 이미 평균과 분산이 알려진 가우시안 랜덤 변수이므로, 분석자는 복조 에러의 확률을 계산으로 얻어낼 수 있으며, 굳이 시뮬레이션을 수행 할 필요가 없다.

전송부분에 비선형 전력 증폭기를 갖고 있는 <그림 4>의 통신 시스템에서 채널은 WGN(White Gaussian Noise)를 신호에 더하고 그 결과를 선형 필터에 통과시킴으로써 심볼간의 간섭(ISI)을 발



<그림 3> 분석적 접근이 용이한 통신 시스템의 모델



〈그림 4〉 변형된 시스템 모형

생시킨다. 비선형 증폭기와 ISI로 인하여 decision metric은 비가우시안화 되고 BER의 계산은 어려워진다. 이러한 경우에도 decision metric이 조건적으로 가우시안임을 보이는 것은 어려운 것이 아니다. 만약 분석자가 데이터 전송 패턴의 평균값을 데이터 패턴만으로 표현되는 함수로 나타내고, 변위를 잡음 레벨로만 표현되는 함수로 나타낸다면, decision metric은 가우시안으로 나타날 것이다. 이 경우 BER의 계산은 세 가지 요소로 나뉘어진다. 하나는 decision metric의 분산을 구하는 것이고, 다음은 decision metric의 조건적인 평균을 구하는 것이며, 그리고 마지막은 전체 확률 정리를 이용하여 BER을 구하는 것이다.

여타의 다른 시뮬레이션 방안들이 BER을 추정하는 것과 달리 SA 방안은 BER을 계산하는 차이점을 가진다. 이 방안은 컴퓨터 자원을 효율적으로 사용하며, 시뮬레이션이 수행되어 데이터의 평균과 분산이 계산될 때 어떠한 SNR에 대해서도 BER을 계산할 수 있다. 따라서 하나의 시뮬레이션에 있어서 일정한 시스템 잡음 레벨에 대한 BER이 결정될 수 있다. 이러한 장점들에 비하여 단점도 있다. 즉, 조건적인 전송 데이터 패턴에 대하여 시스템의 에러율이 계산되어질 수 있어야 한다는 점이다. 일반적으로 잡음이 비가우시안이거나, 잡음과 데이터가 상관관계를 가질 때, 잡음이 가법적(additive)이지 않거나, 잡음이 시변(non-station-

ary)일 때, 그리고 잡음이 삽입된 후에 비선형성이 발생하는 경우는 이러한 계산이 불가능하거나 매우 어렵다.

비록 SA방안이 효과적으로 사용될 수 있는 시스템들이 많이 있지만, 그 경우에도 효율적인 시뮬레이션 기법들이 요구된다.

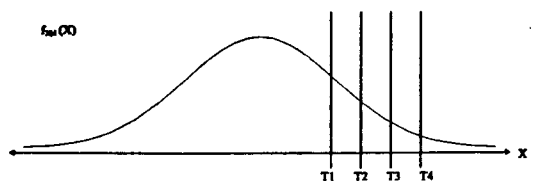
#### 4. IS(Importance Sampling)

비교적 큰 관심을 얻는 기술 가운데 “수정된 Monte Carlo” 혹은 IS(importance sampling) 이론이 있다. 이 방안은 시스템 잡음의 통계치를 이용하여 에러(important event)의 발생 확률을 증대시킴으로써 수행시간을 줄이는 방법이다.

만약 분석자가 효율적인 IS편이(bias)값을 선정한다면 보다 짧은 시간에 정확한 BER을 구할 수 있을 것이다. 그러나 잘못된 편이는 수행시간을 증가시키고 MC보다 느리게 결과를 출력할 수도 있다. 이밖에 편이에 기반한 여러 가지 다양한 방안들이 존재하므로 효율적인 방안의 고려가 필요하다.

#### 5. Tail Extrapolation

BER 추정의 문제는 근본적으로 수치 적분에 대한 문제이다. 이때 주어진 시스템의 BER은 알려지지 않은 pdf 함수의 tail로 볼 수 있다. 이 경우에 있어서 관찰된 데이터와 적절하게 부합되는 pdf를 선정하여야 하며, 이로부터 BER를 추정해낸다. 이것이 tail-extrapolation의 기본 개념이다. 주어진 시뮬레이션에서, 분석가는 〈그림 5〉에 나타난 것과 같이 다중의 문턱(threshold)값을 지정한다. 일반적인 MC 시뮬레이션이 수행되며, 이로부터 문턱값(threshold)을 넘는 횟수가 저장된다. 이때 여러 종류의 pdf들이 사용될 수 있는데, 자주 사용되는 클래스는 일반적인 지수형이다.



〈그림 5〉 Tail extrapolation을 위한 다중 문턱값

주어진 클래스에 관련된 변수들은 얻어진 데이터에 잘 부합되는 pdf를 찾기 위하여 조절된다. 이때 BER은 시스템에 해당하는 pdf의 경계값을 넘는 면적을 수치적으로 계산함으로써 얻을 수 있다. 그러나, 문턱값을 정하는 방법과 어떤 클래스의 pdf를 사용할 것인가에 대한 문제가 발생하며, 대개의 경우 BER 추정치의 정확성을 보장할 만한 신뢰구간을 구한다는 것이 불가능한 경우가 많다.

시스템이 존재하며, 이들에 대한 연구가 광범위하게 연구중이다. 분석가들은 수학적 접근만으로 힘든 복잡한 시스템 기능의 시뮬레이션 수행 시간을 줄이기 위해 많은 가정을 사용한다. 이때 분석가는 또한 시뮬레이션에 의해서 얻어진 결과의 정확성을 판단할 수 있어야 한다. 비록 이러한 문제들이 MC방안에서는 용이하게 풀리지만, 여타의 BER 측정 기술에서는 심각한 문제로 대두된다. <표 1>은 일반적으로 사용되는 시뮬레이션 기술들의 장점과 문제점을 보여주고 있다.

6. 여타의 분산 최소화 방안

극단(extreme) 값에 의한 방법이 효율적인 몇몇

<표 1> BER 계산 기술

항 목 기 술	유 연 성	분석시 어려운 점	수행시간	정 확 도
Monte Carlo	어느 시스템이든 적용 가능	없음	매우 큰편	신뢰구간으로 측정가능
분산 최소화 기술	많은 경우에 적용 가능	Threshold, Biasing 방식 등을 정해야함	다양함 (매우 작은 경우부터 큰경우까지)	접근하기 어려움
Semi-analytic 시뮬레이션	선형 수신기나 AWGN 채널에 한정됨	조건적 BER 통계치가 필요	매우 짧음	DSP 모델의 정확도에 따라 좌우됨
수학적인 분석	이상적인 시스템에 한정됨	수치적인 문제들로 잘 알려진 편임	없음	정확함

7. 부호화된 시스템의 시뮬레이션

부호화된 통신 연결은 큰 규모의 부호화 이득과 함께 Monte Carlo 기술이나 여타의 분산 최소화 방안들이 제공하지 못하는 낮은 에러율을 통하여 정확한 BER을 제공하며, 적당한 시뮬레이션 수행 시간을 필요로 한다. 종종 이러한 시스템의 성능을 측정할 수 있는 방법은 채널을 통하여 전달되는 심볼들의 “raw” 에러율을 계산하는 것이다. 부호화 이론의 가정 및 한계들이 부호화된 시스템의 ‘접대 점’ 성능 측정을 위하여 사용될 수 있다. 이러한 방안은 부호화 이론에 대한 인식을 필요로 하므로 사용시에 주의하여야 한다. 또한, 이때 부호화된 경우와 부호화되지 않은 경우에 에러율이 미치는 영향을 살펴봐야 한다. 예를 들어, 큰 부호화 이득을 얻는 시스템의 경우 부호화되지 않은 시스템에서 측정된 BER의 작은 에러가 부호화된 시스템에서 인정되지 않을 만큼 큰 에러로 나타날 수 있다.

템에서 측정된 BER의 작은 에러가 부호화된 시스템에서 인정되지 않을 만큼 큰 에러로 나타날 수 있다.

8. 검증(Validation)

많은 시뮬레이션 프로그램들이 여러 상황에서 얻어지는 BER을 측정하는 등의 비교적 단순한 결과를 얻기 위하여 개발되고 있다. 시뮬레이션으로 얻어진 결과가 의미를 지니고 사용되기 이전에, 과연 신뢰할 만한 것인가에 대한 평가가 필요하다. 시뮬레이션의 결과가 부정확한 요인은 소프트웨어의 버그와 같은 단순한 요인에서부터 시작하여 정확한 BER을 얻기에는 비교적 적은 데이터량, 모델링시의 잘못 등으로 구분할 수 있다. 특별히 대규모이고 복잡한 시스템에서는 어렵더라도 결과에

대한 검증이 꼭 필요하다.

검증은 원시 프로그램을 중단위로 검토하는 것을 의미하지 않는다. 이런 방식은 대개 시간 소비적이고, 에러를 생성할 수 있고, 실제적으로 불가능하며 모델 정립시의 잘못이나 개념적인 착오 등을 수정할 수 없다. 더구나 패키지를 사용하는 경우 중요 모듈이 원시 프로그램이 공개되지 않은 상태에서 사용되므로 중단위로 이루어지는 수정은 불가능하다.

복잡한 시스템에서는 서브 시스템들이 하나로 통합되기 전에 꼭 검증되어야 한다. 이러한 서브 시스템의 검증이 전체 시스템의 정확성을 보장하는 것은 아니므로, 전체 시스템의 전반적인 동작을 위한 검증이 필요하다. MC 방안에서는 분석가가 특정 지점에서의 신호를 보고, 이에 상응하는 하드웨어 설계의 값과 비교하여 검증할 수 있었다. 그러나 대부분의 BER 측정방안들은 이와 같은 중간 신호를 제공하기가 어렵다. 한편, Perturbation 분석은 시뮬레이션을 검증할 때 유용한 방법이다. 분석가는 시스템을 축소시켜서 논리-분석적으로 추적하기 용이한 형태로 만든다. 즉, 시뮬레이션시에는 작은 정도의 수정만으로 단지 비선형 증폭기를 선형으로 바꾸거나, 송신단과 수신단의 시간 정보를 일치시켜서 동기화하는 것이 용이하므로 시뮬레이션의 BER은 이론적인 결과와 쉽게 비교될 수 있다. 만약, 일치만 한다면 시뮬레이션이 어느 정도는 정확하게 동작하고 있음을 확인할 수 있으므로 분석가는 보다 높은 신뢰도를 갖고 실제의 시스템 분석에 집중할 수 있다.

시스템 성능에 대한 수학적 분석과 시뮬레이션은 상호보완적이다. 분석적 접근은 시뮬레이션의 결과에 대하여 신뢰성을 보장할 수 있고, 시뮬레이션의 결과는 분석에 대한 지침이 될 수 있다. 시뮬레이션 패키지들의 성능이 향상되고 일반화됨에 따라, 이들의 사용에 대한 문제가 제기되고 있다. 예를 들어 두개의 팀이 하나의 문제를 공동으로 다루는 경우, 서로의 코드를 공유하면서 수행하는 것은 어렵다. 하나의 팀내에서도 각 팀원들이 서로의 코드를 공유하기는 어려우며, 따라서 일관된 시뮬레이션 방안으로 접근하는 것이 필요하다. MC 시

뮬레이션의 경우는 비교적 긴 수행시간을 필요로 하지만, 여타의 방안들이 직면하는 문제들은 발생하지 않는다. 따라서, MC 방안으로 시뮬레이션을 수행하고 보다 진보된 방안으로 MC 시뮬레이션의 결과를 검증하는 것도 바람직한 방안이다.

#### IV. 통신 시스템과 신호처리를 위한 시뮬레이션 도구

현재 시뮬레이션은 통신 시스템을 설계하고 분석하며 구현하는데 있어서 중요한 도구로 자리 잡고 있다. 과거에는 시뮬레이션이 주로 시스템의 기능적인 요소를 평가하고, 성능을 측정하고 설계를 위한 사항들을 제정하는데 이용되어 왔고, 위성과 같은 고가격, 고위험성의 시스템들에 대하여 중점적으로 사용되어 왔으나, 현재의 시뮬레이션은 위성통신으로부터 개인휴대서비스에 이르기까지 상업적인 시스템을 설계하고 구현하는데 있어서 훨씬 넓은 분야에서 중요한 위치를 차지하고 있다.

초기의 시뮬레이션 도구들은 “시스템”레벨의 통신 시스템의 설계를 위하여 사용되었으며, 이는 구현적 측면이 배제된 모델로서 구성되었다. 회로와 게이트 레벨의 시뮬레이션 도구들은 독립적으로 구현측면을 집중적으로 다루었으며 이들 두 레벨 간에 차이는 점점 커졌다. 그러나 이러한 차이는 디지털 시스템의 구현에 대한 수요가 증가함에 따라 급격히 줄어들었으며, 앞으로 ESDA(Electronic System Design Automation) 시대의 도구들에서는 시뮬레이션이 중심적인 기능을 수행할 것이며, 시뮬레이션 모델과 실제 구현간의 차이도 최소화될 것이다.

과거 10여년 동안 시뮬레이션 기술은 급격히 성숙하였으며, 설계의 자동화를 위한 추세로 나아가고 있다. 이로써 상위 레벨에서 설계된 시스템이 쉽게 구현적인 레벨까지 연계될 수 있다. 이러한 상황에도 아직까지 RF의 구현이나 광통신 분야에 있어서는 설계와 구현측면간의 차이가 존재하고 있다.

## 1. 모델 생성

현재의 시뮬레이션 도구들은 통신 시스템의 시뮬레이션 모델을 생성하기 위하여 그래픽화된 계층적 블럭 다이어그램 접근법을 제공한다. 점대점 통신 시스템을 위한 시뮬레이션 모델의 경우는 시뮬레이션 소프트웨어에 포함된 라이브러리에서 해당 블럭을 선택함으로써 가능하다. 인코더, 멀티플렉서, 채널, 잡음, 간섭생성, 필터 등 라이브러리에서 제공하는 기능적 블럭들을 라인으로 상호 연결함으로써 신호 흐름 블럭 다이어그램을 그래픽 화면에서 손쉽게 작성할 수 있다. 모델은 일반적으로 하향식(top-down)방식이나 상향식(bottom-up)방식으로 작성될 수 있으며, 일반적으로 하향식방식이 시스템 엔지니어측면에서 많이 사용되는 반면, 하드웨어 설계의 경우는 상향식방식이 널리 사용된다. 계층상의 "leaf"레벨에서 시뮬레이션 모델은 사용되는 의도에 따라 여러 종류의 표현 형태를 가질 수 있는데, "시스템"레벨의 시뮬레이션이 성능평가를 위하여 사용되는 경우에는 "leaf"레벨의 모델이 서브 루틴이나 프로시저로 나타내어진다. 다른 접근방법으로는 신호 처리 알고리즘의 기능적 규격을 기술하는 DFL(Data Flow Language)이나 Silage와 같은 중간계층 언어를 사용하여 모델링을 수행한다. DFL은 DSP 응용을 작성하기 위한 것으로 일반 프로그래밍 언어보다 쉽고 편리한 기능을 제공한다.

모델이 작성되면, 샘플링율, 랜덤 변수를 위한 시드, 시뮬레이션 길이 등과 같은 시뮬레이션 관련 변수들을 구체화하고 파형 레벨의 시뮬레이션을 수행한다. 비트 에러율이나 신호 왜곡 등의 성능 측정치들은 시뮬레이션된 파형에서 얻으며, 설계 변수의 함수로 표현된다. 이때 얻어진 시뮬레이션 결과로부터 분석 및 각종 변수의 최적화 방안을 제시할 수 있다.

## 2. 시뮬레이션 기술

현재의 시뮬레이션 소프트웨어에는 다양한 기술들이 사용되고 있는데 크게 시간 위주, 이벤트 위주, 데이터(시스템) 위주, 혼합 방식의 네 가지로 구분된다. 이들은 시뮬레이션 모델에서 블럭들이

어떻게 호출되는가에 대한 차이를 갖고 있으며, 다양한 정도의 유연성과 시뮬레이션 속도의 차이를 갖고 있다.

(1) 시간 위주의 시뮬레이션 : 가장 단순한 형태로 시뮬레이션 모델내의 각 블럭은 매 시뮬레이션 시간마다 수행되어진다. 전역 시간은 일정 시간마다 증가하고, 모델내의 각 블럭들은 새로운 시간에 의해서 활성화 조건이 만족되면 수행된다.

이 방안은 계산적인 측면에서 매우 비효율적이며, 균등(uniform) 샘플링율을 사용하고 있는데, 모델내의 블럭들이 입력 신호의 하나의 입력 샘플과 함께 수행되어야하는 오버헤드를 가지고 있다. 시뮬레이션의 속도를 향상시키기 위하여 몇 가지 방안들이 제시되고 있는데, 각각의 입력 신호에 대하여 n개의 샘플마다 블럭을 호출하고 n개의 샘플을 출력신호로 발생시키는 벡터 프로세싱이 사용되고 있다. 벡터방식의 프로세싱은 버퍼링을 필요로 하며 피드-백 정보가 없어야 하는 특성을 갖고 있다.

서로 다른 샘플링율의 성분이 합쳐질 때에는 데시메이션(decimation)과 보간(interpolation) 작업을 수행하여 적절한 샘플링율 변환이 수행되도록 한다. 또한, 블럭들의 호출에 대한 순서를 조정하기 위하여 어느 정도의 동적이거나 정적인 스케줄링 방식이 정의되어야 한다.

(2) 이벤트 위주의 시뮬레이션 : 이벤트 위주의 시뮬레이터는 비동기적인 특성으로 인하여 로직 시스템이나 네트워크에 적합한 모델로서 사용되고 있다. 각 블럭들은 새로운 전역 시간과 일관성을 유지하기 위하여 자신의 내부 상태를 갱신하여야 한다. 일반적으로 소수의 블럭들만이 자신의 내부 상태를 변경하기 위하여 활성화되며, 이벤트들간의 inter-arrival time에서는 어떠한 작업도 발생하지 않는다. 그러나, 이 방식은 신호처리를 위한 파형 레벨의 시뮬레이션에서는 느린 속도로 인하여 불합리적이다. 일부의 시스템들에서는 여러 시뮬레이션 모델들의 혼합형이 필요할 수 있으며, 패키지들 가운데에서도 이러한 기능을 지원하는 경우가 있다.

(3) 데이터 위주의 시뮬레이션 : 데이터 위주의



시뮬레이터에서는 시스템으로 유입되는 입력들에 의하여 해당되는 블럭이 호출되는 방식이다. 만약, 입력율 및 출력율이 일정한 블럭이 존재한다면 시뮬레이션의 수행 이전에 고정적으로 스케줄링을 수행할 수 있으며 이를 정적-스케줄링이라고 한다.

한편, 블럭의 입력과 출력이 일정하지 않은 상태로 발생한다면, 정적-스케줄링은 불가능하며 데이터가 발생하는 순간 순간 마다 블럭이 호출되어야 한다. 데이터 항목의 유무는 시뮬레이션의 실행 시간 동안에 검사되어야 하며, 블럭의 활성화에 대한 스케줄링은 동적으로 이루어져야 한다. 시간 위주 방식과 데이터 위주 방식의 차이는 전역 시간의 문제로서 데이터 위주 방식의 경우는 커널에 이러한 전역 시간 정보가 없으며, 대신 블럭들 내부에 지역 시간개념이 존재한다. 이때, Time-semantic은 시뮬레이터가 아닌 시뮬레이터의 사용자에 의해서 주어진다.

### 3. 대화형 시뮬레이션

시뮬레이션의 방법론과 상관없이 대다수의 시뮬레이션 소프트웨어는 대화형 방식을 지원하며 이를 통하여 파형의 변화나 시뮬레이션의 결과를 "one-step" 단위로 관찰할 수 있다. 사용자는 임의의 조건에서 시뮬레이션을 중단시킬 수도 있으며, 특정 변수를 변경하고 수행할 수도 있다. 또한 이 방식은 동적인 시스템을 관찰하여 잘못된 부분을 수정하는 데에도 유용하다.

### 4. 분산형 시뮬레이션

일부의 시뮬레이션 패키지는 규모가 큰 문제를 몇 부분으로 나누어서 다수의 프로세서에서 수행할 수 있는 기능을 지원한다. 일반적으로 문제를 나누는 절차는 사용자에 의해서 이루어지며 이들 간의 상호 통신이나 다수 프로세서에서의 실행은 시뮬레이션 도구에서 자동으로 지원한다. 이 분야는 현재 연구가 활발히 진행되고 있는데 보다 많은 연구가 필요하다.

## V. 시뮬레이션 개발

시뮬레이션은 통신 시스템을 설계하고 분석하기 에 유용한 도구이다. 실제로, 오늘날 대부분의 복잡한 시스템 상황에서 시스템 내부의 기능 및 성능을 예측하고자 할 때 어느 정도의 시뮬레이션은 필수적이다. 이때 시뮬레이션을 개발하고 수행시키는 단계는 다음과 같다.

첫번째 단계는 대상이 되는 시스템의 모델을 만드는 것이다. 일반적으로 이러한 모델은 전체 시스템을 구성하는 서브 시스템들이 블럭 다이어그램으로 연결된 형태를 지닌다. 이때 모델을 작성하는데 사용된 가정들을 정리하는 것이 필요하다. 또한, 각 서브 시스템들에게 중요한 의미를 가지는 변수들을 정리하여 시뮬레이션 수행동안에 사용하는 것도 중요하다.

두번째 단계는 전체시스템 가운데 서브 시스템들이 수행할 기능들을 구체화하는 것이다. 이때 각 서브 시스템을 위한 수학적 모델을 고려하고 어떤 신호를 복소수형으로 표현할 것인가도 선택한다. 아날로그 필터에 대한 디지털 표현도 고려해야 하는데 이 단계에서 사용되는 가정들에 대한 관리도 필요하다.

세번째 단계에서는 시뮬레이션을 통하여 얻어질 출력물을 고려한다. 만약, 전체 시스템의 비트 에러율과 같은 성능 예측이 목적이라면 이를 계산할 수 있는 적절한 방법을 선택한다. 먼저, 모든 에러를 동등하게 취급하고 decision metric에 대하여 가정이 없는 Monte Carlo 방식 뿐만 아니라 decision metric에 다양한 가정을 도입하는 여러 진보된 방안들을 고려한다. 이때 수행시간과 시스템의 성능을 파악하는데 있어서 여러 장단점이 있음을 유의해야 한다.

네번째 단계에서는, 시뮬레이션의 구조를 정립하고, 소프트웨어를 작성한다. 시뮬레이션 전용의 언어를 사용하는 경우에는 제공되는 각종 모델 가운데 하나를 선택하게 되며, 성능 평가를 위한 방법 및 각종 측정 루틴을 포함시킨다. 다음으로 시뮬레이션 결과를 표현하기 위한 각종 그래픽 기법들을

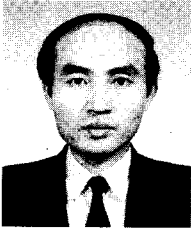
고려한다. 만약 일반 언어로 시뮬레이션 프로그램을 작성하는 경우에는 앞에서 기술한 각종 내용 및 시스템을 표현하기 위한 루틴들을 일일이 작성해야 한다.

다섯째 단계에서는, 작성된 시뮬레이션 프로그램의 결과를 검증한다. 단순한 모델은 에러가 쉽게 수정되어 질 수 있고, 검증하기가 쉬우므로, 복잡한 시스템을 다룰 경우에는 단순화된 모델의 검증부터 시작하는 것이 중요하다. 일단 소규모의 시스템이 검증되면 여기에 부가적인 기능들을 추가함으로써 복잡한 시스템의 형태로 진화시켜 나가면서 검증하는 것이 바람직하다.

#### 참 고 문 헌

- [1] M. Jeruchim, P. Balaban and K. Shanmugan, Simulation of Communication Systems, New York, NY : Plenum Press, 1992.
- [2] Bernard. P and Zeigler, Theory of Modeling and Simulation, Robert. E. Krieger Publishing Company, 1985.
- [3] Bernard. P Zeigler, Multifaced Modelling and Discrete Event Simulation, Robert. E. Krieger Publishing Company, 1984.
- [4] Peter G. Harrison, Performance Modelling of Communication Networks and Computer Architectures, Addison Wesley, 1992.
- [5] M. H. MacDougall, Simulating Computer Systems Techniques and Tools, MIT Press, 1987.
- [6] A. Oppenheim, and R. Schafer, Discrete-Time Signal Processing, Englewood Cliffs, NY, Prentice-Hall, 1989.
- [7] R. Rubinstein, Simulation and Monte Carlo Method, New York, NY : Wiley, 1981.
- [8] M. Jeruchim, "Techniques for Estimating the Bit Error Rate in the Simulation of Digital Communication Systems", IEEE JSAC, Vol SAC-2, no.1 pp.153~171, Jan. 1984..
- [9] K.S. Shanmugan, "An update on software packages for simulation of communication links," IEEE JSAC, vol. SAC-6, no.1, pp.5~14, Jan. 1988.
- [10] W. Wade et.al., "Interactive Communication Systems Simulation Model", IEEE JSAC, vol. SAC-2, no.1, pp.102~129, Jan. 1984.

저자 소개



趙 東 浩

1956年 4月 3日生

1979年 2月 서울대학교 전자공학과 공학사

1981年 2月 한국과학기술원 전기 및 전자공학과 공학석사

1985年 2月 한국과학기술원 전기 및 전자공학과 공학박사

1985年 3月~1987年 2月 한국과학기술원 통신공학연구실 선임연구원

1987年 3月~1989年 12月 한국과학기술원 통신공학연구실 위촉연구원

1987年 3月~현재 경희대학교 전자계산공학과 부교수

1989年 9月~현재 경희대학교 전자계산소 소장

주관심분야 : 멀티미디어 통신 및 서비스, 개인 휴대 통신망 및 프로토콜



李 聖 元

1972年 2月 22日生

1994年 2月 경희대학교 전자계산공학과 공학사

1994年 3月~현재 경희대학교 전자계산공학과 석사과정

주관심분야 : 멀티미디어 통신 및 서비스, 프로토콜