Iowegian's dspGuru™

How to Generate White Gaussian Noise

by Matt Donadio

White Gaussian Noise (WGN) is needed for DSP system testing or DSP system identification. Here are two methods for generating White Gaussian Noise. Both rely on having a good uniform random number generator. We will assume that the function "uniform()" returns a random variable in the range [0, 1] and has good statistical properties. (Be forewarned, though, that many standard random number generators have poor statistical properties.) Once we can generate noise signals with normal distribution, we can adjust the signal's mean and variance to use it for any purpose.

Central Limit Thorem Method

The Central Limit Theorm states that the sum of N randoms will approach normal distribution as N approaches infinity. We can outline an algorithm that uses this approach as:

```
X=0
for i = 1 to N
   U = uniform()
   X = X + U
end

/* for uniform randoms in [0,1], mu = 0.5 and var = 1/12 */
/* adjust X so mu = 0 and var = 1 */

X = X - N/2                 /* set mean to 0 */
X = X * sqrt(12 / N)        /* adjust variance to 1 */
```

When the algorithm finishes, X will be our unit normal random. X can be further modified to have a particular mean and variance, e.g.:

```
X' = mean + sqrt(variance) * X
```

The drawback to this method is that X will be in the range [-N, N], instead of (-Infinity, Infinity) and if the calls to uniform are not truly independent, then the noise will no longer be white. Jeruchim, et. al., recommend N >=20 for good results.

Box-Muller Method

The Box-Muller method uses the technique of inverse transformation to turn two uniformly distributed randoms, U1 and U2, into two unit normal randoms, X and Y.

```
do                         /* if U1==0, then the log() below will blow up */
   U1=uniform()
while U1==0
U2=uniform()

X=sqrt(-2 * log(U1)) * cos(2pi * U2)
```

```
Y=sqrt(-2 * log(U1)) * sin(2pi * U2)
```

We can use some trigonometry to simplify this a bit and get:

```
do
   U1=uniform()            /* U1=[0,1] */
   U2=uniform()            /* U2=[0,1] */
   V1=2 * U1 -1            /* V1=[-1,1] */
   V2=2 * U2 - 1           /* V2=[-1,1] */
   S=V1 * V1 + V2 * V2
while S >=1

X=sqrt(-2 * log(S) / S) * V1
Y=sqrt(-2 * log(S) / S) * V2
```

The above is called the Polar Method and is fully described in the Ross book [Ros88].

X and Y will be unit normal random variables (mean = 0 and variance = 1), and can be easilly modified for different mean and variance.

```
X' = mean + sqrt(variance) * X
Y' = mean + sqrt(variance) * Y
```

References:

- [Jer92] Simulation of Communication Systems by M. C. Jeruchim, P. Balaban and K. S. Shanmugan
- [Ros88] A First Course on Probability by S. M. Ross

Home | Up | Contents | Full

mailto:webmaster@iowegian.com