Tune-Up Tuesday #4 Deconvolution for October 9, 2025

Before we talk about deconvolution, let's define convolution. Then, we'll derive a deconvolution algorithm and apply it to two examples. You could have worked either example for the Tune-Up.

Convolution. For a finite impulse response filter with $M + 1$ filter coefficients $b_0, b_1, \dots, b_M$, the output signal $y[n]$ for an input signal $x[n]$ is computed according to

$$y[n] = b_0\, x[n] + b_1\, x[n-1] + \cdots + b_M\, x[n-M]$$

If we input the discrete-time impulse signal $\delta[n]$, which has value of 1 at $n = 0$ and 0 otherwise, the output is called the *impulse response* (response is synonymous with output):

$$h[n] = b_0\, \delta[n] + b_1\, \delta[n-1] + \cdots + b_M\, \delta[n-M]$$

Hence,

$$h[k] = b_k \text{ for } k = 0, 1, \dots, M$$

Otherwise, $h[k] = 0$. Because $h[k] = b_k$ for $k = 0, 1, \dots, M$,

$$y[n] = h[0]x[n] + h[1]x[n-1] + \cdots + h[M+1]x[n-(M+1)] = \sum_{k=0}^{M} h[k]\, x[n-k]$$

This computation is known as convolution of $h[n]$ and $x[n]$:

$$y[n] = h[n] * x[n]$$

Given input signal $x[n]$ and impulse response $h[n]$, we can compute output signal $y[n]$. If $x[n]$ is also finite in length, then the length of $y[n]$ will be the length of $h[n]$ plus the length of $x[n]$ minus 1.

Deconvolution. Whereas convolution computes the output signal $y[n]$ from input signal $x[n]$ and an impulse response $h[n]$ of a FIR filter, deconvolution seeks to find impulse response $h[n]$ given input signal $x[n]$ and output signal $y[n]$. We can choose the input signal $x[n]$, also known as a test signal, and observe the output signal.

Practical scenario. We would start the test signal and the observation at a particular point in time, which we'll say is at $n = 0$ without loss of generality. Further, we will assume that $x[n] = 0$ for $n < 0$; i.e., $x[n]$ is a causal signal.

Deconvolution Algorithm. We'll work backwards in the time domain to compute the FIR filter coefficients. We derive a time-domain deconvolution algorithm by first evaluating the output at $n = 0$:

$$y[0] = h[0]\, x[0] + h[1]\, x[-1] + h[2]\, x[-2] + \cdots + h[M]\, x[-M]$$

As mentioned above, we'll assume $x[n]$ is a causal signal; i.e., $x[n] = 0$ for $n < 0$. Since we know $x[n]$ and $y[n]$, we have one equation and one unknown at $n = 0$:

$$y[0] = h[0]\, x[0]$$

and we can compute

$$h[0] = \frac{y[0]}{x[0]}$$

For this calculation to be valid, the first value of the test signal, $x[0]$, cannot be zero.

Second output: $y[1] = h[0]\,x[1] + h[1]\,x[1]$, and therefore, $h[1] = \frac{\mathrm{y}[1] - \mathrm{h}[0]\,x[1]}{x[0]}$.

Third output: $y[2] = h[0]\,x[2] + h[1]\,x[1] + h[2]\,x[0]$ and $h[2] = \frac{\mathrm{y}[2] - \mathrm{h}[0]\,x[2] - h[1]\,x[1]}{x[0]}$

In general, for the Nth output, $h[N] = \frac{y[N] - \sum_{i=0}^{N-1} h[i]\,x[N-i]}{x[0]}$.

The MATLAB script utdeconvolve.m implements this algorithm.

Example. *Problem 4.3(b).* In this problem, we're given

- causal input signal $x[n]$ with non-zero values [1 2 3 4 5]
- causal output signal $y[n]$ with non-zero values [1 1 1 1 1 -5]

We can compute the FIR filter coefficients using the above deconvolution algorithm:

$$h[0] = \frac{y[0]}{x[0]} = \frac{1}{1} = 1$$

$$h[1] = \frac{\mathrm{y}[1] - \mathrm{h}[0]\,x[1]}{x[0]} = \frac{1 - 1 \cdot 2}{1} = -1$$

$$h[2] = \frac{\mathrm{y}[2] - \mathrm{h}[0]\,x[2] - h[1]\,x[1]}{x[0]} = \frac{1 - 1 \cdot 3 - (-1) \cdot 2}{1} = 0$$

The values of $h[n]$ for $n > 2$ are zero. The MATLAB script utdeconvolve.m will give the same answer for $h[n]$. We can validate the answer by convolving $h[n]$ and $x[n]$. We can use the Matlab command `conv` to do this:

```
y = conv( [1 -1], [ 1 2 3 4 5 ] )

y =

     1     1     1     1     1    -5
```

Alternately, we could use the filter command. Keeping in mind that the filter command produces as many output samples as there are input samples,

```
y = filter( [1, -1], 1, [ 1 2 3 4 5 0 ] )
y =

     1     1     1     1     1    -5
```

When convolving two finite-length signals $x[n]$ and $h[n]$, the result $y[n]$ has finite length. The length of $y[n]$ is the length of $x[n]$ plus the number of filter coefficients minus 1. Since the length of $y[n]$ is 6 and the length of $x[n]$ is 5, there are 2 filter coefficients.