Wavelets and Filter Banks

Harsha Vardhan Tetali

November 13, 2018

1 Introduction

When we sample a signal, we convert a signal in continuous time to a sequence of numbers. We sample a signal using an impulse train and this is enforced mathematically by the Sampling theorem for Band-limited signals. Let us try to attack the same problem of sampling from a different view-point - the view point of approximating the signal of interest using a piece-wise constant function. Let us say, we have a function $f(t)$, in time as below:

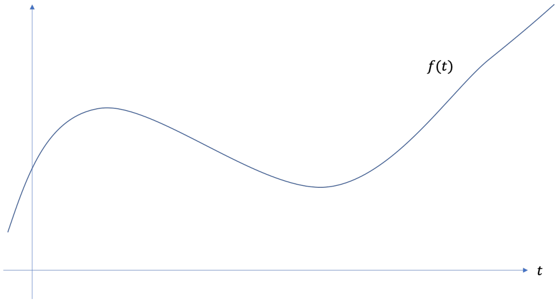


Figure 1: A Signal varying in Time

Now let us try to approximate it with a piece-wise constant function which is constant over an interval T. One possible obvious choice would be something like below:

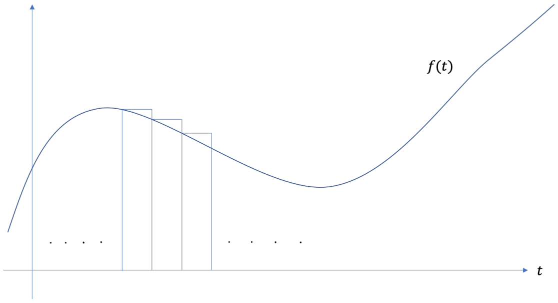


Figure 2: A Possible Approximation with Piecewise Continous Functions

Now let us try to introduce a sense in which we want the signal to be an approximation to the original signal. We want the signal to be the closest to the original signal in a way

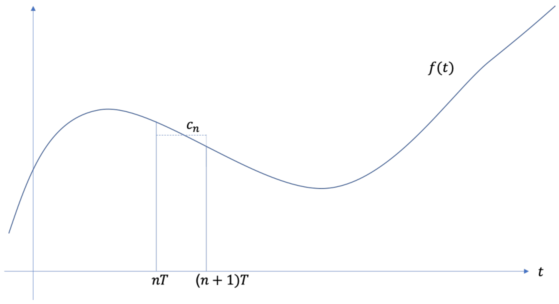


Figure 3: Approximation in the Minimum Integrated Squared error sense

that the integral of the squared difference over the interval is minimized. Let us say, we approximate the signal $f(t)$ in the region $[nT,(n+1)T]$ with a constant c_n.

So to find c_n, we need to minimize the integral:

$$\int_{nT}^{(n+1)T} (f(t)-c_n)^2\,dt$$

Let us call this Q_n, thus:

$$Q_n = \int_{nT}^{(n+1)T} (f(t)-c_n)^2\,dt \tag{1}$$

We can set the derivative of Q_n with respect to c_n to be zero, and find the minimizer, as we can see that the expression is quadratic in c_n, it has a unique minimizer. Thus, we first find the derivative:

$$\frac{dQ_n}{dc_n} = \frac{d}{dc_n}\int_{nT}^{(n+1)T} (f(t)-c_n)^2\,dt = \int_{nT}^{(n+1)T} 2\,(f(t)-c_n)\,dt \tag{2}$$

Setting the above to zero, we find:

$$c_n = \frac{1}{T}\int_{nT}^{(n+1)T} f(t)dt \tag{3}$$

Hence we found the piece-wise constant function which can best approximate the function $f(t)$ over the interval $[nT,(n+1)T]$. Now this can be repeated on every interval and we finally get a piece-wise constant function.

Now if we look at the value of Q_n attained at this value of c_n:

$$Q_n = \int_{nT}^{(n+1)T} \left(f(t) - \frac{1}{T}\int_{nT}^{(n+1)T} f(z)dz\right)^2 dt \tag{4}$$

The above can easily be simplified to:

$$Q_n = \int_{nT}^{(n+1)T} f^2(t)dt - \frac{1}{T}\left(\int_{nT}^{(n+1)T} f(t)dt\right)^2 \tag{5}$$

Observe that Q_n is always non-negative because it is the integral of a squared term and also,

$$0 \leq Q_n \leq \int_{nT}^{(n+1)T} f^2(t)dt \tag{6}$$

If we are considering continous and bounded functions, we can have:

$$0 \leq Q_n \leq B^2 T \tag{7}$$

where B is an upper bound of the the function $f(t)$.

The above inequality imposes a control on Q_n, if we make T arbitrarily small, we can make the error arbitarily small too. This means that we can make the error as small as we please.

Observe that, what we are doing now is contrasting to what we did with Fourier Series. In Fourier Series (or Fourier Transform in General), we were trying to build any function with continuous functions, here in this, we are trying to build a bounded function using discontinuous functions and we are going arbitrarily close to it in the integral squared error sense. Now why is this any interesting? We will now try to pursue this question. Before we pursue this question, we must take note of certain things we have seen in all the above discussion:

1. We have sub-consciously introduced a scaling function:

$$\phi(t) = \begin{cases} 1 & 0 \leq t \leq 1 \\ 0 & \text{otherwise} \end{cases} \tag{8}$$

2. We approximated the function $f(t)$ as a sum of scaled and shifted scaling functions.

2 Moving from one Resolution to Another

Let us say we are given a discrete sequence of numbers obtained from a function $f(t)$ using the procedure defined in the above section with the time interval of integration as T. From this sequence of numbers, can we obtain a different sequence of numbers, namely the ones which are obtained by integrating with the time interval of integration $2T$. Let this new sequence be called d_n.

Observe that, we can write:

$$d_n = \frac{1}{2T} \int_{nT}^{(n+2)T} f(t)dt \tag{9}$$

Now observe that, we can write:

$$d_n = \frac{1}{2} \left[\frac{1}{T} \int_{nT}^{(n+1)T} f(t)dt + \frac{1}{T} \int_{(n+1)T}^{(n+2)T} f(t)dt \right] = \frac{c_n + c_{n+1}}{2} \tag{10}$$

Using our usual convention, we get the above. So, it is easy to observe that, we can reduce the resolution of our signal just by the process of averaging. This will be true of any time interval kT (we will only need to average over the k different coefficients).

Now let us restrict ourselves to the case of $2T$, and ask the question: can we get back c_k's from d_k's, obviously the answer is no! This is because, we have two unknowns, c_n and c_{n+1} and only one equation $d_n = \frac{c_n + c_{n+1}}{2}$.

To recover the original signal, we need two equations, so let us try to come up with one more. Let us say, we also somehow have the value of $\frac{c_n - c_{n+1}}{2}$, lets call this e_n. So we therefore also have:

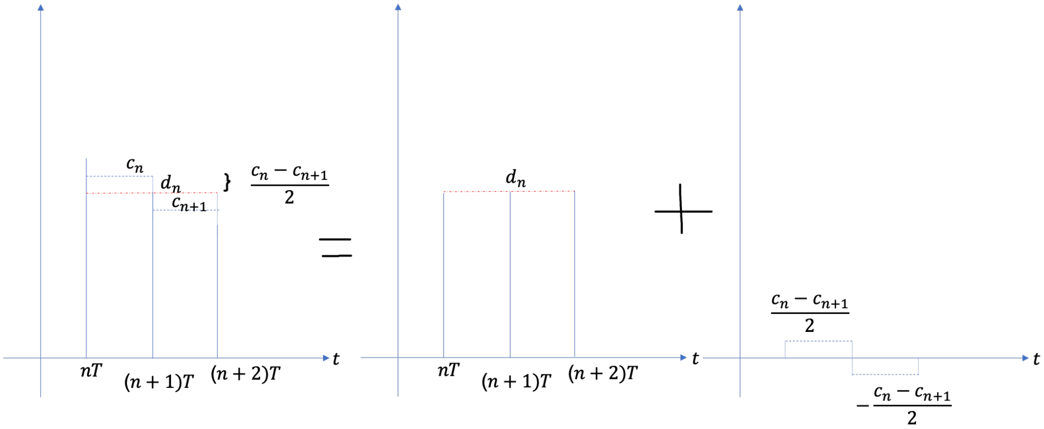
$$e_n = \frac{c_n - c_{n+1}}{2} \tag{11}$$

With this information at hand, we can see that:

$$c_n = d_n + e_n \tag{12}$$

$$c_{n+1} = d_n - e_n \tag{13}$$

So let us try to visualize this on a picture to see what can be inferred from all this:



c_n
d_n
c_{n+1}
$\frac{c_n - c_{n+1}}{2}$
nT
$(n+1)T$
$(n+2)T$
t
=
d_n
nT
$(n+1)T$
$(n+2)T$
t
+
$\frac{c_n - c_{n+1}}{2}$
t
$-\frac{c_n - c_{n+1}}{2}$

Figure 4: The first one can be seen as the sum of the second and the third

From all this, we conclude that, we need another function to move from one resolution to another. In our case, we need a function like:

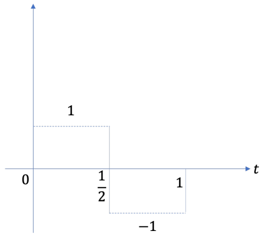


Figure 5: Mother Wavelet

3 Filter Banks

Now lets try to build a system which is inspired from the above to convert the signal into a lower resolution signal and reconvert it back to its own resolution. As we see from above, the first step is to create the sequences d_n and e_n. We will work with the sequence $x[n]$ and we will assume causal systems, therefore, we make slight changes and create the new sequences as:

$$a[n] = \frac{1}{2}\left(x[n] + x[n-1]\right) \tag{14}$$

$$b[n] = \frac{1}{2}\left(x[n] - x[n-1]\right) \tag{15}$$

If $A[n]$ and $B[n]$ are the system functions, then we can easily see that:

$$A[n] = \frac{1}{2}\left(1 + z^{-1}\right) \tag{16}$$

$$B[n] = \frac{1}{2}\left(1 - z^{-1}\right) \tag{17}$$

Now observe that there is no loss in information, when we downsample the signal by a factor of 2. This is because:

$$x[2n] = a[2n] + b[2n] \tag{18}$$

and

$$x[2n-1] = a[2n] - b[2n] \tag{19}$$

and we do not need $a[2n-1]$ and $b[2n-1]$. Therefore, we created two sequences from the original sequence without any loss of information. Let us show this in a picture:

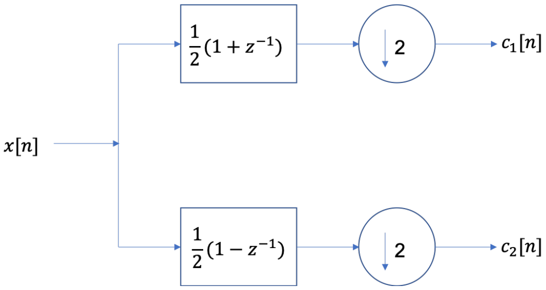


Figure 6: Analysis Part of Filter Bank

Where,

$$c_1[n] = a[2n] \tag{20}$$

$$c_2[n] = b[2n] \tag{21}$$

Now let us try to synthesis the signal $x[n]$ from $c_1[n]$ and $c_2[n]$.
We have already seen:

$$x[2n] = a[2n] + b[2n] = c_1[n] + c_2[n] \tag{22}$$
$$x[2n-1] = a[2n] - b[2n] = c_1[n] - c_2[n] \tag{23}$$

Let us first try to first implement $c_1[n] + c_2[n]$ and $c_1[n] - c_2[n]$:

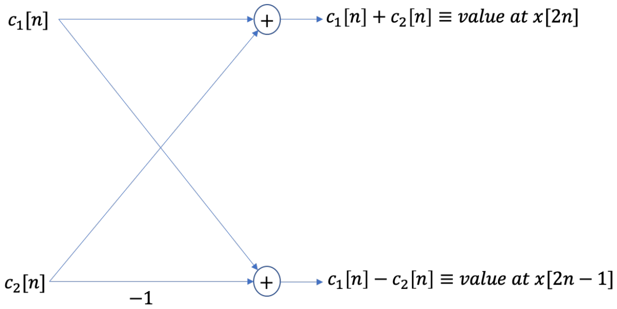

$c_1[n]$
+
$c_1[n] + c_2[n] \equiv value\ at\ x[2n]$
$c_2[n]$
−1
+
$c_1[n] - c_2[n] \equiv value\ at\ x[2n-1]$

Figure 7: The Sum and Difference Operations

Now to construct the signal $x[n]$ back, we need to bring these back to the original rate, therefore, we now upsample the two signals, thus we will have something like this:

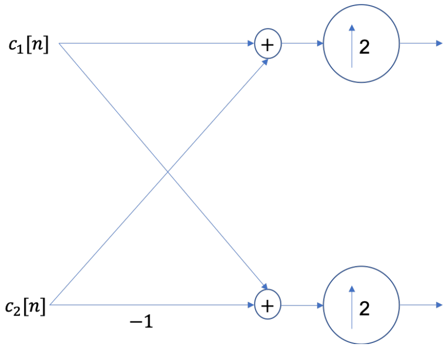


Figure 8: The Sum and Difference Operations with the upsampler in place

Before we add the two signals, we want to delay the difference branch by one unit so that we accomodate for $x[2n-1]$. Thus we have:

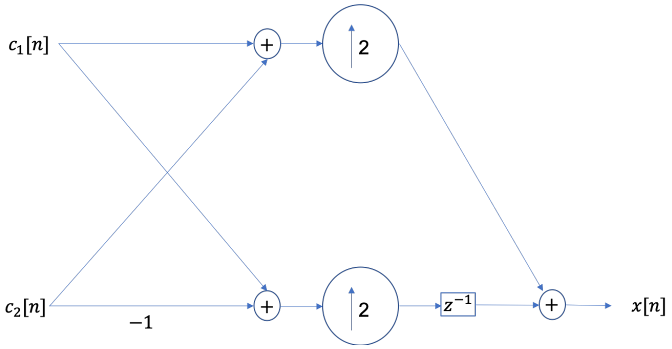


Figure 9: The Synthesis part

Now it turns out, we can interchange the upsampling operation with the sum and difference portion; which leads to the following:

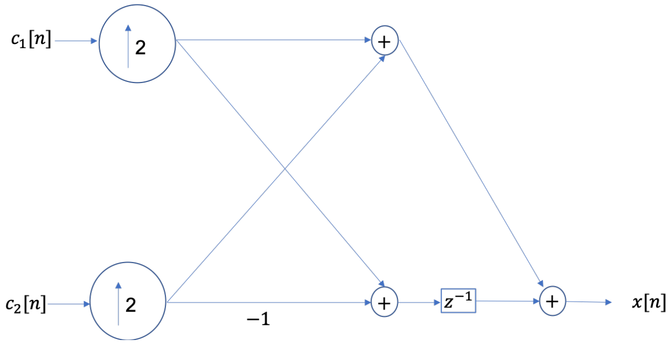


Figure 10: Exchanging Sum and Difference portion with the Upsampling portion

Now, the part after the upsamplers can be simplified and two corresponding system functions can be obtained:

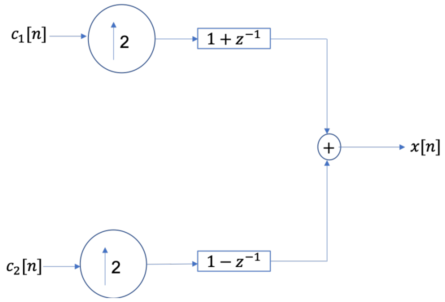


Figure 11: The Synthesis Part