In this example, an input discrete time signal $x[n]$ (representing samples of a continuous time signal $x(t)$ at $T = 1/4$ sec) is converted to $x_b[n]$, representing samples of a continuous time signal $x(t)$ with sampling period $T = 1/6$ sec. The block diagram in Figure 1, shows the processing steps involved.

Let $x(t) = \cos(2\pi t)$ with sampling period $T_s = 0.25$ sec, and $x[n] = \cos(\pi n/2)$.

The CTFT for the sampled signal is calculated in the usual way:

$$x_\delta(t) = \sum_{n=-\infty}^{\infty} x(nT_s)\delta(t-nT_s)$$

The CT spectrum for the sampled signal is:

$$X_\delta(j\omega') = X(j\omega') \ast \frac{1}{T_s} \sum_{k=-\infty}^{\infty} \delta(\omega' - k\frac{2\pi}{T_s})$$

or

$$X_\delta(j\omega') = \sum_{n=-\infty}^{\infty} x(nT_s) e^{-j\omega'nT_s}$$

To calculate the normalized DTFT, let $\omega = \omega'T_s$, then

$$X(e^{j\omega}) = \sum_{n=-\infty}^{\infty} x[n] e^{-jn\omega} = X_\delta\left(\frac{j\omega}{T_s}\right)$$

1 Upsample

To upsample $x[n]$ by a factor of 3, two extra samples are added in between every original sample, thus $x_p[0] = x[0], x_p[3] = x[1], x_p[6] = x[2], ...$, then:

$$X_p(e^{j\omega}) = \sum_{n=-\infty}^{\infty} x_p[n] e^{-jn\omega}$$

(1)

$$= ... + x_p[0]e^{-j0} + x_p[3]e^{-j3\omega} + x_p[6]e^{-j6\omega} + ...$$

(2)

$$= ... + x[0]e^{-j0} + x[1]e^{-j3\omega} + x[2]e^{-j6\omega} + ...$$

(3)

Thus $X_p(e^{j\omega}) = X(e^{j3\omega})$. (See Fig. 2.)
An interpolation filter \( h[n] \) is used to fill in intermediate values, with

\[
h[n] = 3 \frac{\sin(n\pi/3)}{\pi n}.
\]

Note that \( h[0] = 1 \) so that the original sample points will have the original heights, i.e. \( x_u[0] = x[0] = x_p[0] \). The spectrum of the interpolated signal is \( X_u(e^{j\omega}) = X_p(e^{j\omega})H(e^{j\omega}) \).

The resulting spectrum \( X_u(e^{j\omega}) \) is equivalent to the DTFT of the original cosine sampled at \( T_u = 1/12 \) sec.

2 Down sample

To down sample by a factor of 2, first every other sample is set to zero (giving \( x_d[n] \)), and finally half the samples are discarded, giving \( x_b[n] \). Here, the final output \( x_b[n] \) is equivalent to sampling the original cosine at \( T_d = 1/6 \) sec.

Subsampling \( x_u[n] \) by 2 is equivalent to multiplying by a pulse train \( p[n] = \sum_{k=-\infty}^{\infty} \delta[n - 2k] \). The DTFT spectrum is calculated from the circular convolution:

\[
p[n] \cdot x_u[n] \rightarrow_{\text{DTFT}} \frac{1}{2\pi} \int_{-\pi}^{\pi} P(e^{j\theta})X_u(e^{j(\omega-\theta)})d\theta = X_d(e^{j\omega})
\]

where \( P(e^{j\omega}) = \sum_{k=-\infty}^{\infty} \pi \delta(\omega - k\pi) \).

Finally, by discarding odd samples, \( x_b[n] = x_d[2n] \) and hence \( X_b(e^{j2\omega}) = X_d(e^{j\omega}) \). The figure shows how each frequency component is scaled in frequency.
Figure 2. Time and frequency plots for combined up/down sample example.