MULTIRATE FILTERING: HOW IT MAKES DSP EASIER

GIVEN: Signal bandlimited to $10kHz$, sampled at $f_{Nyquist} = 2(10kHz) = 20kHz$.

WANT: To low-pass-filter (LPF) to $100Hz$.

SPECS: $H(f) = \begin{cases} 1, & \text{if } 0 < |f| < 95Hz; \\ any, & \text{if } 95Hz < |f| < 100Hz; \\ 0, & \text{if } 100Hz < |f| < 10kHz. \end{cases}$

Note $\frac{transition\,width}{maximum\,freq} = \frac{5}{10^4}$ →sharp filter.

IDEA:

1. After LPF, can $\downarrow 100$ (since $\frac{10kHz}{100Hz} = 100$). Recall LPF followed by $\downarrow 100$ is decimation.
2. If decimate in stages, can use shorter filters →much less computation and storage.
3. For example, decimate by 10 twice:

FIRST STAGE DECIMATION:

$\rightarrow H_1(f) \rightarrow \downarrow 10 \rightarrow$ where

$$H_1(f) = \begin{cases} 1, & \text{if } 0 < |f| < 95Hz; \\ any, & \text{if } 95Hz < |f| < 1905Hz; \\ 0, & \text{if } 1905Hz < |f| < 10kHz. \end{cases}$$

1. $\frac{transition\,width}{maximum\,freq} = \frac{1810}{10^4}$ →dull filter.
2. $1905 = \frac{20kHz}{10} - 95; \quad 1810 = 1905 - 95$

1. Dull filter→short-length FIR filter→ relatively little computation and storage.
2. There is much aliasing, BUT:
3. $0 < |f| < 95Hz$ IS NOT ALIASED!
   And $0 < |f| < 95Hz$ is all we care about!
   WE DON'T CARE about aliasing the rest!

SECOND STAGE DECIMATION:

$\rightarrow H_2(f) \rightarrow \downarrow 10 \rightarrow$ where

$$H_2(f) = \begin{cases} 1, & \text{if } 0 < |f| < 95Hz; \\ any, & \text{if } 95Hz < |f| < 100Hz; \\ 0, & \text{if } 100Hz < |f| < 1kHz. \end{cases}$$

1. These are same specs as original filter? Where are the computational savings?
2. Filter operates on DOWNSAMPLED signal: Recall $\rightarrow\downarrow 10 \rightarrow H_2(z) \rightarrow$ is equivalent to $H_2(z^{10}) \rightarrow\downarrow 10$ (only need every 10^{th} output)
3. $\frac{transition\,width}{maximum\,freq} = \frac{5}{10^3}$ →duller than before.
4. Break up into many stages→greater savings
5. Used in digital audio tape, PC sound cards, etc. so don't need sharp *analog* antialiasing filter.