Digital signals/Frequency domain

Frequency domain

- **discrete Fourier transform**, time domain → frequency domain

$$X_k = \sum_{i=1}^{N} x_i \, \mathrm{e}^{-2\pi\mathrm{i}\frac{(i-1)(k-1)}{N}} \in \mathbb{C} \quad \text{with} \quad k \in \{1, \ldots, N\}$$

```
>> Xk = fft( xi ) / N; % complex Fourier coefficients
```

- k is a **frequency index** (as i was a time index)

$$k \to f_k = \frac{k-1}{N} f_\mathrm{S}$$

```
>> fk = (0:N-1) / N * fS; % frequency values
```

- frequencies beyond Nyquist frequency are **negative frequencies**

$$f_k \to \begin{cases} f_k - f_\mathrm{S} & \text{if } f_k > f_\mathrm{Ny} \\ f_k & \text{otherwise} \end{cases}$$

```
>> fk(fk > fNy) = fk(fk > fNy) - fS; % imply negative frequencies
```

- **power spectral density** (also known as **power spectrum**)

$$P_k = |X_k|^2 \in \mathbb{R} \quad \Leftarrow \quad \sum_{k=1}^{N} P_k = P$$

```
>> Pk = abs( Xk ) .^ 2; % power spectral density
```

- **real valued signals** ($x_i \in \mathbb{R}$) imply a special symmetry

$$X_{f_k} = X^*_{-f_k} \quad \Rightarrow \quad P_{f_k} = P_{-f_k}$$

- restrict to **one-sided spectrum**

```
>> Pk(fk < 0) = []; % remove negative frequency components
>> Xk(fk < 0) = [];
>> fk(fk < 0) = [];
>> Pk(2:end) = 2 * Pk(2:end); % rescale to match total power
>> Xk(2:end) = sqrt( 2 ) * Xk(2:end);
```

- P_1 is **DC offset**, $P_{k>1}$ are **contributions of sines** with frequencies $f_{k>1}$

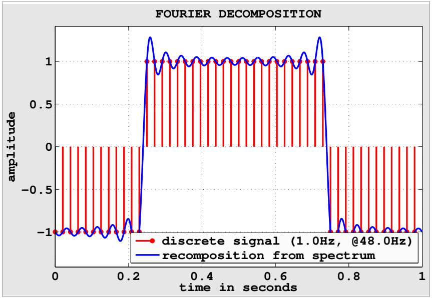
$$x(t) = \sqrt{P_1} + \sqrt{2} \sum_{k>1} \sqrt{P_k} \sin(2\pi f_k t)$$

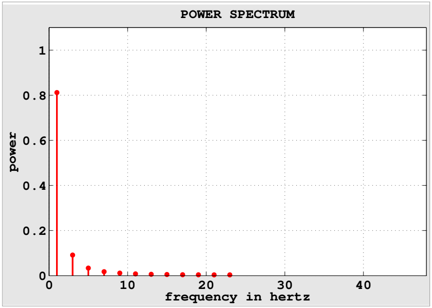
Frequency domain

- complex valued but without loss of phase information

$$x(t) = X_1 + \sqrt{2} \sum_{k>1} X_k e^{2\pi i f_k t}$$

- **example:** `matlab/fdomain.m`





- **exercise:**
  - examine spectra of different wave forms (sines, square, sawtooth, ...)
  - examine spectral **frequency range**
  - verify loss of **phase information** in (real valued) power spectra