

Various signal sampling and reconstruction methods

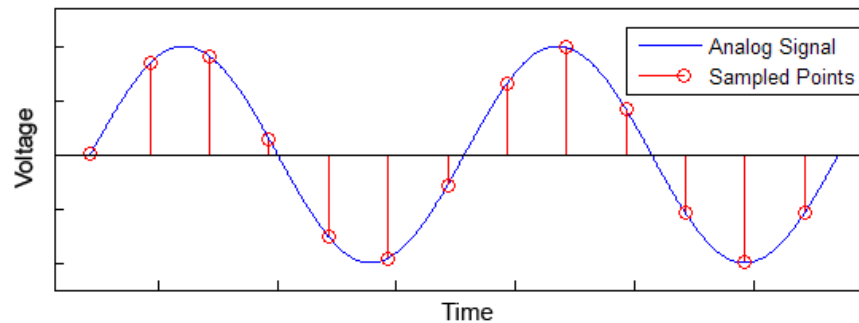
Rolands Shavelis, Modris Greitans
Institute of Electronics and Computer Science
14 Dzerbenes str., Riga LV-1006, Latvia

Contents

Classical uniform sampling and reconstruction
Advanced sampling and reconstruction methods
Signal-dependent sampling and reconstruction
Conclusions

Classical uniform sampling

- Data acquisition systems capture real-world signals and convert them to digital signals

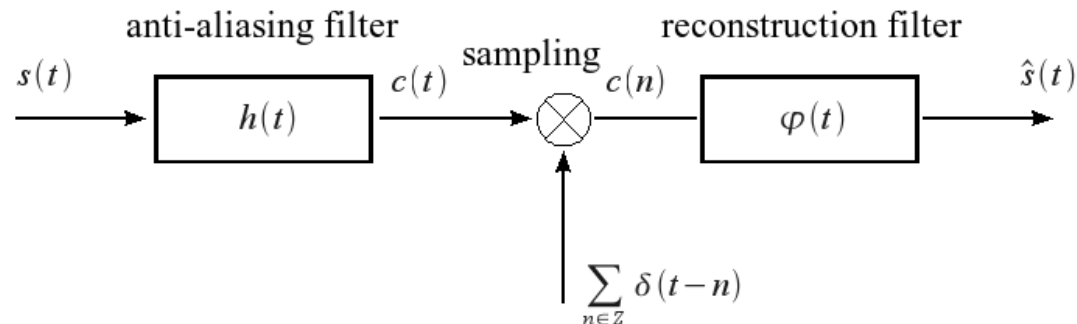


- **Shannon's sampling theorem:** if a function $s(t)$ contains no frequencies higher than f_{max} (in cycles per second), it is completely determined by giving its ordinates at a series of points spaced $T = 1/(2 f_{max})$ seconds apart
- The corresponding reconstruction formula

$$s(t) = \sum_{n \in \mathbb{Z}} s(nT) \operatorname{sinc}\left(\frac{\pi t}{T} - n\pi\right), \quad \operatorname{sinc}(t) = \frac{\sin(t)}{t}$$

Classical uniform sampling

- When the input signal $s(t) \in L_2$ is not band-limited, the standard procedure is to apply a low-pass filter prior to sampling in order to suppress aliasing



- In the traditional approach $h(t) = \varphi(t) = \text{sinc}(\pi t)$, but in general the analysis and synthesis functions $h(t)$ and $\varphi(t)$ are not identical

Classical uniform sampling

- Every band-limited signal $s(t) \in V$ with $T=1$ can be expressed as

$$s(t) = \sum_{n \in \mathbb{Z}} s(n) \operatorname{sinc}(\pi(t-n))$$

- If $\operatorname{sinc}(\pi t)$ is replaced by a more general template $\varphi(t)$, the generalization to other classes of functions is achieved
- The basic approximation space

$$V(\varphi) = \left\{ s(t) = \sum_{n \in \mathbb{Z}} c(n) \varphi(t-n) : c \in l_2 \right\}$$

- In order to represent a signal $s(t) \in L_2$ in the space $V(\varphi)$ the minimum-error approximation of $s(t)$ into $V(\varphi)$ is made

$$\hat{s}(t) = \sum_{n \in \mathbb{Z}} c(n) \varphi(t-n),$$

where the coefficients $c(n)$ ensure the minimum error

$$\int_{-\infty}^{\infty} |s(t) - \hat{s}(t)|^2 dt$$

Classical uniform sampling

- Examples of generating functions

- Band-limited functions

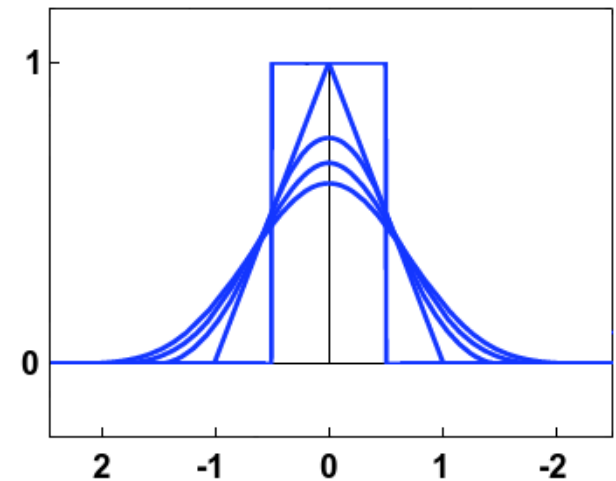
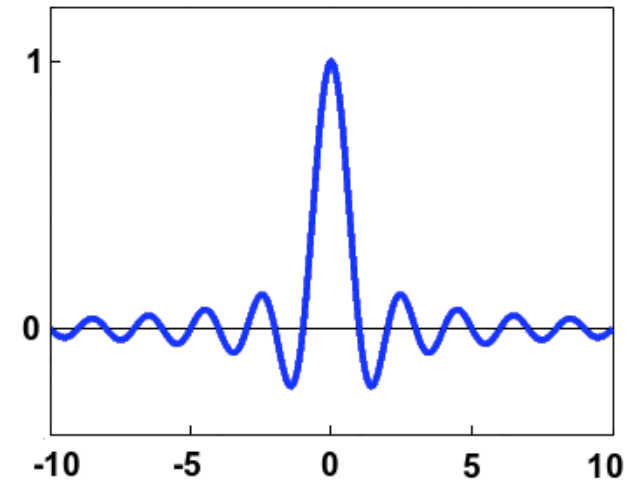
$$\varphi(t) = \text{sinc}(\pi t)$$

- Piecewise-constant functions

$$\varphi(t) = \beta^0(t) = \begin{cases} 1, & |t| \leq 1/2 \\ 0, & |t| > 1/2 \end{cases}$$

- Polynomial splines of degree $n \geq 1$

$$\varphi(t) = \beta^n(t) = \beta^0(t) * \beta^{n-1}(t)$$

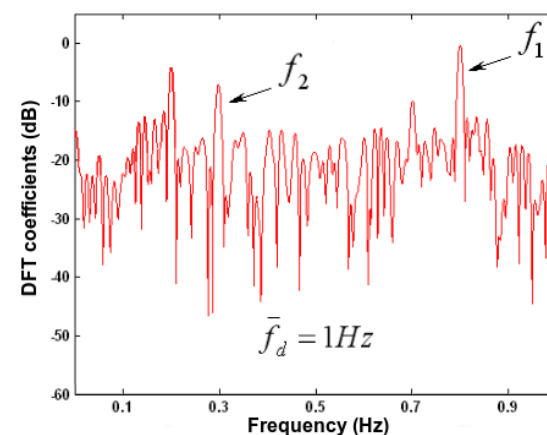
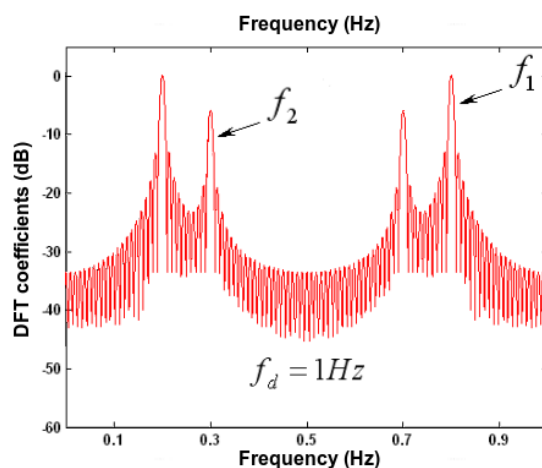
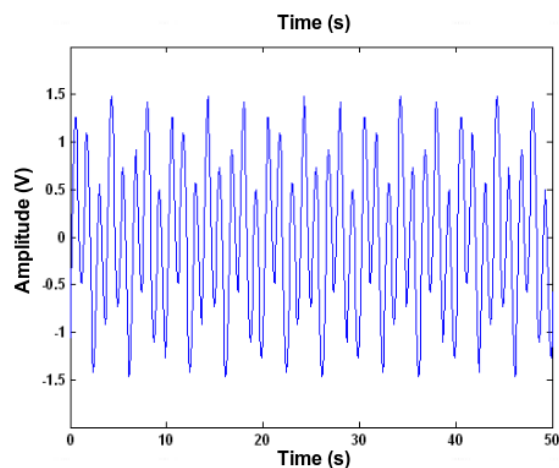


Advanced sampling and reconstruction methods:

- **Amplitude encoding** (information is carried by signal samples s_n):
 - Digital alias-free signal processing
 - Compressive sensing
 - Finite rate of innovation
- **Time encoding** (information is carried by sampling instants t_n):
 - Time encoding machines (signal-dependent sampling)

Digital Alias-free Signal Processing

- Digital Alias-free Signal Processing (DASP) is a set of methodologies that use non-uniform sampling and specialised signal processing algorithms that allow processing signals with unknown spectral support in the frequency ranges exceeding half the average sampling rate

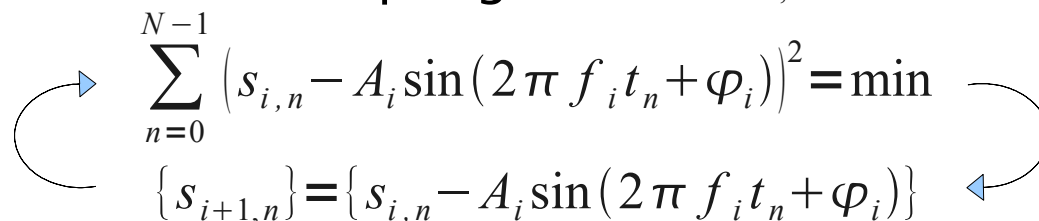


Digital Alias-free Signal Processing

- A priori information: signal consists of one or more discrete spectral components

$$s(t) = \sum_{m=1}^M A_m \sin(2\pi f_m t + \varphi_m)$$

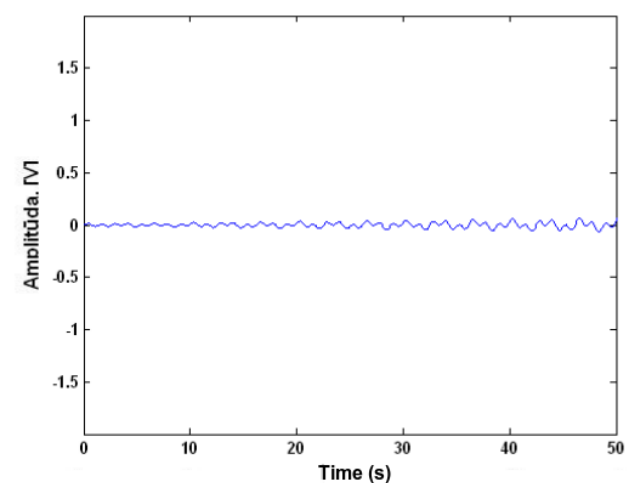
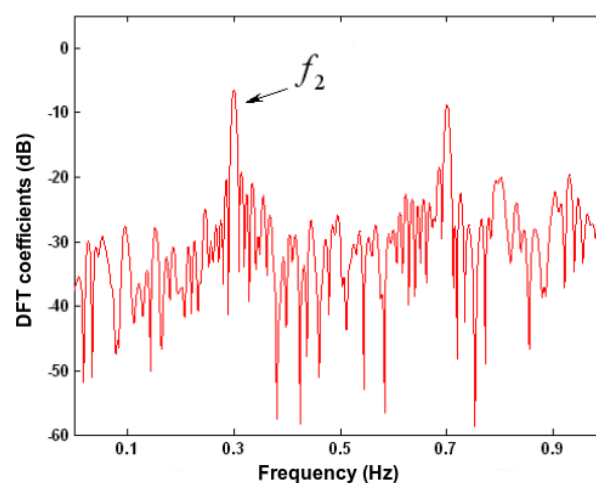
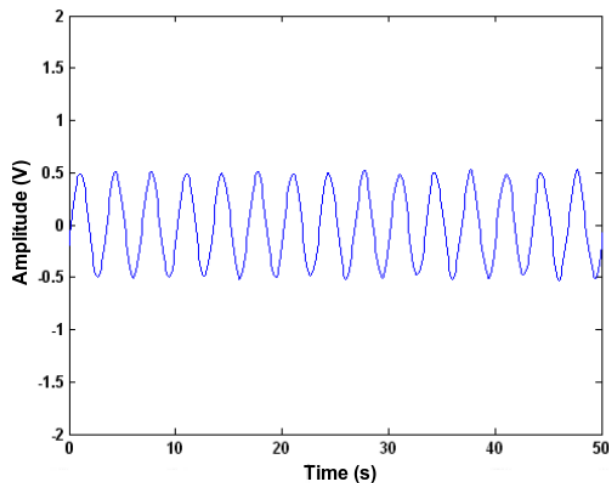
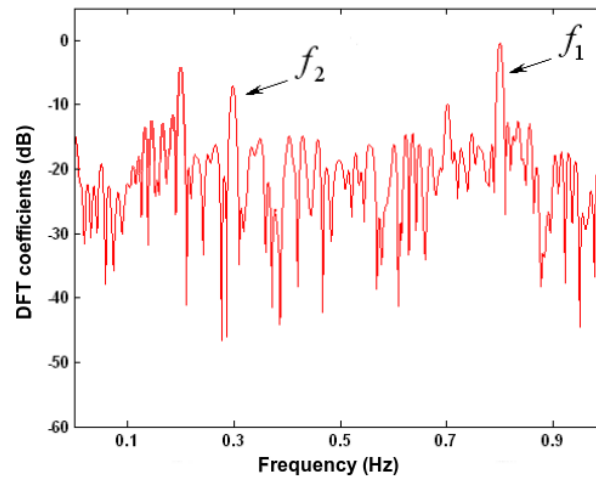
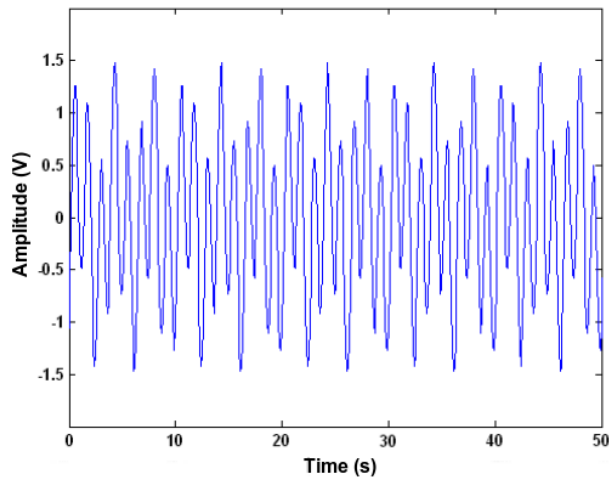
- The task is to estimate the parameters of components using signal samples $\{s(t_n)\} = \{s_n\}_{n=0,1,\dots,N-1}$
- An effective approach is sequential component extraction method (SECOEX), which is cyclic and based on the extraction of one discrete spectral component $A_i \sin(2\pi f_i t_n + \varphi_i)$ in each cycle $i=1,2,\dots$
- Starting with the initial sampling values $\{s_{1,n}\} = \{s_n\}$ the algorithm is

$$\sum_{n=0}^{N-1} (s_{i,n} - A_i \sin(2\pi f_i t_n + \varphi_i))^2 = \min$$
$$\{s_{i+1,n}\} = \{s_{i,n} - A_i \sin(2\pi f_i t_n + \varphi_i)\}$$


- Cycles are continued until the predefined limits (number of components or residual power) are reached.

Digital Alias-free Signal Processing

- Example of SECOEX



Digital Alias-free Signal Processing

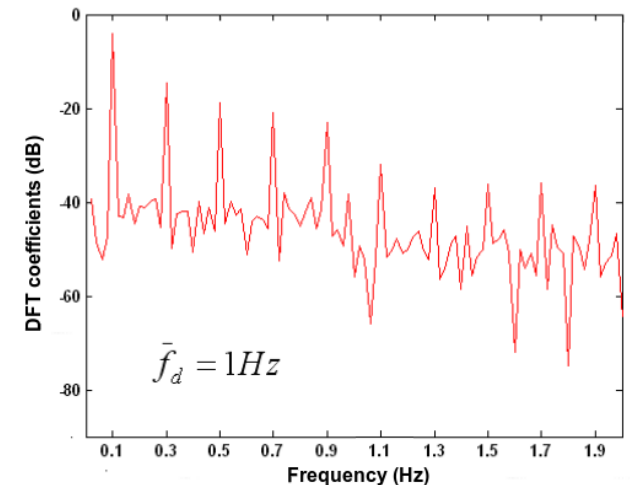
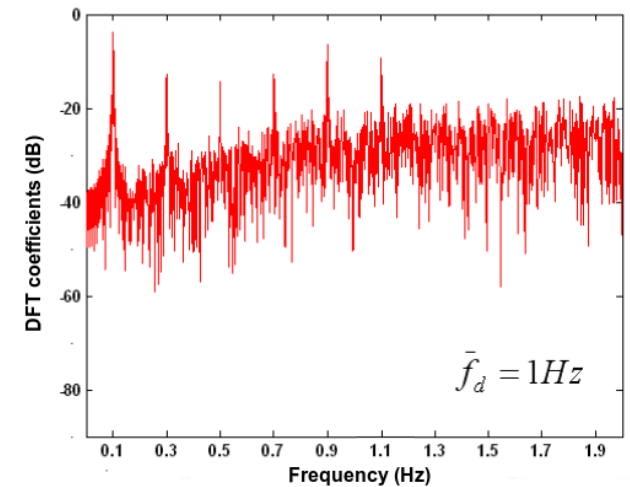
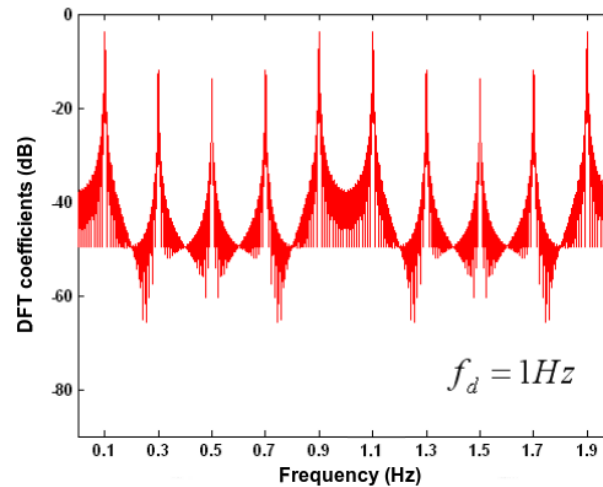
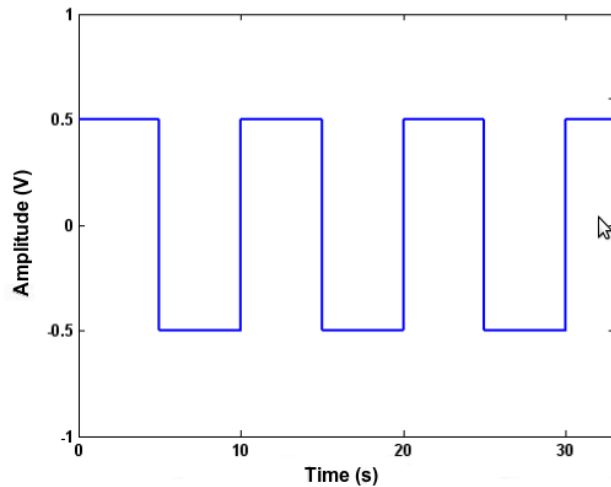
- In order to analyze periodic signals or signals with limited bandwidth not exceeding half the average sampling rate, the least mean squares (LMS) method can be used
- Given the frequencies $\{f_m\}_{m=1,2,\dots,M}$ the amplitudes $\{A_m\}$ and phases $\{\varphi_m\}$ are found to ensure the minimum error

$$\sum_{n=0}^{N-1} \left(s_n - \sum_{m=1}^M A_m \sin(2\pi f_m t_n + \varphi_m) \right)^2$$

- The total number of frequencies $M \leq \frac{N}{2}$ should not exceed half the number of samples

Digital Alias-free Signal Processing

- Example of LMS

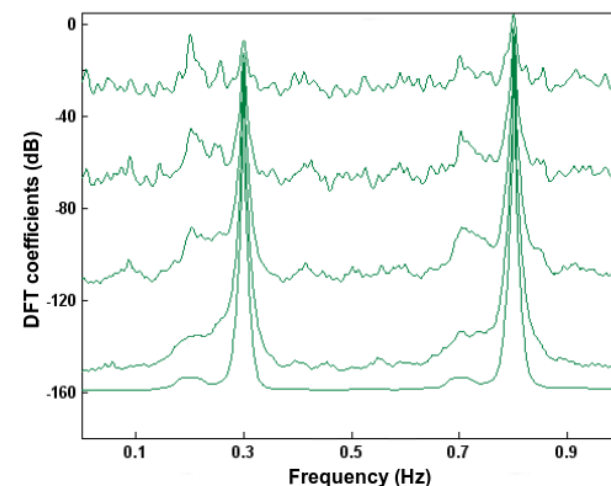
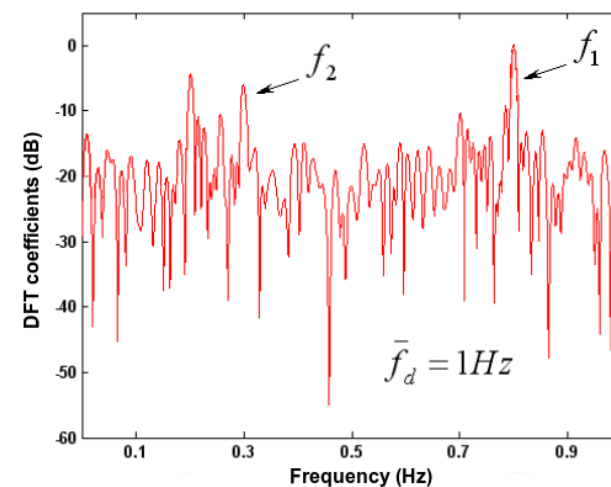
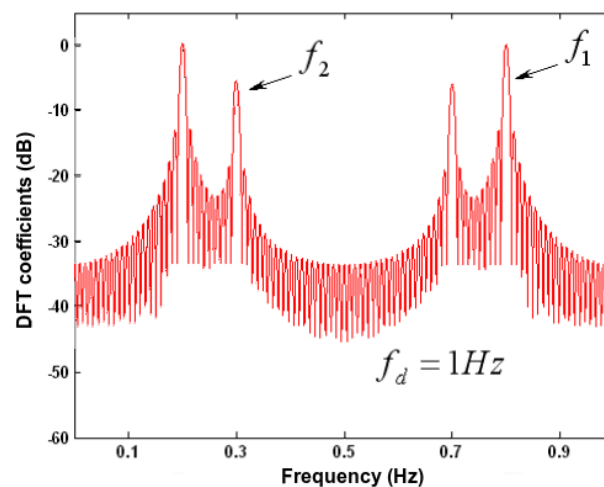
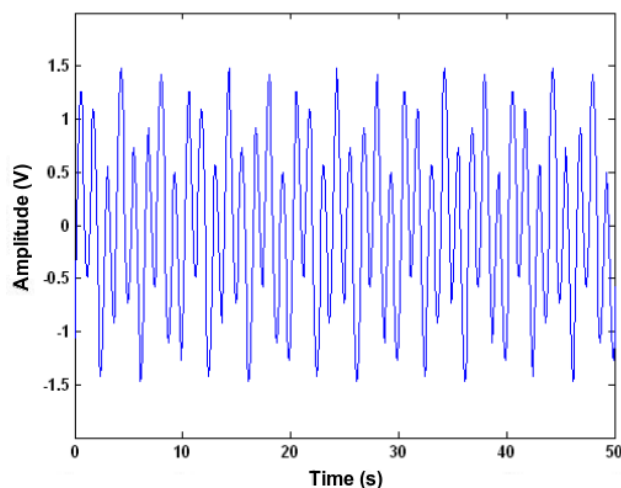


Signal-dependent transform

- Signal-dependent transform (SDT) is based on application of Minimum variance (MV) filter, the frequency response of which adapts to the spectral components of the input signal on each frequency of interest f_0
- Filter coefficients $\mathbf{a}_{MV}(f_0) = \frac{\mathbf{R}^{-1} \mathbf{e}(f_0)}{\mathbf{e}^H(f_0) \mathbf{R}^{-1} \mathbf{e}(f_0)}$ ensures
 - the sinusoidal signal with frequency f_0 passes through the filter designed for this frequency without distortion
 - the minimum variance of the output signal
- Square value of output signal $p(f_0) = |\mathbf{s} \mathbf{a}_{MV}(f_0)|^2$ indicates the power of spectral components of the input signal at frequency f_0
- The frequency band of spectral analysis is covered by a set of such MV filters

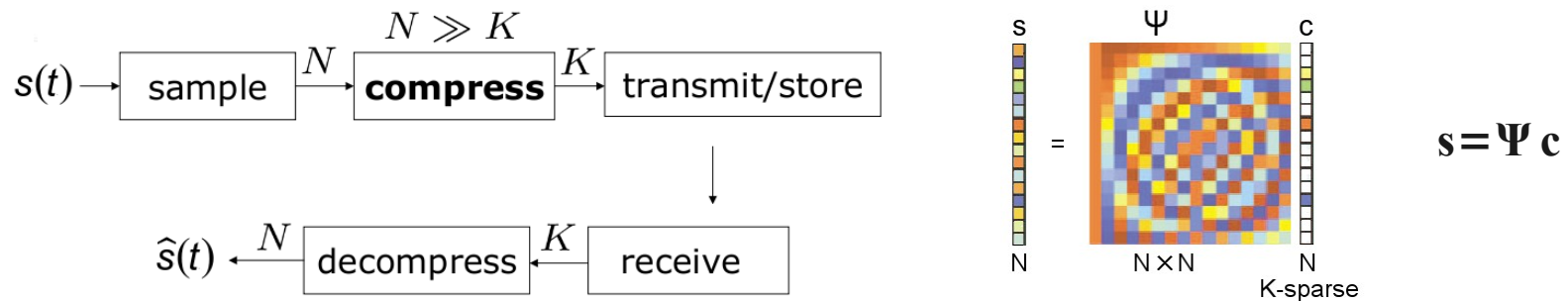
Signal-dependent transform

- Example of SDT



Compressive sensing

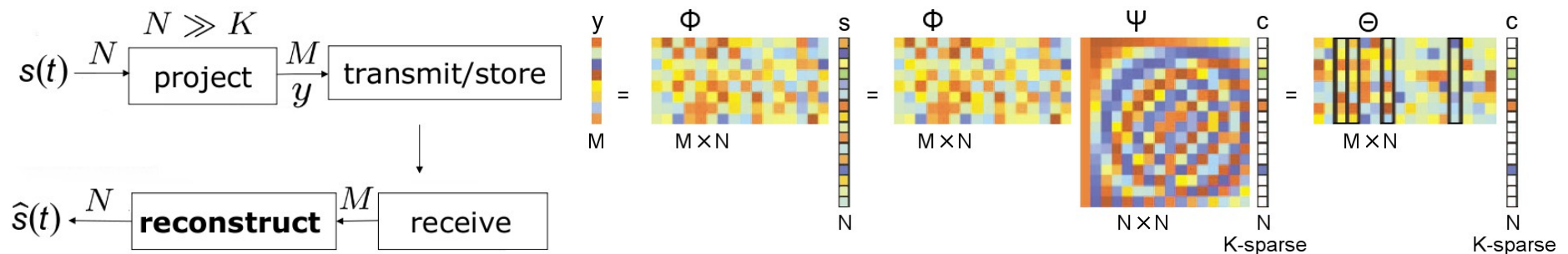
- The sampling theorem specifies that to avoid losing information when capturing the signal, the sampling rate must be at least twice the signal bandwidth
- In many applications the Nyquist rate is so high that too many samples result, making compression a necessity prior to storage or transmission



- Sample-then-compress framework has three inefficiencies: large N , all N coefficients must be computed, and the locations of large coefficients must be encoded
- In other applications due to large signal bandwidth it is difficult to sample at Nyquist rate

Compressive sensing

- Compressive sensing is a method to capture and represent compressible signals at a rate significantly below the Nyquist rate
- Samples are replaced by few nonadaptive linear projections $y = \Phi s = \Phi \Psi c = \Theta c$ that preserve the structure of the signal



Φ – measurement matrix (random Gaussian), Ψ – orthonormal basis, Θ – compressed sensing reconstruction matrix, $K < M \ll N$, $M = a K \log(N/K)$

- The signal is recovered from these projections using an optimization process

$$\hat{c} = \arg \min \sum_{n=1}^N |\hat{c}_n| \quad \text{such that} \quad \Theta \hat{c} = y$$

Finite rate of innovation

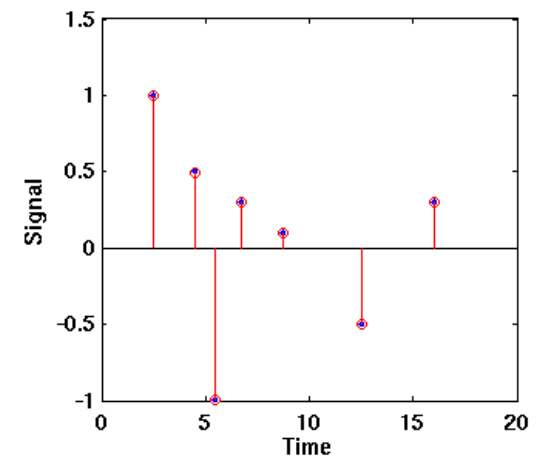
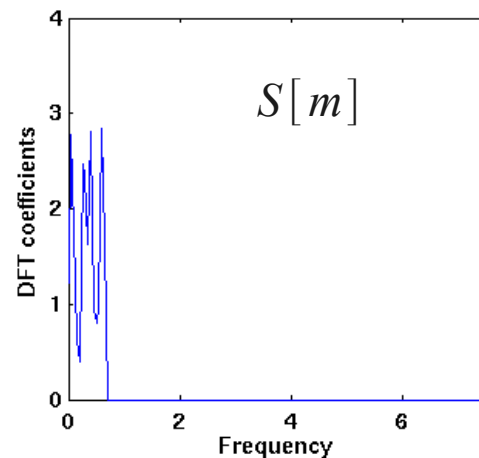
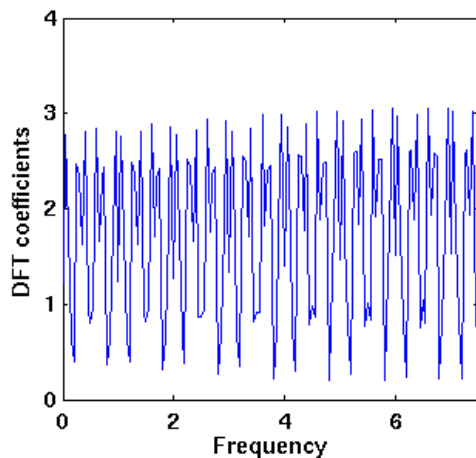
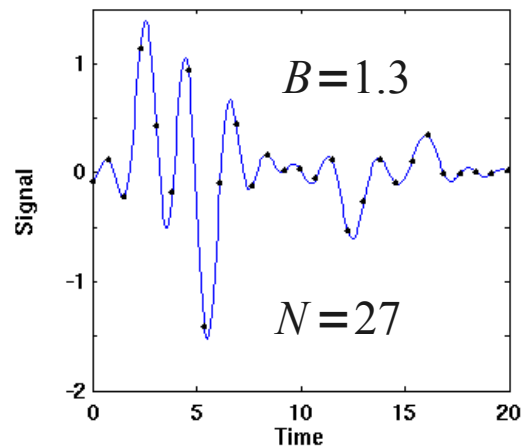
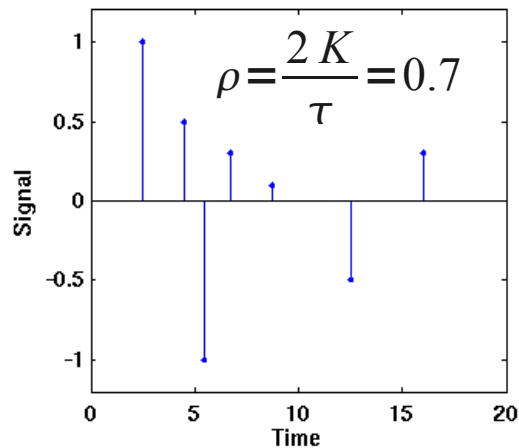
- Another compressive acquisition scheme is developed for signals with finite rate of innovation (FROI), e.g., streams of Dirac pulses, nonuniform splines, and piecewise polynomials
- Even though these signals are not band-limited, they can be sampled uniformly at (or above) the rate of innovation using an appropriate kernel, and then perfectly reconstructed by solving systems of linear equations
- For a stream of K Diracs periodized with period τ , $s(t) = \sum_{k=0}^{K-1} c_k \sum_{n \in \mathbb{Z}} \delta(t - t_k - n\tau)$, the rate of innovation is $\rho = 2K/\tau$
- After filtering the signal with $h(t) = B \text{sinc}(\pi B t)$, $B \geq \rho$ the samples at $N \geq 2M + 1$ uniform locations are taken, $M = \lfloor B\tau/2 \rfloor$
- The Fourier coefficients $S[m]$, $|m| \leq M$ are found, and the estimation of the annihilating filter $A[m]$ follows

$$A[m] * S[m] = 0$$

which lead to the K locations $\{t_k\}$ and weights $\{c_k\}$

Finite rate of innovation

- An example

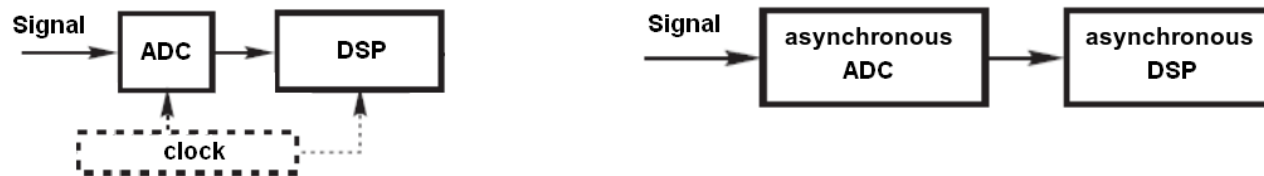


Advanced sampling and reconstruction methods:

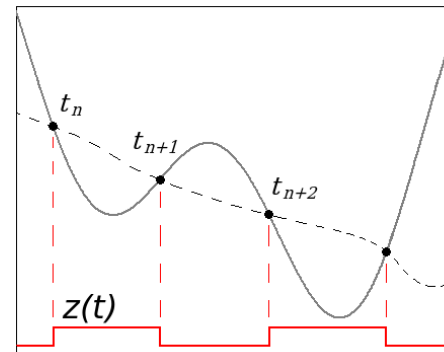
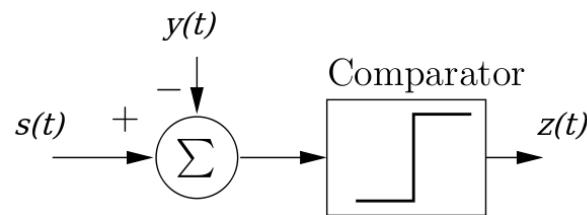
- **Amplitude encoding** (information is carried by signal samples s_n):
 - Digital alias-free signal processing
 - Compressive sensing
 - Finite rate of innovation
- **Time encoding** (information is carried by sampling instants t_n):
 - Time encoding machines (signal-dependent sampling)

Time encoding

- Time encoding is a real-time, asynchronous mechanism for encoding the amplitude information of an analog band-limited signal into a time sequence, or time codes, based on which the signal can be reconstructed



- Time codes can be generated by simple nonlinear asynchronous analog circuits (time encoding machines) with low power consumption



Time decoding

- For a given time sequence $\{t_n\}_{n \in \mathbb{Z}}$ a possible representation of the signal is given

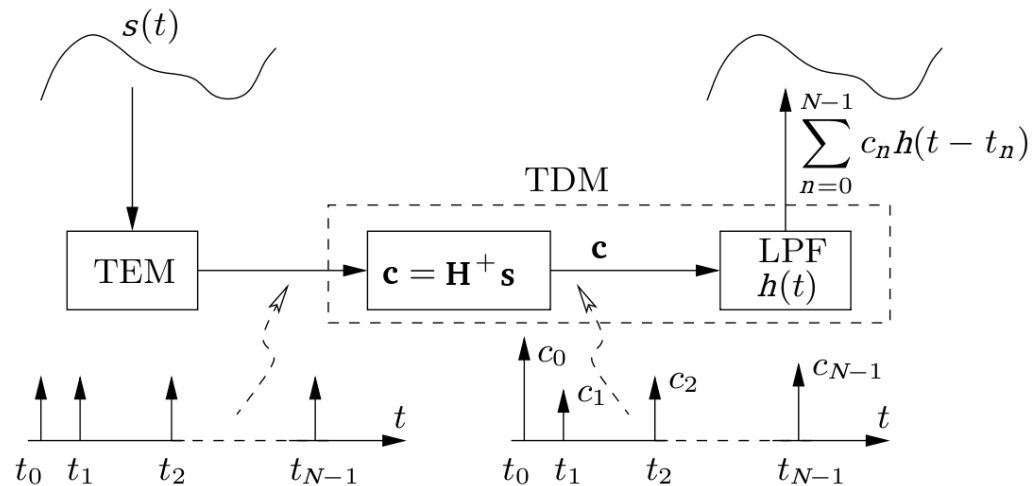
$$s(t) = \sum_{n \in \mathbb{Z}} c_n h(t - t_n), \quad h(t) = \text{sinc}(2\pi f_{\max} t)$$
- The unknown coefficients can be calculated from equation

$$\mathbf{s} = \mathbf{H} \mathbf{c}, \quad [\mathbf{s}]_m = s(t_m), \quad [\mathbf{c}]_m = c_m, \quad [\mathbf{H}]_{mn} = h(t_m - t_n)$$
- Unknown $\mathbf{c}$ for perfect recovery is possible if $\max_n (t_{n+1} - t_n) \leq 1/(2f_{\max})$
- In practice, a limited number of samples is available, therefore

$$\hat{s}(t) = \sum_{n=0}^{N-1} c_n h(t - t_n),$$

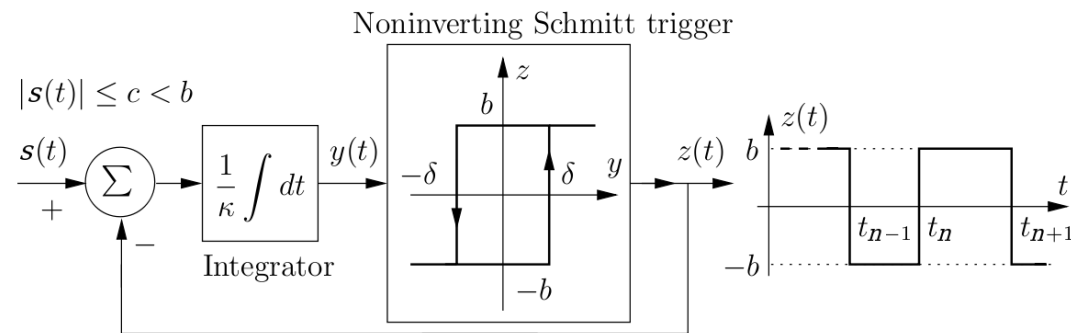
where

$$\mathbf{c} = \mathbf{H}^+ \mathbf{s}$$



Asynchronous sigma-delta modulator

- Time encoding machine: asynchronous sigma-delta modulator



- The integrator output is a strictly increasing or decreasing function for $t \in [t_n, t_{n+1}]$ so that $y(t_n) = (-1)^n \delta$ and

$$\int_{t_n}^{t_{n+1}} s(t) dt = (-1)^n (2\kappa\delta - b(t_{n+1} - t_n))$$

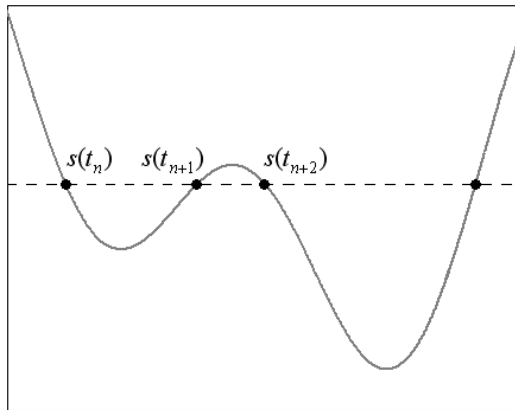
- The difference between the neighboring time codes is bounded by circuit parameters κ , δ and b , and the amplitude bound of the input c

$$\frac{2\kappa\delta}{b+c} \leq t_{n+1} - t_n \leq \frac{2\kappa\delta}{b-c}$$

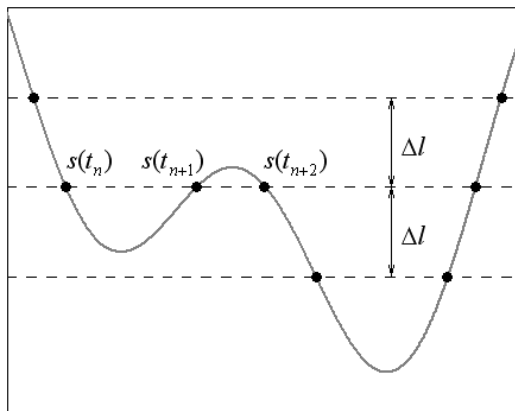
Time encoding

- Other time encoding methods:

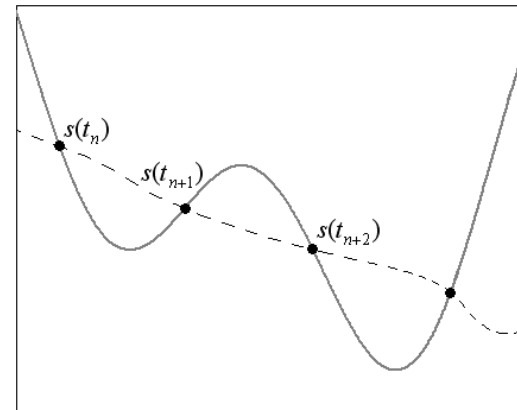
zero crossing



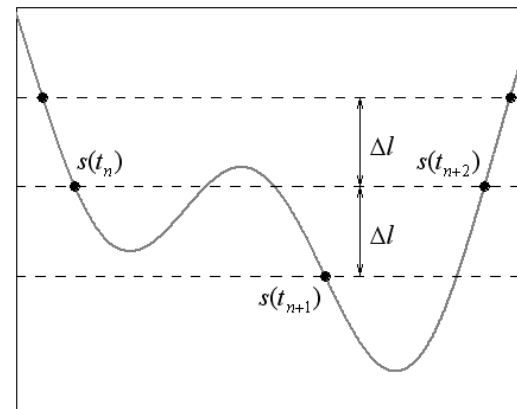
level-crossing



reference signal crossing

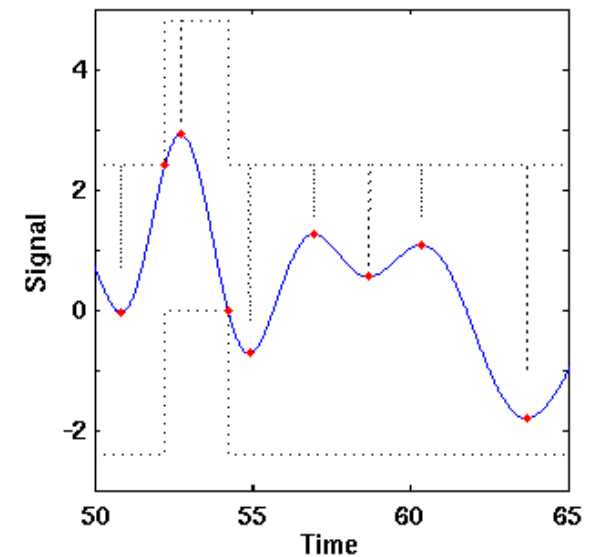
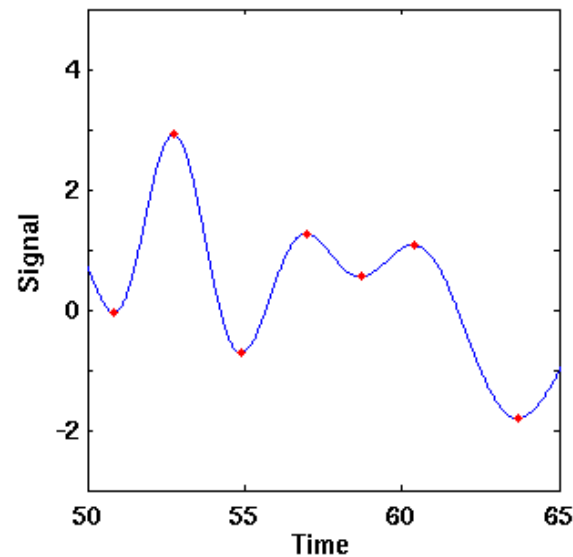
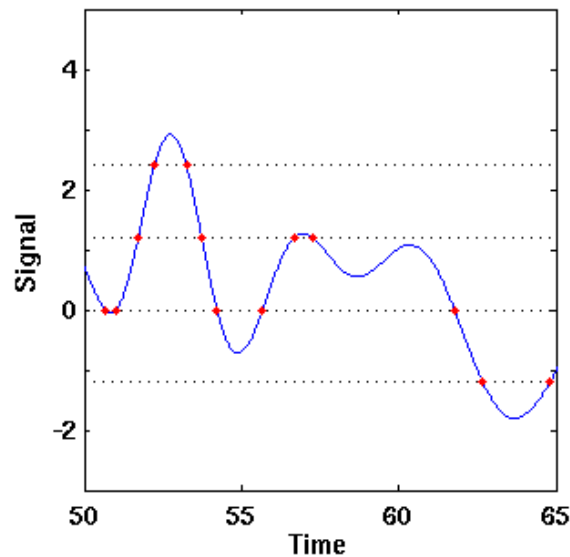


send-on-delta



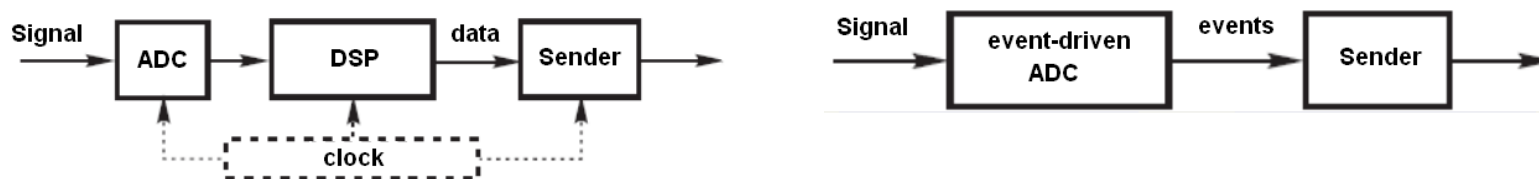
Signal-dependent sampling and reconstruction

- Level-crossing sampling
- Mini-max sampling
- Adaptive level-crossing sampling



Signal-dependent sampling

- It is important to reduce energy consumption of wireless sensor networks
- Most of the energy is consumed by data transmission, thus the option is to reduce the amount of data to be transferred by using event-driven, signal-dependent sampling
- In result information about signal is transferred only when events occur
- Two different schemes for data transmission:

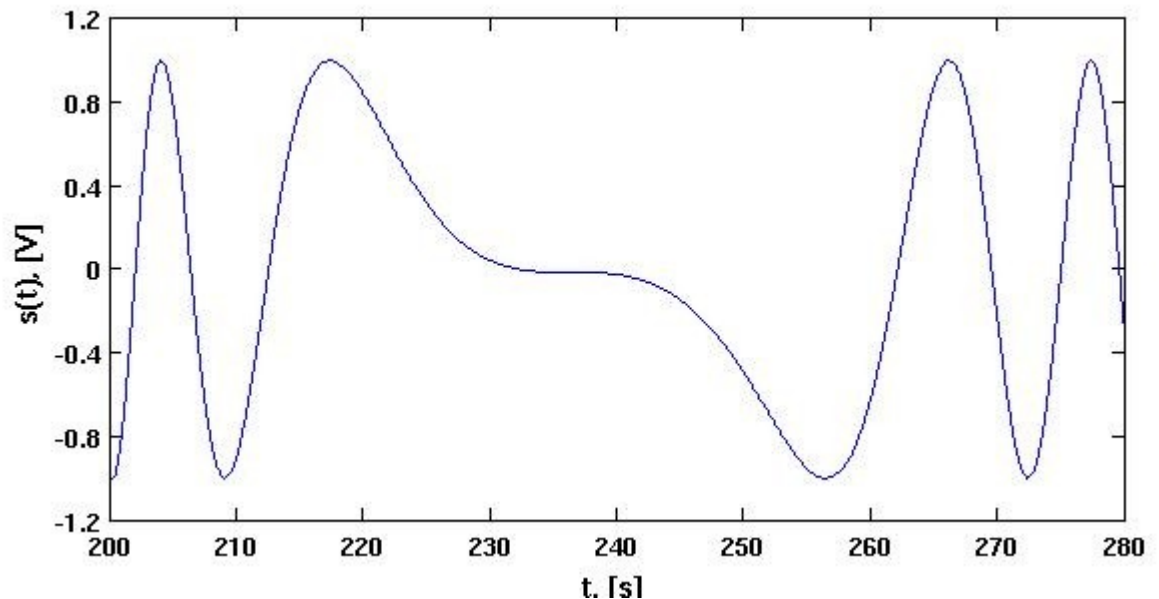
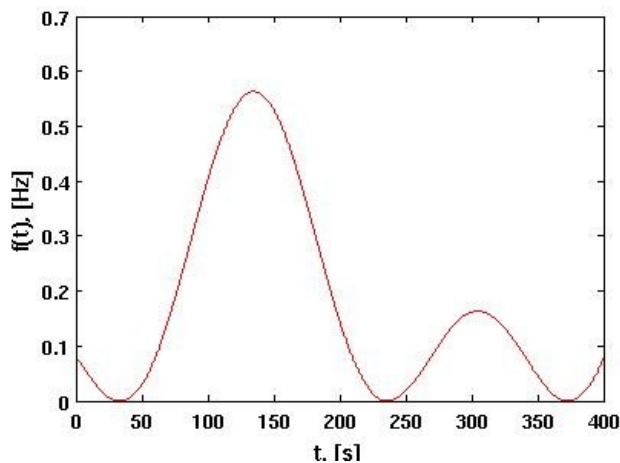


- In signal-dependent sampling case event-driven ADC is driven by signal being sampled

Sampling by local maximum frequency

- Bandlimited signal $s(t)$ with frequency $f \leq F_{max}$ traditionally is sampled uniformly with sampling frequency $f_d \geq 2 F_{max}$
- Given the discrete signal samples $s(t_n)$ the original analogue signal can be calculated by formula

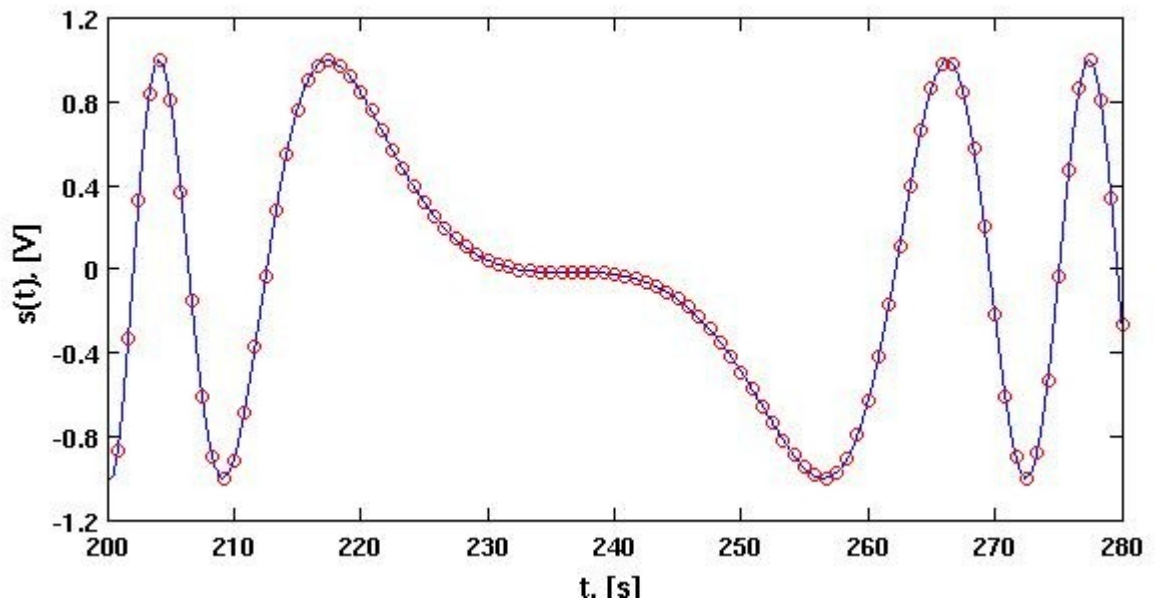
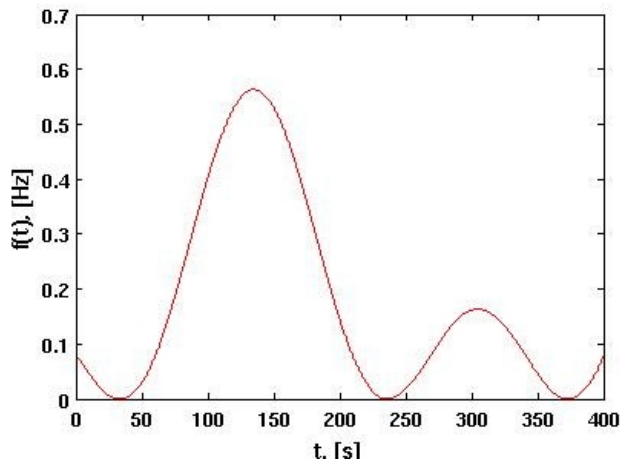
$$s(t) = \sum_{n=-\infty}^{\infty} s(t_n) h(t, t_n), \quad h_1(t, t_n) = \text{sinc}(2\pi F_{max}(t - t_n)), \quad t_n = \frac{n}{2F_{max}}$$



Sampling by local maximum frequency

- Bandlimited signal $s(t)$ with frequency $f \leq F_{max}$ traditionally is sampled uniformly with sampling frequency $f_d \geq 2 F_{max}$
- Given the discrete signal samples $s(t_n)$ the original analogue signal can be calculated by formula

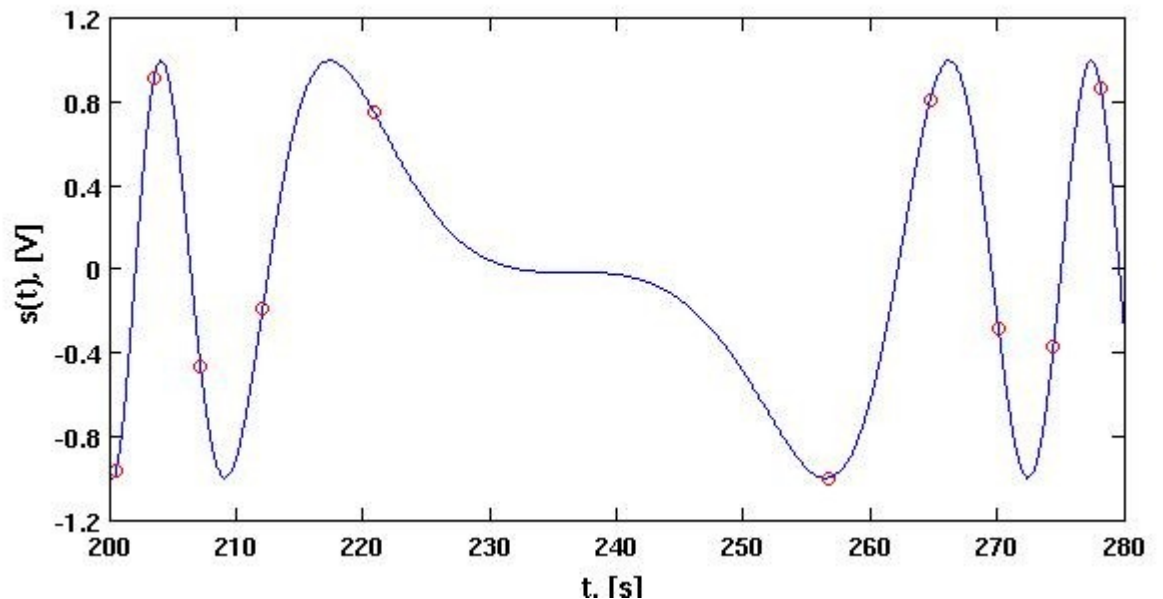
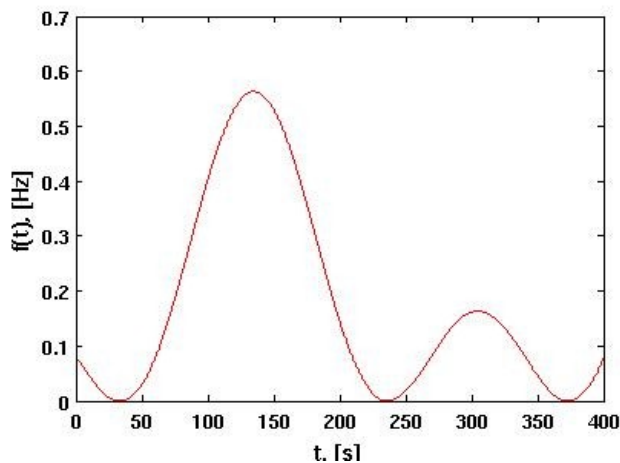
$$s(t) = \sum_{n=-\infty}^{\infty} s(t_n) h(t, t_n), \quad h_1(t, t_n) = \text{sinc}(2\pi F_{max}(t - t_n)), \quad t_n = \frac{n}{2F_{max}}$$



Sampling by local maximum frequency

- In signal-dependent sampling case the capturing of signal samples depends on time-varying properties of the signal

$$s(t) = \sum_{n=0}^{\infty} s(t_n) h_2(t, t_n), \quad h_2(t, t_n) = \text{sinc}(\Phi(t) - \Phi(t_n)), \quad \Phi(t) = 2\pi \int_{t_0}^t f_{\max}(t) dt$$
$$\Phi(t_n) = n\pi$$

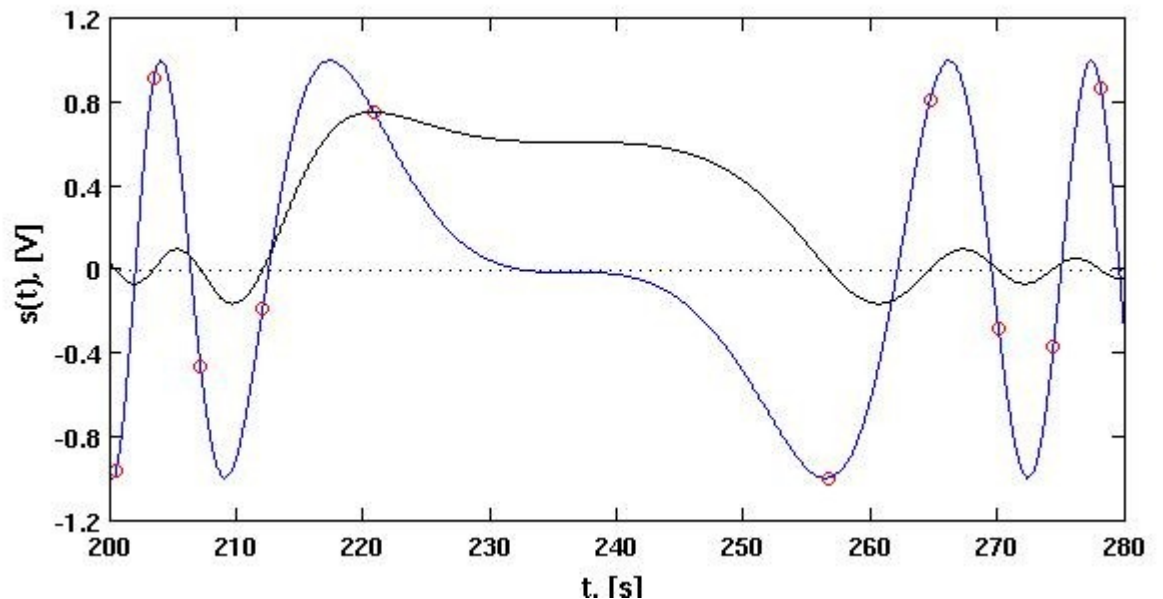
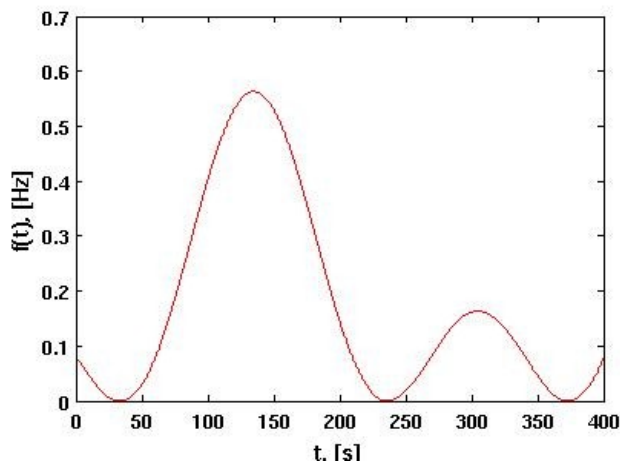


Sampling by local maximum frequency

- In signal-dependent sampling case the capturing of signal samples depends on time-varying properties of the signal

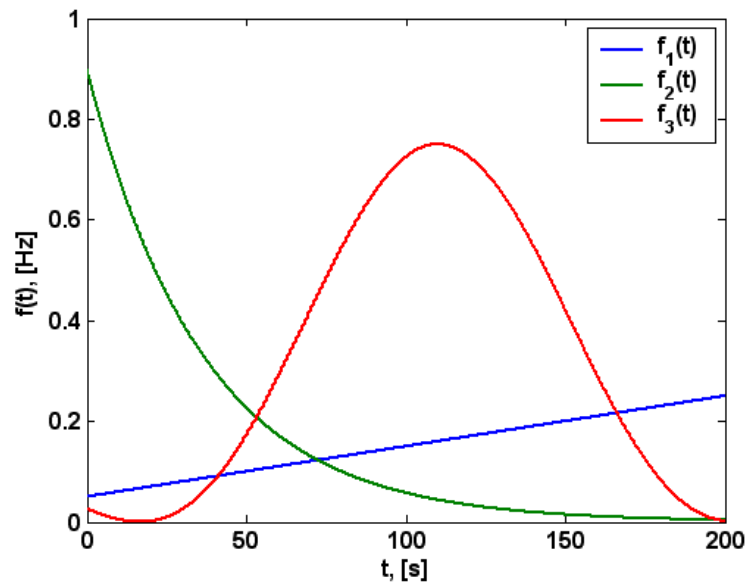
$$s(t) = \sum_{n=0}^{\infty} s(t_n) h_2(t, t_n), \quad h_2(t, t_n) = \text{sinc}(\Phi(t) - \Phi(t_n)), \quad \Phi(t) = 2\pi \int_{t_0}^t f_{\max}(t) dt$$

$$\Phi(t_n) = n\pi$$



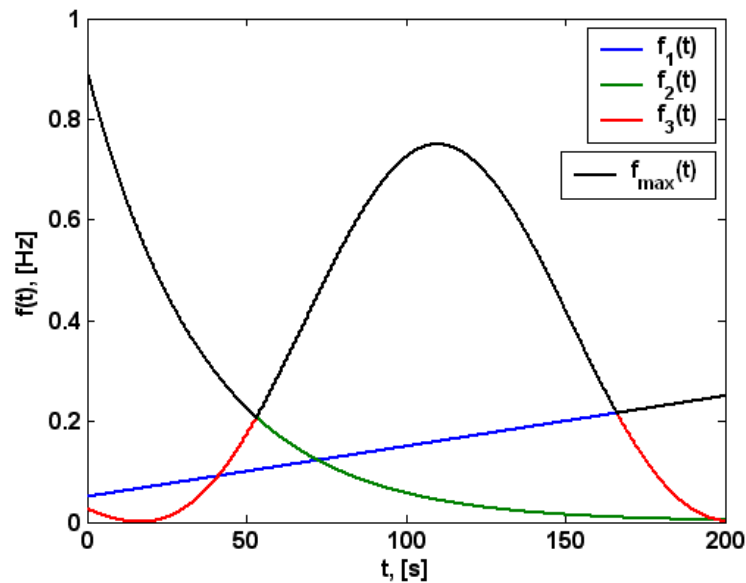
Sampling by local maximum frequency

- Example



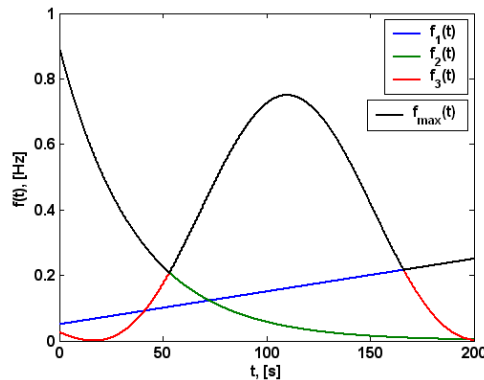
Sampling by local maximum frequency

- Example



Sampling by local maximum frequency

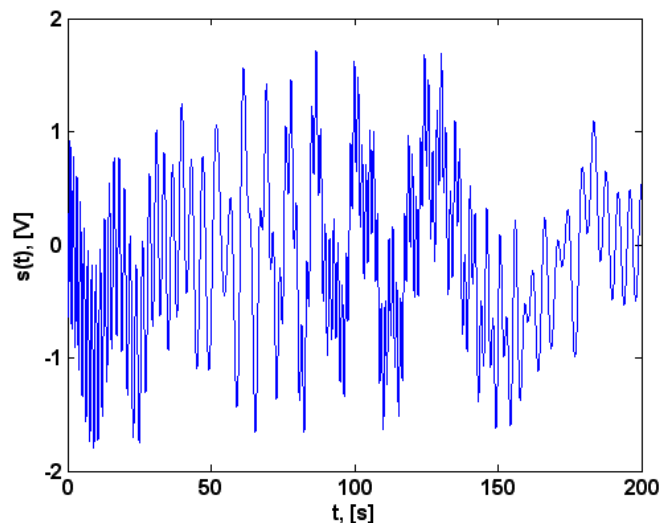
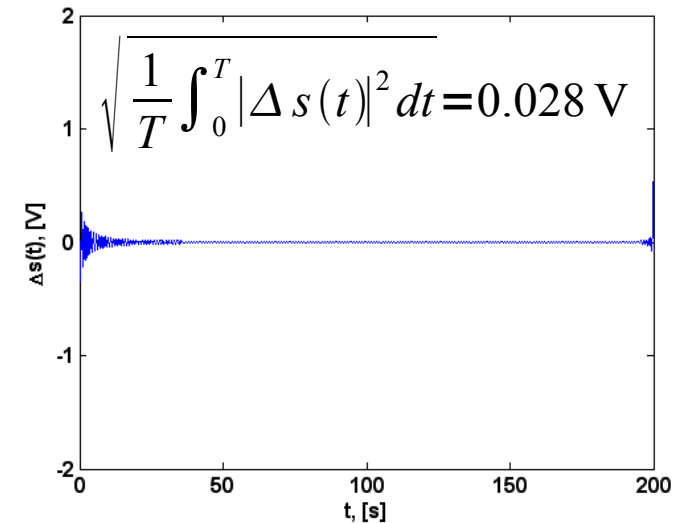
- Example



$$t_n = \frac{n}{2 F_{max}}, \quad N_1 = 396$$

$$\hat{s}(t) = \sum_{n=0}^{N_1-1} s(t_n) h_1(t, t_n)$$

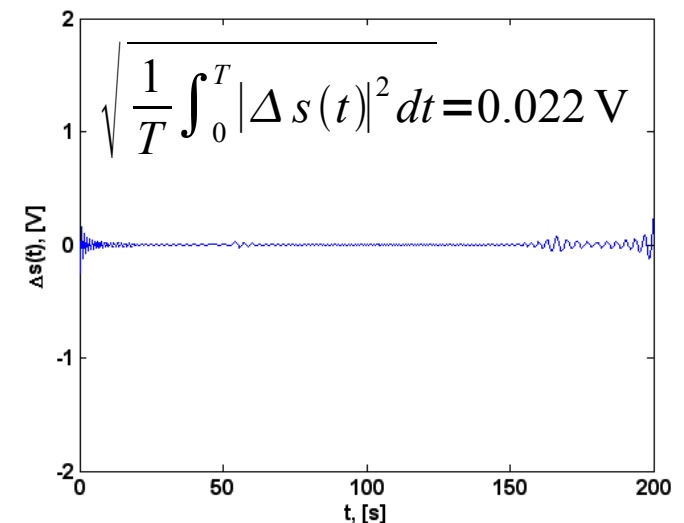
$$\Delta s(t) = s(t) - \hat{s}(t)$$



$$\Phi(t) = 2\pi \int_0^t f_{max}(t) dt$$

$$\Phi(t_n) = n\pi, \quad N_2 = 239$$

$$\hat{s}(t) = \sum_{n=0}^{N_2-1} s(t_n) h_2(t, t_n)$$



Signal reconstruction from level-crossing samples

- In practice no information about instantaneous maximum frequency $f_{max}(t)$ of the signal is usually provided, thus an alternative choice is level-crossing sampling
- The locations $\{t_m\}$ of level-crossing samples depend on time-varying frequency properties of the signal

$$s(t) = \sum_{k=0}^{\infty} s(t_k) h(t, t_k), \quad h(t, t_k) = \text{sinc}(\Phi(t) - \Phi(t_k)), \quad \Phi(t) = 2\pi \int_{t_0}^t f_{max}(t) dt, \\ \Phi(t_k) = k\pi$$

- The signal is reconstructed as

$$\hat{s}(t) = \sum_{n=0}^{N-1} \hat{s}(t_n) \hat{h}(t, t_n), \quad \hat{h}(t, t_n) = \text{sinc}(\hat{\Phi}(t) - \hat{\Phi}(t_n)), \quad \hat{\Phi}(t) = 2\pi \int_{t_0}^t \hat{f}_{max}(t) dt, \\ \hat{\Phi}(t_n) = n\pi$$

- Unknown $\hat{s}(t_n)$ and $\hat{f}_{max}(t)$ are estimated from level-crossing samples

Signal reconstruction from level-crossing samples

- To estimate the frequency $f_{\max}(t)$ from level-crossing samples $s(t_m)$, starting with the initial index value $m=0$

- Two pairs of successive level-crossing samples are found

$$1) s(t_{m'_j}) = s(t_{m'_{j+1}})$$

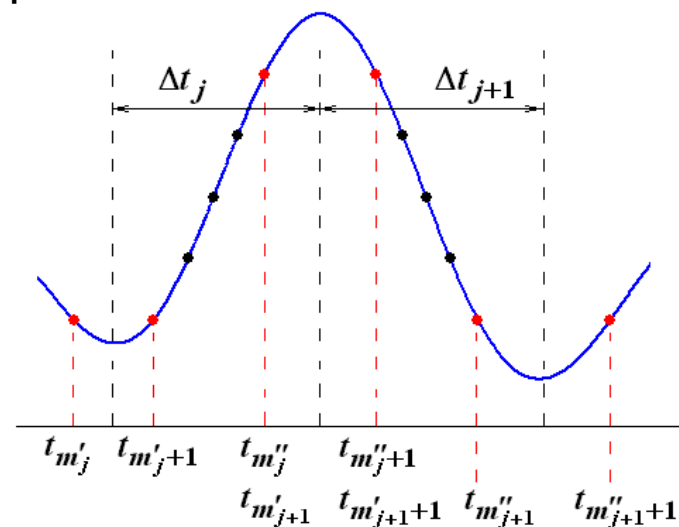
$$2) s(t_{m''_j}) = s(t_{m''_{j+1}})$$

such, that the distance $m''_j - m'_j \geq 1$ is minimal

- From $t_{m'_j}$, $t_{m'_{j+1}}$, $t_{m''_j}$ and $t_{m''_{j+1}}$ the value $f(t_j)$ is calculated

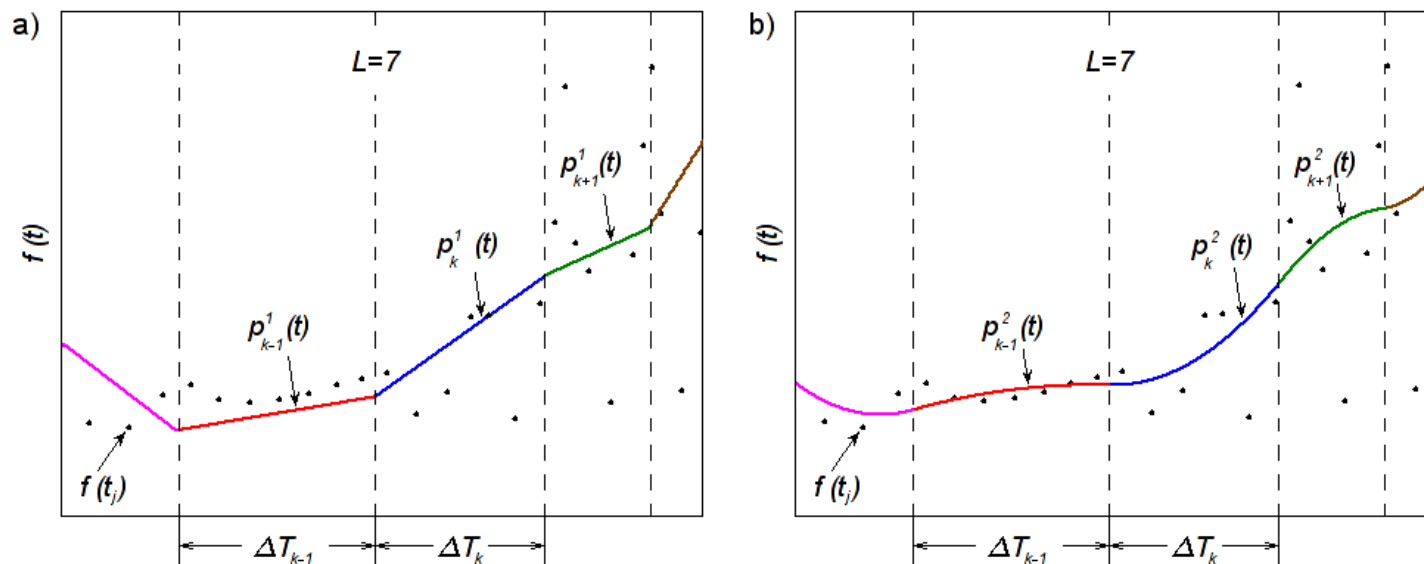
$$t_j = \frac{t_{m''_j} + t_{m''_{j+1}} + t_{m'_j} + t_{m'_{j+1}}}{4}, \quad \Delta t_j = \frac{t_{m''_j} + t_{m''_{j+1}}}{2} - \frac{t_{m'_j} + t_{m'_{j+1}}}{2}, \quad f(t_j) = \frac{1}{2\Delta t_j}$$

- Thereafter the next pairs $j+1, \dots$ are found to calculate $f(t_{j+1}), \dots$



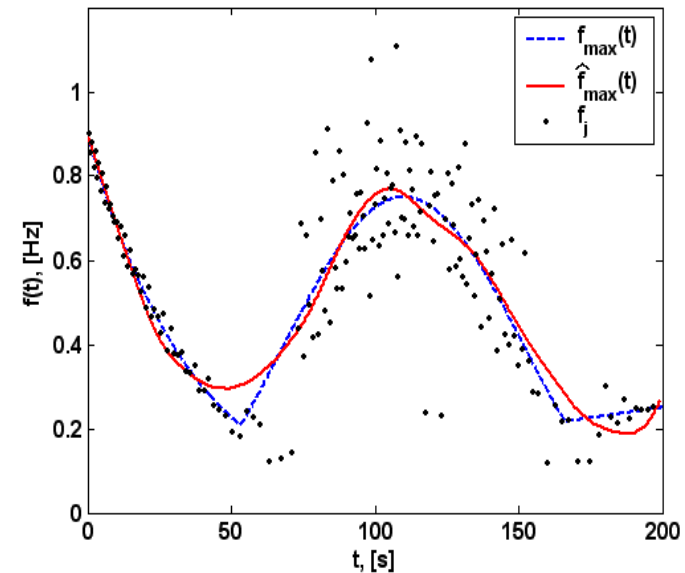
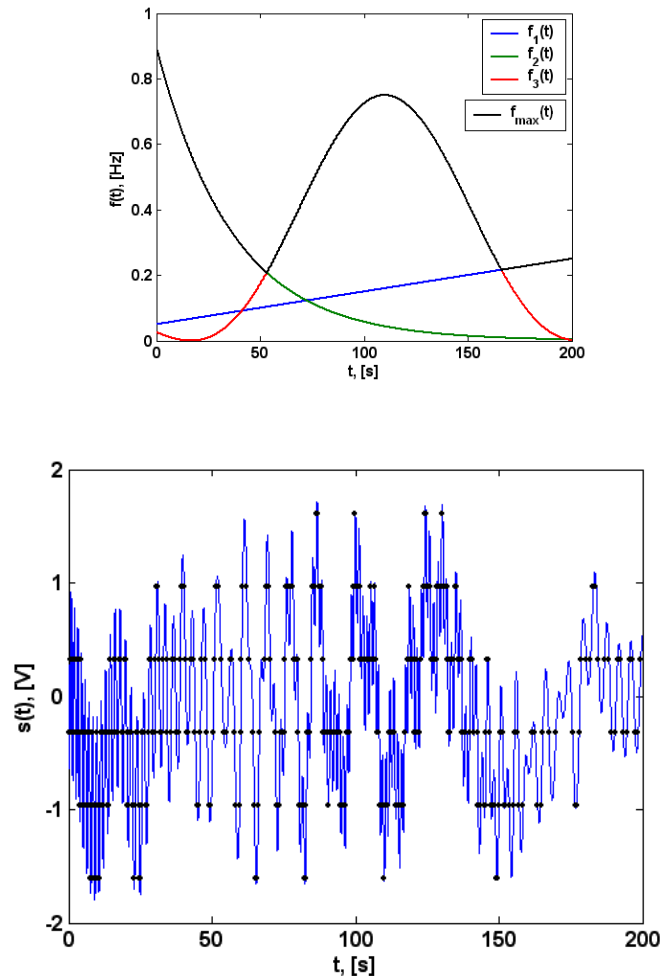
Signal reconstruction from level-crossing samples

- ✓ For a single sinusoid all $f(t_j)$ values are similar and equal the frequency of the sinusoid
- ✓ For a multicomponent signal the mean $f(t_j)$ value $\frac{1}{J} \sum_{j=1}^J f_j$ is close to the frequency of the highest component
- ✓ Thus, the estimate of function of instantaneous maximum frequency $f_{\max}(t)$ can be obtained by $f(t_j)$ approximation with piecewise polynomials of order L or



Signal reconstruction from level-crossing samples

- Example



Signal reconstruction from level-crossing samples

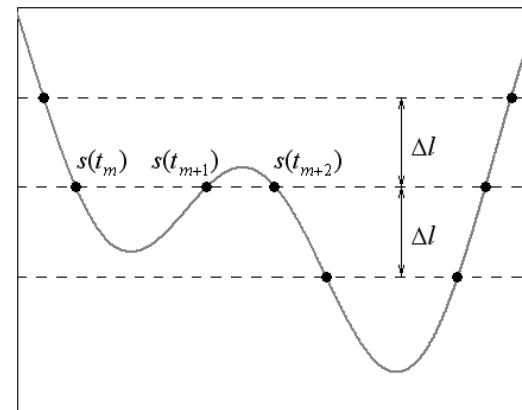
- Reconstruction by interpolation and filtering method

$$\hat{s}_0(t_n) = \check{s}_{s(t_m)}(t_n);$$

$$\hat{s}_0(t) = C[\hat{s}_0(t_n)]; \quad C[s(t_n)] = \sum_{n=0}^{N-1} s(t_n) h(t, t_n)$$

$$\hat{s}_i(t_n) = \hat{s}_{i-1}(t_n) + \check{s}_{(s-s_{i-1})(t_m)}(t_n);$$

$$\hat{s}_i(t) = C[\hat{s}_i(t_n)]$$



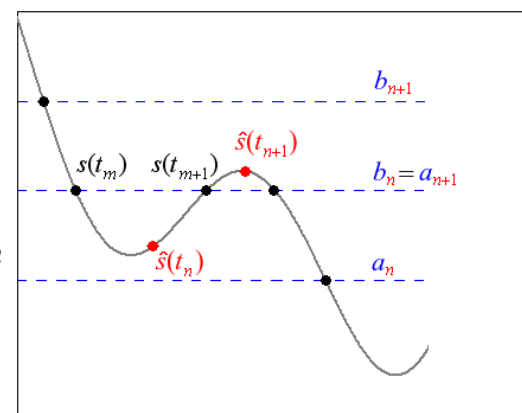
- Reconstruction by LMS method

$$\hat{\mathbf{s}} = \mathbf{H}^+ \mathbf{s}, \quad [\hat{\mathbf{s}}]_n = \hat{s}(t_n), \quad [\mathbf{s}]_m = s(t_m), \quad [\mathbf{H}]_{mn} = h(t_m, t_n)$$

- Reconstruction by finding a minimum

$$\min_{\hat{\mathbf{s}}} \hat{\mathbf{s}}^T (\mathbf{H}^T \mathbf{H}) \hat{\mathbf{s}} - 2(\mathbf{H}^T \mathbf{s})^T \hat{\mathbf{s}} \quad \text{such that} \quad a_n \leq \hat{s}(t_n) \leq b_n$$

- Reconstruction result depends on the lengths of the sampling intervals $t_{m+1} - t_m$



Signal reconstruction from level-crossing samples

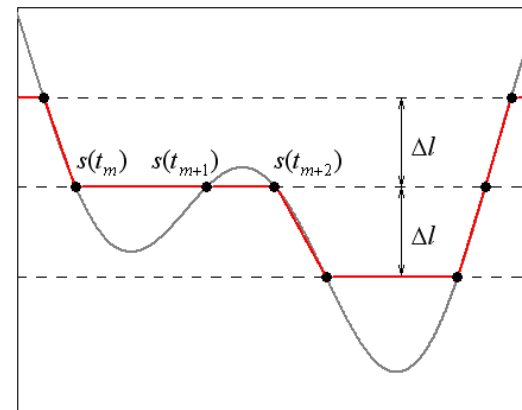
- Reconstruction by interpolation and filtering method

$$\hat{s}_0(t_n) = \check{s}_{s(t_m)}(t_n);$$

$$\hat{s}_0(t) = C[\hat{s}_0(t_n)]; \quad C[s(t_n)] = \sum_{n=0}^{N-1} s(t_n) h(t, t_n)$$

$$\hat{s}_i(t_n) = \hat{s}_{i-1}(t_n) + \check{s}_{(s-\hat{s}_{i-1})(t_m)}(t_n);$$

$$\hat{s}_i(t) = C[\hat{s}_i(t_n)]$$



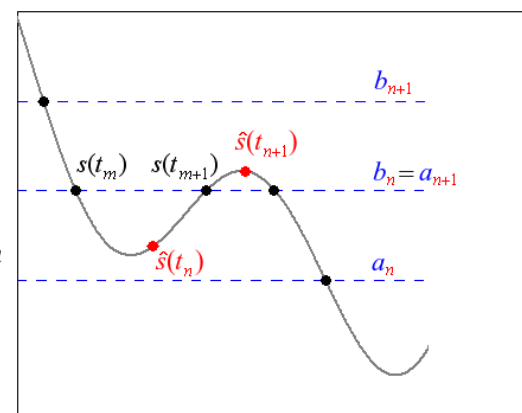
- Reconstruction by LMS method

$$\hat{\mathbf{s}} = \mathbf{H}^+ \mathbf{s}, \quad [\hat{\mathbf{s}}]_n = \hat{s}(t_n), \quad [\mathbf{s}]_m = s(t_m), \quad [\mathbf{H}]_{mn} = h(t_m, t_n)$$

- Reconstruction by finding a minimum

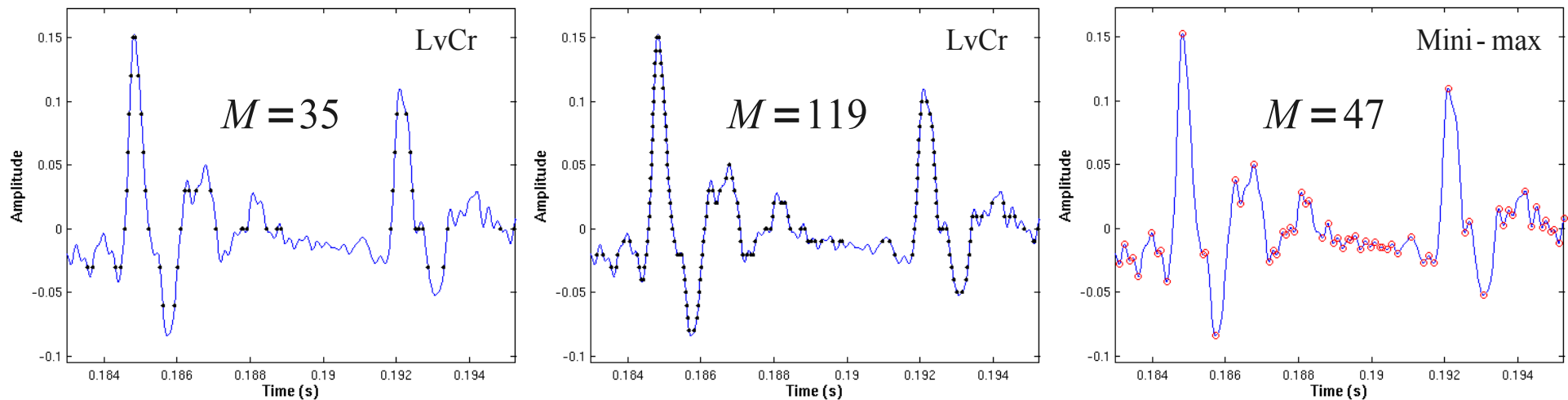
$$\min_{\hat{\mathbf{s}}} \hat{\mathbf{s}}^T (\mathbf{H}^T \mathbf{H}) \hat{\mathbf{s}} - 2(\mathbf{H}^T \mathbf{s})^T \hat{\mathbf{s}} \quad \text{such that} \quad a_n \leq \hat{s}(t_n) \leq b_n$$

- Reconstruction result depends on the lengths of the sampling intervals $t_{m+1} - t_m$



Adaptive level-crossing sampling

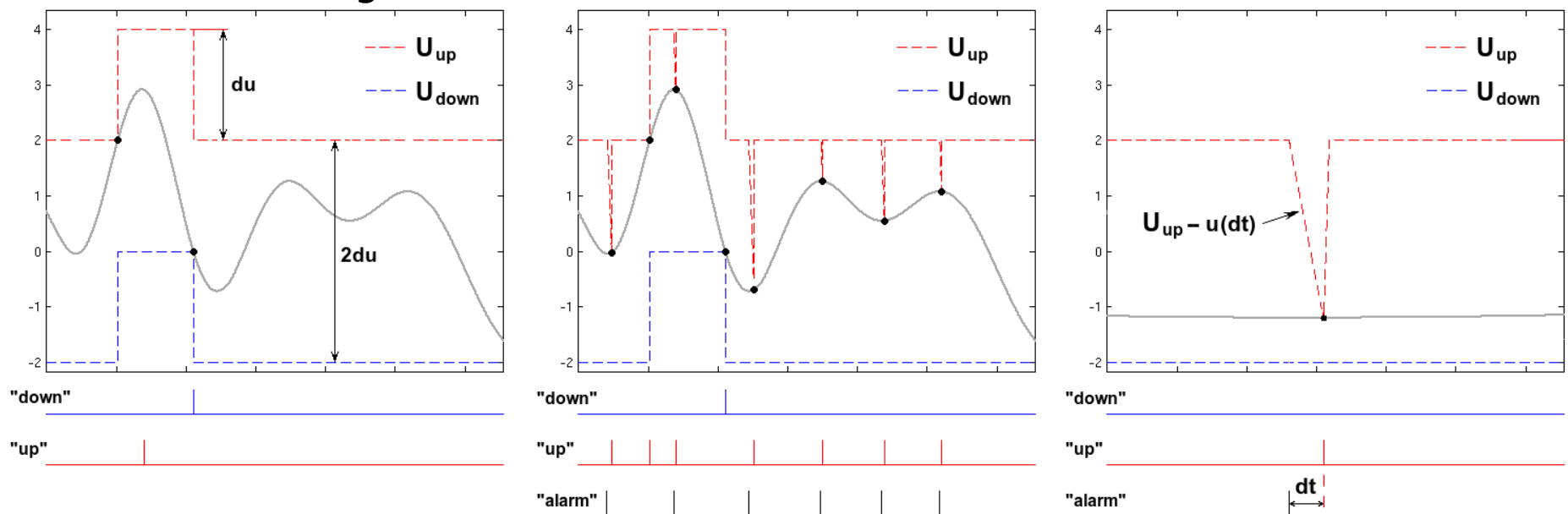
- Sampling intervals $t_{m+1} - t_m$ depend on the signal and the locations of quantization levels



- Too sparsely placed levels lead to information loss about signal changes occurring between the levels, and this can reduce signal reconstruction quality
- If the levels are placed more densely, less significant information is lost, however, high sampling density is created in places where it is not necessary

Adaptive level-crossing sampling

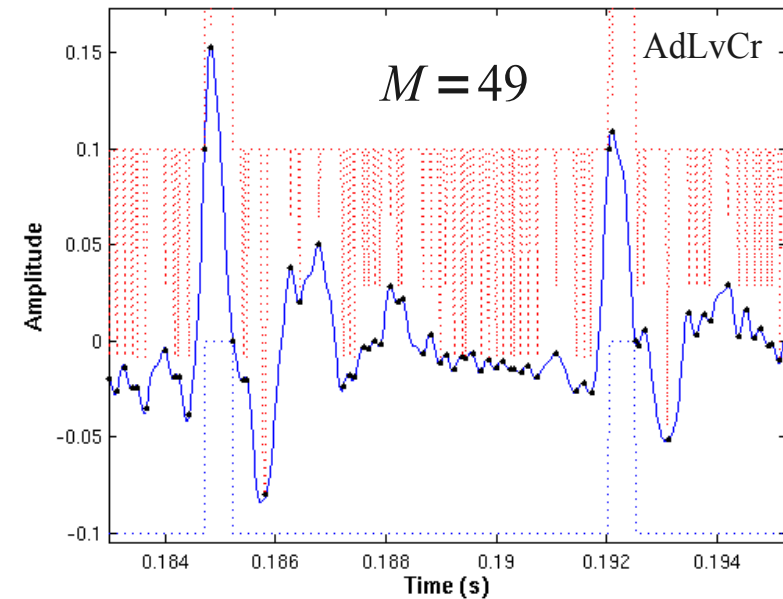
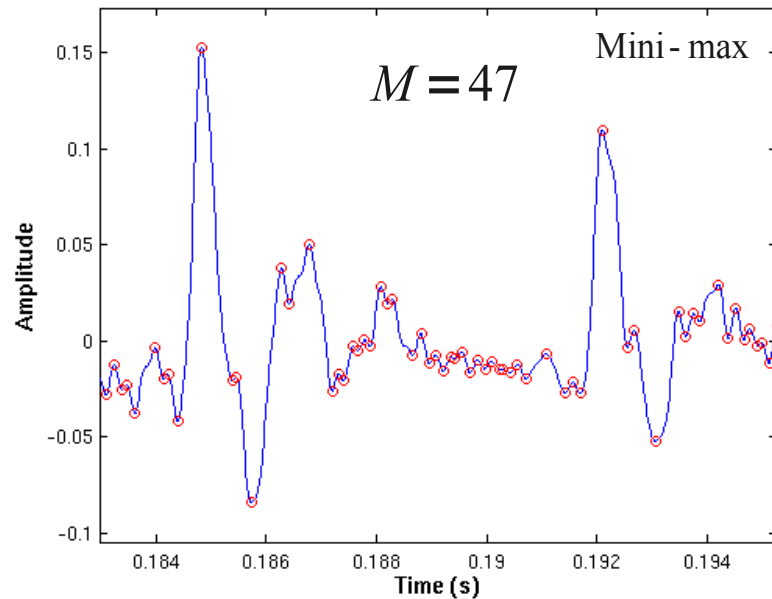
- Adaptive level-crossing sampling allows to keep sparse placement of the levels and at the same time not to lose significant information about signal waveform changes between the levels



- It takes into account the signal derivative value to change the corresponding quantization level U_{up} , and samples close to local maximums and minimums are captured in addition to level-crossing samples
- For signal reconstruction purpose pulses “down”, “up” and “alarm” are used

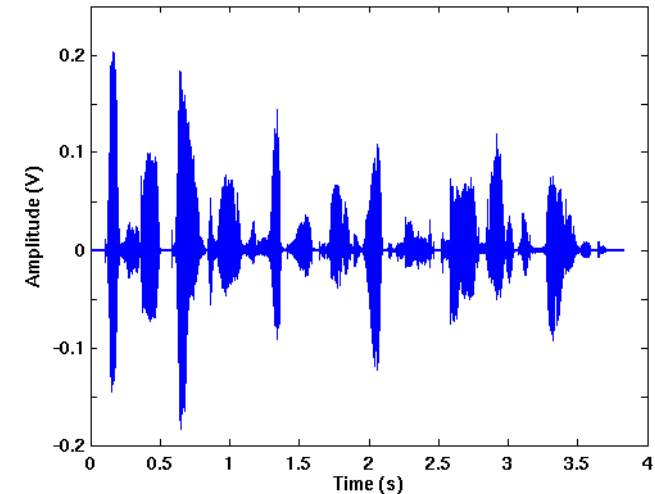
Adaptive level-crossing sampling

- Example



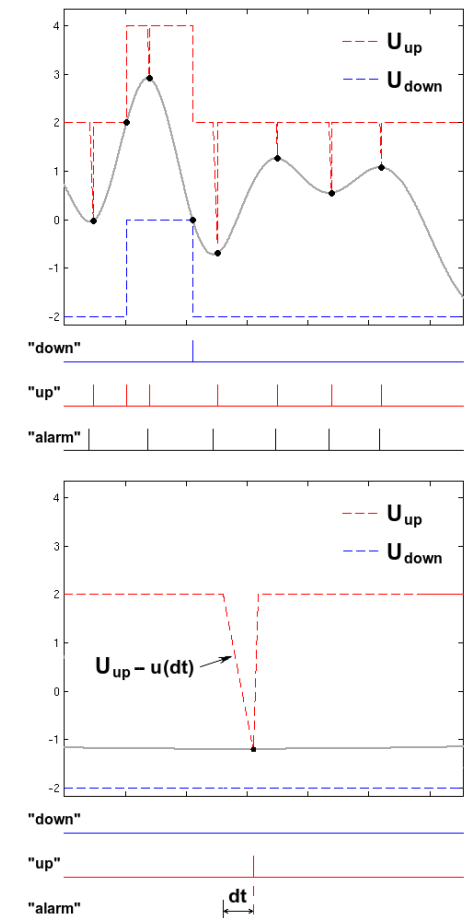
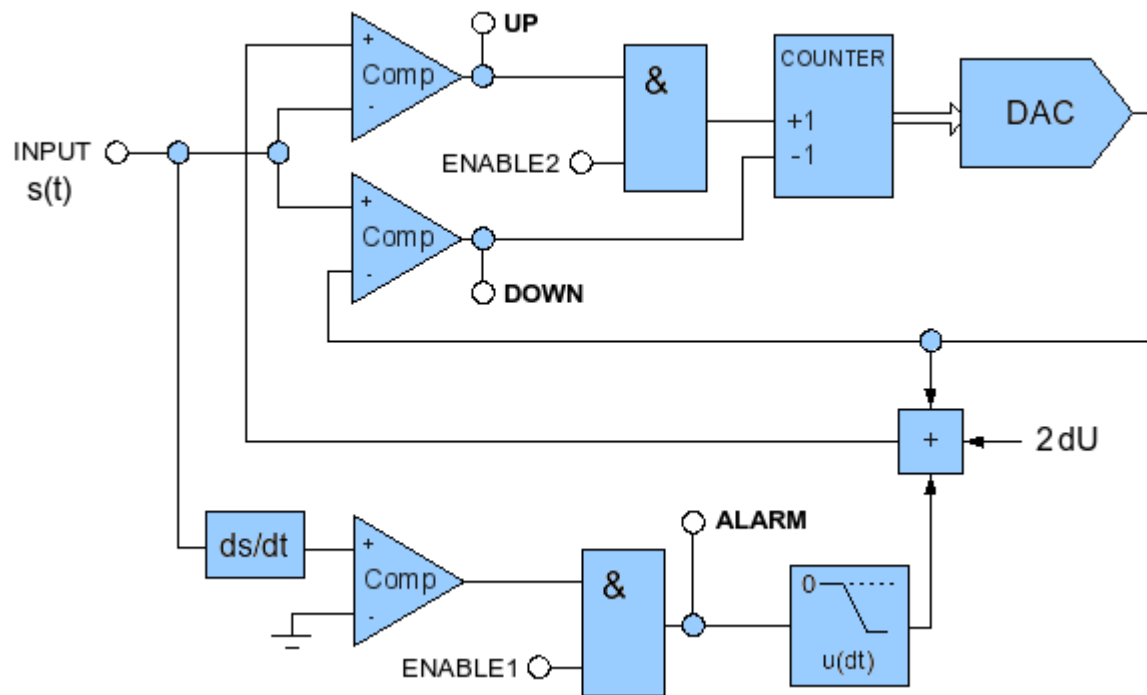
Adaptive level-crossing sampling

- Speech signal of length 3.82 s
- Uniform sampling at 8 kHz provides 30625 samples
- Level-crossing sampling based on SoD scheme with $du=0.01$ V provides 11959 samples (11959 pulses)
- Mini-max sampling provides 19021 samples, if condition $|s_{n+1}-s_n|\geq 6$ mV is set, the number decreases to 6209 samples
- Adaptive level-crossing sampling with $du=0.15$ V provides 18918 samples (37804 pulses), if condition $|s_{n+1}-s_n|\geq 6$ mV is set, the number decreases to 6076 samples (12120 pulses)



Adaptive level-crossing sampling

- Adaptive level-crossing sampling block scheme



Conclusions

- There are different sampling schemes for different applications
- The classical is uniform sampling with sampling rate determined by signal bandwidth
- Due to inefficiencies and limitations of uniform sampling alternative solutions for various applications are developed
- Signal-dependent sampling can be efficiently used in wireless sensor networks since the data acquisition activity depends on signal itself
- ...