FAKULTA
APLIKOVANÝCH VĚD
ZÁPADOČESKÉ
UNIVERZITY
V PLZNI

University of West Bohemia in Pilsen
Department of Computer Science and Engineering
Univerzitni 8
30614 Pilsen
Czech Republic

Short Time Delay Measurement

Ing. Karel Dudáček

Technical Report No. DCSE/TR-2015-03
July, 2015

Distribution: public

Technical Report No. DCSE/TR-2015-03
July 2015

Short Time Delay Measurement

Ing. Karel Dudáček

Abstract

This paper describes the state of the art of the short time delay measurement and proposes new method for the delay measurement.

A brief introduction to the delay measurement and the problem specification are in the first two chapters. Chapter 3 presents the state of the art of the delay measurement. In the next chapter, the evaluation and comparison of selected delay measurement methods are presented. Also the problems of using these methods are described in this chapter. Chapter 5 describes the new method for decreasing the number of the samples necessary for the time delay measurement. Chapter 6 brings a description of the further work and the goals of the Ph.D. thesis.

This work has been supported by the UWB grant SGS–2013–029 Advanced Computer and Information Systems and by the European Regional Development Fund (ERDF) — project NTIS (New Technologies for Information Society), European Centre of Excellence, CZ.1.05/1.1.00/0.2.0090..

Copies of this report are available on
http://www.kiv.zcu.cz/publications/
or by surface mail on request sent to the following address:

University of West Bohemia in Pilsen
Department of Computer Science and Engineering
Univerzitni 8
30614 Pilsen
Czech Republic

Contents

Chapter 1

Introduction

In many applications, precise measurement of short delays between analogue signals is required. A number of methods were developed for this purpose, supposing measurement of signals with sharp edges. However if the signal has no sharp edge, it is common to use some of the analogue methods which are based on various principles. Usage of analogue measurement methods requires highly precise production and calibration of each manufactured piece. More recently, there has been a great tendency towards digital signal processing making product development faster and production cheaper. This work compares existing measurement methods and suggests a new method.

Chapter 2

Problem specification

In many applications, e.g. radar and sonar sensors, laser range-finders etc., it is necessary to measure the delay between two pulses. These devices use signals with a high propagation speed for measurement in shorter and shorter ranges, thus the delay between transmitted impulse and received echo is decreasing. This results in more stringent requirements for precision of time delay measurement between analogue signals. This chapter describes typical features of processed signals and requirements for precision of the measurement methods.

2.1 Signal features

Suppose we have acquired two signals (signal pair). Suppose this signal pair has the following features:

- Signals appear in bursts of finite waveform length,
- both waveforms in the pair have a similar envelope shape, but are time-shifted by Δt,
- bursts are non-periodic (each burst is unique),
- signals in the burst are harmonically oscillating at unknown frequency f_p,
- the analytical description of the signal envelope is not known.

The above-described signal may be obtained from some sort of sensors, for example from ultrasonic detectors. An example of such a waveform can be seen in Figure 2.1.

2.2 Properties of the measurement

Suppose the above-described signal pair is acquired in some appliance regardless of whether it is radar, a laser range-finder or something else. The basic principle of many devices is a measurement of the time shift Δt between the two waveforms in the signal pair with high precision. The measured time shift is usually very short ($\Delta t \approx 10^{-6} \div 10^{-11}\,s$). In many applications, very fine measurement resolution (in the order of $10ps$) is required. Simple correlation methods commonly used for such measurements have a resolution equal to the sampling period. To achieve the required resolution of $10ps$, the sampling frequencies would be unattainably high ($100Gs \cdot s^{-1}$). Thus more advanced methods must be used.

Summary of the properties of the measurement:

- Ability to measure very short time delays ($\Delta t \approx 10^{-6} \div 10^{-11}\,s$).
- Measurement resolution in the order of $10\,ps$.
- High accuracy of measurement (error in the order of 0.1%).

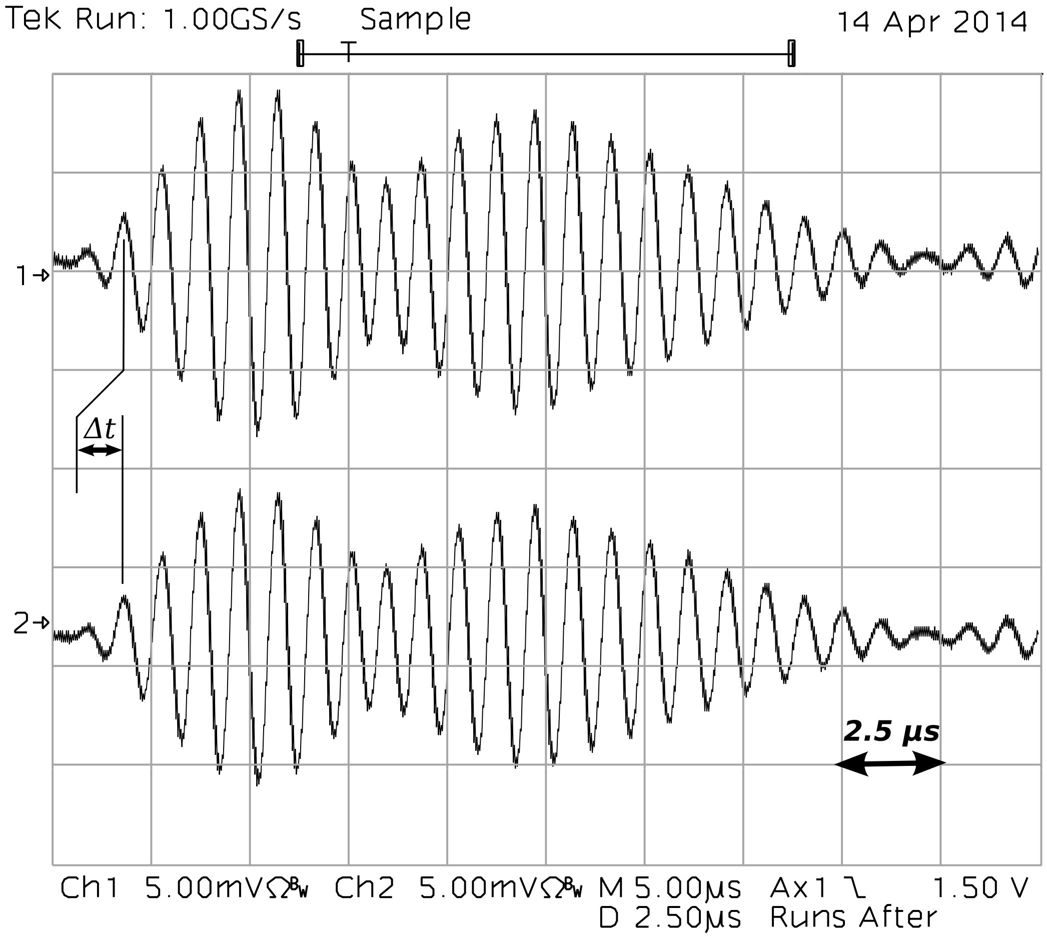

Figure 2.1: Waveform example.

An example of a signal pair recorded on an ultrasonic flowmeter prototype. The time shift is hard to notice because $\Delta t \approx 0.01 \cdot f_p^{-1}$.

Chapter 3

State of the art of delay measurement methods

The problem of the short time delay measurement is very old. Thus many measurement methods were developed in the past. The first methods were fully analogue. Then came methods using analogue preprocessing for digital measurement. In recent years, there is a great tendency to use fully digital signal processing. This chapter describes the state of the art of the delay measurement methods focusing on the digital methods.

3.1 Basic time delay measurement methods

A few basic measurement methods exist for time delay measurement. These methods are easy to use but they are unsuitable for many applications. Basic delay measurement methods are:

- the cross-correlation method,
- the pulse counting method,
- the phase shift method,
- the delay line method.

In this section, there is a brief description of these basic methods.

3.1.1 Cross-correlation method

A common method for signal delay computation is finding the maximum of their cross-correlation function (3.1). This maximum corresponds to the time shift with the best similarity between signals.

$$\begin{aligned} R_{xy}(k) &= \sum_m x(m) \cdot y(k+m) \\ L_{xy}(k) &= k \end{aligned} \tag{3.1}$$

Each point of cross-correlation function $R_{xy}(k)$ is a dot product of discrete signals $x(m)$ and $y(m)$ mutually shifted by k samples. Time shift k is known as a *lag*; dependence of lag k on time (sample number) is linear increasing function $L_{xy}(k)$.

The distance between samples is sampling period $T_s = f_s^{-1}$, where f_s is sampling frequency. The distance between points in the cross-correlation function is also T_s. This means that maximal resolution of this method is equal to the sampling period T_s. Many applications need resolution that corresponds to very high sampling frequencies ($10Gs/s$). In these applications, this method is not usable without modifications providing finer resolution.

3.1.2 Pulse counting method

The second basic time shift measurement method is the pulse counting method. A reference clock generator produces pulses with period T_r. These pulses are routed to the counter. When the first event (e.g. signal transition on signal x) occurs, the counter starts counting reference pulses; when the second event occurs (e.g. transition on signal y), the counter stops counting. The delay between events is $\Delta t = n \cdot T_r$, where n is the counter value.

Maximal measurement error is $E_{max} = 2 \cdot T_r$, mean error is $E = T_r$.

3.1.3 Phase shift method

Let us consider a pair of discrete (attenuated) harmonic signals $x(t)$ and $y(t)$. Both signals have frequency f_p. If phase shift $\Delta\varphi$ is known, it is possible to compute time shift Δt of waveforms using (3.2).

$$\Delta t = \frac{\Delta\varphi}{\omega_p} = \frac{\Delta\varphi}{2 \cdot \pi \cdot f_p} \tag{3.2}$$

The phase shift and frequency f_p can be easily computed using the discrete Fourier transform. The phase shift computed using the Fourier transform is in range $< -\pi, +\pi >$; the time shift is then in range $\left\langle -\frac{1}{2 \cdot f_p}, +\frac{1}{2 \cdot f_p} \right\rangle$.

When using this method, frequency f_p of the signal must be found. It is possible to use a priori knowledge of the frequency or determine the frequency from the signal spectra. When the ratio of sampling frequency f_s and signal frequency f_p is not an integer (3.3), no point in the discrete spectra corresponds to the frequency of the signal, but more points are influenced by this frequency.

$$f_s = k \cdot f_p; \quad k \in \mathbb{N} \tag{3.3}$$

3.1.4 Delay line method

Usage of a delay line is another basic delay measurement method. This method is suitable only for signals with edges or similar synchronization points. One signal is connected to the input of the delay line, the other signal is used as a clock signal for memory elements (e.g. flip-flops). When an event occurs on the first signal, it propagates along the delay line. Propagation is delayed in each element by its propagation delay. When an event occurs on the second signal, it triggers memory elements capturing a state of the delay elements, i.e. the position of the first event can be found. The time difference between events can be computed from events positions and the propagation delay of delay elements. The basic delay line concept can be seen in the Figure 3.1.

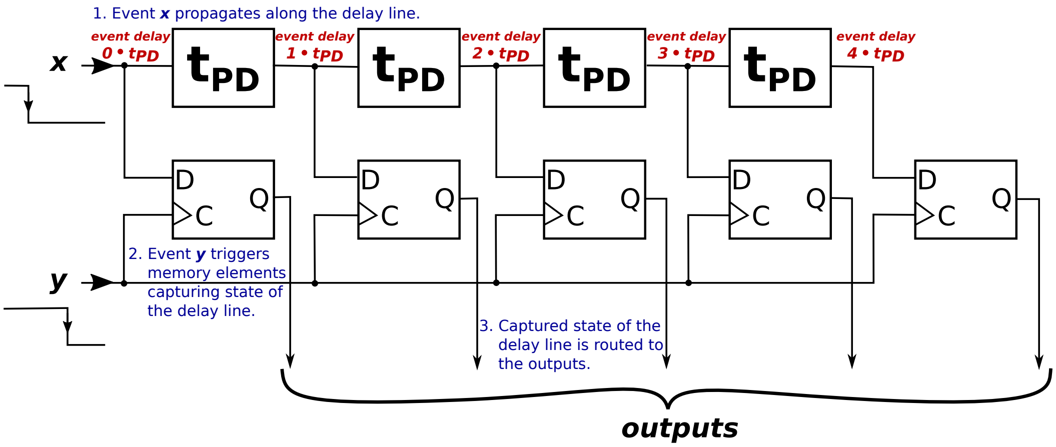
1. Event x propagates along the delay line.
event delay 0 • tPD
event delay 1 • tPD
event delay 2 • tPD
event delay 3 • tPD
event delay 4 • tPD
x
y
tPD
tPD
tPD
tPD
D C Q
D C Q
D C Q
D C Q
D C Q
2. Event y triggers memory elements capturing state of the delay line.
3. Captured state of the delay line is routed to the outputs.
outputs

Figure 3.1: Basic delay line concept.
An event on signal x propagates along the delay line; an event on signal y triggers memory elements (D flip–flops) and captures a state of the delay line.

3.2 Various measurement methods described in the literature

In the literature, many methods for measurement of the time shift of signals are described. In this section, there is a brief description of these methods.

3.2.1 Using a harmonic reference clock for the pulse counting method

Paper [1], describes a method for increasing the precision of the pulse counting method using a harmonic reference clock signal instead of a rectangular clock signal. For a coarse measurement of the time interval between two events, counting reference clock pulses (periods of sinus) is used. The distance between the *start* event and the first pulse (and between the last pulse and the *stop* event) is estimated using measurement of a phase of the reference clock at the moment of the event. This means that delay measurement precision is in fractions of the reference clock period.

3.2.2 Edge transformation using a SAW filter

Paper [2] describes a method for time measurement using a transversal surface acoustic wave (SAW) filter. When signals are sampled, it is problematic to measure the precise position of sharp narrow peaks or transitions because of the sampling theorem (a sharp edge has an infinite spectrum; thus sampling frequency must also be infinite — practically, it means very high). The authors of this paper used a SAW filter for transformation of the sharp edge into finite deterministic oscillations. A SAW filter's oscillations have a narrow spectral line at a known fixed frequency; thus waveforms can be easily sampled and reconstructed from the samples. Due to the known behaviour of the SAW filter, the time of arrival of the excitation pulses can be computed.

3.2.3 Dividing delay line into coarse and fine blocks

The authors of paper [3] presented an improved delay line method for measurement of on-chip signal delays. According to the paper, it is a problem to produce a long on-chip delay line with high precision. The method presented in the paper divides the delay line into a *coarse* block and a *fine* block. When the signal pair arrives, it is routed to the coarse block. In the coarse block, a coarse delay (t_C) between signals as c multiples of coarse block elements' delay T_C is measured. The delay between signals is also reduced by t_C in the coarse block. From the coarse block the signals are routed into the fine block. In the fine block, a fine delay (t_F) between signals as f multiples of fine block elements' delay T_F is measured. The measured delay between signals is computed as a sum of fine and coarse delay $\Delta t = t_C + t_F = c \cdot T_C + f \cdot T_F$. The values c and f are integers not greater than the number of delay elements in the appropriate block.

3.2.4 Multichannel cross-correlation

Paper [4] introduces an improvement of the cross-correlation method . The time resolution of the basic cross-correlation method is equal to the sampling period T_s. The authors of this paper increased the resolution of the method to T_s/N where N is a small integer.

One of the signals (x) in the pair is directly sampled, the other signal (y) is routed to an analogue delay line with $N - 1$ stages. The delay of each stage is T_s/N. The original signal and outputs of all stages are sampled giving N delayed waveforms y_n where n is the number of delay stages in the path ($n = 0, 1, \ldots, N - 1$). Time shifts Δt_n are computed using the correlation method. Linear approximation function $\widehat{\Delta t}(n)$ is fitted using the least squares method and final time shift Δt is computed as $\widehat{\Delta t}(0)$.

3.2.5 Interpolation methods for the cross-correlation method

Paper [5] describes four interpolation methods for resolution improvement of the cross-correlation method . In the article, following interpolation methods are described:

- Parabolic fitting with bias compensation.
- Parabolic-fit interpolation combined with linear filter interpolation.
- Parabolic interpolation to the complex correlation function envelope.
- Matched filtering for interpolation.

3.2.6 Design of an optimal filter for the cross-correlation method

The authors of paper [6] derived an optimal filter for use in cross-correlators. They derived a mathematical model of cross-correlator for measurement of the time shift between two signals in a noisy environment. According to the model, two optimization criteria were found:

- The ratio between the cross-correlation function peak and background noise must be maximized at a position (lag) corresponding to the time shift of the signals.
- The mean-square difference between the original signal (signal before addition of noise) and the filtered signal must be minimized.

Using these two criteria, the paper's authors derived an optimal filter for filtering the cross-correlation function in order to minimize the influence of the noise on the peak detector.

3.2.7 Time delay estimation based on SINC filtering

Papers [7] and [8], describe a method for time delay estimation using a *sinc* function coefficients estimation. The papers describe the relation between two delayed signals x_k and y_k as a function $y_k = \sum x_{k-n} \cdot \mathrm{sinc}(n + m + a)$. The authors have shown that the delay between signals is equal to the peak position (in time axis) of the *sinc* function. According to paper [7], this peak position can be estimated using an LMS adaptive algorithm.

3.2.8 Signal filtering using an auto-adjusting Wiener filter

The authors of paper [9] used a Wiener filter for reconstruction of a noisy signal from an ultrasonic receiver. For adjusting the Wiener filter, it is necessary to the know system's impulse response and the power spectra of the signal and noise. The authors of the paper used the part of the received signal with high amplitude for measurement of the signal parameters and the part of the signal with low amplitude for measurement of noise parameters. The impulse response of the system is obtained from the transmitter—receiver acoustic crosstalk.

3.2.9 Cross-wavelet transformation

The method described in paper [10] uses wavelet and cross-wavelet transforms for time shift computation. Received signals $x(s,\tau)$ and $y(s,\tau)$ are transformed from the time domain into the wavelet domain using wavelet transform ψ giving their wavelet domain images $X_\psi(s,\tau)$ and $Y_\psi(s,\tau)$. In the wavelet domain the image of their cross-wavelet function $C_\psi = X_\psi \cdot Y_\psi^*$ is computed. The cross-wavelet function is transformed back to the time domain (i.e. we get $c(s,\tau)$) and its maximum $c_{max}(s_{max},\tau_{max})$ is found. The time shift between signals corresponds to τ_{max} and the ratio between amplitudes of signals corresponds to s_{max}.

3.2.10 Third order cumulant method

Paper [11], describes two time delay computation methods: the *Third order cumulant method* and the *Bispectrum method*[1]. The third cumulant of the normal distribution is zero. The third order cumulant method computes third order cumulant $C_{xxx}(m_1,m_2)$ of signal $x(t)$ and cross-cumulant $C_{xxx}(m_1,m_2)$ of signals $x(t)$ and $y(t) = x(t-\tau)$ (i.e. τ is the time shift between signals). These cumulants do not contain white Gaussian noise. Then the cumulants are sliced at $m_2 = 0$ and the cross-correlation of the slices is computed $R(m_1) = \text{xcorr}\{C_{xxx}(m_1,0),\ C_{xyx}(m_1,0)\}$. The maximum of the cross-correlation function is at the position $m_1 = \tau$, i.e. $\max\{R(m_1)\} = R(\tau)$.

3.2.11 Bispectrum method

The Bispectrum method is the second method described in paper [11]. The bispectrum is the Fourier transform of the third order cumulant (it suppresses white noise). According to this paper, the bispectra $B_{xxx}(\omega_1,\omega_2) = \mathcal{FFT}\{C_{xxx}(m_1,m_2)\}$ and $B_{xyx}(\omega_1,\omega_2) = \mathcal{FFT}\{C_{xyx}(m_1,m_2)\}$ are computed. From these bispectra function $T(t) = \iint (B_{xyx}/B_{xxx})\, d\omega_1\, d\omega_2$ is computed, having its peak at position $t = \tau$ (signals' time shift).

[1] The bispectrum method is described in subsection 3.2.11.

3.3 Summary of the methods.

The previous two sections briefly describe various time delay estimation methods whether well–known or described in literature. These methods can be divided into two groups:

1. Methods for measuring time delay between sharp edges in the signals.
2. Methods for measuring time delay between signals without sharp edges.

Methods described in subsections 3.1.2, 3.1.4, 3.2.1, 3.2.3 belong to the first group. These methods are intended for measurement of delay between digital signals. For measurement of delay between analogue signals they are suitable only in a limited number of cases. These methods are useless for processing signals described in Chapter 2. The method described in subsection 3.2.2 belongs to this group, but only partially, because it converts a problem of measuring delay between sharp edges to the problem belonging to the other group.

Methods described in subsections 3.1.1, 3.1.3, 3.2.4, 3.2.5, 3.2.6, 3.2.7, 3.2.8, 3.2.9, 3.2.10 and 3.2.11 belong to the second group. Method 3.2.2 partially belongs to this group too. These methods are intended for measurement of delay between analogue signals.

Many of these methods are based on the basic cross-correlation method , improving some of its features. The basic cross-correlation method has two main disadvantages. The first disadvantage of this method is coarse time resolution and the second disadvantage is the problem with precise peak identification when the cross-correlation function is flat and signals are noisy.

- Time resolution of the basic cross-correlation method is equal to the sampling period of processed signals. Methods described in subsections 3.2.4, 3.2.5 and 3.2.2 improve the time resolution of the cross-correlation method in various ways. Method 3.2.4 uses an analogue delay line and multiple AD converters, the methods described in 3.2.5 use various interpolation methods and method 3.2.2 uses computation in a continuous time domain.
- When the amplitude of the processed signals changes slowly, their cross-correlation function has a flat peak. Then the maximum detection is difficult and can be easily influenced by noise. Methods described in subsections 3.2.6, 3.2.8, 3.2.9 and 3.2.10 enhance the cross-correlation function in order to get sharper maxima.

Method 3.2.9 uses a wavelet transform instead of the Fourier transform to compute cross-correlation. The main idea behind this method is a description of the signal using functions more similar to the signal (in paper [10] short oscillating impulses are processed) than sinusoids are.

Methods not based on the cross-correlation method are phase shift method 3.1.3, *sinc* filtering method 3.2.7 and bispectrum method 3.2.11.

3.4 Methods based on cross-correlation

The basics of the cross-correlation method are described in subsection 3.1.1, but for a better understanding of the main advantages and drawbacks of this method and its improved versions, it is necessary to provide a wider introduction to cross-correlation methods.

3.4.1 Continuous time cross-correlation method

Let us consider a pair of time-shifted continuous time signals (3.4). Their cross-correlation function is in (3.5). The time shift between these signals can be computed by maximizing the cross-correlation function. Substituting (3.4) into (3.5) we get (3.6).

$$\begin{aligned} &x(t) \\ &y(t) = x(t-\Delta t) \end{aligned} \tag{3.4}$$

$$\begin{aligned} R_{xy}(\tau) &= \int_{-\infty}^{+\infty} x(t)\cdot y(t+\tau) \quad dt \\ L_{xy}(\tau) &= \tau \end{aligned} \tag{3.5}$$

$$R_{xy}(\tau) = \int_{-\infty}^{+\infty} x(t)\cdot x(t+\tau-\Delta t) \quad dt \tag{3.6}$$

The maximum of the cross-correlation function R_{xy} is at the position defined by equation (3.7). Thus the time shift between signals can be found as an argument of the cross-correlation function at its maximum (3.8).

$$\max\{R_{xy}\} = \int_{-\infty}^{+\infty} x^2(t)dt = \int_{-\infty}^{+\infty} x(t)\cdot x(t+\underbrace{\tau-\Delta t}_{=0}) \quad dt \tag{3.7}$$

$$\Delta t = \arg\left\{\max_{\tau}\{R_{xy}(\tau)\}\right\} = \arg\left\{\max_{\tau}\left\{\int_{-\infty}^{+\infty} x(t)\cdot x(t+\underbrace{\tau-\Delta t}_{=0}) \quad dt\right\}\right\} \tag{3.8}$$

If the signals x and y are zero outside the interval $\langle A, B\rangle$, the cross-correlation function can be normalized (3.9).

$$R_{xy}(\tau) = \frac{1}{(B-A)}\int_{2A-B}^{2B-A} x(t)\cdot x(t+\tau-\Delta t) \quad dt \tag{3.9}$$

If the signals are noisy, equation (3.9) can be rewritten as (3.10), where $r(t)$ and $s(t)$ are noises with zero mean value ($\mathrm{E}(s) = 0$ and $\mathrm{E}(r) = 0$).

$$\widehat{R_{xy}}(\tau) = \frac{1}{(B-A)}\int_{2A-B}^{2B-A} (x(t)+s(t))\cdot(x(t+\tau-\Delta t)+r(t+\tau)) \quad dt \tag{3.10}$$

If the signal length increases $(B-A)\to\infty$, the cross-correlation function of the noisy signals approaches the cross-correlation function of the original signals $\widehat{R_{xy}}(\tau)\to R_{xy}(\tau)$.

$$\begin{aligned} \widehat{R_{xy}}(\tau) \quad &= \tfrac{1}{(B-A)}\int_{2A-B}^{2B-A} x(t)\cdot x(t+\tau-\Delta t) \quad dt + &&\overset{(B-A)\to\infty}{\longrightarrow} R_{xy}(\tau) \\ &+\tfrac{1}{(B-A)}\int_{2A-B}^{2B-A} x(t)\cdot r(t+\tau) \quad dt + &&\overset{(B-A)\to\infty}{\longrightarrow} 0 \\ &+\tfrac{1}{(B-A)}\int_{2A-B}^{2B-A} x(t+\tau-\Delta t)\cdot s(t) \quad dt + &&\overset{(B-A)\to\infty}{\longrightarrow} 0 \\ &+\tfrac{1}{(B-A)}\int_{2A-B}^{2B-A} s(t)\cdot r(t+\tau) \quad dt &&\overset{(B-A)\to\infty}{\longrightarrow} 0 \end{aligned} \tag{3.11}$$

3.4.2 Discrete time cross-correlation method

If the signal is sampled with sampling period T_s, the equations in (3.4) change to (3.12).

$$\begin{aligned} x(t) &= x(n \cdot T_s) \\ y(t) &= x(t - \Delta t) = x(n \cdot T_s - \Delta t) \\ n &= -\lfloor N/2 \rfloor, \ldots -2, -1, 0, 1, 2, \ldots, \left\lfloor \frac{N-1}{2} \right\rfloor \end{aligned} \tag{3.12}$$

The cross-correlation function of sampled signals is in (3.13). Substituting (3.12) to (3.13), we get (3.14). The n is the sample number; N is the number of samples of each signal. If the signal is infinite, $N \to \infty$.

$$\begin{aligned} R_{xy}(k \cdot T_s) &= \sum_n x(n \cdot T_s) \cdot y((n+k) \cdot T_s) \\ L_{xy}(k \cdot T_s) &= k \cdot T_s \\ k &= -(N-1), \ldots, -2, -1, 0, 1, 2, \ldots, (N-1) \end{aligned} \tag{3.13}$$

$$R_{xy}(k \cdot T_s) = \sum_n x(n \cdot T_s) \cdot x(n \cdot T_s + k \cdot T_s - \Delta t) \tag{3.14}$$

The time shift between sampled signals can be computed using (3.15) (modification of (3.8) for sampled signal). The position of the maximum of the cross-correlation function of sampled signals corresponds to the position of the maximum of the cross-correlation function of original (i.e. continuous) signals only if the time shift between signals is equal to the integer multiple of the sampling period (3.16).

$$\widehat{\Delta t} = \arg\left\{ \max_{k \cdot T_s} \{R_{xy}(k \cdot T_s)\} \right\} = \arg\left\{ \max_{k \cdot T_s} \left\{ \sum_n x(n \cdot T_s) \cdot x(n \cdot T_s + \underbrace{k \cdot T_s - \Delta t}_{\substack{=0 \text{ if} \Delta t = m \cdot T_s \\ m \in \mathbb{Z} \\ \neq 0 \text{ otherwise}}}) \right\} \right\} \tag{3.15}$$

$$\widehat{\Delta t} \begin{cases} = \Delta t & \text{if} \quad \Delta t = m \cdot T_s; \quad m \in \mathbb{Z} \\ \in \langle \Delta t - T_s, \Delta t + T_s \rangle & \text{otherwise} \end{cases} \tag{3.16}$$

It is possible to normalize the cross–correlation function of sampled signals (3.17).

$$R_{xy}(k \cdot T_s) = \frac{1}{N} \sum_n x(n \cdot T_s) \cdot x(n \cdot T_s + k \cdot T_s - \Delta t) \tag{3.17}$$

3.4.3 Cross-correlation of the harmonic functions

If the signals $x(n)$ and $y(n)$ are harmonic functions with angular frequency ω (3.18), then their cross-correlation function is a harmonic function with the same frequency (3.19).

$$\begin{aligned} x(n) &= A \cdot \sin(\omega n) \\ y(n) &= B \cdot \sin(\omega n + \omega \underbrace{\frac{\Delta t}{T_s}}_{\overset{s}{=} \Delta n}) = B \cdot \sin(\omega(n + \Delta n)) \end{aligned} \tag{3.18}$$

$$
\begin{aligned}
R_{xy}(k) &= \sum_n x(n)\cdot y(n+k) = \sum_n A\cdot\sin(\omega n)\cdot B\cdot\sin(\omega(n+k+\Delta n)) = \\
&= AB\sum_n \sin(\omega n)\cdot\sin(\omega(n+k+\Delta n)) = \\
&= \underbrace{\frac{AB}{2}}_{\overset{\mathrm{s}}{=}C}\sum_n\left(\underbrace{\cos(\omega\Delta n-\omega k)}_{\text{const. in } n}-\cos(2\omega n+\omega k-\omega\Delta n)\right) = \\
&= CN\cos(\omega k-\omega\Delta n) - C\sum_n\cos(\omega k-\omega(2n+\Delta n)) = \\
&= CN\cos(\omega k-\underbrace{\omega\Delta n}_{\psi}) - C\cos(\omega k-\underbrace{\omega(2n_0+\Delta n)}_{\varphi_0})- \\
&\quad - C\cos(\omega k-\underbrace{\omega(2n_1+\Delta n)}_{\varphi_1}) - \cdots \\
&\quad \cdots - C\cos(\omega k-\underbrace{\omega(2n_{N-1}+\Delta n)}_{\varphi_{N-1}}) = \\
&= -\frac{C}{2}\Big(e^{j(\omega k+\varphi_0)}+e^{-j(\omega k+\varphi_0)}+e^{j(\omega k+\varphi_1)}+e^{-j(\omega k+\varphi_1)}+\cdots \\
&\quad \cdots + e^{j(\omega k+\varphi_{N-1})}+e^{j(\omega k+\varphi_{N-1})}-Ne^{j(\omega k+\psi)}-Ne^{j(\omega k+\psi)}\Big) = \\
&= -\frac{C}{2}\,e^{j\omega k}\underbrace{\left(e^{j\varphi_0}+e^{j\varphi_1}+\cdots+e^{j\varphi_{N-1}}-Ne^{j\psi}\right)}_{\overset{\mathrm{s}}{=}D} - \\
&\quad -\frac{C}{2}\,e^{-j\omega k}\underbrace{\left(e^{-j\varphi_0}+e^{-j\varphi_1}+\cdots+e^{-j\varphi_{N-1}}-Ne^{-j\psi}\right)}_{\overset{\mathrm{s}}{=}E} = \\
&= -\frac{C}{2}\left(De^{j\omega k}-Ee^{-j\omega k}\right) = -\frac{C}{2}\left((D+E)\cos(\omega k)+j(D-E)\sin(\omega k)\right) = \\
&= \underbrace{C\left(N\cos(\omega\Delta n)-\sum_n\cos(2\omega n+\omega\Delta n)\right)}_{\overset{\mathrm{s}}{=}F\ \text{(constant in } k\text{)}}\cos(\omega k) + \\
&\quad \underbrace{C\left(N\sin(\omega\Delta n)-\sum_n\sin(2\omega n+\omega\Delta n)\right)}_{\overset{\mathrm{s}}{=}G\ \text{(constant in } k\text{)}}\sin(\omega k) = \\
&= \frac{AB}{2}\left(F\cos(\omega k)+G\sin(\omega k)\right) \qquad A,B,F,G\in\mathbb{R}
\end{aligned}
\tag{3.19}
$$

If the signals have finite lengths, the cross-correlation function is a harmonic function modulated by the cross-correlation function of their envelopes (it is trapezoid for rectangular envelopes because of the summation of different numbers of signal points at different positions on k axis).

When the time shift computation is done according to equation (3.15), the value $\widehat{\Delta t}$ occurs at any point in the interval $\langle\Delta t-\frac{T_s}{2},\ \Delta t+\frac{T_s}{2}\rangle$ with same probability (always valid because (3.20) holds for all cross-correlation functions).

$$
\begin{aligned}
f(x_{max}+\Delta x) &= f(x_{max}-\Delta x) \\
x_{max} &= \arg\left\{\max_x\left\{f(x)\right\}\right\} \\
\Delta x \leq \varepsilon &\quad \varepsilon\in\mathbb{R}_+
\end{aligned}
\tag{3.20}
$$

Thus the more precise estimation of value Δt should be done using knowledge of the shape of cross-correlation function and using more points of the cross-correlation function than only $\widehat{\Delta t}$. Some methods for such estimations were described in the literature and briefly described in subsections 3.2.4, 3.2.5 and 3.2.2.

3.4.4 Computation complexity of the cross-correlation method

Computation of the cross-correlation function according to the definition in 3.13 requires $N \cdot K = N(2N-1) = 2N^2 - N$ of the multiplications and the same number of summations. The localization of the maximum requires $K = 2N - 1$ comparisons. Thus the computation complexity of the cross-correlation method is $O(N^2)$.

The cross-correlation function can be also computed using a fast algorithm (3.21). Computation complexities of FFT and IFFT are $O(N \cdot \log N)$, thus the computation complexity of the cross-correlation method is decreased to $O(N \cdot \log N)$.

$$\begin{aligned} R_{xy}(t) &= \mathcal{IFFT}\{\mathbf{R_{xy}}(f)\} \\ \mathbf{R_{xy}}(f) &= \mathbf{X}(f) \cdot \mathbf{Y}^*(f) \\ \mathbf{X}(f) &= \mathcal{FFT}\{X(t)\} \\ \mathbf{Y}(f) &= \mathcal{FFT}\{Y(t)\} \\ Y^* & \text{ is the complex conjugate of } Y \end{aligned} \tag{3.21}$$

3.4.5 Maxima detection

The problem of localization of maxima of the discrete cross-correlation function seems to be easy, but when sampling frequency decreases to the Nyquist limit, it became to be tricky.

Let us consider hypothetical cross-correlation function described by (3.22). This function is used just for demonstration; any real cross-correlation function of finite length signals is more complicated.

$$x(t) = t \cdot sin(\omega t) \tag{3.22}$$

If this function is continuous, the value of any local maximum (3.23) of this function is greater than the values of all previous local maxima (3.24).

$$\begin{aligned} x_{max} &= \max_D\{x(t)\} = x(t_{max}) \\ D &= (-\infty, \tau\rangle \\ \tau &\in \mathbb{R} \\ \text{i.e. } t_{max} &= \arg\{x_{max}\} \end{aligned} \tag{3.23}$$

$$\begin{aligned} x_{max} &> \max_{D_\varepsilon}\{x(t)\} \\ D_\varepsilon &= (-\infty, t_{max}) \end{aligned} \tag{3.24}$$

If this function is sampled, i.e. (3.25), the relation (3.24) isn't valid for some points, even if the Nyquist's condition is met.

$$x(n) = nT_s \cdot sin(\omega n T_s) \quad n \in \mathbb{Z} \tag{3.25}$$

The violation is unavoidably caused by sample timing if the sampling frequency is not an integer multiple of the signal frequency. See Figure 3.2. If the sampling frequency $1/T_s$ decreases, the relation (3.24) is violated more and more often. Thus the sampling frequency should be much higher than Nyquist's limit or the localization of the maxima of the cross-correlation function cannot be done using a basic search for the highest value of the cross-correlation function .

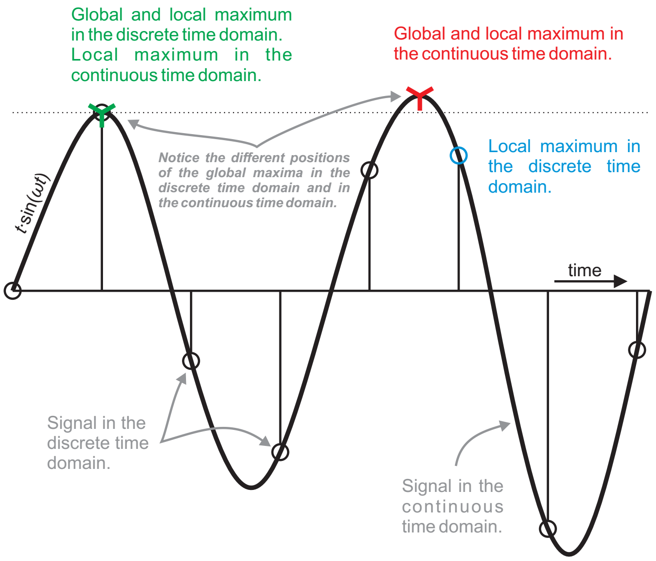
Global and local maximum in the discrete time domain. Local maximum in the continuous time domain.
Global and local maximum in the continuous time domain.
Notice the different positions of the global maxima in the discrete time domain and in the continuous time domain.
Local maximum in the discrete time domain.
t·sin(ωt)
time
Signal in the discrete time domain.
Signal in the continuous time domain.

Figure 3.2: False peak detection
Local and global maxima of the signal $x(t) = t \cdot sin(\omega t)$. Notice the different positions of global maxima for the continuous time domain and the discrete time domain.

3.5 Summary of Chapter 3

In the literature, many methods for short time delay measurement are described. When the methods requiring sharp edges in the signal are eliminated, the rest of the methods have the following common features:

1. Requires high sampling frequency (in the sense of multiples of signal bandwidth). Thus high speed AD converters are required, e.g. for sampling of a 1 MHz signal, 20 MS/s AD converter is necessary.
2. When the signals are sampled, a large amount of data is produced and must be stored or transmitted for processing; thus high speed communication is required.
3. For real-time computations fast computation units (e.g. digital signal processors) are required.

Chapter 4

Experimental evaluation of methods

For better evaluation of the features of various methods described, several methods were implemented and their features were evaluated and compared under various conditions. As input signals for the method comparison, a pair of the signals imitating typical signals from the real appliance were used.[1]

In this section following methods are compared:

1. interpolation followed by correlation,
2. approximation of the cross-correlation function,
3. time shift computation from the phase shift,
4. signal approximation using the least squares method and statistical evaluation.

4.1 Description and evaluation of the methods

In this section, details of the evaluated time delay measurement methods are described and different versions of these methods are compared for various sampling frequencies and noise levels.

A numeric method evaluation was carried out using the above-described signals. The main features of the simulated signal are in accordance with the signal features described in the Chapter 2.1, $f_p = 1\,MHz$. White Gaussian noise at various signal to noise ratios was added to the simulated signal in order to evaluate the method's noise sensitivity.

Time shift computations for various time shifts were carried out for many times (5 to 1000 times, depending on computation complexity) and errors were statistically evaluated. The computations were carried out in double precision IEEE 754 floats.

Two different error behaviours were observed regarding dependence between time shift and error magnitude:

- The error depends only on the method's parameters. In the following chapters the error is considered as absolute (4.1).
- The error depends on the method's parameters and the time shift. In the following chapters the error is considered as relative (4.2).

[1] The amplitude of the signal is exponentially increases and then exponentially decreases.

$$E_{abs} = \widehat{\Delta t} - \Delta t \tag{4.1}$$

$$E_{rel} = \left| \frac{\widehat{\Delta t} - \Delta t}{\Delta} t \right| \tag{4.2}$$

The results for various combinations of parameters are compared in the following statements and in the next chapter (Chapter 4.2) the methods are compared to one another.

4.1.1 Interpolation followed by correlation

This method is based on the common correlation method. Analogue signals are sampled (sampling period is less than the required resolution of the result) and interpolated to the required sampling rate. The common correlation method is applied to these interpolated (i.e. up-sampled) signals .

The interpolation ratio can be computed from equation (4.3) where f_s is the sampling frequency of the original signal, f_{interp} is the sampling frequency of the interpolated signal and res is the required resolution.

$$\frac{1}{n} = \frac{f_s}{f_{interp}} = res \cdot f_s \tag{4.3}$$

The interpolation can be carried out in different ways:

- Lagrange's (Newton's) polynomial interpolation,
- Hermitian polynomial interpolation,
- interpolation using zero–value samples insertion and filtration,
- spline interpolation.

Because the interpolation ratio can be very high, the usability of this method is limited due to very high computation complexity.

Interpolation methods comparison

Comparison of the following interpolation methods was carried out: interpolation using zero-valued samples insertion and filtration [12], linear interpolation and spline interpolation using natural cubic spline [13] and pchip spline [14]. Lagrangian and Hermitian interpolation are not suitable for interpolating oscillating signals due to their properties.

The interpolation method using zero-valued samples insertion and filtration was found to get much better results than the other methods, see Figure 4.1.

Effect of filtration

The effect of signal filtration was analysed. As signal disturbed by white Gaussian noise at various levels was filtered using a bandpass filter with cut-off frequencies $0.5 \cdot f_p$ and $2 \cdot f_p$. The signals were filtered before or after interpolation. Experiments confirmed that filtration after interpolation has no effect on results, see Figure 4.2. When filtration is carried out before interpolation, error magnitude increases rapidly.

4.1.2 Approximation of the correlation function

This method is another modification of the correlation method increasing its resolution. As signal pair is sampled and then cross-correlation of signals is computed. The correlation function is approximated using the least square method near its maximum. The maximum

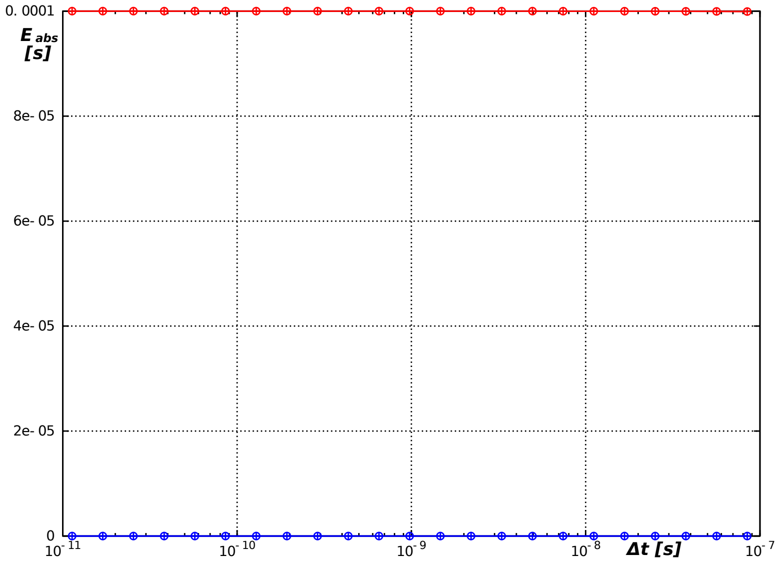

Figure 4.1: Interpolation methods comparison.

Absolute computation error for various interpolation methods, $snr = 20\,dB$, $f_s = 50\,Msample/s$. Interpolation using zero-valued samples insertion and filtration; interpolation using natural cubic and pchip splines.

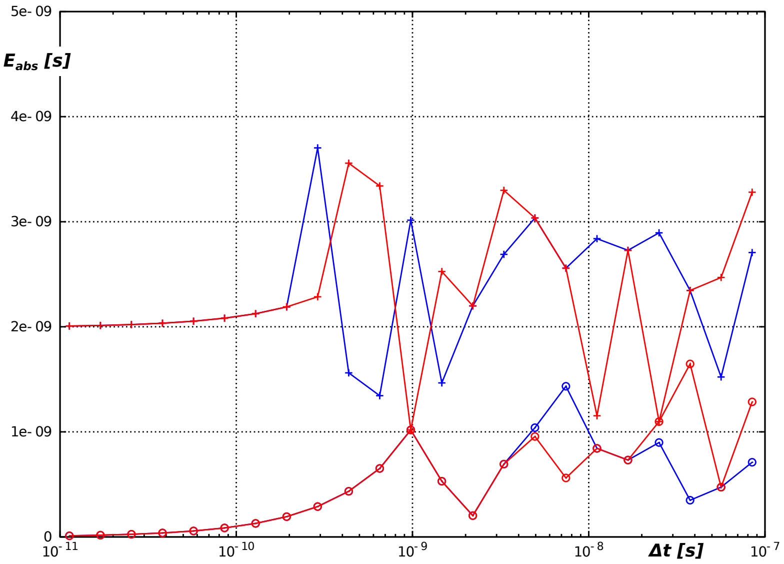

Figure 4.2: Effect of filtration after interpolation.

Absolute computation error for method of interpolation followed by correlation. Filtration after interpolation, $f_p = 1\,MHz$, passband $0.5 \cdot f_p$ to $2 \cdot f_p$, $snr = 20\,dB$, $f_s = 5\,Msample/s$.
Without filtration in blue, with filtration in red.
o . . . mean error, + . . . maximal error

of the approximation function is analytically computed and corresponds to the time shift between signals in the pair.

The signal (maximal frequency f_{max}) is sampled at sampling frequency f_s ($f_s > 2 \cdot f_{max}$) and the correlation function R_{xy} is computed according to the equation (3.1). Its maximum $R_{max} = \max(R_{xy})$ and corresponding lag L_{max} are found. The lag corresponds to the time shift with resolution T_s. In the vicinity of L_{max} the correlation function is approximated by function $\widehat{R}_{xy}$ and its maximum $\widehat{R}_{max} = \max(\widehat{R}_{xy})$ is found. Having corresponding lag $\widehat{L}_{max}$ time shift $\widehat{\Delta t}$ can be computed with a resolution much higher than T_s, see equation (4.4).

For approximation of the correlation function it is advantageous to use a polynomial approximation using the least squares method (at least in the case of cross-correlation of signals specified in Chapter 2.1). In the neighbourhood of correlation maxima, n points in both directions are picked up. Coordinates of these points are $[L_{max-n}, R_{max-n}]$, …, $[L_{max}, R_{max}]$, …, $[L_{max+n}, R_{max+n}]$. These points are polynomially fitted by polynomial $P_m(L)$ of degree m (in the least-squares sense). Local maxima of this polynomial can be computed analytically. Local maximum $[\widehat{L}_{max}, \widehat{R}_{max}]$ nearest (in the sense of $\min|L_{max} - \widehat{L}_{max}|$) to the cross-correlation function maximum $[L_{max}, R_{max}]$ determines the time shift $\widehat{\Delta t}$, see equation (4.4).

$$\widehat{\Delta t} = \hat{L}_{max} \cdot T_s = \frac{\hat{L}_{max}}{f_s} \tag{4.4}$$

Appropriate degree of the approximation polynomial

The appropriate number of points n for polynomial fitting depends on the degree of the polynomial (m). If the quadratic curve is fitted, it is a good choice to fit it between inflexion points of the correlation function. For the harmonic function, the inflexion points are situated at zero crossings. In this case, n is given by equation (4.5). For a higher polynomial degree, more points can be used.

$$n = \left\lfloor \frac{f_s}{4 \cdot f_p} \right\rfloor \tag{4.5}$$

Increasing the polynomial degree also brings increasing ripple and approximation error in the intervals between nodes (samples). An extreme case arises when $m + 1$ points ($n = m/2$) are fitted with a polynomial of degree m. In this case the least square method changes into Lagrange interpolation with all its consequences.

Experimental comparison of approximation polynomials

Comparison of results has been done for approximation of the cross-correlation function by polynomials of degree two and four. A polynomial of degree two gives better results for some combinations of sampling frequency and noise level and a polynomial of degree four is better for the other combinations (see Figure 4.3 and 4.4). The results of the experiment are summarised in Table 4.1.

Effect of filtration

The effect of signal filtering has been evaluated using low-pass filter with cut-off frequency $2 \cdot f_p$ (for definition of f_p, see Chapter 2.1). It was found that filtration has no effect on computation error.

4.1.3 Signal approximation using the least squares method and statistical evaluation

Sampled signals are divided into small elements (e.g. half–waves) and each element is approximated using the least square method. Time shifts are analytically computed for corre-

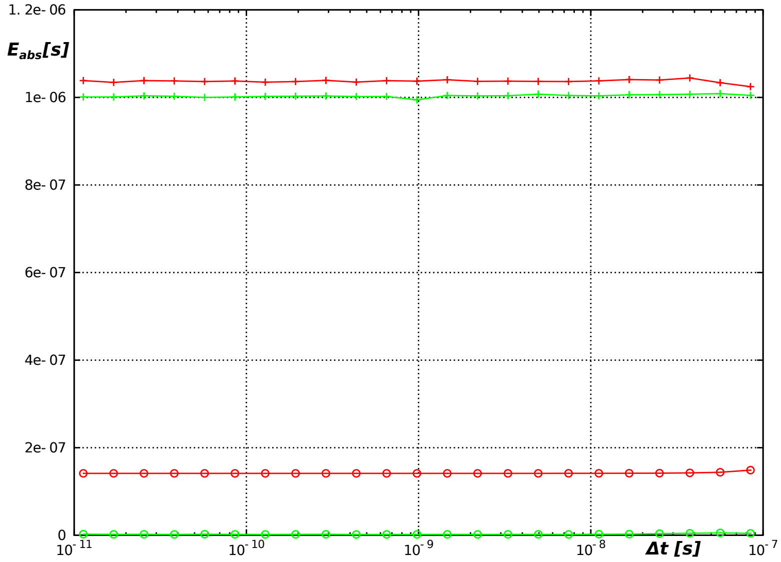

Figure 4.3: Computation error — approximation of correlation function.
Absolute computation error for method of approximation of correlation function, $snr = 10\,dB$, $f_s = 5\,Msample/s$.
Polynomial of degree two in green, of degree four in red.
o . . . mean error, + . . . maximal error

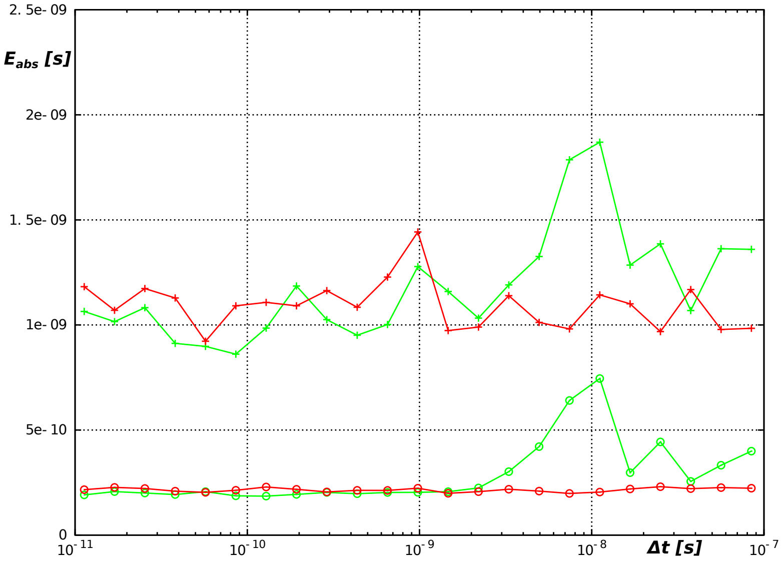

Figure 4.4: Computation error — approximation of correlation function.
Absolute computation error for method of approximation of correlation function, $snr = 20\,dB$, $f_s = 50\,Msample/s$.
Polynomial of degree two in green, of degree four in red.
o . . . mean error, + . . . maximal error

Table 4.1: Appropriate degree of approximation polynomial.

Appropriate degree of approximation polynomial for method of approximation of correlation function for various sampling frequencies and noise levels. $f_p = 1\,MHz$

	$50\,Ms/s$	$10\,Ms/s$	$5\,Ms/s$
Without noise	4	4	2
$snr = 20\,dB$	4	4	2
$snr = 10\,dB$	2	2	2
$snr = 0\,dB$	2	x	x

2 or 4 . . . appropriate degree of polynomial is 2 or 4;
x . . . differences between degrees 2 and 4 are insignificant.

sponding pairs of elements. This set of elementary time shifts is statistically evaluated.

Let us consider a pair of time shifted signals, x_1 and x_2. Both of them are divided in some way to n elements $x_{11}, x_{12}, \ldots, x_{1n}$ and $x_{21}, x_{22}, \ldots, x_{2n}$. Each element is approximated (in the sense of least squares) by an appropriate function (a polynomial is a good choice); we get n functions for each signal $\widehat{x}_{11}, \widehat{x}_{12}, \widehat{\ldots}, \widehat{x}_{1n}$ and $\widehat{x}_{21}, \widehat{x}_{22}, \ldots, \widehat{x}_{2n}$. For each pair of corresponding functions, their time shift $\widehat{\Delta t}_1 \cdots \widehat{\Delta t}_n$ is computed. The algorithm for computation of time shifts depends on the approximation function used; for a quadratic approximation it may be the distance between peaks, for linear approximation it may be the distance between zero-crossings, etc. The set of elementary time shifts $\{\widehat{\Delta t}_i\}$ is statistically evaluated in order to get final time shift $\widehat{\Delta t}$. Basic methods of statistical evaluation are computations of the arithmetic mean or median of the set $\{\widehat{\Delta t}_i\}$.

Experimental comparison of approximation polynomials

This method was evaluated in versions using polynomials of degree two or four as an approximation function $\widehat{x_{ik}}$. For a quadratic function the time shift $\widehat{\Delta t_k}$ was computed as a distance of vertexes in t–axis. Local extrema were found for polynomials of degree four. Time shifts (in t–axis) were computed for extrema laying at particular intervals. These time shifts were used as $\widehat{\Delta t_k}$.

The comparison of both versions of the method can be found in Table 4.2. In the table, it can be seen that the polynomial of degree of two is better then the one of degree of four, with the exception of noiseless signals sampled at low frequency. Dependence of computation error on sampling frequency can be seen in the Figure 4.5.

Statistical evaluation of elementary time shifts

For statistical evaluation of the set of time shifts $\left\{\widehat{\Delta t_1}, \widehat{\Delta t_2}, \ldots, \widehat{\Delta t_n}\right\}$, the arithmetic mean and median were used. Computation error is many times $(10^2 - 10^{10})$ higher when using the arithmetic mean than when using the median.

4.1.4 Phase shift methods

In this subsection, there two variants of the method for time shift computation from the phase shift are compared; the basic phase shift method (see 3.1.3) and the phase shift method with phase interpolation (described below).

Table 4.2: Appropriate degree of approximation polynomial.
Appropriate degree of approximation polynomial for method using approximation and statistical evaluation for various sampling frequencies and noise levels. $f_p = 1\,MHz$

	$50\,Ms/s$	$10\,Ms/s$	$5\,Ms/s$
Without noise	2	4	4
$snr = 20\,dB$	2	2	2
$snr = 10\,dB$	2	2	2
$snr = 0\,dB$	2	2	2

2 or 4 . . . polynomial of degree two or four gives a smaller computation error.

Phase shift method with phase interpolation

This method is based on the basic phase shift method (see 3.1.3). The phase spectrum of the signal is computed using the Fourier transform and n points of the phase spectrum is chosen in the neighbourhood of the frequency f_p. Then an approximation polynomial $\widehat{\varphi}(f)$ is fitted through these points. The phase shift between signals $\widehat{\Delta\varphi}$ is computed according to equation (4.6). The time shift is computed according to (3.2).

$$\widehat{\Delta\varphi} = \widehat{\varphi_1}(f_p) - \widehat{\varphi_2}(f_p) \tag{4.6}$$

Comparison of methods with and without phase interpolation

A comparison of the phase shift method with and without phase interpolation was carried out. Methods were compared for signals with signal to noise ratios $20, 10$ and $0\,dB$ sampled at frequencies $50, 10$ and $5\,Msample/s$. The method without phase interpolation was better in all evaluated cases. See Figures 4.6 and 4.7.

Influence of the windowing function

The influence of the windowing function used for sample weighting during Fourier transform computation was evaluated. It was found that differences between results when using rectangular or different windows are negligible and thus can be ignored, see Figure 4.8.

4.2 Methods comparison

Various modifications of methods were compared in the previous section (section 4.1) and for each method the best variant was found. In this section, the above mentioned methods are compared with one another for various combinations of noise level and sampling frequency. The results are summarized in Table 4.3. Because the computation complexity of interpolation followed by the correlation method is much higher than the computation complexity of the other methods, also the next best method has been indicated in the cases when this method was the best.

In Table 4.3 it can be seen that the choice of method for a particular application depends on the signal to noise ratio and is independent of the sampling frequency. The error magnitude depends on the sampling frequency. (Excluding cases of very low sampling frequency — near the Nyquist frequency.)

Error magnitudes for various combinations of noise level and sampling frequency can be found in Table 4.4. Methods used for computations are summarized in Table 4.3. [2] All computation were carried out in double precision IEEE 754 floating point arithmetic.

[2] Methods comparison was accepted for publication at *Applied Electronics Conference 2015.*

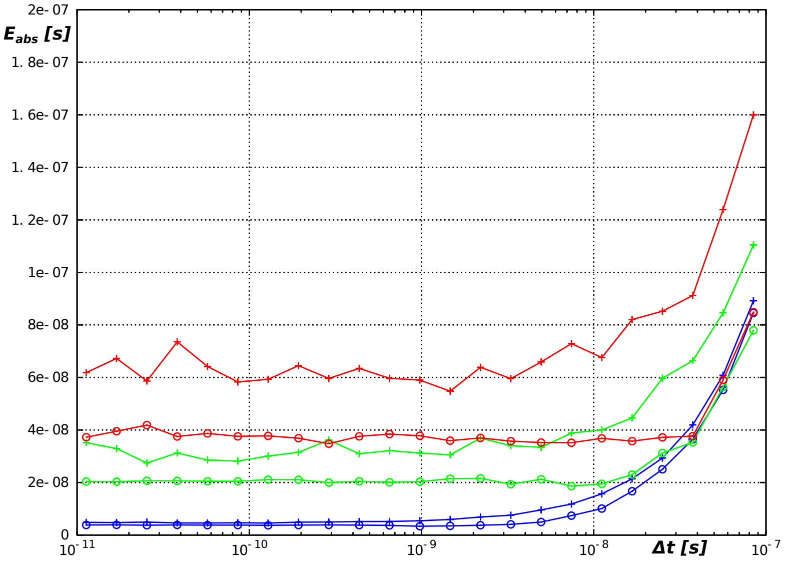

Figure 4.5: Signal approximation using the least squares method — dependence of error on sampling frequency.

Absolute computation error for signal approximation and statistical evaluation method. Degree of approximation polynomial is two, $snr = 0\,dB$.

Sampling frequency $f_s = 50\,Msample/s$ in blue, $f_s = 10\,Msample/s$ in green and $f_s = 5\,Msample/s$ in red.

o . . . mean error, + . . . maximal error.

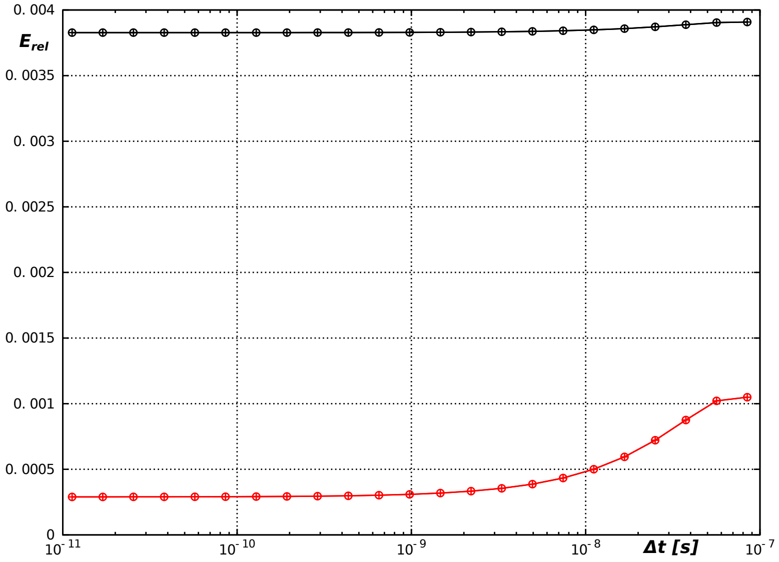

Figure 4.6: Time shift computation from the phase shift I.

Relative computation error for time shift computation from the phase shift with and without phase interpolation. Without noise, $f_s = 10\,Msample/s$.

Without phase interpolation in red, with phase interpolation in black.

o . . . mean error, + . . . maximal error

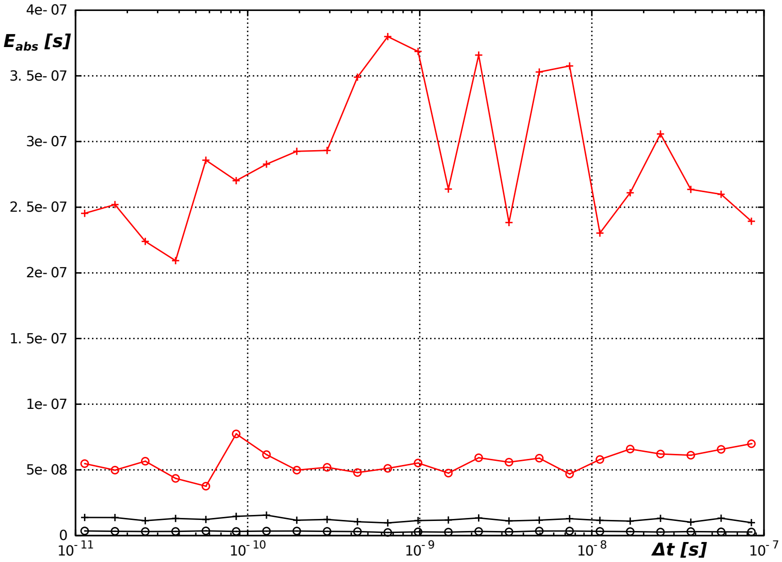

Figure 4.7: Time shift computation from the phase shift II.
Absolute computation error for time shift computation from the phase shift with and without phase interpolation. $snr = 10\,dB$, $f_s = 10\,Msample/s$.
Without phase interpolation in red, with phase interpolation in black.
o . . . mean error, + . . . maximal error

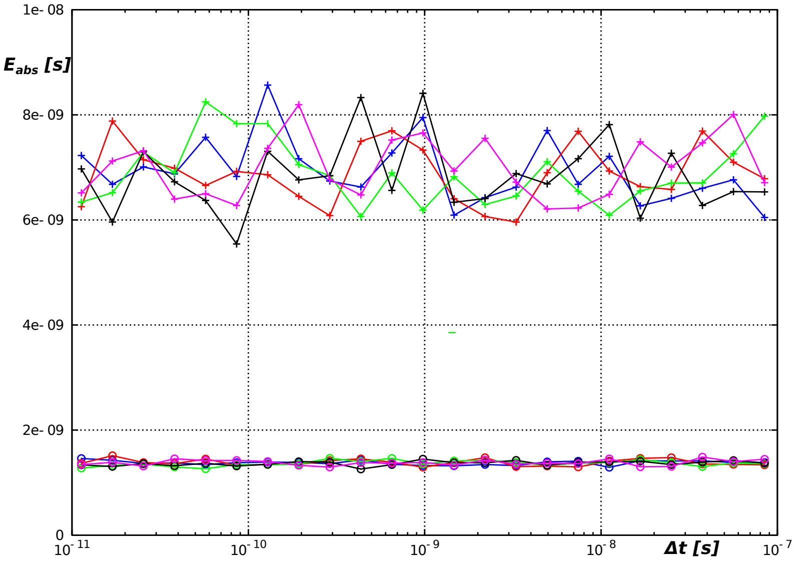

Figure 4.8: Time shift computation from the phase shift - influence of the window.
Absolute computation error for time shift computation from the phase shift. $snr = 20\,dB$, $f_s = 5\,Msample/s$.
Windowing function: rectangular in blue, triangular in red, hamming in green, hann in black and kaiser in magenta.
o . . . mean error, + . . . maximal error.

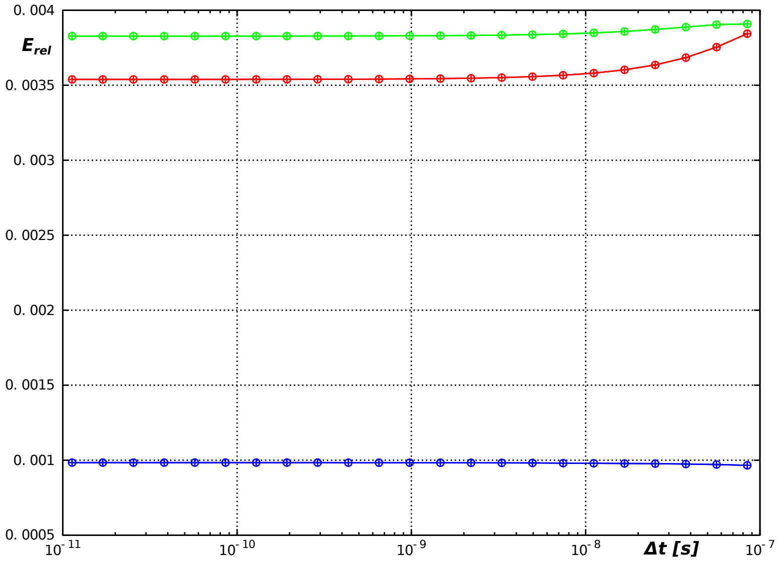

Figure 4.9: Computation error for signal time shift computation using the Fourier transform — without noise.

Relative computation error for signal time shift computation using the Fourier transform. $f_p = 1\,MHz$, without noise.

Sampling frequency $f_s = 50\,Msample/s$ in blue, $f_s = 10\,Msample/s$ in green, $f_s = 5\,Msample/s$ in red.

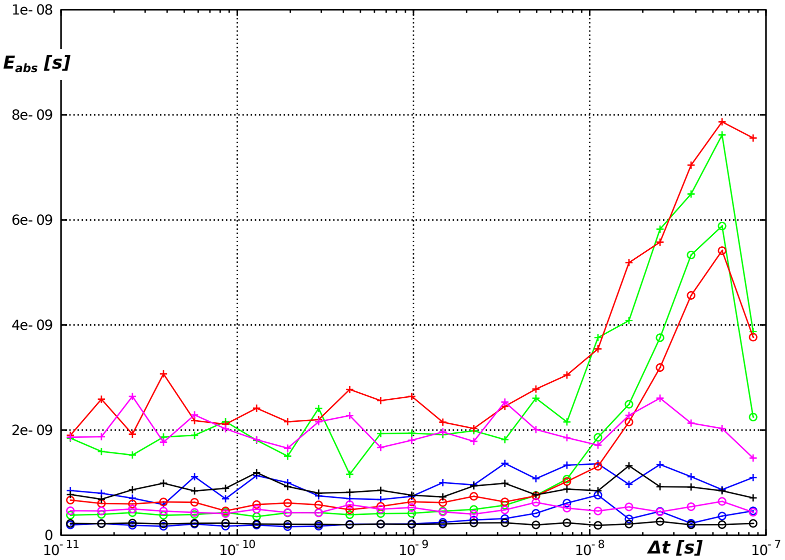

Figure 4.10: Computation error for method of approximation of correlation function I.

Absolute computation error for method of approximation of correlation function. $f_p = 1\,MHz$, $snr = 20\,dB$.

Approximation polynomial of degree 2: Sampling frequency $f_s = 50\,Msample/s$ in blue, $f_s = 10\,Msample/s$ in green, $f_s = 5\,Msample/s$ in red.

Approximation polynomial of degree 4: Sampling frequency $f_s = 50\,Msample/s$ in black, $f_s = 10\,Msample/s$ in magenta.

o . . . mean error, + . . . maximal error.

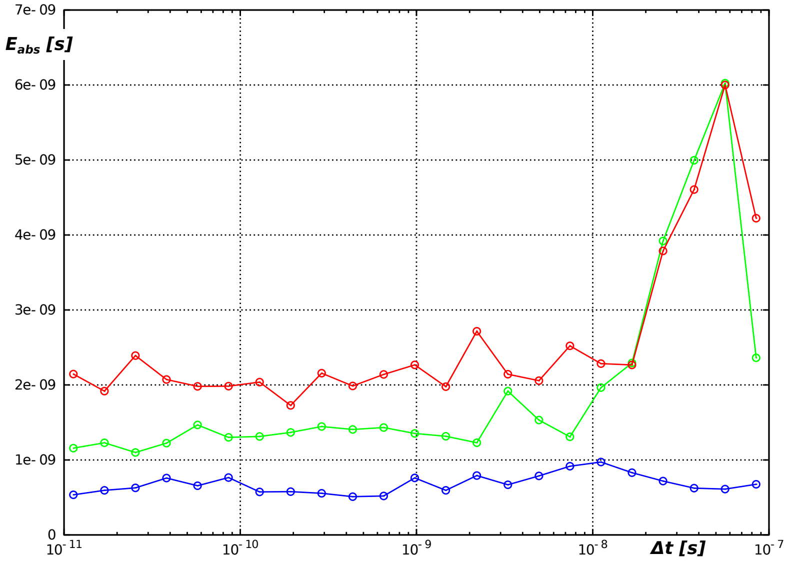

Figure 4.11: Computation error for method of approximation of correlation function II. Absolute computation error for method of approximation of correlation function. $f_p = 1\,MHz$, $snr = 10\,dB$.
Approximation polynomial of degree 2. Sampling frequency $f_s = 50\,Msample/s$ in blue, $f_s = 10\,Msample/s$ in green, $f_s = 5\,Msample/s$ in red.

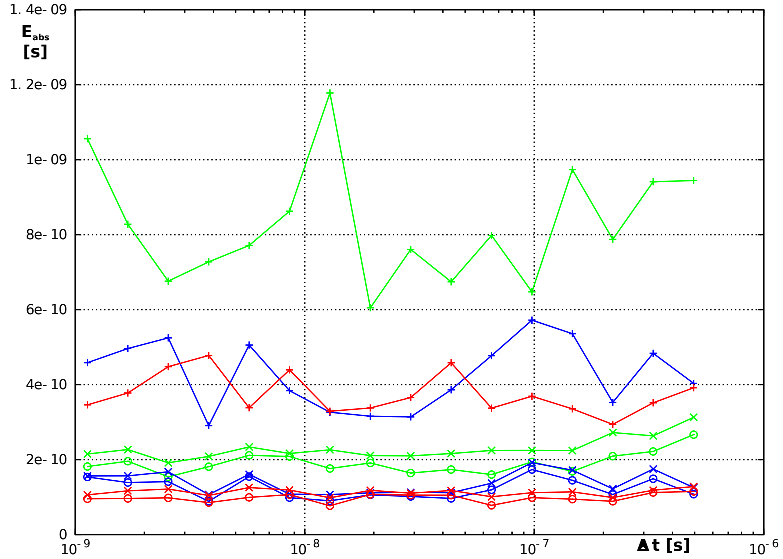

Figure 4.12: Comparison of absolute computation errors for $f_s = 100\,Msample/s$.
$f_s = 100 \cdot f_p$, $SNR = 20\,dB$, $f_p = 1\,MHz$,
Approximation of the correlation function using quadratic polynomial in blue, using polynomial of order of four in red. Phase shift method in green.
Computation run repeatedly:
”x”. . . mean error, ”o”. . . median of error, ”+”. . . maximal error.

Table 4.3: Methods comparison.

This table summarizes the best methods for various combinations of noise level and sampling frequency. $f_p = 1\,MHz$.

	$50\,Ms/s$	$10\,Ms/s$	$5\,Ms/s$
Without noise	FP	FP	FP
$snr = 20\,dB$	IK (KC4)*	KC4	KC2
$snr = 10\,dB$	KC2	KC2	KC2
$snr = 0\,dB$	PS	PS	PS

IK . . . Interpolation followed by correlation.
FP . . . Time shift computation using Fourier transform.
PS . . . Signal approximation using least squares and statistical evaluation.
KC2 or KC4 . . . Approximation of correlation function using polynomial of degree two or four.

* This method has much higher computation complexity than the other methods. The next best method is KC4.

Table 4.4: Magnitude of computation error.

This table summarizes magnitudes of computation errors for methods from the Table 4.3. Computations were carried out in double precision floating point arithmetic. $f_p = 1\,MHz$.

	$50\,Ms/s$	$10\,Ms/s$	$5\,Ms/s$	Figure number
Without noise	0.1 %	0.38 %	0.35 %	4.9
$snr = 20\,dB$	170 ps / 220 ps [1] [2]	450 ps / 400 ps [1]	700 ps	4.10
$snr = 10\,dB$	0.5 ns	1.5 ns	2 ns	4.11
$snr = 0\,dB$	4 ns	20 ns	37 ns	4.5

[1] Error magnitudes for methods KC4 / KC2.

[2] Error magnitude for interpolation followed by correlation (IK) is 200 ps. (Interpolation using zero–valued samples insertion and filtration. Interpolation rate 1:100.) Computation time for this method is much higher than for other methods.

4.3 Summary of the evaluation

The above-compared methods can be used for delay measurement, but the common disadvantage of all methods mentioned is the requirement that the sampling frequency must be much higher than the Nyquist limit. Thus the sampling frequency of the AD converter must be high even for relatively low-frequency signals and a large amount of data is produced and must be processed. This requires high speed communication between the AD converter and the processing unit, a large memory for data storage and a high speed computation unit for data processing. This makes development of such a device difficult and the resulting device expensive. For wider usage of digital delay measurement it is necessary to reduce the design complexity of delay measurement devices. **Thus the development of methods for decreasing the sample amount is required.**

Chapter 5

Nonuniform sampling for sampling frequency reduction

A high frequency of processed signals requires high sampling frequency. When long signals are sampled, a large amount of data is produced increasing processing system complexity.

In the author's article [15], a method for decreasing sampling frequency using nonuniform sampling proposed. This method is described in the following sections.

5.1 Difficulties when using the Fourier transform for signals time shift measurement

Suppose we have the pair of signals described in Chapter 2.1. For time shift computation, signals have to be sampled at least at Nyquist's sampling rate $f_s = 2 \cdot f_p$. Therefore, if f_p is very high, the signal has to be sampled at a very high sampling frequency. If the sampled segment of the waveform is long at the same time, we get a huge amount of sample data to process. It can require a large amount of memory, high speed communication for real-time data transfers etc. To reduce the amount of data, we can consider reducing the sampling frequency or sampling only a part of the waveform. Both approaches have disadvantages: Reducing sampling frequency can cause problems with aliasing. If, on the other hand, only a short part of the waveform is sampled, short-time noise could influence the results[1]. This problem can be solved by using non-uniform sampling.

To use non-uniform sampling, we use the time measurement resolution dt instead of sampling period $\frac{1}{f_s}$, and thus the average sampling rate can be lower then $2 \cdot f_p$. This means that the signal can be sampled along the whole waveform length, while providing a lower amount of data without the danger of aliasing. Using a non-uniform Fourier transform, the signals time shift can be computed.

5.2 Non-uniform sampling

Let us consider a band limited signal (bandwidth B). A well-known Nyquist theorem says that sampling frequency has to be at least $2 \cdot B$ to avoid aliasing. Using an infinite sampling frequency, we can get an infinite bandwidth. In a real system, infinite frequency is unreachable, but we can approximate it. If we sampled a signal of infinite length composed of harmonic waveforms at random sample intervals, we would reach an infinite bandwidth [16]. But this is also impossible for two reasons; firstly no available signal is of infinite length,

[1] It also influences the result in the case of whole waveform sampling, but in this case the influence is reduced by the amount of other samples.

and, secondly we cannot measure time at infinite precision. However, it can be done using an approximation, if we comply with several conditions. If the signal is of finite length and we take enough samples, it will be statistically similar to an infinite length signal. Precision of finite time measurement resolution becomes evident in limited bandwidth and therefore latent sampling frequency.

5.2.1 Latent sampling frequency

If the signal is sampled at random points in time, it is important to specify not only the sample number m, but also the time t_m, when each one of the samples is taken. Time intervals between sampling time points can be specified as multiples of a time quantum dt. As a result, time measurement has limited resolution dt. (Note that time quantum does not represent the time measurement accuracy but the time measurement resolution.) Sampling times t_m can be written as multiplies of this time quantum (5.1).

$$\begin{aligned} t_m &= n \cdot dt \\ &n \ldots \text{random integer} \\ &m \ldots \text{sample number} \end{aligned} \tag{5.1}$$

Having this equation, the non-uniform sampling can be regarded as uniform sampling at frequency dt^{-1} with some samples missing. If the sampling frequency is known, the Nyquist theorem can be applied to figure out the allowed signal bandwidth (5.2).

$$\begin{aligned} B &= \frac{1}{2 \cdot dt} \\ &B \ldots \text{allowed bandwidth} \end{aligned} \tag{5.2}$$

5.2.2 Sampling points

For sampling points distribution, it is convenient to use (5.3). This distribution assures a flat probability mass function when time goes to infinity.

$$\begin{aligned} &t_{n+1} = t_n + r_n \cdot dt \\ &r_n \sim Po(\frac{\overline{f_s}}{dt}); \\ &r_n \ldots \text{random variable} \\ &\overline{f_s} \ldots \text{average sampling frequency} \end{aligned} \tag{5.3}$$

Suppose we have an AD converter with maximal sampling frequency f_{max}. For sampling (uniform or non-uniform), we must ensure $f_s \leq f_{max}$. Because sampling times are random, we can't ensure it for each sample, but we can set the probability of violation to acceptable level p.

From cumulative distribution function (5.4) of the Poisson distribution [17, pp. 32–38], we can find r_{min} [2]. At probability level $1 - p$, the value r_{min} is the minimum of the set $\{r_n\}$; $n = 0, 1, \ldots N-1$. Therefore, the sampling period is greater than $r_{min} \cdot dt$ and the maximal frequency can be enumerated.

$$\begin{aligned} &\frac{p}{N} > \sum_{k=0}^{r} \frac{\lambda^k}{k!} e^{-\lambda} = F(r) \\ &N \ldots \text{number of samples} \\ &\lambda = \frac{\overline{f_s}}{dt} \end{aligned} \tag{5.4}$$

[2] Values of $F(r)$ are tabulated for various values of λ, see [17, pp. 104].

The AD converter sampling frequency has to comply with (5.5).

$$f_m > \frac{1}{r_{min} \cdot dt} \tag{5.5}$$

For values of λ greater than 9 approximation (5.6) can be used [17, pp. 104].

$$Po(\lambda) \approx N(\mu = \lambda, \sigma^2 = \lambda) \tag{5.6}$$

5.3 Non-uniform Fourier Transform

Let us consider a signal $x(t)$. The Fourier transform $\mathbb{X}(k)$ of $x(t)$ definition is in (5.7)[3]. [18] N is the total number of samples, t_n is the sampling time, and $\Delta\omega$ is the circular frequency step.

$$\mathbb{X}(k) = \sum_{n=0}^{N-1} x(t_n) \cdot e^{-j \cdot k \cdot \Delta\omega \cdot t_n} \tag{5.7}$$

Assigning frequency step (5.8) and sampling time $t_n = n$, we get the standard DFT definition (5.9).

$$\Delta\omega = \frac{2 \cdot \pi}{N} \tag{5.8}$$

$$\mathbb{X}(k) = \sum_{n=0}^{N-1} x(n) \cdot e^{-j2\pi k \frac{n}{N}} \tag{5.9}$$

Assigning frequency step (5.10), we get non-uniform DFT definition (5.11).

$$\begin{gathered}\Delta\omega = \frac{2 \cdot \pi}{T} \\ T \ldots \text{signal (window) length}\end{gathered} \tag{5.10}$$

$$\mathbb{X}(k) = \sum_{n=0}^{N-1} x(t_n) \cdot e^{-j2\pi k \frac{t_n}{T}} \tag{5.11}$$

This definition expects continuous time, but we can measure time only in discrete time quanta (clock ticks) lasting dt. Then we can use the number of clock ticks instead of the time (5.12).

$$m = \frac{t}{dt} \tag{5.12}$$

Substituting (5.12) into (5.11), we get (5.13). Informally said, we sample the signal in N points of M possible.

$$\begin{gathered}\mathbb{X}(k) = \sum_{n=0}^{N-1} x(m_n) \cdot e^{-j2\pi k \frac{m_n}{M}} \\ M = \frac{T}{dt}; \quad m_n = \frac{t_n}{dt}\end{gathered} \tag{5.13}$$

$M \ldots$ signal (window) length in clock ticks dt

$m_n \ldots$ sampling time in clock ticks dt

Frequency resolution and maximal nonaliased frequency of standard (DFT) and non-uniform (NDFT) Fourier transform versions can be found in Table 5.1.

[3] In this and all following equations we have omitted normalization considering the number of samples

Table 5.1: Fourier transform modifications comparison.

	DFT	NDFT	Padded DFT[a]
Frequency resolution	$\frac{1}{T}$	$\frac{1}{T}$	$\frac{1}{l \cdot T}$
Maximal nonaliased frequency	$\frac{fs}{2}$	$\frac{1}{2 \cdot dt}$	$\frac{fs}{2}$

[a] Zero padded to l times original length.

The non-uniform Fourier transform can be computed from its definition with computational complexity $O(N^2)$, using the FFT algorithm with computational complexity $O(M \cdot \log(M))$ [4], or faster using an approximation algorithm, for example [19–21].

5.4 Experimental evaluation of the NDFT method

Numerical experiments were carried out to prove the method's desired features. Pairs of time shifted waveforms were generated and noise of a defined level was added. Then the waveforms' time shift was computed and relative and absolute errors were evaluated. This procedure was repeated for many combinations of sampling frequency and signal to noise ratio. For each combination of parameters, the computation was repeated many times and results were statistically evaluated.

When SNR is high (in the context of this article, $SNR \approx 20\,dB$) and sampling frequency is high ($fs \approx 50 \cdot f_p$), the NDFT method is inferior to the correlation or polynomial fit methods. But as noise gets stronger ($SNR \to 0\,dB$) and sampling frequency decreases, Fourier transform methods start to perform better than the others. See Figures 5.1 and 5.1 for details. When sampling frequency approaches the Nyquist limit ($fs = 2 \cdot f_p$) and then decreases, NDFT is preferable to classical (uniform) DFT, see Figures 5.3 and 5.4.

[4] Each clock tick, we have not used for sampling, as zero-valued sample for computation using FFT is used.

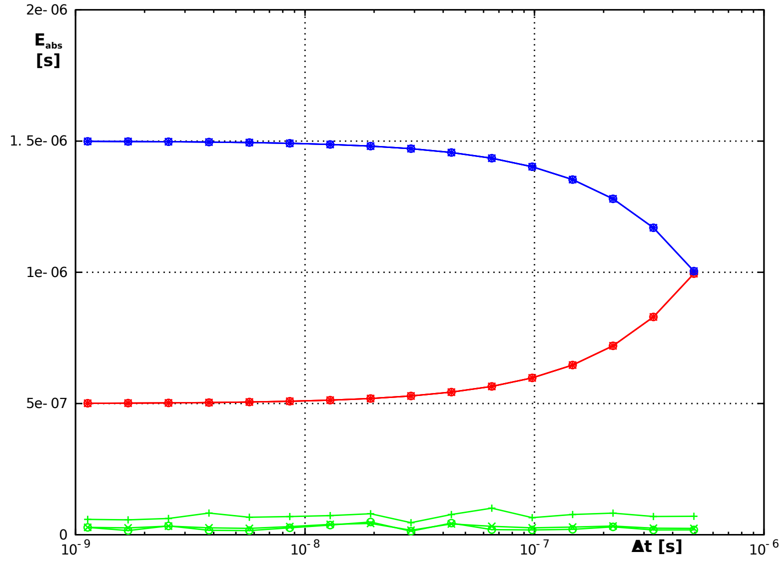

Figure 5.1: Methods comparison I.

$f_s = 2 \cdot f_p$, $dt = (100 \cdot f_p)^{-1}$, $f_p = 1\,MHz$, without noise

Approximation of the cross-correlation function in blue, time shift computation from the phase shift (classical) in red, non-uniform Fourier transform method in green.

Computation run repeatedly:

"x"... mean error, "o"... median of error, "+"... maximal error.

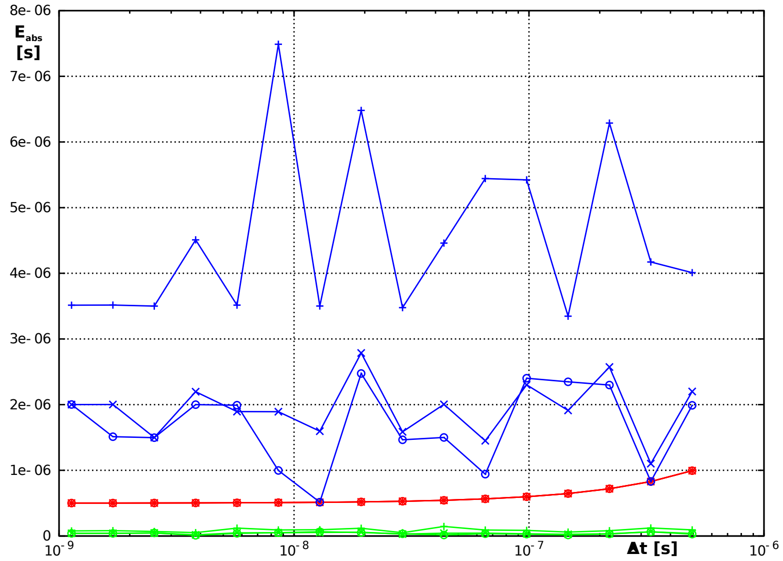

Figure 5.2: Methods comparison II.

$f_s = 2 \cdot f_p$, $dt = (100 \cdot f_p)^{-1}$, $f_p = 1\,MHz$, $SNR = 0\,dB$

Approximation of the cross-correlation function in blue, time shift computation from the phase shift (uniform) in red, non-uniform Fourier transform method in green.

Computation run repeatedly:

"x"... mean error, "o"... median of error, "+"... maximal error.

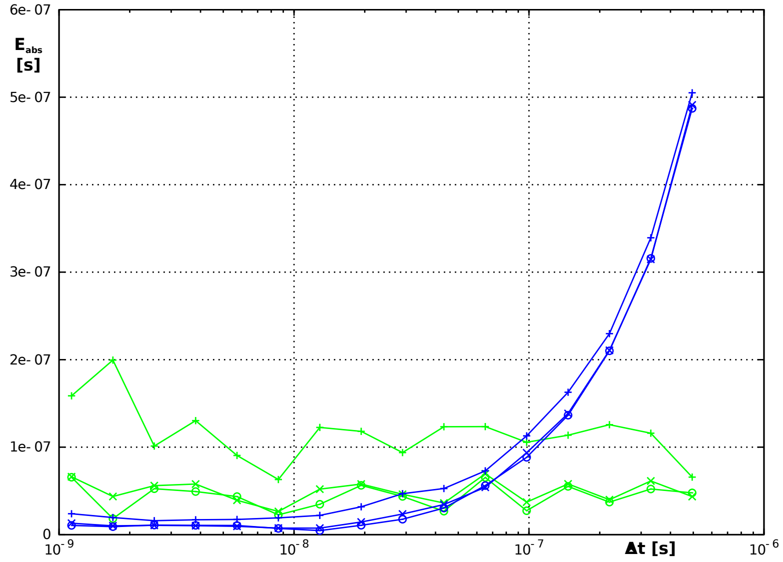

Figure 5.3: NDFT method's absolute error I.

$f_s = 1 \cdot f_p$, $dt = (500 \cdot f_p)^{-1}$, $SNR = 0\,dB$, $f_p = 1\,MHz$, $N \doteq 125$, $M = 50\,000$

Signal approximation using least squares (see 4.1.3) in blue, non-uniform Fourier transform method in green. Classical (uniform) phase shift method failed.

Computation run repeatedly:

"x"... mean error, "o"... median of error, "+"... maximal error.

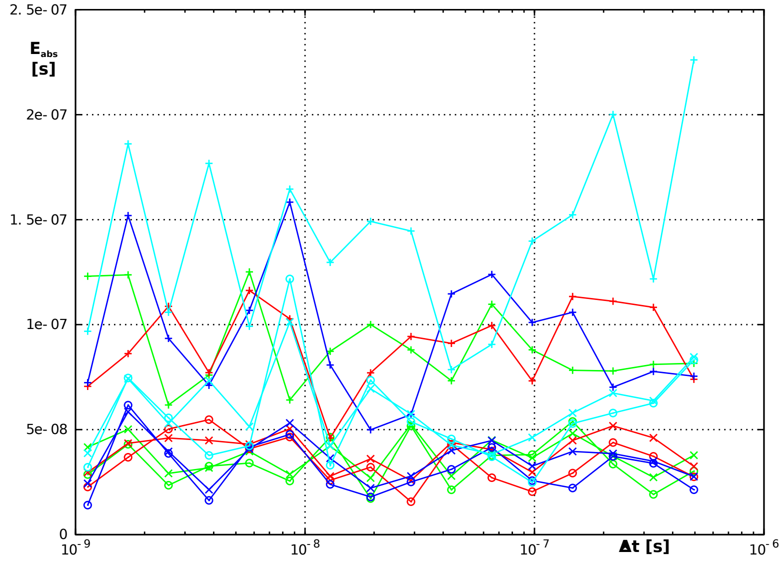

Figure 5.4: NDFT method's absolute error II.

$f_s = 0.5 \cdot f_p$, $dt = (100 \cdot f_p)^{-1}$, $f_p = 1\,MHz$, $N \doteq 62$, $M = 10\,000$

Diagram only for non-uniform Fourier shift method, other methods failed due to lack of samples.

Without noise in green, $SNR = 20\,dB$ in red, $SNR = 10\,dB$ in blue, $SNR = 0\,dB$ in cyan.

Computation run repeatedly:

"x"... arithmetic mean of the error, "o"... median of the error, "+"... maximal error.

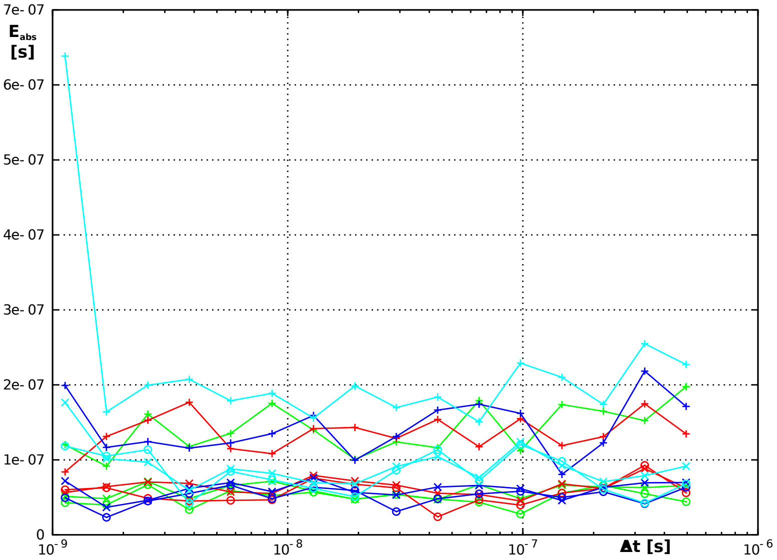

Figure 5.5: NDFT method's absolute error III.

$f_s = 0.2 \cdot f_p$, $dt = (100 \cdot f_p)^{-1}$, $f_p = 1\,MHz$, $N \doteq 25$, $M = 10\,000$

Diagram only for non-uniform Fourier shift method, other methods failed due to lack of samples.

Without noise in green, $SNR = 20\,dB$ in red, $SNR = 10\,dB$ in blue, $SNR = 0\,dB$ in cyan.

Computation run repeatedly:

"x"... arithmetic mean of the error, "o"... median of the error, "+"... maximal error.

Chapter 6

Goals of the Ph.D. thesis and further work

6.1 Goals of the Ph.D. thesis

The goals of the Ph.D. thesis are the following:

1. To study and compare the methods for time delay measurement. — *Done.*
2. To develop new procedures for decreasing the sampling frequency and amount of sampled data necessary for time delay measurement. — *In progress.*
3. To verify properties of the proposed methods. — *To be done.*

The hypothesis emerging from the current research is: *Using non-uniform sampling and the non-uniform Fourier transform, it is possible to decrease the number of samples and the sampling frequency needed for measurement of the time shift between analogue signals compared to methods using periodic sampling.*

6.2 Objectives to be met

In Chapter 6.3 the main problems to be solved are described. In this section, those problems are summarised into several objectives planned for further studies in order to accomplish the goals of the Ph.D. thesis:

1. Modify the sampling scheme in order to ensure the minimal sampling period with 100 % probability.
2. Reduce noise and secondary aliasing in the spectrum.
3. Examine the relation between the number of samples and accuracy.
4. Adapt the method for the use of general purpose AD converters.
5. Implement and test the method in a near-to-real appliance.

6.3 Further work

The basic principle of the time delay measurement method using nonuniform sampling was introduced. But to make the method usable, more work must be done.

First of all, the properties of the non-uniform sampling must be better studied. In section 5.2, the minimal sampling frequency of the AD converter ensuring at certain probability level that no sampling instant would be missed is derived. It would be useful to modify the sampling scheme (e.g. modifying the probability density function or using precomputed pseudo-random number sequences in (5.3)) to ensure this at the 100% probability level.

It is necessary to improve an algorithm for detection of the frequency of the signal. When the signal to noise ratio is low, peaks in the spectra are hardly distinguishable. And *fuzzy* or *secondary aliasing* in the results of the Fourier transform can also appear(see [22, chapter 9]).

For wider use of the method, it is necessary to specify conditions for method usability. When the signal gets shorter, the number of samples taken decreases. Thus the probabilistic properties of the sampling process get worse and results distortion appears. When the average sampling frequency decreases, a similar effect appears.

The construction of a dedicated AD converter for non-uniform sampling is improbable; thus general purpose AD converters would be used in real appliances. According to [22, pp. 58], the bandwidth of analogue inputs of common AD converters is typically 4 to 8 times higher than half of the maximal (periodic) sampling frequency of the AD converter. This means that using non-uniform sampling, a much wider bandwidth can be achieved compared to periodic sampling using the same AD converter. Thus, the features of common AD converters should be carefully studied and the time delay computation method should be adapted for non-uniform sampling carried out using common AD converters originally intended for periodic sampling.

Many microcontrollers are equipped with AD converters working at a sampling frequency of several hundreds of kilosamples. Using the appropriate method, it may be (or may not be) possible to measure the delay between signals at frequencies of several megahertz using these cheap devices.

For practical verification of the intended properties of the method, some time delay measurement device should be implemented using this method and tested in a real or near-to-real appliance.

Chapter 7

Conclusion

In the first two chapters, there is a brief introduction to short time delay measurement and the problem specification. The third chapter describes the state of the art of delay measurement. The selected methods are experimentally evaluated and compared in Chapter 4. All methods require a high sampling frequency (in the sense of multiples of the signal bandwidth). Because of this, high speed AD converters must be used, which produce a large amount of the data to be stored and processed, thus increasing the complexity of the device.

In the fifth chapter a new method for decreasing the number of samples necessary for time delay measurement using nonuniform sampling and nonuniform Fourier transform is described. An experimental evaluation of the method was carried out. It was found that results obtained using this method are better than results obtained using the other methods. These results are promising, but for the full usability of the method, more work must be done.

Chapter 6 describes the further work. Also the goals of the Ph.D. thesis are described in that chapter.

Acknowledgement

This work has been supported by UWB grant SGS–2013–029 Advanced Computer and Information Systems and by the European Regional Development Fund (ERDF) — project NTIS (New Technologies for the Information Society), European Centre of Excellence, CZ.1.05/ 1.1.00/ 0.2.0090.

Bibliography

[1] Xiangwei Zhu, Guangfu Sun, Shaowei Yong, and Zhaowen Zhuang. A high-precision time interval measurement method using a phase-estimation algorithm. *Instrumentation and Measurement, IEEE Transactions on*, 57(11):2670–2676, Nov 2008. ISSN 0018-9456. doi: 10.1109/TIM.2008.925025.

[2] P. Pánek. Time-interval measurement based on SAW filter excitation. *Instrumentation and Measurement, IEEE Transactions on*, 57(11):2582–2588, Nov 2008. ISSN 0018-9456. doi: 10.1109/TIM.2008.925014.

[3] Ming-Chien Tsai and Ching-Hwa Cheng. A full-synthesizable high-precision built-in delay time measurement circuit. In *Design Automation Conference, 2009. ASP-DAC 2009. Asia and South Pacific*, pages 123–124, Jan 2009. doi: 10.1109/ASPDAC.2009.4796463.

[4] Xu Wei, Chen Qian, and Gu Guo-hua. Research on laser ranging system based on time delay estimation. In *Photonics and Optoelectronics (SOPO), 2011 Symposium on*, pages 1–3, May 2011. doi: 10.1109/SOPO.2011.5780683.

[5] Xiaoming Lai and H. Torp. Interpolation methods for time-delay estimation using cross-correlation method for blood velocity measurement. *Ultrasonics, Ferroelectrics, and Frequency Control, IEEE Transactions on*, 46(2):277–290, March 1999. ISSN 0885-3010. doi: 10.1109/58.753016.

[6] Joseph C. Hassab and R. Boucher. Optimum estimation of time delay by a generalized correlator. *Acoustics, Speech and Signal Processing, IEEE Transactions on*, 27(4):373–380, Aug 1979. ISSN 0096-3518. doi: 10.1109/TASSP.1979.1163269.

[7] Wen Ma and Jianguo Huang. Accurate time delay estimation based on SINC filtering. In *Signal Processing, 2002 6th International Conference on*, volume 2, pages 1621–1624 vol.2, Aug 2002. doi: 10.1109/ICOSP.2002.1180109.

[8] Y.T. Chan, J. Riley, and J. Plant. A parameter estimation approach to time-delay estimation and signal detection. *Acoustics, Speech and Signal Processing, IEEE Transactions on*, 28(1):8–16, Feb 1980. ISSN 0096-3518. doi: 10.1109/TASSP.1980.1163361.

[9] Fan Di, Cao Maoyong, and Sun Nongliang. Time delay estimation based on wiener filter in ultrasonic detection of sediments in drilling hole. In *Computer Science and Engineering, 2009. WCSE '09. Second International Workshop on*, volume 2, pages 582–585, Oct 2009. doi: 10.1109/WCSE.2009.880.

[10] C. Guetbi, D. Kouame, A. Ouahabi, and J.P. Chemla. Methods based on wavelets for time delay estimation of ultrasound signals. In *Electronics, Circuits and Systems, 1998 IEEE International Conference on*, volume 3, pages 113–116 vol.3, 1998. doi: 10.1109/ICECS.1998.813947.

[11] Shiyuan Zhou, Yao Xu, Hongbo Wang, and Chunguang Xu. Time delay estimation via third-order cumulant. In *Nondestructive Evaluation/Testing: New Technology Application (FENDT), 2013 Far East Forum on*, pages 77–81, June 2013. doi: 10.1109/FENDT.2013.6635533.

[12] Jiří Jan. *Číslicová filtrace, analýza a restaurace signálů.* Vysoké učení technické v Brně, VUTIUM, Brno, Czech Republic, 2002. ISBN 80-214-1558-4.

[13] Dana Černá. *Interpolace.* [online], April 2010. [cit. 2013-06-20] Available: kmd.fp.tul.cz/lide/cerna/MA3_FS/interpolace.pdf.

[14] David Kahaner, Cleve Moler, and Stephen Nash. *Numerical Methods and Software.* Prentice Hall, Phoenix, New Jersey, 1988. ISBN 0-13-627258-4.

[15] Karel Dudáček jr., Karel Dudáček, and Vlastimil Vavřička. Short delay measurement using non-uniform fourier transform. In *14th Biennial Baltic Electronics Conference, Proceedings of the*, pages 165–168, Tallinn, Estonia, October 2014. Tallinn University of Technology. ISBN 978-9949-23-672-5.

[16] Jian-Jiun Ding. *Non-uniform Sampling.* [online]. Available: http://djj.ee.ntu.edu.tw/Nonuniform_Sampling.docx [2014, Apr. 3].

[17] Zdeněk Kobeda and Jiří Reif. *Úvod do pravděpodobnosti a spolehlivosti.* University of West Bohemia, Pilsen, Czech Rep., 2nd edition, 2004. ISBN 80-7043-333-7.

[18] Jae-Jeong Hwang, Sang-Gyu Cho, Joon Moon, and Jae-Wan Lee. Nonuniform DFT based on nonequispaced sampling. *WSEAS Transactions on Information Science and Applications*, 2(9):1403–1408, 2005.

[19] Alok Dutt. *Fast Fourier Transform for Nonequispaced Data.* PhD thesis, Yale University, Connecticut, 1993.

[20] Alok Dutt and V. Rokhlin. *Fast Fourier Transform for Nonequispaced Data II.* Technical report, Yale University, Connecticut, 1993. Research Report 980.

[21] Daniel Potts. *NFFT.* [Online], January 2014. Available: http://www-user.tu-chemnitz.de/~potts/nfft/ [2014, Apr. 3].

[22] Ivars Bilinskis. *Digital Alias-free Signal Processing.* John Wiley & Sons, Ltd, Chichester, England, 2007. ISBN 978-0-470-02738-7.