Digital Filtering for Robust 50/60 Hz Zero Crossing Detectors

Olli Vainio[1] and Seppo J. Ovaska[2]

[1]Tampere University of Technology, Signal Processing Laboratory,
P. O. Box 553, FIN-33101 Tampere, Finland
[2]Lappeenranta University of Technology, Electronics Laboratory,
P. O. Box 20, FIN-53851 Lappeenranta, Finland

ABSTRACT

An improved digital filtering method for line frequency zero crossing detectors is proposed. The multistage filter efficiently attenuates harmonics, wide-band noise, commutation notches, and other impulsive disturbances without causing any phase shift on the primary sinusoidal waveform. Our novel signal processing system is a cascade of a median filter and an adaptive sinusoid predictor, followed by up-sampling and interpolation. The three-point median filter effectively removes impulses, and the predictor provides wide-band noise attenuation while compensating for delays in the other processing steps. The predictor adapts to possible line frequency variations within the specified range by changing the set of coefficients, based on an estimate of the instantaneous line frequency. The adaptive approach allows the use of highly selective IIR bandpass predictors.

1. INTRODUCTION

Accurate detection of true zero crossings is important in many fields of industrial and consumer electronics, particularly in triac and thyristor power converters, where satisfactory line synchronization requires reliable zero crossing information. In practice, the 50/60 Hz sinusoid is seldom distortionless, and heavy noise that typically consists of deep and wide notches, can easily lead to harmful disturbances in firing synchronization. Signal processing is therefore needed for reliable zero crossing detection [1].

A concise comparative summary of existing solutions to zero crossing noise reduction was recently published by Weidenbrüg et al. [2]. Although they classified totally six major approaches to the present problem, none of the existing implementations can be considered as the ultimate general-purpose zero crossing detector. This is partly because the older methods have severely limited performance due to the limitations of the underlying component technology. On the other hand, the newer microcomputer-based approaches, like the one reported by Kumar et al. [3], are usually strongly tailored to the particular operating environment.

A novel digital filter-based zero crossing detector was proposed by the authors in [4]. It is based on a multistage filter which suppresses notch-type disturbances efficiently, and due to its bandpass nature, it also attenuates wide-band noise. Computational efficiency is a natural consequence of the straightforward structure, and the algorithm can easily be implemented in a low-cost microcontroller that is a standard component in modern motor drives. A drawback in this approach is limited filter selectivity, dictated by the requirement to support ±2 % frequency variations. The main limiting factor is phase delay variation as the line frequency deviates from the nominal value.

In this paper, an improved multistage filter is proposed, adapting to the instantaneous line frequency. The adaptive approach allows us to use more selective bandpass predictors than the fixed algorithm. The dc component and the first odd harmonic are also fully suppressed by the improved predictors. Furthermore, the interpolating filter is specifically designed for the sinusoidal waveform, which gives better accuracy than linear interpolation.

This paper is organized as follows. In Section 2, we describe the specifications and structure of the adaptive multistage filter for line frequency signals, and show how the building blocks operate. 50 Hz nominal line frequency is assumed throughout this paper, but the method is applicable for 60 Hz as well, if the sampling rate or the other design parameters are changed correspondingly. Section 3 shows the results of successfully processing a noisy test signal. Section 4 outlines the implementation in a microcontroller environment and Section 5 concludes the paper.

0-7803-2615-6/95/$4.00 © 1995 IEEE

2. THE ADAPTIVE MULTISTAGE FILTER

Our objective is to remove disturbances from the sinusoidal signal without causing any phase shift on the sinusoid. Thus zero crossings can be detected in real time from sign changes of the filtered output. A suitable filtering approach is to cascade median filtering with predictive linear filtering. Cascaded nonlinear and linear filters have been previously used, for instance, in speech processing [8].

The objective is $100\mu s$ resolution, but the input sampling rate is lower than 10 kHz, for two reasons. First, we want to keep the computational load low. Secondly, the worst case duration of commutation disturbances in thyristor converters is about 600 microseconds [3]. By choosing the sample rate of 1.67 kHz ($= 1/600\mu s$), we typically get only one severely corrupted sample from each commutation notch, among the 'better' samples (which may also be noisy). The final desired resolution is obtained by interpolation, after first filtering for sufficient noise attenuation.

The proposed multistage digital filter for zero crossing detectors is shown in Fig. 1. The first block is a three-point median filter which removes the disturbing impulses. The median filter is a nonlinear filter which operates by sorting the samples inside the moving filter window by magnitude, and choosing the middle value, i.e., the median, as the output. The median filter is very suitable for this application, as it completely removes isolated impulses, regardless of their magnitude [6]. However, the median filter has some drawbacks: it causes a one sample delay, and it does not fully restore a sinusoidal signal after removing impulses.

Both of these problems are compensated by the adaptive sinusoid predictor. The filter predicts *two* steps ahead: one to compensate for the delay of the median filter, and the other to allow interpolation rather than extrapolation in the final processing stage. The sinusoid predictor consists of an FIR part and a single feedback loop. Because of the feedback, the predictor is an IIR system. Adaptation affects only the coefficients of the FIR block while the feedback remains fixed.

The time resolution of the proposed zero crossing detector depends on the output sampling rate, which is also an application specific parameter. We have used 10 kHz output sampling rate. Interpolation by the factor of six is therefore required, and the waveform is restored in the FIR interpolator with 11 coefficients. The output of the multistage filter is thus the line frequency sinusoid, sampled at 10 kHz, to be used for zero crossing detection.

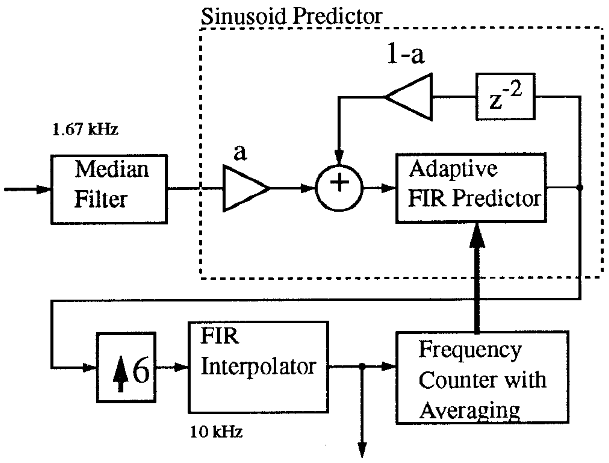

Fig. 1. Block diagram of the multistage digital filter for zero crossing detectors.

The instantaneous frequency is estimated by counting the cycles of the 10 kHz clock between zero crossings, detected from the filtered sinusoid. For added reliability, a running average of such estimates is computed. As the line frequency tolerance is specified to ±2 %, a full period of the sinusoid takes 196 ... 204 clock cycles. In order to achieve accurate prediction, the coefficients are selected based on the cycle time. As there are nine possible counter values, a table with nine sets of coefficients is stored for the predictor.

In the following, analytical methods are given for obtaining the coefficients of the predictor and the FIR interpolator.

2.1. Predictor Design

Let us assume that the input signal, in the absence of noise, can be modeled by a sinusoid:

$$x(n) = \sin(\omega_0 n + \phi), \tag{1}$$

where ω_0 is the nominal frequency and ϕ is an arbitrary phase shift. We want to estimate $x(n-p)$ by $\hat{x}(n-p)$ which is obtained by filtering the input samples $x(n-k), k = 1, \ldots, N$, with a predictive FIR filter of length N, having the coefficients $h(k), k = 1, \ldots, N$. Thus the parameter p determines the length of the prediction step, for example, $p = -1$ for two steps-ahead prediction.

In our case, there are three preconditions for the task of determining the coefficients $h(k)$. Exact prediction of sinusoids of the nominal frequency is required, i.e.,

$$\sin[\omega_0(n-p)+\phi] = \sum_{k=1}^{N} h(k)\sin[\omega_0(n-k)+\phi]. \tag{2}$$

Second, the dc component must be removed. Third, the first odd harmonic of $3\omega_0$ must be fully suppressed, as it is usually the dominant cause of harmonic distortion. The remaining degrees of freedom are used for maximizing the white noise attenuation, as this approach leads to favorable bandpass characteristics. Because the noise components in each sample are assumed to be independent, the noise power gain is given by $\sum_{k=1}^{N}[h(k)]^2$ [5]. The optimization can be carried out analytically using the method of Lagrange multipliers, which has been previously applied for the design of polynomial predictors [5],[6].

The following constraints are applied in optimization:

$$g_0 = \sum_{k=1}^{N} h(k)[\cos\omega_0 \cos k\omega_0 + \sin\omega_0 \sin k\omega_0] - \cos[(1-p)\omega_0] = 0 \qquad (3a)$$

$$g_1 = \sum_{k=1}^{N} h(k)[\sin\omega_0 \cos k\omega_0 - \cos\omega_0 \sin k\omega_0] - \sin[(1-p)\omega_0] = 0 \qquad (3b)$$

$$g_2 = \sum_{k=1}^{N} h(k) = 0 \qquad (3c)$$

$$g_3 = \sum_{k=1}^{N} h(k)[\cos 3\omega_0 \cos 3k\omega_0 + \sin 3\omega_0 \sin 3k\omega_0] = 0 \qquad (3d)$$

$$g_4 = \sum_{k=1}^{N} h(k)[\sin 3\omega_0 \cos 3k\omega_0 - \cos 3\omega_0 \sin 3k\omega_0] = 0. \qquad (3e)$$

Equations (3a) and (3b) specify the desired property of predicting $1-p$ steps ahead on the nominal frequency ω_0. Zero dc gain is the result of constraint g_2 in (3c). Suppression of the odd harmonic of $3\omega_0$ is specified in (3d) and (3e).

The Lagrange function is

$$L(h(1),\ldots,h(N),\lambda_0,\lambda_1,\lambda_2,\lambda_3,\lambda_4) = \sum_{k=1}^{N}[h(k)]^2 + \lambda_0 g_0 + \lambda_1 g_1 + \lambda_2 g_2 + \lambda_3 g_3 + \lambda_4 g_4. \quad (4)$$

The optimal coefficients $h(k), k = 1,\ldots,N$, are found by setting the partial derivatives of $L(h(1),\ldots,h(N),\lambda_0,\ldots,\lambda_4)$ with respect to all of the arguments equal to zero. Details of the method are found in [4],[5], and [6]. The procedure is easy to program, for example, using the Mathematica software [7].

In order to cover the frequency band of 50 Hz ±2%, nine sets of coefficients are used. As the sampling rate is 1.67 kHz, the nominal angular frquencies are in the range $0.06\pi \pm 2\%$. Choosing $N = 22$ and setting $p = -1$, the resulting sets of coefficients are listed in the Appendix. Because of the narrow frequency band, the coefficient sets and the frequency responses of these predictors are close to each other. Figure 2 shows the amplitude response of the FIR predictor with $\omega_0 = 0.06\pi$.

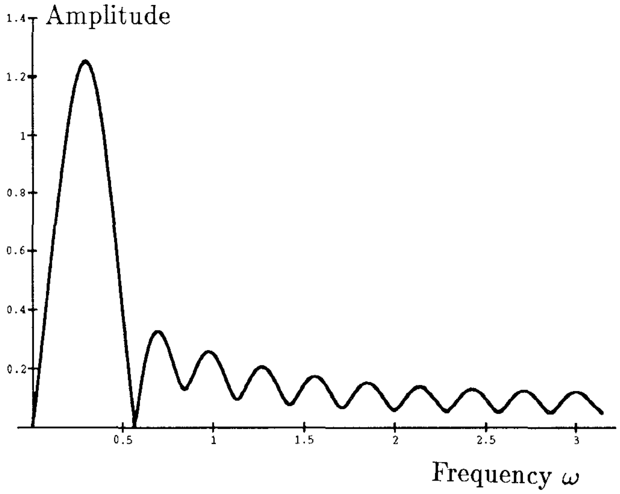

Fig. 2. Amplitude response of the FIR predictor, for $\omega = 0\ldots\pi$. $\omega_0 = 0.06\pi$, $N = 22$, $p = -1$.

The value of coefficient a is chosen with two considerations in mind. A small value of a would give high stopband attenuation and a very narrow passband at ω_0. But a also affects the phase response, and the error in phase delay is not allowed to exceed the duration of the output sampling period, as the frequency deviates from ω_0. A smaller value of a causes a steeper slope of the phase delay curve at ω_0. We choose $a = 0.25$ which limits the phase delay error to ±57 μs with the above mentioned parameters, allowing the output sampling rate to be interpolated up to 17.5 kHz. Increasing the sampling rate further would not improve the resolution, as the phase errors in the predictors become the dominating limitation. If no feedback is used ($a = 1$), the timing resolution is ±14.2 μs.

The feedback has the effect of changing the amplitude response of Fig. 2 to that shown in Fig. 3. The phase delay of the predictor is shown in Fig. 4 for $\omega = 0.05\pi\ldots0.07\pi$. On the nominal frequency of 0.06π, the amplitude response is equal to unity, and the phase delay is -2, corresponding to a *forward* phase shift of two sampling periods.

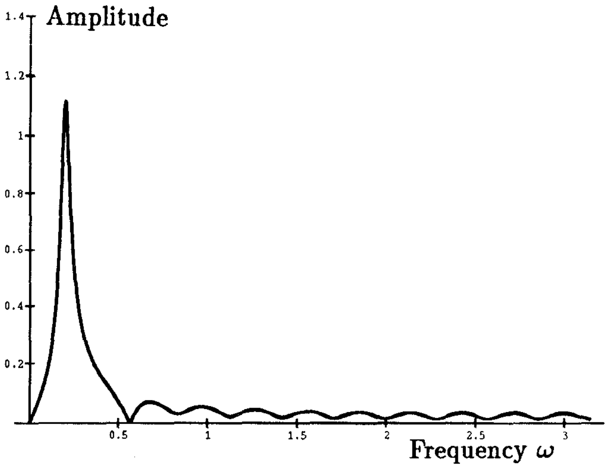

Fig. 3. Amplitude response of the predictor with feedback, $a = 0.25$, for $\omega = 0 \ldots \pi$.

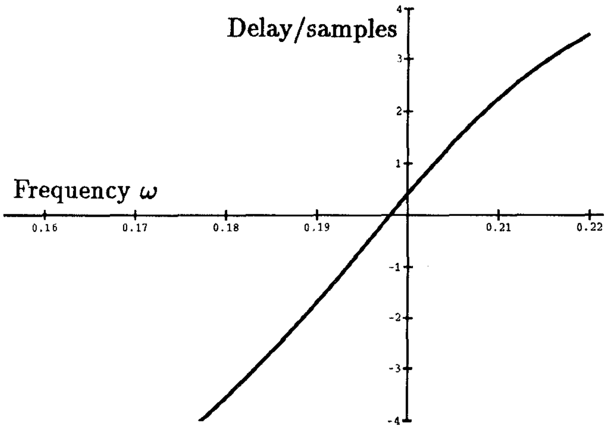

Fig. 4. Phase delay of the predictor, for $\omega = 0.05\pi \ldots 0.07\pi$.

2.2. The Interpolator

A linear interpolator was used previously [4] which is feasible in zero crossing detectors as the sinewave is approximately linear near zero crossings. A better solution is, however, to use an interpolating filter which is specifically tailored for a sinewave of the nominal line frequency.

For interpolation by six up to 10 kHz sample rate, we use an FIR interpolator of length 11, having the coefficients $h(0), \ldots, h(10)$. We set $h(5) = 1$ and solve the other coefficients from the following system of 10 equations:

$$h(i)\sin(n+6)0.01\pi + h(i+6)\sin n0.01\pi = \sin(n+i+1)0.01\pi \tag{5a}$$

$$h(i)\sin(n+12)0.01\pi + h(i+6)\sin(n+6)0.01\pi = \sin(n+i+7)0.01\pi, \tag{5b}$$

$i = 0, 1, \ldots, 4,$
with arbitrary n, giving:
$h(0) = h(10) = 0.167630$, $h(1) = h(9) = 0.335095$, $h(2) = h(8) = 0.502229$, $h(3) = h(7) = 0.668867$, $h(4) = h(6) = 0.834846$.

3. EVALUATION

By construction, the system completely removes isolated impulses of length less than $600\mu s$, such as those observed by Kumar et al. [3]. For evaluation of more practical interest, we have used an approximative replica of the test signal used recently by Weidenbrüg et al. in [2]. The input signal is shown in Fig. 5 and the output of the proposed multistage filter is shown in Fig. 6. The output is seen to be an almost pure sinusoid. This is despite the fact that the input signal is severely corrupted by strong impulsive disturbances, some of which are very close to true zero crossings.

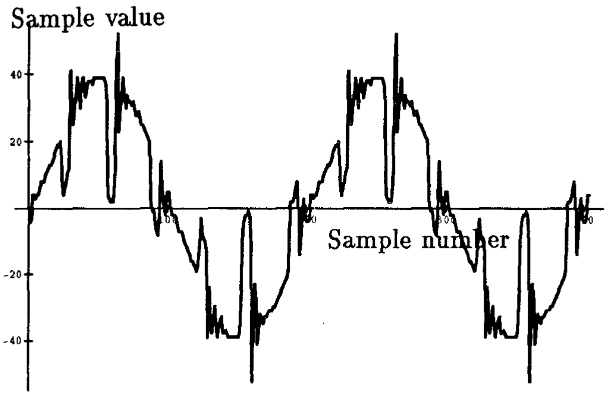

Fig. 5. Test signal input.

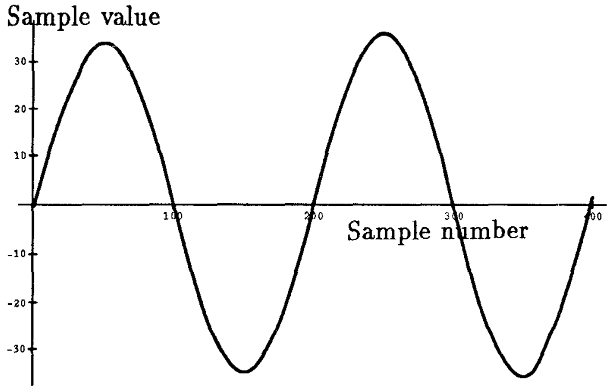

Fig. 6. Steady-state response to the test input.

In fact, the filtering algorithm is powerful enough to allow *single-bit* input quantization. Thus a simple comparator can be used instead of an A/D converter, which is a hardware advantage and reduces signal latency. One-bit quantization actually converts the sinusoid to a square wave with strong harmonic contents. The harmonics are attenuated in the filtering process below the -40 dB level. This is seen in Fig. 7 which shows the output spectrum, when the input is a pure sinusoid of the nominal frequency and one-bit quantization is applied. Total distortion is -35.7 dB.

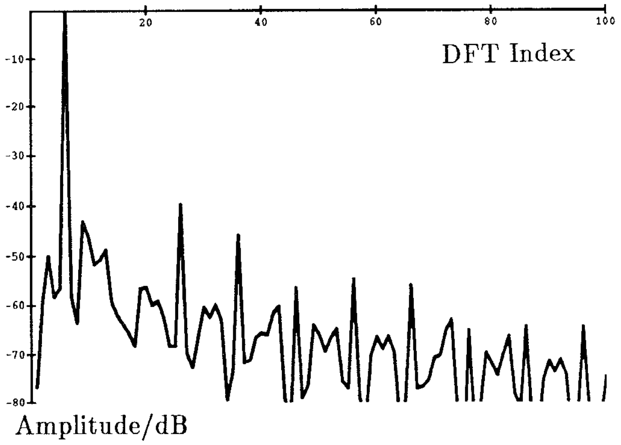

Fig. 7. Spectrum of the output, when the input is a one-bit quantized sinusoid, $\omega = 0.06\pi$.

A drawback of single-bit quantization is that continuously asymmetrically displaced zero crossings in the input signal will cause displaced output, because the median filter only removes isolated impulses. Multi-bit quantization works better in such a situation because phase information of the fundamental sinusoid is available.

4. IMPLEMENTATION CONSIDERATIONS

Microprocessors are routinely used in modern power converters [3], although the trend is towards higher performance signal processors. Therefore, we discuss the suitability of our filtering method for a microcontroller environment. As explained in the above sections, the entire method is computationally very efficient, and hence it can usually be executed in parallel with other system functions, such as finite state machines, regulator and servo controllers, as well as supervision and sampling tasks. If the computational burden of these tasks is heavy, then a 16-bit processor is recommended, because the required 16-bit arithmetic in an 8-bit environment is typically too time consuming. On the other hand, when the cost pressure of the overall product is high, the one-bit quantization option leads us to an attractive low-cost solution, because no A/D converter channel is needed for the zero-crossing filtering. In intelligent light dimmers [1], where the other tasks are computationally inconsiderable, a simple 8-bit microcontroller is undoubtedly an adequate implementation environment. A digital ASIC is another possible implementation platform for the zero-crossing detector with a one-bit quantizer.

Adaptation of the predictor coefficients is based on robust frequency estimation and dynamic selection of the applied coefficient set. Selection of the current predictor coefficients is done in software simply by changing a column index, if the coefficient sets are stored in a two-dimensional data structure. This is a very fast operation, and only an addressing offset must be changed on the machine language level. Consider the indexed addressing mode and a multiplication instruction:

MUL R1, (R2 + R3)

that can be interpreted as $\mathrm{R1} \leftarrow \mathrm{R1}\times\mathrm{mem}[\mathrm{R2}+\mathrm{R3}]$. Here the register R2 contains the coefficient set offset and R3 the running coefficient index.

5. CONCLUSIONS

A powerful adaptive digital filtering algorithm was proposed for noise reduction in zero crossing detectors. The proposed method is robust against strong impulsive disturbances, and tolerates also other kinds of noise. Analytical design procedures were described for the filters. Therefore the proposed design can easily be adapted for different system specifications and noise characteristics.

REFERENCES

[1] H. F. Mikkelsen, "Using digital signal processing techniques in light controllers," *IEEE Trans. Consumer Electronics,* vol. 39, pp. 122-130, May 1993.

[2] R. Weidenbrüg, F. P. Dawson, and R. Bonert, "New synchronization method for thyristor power converters to weak AC-systems," *IEEE Trans. Ind. Electron.,* vol. 40, pp. 505-511, Oct. 1993.

[3] P. P. Kumar, R. Parimelalagan, and B. Ramaswami, "A microprocessor-based DC drive control scheme using predictive synchronization," *IEEE Trans. Ind. Electron.,* vol. 40, pp. 445-452, Aug. 1993.

[4] O. Vainio and S. J. Ovaska, "Noise reduction in zero crossing detection by predictive digital filtering," *IEEE Trans. Ind. Electron.,* vol. 42, no. 1, Feb. 1995.

[5] P. Heinonen and Y. Neuvo, "FIR-median hybrid filters with predictive FIR substructures," *IEEE Trans. Acoust., Speech, Signal Processing,* vol. 36, pp. 892-899, June 1988.

[6] T. G. Campbell, "Design and implementation of image filters," Doctoral Dissertation, Tampere University of Technology, Tampere, Finland, 1992.

[7] S. Wolfram, *Mathematica: A System for Doing Mathematics by Computer.* Reading, MA: Addison-Wesley, 1988.

[8] L. R. Rabiner, M. R. Sambur, and C. E. Schmidt, "Applications of a nonlinear smoothing algorithm to speech processing," *IEEE Trans. Acoust., Speech, Signal Processing,* vol. ASSP-23, pp. 552-557, Dec. 1975.

APPENDIX

Predictor Coefficients $h(1) \ldots h(22)$

$\omega_0 =$ 0.0588π	0.0591π	0.0594π	0.0597π	0.06π	0.0603π	0.0606π	0.0609π	0.0612π
0.161107	0.160254	0.159415	0.158591	0.15778	0.156983	0.156199	0.155429	0.154672
0.132909	0.132283	0.131648	0.131003	0.130349	0.129686	0.129013	0.12833	0.127638
0.10773	0.107498	0.107245	0.106968	0.106669	0.106347	0.106002	0.105632	0.105239
0.0861204	0.0863237	0.0865021	0.0866553	0.0867828	0.0868842	0.0869591	0.087007	0.0870273
0.0670132	0.0675249	0.0680165	0.0684875	0.0689376	0.0693664	0.0697735	0.0701585	0.0705209
0.0482957	0.048845	0.0493813	0.0499042	0.0504137	0.0509095	0.0513916	0.0518596	0.0523135
0.0277106	0.0279599	0.0282017	0.028436	0.0286629	0.0288827	0.0290954	0.0293012	0.0295004
0.0037984	0.0034547	0.0031063	0.0027537	0.0023971	0.0020371	0.0016742	0.0013089	0.0009419
-0.0234064	-0.0244828	-0.0255619	-0.026643	-0.0277255	-0.0288087	-0.0298917	-0.0309737	-0.0320537
-0.052143	-0.0538706	-0.0555963	-0.0573193	-0.0590387	-0.0607536	-0.062463	-0.0641659	-0.0658611
-0.0792344	-0.0813095	-0.083373	-0.0854237	-0.0874607	-0.089483	-0.0914896	-0.0934794	-0.0954512
-0.100854	-0.102822	-0.104764	-0.106679	-0.108564	-0.11042	-0.112245	-0.114039	-0.115801
-0.113523	-0.114908	-0.116249	-0.117546	-0.118798	-0.120004	-0.121166	-0.122281	-0.12335
-0.115056	-0.115496	-0.115881	-0.11621	-0.116485	-0.116706	-0.116871	-0.116983	-0.117042
-0.105153	-0.104519	-0.103831	-0.103089	-0.102294	-0.101448	-0.100552	-0.0996078	-0.0986159
-0.085487	-0.0839283	-0.082331	-0.0806972	-0.0790288	-0.077328	-0.0755969	-0.0738378	-0.0720528
-0.0592498	-0.057152	-0.0550453	-0.0529322	-0.0508153	-0.0486972	-0.0465806	-0.0444681	-0.0423622
-0.030284	-0.028154	-0.0260495	-0.023973	-0.0219269	-0.0199135	-0.0179351	-0.0159941	-0.0140925
-0.0020716	-0.0003779	0.001264	0.0028526	0.0043865	0.0058647	0.007286	0.0086494	0.0099542
0.0231323	0.0241112	0.025033	0.025898	0.0267068	0.02746	0.0281585	0.028803	0.0293947
0.0447638	0.045033	0.0452687	0.0454732	0.045649	0.0457986	0.0459246	0.0460296	0.0461162
0.0638817	0.0637324	0.0636003	0.063489	0.0634021	0.063343	0.0633149	0.0633211	0.0633646