An ultra-precise Fast Fourier Transform

Henry, M.

Original citation:
Henry, M 2024, 'An ultra-precise Fast Fourier Transform', Measurement: Sensors, vol. 32, 101039. https://doi.org/10.1016/j.measen.2024.101039

DOI 10.1016/j.measen.2024.101039
ESSN 2665-9174

Publisher: Elsevier

Measurement: Sensors 32 (2024) 101039

Contents lists available at ScienceDirect

Measurement: Sensors

journal homepage: www.sciencedirect.com/journal/measurement-sensors

An ultra-precise Fast Fourier Transform

Manus Henry [a,b,*]

[a] Fluid and Complex Systems Research Centre, Coventry University, UK
[b] Department of Engineering Science, University of Oxford, UK

ARTICLE INFO

Keywords:
FFT
Prism signal processing
Spectral analysis
Spectral leakage
Hidden tones

ABSTRACT

The Fast Fourier Transform (FFT) is a cornerstone of digital signal processing, generating a computationally efficient estimate of the frequency content of a time series. Its limitations include: (1) information is only provided at discrete frequency steps, so further calculation, for example interpolation, may be required to obtain improved estimates of peak frequencies and amplitudes; (2) 'energy' from spectral peaks may 'leak' into adjacent frequencies, potentially causing lower amplitude peaks to be distorted or hidden; (3) the FFT is a discrete time approximation of continuous time mathematics. A new FFT calculation addresses each of these issues through the use of two windowing functions, derived from Prism Signal Processing. Separate FFT results are obtained by applying each windowing function to the data set. Calculations based on the two FFT results yields high precision estimates of spectral peak location (frequency) amplitude and phase while suppressing spectral leakage.

1. Introduction

The Fast Fourier Transform (FFT) has been described as "the most important numerical algorithm of our lifetime", used billions of times a day [1]. Improving FFT performance is a long-standing research topic [2], whether by improving the precision of frequency, amplitude and phase (FAP) estimates of spectral components, or by locating low amplitude 'hidden tones'. A recent paper [3] applies Prism Signal Processing (PSP) [4] to the FFT problem and provides a useful background to the current work.

The Prism is a signal processing node which provides one or two linear phase FIR filters via a recursive, low-cost calculation. Prism networks can instantiate bandpass [5] and lowpass filters [6], as well as 'trackers' [6] which calculate the FAP content of a sinusoidal input. The Prism was conceived as an alternative to the (almost universally used) convolutional form of FIR filter, and was intended for real-time applications, providing sample-by-sample outputs for a sample-by-sample time series input.

By contrast, the FFT is usually viewed as a single calculation applied to a static data set, generating a vector of frequency and (complex) amplitude results. In Ref. [3], PSP was applied to spectral analysis by treating the data set as a time series. Narrow bandpass filters and trackers are created to locate spectral peaks and calculate the corresponding amplitude and phase values. While this approach can successfully determine spectral peak parameters to high precision, and detect 'hidden tones' (for example, locating a low amplitude peak 10^{-9} times smaller than an adjacent peak), the computational cost associated with the design and operation of many filters is much higher than the *nlog(n)* efficiency of the FFT.

This paper presents a more efficient synthesis of PSP and FFT, as follows. A 'low pass' Prism filter [6] provides a narrow passband and high stop band attenuation, where the filter peak is close to, but offset from, zero Hz. Combined with a Prism tracker, FAP values can be calculated for signal components at or close to the filter peak [6]. A fixed filter/tracker structure can be used to target an arbitrary frequency via heterodyning [6]. This shifts the target frequency to coincide with the filter peak, multiplying the input data by a pure sinusoid with its frequency equal to the difference between the filter peak and target

* Fluid and Complex Systems Research Centre, Coventry University, UK.
E-mail address: manus.henry@coventry.ac.uk.

https://doi.org/10.1016/j.measen.2024.101039
Received 25 July 2022; Received in revised form 7 August 2023; Accepted 21 January 2024
Available online 23 January 2024

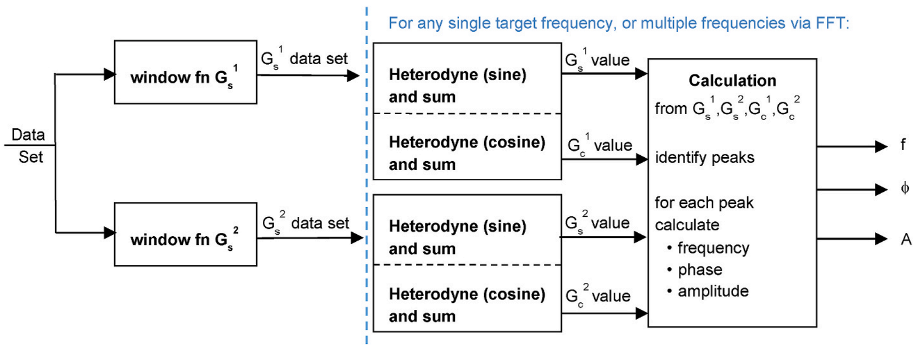

Fig. 1. Prism FFT method.

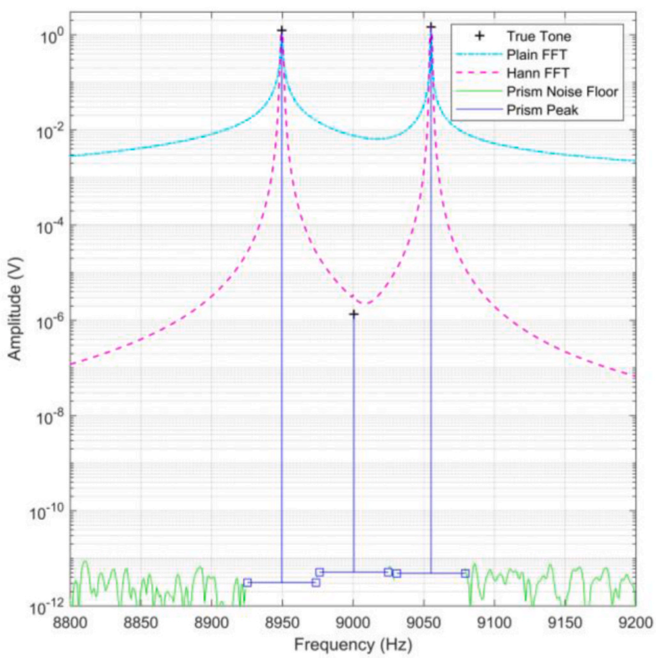

Fig. 2. FFT results for three tone simulation.

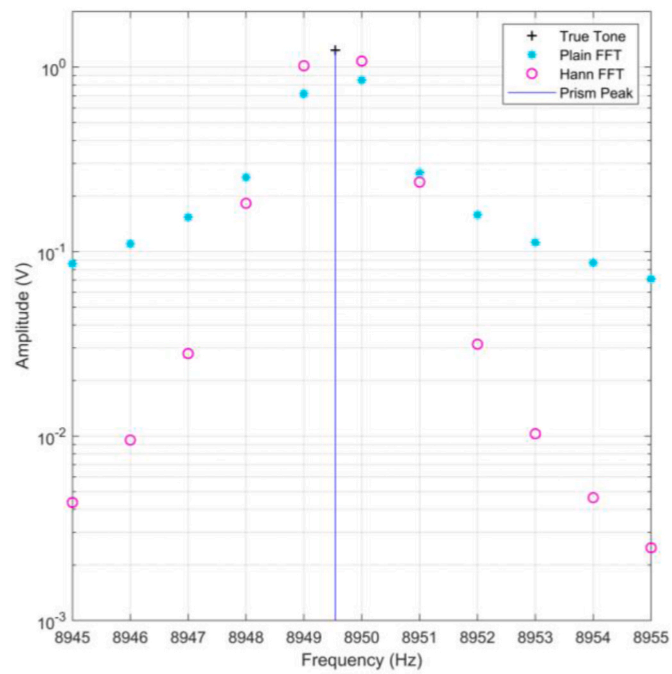

Fig. 3. FFT results around first peak.

frequencies. Instead of treating the data set as a time series, convolutional equivalents of the Prism filter/tracker structure are created to enable single shot calculations across the entire data set. Finally, the natural order of calculation (heterodyne, then filter/track) is reversed. The convolutional forms of the filter/tracker are applied as windowing functions to the data set (i.e. deferring the final summation); the conventional FFT calculation applied to the windowed data then provides an efficient means of applying regularly-spaced heterodynes (including a final summation). Prism tracking calculations applied to the resulting values yields high precision FAP data across the spectrum.

Fig. 1 summarizes the technique. Windowing functions G_s^1 and G_s^2, are convolutional equivalents of Prism networks, combining a common low pass filter with different tracking Prisms. The resulting windowed data sets can be 'interrogated', either at a single frequency, or systematically via FFT calculations, to generate sets of values from which peak locations (i.e. frequency), amplitude and phase can be calculated.

2. Simulation

A simulation is used to compare the performance of the Prism FFT technique with conventional FFT methods, using a 1 s window of data sampled at 48 kHz. Three tones are present with frequency and amplitude values $f_1 = 8949.54321$ Hz, $a_1 = 1.23456$ V, $f_2 = 9000.56789$ Hz, $a_2 = 1.34567\text{e-}6$ V, $f_3 = 9055.12345$ Hz, $a_3 = 1.45678$ V. Random initial phases are supplied to each tone, and white noise with standard deviation 1e-10 V is added. Fig. 2 shows the true tones and three spectral analyses: FFT only ('Plain'), FFT using the Hann(ing) windowing function [7], and Prism FFT. On this scale, all methods appear to correctly identify the outer, high amplitude peaks, but the central peak is only successfully located by the Prism technique, which has a noise floor several orders of magnitude lower than the others. Fig. 3 shows the FFT results in the vicinity of the first peak in more detail. Amplitude results are provided in 1 Hz steps for the Plain and Hann methods, and the location and amplitude of the peak can only be approximately

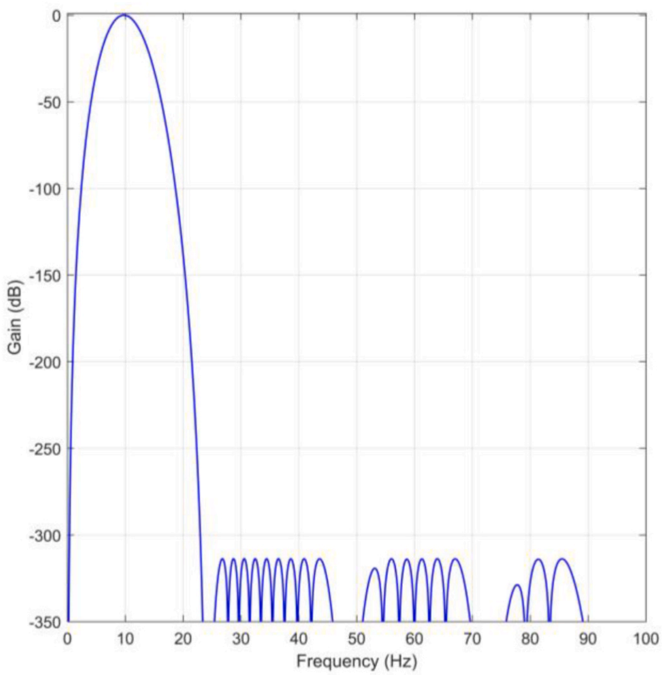

Fig. 4. Prism low pass filter.

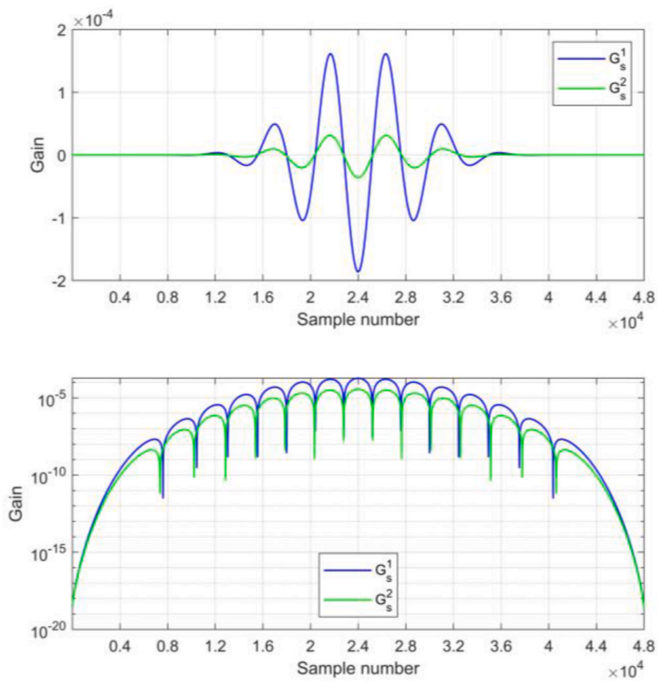

Fig. 5. Windowing functions G_s^1 and G_s^2.
(above) linear scale; (below) logarithmic scale.

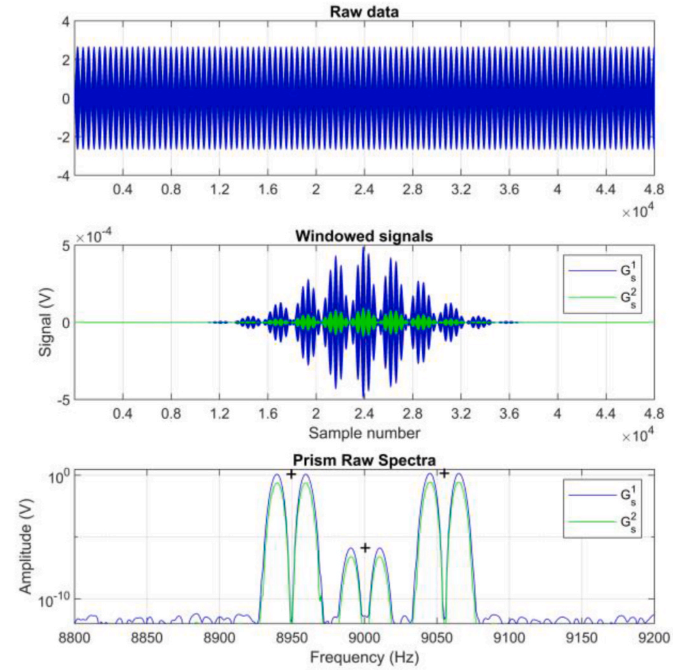

Fig. 6. Windowing functions applied to data.
(top) raw data; (middle) data after applying windowing functions; (bottom) FFT results for windowed data.

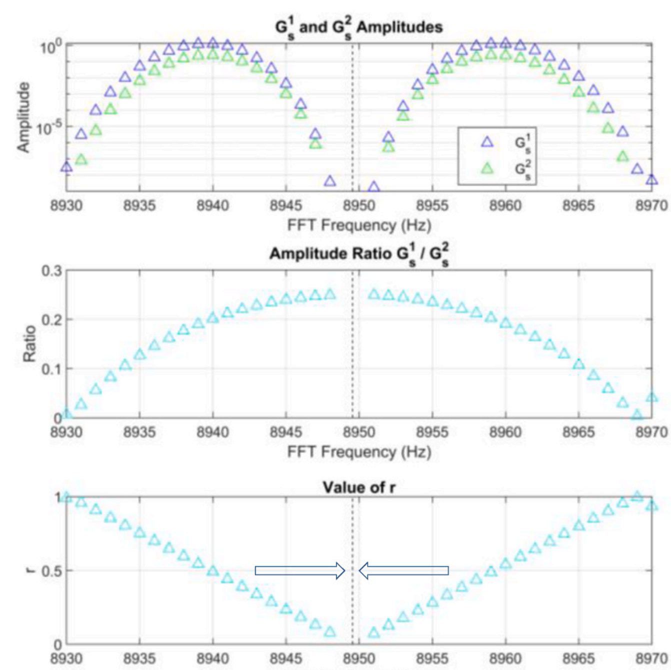

Fig. 7. Calculation of *r*. (top) amplitudes from G_s^1 and G_s^2 windowing functions; (middle) amplitude ratio; (bottom) value of r.
Dashed line shows true peak frequency at ~ 8949.5 Hz.

determined. By contrast the Prism peak estimate still appears to be accurate at this level of detail: as shown later, the frequency, amplitude and initial phase of the peak are each correctly calculated by the Prism

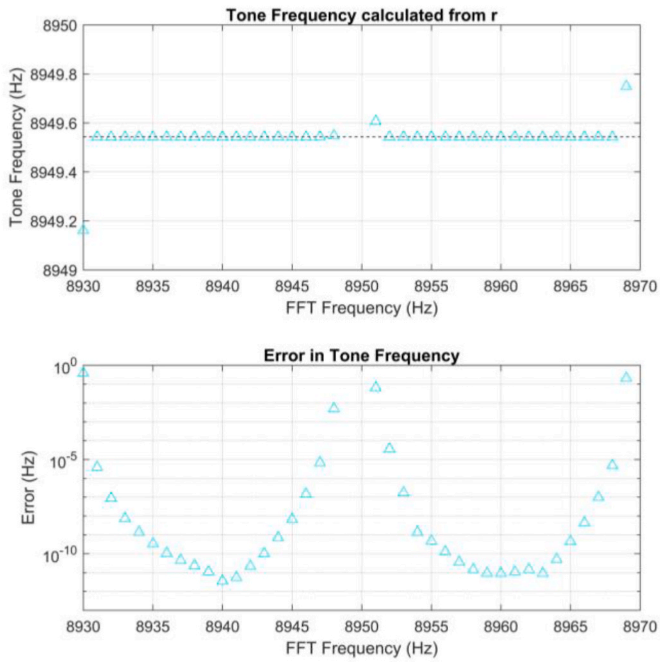

Fig. 8. Calculation of Tone 1 location
(above) estimates of peak based on data at each FFT reporting frequency; (below) frequency error.

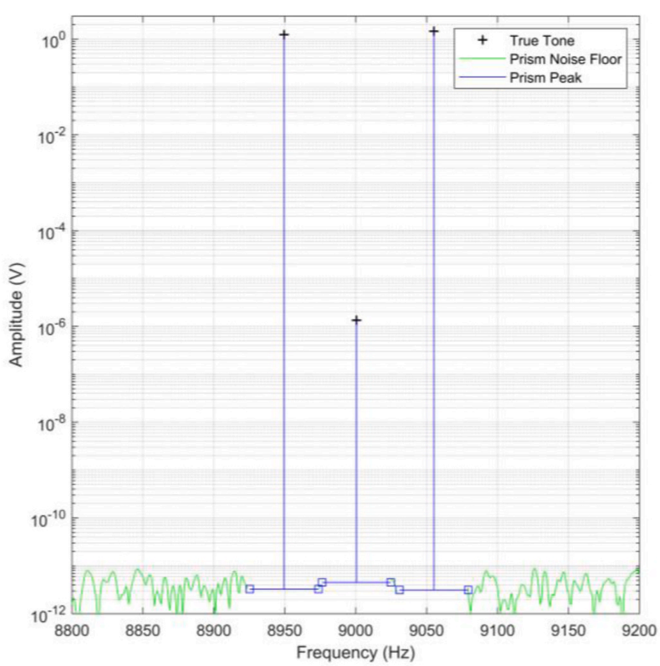

Fig. 9. Prism spectrum including dead-zones.

technique to approximately 1 part in 10^{12}.

Fig. 4 shows the low pass filter design, while Fig. 5 shows the windowing functions G_s^1 and G_s^2, each composed of the low pass filter followed by a (different) tracking Prism, in convolutional form. Fig. 6 shows the windowing functions applied to the raw data. The corresponding FFT results reflect the characteristics of the low pass filter. Frequency leakage is suppressed, so the noise floor is low (~1e-12V) and the low amplitude middle tone is revealed. However, there are double peaks, offset by approx. ± 10 Hz from each true tone. These arise because the low pass filter (Fig. 4) has a peak at ~ 10 Hz; viewed as a set of heterodynes, the FFT calculation generates both positive and negative frequency offsets. This effect does not occur for conventional FFT windowing functions (e.g. Hann) where the peak frequency is zero Hz. Figs. 7 and 8 show how the raw Prism spectra results are used to locate tone 1.

In Fig. 7, the amplitude ratio G_s^2/G_s^1 is calculated for each FFT frequency step. A quadratic transformation of the ratio calculates *r* [4], a dimensionless frequency, here indicating the distance to the local peak. The positive and negative slopes of *r* resolve the ambiguity of the dual peak. In Fig. 8, estimates of the true tone frequency are shown, with the corresponding errors falling to 2e-12 Hz.

Having located the true tone frequency, accurate amplitude and phase results are calculated from new G_s^1 and G_c^1 values (Fig. 1) obtained at this exact frequency (i.e. by performing an additional, single frequency heterodyne on the windowed data sets). Fig. 9 shows the final Prism spectrum. Outside of peak regions, the noise floor is provided by the G_s^1 amplitude data (Fig. 6, lowest plot). Around each peak, a 'dead-zone' is created, to suppress local frequency leakage, in particular the double peak effect. The dead-zone width is determined by the filter passband (here approx. ± 25 Hz, Fig. 4) while the dead-zone amplitude is selected to match that of the adjacent raw noise floor.

Table 1 shows the mean error and standard deviation for each tone parameter obtained from 5000 simulations, where additive white noise and the initial phase of each tone is randomized. Good results are obtained even for Tone 2, which is not observable using the conventional FFT techniques (Fig. 2). The computational cost is approximately double that of the Hann FFT.

3. Concluding note

This paper is a revised version of an extended abstract submitted to the MeSSAC 2022 Conference. A fuller description of the Prism FFT has now been published as [8].

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests:

Manus Henry has patent Filter System and Method of Designing a Convolutional Filter pending to Oxford University Innovation. Manus Henry has patent Methods and Systems for Filtering an Input Signal and for Estimating Characteristics of an Input Signal pending to Oxford University Innovation. Manus Henry has patent METHOD AND SYSTEM FOR TRACKING SINUSOIDAL WAVE PARAMETERS FROM A RECEIVED SIGNAL THAT INCLUDES NOISE pending to Oxford University Innovation. Manus Henry has patent METHOD AND SYSTEM FOR ULTRA-NARROWBAND FILTERING WITH SIGNAL PROCESSING USING A

Table 1
Prism FFT results for 5000 simulations.

	Tone 1		Tone 2		Tone 3	
	Mean error	standard deviation	Mean error	standard deviation	Mean error	standard deviation
Frequency (Hz)	−8.09e-13	2.14e-12	−6.10e-07	1.60e-06	−1.50e-12	2.07e-12
Amplitude (V)	−7.83e-13	1.55e-12	−2.83e-13	1.55e-12	−6.28e-13	1.58e-12
Phase (radians)	2.32e-12	7.29e-12	1.89e-06	5.18e-06	2.35e-12	6.81e-12

CONCEPT CALLED PRISM issued to Oxford University Innovation.

Data availability

No data was used for the research described in the article.

References

[1] D. Schneider, "A Faster Fast Fourier Transform", IEEE Spectrum, March 2012.
[2] P. Duhamel, M. Vetterli, Fast Fourier Transforms: a tutorial review and a state of the art, Signal Process. 19 (1990).
[3] M.P. Henry, Spectral analysis techniques using Prism signal processing, Measurement 169 (2021).
[4] M.P. Henry, "The prism: recursive FIR signal processing for Instrumentation applications" IEEE, Trans. Instr. Meas. 69 (2020).
[5] M.P. Henry, Building Billion Tap Filters, IEEE Industrial Electronics Magazine, Nov 2020.
[6] M.P. Henry, Low pass, low cost prism filters, in: 2020 IEEE International Workshop on Metrology for Industry 4.0 & IoT, June 2020. Rome.
[7] D. Belega, D. Petri, D. Dallet, Accurate amplitude and phase estimation of noisy sine-waves via two-point interpolated DTFT algorithms, Measurement 127 (2018).
[8] M.P. Henry, An ultra-precise Fast fourier Transform, Measurement 220 (2023) 113372, https://doi.org/10.1016/j.measurement.2023.113372. ISSN 0263-2241.