



**G. D. Conway**

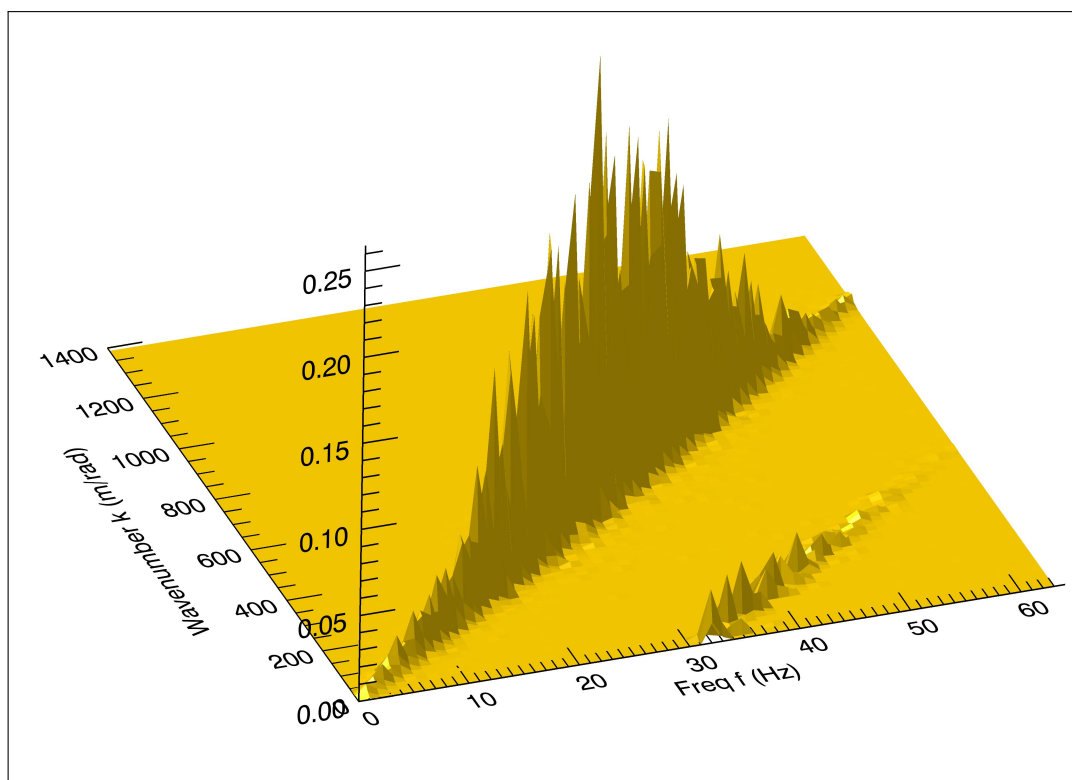
**A short manual to discrete  
time series analysis**

**IPP 2024-04  
Januar 2024**

not so ✓

# A short manual to discrete time series analysis

G.D.Conway



Max-Planck-Institut für Plasmaphysik, Boltzmannstrasse 2, Garching D-85748, Germany

Revision: v.1.0 : February 6, 2015

v.1.2 : April 9, 2020

v.2.1 : June 2, 2023

v.2.2 : January 22, 2024

Cover figure: Local frequency-wavenumber spectra using the 2-point Beall histogram method.

Manuscript prepared using L<sup>A</sup>T<sub>E</sub>X2e in 12pt Times font.

# Contents

<b>1</b>	<b>Outline</b>	<b>5</b>
1.1	Definitions . . . . .	6
1.1.1	Terminology . . . . .	6
1.1.2	Symbols . . . . .	7
1.2	Digitization basics . . . . .	8
1.2.1	Discretization . . . . .	8
1.2.2	Sampling . . . . .	8
<b>2</b>	<b>Synthetic signals</b>	<b>9</b>
2.1	Basic harmonic signals . . . . .	9
2.2	Basic noise signals . . . . .	9
2.3	Distorted harmonic signals . . . . .	11
2.4	Tailored broadband signals . . . . .	12
2.5	Non-stationary signals . . . . .	14
2.6	Intermittent signals . . . . .	14
<b>3</b>	<b>Probability functions</b>	<b>17</b>
3.1	Moments and cumulants . . . . .	19
3.2	Errors in moment estimations . . . . .	21
3.3	Temporal asymmetry . . . . .	21
3.4	Types of distributions . . . . .	22
3.4.1	Poisson distribution . . . . .	22
3.4.2	Gaussian (or normal) distribution . . . . .	22
3.4.3	Lorentzian (or Cauchy) distribution . . . . .	22
3.4.4	Log-normal distribution . . . . .	22
3.4.5	Rayleigh (Exponential) distribution . . . . .	22
3.4.6	Gamma distribution . . . . .	22
3.4.7	Weibull distribution . . . . .	23
3.4.8	Rician (or Rice) distribution (non-zero mean) . . . . .	23
3.4.9	Voigt distribution . . . . .	23
<b>4</b>	<b>Stationarity / Randomness tests</b>	<b>25</b>
4.1	Runs test . . . . .	25
<b>5</b>	<b>Correlation functions</b>	<b>27</b>
5.1	Lagged Auto & Cross-association . . . . .	27
5.2	Lagged Auto & Cross-correlation . . . . .	27
5.3	Confidence limits & confidence intervals . . . . .	29
5.3.1	Standard error SE (normal) . . . . .	29
5.3.2	Fisher transform (non-normal) . . . . .	29

5.3.3	Large lag standard error . . . . .	30
5.4	Scaled correlations . . . . .	30
5.5	Weighted correlations . . . . .	31
5.6	Nonlinear correlations . . . . .	32
<b>6</b>	<b>Spectral analysis - Linear</b>	<b>33</b>
6.1	Least squares fitting . . . . .	34
6.2	Fourier transforms . . . . .	34
6.2.1	Discrete Cosine transform . . . . .	35
6.2.2	FFT . . . . .	35
6.3	Auto and cross-spectra . . . . .	36
6.4	Fourier transform issues . . . . .	37
6.4.1	Periodicity & Aliasing . . . . .	37
6.4.2	Leakage and data windowing . . . . .	37
6.4.3	Zero padding . . . . .	39
6.4.4	Mode amplitude from FFT . . . . .	40
6.4.5	Truncation and partial zero padding . . . . .	41
6.4.6	Convolution and correlation . . . . .	43
6.4.7	Overlap averaging . . . . .	43
6.4.8	Cepstrum . . . . .	43
6.5	Confidence limits & confidence intervals . . . . .	44
6.5.1	Coherence significance level . . . . .	45
6.6	Blackman-Tukey method . . . . .	46
<b>7</b>	<b>Spectral analysis - Non-linear</b>	<b>49</b>
7.1	Autoregressive moving-average models . . . . .	49
7.1.1	AR spectra and Yule-Walker equations . . . . .	49
7.1.2	Maximum Entropy Method (MEM) . . . . .	53
7.1.3	Least-Squares spectral estimator . . . . .	53
7.1.4	Moving average spectra (ARMA) . . . . .	54
7.1.5	AR summary . . . . .	54
7.2	Eigenvector methods . . . . .	55
7.2.1	Pisarenko Harmonic Decomposition (PHD) . . . . .	55
7.2.2	MUSIC method (MULTiple Signal Classification) . . . . .	56
7.2.3	ESPRIT method . . . . .	60
<b>8</b>	<b>Frequency-wavenumber spectra</b>	<b>61</b>
8.1	Double transform method $S(\omega, k)$ . . . . .	61
8.2	Multichannel AR . . . . .	61
8.3	Two point correlation methods . . . . .	62
8.3.1	Narrow-band covariance . . . . .	62
8.3.2	Cross-phase spectra . . . . .	62

8.4	Local frequency-wavenumber spectra $S_l(\omega, k)$ . . . . .	63
8.5	Space-time correlations . . . . .	66
8.5.1	Spectral line shape . . . . .	66
8.5.2	Correlation phase velocity . . . . .	66
8.5.3	Statistics of the auto-correlation . . . . .	67
8.6	Biorthogonal decomposition . . . . .	68
<b>9</b>	<b>Higher order spectra</b>	<b>69</b>
9.1	Bispectrum and bi-coherence . . . . .	69
9.1.1	Non-linear test signal . . . . .	70
9.1.2	Statistical error in $b$ . . . . .	70
9.2	Triple correlations . . . . .	73
9.3	Trispectrum $T_{xyzw}$ . . . . .	73
9.4	Energy transfer . . . . .	75
9.4.1	Power transfer function (PTF) . . . . .	75
9.4.2	Amplitude Correlation Technique (ACT) . . . . .	76
9.4.3	Kinetic energy transfer (KET) . . . . .	76
<b>10</b>	<b>Time-frequency analysis</b>	<b>77</b>
10.1	Short time or sliding FT . . . . .	77
10.2	Wigner-Ville spectrum . . . . .	77
10.3	Choi-Williams distribution . . . . .	78
10.4	S-method . . . . .	79
10.5	Wavelet transform . . . . .	80
10.5.1	Cross-wavelet spectra . . . . .	81
10.5.2	End effects and Confidence limits . . . . .	82
10.5.3	Wavelet bicoherence $b_w^2$ . . . . .	83
10.6	Adaptive AR modelling . . . . .	84
10.7	Hilbert transform . . . . .	85
10.7.1	Hilbert spectral analysis . . . . .	85
10.7.2	Envelope detection . . . . .	86
10.7.3	Complex demodulation . . . . .	86
10.8	Hilbert-Huang transform HHT . . . . .	87
<b>11</b>	<b>Non-linear structure</b>	<b>89</b>
11.1	Increments and difference functions . . . . .	89
11.2	Structure functions . . . . .	89
11.3	Hurst exponent . . . . .	89
11.3.1	Generalized exponent . . . . .	90
11.3.2	Rescaled range analysis . . . . .	91
11.3.3	Fractal dimension $D$ . . . . .	91
11.4	Nonlinear correlation functions . . . . .	92

<b>12 Conditional and non-stationary analysis</b>	<b>93</b>
12.1 Event amplitude . . . . .	93
12.1.1 Conditional probability density functions . . . . .	93
12.1.2 Conditional (box-car) averaging . . . . .	93
12.1.3 Conditional & Gated correlations . . . . .	95
12.1.4 Wavelet analysis . . . . .	96
12.2 Event timing . . . . .	96
12.2.1 Level crossing . . . . .	96
12.2.2 Spectral exponent . . . . .	97
12.2.3 Clustering exponent . . . . .	97
12.2.4 Waiting times . . . . .	100

**Acknowledgments** - special mention to Boudewijn van Milligen, Patrick Simon, Pascale Hennequin, Christian Yoo and to the many colleagues and students over the last 40 years who have contributed one way or another.

# 1 Outline

This manual describes some useful techniques and methods applied to discrete signal analysis, e.g. digitized signals. Digital signal processing is a wide ranging topic with a long history. There is an extensive range of excellent and detailed books and monographs available, as well as thousands of journal publications in a range of scientific (and non-scientific) areas. The various journals and Proceedings of the IEEE are a standard source for early papers on the fundamental techniques. In terms of references - the standard reference is probably the Bendat and Piersol tome, which is often referred to as the bible of signal processing.<sup>1</sup> It is wide ranging and highly accessible. Other, somewhat older, references on the basics include:<sup>2 3 4 5 6 7</sup> There are many more-recent text books drawn from varying fields of application - here preference is a matter of taste!

I can also highly recommend the lecture notes of Dave Meko on applied time series analysis.<sup>8</sup> For method implementation the standard must be the excellent Numerical Recipes.<sup>9</sup> Not least, Wikipedia can be a useful source, although the quality of information can be rather mixed.

Where possible, key original journal references (drawn from generally accessible journals), as well as illustrative references to applications in the field of plasma turbulence are provided for each topic. In general, each topic is supported with relevant formulas, but not lengthy derivations, as well as code snippets (in IDL) and demonstrative example figures. IDL (Interactive Data Language) is a well established, if rather old, high level programming language for data analysis. The code snippets should be relatively self-evident, and thus easily translatable to alternate languages, such as python etc.

A passing note, just about every signal analysis technique has its basis in statistics - and the starting point is the probability distribution and its moments. It is not essential to be an expert in statistics to use signal analysis methods, but a basic knowledge will, at least, provide an appreciation of why and how certain techniques are used.

This guide is not by any means fully comprehensive, but is intended to provide the basics and to point perhaps to some unfamiliar analysis methods. Enjoy.

---

<sup>1</sup> Bendat J.S. and Piersol A.G. *Random data: Analysis and measurement procedures*, 4th edn. Wiley, N.Y. (2010). First published in 1971.

<sup>2</sup> Otnes R.K. & Enochson L. *Applied time series analysis*, Wiley N.Y. (1978).

<sup>3</sup> Marple, S.L. *Digital spectral analysis with applications*, Prentice-Hall (1987).

<sup>4</sup> Oppenheim A.V. *Discrete-time signal processing*, Prentice Hall (1989).

<sup>5</sup> Mitra S.K. *Digital signal processing*, 2nd edn. McGraw-Hill (2001).

<sup>6</sup> Proakis J.G. & Mandakis D.G. *Digital signal processing*, 4th edn. Pearson (2006).

<sup>7</sup> Chatfield C. *The analysis of time series: An introduction*, 6th edn. Chapman & Hall, New York (2004).

<sup>8</sup> Meko D.M. *GEOS585A, Applied time series analysis : Course notes*, Univ. of Arizona (2019). Download from <http://www.ltr.arizona.edu/~dmeko/>

<sup>9</sup> Press W. et al *Numerical recipes in Fortran - The art of scientific computing*, 2nd edn. Cambridge Uni. Press (1992).



## 1.1 Definitions

### 1.1.1 Terminology

To begin with, a few basic definitions, with reference to symbols in next section:

**Deterministic** means that the signal  $x_n$  can be exactly predicted from previous values. When not, then  $x_n$  is termed **stochastic** and can only be predicted with some probability.

**Stationary vs non-stationary.** Strict stationary means that the probability density function pdf( $x$ ) (see below) is time invariant. However, weak stationarity means that only the moments of the pdf are time invariant, while the pdf itself may not be.

**Periodic vs aperiodic** - the signal is dominated by a repetitive behaviour, or not.

**Univariate vs multivariate** - the signal is dependent on one, or many independent variables.

**Densities and distributions** - Densities have units, distributions can be dimensionless.

**Orthogonal** - if two signal distributions  $p_x$  and  $p_y$  are independent then they are said to be orthogonal.

**Adaptive vs non-adaptive.** Basis functions are either predefined (non-adaptive), as in sinewaves or wavelets for Fourier and Wavelet transforms, or they are obtained from the data (ie. they adapt or change), as in the Empirical mode decomposition.

**Local vs non-local** - the signal is localized in time or space, or not localized.

**Linear vs non-linear** - Whether the signal amplitude behaviour is linear or not.

**Expected value** - or expectation operator  $E[\cdot]$  is a fundamental concept in probability theory. Put simply, it is the mean, or ensemble average value.

**Estimator.** The imperfect estimation of a real parameter comes with an associated bias and with errors. Estimators are indicated by the hat symbol  $\hat{A}$ .

**Bias error** - bias $[\hat{A}]$  is the amount by which an estimator consistently over or under estimates the real parameter.

**Variance error** - var $[\hat{A}]$  is the amount of variation of the estimator about the real parameter. The normalized standard error (random) is  $\epsilon[\hat{A}] = \sigma[\hat{A}]/A$  (ie. normalized standard deviation).

**Degrees of freedom (dof)** The dof  $\nu = N - p$  is the sample size  $N$  minus  $p$  the number of parameters (or constraints) calculated. Thus the dof of the sample mean is  $\nu = N - 1$  etc.

**PDF** - Probability Density Function, pdf  $p(x)$ .

**PSD** - Power Spectral Density, psd( $f$ ). Often simply called the power spectrum  $S(f)$ .

**Periodogram** - The PSD computed from the squared Fourier transform (FT)  $\mathcal{F}$  of the signal  $(\mathcal{F}\{x_n\})^2 \rightarrow S(f)$ .

**Correlogram** - The PSD from the FT of the autocorrelation (auto covariance)  $R_{xx}(\tau)$ .

**Windowing** - is the application of a shaped profile (of limited extent, ie. a window) to either the signal  $x(t)$ , the time lagged correlation  $R(\tau)$ , or the frequency spectral estimate  $S(f)$ .

### 1.1.2 Symbols

Symbol	Meaning	Range
$i$	$\sqrt{-1}$	
$n$	Time index	$\{n = 0, \dots, N - 1\}$
$j$	Frequency index	$\{j = 0, \dots, J - 1\}$
$k$	Lag index	$\{k = 0, \dots, K - 1\}$
$l$	Spatial index	$\{l = 0, \dots, L - 1\}$
$m$	Ensemble index & Signal mean	$M = \text{total number of ensembles/averages}$
$b^2$	Bicoherence squared	
$f, \omega$	Frequency ( $\omega = 2\pi f$ )	
$k$	Wavenumber ( $k = 2\pi/\lambda$ )	
$p, q$	Moment order of pdf or spectrum	
$p(x)$	Probability	
$R$	Correlation function	
$r, \rho$	Correlation coefficient	$-1 < r < +1$
$S$	Power spectral density (psd), Skewness	
$t$	Time	$0 \leq T$
$w, h$	Window or weight fcnct.	
$x(t)$	Continuous data	
$\{x_n\}$	Discretized data	$\{n = 0, \dots, N - 1\}$
$\alpha$	Confidence level, $p = (1 - \alpha)$	
$\beta$	Spectral index (also sometimes $m$ )	
$\delta_{f,k}$	Spectral widths (freq., wavenumber)	
$\delta x$	Increment (signal or spatial)	
$\epsilon$	Normalized standard error	
$\gamma$	Coherence	
$\lambda$	Wavelength	
$\lambda_i$	Eigenvalue	
$\mu_p, \kappa_p$	Moment / Cumulant of order $p$	
$\sigma$	Standard deviation, $var = \sigma^2$	
$\tau$	Time delay or lag	$\{\tau = 0, \dots, K\Delta t\}$
$\theta$	Cross-phase	
$\xi$	Spatial interval or separation	
$\Delta t$	Sample interval or period	
$\Delta f$	Frequency interval	
$\mathcal{F}$	FT transformation	
$\mathcal{H}$	Hilbert transform	
$cov[\cdot]$	Operator	
$cov(\cdot)$	Function	
$\hat{S}, E[\cdot]$	Estimated (hat), Expected value	

## 1.2 Digitization basics

### 1.2.1 Discretization

Digital sampling of an analog time varying signal  $x(t)$  involves sampling in both time and amplitude. The sampling rate or frequency  $f_s = 1/\Delta t$  (where  $\Delta t$  is the sample period or interval) defines the **maximum resolvable frequency** of the signal. To avoid aliasing effects the sample rate should be at least twice the maximum expected frequency component, defined by the Nyquist frequency  $f_N$  - see below.

The signal amplitude is also quantized into discrete levels set by the bit resolution and the input (voltage  $V$ ) range of the digitizer. The **quantization error** (assuming a uniform signal distribution between quantization levels) has a standard deviation of approx.  $0.29\Delta V$ . Thus for a modest 12-bit digitizer the maximum signal to noise ratio (SNR) is  $2^{12}\Delta V/0.29\Delta V$  or 83 dB - which is a lot! Thus dynamic range is rarely an issue for modern computers and electronics. See Bendat and Persol for fuller details. <sup>1</sup>

### 1.2.2 Sampling

An issue of signal discretization is the occurrence of **aliasing** where only signal frequency components  $f < f_N$  less than the **Nyquist frequency** can be resolved - that is, one needs at least 2 time samples per shortest cycle in order to resolve it unambiguously. To meet the Nyquist frequency one must either sample the data at at-least twice the highest frequency, or analog low-pass filter (anti-aliasing) the signal to below  $f_N = f_s/2 = 1/(2\Delta t)$  before digitizing.

It is often required to re-sample the digitized signal to a new sample rate. To **down-sample** the data the signal should first be digitally filtered to just below the new Nyquist frequency to avoid aliasing. Then re-sampled, using where possible interleaving to recover as much of the data as possible.

Example: Down-sample by a factor of  $d = 3$ :  $\Delta t_{(new)} = d \times \Delta t$  and  $f_{(N_{new})} = 1/(2.d.\Delta t)$

$$\begin{array}{ll} x = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, \dots N-1\} & T = N.Dt \\ x1 = \{0, \quad 3, \quad 6, \quad 9, \quad \dots N-3\} & T = N/3 \quad (3.Dt) \\ x2 = \{ \quad 1, \quad 4, \quad 7, \quad 10, \quad \dots N-2\} & \\ x3 = \{ \quad 2, \quad 5, \quad 8, \quad 11, \quad \dots N-1\} & \end{array}$$

For a fixed, or restricted measurement period  $T = N\Delta t$ , oversampling the signal and then digital down-sampling with interleaved ensemble averaging can bring notable improvements in the spectral SNR, but with additional computational cost. On the other hand, if SNR is not an issue, for example if using digital Inphase-Quadrature (IQ) detection, cf. <sup>10</sup> then the reduced quantization levels (number of bits) can massively reduce computational cost and processing speed.

<sup>10</sup> MIT (EE CS) *Lecture 9: Analog and digital I/Q modulation* : 4/Nov/2006, [web.mit.edu/6.02/www/f2006/handouts/Lec9.pdf](http://web.mit.edu/6.02/www/f2006/handouts/Lec9.pdf) (Checked Feb. 2023).

## 2 Synthetic signals

Testing of codes and analysis methodologies often requires synthetic test signals with known or defined properties - such as harmonic type perturbations to model coherent modes, or random fluctuations with Gaussian amplitude distributions to model broadband turbulence.

### 2.1 Basic harmonic signals

The most universal test signal is a simple harmonic (sinewave) corrupted by white noise generated by a random number generator. Most compute language packages have (pseudo) random number generators with a choice of normal (Gaussian) or uniform amplitude distributions.

```
npt = 8192 & an = 0.1 & fo = 110. & seed = 107 & dt=1/1000 .
time = Findgen(npt)*dt
sinw = Sin(2.*!pi*time*fo)
msig = Moment(sinw)
Print, 'Mom Sig:', msig
noise = an * RandomN(seed, npt)
mnos = Moment(noise)
Print, 'Mom Nos:', mnos
data = sinw + noise
mdat = Moment(data)
Print, 'Mom Data:', mdat
```

*Example. 1 : Code snippet (IDL) for generating, and calculating moments of, a sinewave with added Gaussian distributed white noise.*

### 2.2 Basic noise signals

**2.2.1** In addition to the random number generators mentioned above, a simple way of generating a random spiky signal is to create an exponential signal from a random normal distribution. The frequency distribution retains the same white spectrum but the amplitude distribution is now skewed - see section 3 on probability density functions (pdfs) and their moments.

```
data = RandomN(seed, npt) .
data = (1. - Exp(data))
```

*Example. 2 : Code snippet (IDL) for generating a skewed/exponential random signal.*

**2.2.2** Noise comes in different **colours**. White noise (such as generated by a generic random noise generator routine - with either a uniform or Gaussian / normal amplitude distribution) has a flat or constant spectral intensity or power at all frequencies. Pink noise is dominated by the lower frequencies while blue or violet noise is tilted to the higher frequencies. The shape of the spectral power (amplitude squared) profile is often defined by a spectral index  $P \propto f^\beta$ , where  $\beta = 0$  for white noise,  $\beta = -1$  for pink noise and  $\beta = -2$  for Brownian noise.<sup>11</sup> Blue and violet noise have indexes of  $\beta = 1$  and  $\beta = 2$  respectively. In practice the spectral shape, can be formed by selectively filtering a white noise signal. A quick and simple way to generate

<sup>11</sup> Wikipedia - [https://en.wikipedia.org/wiki/Colors\\_of\\_noise](https://en.wikipedia.org/wiki/Colors_of_noise) (Checked Jan. 2023).

pink-like and violet-like noise is to use a moving average and a moving difference filter (part of the broader category of recursive and non-recursive filters) on a white noise input. Figure 1 shows two example spectra (see section 6.3) generated using the code snippet below. The weight factors  $w_0$ ,  $w_1$  and  $w_2$ , as well as the temporal pre and post index steps, can be adjusted (optimized slightly) to obtain the precise desired spectral shape.

```

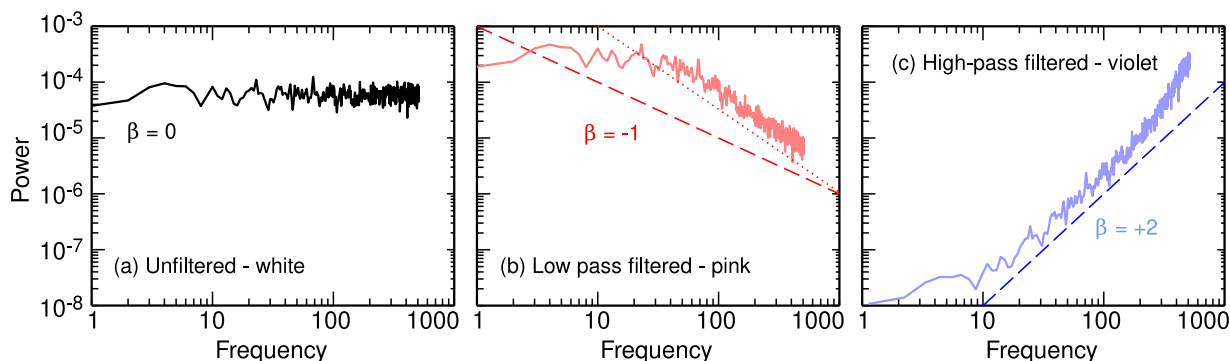
seed = 104 & npt = 8192 & dt = 1/1000 & nft = 1024 & ilp=0 .
noise = 0.4 * RandomN(seed, npt) ; Gaussian random
nsL = noise
w0 = 1.35 & w1 = 1.0 & w2 = 1. ; pinkish
FOR i = 2,npt-3 DO nsL[i] = (w0*nsL[i]+w1*nsL[i+1]+w2*nsL[i-2])/2.
nsH = noise
w0 = 0.95 & w1 = -1.0 & w2 = -1.0 ; violet
FOR i = 2,npt-3 DO nsH[i] = (w0*nsH[i]+w1*nsH[i+1]*w2*nsH[i-1])/2.

; Example FFT spec
win = (1.-Cos(2.*!pi*Findgen(nft)/nft))/2. ; window
SpL = Fltarr(nft) & frq = Findgen(nft)/nft/dt
WHILE itp LT npt-nft-1 DO BEGIN ; 50% overlap sum
  SpL += Abs(FFT(win*nsL[itp:itp+nft-1],-1))^2
  SpH += Abs(FFT(win*nsH[itp:itp+nft-1],-1))^2
  itp += nft/2 & ilp +=1 ; inc. index
ENDWHILE
SpL /= ilp & SpH /= ilp ; normalize

```

*Example. 3 : Code snippet (IDL) to generate pseudo pink and violet noise from white - cf. fig. 1.*

Note in the example shown there is a degree of inherent flattening of the spectra at low frequencies. It is tricky to get perfect behaviour over many frequency decades, nevertheless, it gives an example of what is possible.



*Figure. 1: Synthetic noise spectra: (left) unfiltered white, (middle) pink and (right) violet noise. Ensemble averaged FFT spectra with  $npt = 8192$ ,  $nft = 1024$ , and Hann window.*

As an aside, the use of temporal weights for modifying the spectral domain has a parallel in the use of a Hann window in the time domain and its equivalence to applying *spectral* weights of  $-0.25, +0.5, -0.25$  to adjacent  $[i - 1, i, i + 1]$  frequency elements.<sup>12</sup>

<sup>12</sup> Smith D.E. et al *Fast-Fourier-Transform spectral analysis techniques as a plasma fluctuation diagnostic tool*, IEEE Trans. Plasma Sci. **PS-2** (12) 261-272 (1974).

**2.2.3** Sometimes a semi-random signal is required. One way is to narrow-band filter the white noise signal. The code snippet below uses a convolving filter to obtain a squarish notch or pass-band. It is used later in demonstrating the autocorrelation error bars.

```

seed = 30 & npt = 4096 & an = 1. & dt = 0.2
fs = 0.2 & fw = 0.05 ; centre f & width
data = an * RandomN(seed, npt, /Double)
fN = 1./(2.*dt) ; Nyquist freq.
flow = 0. > (fs-fw)/fN ; express freq. in
fhigh = (fs+fw)/fN < 1. ; multiples of fN
; for LPF flow=0, for HPF fhigh=1
a = 50 ; typ. value
nterms = 320
filt = Digital_Filter(flow, fhigh, a, nterms) ; calc. filt coefs
dout = Convolve(data, filt, /Edge_wrap) + Mean(data) ; Conv. & add mean

```

*Example. 4 : Code snippet (IDL) for generating narrow-band random signal - ref. IDL help file.*

## 2.3 Distorted harmonic signals

In fusion plasmas, experimental signals with sawtooth-like temporal behaviour, or sinewaves with distorted peaks and troughs often appear due to rotating (magnetic) islands or by ELM (edge localized mode) perturbations.

**2.3.1** A **sawtoothed-like** signal can be generated from a skewed or distorted sinewave by applying a harmonic frequency modulation (FM):<sup>13</sup>

$$x(t) = a \cos \left( 2\pi f_o t + \varphi_o - \frac{\pi}{4} S \cos(2\pi f_o t + \varphi_o) \right) \quad (1)$$

where  $\varphi_o$  is the perturbation phase shift and  $S$  is a skew factor between  $\pm 1.0$  which distorts the sinewave ( $S = 0$ ) into a smooth sawtooth perturbation, as shown in fig. 2 using the code snippet below.

**2.3.2** Alternatively the sine amplitude could be distorted with a **cusp-like** shape using:

$$x(t) = a \left\{ \frac{1}{2^{(C-1)}} [\cos(2\pi f_o t + \varphi_o) + 1]^C - 1 \right\} \quad (2)$$

where  $C > 0$  is a cusping factor. For a sinewave  $C = 1$ .

These two types of asymmetric perturbations are quite distinct. The first is a distortion of the slopes (or time derivative) of the perturbation, i.e. a non-symmetric  $x' = dx/dt$  probability density function (pdf). The second is distortion to the amplitude distribution, i.e. a non-symmetric amplitude  $x$  pdf. As shown in the code snippet below they can also be combined together.

```

t = Findgen(npt)
sigS = Cos(2.*!pi*fo*t - (!pi*Sq/4.)*Cos(2.*!pi*fo*t + ph)+ph)
sigC = (1./2^(Cp-1.))* (Cos(2.*!pi*fo*t + ph) + 1)^Cp - 1.

```

*Example. 5 : Code snippet (IDL) for generating skewed and cusped sinewaves - cf. fig. 2.*

<sup>13</sup> Conway G.D. *Effects of reflectometer asymmetries on fluctuation measurements*, Plasma Phys. Control. Fusion **41** 65 (1999).

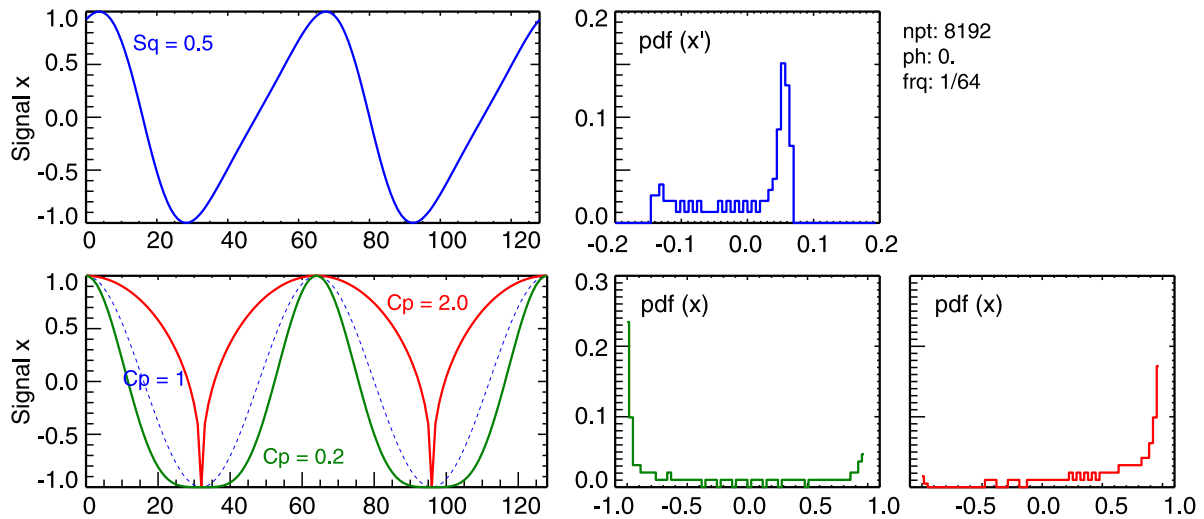


Figure. 2: Synthetic harmonic signal with (top) slewing plus pdf of its time derivative, and (bottom) cusping and corresponding amplitude pdf.

**2.3.3** These types of signal are close to the family of Stokes waves  $x(t) = \cos(\omega t + \epsilon \sin(2\omega t))$  which creates an FM or **intra-wave modulation**. The Stokes wave is also an approximate solution of the non-linear Duffing equation  $x'' + x + \epsilon x^3 = \gamma \cos(\omega t)$  which has an approximate solution of  $x(t) = \cos(\omega t) + \epsilon \cos(3\omega t) + \epsilon^2 \cos(6\omega t) + \dots$  With  $\gamma = 0.1$ ,  $\epsilon = -1$  and  $\omega = 1/25$  this will create strong symmetric peaking at the crests and troughs.

```

npt = 10000 & tinc = 200. & phs = 0.5 & omg = 2.*!pi*1.
cf = 1.0/(2.0**(c-1.))
do i = 1, npt
  t=real(i)/tinc
  x[i]=Cos(omg*t + phs)
  del = ((cos(omg*t+phs)+1.0)**c)*cf - 1.
  do il = 1, i, 3
    x[i] = ((cos(omg*t - (pi/2.)*s*del+phs)+1.0)**c)*cf - 1.
    del = x[i]
  end do
end do

```

Example. 6 : Code snippet (Fortran) for generating skewed and cusped sinewave.

## 2.4 Tailored broadband signals

Broadband fluctuation signals can be generated from a Fourier summation with a specified spectral shape or weighting function  $h(\omega)$ , such as a Gaussian (with half-width  $\omega_w$  and centre  $\omega_m$ ):

$$x(t) = \frac{1}{\sigma_x} \sum_{i=1}^N h(\omega_i) \cos(\omega_i t + \varphi_i) \quad (3)$$

$$h(\omega_i) = \exp\left(-(\omega - \omega_m + \delta_i)^2 / \omega_w^2\right) \quad (4)$$

However, care is required to avoid hidden periodicities. It is recommended to add randomized phase offsets  $\varphi$  (with a uniform distribution between  $\pm\pi$ ), and to use non-uniform spacing of the the frequency  $\omega$  intervals by adding a small random jitter  $\delta_i$  (of the order of 10%) to  $\omega$ .

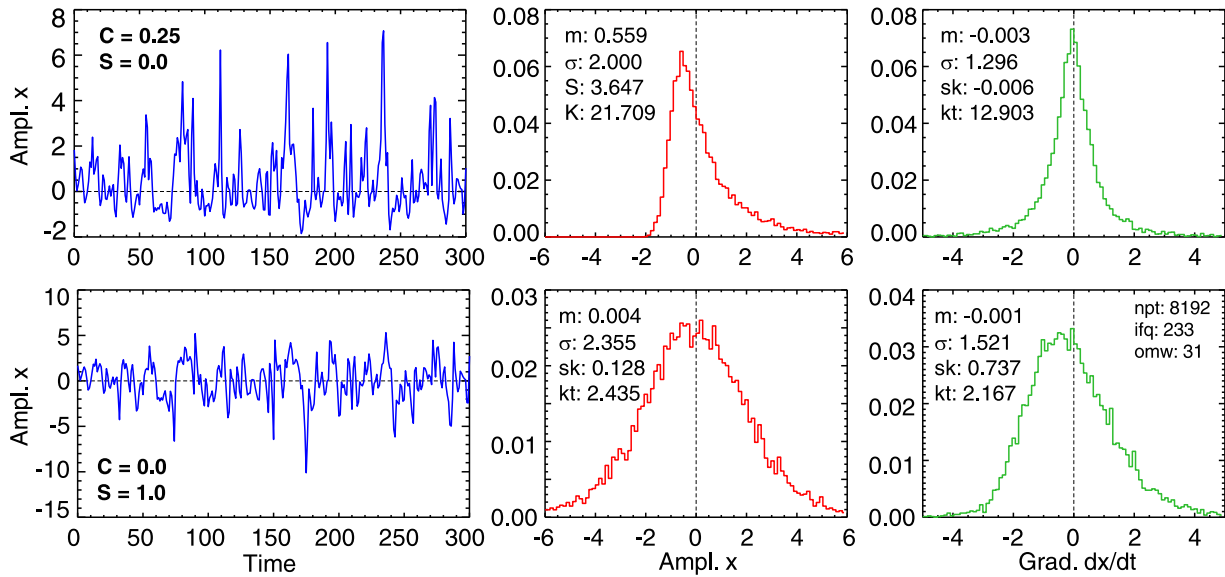


Figure 3: Examples of synthetic broadband time signals with skew and cusping plus pdfs of the signal  $x$  and it's time derivative  $x' = dx/dt$ .

Again, the two types of distortions (asymmetric amplitude and/or asymmetric gradient/time derivative pdf) can be applied to the broadband fluctuations. For example, the signal can be made either spiky or sawtoothed by multiplying  $x(t)$  with exponential functions such as:

$$x(t) = x(t) \times (\exp(Cx(t)) + 1) \quad (5)$$

$$x(t) = x(t) \times (\exp(Sx'(t)) + 1) \quad \text{where } x'(t) = dx(t)/dt \quad (6)$$

prior to normalization, where  $C$  and  $S$  are the cusping and skewing factors between  $\pm 1$ .

Figure 3 illustrates example signals and pdfs for extreme cusped and skewed forms. A signal with a preponderance of either negative or positive going spikes would result in an asymmetric amplitude distribution, while random sawteeth would result in an asymmetric gradient distribution. Both types of sequences are commonly observed in Langmuir probe ion-saturation currents.

```

seed = 104 & df = 0.991 & npt = 8192 & ifr = 233 & frw = 31 .
Cp = 0.25 & Sq = 1.0
t = Findgen(npt) & sig = Fltarr(npt)
frq = Fltarr(ifr) & h = Fltarr(ifr)
jit = 0.19*(RandomU(seed, ifr)-0.4999) ; Uniform random jitter
FOR i = 0, ifq-1 DO BEGIN
  frq[i] = (Float(i) + jit[i]) ; freq + small jitter
  h[i] = Exp(-frq[i]^2/frw^2) ; spectral weight fnct.
ENDFOR
ph = (RandomU(seed, ifq)-0.4999)*!pi*2. ; Uniform random phase
FOR i = 0, ifq-1 DO sig = sig + h[i]*Cos(2.*!pi*frq[i]*df*t +ph[i])
sig = sig*(Exp(1. + Cp*sig))/(2.^(Cp-1) ; Apply warp/cusp)
sig = sig/Stddev(sig) ; Normalize to sigma
sder = Deriv(sig) ; Take derivative
sig = sig*(1. + Exp(Sq*sder)) ; Apply skew

```

Example 7: Code snippet (IDL) for generating skewed and cusped broad-band signal - cf. fig. 3.



## 2.5 Non-stationary signals

Non-stationary means the signal pdf (ie. amplitude or frequency moments etc.) are evolving or changing in time. In the case of a non-stationary frequency a simple continuous chirping signal can be generated via  $x(t) = \exp(iat^2/2)$  where the frequency increases *linearly* with time. The parameter  $a$  sets the lowest starting frequency. See fig. 35 for an example in section 10.5.

```
npt = 2048 & a = 16.
time = Findgen(npt)/Float(npt)
data = Sin(!pi*(a*time)^2)
```

*Example. 8 : Code snippet (IDL) for generating a frequency chirp signal - cf. fig. 35.*

Likewise, non-stationary amplitude moments can be generated with additional terms to the above harmonic and broad-band fluctuation code snippets which, for example, evolve the signal amplitude either continuously, or intermittently.

## 2.6 Intermittent signals

Intermittent events are very commonly observed in the edge regions of confined fusion plasmas, and in particular are often related to density filamentary behaviour or to edge localized modes (ELMs). Typically these appear as semi-random spikes, but with a well defined temporal structure.

A synthetic event (ELM-like) signal may be generated in various ways. As an example of a construction procedure, first the basic event features are defined as follows:

### 1. Shape

The basic event shape is fixed, such as a spike with a fast linear rise to a maximum, followed by a slower exponential decay. The decay rate maybe:

- (a) Fixed or proportional to the rise rate.
- (b) Proportional to the event period.

### 2. Frequency / event period

- (a) Periodic. Here one can use a clipped sinewave (square-wave) as the event trigger.
- (b) Semi-periodic, ie. a broadened base frequency - use sinewave with a uniform  $\pm 1\pi$  random phase modulation, or frequency offset.
- (c) A normally distributed random frequency within a set range.

### 3. Peak amplitude

- (a) Follows a pre-defined pdf, such as the one-sided log-normal distribution, or a Poisson distribution would be a natural choice.
- (b) Uniformly distributed random amplitude within a set range.
- (c) Amplitude is proportional to the event period, eg. large and infrequent events or small and frequent.

### 4. Finally, add some random white noise background.

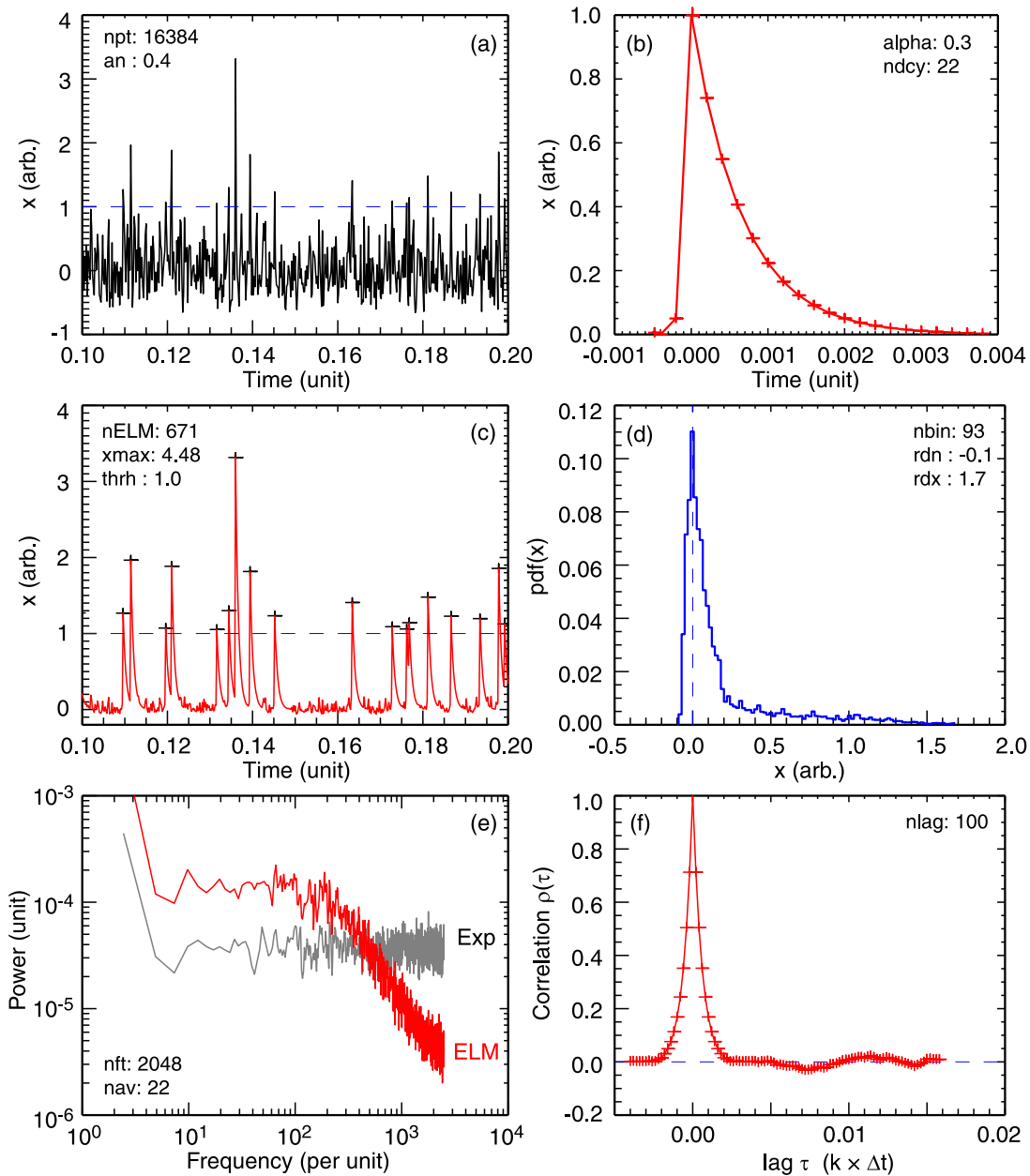


Figure 4: Synthetic intermittent ELM-like signal generated from (a) exponential random signal with (b) decaying spike shape, giving (c) ELM signal. (d - f) show pdf, spectra and autocorrelation of ELM signal.

Figure 4 shows an example of the kind of synthetic random, intermittent, explosive (ELM-like) positive going spike signal that can be created following the description above. In this example the starting point is a random Gaussian amplitude white noise signal which is converted to a positive going exponential signal - as illustrated in the code snippet below. A threshold detector selects the *singular* peak points of the large amplitude spikes. Around these peak points is wrapped a temporal shape with a fast rise and an exponential decay  $\exp(-at)$  as illustrated in fig. 4(b) resulting in the ELM-like signal in fig. 4(c) where additional white noise baseline has been added. The individual spikes retain the amplitude distribution of the original exponential signal resulting in the highly skewed pdf shown in the accompanying plot. As the spike

decay rate is fixed the full-width half-maximum (fwhm) of the spike scales with its amplitude. The corresponding spectra shows (red) a high-frequency  $1/f$ -like decay above an enhanced low-frequency 'bump'. In this example there is no specific coherent event/spike spectral peak since the source was a random noise. With an appropriate source choice, singular ELM-like frequencies or quasi-coherent like structures can like-wise be created.

Section 12.1.2 shows an example of how the above signal is used to demonstrate the operation of box-car (conditional averaging) of an intermittent signals.

```

npt = 16384 & seed = 104 & dt = 1./5000 & nft = 2048 .
an = 0.4 & thresh = 1.0 ; amp & threshold
time = Findgen(npt)*dt
freq = Findgen(nft)/Float(nft)/dt
; Make exponential random signal
noise = an * RandomN(seed, npt) ; normal random
sig0 = (Exp(noise) - 1.) ; -> exponential
; Make shape
p = Where(sig0 GT thresh, npt)
sig2 = sig0/10.
alpha = 0.3 ; decay rate
nr = 2 & nd = 20 ; up/dn widths
rise = [0.0, 0.05] ; linear rise &
decay = Exp(-1.0*Findgen(nd)*alpha) ; exp decay
shape = [rise, decay] ; add up&down
ns = nr + nd
; Scale spikes and (constructively) add to original exp. signal
nd = ns - nr - 1
FOR i = 0, ntp-1 DO BEGIN ; scale peaks
sig2[p[i]-nr:p[i]+nd] = shape*sig0[p[i]] > sig2[p[i]-nr:p[i]+nd]
ENDFOR
sig2 = (noise/5. > sig2) ; add 1/5 noise

```

*Example. 9 : Code snippet (IDL) for generating an intermittent 'ELM'-like' signal from white noise - cf. fig. 4.*

### 3 Probability functions

The starting point is the **probability distribution** of the signal amplitude. This is defined as the probability  $P$  of  $x(t)$  falling within a range  $x$  and  $x + \Delta x$  (full-width)

$$P(x) = \text{Prob}[x < x(t) \leq x + \Delta x] = \lim_{T \rightarrow \infty} \frac{T_x}{T} \approx p(x) \Delta x \quad (7)$$

where  $T_x = \sum \Delta t$  is the sum of all the periods where the above condition is met, and  $T$  is the total length of the time series.

The **Probability Density Function** (pdf)  $p(x)$  is the probability distribution  $P(x)$  divided by the amplitude interval or bin size  $\Delta x$ <sup>14 15</sup>

$$p(x) = \lim_{\Delta x \rightarrow 0} \frac{1}{\Delta x} \lim_{T \rightarrow \infty} \frac{T_x}{T}. \quad (8)$$

The pdf graphically displays how the signal amplitude is distributed. For discrete sampled data the  $p(x)$  function is estimated via histogramming, or binning the data sequence  $\{x_n\}$  into amplitude bins

$$\hat{p}_x = \frac{N_x}{NW} \quad (9)$$

where  $N_x$  is the number of data points that fall within the bin width  $x \pm W/2$  and  $N$  the total number of data points. The hat symbol is used to indicate that  $\hat{p}_x$  is an estimate of the real  $p(x)$ .

The pdf can be defined in terms of two, or more, independent variables to give a **joint pdf**

$$p(x, y) \rightarrow \hat{p}_{x,y} = \frac{N_{xy}}{NW_x W_y}. \quad (10)$$

If  $x$  and  $y$  are uncorrelated then  $\hat{p}_{x,y} = \hat{p}_x \hat{p}_y$  etc. In addition, a conditioned or **conditional pdf**  $p(x|y)$  can be defined where  $y$  is some condition to be met in the summation of  $T_x$  - see section 12.1.1.

Less familiar perhaps is the **Cumulative Distribution Function** (cdf) - also called the empirical probability distribution. This is useful for quantifying the distribution percentage-wise, as well as for identifying statistical outliers. The cdf is defined as

$$\text{cdf}(x) = p(u \leq x) = \sum_{u < x} \hat{p}(u) \quad (11)$$

The **Quantile plot**  $q(f)$  is an alternate representation of the cdf. Here  $f_n = (n - 0.5)/N$  is the decimal fraction of data below the  $f$  quantile. The approach is to sort the data into ascending values and then rank the values by assigning an  $f$ -value to each data point. The code example below indicates how this is computed in reality. Figure 5 shows example plots computed for a normal and for an exponential type signal.

<sup>14</sup> Bendat J.S. and Piersol A.G. *Random data: Analysis and measurement procedures*, 4th edn. Wiley, N.Y. (2010). First published in 1971.

<sup>15</sup> Bevington P.R. et al *Data reduction and error analysis for the physical sciences*, McGraw-Hill, 2nd ed. (1992).

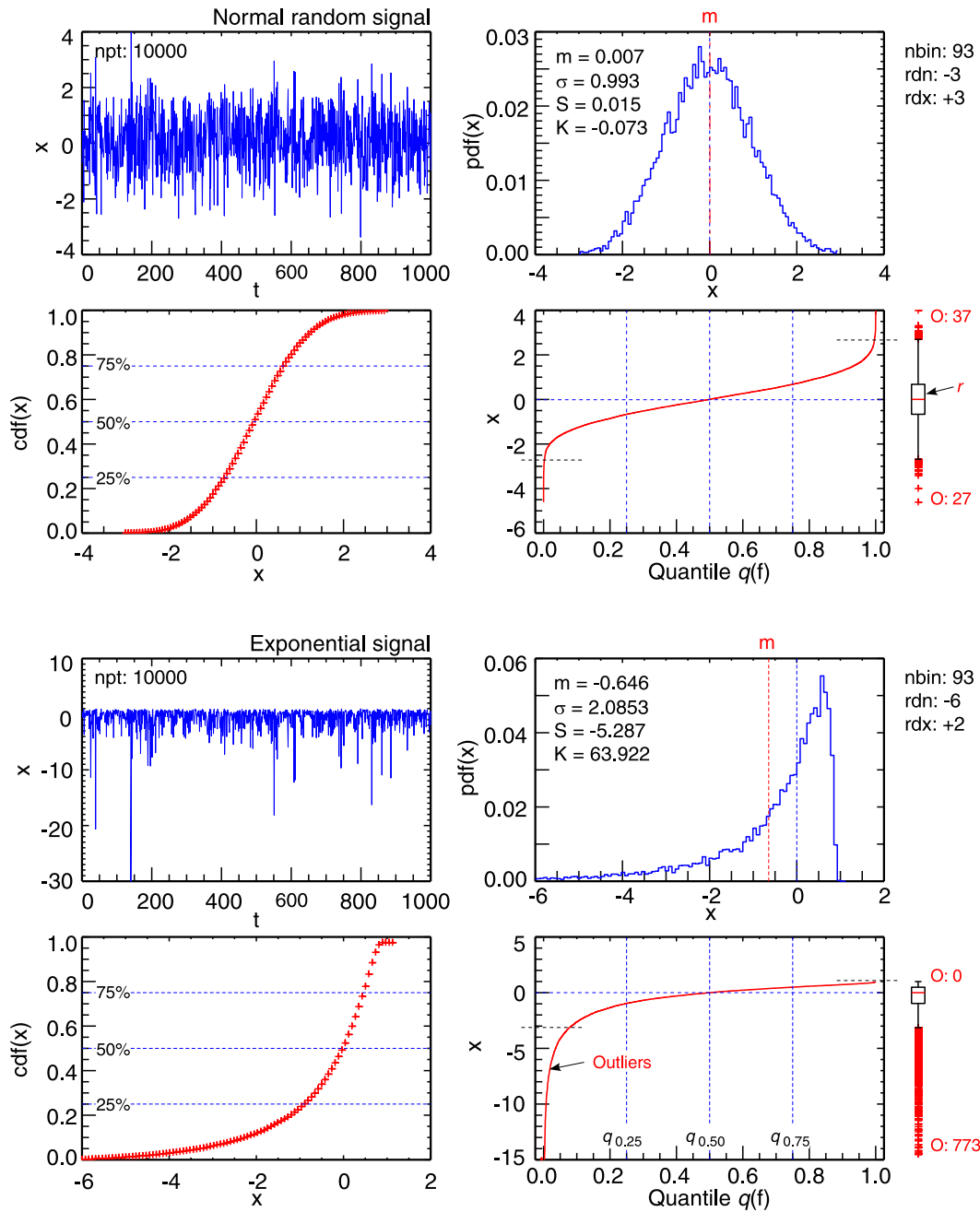


Figure 5: Time traces (first 1000 pts.), pdfs, cdfs, quantile  $q(f)$  plus box plots for 2 synthetic signals: (Top) a normal distributed signal, and (bottom) an exponential signal.

The 25% and 75% quantiles, or percentiles, are used for statistical tests. For example,  $r = q_{0.75} - q_{0.25}$  is the **inter-quantile range** which encompasses 50% of the data. Note this is not the standard deviation. The 50% quantile is the pdf **median**. The q-plot is equivalent to the inverse of the cdf, except using the full  $N$  points of the raw data rather than the cumulated pdf bins.

**Outliers** are defined as data points that lie outside of the  $q_{0.25} - 1.5r$  and  $q_{0.75} + 1.5r$  ranges. These are usually drawn using the so-called box-plot - as shown in fig. 5 - with the box showing the extent of the inter-quantile range, the red bar the median, the extender bars the  $\pm 1.5$  ranges above and below the 25% and 75% quantiles, and finally the red points are the outliers. In fig. 5

one sees that for the normal distribution  $\pm 0.37\%$  of the points lie outside the  $1.5r$  ‘buffer’. For the exponential distribution the pdf is highly skewed and the number of negative outliers rises to some 7.7%. How one interprets outliers depends on the measurement, they could be real data points, or equally measurement errors. Statistically, outliers are simply defined as being outside the expected normal distribution.

The **quantile-quantile plot** or q-q-plot is useful for comparing two signals, if  $x$  and  $y$  come from the same pdf then the q-q-plot is a straight line - even if the means and standard deviations of  $x$  and  $y$  are different.<sup>16</sup>

```

; Probability density function
  dx = (rdx-rdn)/Float(nbin)
  xbin = Findgen(nbin)*dx+rdn
  ybin = Histogram(data, Max=rdx, Min=rdn, Binsize=dx)/pts
; Cumulative distribution function
  xcdf = Findgen(nbin)*dx +rdn
  ycdf = Total(ybin, /Cumulative)
; Quantile
  xq = (Findgen(npt)-0.5) / pts
  yq = data[Sort(data)]

```

*Example. 10 : Code snippet (IDL) for calculating the cdf and quantile plots - cf. fig. 5.*

### 3.1 Moments and cumulants

The (sample) moments are quantities related to the shape of the pdf and are defined in terms of the expectation operator  $E[x]$  or expected value, ie. the ensemble average.  $E[x]$  is the probability weighted average of all  $x$  values, eg. for the raw mean or first moment of a time series:

$$E[x] = \sum_{n=1}^{\infty} x_n p_n \quad \text{where } p_n \text{ are the probabilities for each data point.} \quad (12)$$

For the usual case of uniform measurement likelihoods  $p_n = 1$ , the  $p$ -th moment (about zero) of a real (discrete) valued distribution is defined as

$$\mu_p = E[x^p] = \langle x^p \rangle \quad (13)$$

Central moments are defined about the mean  $m$  rather than zero

$$\mu_p = \frac{1}{N} \sum_{n=1}^N (x_n - m)^p \quad (14)$$

Standardized moments are the  $p$ -th central moments normalized to  $\sigma^p$ , the standard deviation to the power of  $p$ .

**1st moment: Mean (Raw) :** Middle point

$$m = \frac{1}{N} \sum_{n=1}^N x_n \quad (15)$$

<sup>16</sup> Meko D.M. *GEOS585A, Applied time series analysis : Course notes*, Univ. of Arizona (2019). Download from <http://www.ltr.arizona.edu/~dmeko/>

**2nd moment: Variance** (Central): Standard deviation squared, or ‘width’ of the pdf

$$\text{var}[x] = \sigma_x^2 = \frac{1}{N} \sum_{n=1}^N (x_n - m)^2 \quad (16)$$

Note, a so-called ‘standardized’ distribution would have a mean = 0, and variance = 1.

For a bivariate distribution the **covariance** is

$$\text{cov}[x, y] = \sigma_{xy}^2 = \frac{1}{N} \sum_{n=1}^N (x_n - m_x)(y_n - m_y) \quad (17)$$

**3rd moment: Skewness** (Standardized) : Asymmetry of the pdf about the mean

$$S = \frac{1}{N} \sum_{n=1}^N (x_n - m)^3 / \sigma_x^3 \quad (18)$$

**4th moment: Kurtosis** (Standardized) : Flatness or tailness of the pdf relative to the centre

$$K = \frac{1}{N} \sum_{n=1}^N (x_n - m)^4 / \sigma_x^4 \quad (19)$$

A normal or Gaussian distribution has a  $K = 3$ . Thus the excess Kurtosis =  $K - 3$  is defined as flatness relative to the normal distribution.

Higher moments are also defined, **5th moment: hyperskewness**, **6th moment: hyperflatness** but are not often used. The higher the moment the longer the time series (ie.  $N$ ) required for statistical stability. For **sample** moments the normalization is usually replaced with  $1/(N - 1)$ , where  $\nu = N - 1$  is the degrees of freedom (dof), to produce a better unbiased estimate.

The **median** (also the  $\nu$  symbol) is the value which separates a sample distribution into two equal halves, ie. equal number of points above and below  $\nu$ . The sample **mode** is the most frequent value in the data set, ie. the  $x$  value of the  $p(x)$  peak.

The **standard error** (SE), ie. the standard deviation  $\sigma$ , of some parameter or estimator  $\hat{A}$  is an important concept in data analysis and can be expressed in terms of the expected value:

$$\sigma^2 = E[(\hat{A} - E[\hat{A}])^2]. \quad (20)$$

The **normalized standard error**  $\epsilon$  given by:  $\epsilon[\hat{A}] = \sigma[\hat{A}]/\hat{A}$  for  $\hat{A} \neq 0$ .

Cumulants  $\kappa_p$  are alternative expressions of the moments  $\mu_p$ . The 1st cumulant  $\kappa_1$  is equal to the mean, the 2nd  $\kappa_2$  is also equal to the variance (2nd central moment), likewise the 3rd cumulant  $\kappa_3$  is also equal to the 3rd central moment. However, the 4th and higher order cumulants are not equal to the respective central moments.<sup>17</sup>

<sup>17</sup> Wikipedia - <https://en.wikipedia.org/wiki/Cumulant> (March 2023).

### 3.2 Errors in moment estimations

The pdf moment estimations are themselves subject to statistical uncertainties. These can be calculated as the standard error (SE) of the moment (usually via their gradients.<sup>15</sup>). As an example the uncertainty in the mean is  $\sigma[m] = \sigma/\sqrt{N}$ .

Another approach<sup>18</sup> is to divide the pdf range  $p(x)$  into  $i = 0, \dots, K$  intervals of width  $x \pm \Delta x/2$ , each with an average pdf value  $\bar{p}(x_i) = N_{xi}/N\Delta x$  centered on  $x_i$  with  $N_{xi}$  points. The corresponding moments of order  $p$  are then

$$\mu_p = \sum_{i=1}^K \bar{p}(x_i) (\bar{x}_i - m)^p \Delta x$$

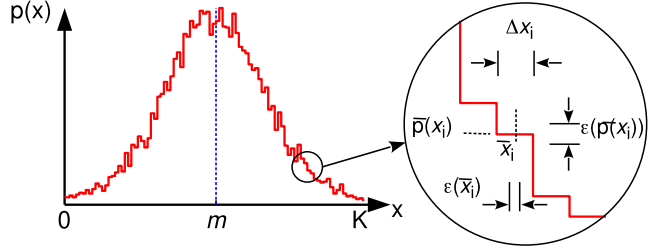


Figure. 6: Errors in pdf estimation. (21)

where  $m$  is the mean of  $x_i$  and  $\bar{x}_i$  is the so-called interval class mark

$$\bar{x}_i = \left( \frac{1}{N_{xi}} \sum_{j=1}^{N_{xi}} (x_j - m)^p \right)^{1/p} + m$$

where the sum is over the  $i$ -th interval. There are two possible error sources: the error in estimating  $p$  at each interval  $\epsilon[\bar{p}(x)]$ ; and the error in the interval class mark given by

$$\epsilon[\bar{x}_i^p] = \text{Max}(|x_{Mi}^p - \bar{x}_i^p|, |\bar{x}_i^p - x_{mi}^p|)$$

where  $x_{Mi}$  and  $x_{mi}$  are the upper and lower bounds of the  $i$ -th  $x$  interval. The total error of  $\mu_p$  is then

$$\epsilon[\mu_p] = \frac{1}{N\sqrt{K}} \left[ \sum_{i=1}^K \sqrt{N_{xi}} |(\bar{x}_i - m)^p| + \sum_{i=1}^K N_{xi} \epsilon[\bar{x}_i^p] \right].$$

### 3.3 Temporal asymmetry

Distributions and moments can also be calculated for the temporal derivative of the signal. An example is the **temporal asymmetry**, ie. skewness of the time derivative  $x' = \partial x/\partial t$ , defined as

$$A(x) = \frac{\langle (x' - \langle x' \rangle)^3 \rangle}{\sigma^3(x')}$$

where angular brackets mean temporal average.<sup>18</sup> An  $A = 0$  corresponds to a signal with a symmetric distribution for its time derivative.  $A > 0$  implies that the rise time of  $x(t)$  is on average shorter than the decay time, and  $A < 0$  the opposite. This parameter is useful for studying relaxation events. As an example, a sawtooth waveform with a rise time  $t_r$  and a decay

<sup>18</sup> Sanchez E. et al *Statistical characterization of fluctuation wave forms in the boundary region of fusion and nonfusion plasmas*, Phys. Plasmas **7** (5) 1408 (2000).



time  $t_d$  would have a skewness  $S = 0$  but an asymmetry  $A = (t_d - t_r)/\sqrt{t_r t_d}$ , which nicely quantifies the time asymmetry of the sawtooth. Likewise, the temporal Kurtosis can also provide useful information on the shape of the sawtooth, or other perturbation shapes etc.

### 3.4 Types of distributions

There are a range of common (continuous) distribution functions that occur in the context of signal analysis and in the fitting of discrete signal histograms and spectral peaks. Some of the more important distributions may include the following (see fig. 7 for comparisons):

#### 3.4.1 Poisson distribution

$$p(x) = \frac{\mu^x}{x!} \exp(-\mu) \quad (26)$$

a one-sided distribution where  $\mu$  is the mean and the variance. Here, the factorial is computed using the gamma function with  $x! = \Gamma(x + 1)$  for positive non-integers.

#### 3.4.2 Gaussian (or normal) distribution

$$p(x) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{(x-m)^2}{2\sigma^2}\right) \quad (27)$$

where  $m$  is the mean and  $\sigma$  the standard deviation ( $1/e$  distribution width).

#### 3.4.3 Lorentzian (or Cauchy) distribution

$$p(x) = \frac{1}{\pi} \frac{\gamma_L/2}{(x-m)^2 + (\gamma_L/2)^2} \quad (28)$$

where  $\gamma_L$  is the full-width at half maximum fwhm (or 'half-height width').

#### 3.4.4 Log-normal distribution

$$p(x) = \frac{1}{\sigma\sqrt{2\pi}} \frac{1}{x} \exp\left(-\frac{(\ln x)^2}{2\sigma^2}\right) \quad \text{for } x \geq 0. \quad (29)$$

a one-sided distribution (for  $x \geq 0$ ) often observed in the distribution of radar returns.

#### 3.4.5 Rayleigh (Exponential) distribution

$$p(x) = \frac{x}{\sigma^2} \exp\left(-\frac{x^2}{2\sigma^2}\right) \quad \text{for } x \geq 0. \quad (30)$$

where  $\sigma$  is a scale parameter. This distribution is often observed in the vector magnitude of signals which have normally distributed amplitudes and directions.

#### 3.4.6 Gamma distribution

$$p(x) = (\beta^\alpha/\Gamma(\alpha)) x^{\alpha-1} \exp(-\beta x) \quad \text{for } x \geq 0, \text{ and } \alpha, \beta > 0. \quad (31)$$

where  $\Gamma(\alpha) = (\alpha - 1)!$  is the gamma function.

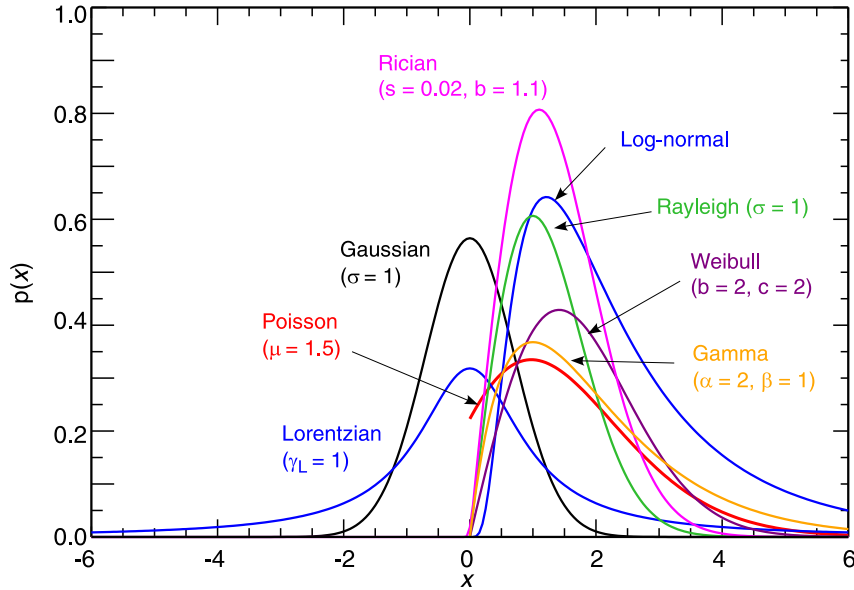


Figure. 7: Comparison of various distribution functions.

### 3.4.7 Weibull distribution

$$p(x) = (c/b) (x/b)^{(c-1)} \exp(-(x/b)^c) \quad (32)$$

where  $b$  is a scale parameter and  $c$  is a shape parameter. The Weibull joins the exponential ( $c = 1$ ) and the Rayleigh ( $c = 2$  and  $b = \sqrt{2}\sigma$ ) distributions.

### 3.4.8 Rician (or Rice) distribution (non-zero mean)

$$p(x) = \frac{s}{b} \exp\left(\frac{-x^2}{2s^2b^2}\right) + \left(\frac{\sqrt{1-s^2}}{2b^2}\right) x \exp\left(\frac{-x^2}{2b^2}\right) \left\{ 1 + \operatorname{erf}\left[\frac{x}{\sqrt{2}b} \left(\frac{\sqrt{1-s^2}}{s}\right)\right] \right\} \quad (33)$$

is a bivariate normal distribution with  $s$  and  $b$  defining different scale/shape parameters and erf the error function.

### 3.4.9 Voigt distribution

A distribution often used for fitting spectral peaks is the Voigt function. This is a convolution of Gaussian  $f_G(x)$  and Lorentzian  $f_L(x)$  functions

$$f_V(x) = \int_{-\infty}^{\infty} f_G(x - m; \sigma) f_L(x - m; \gamma_L) dx = \frac{\operatorname{Re} w(z)}{\sigma\sqrt{2\pi}} \quad (34)$$

where  $w(z) = \exp(-z^2) \operatorname{erfc}(-iz)$  is the Feddeeva fnct. with  $z = (x + i\gamma_L)/(\sigma\sqrt{2})$  and erfc the complementary complex error function.  $f_V(x)$  is often approximated by the **pseudo Voigt**

$$f_{pV}(x_1) = (1 - \eta) f_G(x_1) + \eta f_L(x_1) \quad (35)$$

where  $x_1 = x - m$  (zero centered) and  $0 < \eta < 1$  is a mixing fraction.

One approximation gives: <sup>19 20</sup>

$$f_{pV}(x_1) = (1 - \eta) \left( \frac{2\sqrt{\ln(2)}}{\Gamma\sqrt{\pi}} \right) \exp\left( \frac{-4 \ln(2) x_1^2}{\Gamma^2} \right) + \eta \left( \frac{2\Gamma}{\pi} \right) \left[ 1 + \frac{4x_1^2}{\Gamma^2} \right]^{-1} \quad (36)$$

which can be fitted directly with  $\eta$  and the Voigt full-width half-maxima (fwhm)  $\Gamma$ , with or without amplitude scale factors, or they can be approximated in terms of respective Gaussian and Lorentzian fwhm:

$$\eta = 1.36603 (\Gamma_L/\Gamma) - 0.47719 (\Gamma_L/\Gamma)^2 + 0.11116 (\Gamma_L/\Gamma)^3. \quad (37)$$

$$\Gamma = (\Gamma_G^5 + 2.69269 \Gamma_G^4 \Gamma_L + 2.42843 \Gamma_G^3 \Gamma_L^2 + 4.47163 \Gamma_G^2 \Gamma_L^3 + 0.07842 \Gamma_G \Gamma_L^4 + \Gamma_L^5)^{1/5} \quad (38)$$

where  $\Gamma_G = 2\sqrt{\ln(2)} \sigma$  and  $\Gamma_L = 2\gamma_L$ .

The Voigt fwhm can also be approximated by

$$\Gamma_V \approx 0.5346 \Gamma_L + \sqrt{0.2166 \Gamma_L^2 + \Gamma_G^2}$$

There are also other forms in the literature <sup>21</sup> where eqn. (35) is expressed in terms of

$$f_G(x) = (1/\sqrt{\pi} \sigma) \exp(-x^2/\sigma^2)$$

$$f_L(x) = (1/\gamma_L \pi) [1 + x^2/\gamma_L^2]^{-1}$$

with  $\Gamma$  and  $\eta$  as given above.

Figure 8 shows an example pseudo Voigt together with the composite Gauss and Lorentzian functions for  $\sigma = \gamma_L = 1$ .

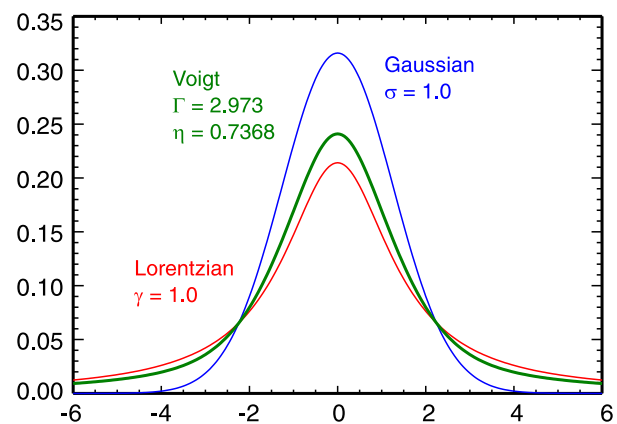


Figure. 8: Voigt plus Gauss and Lorentzian.

<sup>19</sup> Thompson P. et al *Rietveld refinement of Debye-Scherrer synchrotron X-ray data from Al<sub>2</sub>O<sub>3</sub>*, J. Appl. Cryst. **20** 79 (1987).

<sup>20</sup> Di Rocco, H.O. et al *The Voigt profile as a sum of a Gaussian and a Lorentzian functions, when the weight coefficient depends only on the widths ratio*, Phys. Polonica A **122** 666 (2012)

<sup>21</sup> Ida T. et al *Extended pseudo-Voigt function for approximating the Voigt profile*, J. Appl. Cryst. **33** 1311 (2000).

## 4 Stationarity / Randomness tests

Many analysis techniques (eg. FFTs) either presume or require the data to be stationary. Thus testing for stationarity is important. The standard definition of a stationary signal  $x(t)$  is

$$\begin{aligned}
 E[|x_n^2|] &< \infty \\
 E[x_n] &= m \\
 \text{cov}[x_{t_1}, x_{t_2}] &= \text{cov}[x_{t_1+\tau}, x_{t_2+\tau}] = \text{cov}(t_1 - t_2)
 \end{aligned}
 \tag{39}$$

for all  $t$ , with  $E[\cdot]$  the expected value, eg. ensemble average, and  $\text{cov}[\cdot]$  the covariance.<sup>22</sup> **Strict stationary** means that the pdf( $x$ ) is time invariant. However, **weak stationarity** means that only the moments of the pdf (ie. the mean and standard deviation etc.) are time invariant, while the pdf itself may not be. The analysis of non-stationary signals is addressed in section 10.

### 4.1 Runs test

One of the better known stationarity tests is the (Wald-Wolfowitz) **Ordinary Runs test** of the mean square deviation about the median.<sup>23</sup> The procedure is as follows:

- (1) Divide time series  $x_n$  into  $M$  equally long (independent and similarly distributed) segments.
- (2) Create a new time series of segment mean squares  $y = \{x_1^2, x_2^2, \dots, x_M^2\}$ .
- (3) For each  $y_i$  assign a “+” if  $y_i > \nu_y$  and a “-” if  $y_i < \nu_y$  where  $\nu_y$  is the **median** of the  $y$  mean squares sequence.

A sequence of “+”s or “-”s is called a run. The relative balance of the total number of alternating positive and negative runs is an indicator of the stationarity of the data.

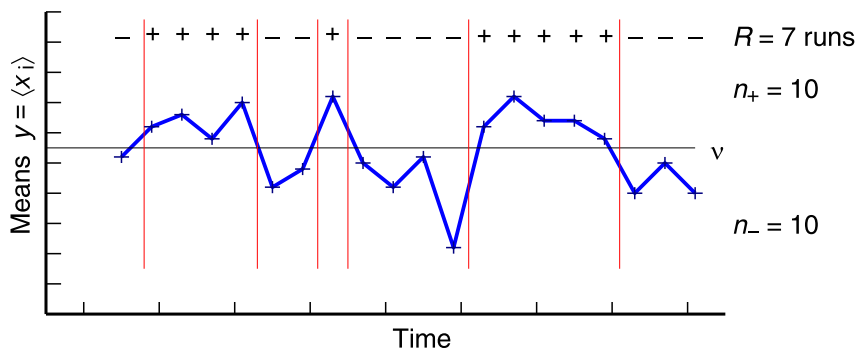


Figure. 9: Example of performing a mean squares runs test with  $n = 20$  sample means.

In the example of fig. 9 there are  $R = R_+ + R_- = (3 + 4) = 7$  runs with  $n = n_+ + n_- = 20$  values. For a null hypothesis of randomness every arrangement of “+” and “-” signs should be equally probable with no clear trends.

<sup>22</sup> Huang N.E. *The empirical mode decomposition and the Hilbert spectrum for nonlinear and non-stationary time series analysis*, Proc. R. Soc. Lond. A **454** 903 (1998)

<sup>23</sup> Bendat & Piersol *Random data: Analysis and measurement procedures*, Wiley (1971)

That is to say, a large discrepancy between  $R_+$  and  $R_-$  would be an indicator of non-stationary. Put more precisely, for a large  $n_+$  and  $n_-$  (ie.  $> 10$ ) the runs should have approximately a normal conditional distribution with mean and variance

$$m_R = \frac{2n_+n_-}{n} + 1 \quad \text{and} \quad \sigma_R^2 = \frac{2n_+n_-(2n_+n_- - n)}{n^2(n-1)}. \quad (40)$$

Taking the probability of false alarm  $P_{FA}$  as  $\alpha$ , then the critical region is

$$\frac{|R - m_R|}{\sigma_R} \geq Q(\alpha/2) \quad \text{where} \quad Q(\alpha/2) = \frac{1}{\sqrt{2\pi}} \int_{\alpha/2}^{\infty} e^{-t^2/2} dt. \quad (41)$$

For a 5% significance level, ie.  $\alpha = 0.05$ , a  $z$ -test statistic with an absolute value greater than 1.96 would indicate non-randomness. Thus, for stationarity expect  $R_{n,1-\alpha/2} < R \leq R_{n,\alpha/2}$ .

For smallish sets of runs it is usual to use a half point correction  $R \rightarrow R \pm 0.5$  for ‘continuity’. Add 0.5 to  $R$  if it is smaller than  $m_R$  and subtract 0.5 if it is greater. There are a large number of more recent alternative runs tests in the literature.<sup>24 25</sup>

Other alternatives for testing signal stationarity include modelling each set as an autoregressive (AR) process (see section 7.1 and 10.6) with time varying coefficients - the so-called adaptive AR technique - and evaluating their temporal variation.<sup>26</sup>

---

<sup>24</sup> Modarres R. et al *A modified runs test for symmetry*, Statist. Prob. Lett. **31** (2) 107 (1996).

<sup>25</sup> O’Brien P.C. et al *A runs test based on run lengths*, Biometrics **41** (1) 237 (1985).

<sup>26</sup> Ulrych T.J. & Ooe M. *Chapter 3: Autoregressive and mixed autoregressive-moving average models and spectra* in Nonlinear methods of spectral analysis, 2nd edn. Springer (1983).

## 5 Correlation functions

### 5.1 Lagged Auto & Cross-association

The most basic measure of similarity between two time sequences is the so-called ‘matching state’.<sup>27</sup> At any position within the paired sequences, their respective values are either identical (match = 1) or different (mismatch = 0). The degree of association, or **index of similarity** is then defined as  $a_L = M/N$ , the ratio of the total number of matched pairs  $M$  to the total number of compared pairs  $N$ . As with correlations the sequences can be shifted (lagged  $L$ ) relative to each other to reveal repetitive patterns or time delays in the sequences. Mathematically  $a_L$  is expressed as:

$$a_L = \frac{1}{N-L} \sum_{n=0}^{N-L-1} 1 | (x_n = x_{n+L}) \tag{42}$$

which can range from 0 (no association) to 1 (perfect similarity).

Example: lagged auto-association of a time series  $\{x_n\}$  with an integer sequence:

$$\begin{aligned} x &= \{1, 1, 0, 1, 1, 2, 1, 0, 0, 1, 1, 2, 2\} \\ L = 1 & \{, 1, 1, 0, 1, 1, 2, 1, 0, 0, 1, 1, 2, 2\} \\ \text{match} & \{, 1, 0, 0, 1, 0, 0, 0, 0, 0, 1, 0, 1\} \rightarrow a_1 = 4/12 = 0.333 \end{aligned}$$

### 5.2 Lagged Auto & Cross-correlation

The correlation coefficient is a measure of the degree of *linear* association between two variables based on a sufficiently large data set. For time series this usually means the time lagged cross-correlation. The cross correlation function  $R_{xy}(\tau)$  as a function of time delay  $\tau$  is defined as:

$$R_{xy}(\tau) = \lim_{T \rightarrow \infty} \frac{1}{T} \int_0^{\infty} x(t)^* y(t + \tau) dt \tag{43}$$

where  $x^*$  means the complex conjugate of  $x$ .

For discrete sampled series  $\{x_n\}$  and  $\{y_n\}$  at time index  $n = t/\Delta t$  for  $(n = 0 \dots N - 1)$  with delay (lag) index  $k = \tau/\Delta t$  with  $(k = 0 \dots m)$  and  $m \ll N$  the correlation is:

$$\text{cor}_{(xy)}(k, n) = E[x_{n+k} y_n^*] \tag{44}$$

The **cross-covariance** is similar to the correlation except that the signals are mean centred:

$$\text{cov}_{(xy)}(k, n) = E[(x_{n+k} - m) (y_n - m)^*] \quad (\text{mean subtracted}). \tag{45}$$

The **sample correlation** is thus given by

$$\hat{R}_{xy}(k) = \frac{1}{N-k} \sum_{n=1}^{N-k} (x_n - \bar{x})^* (y_{n+k} - \bar{y}) \tag{46}$$

---

<sup>27</sup> GITTA - Geographical Information Technology Training Alliance *Time as a sequence of events (with regular intervals)*, <http://gitta.info> (2014).

Note that the zero-lag auto-correlation  $\hat{R}_{xx}(0) = \sigma_x^2$  and  $\hat{R}_{xy}(0) = \text{cov}[x, y] = \sigma_{xy}$ .

The general properties of the correlation are:

- (1) When the mean  $m = 0$  then covariance equals the correlation,  $\text{cov} = \text{cor}$ .
- (2)  $R(-k) = R^*(k)$ , ie. the auto-correlation is Hermitian.
- (3) When the pdf moments  $\mu_x(n) = \mu = \text{constant}$  then the data and the correlation are stationary.

The correlation function can also be extended to multi-dimensional parameters, e.g.:

$$\hat{R}_{xy}(p, q, r) = \frac{1}{N} \sum_{i, j, k} (x_{i, j, k} - \bar{x})^* (y_{i+p, j+q, k+r} - \bar{y}). \quad (47)$$

Normalizing the correlation function  $R_{xy}$  gives the **correlation coefficient**  $-1 \leq \rho(\tau) \leq 1$ , aka the **sample Pearson correlation**  $\hat{r}_k$  for discrete data

$$\rho(\tau) = \frac{R_{xy}(\tau)}{(R_{xx}(0) R_{yy}(0))^{1/2}} \quad (48)$$

$$\hat{r}_k = \frac{1}{N - k} \sum_{n=0}^{N-k-1} \frac{(x_n - \bar{x})(y_{n+k} - \bar{y})}{\sigma_x \sigma_y} \quad (49)$$

Note the  $1/(N - k)$  normalization in eqn. 49 compared to  $1/N$  in eqn. 47. For small lag values  $k \ll N$  relative to the sample length  $N$ , the difference is not too important. But, as seen in fig. 10 (and corresponding code snippet) for the autocorrelation of a sinewave, the effect is significant for  $k \lesssim N$  when normalizing incorrectly to  $1/N$  (black) and correctly to  $1/(N - k)$  (red),

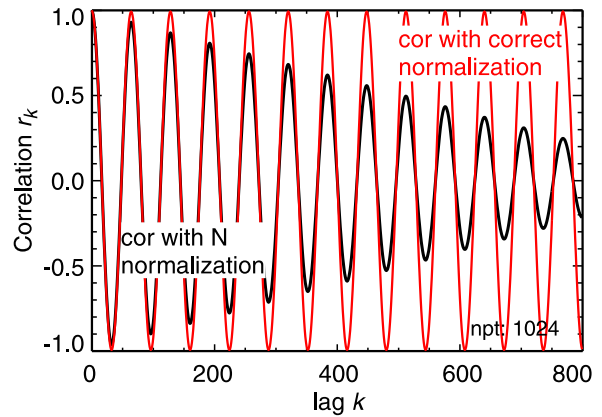


Figure. 10: Example of sinewave correlation with  $1/N$  and  $1/(N - k)$  normalization.

As discussed in section 6.3 the coherence and cross-phase spectra are the Fourier equivalents in frequency space and contain the same information as the time-delayed cross-correlation.

```

data = Sin(2.*!pi*Findgen(npt)*16./npt) & lag = Indgen(nlag) .
M = Abs(lag)
; Incorrect for nlag =< npt, but okay for nlag << npt
FOR k = 0L, nlag-1 DO $
    cor[k] = Total(data[0:npt-1-M[k]] * data[M[k]:*]) / Total(data^2)
; Correct always
FOR k = 0L, nlag-1 DO $
    cor[k] = Total(data[0:npt-1-M[k]] * data[M[k]:*])/$
    Sqrt(Total(data[M[k]:*]^2)*Total(data[0:npt-1-M[k]]^2))
    
```

Example. 11 : Code snippet (IDL) for correct and incorrect correlation normalization - cf. fig. 10.

### 5.3 Confidence limits & confidence intervals

The correlation estimators have uncertainties. These uncertainties are reflected in the minimum **significance level (SL)** above which the data is meaningful, and the **confidence interval (CI)** or band, ie. the error bar about the correlation point. These are specified in terms of probability confidence level  $p$  and the significance  $\alpha = 1 - p$ . A confidence of  $p = 95\%$  equates to an  $\alpha = 0.05$

#### 5.3.1 Standard error SE (normal)

Figure 11 shows the standard normal (Gaussian) distribution (with mean  $m = 0$  and variance  $\sigma^2 = 1$ ) with various probability regions about the mean:

$p$	$z_{\alpha/2} \sigma$
68.3% lies within	$\pm \sigma$
95.0%	$\pm 1.96 \sigma$
95.5%	$\pm 2 \sigma$
99.0%	$\pm 2.53 \sigma$
99.7%	$\pm 3 \sigma$

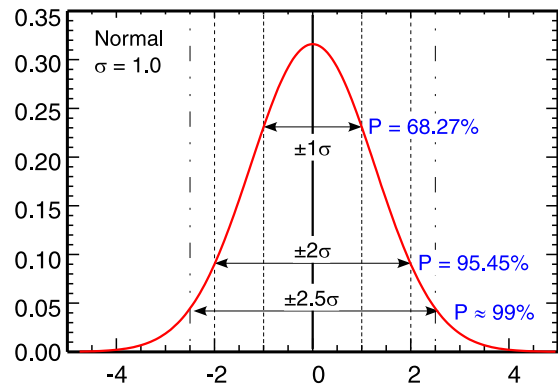


Figure. 11: Standard normal distribution.

For the correlation the SL and CI depend primarily on the sample size  $N$ , or more precisely on the degrees of freedom (dof). If at least one of the signals is white noise, or  $N$  is large, then the **error distribution** is approximately normal with zero mean and a standard deviation of  $\sigma = 1/\sqrt{N}$ . The standard error (SE) equals the  $\sigma$ . The CI is then  $\pm SE$  times the  $z_{\alpha/2}$  ( $z$ -score) for the chosen confidence level  $\alpha$ .

For example, a 99% confidence level ( $\alpha = 0.01$ ) gives  $\pm 2.53/\sqrt{N}$ , and 95% level a  $\pm 1.96/\sqrt{N}$  error bar. Wikipedia<sup>28</sup> gives an alternate standard error  $\sigma_r = \sqrt{(1 - \hat{r}_k^2)/(N - 2)}$  (derived assuming a t-distribution).

#### 5.3.2 Fisher transform (non-normal)

However, not all signals are normal distributed but may be bounded by zero, or clipped by some threshold or limit level thus resulting in a non-normal error distribution. In these cases the Fisher transform  $F(\hat{r}_k) = \tanh^{-1}(r_k)$  is recommended,<sup>28</sup> since  $F$  is approximately normal distributed with a mean  $m = \tanh^{-1}(\hat{r}_k)$  and standard error  $SE = \sigma = 1/\sqrt{N - 3}$ , ie. a dof  $\nu = N - 3$ .

Thus the  $\hat{r}_k$  confidence interval for a  $(1 - \alpha)\%$  statistic becomes

$$\hat{r}_k \in \left[ \tanh \left( \tanh^{-1}(\hat{r}_k) \pm \frac{z_{\alpha/2}}{\sqrt{N - 3}} \right) \right] \tag{50}$$

where  $z_{\alpha/2} = 1.96$  for a 95% probability level and  $z_{\alpha/2} = 2.53$  for a 99% probability level.

<sup>28</sup> [https://en.wikipedia.org/wiki/Pearson\\_correlation\\_coefficient](https://en.wikipedia.org/wiki/Pearson_correlation_coefficient) (Feb. 2023).



### 5.3.3 Large lag standard error

At large lag  $k$  values the correlation coefficient may become self-correlated, ie. enhancing the correlation. Thus, it is usual to adjust the correlation variance  $\sigma^2$  as follows: <sup>29</sup>

$$\text{var}[\hat{r}_k] = \frac{1}{N} \left( 1 + 2 \sum_{i=1}^K \hat{r}_i^2 \right) \tag{51}$$

where  $K < k$ . The resulting standard error of the correlation  $SE = z_{\alpha/2} \sqrt{\text{var}[r_k]}$  broadens with increasing  $\pm k$  lag - see fig. 12 below for an example of a narrow-band signal. Note that this correction is not appropriate for strongly harmonic signals.

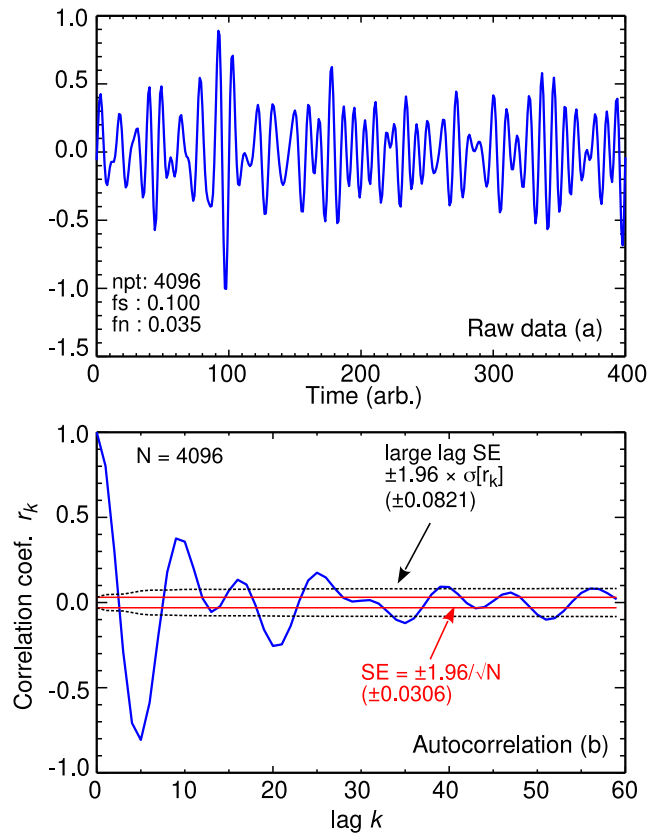


Figure. 12: Example of correlation SE and large-lag SE for a narrow-band noise signal.

### 5.4 Scaled correlations

Usually the correlation would be calculated using a single full length block of  $N$  samples. However, if the signal is broken into several  $M = \text{round}(T/s)$  shorter segments each of length  $s$  seconds, which are correlated and then ensemble averaged

$$\hat{r}_M = \frac{1}{M} \sum_{m=1}^M \hat{r}_m. \tag{52}$$

it is claimed that by adjusting  $s$ , the resulting  $\hat{r}_M$  correlation can be made less sensitive to the lower frequency signal content, leaving the correlation of faster varying components. <sup>28</sup>

<sup>29</sup> Meko D.M. *GEOS585A, Applied time series analysis : Course notes*, U. of Arizona (2019).

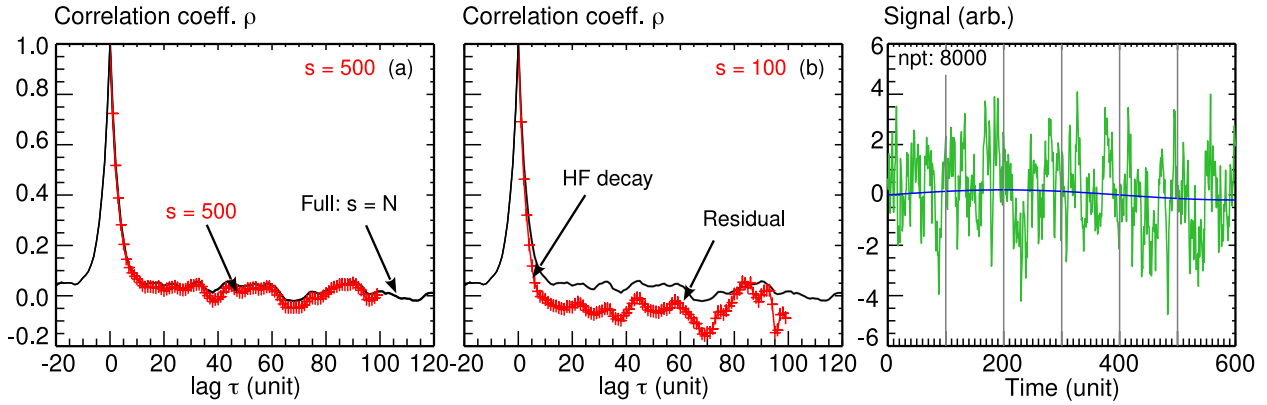


Figure. 13: Example of scaled correlations for a low-frequency dominated pink noise with block lengths of (a)  $s = 500$  and (b)  $s = 100$  points (red crosses) with  $N = 8000$  points compared with  $s = N$  (black lines).

As an example, figure 13 compares time lagged autocorrelations of a low frequency dominated (pink) noise signal over a full sample length  $s = N = 8000$  points (black line) with ensemble averaged shortened segments (red crosses) of (a)  $s = 500$  and (b)  $s = 100$  points. The correlation decays initially with a rate set by the highest frequency component, followed by a slower residual ripple due to the lower frequency components. For block lengths  $s > 500$  there is little effect of segmenting, but for  $s = 100$  the low-frequency structure is significantly reduced and the initial correlation decay is a little faster and the residual somewhat enhanced.

### 5.5 Weighted correlations

There are occasions where it maybe necessary or desirable to apply a weight factor  $w$  to the data before correlating, for example in the Blackman-Tukey method of calculating the spectral density from the Fourier transform of the correlation  $R$  function which has been windowed - see section 6.

The weighted covariance is given by

$$\text{cov}(x, y; w) = \frac{\sum_n w_n (x_n - m(x; w)) (y_n - m(y; w))}{\sum_n w_n} \tag{53}$$

where the weighted mean is

$$m(x; w) = \frac{\sum_n w_n x_n}{\sum_n w_n} \tag{54}$$

resulting in the correlation coefficient

$$\rho_w(x, y) = \frac{\text{cov}(x, y; w)}{\sqrt{\text{cov}(x, x; w) \text{cov}(y, y; w)}} \tag{55}$$

## 5.6 Nonlinear correlations

The standard Pearson correlation coefficient  $\rho_{xy}$  is an index of the *linear* relationship between 2 parameters (dependent and variable), or more parameters (correlation of multiple independent variables with the dependent), or partial correlation of variables and a control variable etc. But there are also many situations where a correlation can exist but  $\rho = 0$  nevertheless. For example any non-linear relation, such as quadratic  $y = ax^2$  has a functional dependency but, because of the  $\pm x$ , has a zero correlation. Such dependencies would need a transformation of  $x$  or  $y$  or both to make the relationship linear, for example using  $\log(x)$  or  $x^p$  or  $-x^p$  etc.<sup>30</sup>

The alternative to data transformation is to use a non-linear correlation index - see section 11.4.

---

<sup>30</sup> Asuero A.G. et al *The correlation coefficient: An overview*, Crit. Rev. Analyt. Chem. **36** (1) 41 (2006).

## 6 Spectral analysis - Linear

This and the next section review linear and non-linear spectral estimation techniques of stationary stochastic and harmonic signals. The analysis of non-stationary signals is addressed in section 10. Frequency-wavenumber spectra are covered in section 8, and higher-order spectra in section 9. Spectral analysis methods can be roughly categorized into two basic groups: (1) **Linear** methods such as Discrete FTs and the Blackman-Turkey (BT) method. These are **Non-parametric** methods where no specific signal structure is assumed. (2) **Non-linear** methods include **Parametric** methods, where the signal is modelled by a parameterized function (as in **autoregressive** models) and **Eigenvector** methods based on the correlation matrix and *sub-domain* separation. The non-parametric group includes specific stationary and non-stationary methods (eg. Wavelets), while parametric and eigenvector methods are generally applicable to both non- & stationary data.

<b>Non-parametric</b>		
Stationary		
	Blackman-Tukey	FT of smooth/windowed and/or truncated autocor
	Periodogram	DFT and FFT with or w/o window/filter
	Welch's	Overlapped windowed segments
Non-stationary		
	SFFT	Short-form or sliding windowed FFT- Non-adaptive
	Wavelet trans.	Non-adaptive wavelet basis fnct.
	Hilbert Spect.	Hilbert transform $\rightarrow$ analytic signal, IMF
	Hilbert-Huang	HHT - Empirical mode decomposition - Adaptive IMF
<b>Parametric</b>		
Stochastic		
	Autoregressive	AR : Model signal
	Moving average	MA : Model noise
	Prony's method	Not discussed here
<b>Eigenvector</b>		
Sinusoidal		
	Pisarenko	Single (noise) $\lambda_{min}$ eigenvalue
	MUSIC	Use all noise subspace $\lambda_n$ eigenvalues
	ESPRIT	Use signal subspace $\lambda_s$ eigenvalues

Table 1: Summary of main spectral estimation techniques.

Table 1 summarizes the more well know methods. There are many reviews in the literature comparing different models. Some of the better include<sup>31 32 33</sup>. Übeyli in particular gives a good review of the pros and cons of the various methods. Several hybrid techniques have also been developed, which mix schemes in an attempt to exploit the strengths of certain methods while limiting their short-comings.

<sup>31</sup> Kay S.M. et al. *Spectrum analysis - A modern perspective*, Proc. IEEE **69** (11) 1380 (1981).

<sup>32</sup> Übeyli E.D. et al *Comparison of eigenvector methods with classical model-based methods in analysis of internal carotid arterial Doppler signals*, Comput. Biol. Med. **33** 473 (2003).

<sup>33</sup> Jain S.K et al *Harmonics estimation in emerging power system: Key issues and challenges*, Elec. Pow. Syst. Res. **81** 1754 (2011).

The **power spectral density (PSD)**, aka the spectrum, is essentially just the frequency distribution of the signal amplitude variance  $\sigma^2$  (2nd moment)! Or put another way, how the signal deviates from the mean as a function of frequency. The power spectral area (summed PSD) should, of course, equal the signal  $\sigma^2$ . Other frequency distributions of signal moments are also possible - for example an increasing Kurtosis (4th moment) with frequency has been used as an indicator of non-stationarity.<sup>34</sup>

### 6.1 Least squares fitting

One of the most effective and basic methods is a simple visual inspection of the raw data. Depending on the signal-to-noise ratio (SNR), coherent features are often quite easily identified by eye - and thus are amenable to linear Least Squares (LS) fitting to a simple model, such as shown in fig. 14 of a single sinusoid of amplitude  $A$ , frequency  $\omega$  and phase  $\phi$ . In this sense, the spectrum is just a LS fit of a set of complex exponentials (ie. multiple sinusoids) to the data.

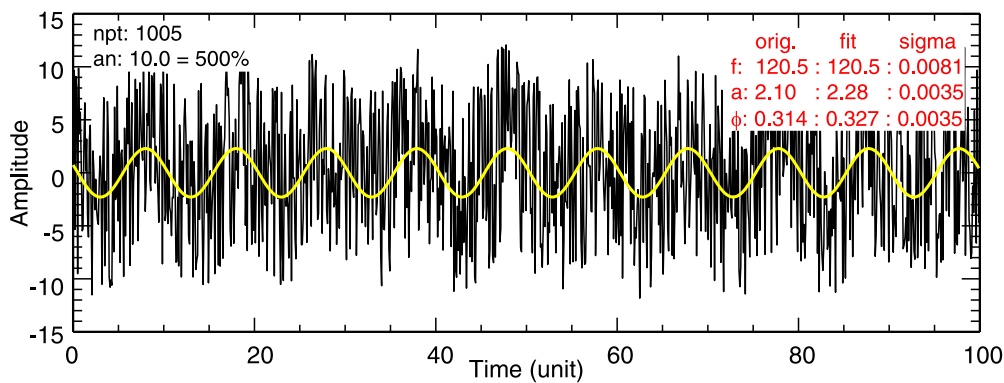


Figure. 14: Example of least-squares fitting showing recovery of a single sinewave buried in uniform Gaussian white noise (SNR = -11.7 dB).

### 6.2 Fourier transforms

The continuous Fourier transform (CFT) and its inverse are defined as

$$X(f) = \int_{-\infty}^{\infty} x(t)e^{-i2\pi ft} dt \quad \text{and} \quad x(t) = \int_{-\infty}^{\infty} X(f)e^{i2\pi ft} df \tag{56}$$

The discrete FT (DFT) forms for a digitized signal of finite length  $T = N\Delta t$  are defined as

$$X_j = \frac{1}{N} \sum_{n=0}^{N-1} x_n \exp(-i2\pi j n/N) \quad \text{and} \quad x_n = \sum_{j=0}^{N-1} X_j \exp(i2\pi j n/N) \tag{57}$$

where  $t = n \Delta t$  with  $n = 0, \dots, N - 1$  the data time index, which gives  $f = j \Delta f$  discrete frequencies, with  $j = 0, \dots, N - 1$  the frequency index. However, due to the sampling theorem only the first  $N/2$  frequency components  $X_j$  are unique (maximum freq.). The frequency resolution (component spacing) is the inverse of the data sample length  $\Delta f = 1/T$ .

<sup>34</sup> Mahdizadeh N et al *Investigation of intermittency in simulated and experimental turbulence data by wavelet analysis*, Phys. Plasmas **11** (8) 3932 (2004).

### 6.2.1 Discrete Cosine transform

The discrete cosine transform (DCT) is a straight forward formulation of the real part of the discrete FT. It is widely used in many areas of science and industry, cf. Wikipedia<sup>35</sup> There are many variations, which are optimizations of the basic form, aimed at different applications. One of the most common forms is the DCT-II, defined as

$$X_j = \sum_{n=0}^{N-1} x_n \cos \left[ \frac{\pi}{N} \left( n + \frac{1}{2} \right) j \right] \quad j = 0, \dots, N-1. \quad (58)$$

The  $(n + 1/2)$  factor in the cosine is intended to give the best transform at the data boundaries.

### 6.2.2 FFT

The Fast Fourier Transform (FFT) is simply a refined and efficient algorithm for computing the DFT. Most FFT routines derive from the original **Cooley-Tukey algorithm** of 1965.<sup>36 37</sup> The basis of the Cooley-Tukey approach is to set  $e^{i2\pi/N} = W_N$  in eqn. (57) to give the form

$$X_j = \frac{1}{N} \left[ \sum_{n=0}^{N-1} x_n^* W_N^{jn} \right]^* \quad (59)$$

where  $*$  is the complex conjugate. Restricting  $N$  to powers of 2 and representing integers  $j$  and  $n$  in binary form allows eqn. (59) to be expanded as a sequence of summations of products involving decreasing powers of  $W_N$ . Taking advantage of various symmetries and redundancies allows the number of complex operations (multiplications and additions) to be reduced from  $N^2$  to  $(N/2) \log_2 N$ . As  $N$  increases the computational savings becomes substantial.

The power spectrum via the FFT (called the periodogram) is then simply given by

$$\begin{aligned} S_{Per}(j\Delta f) &= \langle X_j X_j^* \rangle \\ &= \left\langle \left| \frac{1}{N} \sum_{n=0}^{N-1} x_n w_n \exp(-i2\pi jn/N) \right|^2 \right\rangle. \end{aligned} \quad (60)$$

where  $w_n$  is a window function, which is discussed in section 6.4.2, and the angle brackets indicate ensemble averaging, that is, the required expectation operation  $E[\cdot]$ .

The DFT assumes  $x(t)$  to continue and be periodic outside the sample window  $T$ . For a real valued  $x_n$  this leads to  $\text{Re } X_j$  being symmetric and  $\text{Im } X_j$  being anti-symmetric about  $f_N$ . That is, the  $N/2$  to  $(N-1)$  components  $X_j$  can be interpreted as ‘negative’ frequencies. For a complex valued signal, the  $x_j$  symmetry/anti-symmetry is broken and both the positive and negative frequency  $X_j$  are important. Further issues and ‘restrictions’ of the FFT algorithm are discussed in section 6.4.

<sup>35</sup> [https://en.wikipedia.org/wiki/Discrete\\_cosine\\_transform](https://en.wikipedia.org/wiki/Discrete_cosine_transform) (Dec. 2022)

<sup>36</sup> Bergland G.D. *A guided tour of the Fast Fourier Transform*, IEEE Spectrum **6** 41 (1969).

<sup>37</sup> Bendat J.S. and Piersol A.G. *Random data: Analysis and measurement procedures*, 4th edn. Wiley, N.Y. (2010). First published in 1971.

### 6.3 Auto and cross-spectra

As noted in the previous section, from the Fourier transform component  $X(f)$  of signal  $x(t)$  one computes the (raw) **auto-power** spectrum, defined as <sup>38</sup>

$$S_{xx}(f) = |X^*(f) X(f)| = (\text{Re } X(f))^2 + (\text{Im } X(f))^2 \quad (61)$$

Here, raw means a single un-smoothed spectrum. With the Fourier transform  $Y(f)$  of a second signal  $y(t)$  one can compute the mutual power with the (raw) **cross-power** spectrum, defined as

$$S_{xy}(f) = |X^*(f) Y(f)| = |C(f) + i Q(f)| = (C(f)^2 + Q(f)^2)^{1/2} \quad (62)$$

where the respective (raw) **co-** and **quad-spectra** are given by

$$C(f) = \text{Re } X(f) \text{Re } Y(f) + \text{Im } X(f) \text{Im } Y(f) \quad (63)$$

$$Q(f) = \text{Re } X(f) \text{Im } Y(f) - \text{Re } Y(f) \text{Im } X(f) \quad (64)$$

In practice the respective raw spectra should be averaged over either  $m$  adjacent frequency elements (smoothed) or ensemble averaged over  $M$  different time segments to reduce the spectral variance. The normalized cross-power spectrum is called the **coherence** spectrum, which is computed *after* ensemble averaging the auto and cross spectra. It is defined as

$$\gamma(f) = |\gamma(f)| e^{i\phi(f)} \quad (65)$$

where

$$|\gamma(f)| = \frac{\langle S_{xy}(f) \rangle}{(\langle S_{xx}(f) \rangle \langle S_{yy}(f) \rangle)^{1/2}} \quad (66)$$

with  $0 \leq |\gamma(f)| \leq 1$ , and the corresponding **cross-phase** spectrum

$$\phi(f) = -\tan^{-1} \left( \frac{\langle Q(f) \rangle}{\langle C(f) \rangle} \right) = -\tan^{-1} \left( \frac{\langle \text{Im } S_{xy}(f) \rangle}{\langle \text{Re } S_{xy}(f) \rangle} \right) \quad (67)$$

The angle brackets indicate ensemble averaging of multiple raw spectra instances. Note, without the ensemble averaging the coherence (computed via the FFT) is  $\gamma(f) = 1$  by definition!

A term often used is **coherency**, but confusingly for the complex coherence  $\gamma(f)$ , with coherence meaning  $|\gamma|^2$ . Here, coherency is defined as the frequency averaged spectral coherence:

$$\bar{\gamma} = \frac{\sum |\gamma(f)| S_{xy}(f)}{\sum S_{xy}(f)} \quad (68)$$

---

<sup>38</sup> Smith D.E. et al *Fast-Fourier-transform spectral-analysis techniques as plasma fluctuation diagnostic tool*, IEEE Trans. Plasma Sci. **PS-2** (12) 261-272 (1974).

## 6.4 Fourier transform issues

### 6.4.1 Periodicity & Aliasing

As noted in section 6.2.2 the signal discretion leads to periodicities above the Nyquist frequency  $f_N = f_s/2$  folding back, or aliasing and appearing as lower frequencies. Figure 15(b) shows ensemble averaged FFT spectra ( $N = 256$ ) of 3 (real valued) sinewaves, plotted vs FFT index. It is also common to plot the upper  $N/2$  indexes (above Nyquist  $f_N = 128$  index) as negative frequencies, as shown in fig.15(c). In terms of FFT indexes the real  $f_1 = 30$  cycles per unit time produces also a mirror negative peak at index  $j = (256 - 30) = 226$ , while the  $f_2 = 150$  is aliased to  $j = (256 - 150) = 106$  with a negative peak at  $j = 150$ . Likewise, the real  $f_3 = 311.3$  is aliased to  $j = -(256 - 311.3) = 55.3$  with a negative peak at  $j = 200.4$ . For broadband fluctuation signals the aliasing effect can severally distort the measured spectrum.

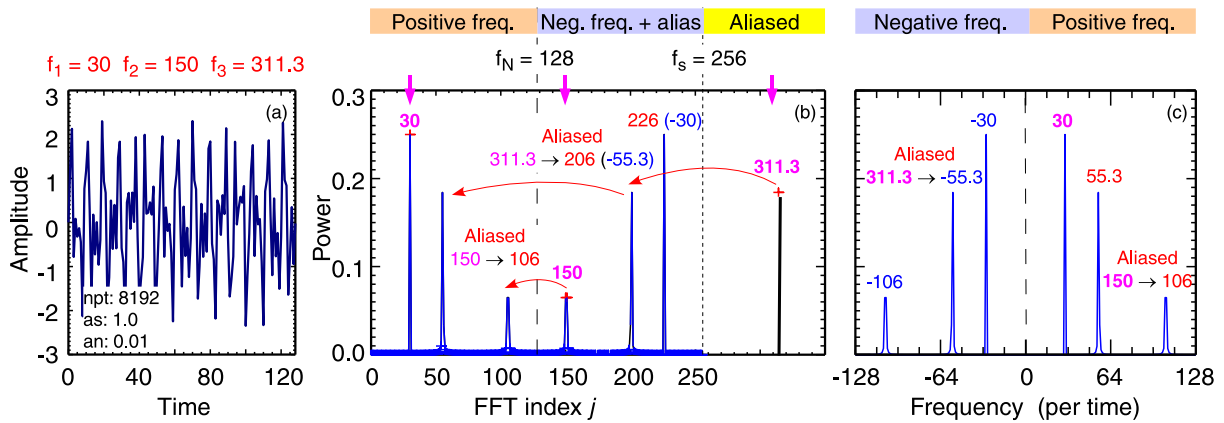


Figure. 15: Example of FFT aliasing of sines above the Nyquist frequency plus negative frequency ‘image’ peaks. (b) Spectra vs FFT index and (c) as double-sided frequency.

### 6.4.2 Leakage and data windowing

Because the DFT data length is limited this is equivalent to multiplying the data  $\{x_n\}$  with a rectangular or boxcar window  $w_n$ . The Fourier transform of a rectangular window  $\{w_n\} = 1$  is a sinc function, which leads to power (positive and negative) leaking from one frequency component  $X_j$  to adjacent components. The solution is to shape the inherent (time restricted) window to one which has reduced FT side-lobes. Generally this means tapering the data amplitude towards zero at either end of the FFT block. Several windows are recommend - cf. Wikipedia.<sup>39</sup>

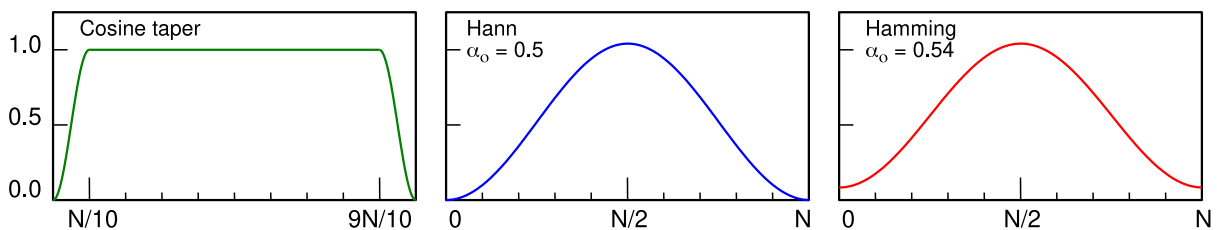


Figure. 16: Various window functions for leakage reduction.

<sup>39</sup> [https://en.wikipedia.org/wiki/Window\\_function](https://en.wikipedia.org/wiki/Window_function) (Dec. 2022).



Some of the more popular window functions are shown in fig. 16 and include:

(1) Extended cosine bell/taper: applied to the first 1/10th and last 1/10th of the data window.<sup>36 40</sup>

$$w_n = \begin{cases} \cos^2(2\pi 10n/N) & \text{for } 0 < n < N/10 \\ 1 & \text{for } N/10 \leq n < 9N/10 \\ \cos^2(2\pi 10(n - N)/N) & \text{for } 9N/10 \leq n < N \end{cases} \quad (69)$$

(2) Cosine-sum windows: with  $\alpha_0 = 0.5$  (Hann) or  $0.54$  (25/46) (Hamming).

$$w_n = \alpha_0 - (1 - \alpha_0) \cos(2\pi n/N) \quad (70)$$

(3) Blackman window:<sup>41 42</sup> with  $\alpha_0 = (1 - \alpha)/2$ ,  $\alpha_1 = 1/2$ , and  $\alpha_2 = \alpha/2$ , with  $\alpha = 0.16$

$$w_n = \alpha_0 - \alpha_1 \cos(2\pi n/N) + \alpha_2 \cos(4\pi n/N) \quad (71)$$

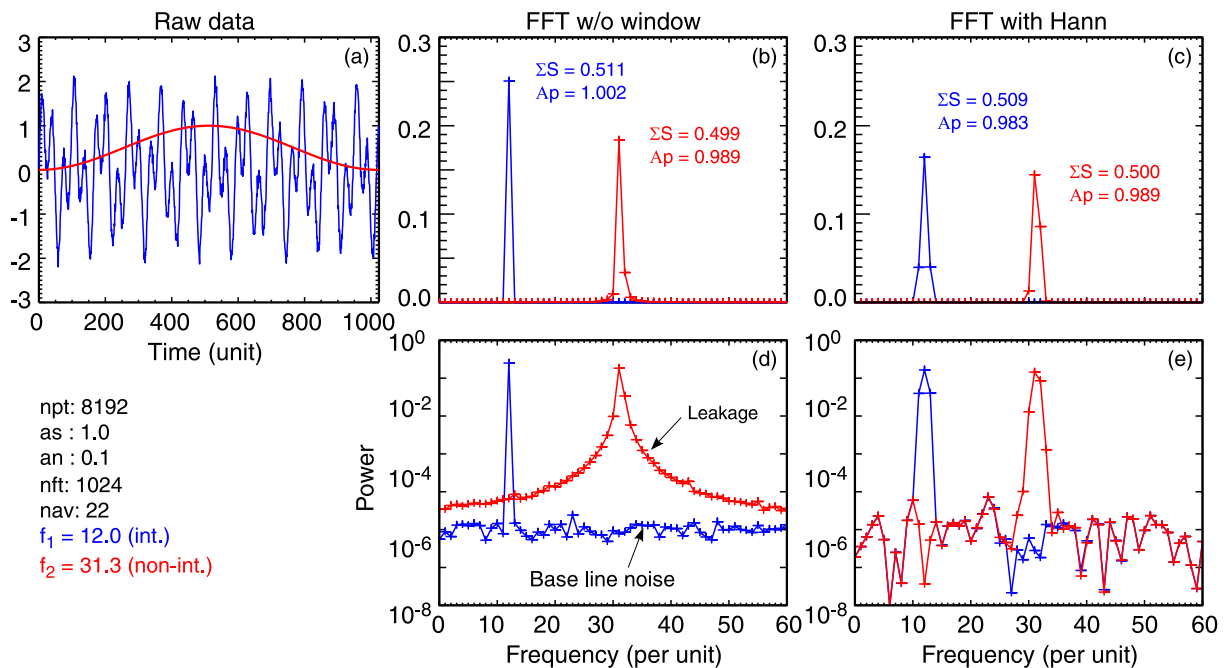


Figure. 17: Example of spectral leakage with integer (blue) and non-integer (red) sinewave frequencies, without (b) & (d) and with (c) & (e) Hann window (inc.  $w_{cor}$  correction). SNR = 34 dB.  $\Sigma S$  = summed spectral power,  $A_p$  = sine amplitude from peak power via eqn. (74).

A related leakage issue is the so-called **Picket-fence effect**. Spectral frequencies that fall between the discrete orthogonal  $X_j$  values will spread their power to adjacent  $X_j$  components, distorting the spectrum. As shown in fig. 17 for two unity amplitude sines, one with an integer (blue) and one with non-integer frequency (red), the effect of leakage can be very pronounced with power bleeding over a very wide frequency band, resulting in spectral peak reduction, cf. red peak in figs. 17(b) linear and (d) log power scale.

<sup>40</sup> Otnes R.K. & Enochson L. *Applied time series analysis*, Wiley N.Y. (1978).

<sup>41</sup> Oppenheim A.V. *Discrete-time signal processing*, Prentice Hall (1989).

<sup>42</sup> Mitra S.K. *Digital signal processing*, 2nd edn. McGraw-Hill (2001).

Note, for a sinusoid of integer frequency and amplitude  $A_{pk} = 1$  the power spectrum has a positive & negative frequency peak, each of magnitude  $pk = (A_{pk}/2)^2 = 0.25$ . Quoted in the spectra plots here are the parameters:  $\Sigma S$  for the total summed spectral power ( $\approx \sigma^2$ ),  $pk$  for the spectral peak maxima, and  $A_p$  for the recovered peak/mode amplitude using eqn. (74) below.

Applying a window function (eg. Hann) confines the spectral leakage to a narrower bandwidth, but at the expense of frequency resolution.<sup>31</sup> For coherent peaks the (controlled) power spread to adjacent  $X_j$  frequency components results in peak broadening - even for integer frequencies - but with off-peak leakage now reduced to below the background noise level, figs. 17(c) and (e).

The effect of convolving the signal with a window is an overall reduction in the spectral power, such that the summed power no longer equals the signal variance. To recover the signal variance  $\sigma_d^2$  the windowed spectrum  $S_w(f)$  must be rescaled by multiplying by a correction factor  $w_{cor}$

$$S(f) = (w_{cor}/H) \times S_w(f) \tag{72}$$

$$w_{cor} = \left( N / \sum^N w_n \right)^2 \tag{73}$$

which is inversely proportional to the area under the window. For a Hann window  $w_{cor} = 1/\alpha_0^2 = 1/0.5^2 = 4$ , and  $1/0.54^2$  for a Hamming window. The extra  $H = 1.5 - 1.52$  factor is required for the controlled leakage. Of course, even with the power correction, coherent peaks will still be smaller than in the non-windowed case due to the power spreading in frequency.

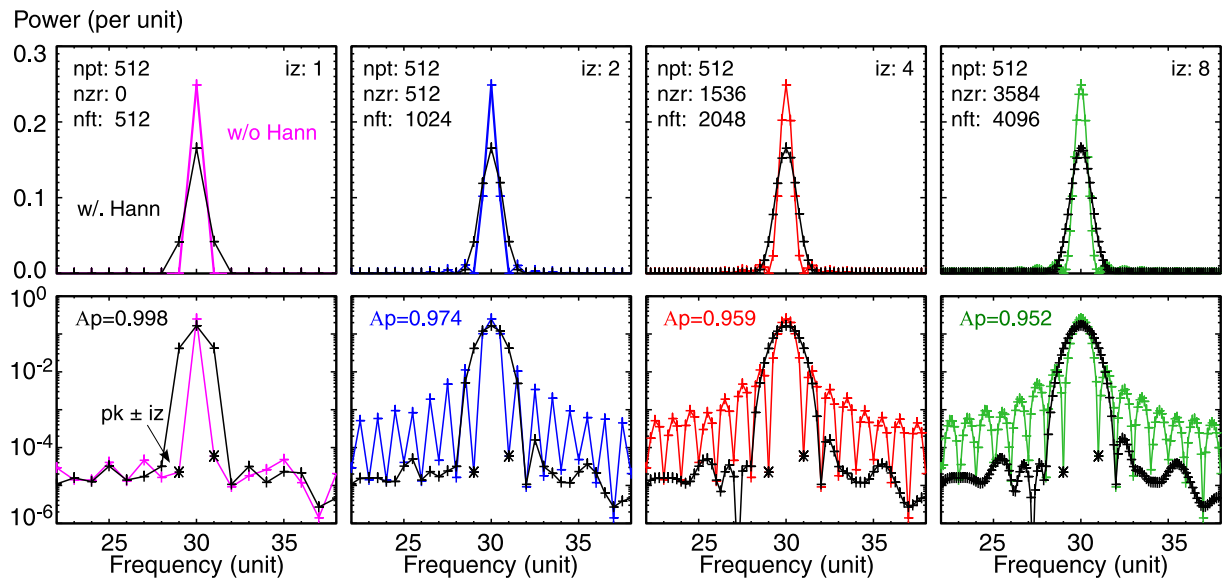


Figure. 18: Effect of zero-padding on FFT spectra of integer frequency ( $f_o = 30$ ) unity amplitude sinewave plus white noise ( $SNR = 42$  dB). Without (coloured) and with (peak power scaled) Hann window (black). Top row: linear, and bottom row: log power scales.  $S_n(f) = S(n) * iz$ .

### 6.4.3 Zero padding

For broad-band spectra the picket-fence effect appears as modulation or ripple in the spectra. One mitigation technique is to essentially smooth the spectrum via complex interpolation of the  $X_j$  coefficients.<sup>36</sup> This can be achieved by padding the data set  $\{x_n\}$  of length  $N = 2^n$  points with

extra blocks of  $N$  zeros. (Adding zeros in the time domain is equivalent to interpolation in the frequency domain.) The result, for a single sinewave, shown in fig. 18 with increasing multiples of  $N$  blocks of zeros, is to make the FFT compute interleaved frequencies. As can be seen, zero padding does not improve the inherent frequency resolution of the peak (cf. black starred points), but simply adds frequencies between the original  $X_j$  components, which may help to resolve multiple peaks. However, a side-effect of zero-padding (without windowing) is the formation of harmonics. With windowing (applied only to the data segment and not to the zero padding) the harmonics are smoothed out, as shown by the black curves and points in fig. 18.

A second side-effect of zero-padding is to reduce the spectral power, which must now be rescaled by  $i_z = \text{nft}/\text{npt}$  (ratio of extended data length to real data length) in order to recover the summed spectral power  $\Sigma S(f) = \sigma_d^2$ . While  $\Sigma S$  is now correct the spectral peak will, nevertheless, still decrease as power is spread over more points with increasing  $i_z$ . In fig. 18 for display purposes the spectra have been multiplied by a further  $i_z$  (see code snippet) to maintain the peak height.

```

npt = 512 & alpha = 0.5 & H = 1.52
wind = alpha - (1.0-alpha)*Cos(2.0*!pi*Findgen(npt)/npt)
wcor = (Float(npt)/Total(wind))^2 ; =alpha !
; For iz=1 => nft=512 iz=2 => 1024 iz=4 => 2048 iz=8 => 4096
freq = Findgen(npt*iz) / iz
zero = Fltarr(npt*(iz-1))
b = FFT([data,zero], -1)
S = Abs(b*Conj(b)) * iz ; recover sigma_d
SS = Total(S[1:*])
pk = Max(S[0:*], ip) ; pwr peak
Ap1 = 2.*Sqrt(Total(S[ip-(iz):ip+(iz)])) ; Amp peak
Ap2 = 2.*Sqrt(Total(S[ip-(iz+1):ip+(iz+1)]))
Sn = S * iz ; invariant peak
; With window
b = FFT([data*wind,zero], -1)
Sw = Abs(b*Conj(b)) * wcor/H * iz
SSw = Total(Sw[1:*])
pkw = Max(Sw[0:*], ipw)
Ap1w = 2.*Sqrt(Total(Sw[ipw-(iz):ipw+(iz)]))
Ap2w = 2.*Sqrt(Total(Sw[ipw-(iz+1):ipw+(iz+1)]))
Swn = Sw * iz

```

*Example. 12 : Code snippet (IDL) demonstrating zero padding of sinewave with white noise. Note windowed spectra are scaled by correction factor  $w_{cor}/H$  - cf. fig. 18.*

#### 6.4.4 Mode amplitude from FFT

As implied above it is quite possible to recover the amplitude of a single frequency harmonic mode from the FFT power spectrum. For moderate a SNR the amplitude  $A_{pk}$  of an isolated frequency spectral component can be recovered by summing the power under the peak. Generally the sum should only be over the coherent peak. To obtain a reasonably good estimate it is sufficient to take a couple of FT index points either side of the peak index, ie.  $i_p \pm 1$  or 2

$$A_{pk} = 2 \times \left( \sum_{j=i_p-(i_z+1)}^{i_p+(i_z+1)} S_{(n)}(f_j) \right)^{1/2} \quad (74)$$

In the case of zero padding one should use  $i_p \pm (i_z + 1)$  where  $i_z$  is the padding ratio, ie.  $i_z = 1$  for no padding. When using zero-padding the peak width is given by  $\pm i_z$ , as indicated by the starred spectral points in fig. 18 with a summation over  $\pm(i_z + 1)$  (ie. Ap2 in the code snippet above). The factor 2 before the square-root is required if the summation is only over one side (positive frequencies) of the FFT. Also, note  $S(f)$  should have already been corrected with  $w_{cor}/H$  if using windowing. These formulas and correction factors work well, even down to SNR of  $< 3$  dB or less - as long as sufficient points are used to resolve the peak.

### 6.4.5 Truncation and partial zero padding

The Cooley-Tukey FFT algorithm is a so-called radix-2 algorithm in that it performs the DFT by dividing (ie. factorizing) the transform into successively smaller  $N/2$  chunks to achieve a computational advantage. Hence the data length restriction to  $N = 2^n$ . However, other factorizations are also possible, leading to mixed-radix, prime-factor, and co-prime or split-radix algorithms, as well as other approaches, such as the chirp-z transform, Rader or Bluestein algorithms.<sup>43</sup>

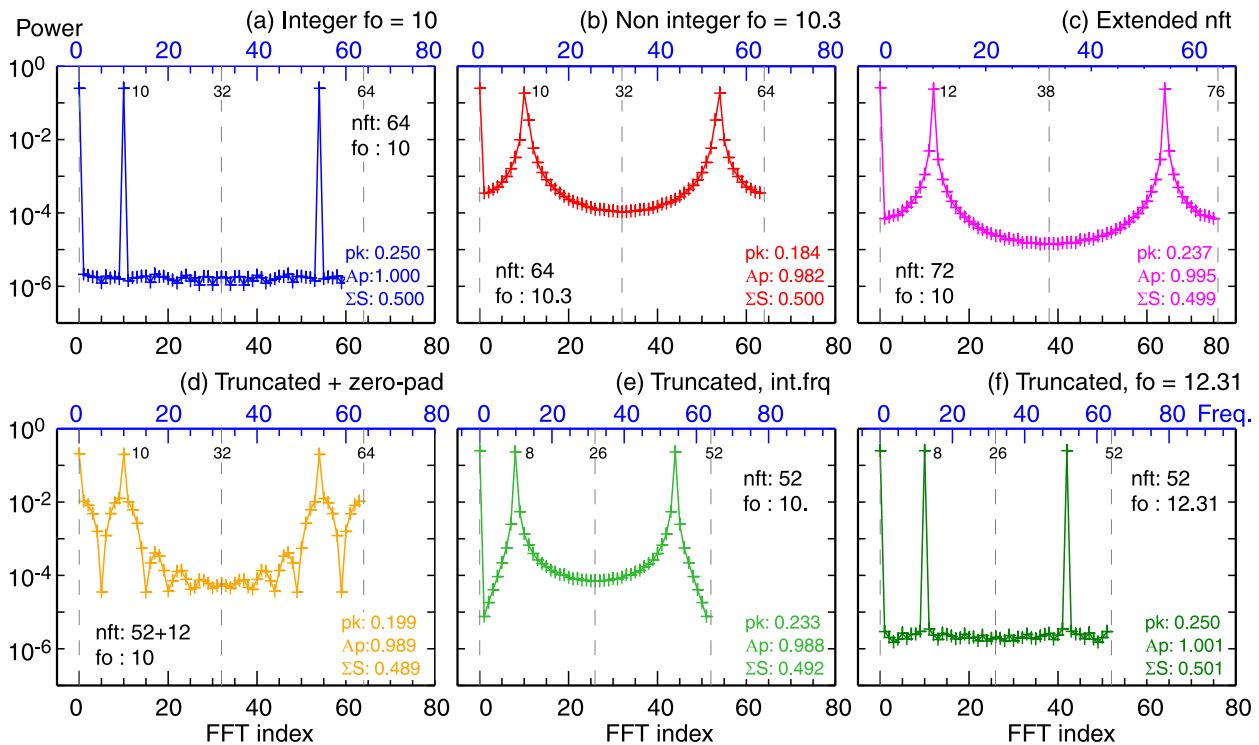


Figure. 19: Spectra (nave=32 w/o window) using IDL mixed-radix FFT of various lengths, demonstrating truncation and zero-padding effects. Sinewave plus Gaussian noise,  $\sigma_d^2 = 0.50$ .

Most computer language packages provide straight-forward FFT routines. Depending on sophistication, for non-power-of-two data lengths  $N \neq 2^n$ , the routine might do one of several things: (1) Complain - not particularly user-friendly! (2) Automatically truncate or zero-pad the data length out to the next  $2^n$  (eg. python numpy.fft). (3) Perform data interpolation. Or (4) switch to an alternate algorithm (eg. IDL : mixed-radix, or python scipy.fft : Bluestein).

<sup>43</sup> Wikipedia - <https://en.wikipedia.org/wiki/FFT> and [https://en.wikipedia.org/wiki/Cooley-Tukey\\_FFT\\_algorithm](https://en.wikipedia.org/wiki/Cooley-Tukey_FFT_algorithm) (Checked Apr. 2023).

As example, fig. 19 shows a selection of spectra of a single unity amplitude sinusoid with frequency  $f_o$  and DC offset of 0.5, plus a small amount of Gaussian white noise, using the IDL provided FFT routine (a mixed-radix algorithm from the Intel Math Kernel Library) - see code snippet below. The number of FFT points (without windowing) is deliberately small  $N < 100$  to highlight effects. Full (positive and negative frequency) spectra are shown with the DC peak of  $\sqrt{1./2}$  at FFT index  $j = 0$  (lower x-axis).

Figures 19(a) for integer  $f_o = 10$ , and (b) non-integer  $f_o = 10.3$  with  $N = 64$  FFT  $X_j$  points, reproduce the effects of spectral leakage shown in fig. 17. In fig. 19(c) the data length is extended to  $N = 76$  and in (e) truncated to  $N = 52$  respectively. (Note the IDL routine returns the same number of  $X_j$  components as the input data length.) Although in both (c) & (e) the sinewave  $f_o = 10$  is integer, the effective FFT frequency  $f = f_o * N/2^n$  with  $2^n = 64$  is non-integer in both cases, and thus spectral peak leaks occurs. The scaled frequency index  $f = j \times 2^n/N$  corresponding to the FFT index is shown in the upper blue x-axis. In fig. 19(f) increasing  $f_o = 12.307$  with  $N = 52$  gives an effective integer frequency of  $f = 10 = \text{Round}(12.307 \times 52/2^6)$  and thus no peak leakage. What is important to note is that the FFT reference length is taken (radix-2 in these cases) to the nearest  $2^n$  points!

```

dt = 1./1000. & nz = 12 & nft = 64 & fo = 10 & ofs = 0.5 .
time = Findgen(2060)*dt
sig = Sin(2.*!pi*time*fo) + ofs ; long Sine + DC
zero = Fltarr(nz)
; length = 64 : Standard (blue) fo = 10.
fs = FFT(sig[0:nft-1], -1)
S1 = Abs(fs*Conj(fs))
; length = 76 : Extended (purple) fo = 10.
fs = FFT(sig[0:nft+12-1], -1)
S3 = Abs(fs*Conj(fs))
freq = Findgen(nft+12)/(nft+12)*nft ; freq adjusted
; length = 64 : Truncate + zero (orange) fo = 10.
fac = 1. + Float(nz)/Float(nft-1-nz)/2.
fs = FFT([sig[0:nft-1-nz],zero], -1)
S4 = Abs(fs*Conj(fs)) * fac^2 ; Note *fac^2
SS4 = Total(S4[1:*]) / fac ; Note /fac
; length = 52 : Truncate, no pad (green) fo = 10.
fs = FFT(sig[0:nft-nz-1], -1)
S5 = Abs(fs*Conj(fs))

```

*Example. 13 : Code snippet (IDL) for computing partial zero/truncated spectrum - cf. fig. 19.*

A common approach in many routines is to either truncate the data down, or zero pad the data up to the nearest  $2^n$  (eg. `numpy.fft`). In fig. 19(d) the 52 data points are zero-padded up to 64. Temporal zero-padding is equivalent to interpolation in the frequency domain, thus while the peaks now align with the expected FFT indexes, the peaks of course are now subject to the zero-padding harmonic/leakage effect noted in section 6.4.3. In all cases it is highly recommended to always use a window on the data before zero padding.

The final message here is, that it is important to understand how your preferred FFT routine actually works. Advice - do not use 'black-box' routines on trust. If in doubt, test the routine.

### 6.4.6 Convolution and correlation

Digital filtering is often performed by convolving the signal with a kernel. The convolution of two signals in the time domain is the same as multiplying the two signals in the frequency domain,  $c(t) = x(t) * h(t) \Leftrightarrow X(f) \cdot H(f)$ . Using FFTs:

$$\hat{c}_n = \mathcal{F}^{-1} \{ \mathcal{F}\{x_n\} \cdot \mathcal{F}\{h_n\} \} \quad (75)$$

where  $\mathcal{F}$  and  $\mathcal{F}^{-1}$  represent the forward and inverse FFT. However, to avoid cyclic effects in the convolution it is necessary to double the length of the signals to  $2N$  and set the second half to zeros - Bergland gives an excellent description of the issues.<sup>36</sup>

$$\begin{aligned} \hat{x}_n &= x_n \quad \text{for } 0 < n < N \\ \hat{x}_n &= 0 \quad \text{for } N \leq n < 2N \end{aligned} \quad (76)$$

### 6.4.7 Overlap averaging

This technique, commonly called the Welch spectra,<sup>44</sup> involves ensemble averaging data segments which are overlapped. Up to a 50% overlap can reduce the spectral standard error (but not the spectral bias error) - see below for discussion of spectral errors. More than 50% temporal overlap does not improve the error further as the segments are no longer fully independent. Segment overlapping is also commonly used in the sliding or short-time STFT for resolving time evolutions - see section 10.1.

### 6.4.8 Cepstrum

An extension to the power spectrum  $P(f) = |\mathcal{F}\{x(t)\}|^2$  is the cepstrum.<sup>45</sup> This is a technique widely used in the field of acoustics and sonar etc. for resolving periodic components in the spectral frequency. The approach is to take the FFT of the log of  $P$

$$C(\tau) = | \mathcal{F} \{ \log( |\mathcal{F}\{x(t)\}|^2 ) \} |^2. \quad (77)$$

Note that the FT of an FT is a 'spectrum' with components of cycles per Hz, eg. seconds. Essentially back to the time domain and towards an autocorrelation. However, taking the log of the first power spectrum has the effect of converting any temporal convolutions to an additive spectrum. The polycepstrum has also been defined as the inverse FT of the log of the corresponding polyspectrum (where poly means 2nd or higher-order spectra)<sup>46 47</sup>

<sup>44</sup> Welch P.D. *The use of fast Fourier transform for the estimation of power spectra: A method based on time averaging over short, modified periodograms*, IEEE Trans. Audio Electro. **AU-15** (2) 70 (1967).

<sup>45</sup> Noll M.A. *Cepstrum pitch determination*, J. Acoust. Soc. Am. **41** (2) 293 (1967).

<sup>46</sup> Swami A. et al. *Higher-order spectral analysis toolbox, User's guide, v.2*, The MathWorks Inc. <https://www.ligo.caltech.edu/rana/mat/HOSA/HOSA.PDF> (1998).

<sup>47</sup> Oppenheim A.V. & Schaffer R.W. *Digital signal processing*, New Jersey, Prentice-Hall (1989).

## 6.5 Confidence limits & confidence intervals

As for correlation estimators (section 5.3), the linear spectral estimators also have a confidence or significance level (SL), ie. the minimum value of the estimate above which the estimate is significant, and a confidence interval (CI) or band, ie. the error bar on the mean value of the estimate. These are often related, but not necessarily the same. Bendat<sup>48</sup> gives perhaps the most definitive description and derivation of the statistical errors associated with the various spectral estimators. Firstly, there are two types of errors: (1) the bias error in the estimate, which expresses the basic goodness of the estimator, ie. does it under or over estimate? And (2) the random or statistical variance error:

$$(1) \text{ bias}[\hat{A}] = E[\hat{A}] - A$$

$$(2) \text{ var}[\hat{A}] = E[(\hat{A} - E[\hat{A}])^2]$$

with the **normalized standard error**  $\epsilon[\hat{A}] = \sigma[\hat{A}]/A$ .

The bias scales with the resolution bandwidth  $B_e = 1/T$  while the variance error  $\epsilon \propto 1/\sqrt{M}$ . Since all spectral estimates need to be ‘smoothed’ either by averaging  $m$  adjacent frequencies or  $M$  ensembles, there will always be a trade-off in either the frequency resolution  $\Delta f = m/\Delta t$ , or the length of each ensemble  $T = N\Delta t$  vs the number of ensembles  $M = T_{tot}/T$ .

With ensemble averaging Bendat gives the following formulas for the normalized standard errors:

$$\epsilon[\hat{S}_{xx}(f)] \approx 1/\sqrt{M} \quad (78)$$

$$\epsilon[\hat{S}_{xy}(f)] \approx 1 / (|\gamma_{xy}(f)| \sqrt{M}) \quad (79)$$

$$\epsilon[\hat{\gamma}_{xy}^2(f)] \approx \sqrt{2} (1 - \gamma_{xy}^2(f)) / (|\gamma_{xy}(f)| \sqrt{M}) \quad (80)$$

and for the transfer function  $|\hat{H}_{xy}(f)|^2 = \hat{\gamma}_{xy}^2(f) \hat{S}_{yy}(f)/\hat{S}_{xx}(f)$

$$\epsilon[\hat{H}_{xy}(f)] \approx (1 - \gamma_{xy}^2(f))^{1/2} / (|\gamma_{xy}(f)| \sqrt{2M}) \quad (81)$$

from which, when  $\epsilon[\hat{H}_{xy}(f)]$  is small, eg.  $\epsilon < 0.1$  gives the cross-phase error

$$\epsilon[\hat{\phi}_{xy}(f)] = \Delta\hat{\phi}_{xy} \approx \sin \Delta\phi \sim \epsilon[|\hat{H}_{xy}(f)|] \quad (82)$$

Smith<sup>38</sup> also provides the following formulas when using frequency averaging for the spectral *variances*. When converted to  $\epsilon$  these are fairly close in form to those of Bendat:

$$\text{var}[\hat{S}_{xy}(f)] \approx |\hat{S}_{xy}(f)|^2 [1 + |\hat{\gamma}_{xy}(f)|^{-2}] / 2m \quad (83)$$

$$\text{var}[\hat{\gamma}_{xy}(f)] \approx [1 - |\hat{\gamma}_{xy}(f)|^2]^2 / 2m \quad (84)$$

$$\text{var}[\hat{\phi}_{xy}(f)] \approx [|\hat{\gamma}_{xy}(f)|^{-2} - 1] / 2m \quad (85)$$

<sup>48</sup> Bendat J.S. *Statistical errors in measurement of coherence functions and input/output quantities*, J. Sound Vibration **59** (3) 405-421 (1978).

### 6.5.1 Coherence significance level

There are also various formulas provided in the literature for the  $\gamma$  coherence statistical significance level (SL)  $c$ . At the most basic level  $c$  depends on the number of ensemble averages  $M$ , and whether averaging of  $m$  neighboring frequencies was used. For a normal error distribution in the coherence this can be approximated as:

$$c \approx 1/\sqrt{mM} \tag{86}$$

However, for small  $\gamma$  values (close to zero) the error distribution is not normal but more like an asymmetric Poisson. In this case, taking a  $\chi^2$  or F-distribution gives a  $\gamma^2$  confidence level for a particular  $\alpha$

$$c^2 = F_{2,2q}(\alpha) / (F_{2,2q}(\alpha) + q). \tag{87}$$

$F_{\nu_1, \nu_2}$  is the quantile percent point function (see section 3) with lower and upper percentiles, ie. dof of  $\nu = 2$  and  $2q = 2(M - 1)$ .<sup>49</sup> There are tables for the  $F$ -distribution for different  $\alpha$  values in most statistics text books cf.<sup>50</sup> and computer packages, such as Matlab, have routines for its computation. Fortunately, the F-statistic for  $\gamma$  can be distilled down to something simpler<sup>51 52</sup>

$$c = \sqrt{1 - \alpha^{1/q}} \tag{88}$$

where  $\alpha = 1 - p$ , (eg.  $\alpha = 0.01$  for  $p = 99\%$  confidence) and  $q = M - 1$ . Note the significance level  $c$  is independent of the frequency  $f$ .

To give a brief summary of the correlation and coherence errors.<sup>53</sup> The concept of statistical stability is set by the resolution bandwidth  $B_e$ , and the (individual) record length  $T = N\Delta t$  where  $\Delta t =$  sample interval.  $B_e = 1/\tau_{max}$  is the correlation bandwidth ( $\tau_{max} = m\Delta t$  and  $m$  is maximum lag), thus the FFT bandwidth is  $B_e = 1/T = M/N\Delta t$  (where  $M$  is the number of elementary components averaged), then the degrees of freedom: dof  $\nu = 2B_eP$ .

If  $N$  is moderately large, say  $N > 30$ , or for small values of  $\epsilon$ , then the distribution approaches Gaussian / normal and the percentage error, or **coefficient of variation** (standard error):  $\epsilon_o = 1/\sqrt{B_eT}$  which leads to:

	$\nu$ dof	$B_e$ b/w	$\epsilon_o$ error
Correl.	: $2N/m$	: $1/\tau_{max}$	: $\sqrt{m/N}$
FFT	: $2M$	: $1/T$	: $\sqrt{1/M}$

<sup>49</sup> Shumway R.H. & Stoffer D.S. *Time series analysis and its applications, with R examples*, 3rd edn. Free texts in Statistics (2014) p159. Subsequently published as *Time series, A data analysis approach using R*, Chapman & Hall (2019).

<sup>50</sup> Bevington P.R. et al *Data reduction and error analysis for the physical sciences*, McGraw-Hill, 2nd edn. (1992).

<sup>51</sup> Thompson R.O. *Coherence significance levels*, J. Atmos. Sci. **36** (10) 2020 (1979).

<sup>52</sup> <https://dsp.stackexchange.com/questions/16558/statistical-significance-of-coherence-values> (Checked 2022).

<sup>53</sup> Otnes R.K. & Enochson L. *Applied time series analysis*, Wiley, NY. (1978) - chapter 8.



## 6.6 Blackman-Tukey method

One of the oldest methods to compute the periodogram is the *indirect* method of Blackman & Tukey (BT).<sup>1</sup> Here the signal auto-covariance estimate  $\hat{R}_{xx}(k)$  is computed using eqn. (46) for a data set of  $x_n$  ( $n = 0, \dots, N - 1$ ), and then Fourier transformed to the frequency domain<sup>31</sup>

$$\begin{aligned}
 S_{BT}(f) &= 2\Delta t \sum_{k=0}^{K-1} \hat{R}_{xx}(k) w_k \exp(-i2\pi f k \Delta t) \\
 &= 2\Delta t \sum_{k=0}^{K-1} \hat{R}_{xx}(k) w_k \cos(2\pi f t) + \hat{R}_{xx}(k) w_k \sin(2\pi f t).
 \end{aligned}
 \tag{89}$$

Here the auto-covariance is computed using the full data set of  $N$  points, but the Fourier transform of  $J$  frequencies, is only made over  $K \ll N$  lags, ie. the covariance is truncated. Originally this was an advantage since the FT is the computationally expensive part. Figure 20(a) shows the raw BT spectra of a sine wave in noise obtained using the code snippet below. Because the expectation operation on the covariance does not transfer through the FT it also necessary to smooth the periodogram, either by averaging over adjacent frequencies - compare fig. 20(a) and (b), or ensemble averaging over several spectra. In this case the  $N$  point data set can be sub-divided into smaller segments before correlating. The most striking feature is how different the BT spectra look compared to equivalent FFT spectrum - much smoother with narrow coherent peaks. In fact, peaks are generally much narrower than can be achieved with comparable FFTs.

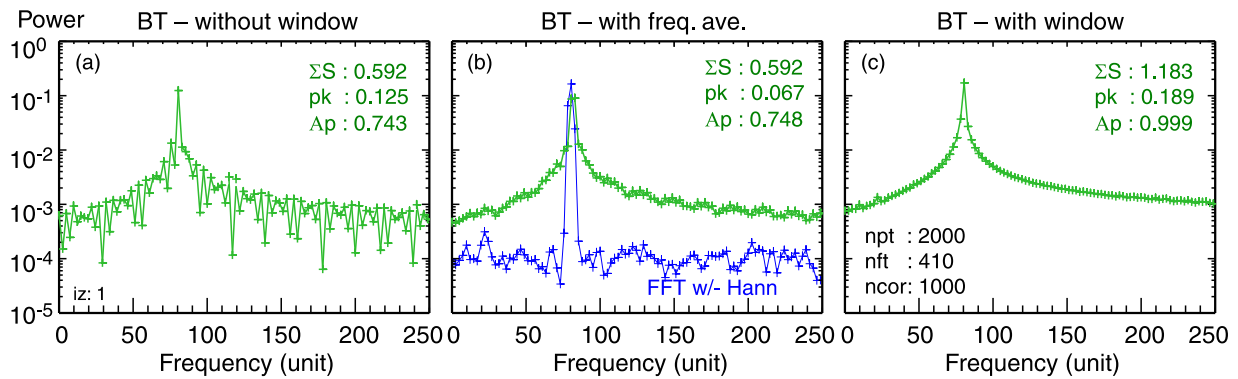


Figure. 20: Comparison of (a) raw BT spectra, (b) with freq. averaging and (c) with window, for  $N = 2000$ ,  $K = 1000$  (ncor) and  $J = 410$  (nft) of sine  $f_o = 80.1$  in noise ( $an=0.2$ )  $\sigma_d^2 = 0.547$ .

As with the FFT spectrum it is also *highly* recommended to use a window on the auto-covariance estimates, in order to reduce leakage effects - compare fig. 20(a) without, and (c) with a half Tukey cosine window  $w_k = 0.5 \times (1 + \cos(2\pi k / K))$  applied to the positive covariance lags. Other windows (eg. Parzen or Bartlett) are also possible. The window removes the covariance edge effects and gives smoother spectra with minimal reduction in frequency resolution. The choice of  $N$ ,  $K$  and  $J$  sets the spectral quality. Lag truncation  $K < N$  is essential for spectral consistency (variance decreasing with larger  $N$ ). A general recommendation<sup>54</sup> is:  $1/20 < K/N < 1/3$ .

<sup>54</sup> Chatfield C. *The analysis of time series: An introduction*, Chapman & Hall, (2004).

```

npt = 8000 & dt = 1/1000 & time = Findgen(npt)*dt & fo = 80.1 .
data = Sin(!pi*2.*time*fo)+an*RandomN(seed, npt) ; build test signal
; Auto covariance
lag = Indgen(ncor) & M = Abs(lag) & acv = Fltarr(ncor)
win = 0.5*(1.+Cos(!pi*Findgen(ncor)/Float(ncor)))
bcor = (Float(ncor)/(Total(win)))^2 & H = 1.5
FOR k = 0L, ncor-1L DO $
    acv[k] = Total(data[0:npt-1- M[k]]*data[M[k]:])/Float(npt-M[k])
; Fourier Transform via complex exponentials
psd = Fltarr(nft) & fqb = Findgen(nft)/Float(nft)/dt
a = Complexarr(ncor)
FOR j = 0L, nft-1L DO BEGIN ; Full freq range
    f = Float(j)/(Float(nft)*dt)
    FOR k = 1,ncor-1 DO a[k]=acv[k]*win[k]*Exp(Complex(0,-2.*!pi*k*f*dt))
    psd[j] = Abs(Total(a))
ENDFOR
psd = dt*(acv[0] + 2.*psd)/!pi * (bcor/H) ; Scale spectrum
    
```

Example. 14 : Code snippet (IDL) for computing BT spectrum, after Meko<sup>16</sup> - cf. fig. 20.

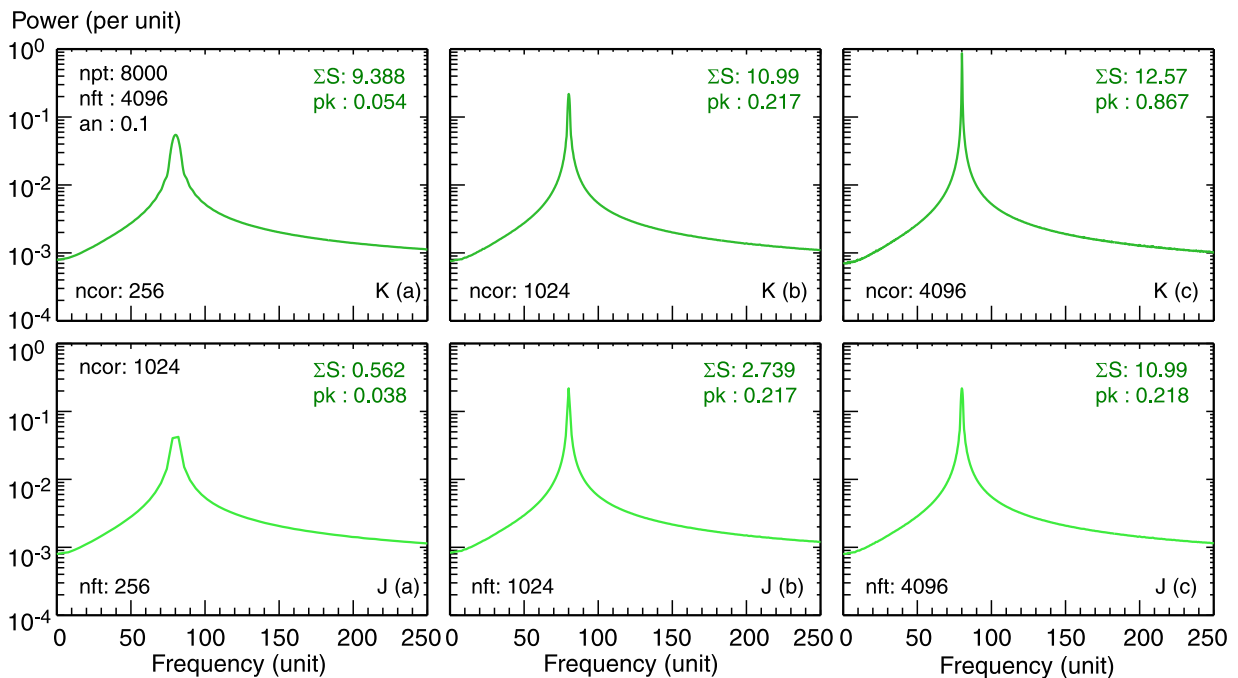


Figure. 21: Effect on windowed BT spectra of (top row) increasing covariance lags  $K$  (ncor) with fixed  $N = 8000$  (npt) and  $J = 4096$  (nft) frequencies, and (bottom row) increasing  $J$  with fixed  $K$  and  $N$ , for sinewave (as = 1, fo = 80.1) with Gaussian noise (an = 0.1).  $\sigma_d^2 = 0.513$

Here, the BT formula proposed by Meko (ie. Chatfield) eqn. 90, was used in preference to the one above, since, together with the with window factor, the peak scales more robustly

$$S_{BT}(f) = \frac{\Delta t}{\pi} \left( \hat{R}_{xx}(0) + \sum_{k=1}^{K-1} \hat{R}_{xx}(k) w_k \exp(-i2\pi f k \Delta t) \right) \times (bcor/H). \quad (90)$$

Figure 21 (top row) shows the effect of increasing lag  $K$  on the spectral resolution with fixed number of reconstructed frequencies  $J$  and data length  $N$ . The larger  $K$  the better the spectral resolution and the narrower and stronger the coherent peaks (ie. larger spectral SNR), but at the cost of increased confidence bands (spectral error bars). The degrees of freedom scales as  $\nu =$

$2.67N/M$  (which sets the  $\chi^2$  confidence interval) while the bandwidth scales as  $bw = 4/(3M)$ . The number of frequencies  $J$  of course also needs to be large enough to take advantage of the inherent resolution - as in fig. 21 J(c) example with  $J = 4096$  (nft). Note that the background noise floor remains constant. Increasing  $N$  does not reduce the noise floor, but it does reduce the error bars. The bottom row of fig. 21 shows in J(a) the under resolved peak, while in fig. 21 J(b) with  $J = K$  (as recommended by Meko<sup>16</sup>) the peak is optimally resolved. In fig. 21 J(c), increasing  $J > K$  simply add more points, filling in the spectrum (as with zero padding of the FFT), within the resolution defined by  $K$ .

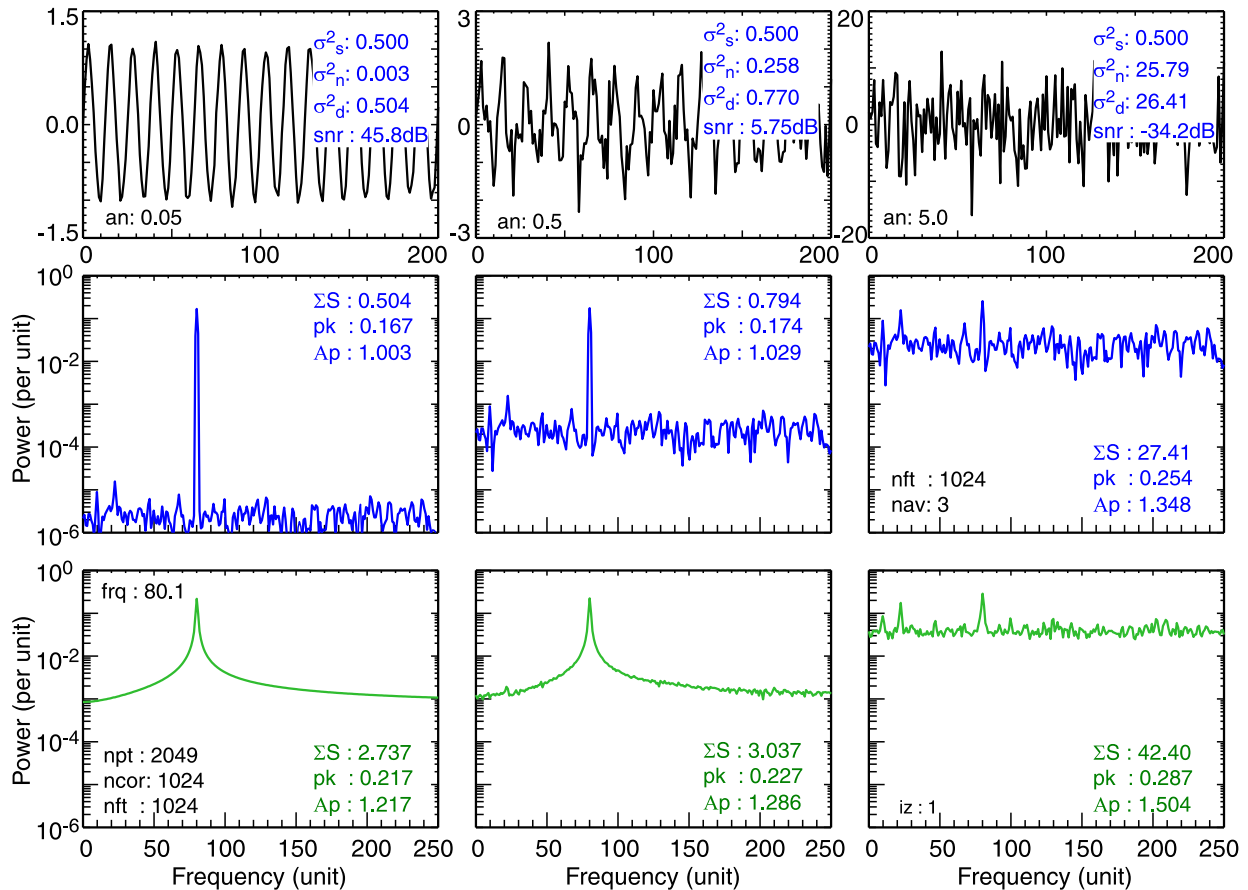


Figure. 22: Effect of increasing Gaussian white noise (+45 to -34 dB SNR) on sinewave windowed BT spectra with  $J = 1024$  frequencies, and  $K = 1024$  lags and  $N = 2049$  data points. FFT with Hann window.

The behaviour of the BT spectra with broadband noise is also rather different to the direct FFT method. As seen in the spectra of fig. 22 with increasing levels of white noise relative to sinewave amplitude, there is generally a fairly constant minimum spectral floor (which unfortunately scales with the overall covariance level). As shown in fig. 22(b) and (c), only at significant noise levels does the broadband background level begin to scale with the noise variance  $\sigma_n^2$ . Increasing the sinewave amplitude raises the coherent peak proportionately. What this example demonstrates well is the ability of the BT method in recovering coherent features at extreme noise levels with **very** short data lengths. That is, in conditions where the direct FFT periodogram struggles.

## 7 Spectral analysis - Non-linear

This section discusses some of the non-linear spectral estimators, beginning with the various flavours of autoregressive methods, and then the eigenvector methods based on the singular value decomposition (SVD) of the auto-covariance matrix.

### 7.1 Autoregressive moving-average models

Autoregression moving-average (ARMA) spectral methods are based on parametric models of the persistence, ie. autocorrelation of the time series. They offer alternatives to discrete Fourier transforms and are particularly useful for short time series or non-stationary time series. They are also of value for extracting harmonic signals from noise. There is a vast body of literature on these and related methods. Some good overviews include: <sup>55 31 56</sup>

#### 7.1.1 AR spectra and Yule-Walker equations

Autoregressive spectral methods (AR) express the current (zero-mean) data value  $x_n$  as a linear series of previous values, plus a residual white noise term  $w_n$

$$x_n = a_1 x_{n-1} + a_2 x_{n-2} + \dots + a_p x_{n-p} + w_n = \sum_{l=1}^p a_l x_{n-l} + w_n \tag{91}$$

where  $a_l$  are predictor or weighting coefficients and  $p$  is the order of the linear series, ie. the number of regression terms. From this the power spectral density is given by <sup>55 57</sup>

$$\hat{S}_{AR}(f) = \frac{\sigma_n^2 \Delta t}{|1 - \sum_{l=0}^p a_l \exp(-i 2\pi f l \Delta t)|^2} = \frac{\sigma_n^2}{|A(f)|^2} \tag{92}$$

where  $A(f) = 1 - a_1 e^{-i2\pi f} - \dots - a_p e^{i2\pi fp}$ . Note eqn. (91) is sometimes defined as  $x_n = -\sum a_l x_{n-l} + w_n$ , in which case the minus sign in eqn. (92) becomes a plus etc.<sup>59</sup> One approach to obtaining the  $a_l$  coefficients and the **noise**  $\sigma_n^2 = E[w_n w_n]$  is via the Yule-Walker recursion equation, which relates  $a$  and the autocorrelation  $R_{xx}$  (with a full derivation in Kay<sup>31</sup>)

$$R_{xx}(k) = \sum_{l=1}^p a_l R_{xx}(k-l) + \sigma_n^2 \tag{93}$$

which can be expressed as a set of equations in matrix form with  $\sigma_n^2 = \sigma_{l=0}^2$ ,

$$\begin{bmatrix} R_{xx}(0) & R_{xx}(-1) & \dots & R_{xx}(-p) \\ R_{xx}(1) & R_{xx}(0) & \dots & R_{xx}(1-p) \\ \vdots & \vdots & & \vdots \\ R_{xx}(p) & R_{xx}(p-1) & \dots & R_{xx}(0) \end{bmatrix} \cdot \begin{bmatrix} 1 \\ a_1 \\ \vdots \\ a_p \end{bmatrix} = \begin{bmatrix} \sigma_0^2 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \tag{94}$$

<sup>55</sup> Ulrych T.J. & Ooe M. *Chapter 3: Autoregressive and mixed autoregressive-moving average models and spectra in Nonlinear methods of spectral analysis*, 2nd edn. Springer (1983).

<sup>56</sup> Takalo R. et al *Tutorial on univariate autoregressive spectral analysis*, J. Clin. Monit. Comput. **19** 401 (2005) and erratum (2006).

The **Levison-Durbin algorithm** is an efficient Gaussian elimination method for solving eqn. (94) and can be summarized as follows:<sup>31</sup> The first step is to initialize for  $k = 1$

$$a_{11} = -R_{xx}(1) / R_{xx}(0) \quad (95)$$

$$\sigma_1^2 = (1 - |a_{11}|^2) R_{xx}(0) \quad (96)$$

followed by a recursive loop for  $k = 2, 3, \dots, p$

$$a_{kk} = - \left[ R_{xx}(k) + \sum_{l=1}^{k-1} a_{k-1,l} R_{xx}(k-1) \right] / \sigma_{k-1}^2 \quad (97)$$

$$a_{k,l} = a_{k-1,l} + a_{kk} a_{k-1,k-l}^* \quad (98)$$

$$\sigma_k^2 = (1 - |a_{kk}|^2) \sigma_{k-1}^2 \quad (99)$$

The double subscripts on  $a$  and  $\sigma^2$  indicate storage of previous values in the summation and parameter updating steps. An advantage of this method is that all values of the coefficients up to order  $p$  are stored. This is useful since the optimal order  $p$  is not necessarily known a priori.

```

npt = 8192 & an = 0.1 & dt = 1/npt & fo = 80.
time = Findgen(npt)*dt
data = Sin(2.*!pi*time*fo) + an*(2*RandomU(seed, npt)-1.)
pmx = 100 ; max AR order no.
aic = Fltarr(pmx)
FOR p = 2, pmx-1 DO BEGIN
  res = TS_COEF(data, p, MSE=sigma2) ; IDL suppl. routine
  aic[p] = Alog(sigma2) + 2*p/npt
ENDFOR

```

*Example. 15 : Code snippet (IDL) to calculate AIC vs order  $p$  - cf. fig. 23(a).*

There are various formulas for selecting  $p$ , such as the Akaike information criterion (AIC)<sup>55 56</sup> which reaches the first significant minimum at the optimal  $p$

$$\text{AIC}(p) = \ln \hat{\sigma}_p^2 + 2p/N \quad (100)$$

where  $\hat{\sigma}_p^2$  (eg. the noise  $\sigma_n^2$ ) is the prediction mean-square-error associated with order  $p$ . In practice the optimal  $p$  may also be taken where  $\sigma_p^2$  saturates or falls to a preselected value. Also there is the Finite Prediction Error (FPE), the Minimum Description Length (MDL) - eqn. (118), and the Criterion Autoregressive Transfer (CAT)<sup>31</sup> - all basically giving similar results, cf.<sup>59</sup>

$$\text{FPE}(p) = \hat{\sigma}_p \left( \frac{N + p + 1}{N - p - 1} \right) \quad (101)$$

Figure 23 shows a comparison of AR spectra for a single sinewave plus Gaussian white noise for increasing AR orders  $p = 20, 40, 80, 160$  and  $320$  using the IDL TS\_COEF routine which employs the Levison-Durbin recursion algorithm - see code snippet below. The optimal spectrum

<sup>57</sup> Iwana N. et al *Least-squares autoregression (maximum entropy) spectral estimation for Fourier spectroscopy and its application to electron cyclotron emission from plasmas*, J. Appl. Phys. **52** (9) 5466 (1981).

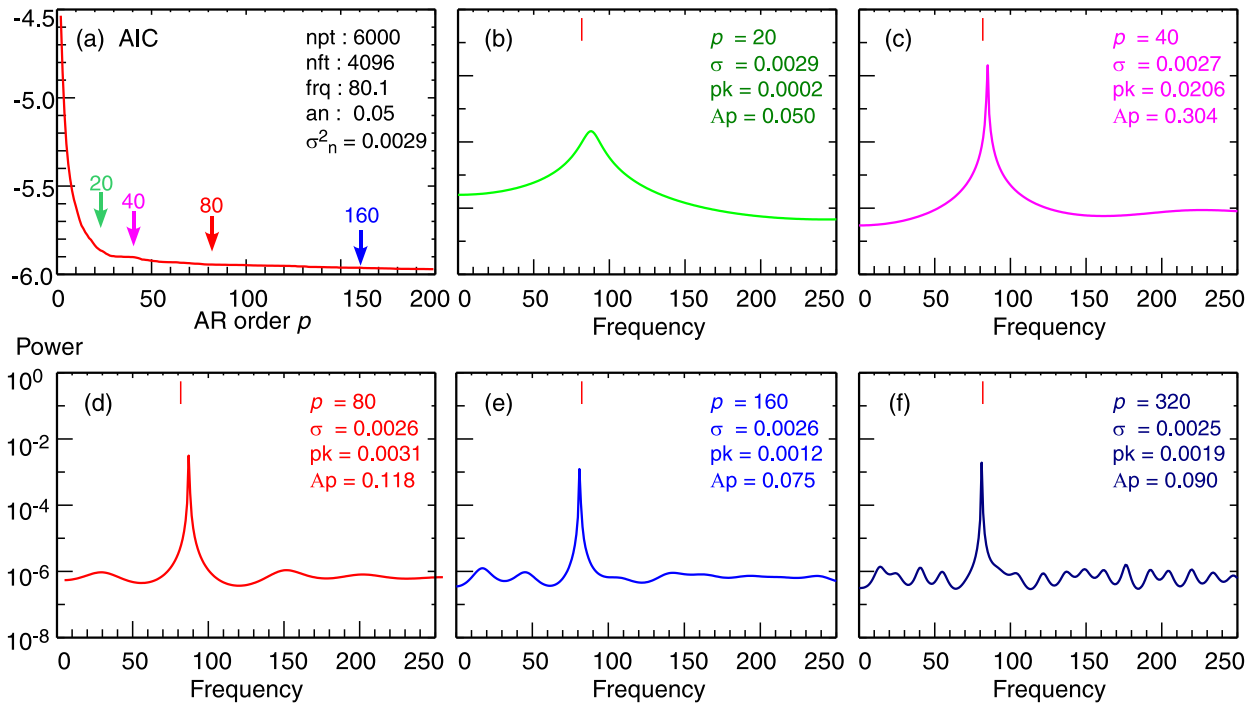


Figure 23: (a) AIC vs AR order  $p$ . (b-f) AR spectra for orders  $p = 20, 40, 80, 160$  and  $320$ , for sinewave (as = 1) plus Gaussian noise (an = 0.05) using IDL TS\_COEF routine employing the Levison-Durbin recursion algorithm.

is with  $p \approx 40 - 60$ . Figure 23(a) shows the AIC as a function of  $p$  (see code snippet) falling rapidly, then leveling off for  $p \geq 40$ . For too low a  $p$  the AR spectral peak in fig. 23(b) is poorly resolved, even with  $J = 4096$  reconstructed frequencies. For  $p = 40$  the peak in fig. 23(c) is now well resolved. With increasing order  $p$  the resolution improves further as the peak base narrows and the peak amplitude stabilizes, but at the expense of minor spectral artifacts (structure) as seen in fig. 23(e). As with the BT spectrum, in comparison with the corresponding FFT spectrum (ensemble averaged) in fig. 24 the spectra is greatly smoothed, depending on the order  $p$ .

```

nft = 4096 & p = 60 & dt = 1./nft           ; FT points & AR order.
a = TS_COEF(data, p, MSE=sigma2)           ; IDL supl. routine
psd = Fltarr(nft)
apr = Complexarr(p)
; Compute FT using complex exponentials
FOR j = 0, nft-1 DO BEGIN
  f = Float(j)/(nft*dt)
  FOR k = 0, p-1 DO apr[k] = a[k]*Exp(Complex(0,-2.*!pi*k*f*dt))
  psd[j] = Abs(1.- Total(apr))^2
ENDFOR
psd = dt*sigma2 / psd                       ; invert & scale

```

Example 16: Code snippet (IDL) to calculate AR spectra via IDL TS\_COEF routine - cf. fig. 23.

Nevertheless, the AR spectrum is sensitive to the signal to noise ratio (SNR). Figure 24 shows the effect of increasing the (Gaussian) noise amplitude from an = 0.005 to 5 (ie. SNR 85.7 to -34.3 dB). As the  $\sigma_n^2$  increases it is necessary to also increase the AR order  $p$  in order to retain the resolution of the peak. Comparing the AR and FFT spectra, the overall spectral shape and the

scaling of the background spectral level with the noise is convincing. However, the integrated (summed) AR spectral power  $\Sigma S(f)$  does not follow the signal variance  $\sigma_d^2$  since the spectra are scaled to the noise variance  $\sigma_n^2$ . In addition, while the FFT retains the constant coherent peak amplitude, the AR peak is compressed at low noise levels (high SNR) since coherent peaks in the AR spectra are not simply linearly related to power.<sup>31</sup> Nevertheless, at sufficiently high SNR the area under the AR peak is proportional to power  $\Sigma pk \propto P$  while the peak magnitude  $S_{pk} \propto P^2$ .

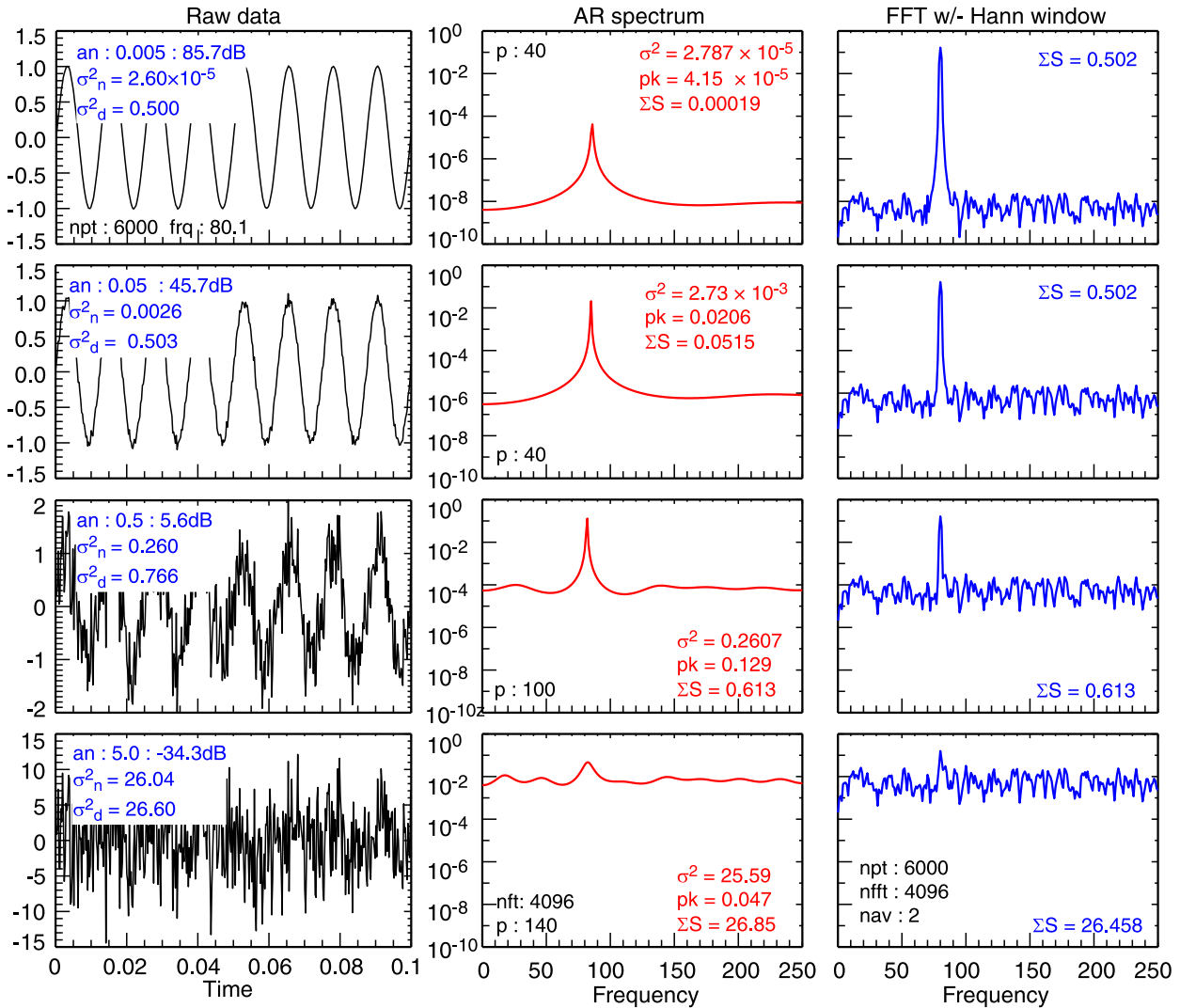


Figure. 24: (Left) raw signal, (middle) AR spectra and (right) FFT power spectra with Hann window, for unity amplitude sinewave of frequency  $f_o = 80.1$  plus varying Gaussian noise amplitude ( $an = 0.005, 0.05, 0.5$  and  $5$ ).  $\sigma^2$  is the AR mean-square error, which is approximately equal to the noise  $\sigma_n^2$ . Frequencies  $J = 4096$  for both AR and FFT, but AR order  $p$  is varied.

Generally, the larger the noise level  $\sigma_n$ , or the signal complexity (ie. bandwidth) the larger the AR order  $p$  required to resolve features. For real data this means scanning  $p$  to find the optimal order. For short data lengths  $p$  can not be too large. Here, reducing unnecessary complexity by prior low-pass filtering the signal would be an option. Overall, the AR spectra nevertheless, remains superior to the FFT for short data lengths.

### 7.1.2 Maximum Entropy Method (MEM)

The maximum entropy (ME) method (also called **Burg's method**) is an alternate way of fitting the AR model. The essence of the method is “to choose the spectrum which corresponds to the most random time series whose autocovariance agrees with the measured values”<sup>58</sup> Inverting the AR eqn. (91) one can define a pair of forward and backward prediction errors (ie.  $\sigma^2$  s)

$$e_{p,n}^f = x_n + \sum_l^p a_l x_{n-l} = e_{p-1,n}^f + \rho_p e_{p-1,n-1}^b \quad (102)$$

$$e_{p,n}^b = x_{n-p} + \sum_l^p a_l x_{n-p+l} = e_{p-1,n-1}^b + \rho_p^* e_{p-1,n}^f \quad (103)$$

for  $n = p + 1, \dots, N$ . The second halves of the equation pairs are recursive expressions, where  $\rho_p = -2 \sum_{p+1}^N e_{p-1}^f e_{p-1}^b / \sum_{p+1}^M (e_{p-1}^f)^2 + (e_{p-1}^b)^2$  are called reflection coefficients, which are related to the  $a_l$  and  $\hat{r}$ . Since  $|\rho_p| \leq 1$  they are also called partial correlations (PARCOR). The prediction error equations can be solved via a Levinson recursion or least-squares. Finally, the spectrum is given by

$$\hat{S}_{ME}(f) = \frac{e_p^f + e_p^b}{|1 + \sum_{l=0}^p \hat{a}_l \exp(-i 2\pi f l)|^2} \quad (104)$$

An alternate ME representation is the **AR autocorrelation extension**. In order to mitigate the frequency smearing induced by a *truncated*  $R_{xx}(k)$ , cf. eqn. (93), the covariance estimates  $\hat{r}_k$  are extended with AR predictions using the previously derived  $a_k$  and  $\sigma_p$  values.

$$\hat{r}_k = \begin{cases} R_{xx}(k) & \text{for } |k| \leq p \\ \sum_{l=1}^p a_l \hat{r}_{k-l} & \text{for } |k| > p \end{cases} \quad (105)$$

From Kay<sup>31</sup>, the result is essentially an extended Blackman-Tukey spectrum:

$$\hat{S}_{ME}(f) = \Delta t \sum_{k=-\infty}^{\infty} \hat{r}_k \exp(-i 2\pi f k \Delta t) \quad (106)$$

### 7.1.3 Least-Squares spectral estimator

Least squares (LS) is another numerical approach to the spectral estimation where the difference between the estimated (ie. from the data) covariance  $\hat{r}_k$  and the predicted covariance from the AR coefficients is minimized.<sup>57</sup> Using the Yule-Walker equation (93) one can write the function to be minimized as

$$I_p(\mathbf{a}, \sigma^2) = \frac{1}{K+1} \sum_{k=0}^K \left( r_k - \sum_{l=1}^p a_l r_{k-l} - \sigma^2 \delta_{k,0} \right)^2 \quad (107)$$

Here one can use standard LS minimization techniques, or via matrix methods, cf.<sup>56 55</sup>

It should be noted that the AR process can be viewed as linear filtering. The **Kalman-Filtering** method is a type of AR process solved via LS.

<sup>58</sup> Haykin S. and Kesler S. *Chapter 2: Prediction-error filtering and maximum-entropy spectral estimation* in Nonlinear methods of spectral analysis, 2nd edn. Springer (1983).



### 7.1.4 Moving average spectra (ARMA)

The AR spectrum has problems resolving sharp troughs. This can be addressed by expressing the noise term  $w_n$  also as a series of lag dependent terms to form an AR-moving average model

$$x_n = \sum_{l=1}^p a_l x_{n-l} + \sum_{l=0}^q b_l w_{n-l} \quad (108)$$

where  $w_n$  is a white noise signal with variance  $\sigma_n^2$ , which again can be solved using a least-mean-squares approach. The corresponding spectrum is <sup>59</sup>

$$\hat{S}_{ARMA}(f) = \sigma_n^2 \left| \frac{B(f)}{A(f)} \right|^2 = \frac{\sigma_n^2 \Delta t \left| \sum_{l=-q}^q b_l \exp(-i2\pi f l \Delta t) \right|^2}{\left| 1 - \sum_{l=0}^p a_l \exp(-i2\pi f l \Delta t) \right|^2} \quad (109)$$

If the  $a_l$  terms are zero then the resulting pure moving average spectrum is equivalent to the Blackman-Tukey spectrum.<sup>31</sup>

$$\hat{S}_{MA}(f) = \sum_{k=-q}^q R_{xx}(k) \exp(-i2\pi f k) \quad (110)$$

### 7.1.5 AR summary

AR methods have several advantages over the FFT, firstly there is the absence of restrictions on the choice of frequency. Once the  $a$  coefficients are computed then  $f$  is a free choice. That is, AR can offer higher frequency resolution than FFTs which are restricted by the data length. Of course, the maximum spectral frequency is still set by the sample period  $\Delta t$ . Secondly, there is no need for windowing the data to control leakage. The AR analysis is also particularly good for short data lengths and for narrow-band signals. However, one disadvantage is that AR methods are relatively more computationally expensive than the FFT, and the spectral discrimination can be sensitive to the SNR.

There are example codes (recursion) in IDL and in python, and of course good source codes in the excellent Numerical Recipes handbook.<sup>60</sup>

<sup>59</sup> Übeyli E.D et al *Comparison of eigenvector methods with classical and model-based methods in analysis of internal carotid arterial Doppler signals*, Comput. Biol. Med. **33** 473 (2003).

<sup>60</sup> Press W. et al. *Numerical recipes in Fortran - The art of scientific computing*, 2nd edn. Cambridge Uni. Press (1992).

## 7.2 Eigenvector methods

The autocorrelation matrix can also be decomposed using eigenvector methods to produce high frequency resolution spectra. These techniques are particularly good for recovering a single, or a few, harmonic components from noise dominated signals. There are several methods (or variations on a theme) including the PHD, MUSIC and ESPRIT sub-domain methods.<sup>59</sup>

### 7.2.1 Pisarenko Harmonic Decomposition (PHD)

The oldest method, developed by Pisarenko<sup>61</sup> starts with the assumption that the signal consists of  $L$  uncorrelated sinusoids (complex exponentials) corrupted by white noise:

$$x_n = \sum_{i=1}^L a_i \exp(i\omega_i n \Delta t) + w_n = \sum_{i=1}^L a_i x_{n-1} + w_n = s_n + w_n \quad (111)$$

which leads to a special case ARMA process that is amenable to eigenvector analysis. The essence is that the signal auto-covariance matrix  $\mathbf{R} = E[\mathbf{x}^* \mathbf{x}^T] = \mathbf{s} + \mathbf{w}$ , formed with  $M \times M$  lags from  $N$  data points can, with some assumptions, be formulated as an eigenequation

$$\mathbf{R} \mathbf{a} = \sigma_n^2 \mathbf{a}. \quad (112)$$

The ARMA parameter  $\mathbf{a} = [1, a_1, \dots, a_L]^T$  is the eigenvector associated with the noise  $\sigma_n^2 = \text{diag}\{\lambda_1, \dots, \lambda_K\}$  eigenvalue array. In Pisarenko's original work the eigenequation was solved order by order until a minimum  $\lambda_{min}$  was reached.<sup>31</sup> A more efficient approach is to use singular value decomposition (SVD) routines<sup>62</sup> where the covariance matrix is expressed as

$$\mathbf{R} = \mathbf{U} \mathbf{W} \mathbf{V}^H \quad (113)$$

with  $\mathbf{U}$  and  $\mathbf{V}$  the left and right eigenvectors and  $\mathbf{W}$  the eigenvalue diagonal matrix.  $H$  is the Hermitian, complex conjugate transposed. The eigenvectors can be divided into a signal subspace (domain) of length  $L$  and an orthogonal noise subspace of length  $K = M - L$ . Since all the sinusoids are assumed to have the same white noise corruption, all the noise eigenvalues *should* be the same and  $\approx \sigma_n^2$ . Thus, one searches  $\mathbf{W}$  for the minimum eigenvalue  $\lambda_{min}$  and then constructs a so-called pseudo-spectrum from the corresponding  $\mathbf{a}$  vector (ie.  $\mathbf{V}$  normalized to the 1st element) via a summation of complex exponentials for each frequency  $f$

$$\hat{S}_{Pis}(f) = \frac{1}{|A(f)|^2} \quad \text{where } A(f) = \sum_{i=L}^{M-L} a_i e^{-i2\pi f i}. \quad (114)$$

The method is particularly good at extracting coherent peaks in the spectrum and can, in principle, can give very high frequency resolution for the sinusoids. However, it is not strongly robust to "false zeros" arising from the use of a single eigenvector belonging to the minimum noise eigenvalue  $\lambda_{min}$ .

<sup>61</sup> Pisarenko V.F. *The retrieval of harmonics from a covariance function*, Geophys. J. R.astro. Soc. **33** 347 (1973).

<sup>62</sup> Kung S-Y et al *Improved Pisarenko's sinusoid spectrum estimate via SVD subspace approximation methods*, 21st IEEE Conf. on Decision & Control, 1312 (1982).

### 7.2.2 MUSIC method (MULTiple Signal Classification)

The MUSIC method extends the Pisarenko method by averaging the pseudo-spectra over all the available eigenvectors of the noise subspace, so as to minimize the effects of false zeros.<sup>63</sup>

$$S_{MU}(f) = \frac{1}{(1/K) \sum_{l=L+1}^{L+K} |A_l(f)|^2}. \quad (115)$$

One could use the covariance matrix  $\mathbf{R} = \mathbf{X}^H \mathbf{X}$ , as in the Pisarenko method, however, a more robust approach is to form an averaged forward and backward linear prediction (FBLP) estimator matrix  $\mathbf{X}$  directly from the  $x_n$  and  $x_n^*$  data.<sup>64 65</sup> which has a double dimension ( $2N_p \times M$ )

$$\mathbf{X} = \frac{1}{\sqrt{2N_p}} \begin{bmatrix} x_M & x_{M-1} & \dots & x_1 \\ x_{M+1} & x_{M+2} & \dots & x_2 \\ \vdots & \vdots & & \vdots \\ x_N & x_{N-1} & \dots & x_{N-M+1} \\ \hline x_1^* & x_2^* & \dots & x_M^* \\ x_2^* & x_3^* & \dots & x_{M+1}^* \\ \vdots & \vdots & & \vdots \\ x_{N-M+1}^* & x_{N-M+2}^* & \dots & x_N^* \end{bmatrix}. \quad (116)$$

where  $N_p = N - M + 1$  with  $N$  the data window size and  $M$  the number of expected eigenvalues, ie. the matrix order. Expressing in SVD form  $\mathbf{X} = \mathbf{U} \mathbf{W} \mathbf{V}^H$ , where the left and right singular eigenvectors are  $\mathbf{U}$  ( $2N_p \times 2N_p$ ) of  $\mathbf{X} \mathbf{X}^H$ , and  $\mathbf{V}$  ( $M \times M$ ) of  $\mathbf{R} = \mathbf{X}^H \mathbf{X}$  respectively, and  $\mathbf{W}$  ( $2N_p \times M$ ) the real, non-negative eigenvalues. Similar to the Pisarenko method, the eigenvalues are used only to identify the noise components, with the MUSIC pseudo-spectrum itself formed from the eigenvectors alone, but in this case all  $K$  noise vectors  $\mathbf{E}_n = [V_{L+1}, V_{L+2}, \dots V_M]$

$$S_{MU}(f) = \frac{\mathbf{e}^H(f) \mathbf{e}(f)}{\mathbf{e}^H(f) \mathbf{E}_n \mathbf{E}_n^H \mathbf{e}(f)} \quad (117)$$

where  $\mathbf{e}(f) = [e^{-i2\pi ft}, e^{-i4\pi ft}, \dots]$  are complex sinusoids. Concerning implementation, the code snippet below gives an adaption of a pmusic code<sup>66</sup> There are two options for the final Fourier transform step, either via complex exponentials, or if the number of reconstructed frequencies is a power of 2 (or zero padded) then via an FFT. Together with the forward-backward matrix this gives a particularly efficient computation.

<sup>63</sup> Schmidt R.O. *Multiple emitter location and signal parameter estimation*, IEEE Trans. Ant. & Prop. **AP-34** (3) 276 (1986).

<sup>64</sup> Haykin S. et al *Chapter 7: Recent advances in spectral estimation*, in Nonlinear methods of spectral analysis, p.245, 2nd edn. Springer (1983).

<sup>65</sup> Iwata T. et al *Application of the multiple signal classification (MUSIC) method for one-pulse burst-echo Doppler sonar data*, Meas. Sci. Technol. **12** 2178 (2001).

<sup>66</sup> Vermare L. et al *Detection of geodesic acoustic mode oscillations using multiple signal classification analysis of Doppler backscattering signal on Tore Supra*, Nucl. Fusion **52** 063008 (2012)

```

dt = 1./1000. & & nl = 2 & nc = 7           : nc = Order (M)      .
nw = 256 & np = nw - nc + 1                ; nw = window size (N)
; Load matrix with real & conj data directly
x = Make_Array(nc, 2*np, /Complex, Value=0)
FOR i = 0, np-1 DO BEGIN                    ; np = data pts. (N-M+1)
  FOR k = 0, nc-1 DO BEGIN
    x[k,i] = data[i-k+nc-1]                ; load matrix w. data &
    x[k,i+np] = Conj(data[i+k])           ; 2nd half w. conjugate
  ENDFOR
ENDFOR
x = x / Sqrt(2.*np)                          ; normalize matrix
; Solve matrix : W = singular-values, U & V = eigenvectors
LA_SVD, x, W, U, V                          ; IDL supplied routine
; Fourier Transform
S = Fltarr(nf)
IF (cexp) THEN BEGIN                        ; Opt.1 via complex exp.
  a = Make_Array(1, nc, /Complex)
  E = V[nl:nc-1,*]                          ; nl:nc-1 = noise comp (K)
  FOR j = 0, nf-1 DO BEGIN                  ; nf = no. frequencies
    f = 1.*j/(nf*dt)
    FOR l = 0, nc-1 DO a[l] = Exp(Complex(0.,-2.*!pi*l*f*dt))
    at = Conj(Transpose(a))
    S[j] = S[j] + Total(Abs(at##E)^2)      ; matrix mult.
  ENDFOR
  S = S / nf^2
ENDIF ELSE BEGIN                            ; Opt.2 via FFT pwr of 2
  z = Make_Array(1, nf, /complex, Value=0)
  FOR i = nl, nc-1 DO BEGIN
    z[0:nc-1] = V[i,*]                      ; Use nc components (M)
    z[nc:nf-1] = 0.                         ; pad with zeros
    fftz = FFT(z, -1)
    S = S + Abs(fftz*Conj(fftz))
  ENDFOR
ENDELSE
S = (dt/S) / nf^2 / np^0.25                 ; Invert & Scale
Sn = S * sigma2d / Total(S)                ; Scale area to sigma^2d

```

*Example. 17 : Code snippet (IDL) to calculate pMUSIC pseudo-spectrum - cf. fig. 25.*

As with the Pisarenko method, the issue of how to determine the optimal number of eigenvectors remains. In addition to the AIC, eqn. (100), there is the *Minimum Descriptive Length* (MDL)<sup>59</sup>

$$\text{MDL}(k) = -Nk \log \left( \frac{\prod_{i=0}^{k-1} \lambda_i^{1/k}}{\sum_{i=0}^{k-1} \lambda_i/l} \right) + \frac{1}{2}(M+1-k)(M+1+l) \log(N) \quad (118)$$

If the eigenvalues are sorted into ascending order, ie. noise values first, then the number of  $K$  noise eigenvectors will be the  $k$  value that minimizes  $\text{MDL}(k)$ . In addition, the minimum eigenvalues will also tend to cluster around the same small value, which should be easy to discriminate by inspection. The signal eigenvectors are also orthogonal to the noise eigenvectors, thus one can use an annihilating or elimination filter<sup>57</sup> to discriminate  $e_i \cdot e_j = \delta_{ij}$  for  $(1 \leq L, j \leq M)$ .

Figure 25(a) shows an example of the singular values  $\lambda$  (in descending value as output by the SVD routine) for a sinewave plus noise test case. Note, depending on the routine it may return either the eigenvalues (EV) or the singular values (SV =  $\sqrt{EV}$ ) - as is the case here for the IDL routine. The first 2 values are the signal  $\lambda_s = \sqrt{2}\sigma_s$ , with  $L = 2l$ , twice the number of expected

coherent sinewaves ( $l = 1$ ) because of the  $2N$  forward-backward matrix, while the remaining  $K = M - L$  values are the noise  $\lambda_n = \sigma_n$ .

There are essentially 4 parameters that affect the pmusic spectrum, the size of the SVD matrix  $N_p$  ( $np = nw - nc + 1$ ), the total number of constituent (signal and noise) eigenvectors chosen  $M$  ( $nc$ ) plus the number of signal eigenvectors to neglect  $L$  ( $nl$ ) in the Fourier transform, and the number of reconstructed spectral frequencies  $J$  ( $nf$ ).

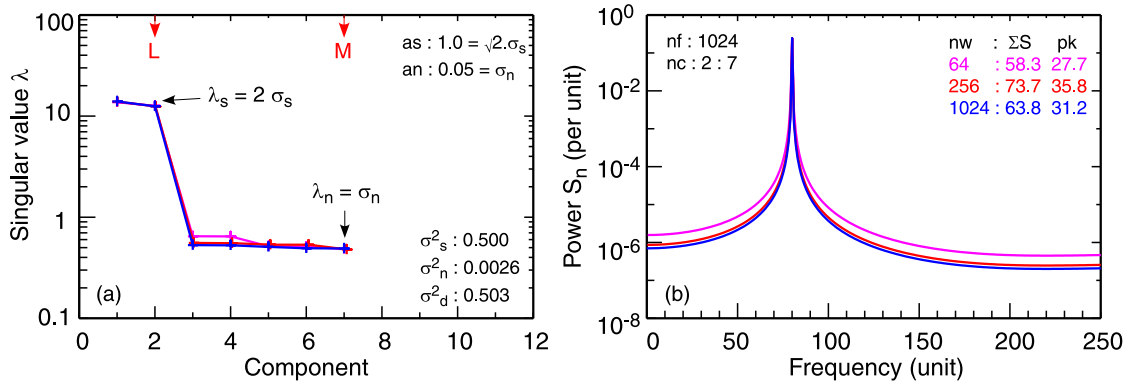


Figure. 25: Example of (a) singular values and (b) normalized pmusic spectra  $S_n(f)$  dependency on SVD matrix size ( $nw$ ) for unity sinewave  $f_0 = 80$  plus Gaussian white noise ( $an = 0.05$ ).

Figure 25 shows normalized pmusic spectra  $S_n$  and their corresponding SV for increasing data window size ( $nw \leq nf$ ), ie. the 2nd dimension of the matrix ( $np = nw - nc + 1$ ) for fixed other parameters. The SV are independent of the pmusic parameters, although both  $\lambda_s$  and  $\lambda_n$  tend to decrease with increasing number of components  $M$  ( $nc$ ).

Scaling the pmusic power spectrum is not straightforward as there are subtle dependencies on the spectral content. Generally the broadband (noise) and coherent peaks do not scale linearly together. To give an indication of the behaviour of the respective noise and signal components the  $\Sigma S$  and peak power  $pk$  are given in the following plots. It is common to normalize the spectral peak intensity to unity. Here, the spectral area are normalized to the data variance  $S_n(f) = S(f) \times \sigma_d^2 / \Sigma S(f)$ , where  $S(f)$  has been scaled by  $(nf^2)$  and  $(np^{0.25})$  - see code. The  $np^{0.25}$  removes much of the matrix size power variation, providing ( $nf \geq nw$ ). With increasing ( $nw$ ) there is a mild impact on sharpening the spectral peak without affecting the peak dynamic range.

As with the BT and AR spectra, it is also important in the eigenvector methods to have sufficient number of reconstructed frequencies  $J$  ( $nf$ ) - as is shown in fig. 26 with increasing ( $nf$ ) with fixed ( $nw$ ) and  $M$  ( $nc$ ). If too few frequencies are used, fig. 26(a), then the peak is not properly resolved and the overall spectrum is poor. If the peak is sufficiently resolved, fig. 26(b), then increasing  $J$  ( $nf$ ) further has little impact and the overall spectrum remains robust. In the examples shown  $J$  ( $nf$ ) covers the full  $\pm f_{max}$  range set by the sample period. However, in practice the reconstructed  $f$  can be concentrated in the range of interest.

The number of reconstructed eigenvalues on the other hand has a stronger impact on the spectrum. Figure 27 shows that increasing  $M$  ( $nc$ ) with  $L = 2$  ( $nl$ ) (one sinewave) reduces the spectral power  $\Sigma S$  as a whole. The spectral peak dynamic range remains roughly constant, but the noise power redistributes to make a flatter background level. In this example a large ( $nf$ ) relative to

(nw) was chosen to highlight the peak base narrowing. In short, increasing  $N_p$  (nw) and  $M$  (nc) to improve resolution (at the expense of longer data sequences) is only useful if  $J$  (nf) is large enough to exploit the resolution. Note, in the case of fig. 27(a) there is only 1 noise component, which is equivalent to the Pisarenko spectrum.

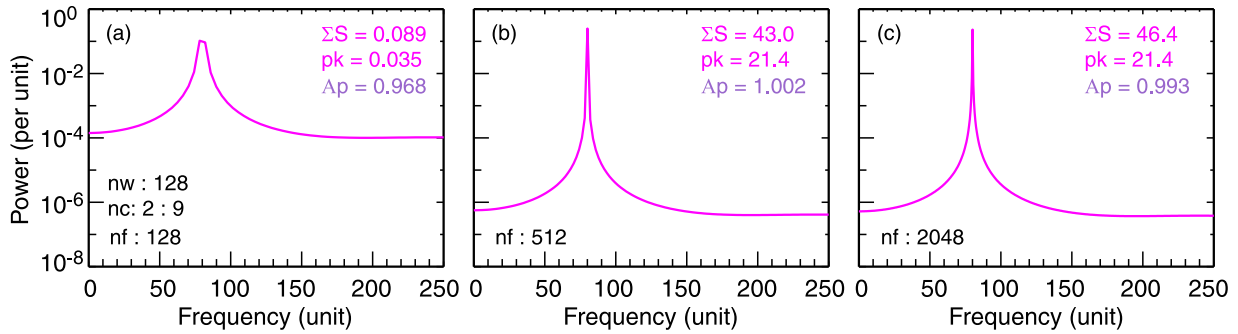


Figure. 26: Effect on pmusic spectrum of increasing number of reconstructed frequencies (nf) for fixed matrix (nw = 128) and (nc = 2 : 9) for a sine wave with (an = 0.05) Gaussian noise.

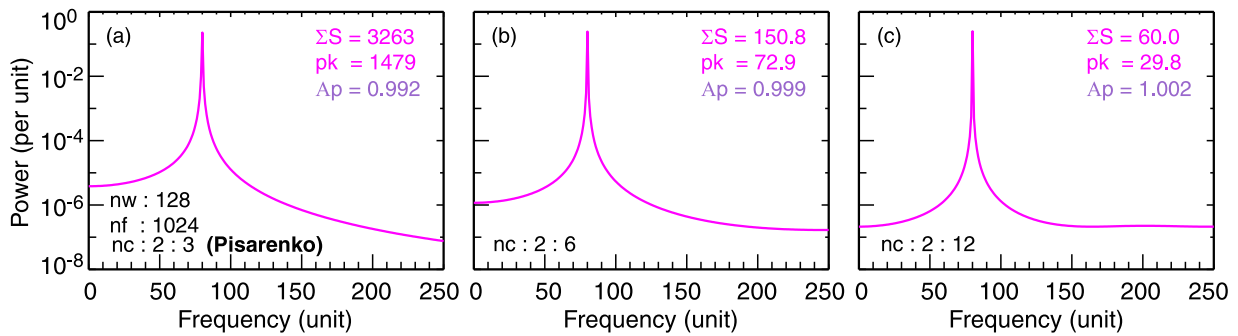


Figure. 27: Effect on pmusic spectrum of increasing number of SVD components / eigenvectors (nc) for fixed (nl = 2), (nf = 1024) and (nw = 128), for a sine plus (an = 0.05) Gaussian noise.

The MUSIC method is exceptionally good at recovering coherent modes from background noise. Figure 28 shows that, as expected, increasing the noise level  $\sigma_n^2$  - cf. noise  $\lambda_n$  - relative to the coherent signal, decreases the spectral peak dynamic range (although the overall  $\Sigma S$  falls). At high  $\sigma_n/\sigma_s$ , the rising background, and associated broadened peak base, begins to swallow-up the peak maxima, reducing the peak resolution. Nevertheless, the pmusic retains its advantage over comparable FFT, particularly for short data lengths - as seen in fig. 25.

Finally, the MUSIC method has also been extended to non-stationary data - the so-called time-frequency MUSIC spectrum.<sup>67</sup> In the case of non-linear signals or sporadic impulsive noise bursts, various derivatives have also been developed, such as ROC-MUSIC (Robust covariance based MUSIC).<sup>68 69</sup>

<sup>67</sup> Belouchrani A. et al *Time-frequency MUSIC*, IEEE Signal Proc. Lett. **6** (5) 109 (1999).

<sup>68</sup> Tsakalides P. *The robust covariation-based MUSIC (ROC-MUSIC) algorithm for bearing estimation in impulsive noise environments*, IEEE Trans. Sig. Proc. **44** (7) 1623 (1996).

<sup>69</sup> Hmidat A.M. et al *Robust multiple signal classification algorithm based on the myriad covariation matrix*, IEE Proc. Vis. Image Signal Proc. **153** (5) 569 (2006).

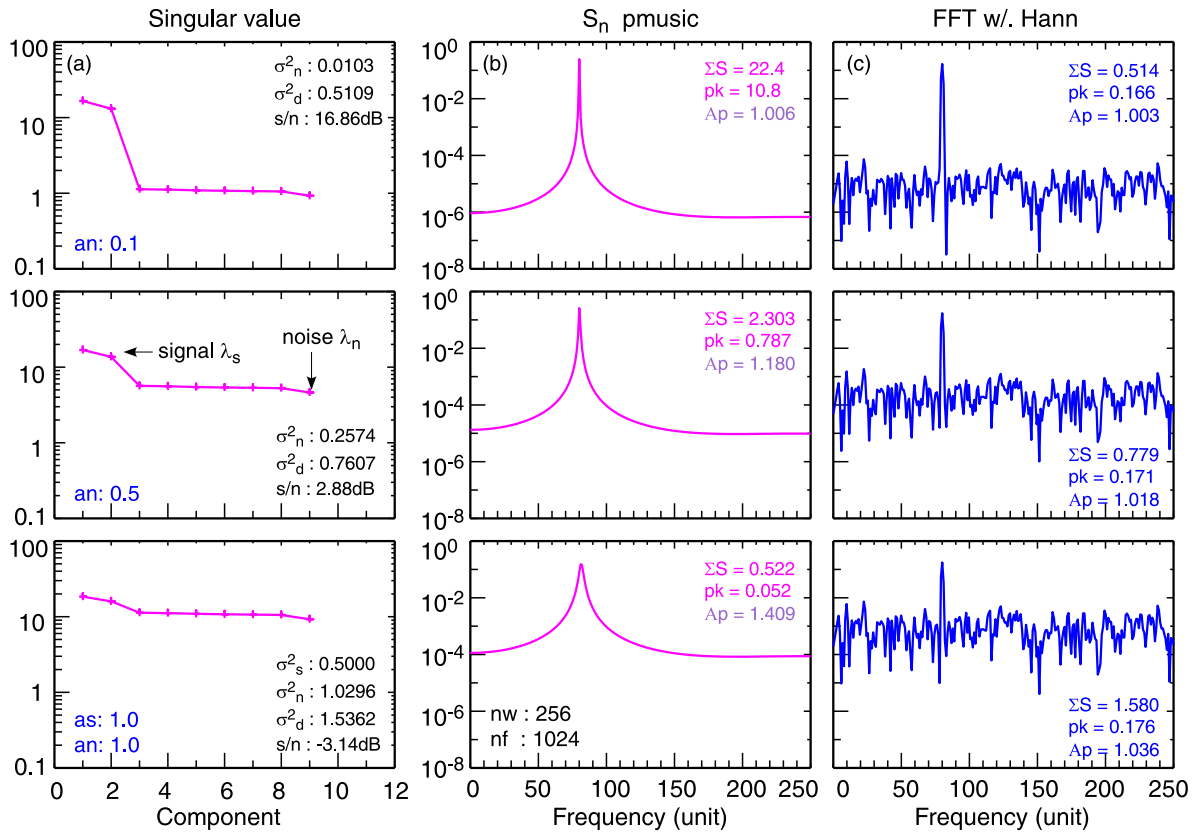


Figure. 28: (a) Eigenvalue, (b) pmusic spectra and (c) FFT (with Hann window) for sine wave (as = 1.0) with increasing Gaussian white noise (an = 0.1, 0.5&1.0) for fixed (nw = 256), (nf = 1024), (nl = 2) and (nc = 9). pmusic spectra scaled by area to data variance  $S(f) \times \sigma_d^2 / \Sigma S(f)$ .

### 7.2.3 ESPRIT method

ESPRIT (Estimation of Signal Parameters via Rotational Invariance Technique) is also a subspace method using eigenvector decomposition, with its roots in array processing and direction finding. It is somewhat similar to the MUSIC method, except it uses the signal subspace rather than the noise subspace.<sup>70 71</sup> The eigensystem is also similar, except some additional constraints are imposed on the input data allowing a significant reduction in computational cost.<sup>70</sup> Further details on the methodology and example algorithms are given in the above references.

A final comment, it should be clear that non-linear spectral methods are not “out of the box” but require (compared to the FFT) more effort in appropriate parameter selection, with usually some trial and error testing to obtain the desired result. Nevertheless, the potential benefits and improved performance are worth the effort. Concerning the spectral estimator errors. While it is possible to obtain confidence limits analytically for linear and for AR estimators, this is rather more complicated for eigenvector estimators.

<sup>70</sup> Roy R.H. & Kailath T *ESPRIT — Estimation of signal parameters via rotational invariance techniques*, Opt. Eng. **29** (4) 296 (1990).

<sup>71</sup> Potts D. et al *Efficient spectral estimation by MUSIC and ESPRIT with application to sparse FFT*, Front. Appl. Math. Stat. **2** Art.1 (2016).

## 8 Frequency-wavenumber spectra

These techniques are used when simultaneous time series data are available from two or more spatially separated channels or measurement points - from which a frequency-wavenumber spectral density  $S(\omega, k)$  can be computed.  $S(\omega, k)$  graphically displays the regions and extents of wave excitation simultaneously in the  $k$  and  $\omega$  domains. From  $S(\omega, k)$  the  $k$ -averaged frequency  $S(\omega)$ , the corresponding  $\omega$ -averaged wavenumber  $S(k)$  spectra, and the dispersion relation  $k(\omega)$  are obtained. There are several approaches, each with their respective pros and cons.

### 8.1 Double transform method $S(\omega, k)$

If many spatial measurement points are available, eg. from probe arrays or scanable probe pairs, in one or more directions then, depending on the spatial sample density, a simple double transform maybe applicable. The procedure is straightforward: For pairs of measurement separations,  $\xi = z_2 - z_1$ , the temporal transforms:  $x(t, z_1) \rightleftharpoons X_1(\omega)$  and  $x(t, z_2) \rightleftharpoons X_2(\omega)$  are made, from which the real and imaginary cross-power components (Co & Quad spectra)  $C(\omega, \xi)$  and  $Q(\omega, \xi)$  are computed. This is repeated for all available spatial separations  $\xi$ .<sup>72</sup> A second round of spatial Fourier transforms of  $C(\omega, \xi)$  and  $Q(\omega, \xi)$  for all values of  $\xi$  for fixed  $\omega$  are needed to give the final frequency-wavenumber spectrum

$$S(\omega, k) = 2 \int_0^{\infty} C(\omega, \xi) \cos(k\xi) + Q(\omega, \xi) \sin(k\xi) d\xi. \quad (119)$$

Depending on the number of spatial points available the spatial FT can be calculated using the trapezium rule or, if appropriate, with a discrete Fourier transform (DFT). As with the temporal FT, the spatial FT is also subject to the usual issues, such as aliasing, resolution and leakage effects.<sup>36</sup> Consequently windowing in both time and space domains maybe desirable.

### 8.2 Multichannel AR

Spatial or array processing techniques have a long history in sonar, radar, seismic and direction finding. In addition to linear methods, the  $S(f, k)$  can be computed from 2D or 3D receiver arrays using auto-regression techniques. The AR models described in section 7.1 were for scalar or univariate data, but they can be extended to vector, ie. multichannel or multivariate data.<sup>58</sup> Here, the Maximum entropy (ME) method is particularly suitable.<sup>73</sup> For example, expressing the temporal data as a vector of spatial points  $x(t, \mathbf{z}_i)$ , for a uniform spaced line array one has

$$\hat{S}(f, \mathbf{k}) = 1 / \sum_{i,j=1}^K \lambda_{ij} \exp[i 2\pi \mathbf{k} \cdot (\mathbf{z}_i - \mathbf{z}_j)] \quad (120)$$

where  $\lambda_{ij}$  are solutions to the AR temporal model. For implementation details see reference<sup>73</sup>.

<sup>72</sup> Ilić D.B & Haker K.J. *Evaluation of plasma-wave spectral density from cross-power spectra*, Rev. Sci. Instrum. **46** (9) 1197 (1975), and Rev. Sci. Instrum. **45** 1315 (1974).

<sup>73</sup> McDonough R.N. *Chapter 6: Application of maximum-likelihood method and maximum entropy method to array processing*, in *Nonlinear methods of spectral analysis*, 2nd edn. Springer (1983).



Maximum likelihood and Levinson methods are also usable. Indeed, the **generalized Burg algorithm**, or LWR algorithm, was originally developed for multi-point or phased array detectors for direction finding. The approach is to build a matrix of AR coefficients (order and channel) from the vector data (time and channel) and calculate the respective cross and auto spectra directly, from which coherence and cross-phase spectra can be computed.<sup>74 75</sup>

As noted previously, another alternative is the eigenvector **ESPRIT model** - section 7.2.3.

## 8.3 Two point correlation methods

### 8.3.1 Narrow-band covariance

If the spatial data is more limited, ie. there are insufficient points to cover at least one spatial wavelength of the lowest  $k$  resolvable, then a range of so-called two fixed point techniques are available. The approach of Iwama<sup>76</sup> describes a method to estimate the mean wavenumber  $\bar{k}$  and spectral width  $\Delta k$  of narrow-band-pass filtered data from the complex covariance at zero time lag for a fixed  $\xi$  spatial separation:  $\text{cov}(0, \xi) = \text{Re cov}(0, \xi) + \imath \text{Im cov}(0, \xi)$ . For a real signal  $x_n$  the covariance would be computed from the analytic signal  $(x_n + \imath \mathcal{H}\{x_n\})$  using the Hilbert transform to obtain the imaginary part - see section 10.7. From the pass-band filtered covariance the first and second moments  $\hat{\mu}_1$  and  $\hat{\mu}_2$  are taken, resulting in

$$\hat{\mu}_1 \equiv -\xi^{-1} \tan^{-1} \left( \frac{\text{Im cov}}{\text{Re cov}} \right) = k_o \quad (121)$$

$$\sqrt{\hat{\mu}_2 - \hat{\mu}_1^2} \equiv \frac{\sqrt{2}}{\xi} \left( 1 - \frac{(\text{Re cov})^2 + (\text{Im cov})^2}{\sigma^2 - \sigma_n^2} \right)^{1/2} = \hat{\Delta}k \quad (122)$$

where  $\sigma_n^2$  is the variance of the background noise.<sup>77</sup> Simply scanning the band-pass filter frequency gives the  $k(\omega)$  dispersion and  $\Delta k(\omega)$  growth-rate spectra using eqns. (121) and (122).

### 8.3.2 Cross-phase spectra

In the very narrow filter pass band limit, i.e. single frequency, the covariance estimate becomes equivalent to the cross spectrum estimate and the  $\hat{\mu}_1$  and  $\hat{\Delta}$  estimators can be rewritten as:<sup>78</sup>

$$\hat{\mu}_1 \rightarrow \hat{k}(\omega) = -\frac{1}{\xi} \tan^{-1} \left( \frac{\langle \hat{Q}(\omega) \rangle}{\langle \hat{C}(\omega) \rangle} \right) \quad \text{or} \quad \hat{k}(\omega) = \frac{\hat{\phi}(\omega)}{\xi} \quad (123)$$

<sup>74</sup> Morf M. et al *Recursive multichannel maximum entropy spectral estimation*, IEEE Trans. Geosci. Electron. **GE-16** (2) 85 (1978).

<sup>75</sup> Marple S.L. *Performance of multichannel autoregressive spectral estimators*, IEEE Intl. Conf. Acoustics, Speech, & Signal Proc. (Tokyo), **ICASSP 86** 197 (1986).

<sup>76</sup> Iwama N. & Tsukishima T. *A correlation method for estimating wave-number spectrum moments of fluctuations*, Appl. Phys. Lett. **31** (11) 783 (1977).

<sup>77</sup> Iwama N. et al *Estimation of wave-number spectrum parameters from fixed probe-pair data*, J. Appl. Phys. **50** (5) 3197 (1979).

<sup>78</sup> Beall J.M. et al *Estimation of wavenumber and frequency spectra using fixed probe pairs*, J. Appl. Phys. **55** (6) 3933 (1982).

$$\hat{\Delta}k \rightarrow \hat{\delta}_k(\omega) = \frac{\sqrt{2}}{\xi} (1 - \hat{\gamma}(\omega))^{1/2}. \quad (124)$$

where  $\phi(\omega)$  is the cross-phase spectrum. These equations represent the statistical dispersion relation and spectral width (inverse growth-rate) respectively. From eqn. (123) a corresponding frequency averaged mean wavenumber can be defined as the cross-power weighted average

$$\bar{k} = \frac{\sum k(\omega) S_{xy}(\omega)}{\sum S_{xy}(\omega)}. \quad (125)$$

#### 8.4 Local frequency-wavenumber spectra $S_l(\omega, k)$

The frequency-wavenumber technique of Beall <sup>78</sup> also uses the familiar cross-spectral estimators. However, its basis and approach are different. Starting from the concept that turbulent fluctuations can be modelled as a superposition of *wave-packets* of temporal and spatial sinusoids centered on single frequencies and wavenumbers, the method constructs a *local* frequency-wavenumber power density spectrum,  $S_l(\omega, k)$  as a histogram by sampling the individual wave-packet power concentrations via the  $k(\omega)$  spectrum (from the cross-phase spectrum between two fixed spatial points) in conjunction with  $S(\omega)$  for successive increments of  $\Delta k$  and  $\Delta\omega$  (see code snippet below).

In practice  $S_l(\omega, k)$  is formed by summing the power of the  $S(\omega)$  for a selected discrete frequency from data sets whose corresponding wavenumber spectrum  $k(\omega)$  lies within a specified sample range  $k$  to  $k + \Delta k$

$$S_l(\omega, k) = \frac{1}{M} \sum_{m=1}^M S(\omega)_m I(k - k(\omega)_m) \quad (126)$$

where  $I(\cdot)$  is an indicator function

$$I(y) = \begin{cases} 1 & \text{for } 0 < y < \Delta k \\ 0 & \text{otherwise} \end{cases} \quad (127)$$

The ensemble averaging diminishes the random noise and incoherent fluctuation power, leaving structure and peaks in the regions of coherent oscillations. Note a successful implementation relies strongly on the choice of the spatial point separation,  $\Delta k_{\min} = 2\pi/\xi$ , as well as the sample length  $T(= N\Delta t)$  of the individual data sets used to compute the  $S(\omega)_m$ , and  $k(\omega)_m$  spectra, as well as the total number  $M$  of spectra in the ensemble. <sup>79</sup>

The spectral moments of  $S_l(\omega, k)$  can then be computed to give the statistical dispersion relation

$$\bar{k}(\omega) = \sum_{m=1}^M k_m \frac{S_l(\omega, k_m)}{S(\omega)} \quad (128)$$

---

<sup>79</sup> Carlson A. et al *The limitations of the local wavenumber*, Proc. EPS Conf. (Amsterdam) ECA **14B**, 1676 (1990).

and wavenumber spectral width

$$\bar{\delta}_k(\omega) = \sigma_k^2(\omega) = \sum_{m=1}^M (k_m - \bar{k}(\omega))^2 \frac{S_l(\omega, k_m)}{S(\omega)} \quad (129)$$

where  $k_m = m\Delta k$  and  $M = \pi/\xi\Delta k$  with  $\xi$  the spatial separation of the measurement points.

Figure 29 shows an example  $S_l(\omega, k)$  spectrum (contour plot and 3D-perspective plots) obtained from a simple time delayed noise signal to simulate the spatial propagation of a broad-band fluctuation, such as  $x(t, r) = \sum_i h_i \cos(\omega_i t + k_i \xi + \phi)$  where  $k_i = v_{ph}/\omega_i$ . Note in this example the spectral amplitude has been artificially shaped by  $h_i$  (*hpf* in code snippet) for visual effect.

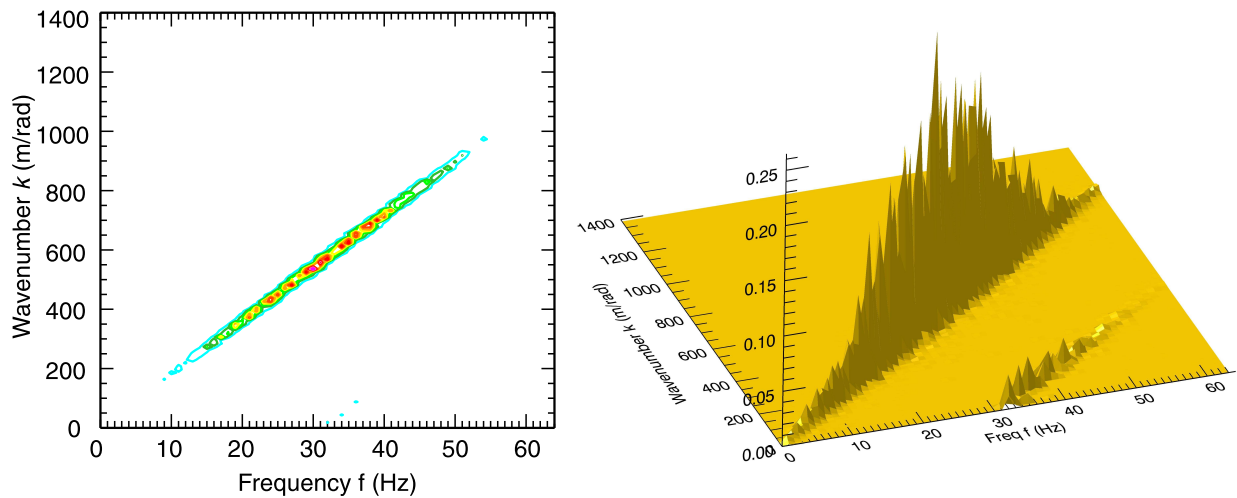


Figure. 29: Example of local  $S_l(\omega, k)$  spectrum using time shifted noise signals with shaped power spectra. Left: contour and, right: perspective plots.

The respective  $S(\omega)$ ,  $k(\omega)$  and  $\delta k(\omega)$  spectra are shown in fig. 30. Note the effect of aliasing in  $k$  in fig. 29(b). As shown in the  $\delta k(f)$  spectrum in fig. 30(c) in this simple example the width only becomes non-negligible at frequencies where the effect of aliasing occurs. Normally, in real data the spectral width  $\delta k$  would indicate spectral spreading or a mode growth rate.

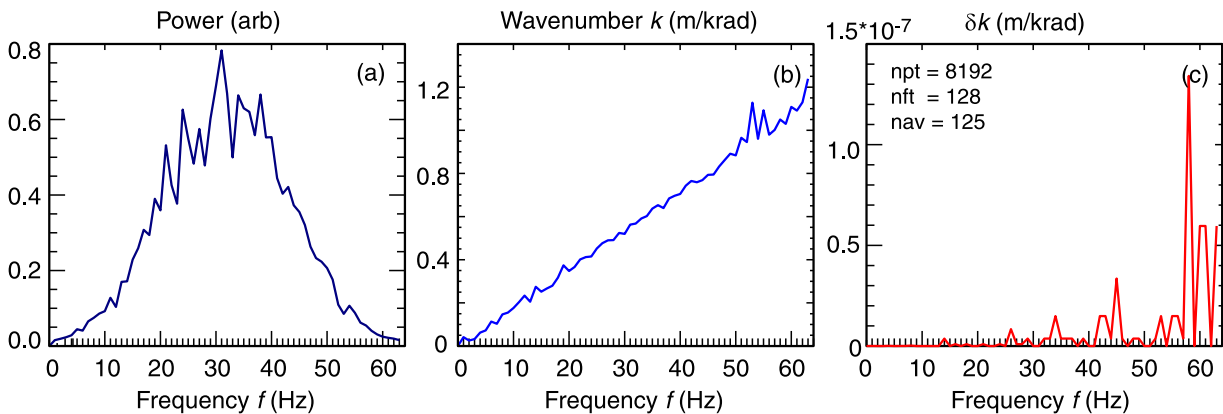


Figure. 30: Example of (a)  $S(\omega)$ , (b)  $k(\omega)$  and (c)  $\delta k(\omega)$  spectra from cross-phase estimates of time delayed noise signals with shaped spectra.

```

; Generate signals. use shift to simulate k dispersion .
npt = 8192 & nft = 128 & nf2 = nft/2 & nkp = 128 & kmax = 1400
sig1 = 2.*(RandomU(seed, npt)-0.4999)
sig2 = Shift(sig1, 4) + 0.05*RandomN(seed, npt) ; delay -> k prop.
hpf = Exp(-(Findgen(nf2+1)-32)^2/(16^2)) ; Shape spectrum
; Define local k-spec params.
delk = kmax/nkp *1.1 ; width of k bin
kt = Findgen(nkp)*kmax/nkp
Sloc = Fltarr(nf2,nkp)
; Loop over data
i1 = 0
WHILE i1 LT npt-nft DO BEGIN
  i2 = i1+nft-1
  f1 = FFT(Complex(sig1[i1:i2] - Mean(sig1[i1:i2]), 0.), -1)
  f2 = FFT(Complex(sig2[i1:i2] - Mean(sig2[i1:i2]), 0.), -1)
  sm = ((Conj(f1[0:nf2])*f1[0:nf2])+(Conj(f2[0:nf2])*f2[0:nf2]))*hpf
  Cc = Real_Part(f1)*Real_Part(f2) + Imaginary(f1)*Imaginary(f2)
  Qc = Real_Part(f1)*Imaginary(f2) - Real_Part(f2)*Imaginary(f1)
  phase = -Atan(Qc[0:nf2], Cc[0:nf2]) ; 1st half of FFT
; Unwrap phase
nn = N_Elements(phase)
thresh = !pi *1.15 ; del-phase threshold
jumpsPos = Where(phase - Shift(phase,1) GT thresh)
jumpsNeg = Where(phase - Shift(phase,1) LT -thresh)
d = phase[0L:nn-2L] - phase[1L:nn-1L]
p = 2*!pi * (Float((d GT thresh) GT 0) -Float((d LT -thresh) GT 0))
r = Total(p, /Cumulative)
phase[1:*] += r
kspec = phase/(eta2-eta1)
; Build S-local spectral histogram
FOR j = 0, nkp-1 DO BEGIN
  FOR i = 0, nft/2-1 DO BEGIN
    IF (kt[j]-kspec[i]) GT 0. AND (kt[j]-kspec[i]) LT delk THEN $$
      Sloc[i,j] = Sloc[i,j] + Sm[i]
    ENDFOR
  ENDFOR
  i1 = i1 + nft/2
ENDWHILE
; Average weighted k and delta-k spectra
kbar = Fltarr(nf2)
dkbar = Fltarr(nf2)
sbar = Total(sloc,2)
FOR i = 0, nkp-1 DO BEGIN
  kbar = kbar + kt[*] * Sloc[* ,i]/Sbar[*]
  dkbar = dkbar + (kt[*]-kbar[*])^2 * Sloc[* ,i]/Sbar[*]
ENDFOR

```

*Example. 18 : Code snippet (IDL) to local  $S_l(\omega, k)$  spectrum. Note the parameter hpf in the sm calculation is used purely for artifice to give shape to the  $\omega - k$  ridge and should be removed in practice - cf. fig. 29 & 30.*

## 8.5 Space-time correlations

### 8.5.1 Spectral line shape

Sharp spectral peaks are often associated with coherent mode activity with a finite  $k_o$  and  $\omega_o$ . The shape and width of the spectral peak can give information on the underlying mode/turbulence structure, such as the linear growth rate,  $\Delta\omega \sim \gamma_l$ .<sup>80</sup> Fitting spectral peaks can be tricky and  $\Delta\omega$  is often better approached in the correlation domain since the peak width and autocorrelation decay time are related  $\Delta\omega \sim 1/\tau_a$ . The autocorrelation may also display mixtures of Lorentzian  $\exp(-\alpha|\tau|)$  or Gaussian  $\exp(-\alpha^2\tau^2)$  forms, where  $\alpha$  is dominated by the slowest decay process. If multi-point data is available then the spatial-temporal autocorrelation decay envelope can be fitted and modelled, eg.  $\rho(\tau, \xi) = \rho_o \exp(-\alpha|\tau|) \exp(-\beta|\xi|)$ . Further, if the data are narrow band-pass filtered prior to correlation then the dispersion relation  $\bar{k}(\omega_f)$  and spatial and temporal growth rates  $\alpha(\omega_f)$  and  $\beta(\omega_f)$  can be extracted as a function of filter centre frequency  $\omega_f$ .<sup>81</sup>

### 8.5.2 Correlation phase velocity

Atmospheric and ionospheric radar studies have been a major source of techniques for measuring the propagation of turbulent structures. Briggs,<sup>82</sup> using cross-correlation techniques developed an analysis methodology to distinguish between the propagation velocity of a spatial density fluctuation and its self-mutation rate. Figure 31 shows schematic 2D contour plots of the cross-correlation coefficient  $\rho$  vs time lag  $\tau$  and spatial separation  $\xi$ .

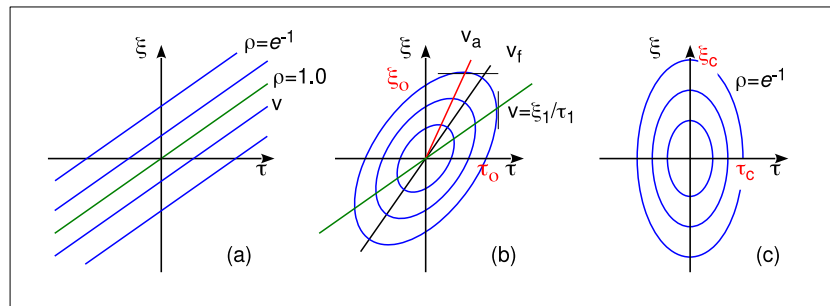


Figure. 31: Spatial-temporal correlation contours (a) purely propagating, (b) propagating and mutating, and (c) purely mutating.

For a purely propagating mode or (narrow band-pass filtered) turbulence, the  $\rho$  contours are inclined lines, fig. 31(a), with  $v = \xi/\tau$ , while for a purely mutating structure the contours are closed and symmetric, fig. 31(c). For small separations the contours are roughly elliptic and  $\rho(\tau, \xi) = \rho_o \exp(-\tau^2/\tau_c^2 - \xi^2/\xi_c^2)$  with  $\xi_c$  and  $\tau_c$  the characteristic lifetimes, cf. Briggs.<sup>83</sup>

<sup>80</sup> Edwards D.N. & Rusbridge M.G. *The experimental determination of fluctuation line-profiles in a weakly turbulent plasma*, Phys. Lett. **86A** (8) 421 (1981).

<sup>81</sup> Conway G.D. & Elliott J. *Digital signal processing techniques for plasma dispersion curve measurements*, J. Phys. E: Sci. Instrum. **20** 1341 (1987).

<sup>82</sup> Briggs B.H. et al *The analysis of observations on spaced receivers of the fading of radio signals*, Proc. Phys. Soc. B, **63** 106 (1950).

<sup>83</sup> Briggs B.H. et al *On the analysis of moving patterns in geophysics - I. Correlation analysis*, J. Atmos. Terr. Phys. **30** 1777 (1986).

More complex forms for  $\rho(\tau, \xi)$  have also been modelled.<sup>84</sup> With a propagation velocity  $v$  the ellipses become inclined with  $\rho(\tau, \xi, ) = \rho_o \exp(-\tau^2/\tau_c^2 - (\xi - v\tau)^2/\xi_c^2)$ . However, due to the mutation rate the ellipse axis is not aligned with  $v$  but with an apparent fading velocity  $v_f$ , where  $\rho(0, \xi_o) = \rho(\tau_o, 0)$ .

There are several approaches to separating the various velocities, the simplest are illustrated in fig. 32. For a reasonably small  $\xi_e$  separation the true propagation velocity is given by the temporal intercept  $\tau_e$  of the auto and cross-correlation curves where  $\rho(\tau_e, 0) = \rho(\tau_e, \xi_e)$  with

$$v = \xi_e/2\tau_e. \quad (130)$$

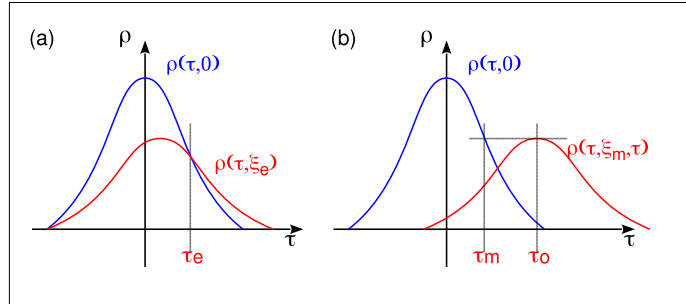


Figure. 32: Auto-cross correlation intercept.

On the other hand, taking the cross-correlation peak delay  $\tau_o$  for some arbitrary separation  $\xi_m$  gives the characteristic  $v_c = \xi_m/\tau_o$ , ie. the speed of mutation. Returning to physics, He et al.<sup>85</sup> gives an interpretation of the space-time correlations in terms of turbulent flow dynamics and couplings. The above temporal correlation methodology can be equally applied to the frequency domain.<sup>86</sup>

### 8.5.3 Statistics of the auto-correlation

This is a method proposed to extract turbulent velocity fluctuations from the single-point auto-correlation function.<sup>87</sup> The model is as follows: assume the turbulent structures have a Gaussian shape in space and time, propagating past the measurement point. Then the characteristic auto-correlation time is given by

$$\tau_c = \frac{\tau_d \tau_v}{\sqrt{\tau_d^2 + \tau_v^2}} \quad (131)$$

where  $\tau_d$  is the structure life-time or decay time, and  $\tau_v = w/v$  is a velocity dependent propagation time with  $w$  the structure size or spatial correlation length and  $v$  the propagation velocity.

1. If  $\tau_d \gg \tau_v$  then  $\tau_c \approx w/v$ . If  $w$  is constant in time then one can measure the time dependent velocity  $v(t)$  and hence its spectrum.
2. If  $\tau_d \ll \tau_v$  then  $\tau_c$  is dominated by structure decay rate.

Of course, some pre-knowledge or estimate of the structure size  $w$ , ie.  $\tau_d$  is required!

<sup>84</sup> Holloway C.L. et al *Cross correlations and cross spectra for spaced antenna wind profilers: 2. Algorithms to estimate wind and turbulence*, Radio Sci. **32** (3) 967-982 (1997).

<sup>85</sup> He G. et al *Space-time correlations and dynamic coupling in turbulent flows*, Annu. Rev. Fluid Mech. **49** 51-71 (2017).

<sup>86</sup> Briggs B.H. & Vincent R.A. *Spaced-antenna analysis in the frequency domain*, Radio Sci. **27** (2) 117 (1992).

<sup>87</sup> Bencze A. & Zoletnik S. *Autocorrelation analysis and statistical consideration for the determination of velocity fluctuations*, Phys. Plasmas **12** 052323 (2005).

## 8.6 Biorthogonal decomposition

Biorthogonal decomposition (BD) is an objective test for identifying and quantifying large-scale coherent structures - with *an priori* assumption of their shape or localization.<sup>88</sup>

Consider a set of signals  $y(t, r)$  measured at  $N$  time points and  $M$  locations. Now assemble the signals into an  $N \times M$  matrix. The BD expands the discrete data  $Y_{ij} = y(t_i, r_j)$  into unique set of orthogonal (ie. independent) modes in time and space:

$$Y_{ij} = \sum_{n=1}^K A_n v_n(t_i) u_n(r_j) \quad (132)$$

where  $K = \min(N, M)$  is the finite global dimension of the data set. The base functions  $u_n(r_j)$  and  $v_n(t_i)$  are eigensolutions of the matrix  $\mathbf{Y}$ , and  $A_n$  are ‘weights’ or eigenvalues (positive, negative or zero), which can be obtained via usual singular value decomposition (SVD) techniques.

Note that there is a direct link between the spatial and temporal modes, ie. a dispersion relation

$$\mathbf{Y}u_n = A_n v_n. \quad (133)$$

Concerning interpretation, highly correlated structures in space and time appear as large weighted components  $A_n$ , while traveling waves appear as pairs of equal weighted components, and random noise has roughly equal weights  $A_1 \approx A_2 \approx \dots \approx A_K$ . This topic is discussed in more detail by van Milligen.<sup>89</sup>

A restriction of the BD is its focus on long-lived, large scale structures. One solution to shorter scales is the windowed BD (WBD)<sup>90</sup> which applies an adaptable temporal-spatial window to the data  $y(t, r)$  before decomposition

$$W_{\Delta t, \Delta r}(t, r) = \begin{cases} 1, & (t_o - \Delta t) < t < (t_o + \Delta t), (r_o - \Delta r) < r < (r_o + \Delta r) \\ 0, & \text{else} \end{cases} \quad (134)$$

where  $\Delta t$  and  $\Delta r$  are temporal/spatial window half-widths, and  $(t_o, r_o)$  is the window centre. Another useful parameter is a coherence factor  $S_{\Delta t, \Delta r}(t_o, r_o) = \left( \sum_{i=1}^{k-1} (A_i - A_{i-1}/(k-1)) \right)^{1/2}$  which ranges between 0 and 1 and is a measure of the coherence of the data set.<sup>90</sup> Finally, the BD has been extended and applied to 2D  $(t, r, z)$  temporal-spatial data.<sup>91</sup>

<sup>88</sup> Benkadda S. *Characterization of coherent structures in tokamak edge turbulence*, Phys. Rev. Lett. **73** (25) 3403 (1994).

<sup>89</sup> van Milligen, B.Ph. *The use of the biorthogonal decomposition for the identification of zonal flows at TJ-II*, Plasma Phys. Control. Fusion **57** 025005 (2015).

<sup>90</sup> Dong L. et al *Identification of short-live and localized coherent structures in plasma turbulence by window biorthogonal decomposition*, Rev. Sci. Instrum. **74** (12) 5093 (2003).

<sup>91</sup> Fenzi C. et al *2D turbulence imaging in DIII-D via beam emission spectroscopy*, Rev. Sci. Instrum. **72** (1) 988 (2001).

## 9 Higher order spectra

### 9.1 Bispectrum and bi-coherence

Bispectral analysis is a well established technique for identifying nonlinear processes in turbulent systems, cf. Collis<sup>92</sup>. Here the auto- and cross-bicoherence spectra (ie. third-order correlations) can directly identify 3-wave couplings, for example in plasma drift-wave turbulence,<sup>93 94 95 96</sup>

The general cross-bispectrum is given by

$$\hat{B}_{xyz}(\omega_1, \omega_2) = \lim_{T \rightarrow \infty} \frac{1}{T} E[X(\omega_1) Y(\omega_2) Z^*(\omega_1 + \omega_2)] \quad (135)$$

$$= \langle X(\omega_1) Y(\omega_2) Z^*(\omega_3) \rangle \quad (136)$$

where  $E[\cdot]$  is the expectation operator,  $\langle \cdot \rangle$  the ensemble averaging and  $X(\omega)$ ,  $Y(\omega)$  and  $Z(\omega)$  are the Fourier components of signals  $x(t)$ ,  $y(t)$  and  $z(t)$  at  $\omega_1$  &  $\omega_2$  with  $\omega_3 = \omega_1 \pm \omega_2$ . The squared bicoherence and the bi-phase are defined in terms of the bispectrum<sup>97</sup> as

$$b_{xyz}^2(\omega_1, \omega_2) = \lim_{T \rightarrow \infty} \frac{1}{T} \frac{|B(\omega_1, \omega_2)|^2}{S(\omega_1) S(\omega_2) S(\omega_1 + \omega_2)} \quad (137)$$

$$= \frac{|\hat{B}_{xyz}(\omega_1, \omega_2)|^2}{\langle X(\omega_1) Y(\omega_2) \rangle \langle Z(\omega_3)^2 \rangle} \quad (138)$$

$$\theta_{xyz}(\omega_1, \omega_2) = \tan^{-1} \left( \frac{\text{Im } \hat{B}_{xyz}(\omega_1, \omega_2)}{\text{Re } \hat{B}_{xyz}(\omega_1, \omega_2)} \right), \quad (139)$$

where  $b^2$  ranges from 0 to 1. For the auto-bispectra,  $X = Y = Z$ .

The total bicoherence is the  $b^2$  integrated over all  $\omega_1$  and  $\omega_2$  frequency space

$$\overline{b^2}_{tot} = \frac{1}{N} \sum_{\omega_1, \omega_2}^N b^2(\omega_1, \omega_2) \quad (140)$$

$\overline{b^2}_{tot}$  is often quoted as an indicator of the total non-linear activity in the spectrum.

<sup>92</sup> Collis, W.B. et al *Higher-order spectra: The bispectrum and trispectrum*, Mech. Sys. Signal Proc. **12** (3) 375 (1998)

<sup>93</sup> Kim Y.C. & Powers E.J. *Digital bispectral analysis of self-excited fluctuation data*, Phys. Fluids **21** (8) 1452 (1978).

<sup>94</sup> Holland C. et al *Evidence for Reynolds-stress driven shear flows using bispectral analysis: theory and experiment*, Plasma Phys. Control. Fusion **44** A453 (2002).

<sup>95</sup> van Milligen B.Ph. et al *Bicoherence during confinement transitions in the TJ-II stellarator*, Nucl. Fusion **48** 115003 (2008).

<sup>96</sup> van Milligen B.Ph. et al *Spatiotemporal and wavenumber resolved bicoherence at the low to high confinement transition in the TJ-II stellarator*, Nucl. Fusion, **53** 113034 (2013).

<sup>97</sup> Kim & Powers, *Digital bispectral analysis and its applications to non-linear wave interactions*, IEEE Trans. Plasma Sci. **PS-7** 120 (1979).



Another common derivative is the summed bicoherence spectrum at  $\omega_3$  defined as:

$$b^2(\omega_3) = \frac{1}{N_{\omega_3}} \sum_{\omega_3=\omega_1+\omega_2}^{N_{\omega_3}} b^2(\omega_1, \omega_2). \quad (141)$$

being a measure of the spectral coupling to the  $\omega_3$  frequency.

### 9.1.1 Non-linear test signal

Kim & Edwards<sup>98</sup> provide a simple 3-oscillator non-linear coupled test signal

$$\begin{aligned} x(t) = & \cos(\omega_{o1}t + \phi_{o1}) + \cos(\omega_{o2}t + \phi_{o2}) + 1/2 \cos(\omega_{o3}t + \phi_{o1} + \phi_{o2}) \\ & + \cos(\omega_{o1}t + \phi_{o1}) \cos(\omega_{o2}t + \phi_{o2}) + n(t) \end{aligned} \quad (142)$$

where  $\omega_{o1}/\omega_N = 0.220$ ,  $\omega_{o2}/\omega_N = 0.375$ , and  $\omega_{o3} = \omega_{o1} + \omega_{o2}$  are the oscillator frequencies with  $\omega_N$  the Nyquist frequency. Here,  $\phi_{o1}$  and  $\phi_{o2}$  are uniformly  $(-\pi, \pi)$  distributed random phases, and  $n(t)$  is a small level of normally distributed random noise.

Figure 33 shows contour plots of the  $b^2$  bispectrum and biphase for the non-linear test signal. The bispectrum is symmetric about the spectral frequencies  $f_2/f_N = 0$  axis, but cross-phase inverted. The strongest  $b^2 \sim 1$  peak occurs at  $f_1/f_N = 0.375$  and  $f_2/f_N = 0.220$  in the top triangle, with corresponding peaks at  $f_1/f_N = 0.595$  and  $f_2/f_N = -0.220$  &  $-0.375$  in the lower sector. The simple FFT spectra  $P(f)$  shows all 3 peaks, while the  $f_3$  and  $f_1$  spectra at the  $\omega_{o3}$  peak nicely separate the drive and coupled peaks.

### 9.1.2 Statistical error in $b$

The approximate variance of the bicoherence (estimator) is given by Kim & Edwards<sup>98</sup>

$$\text{var}[\hat{b}^2] = (1/M) [1 - b^2(\omega_1, \omega_2)] \quad (143)$$

where  $M$  is the number of segments, ie. FFT ensembles.

---

<sup>98</sup> Kim & Powers, *Digital bispectral analysis and its applications to non-linear wave interactions*, IEEE Trans. Plasma Sci. **PS-7** 120 (1979).

```

; Calc auto-bispectrum - After S.Vaughan (2007)
npt = 8192 & seed1 = 104 & seed2 = 30 & seed3 = 77 & dt = 1.
noise = 0.01 * namp*Reform(RandomN(seed1, npt))
fN = 1./(2.*dt)
fo1 = 0.220*fN & fo2 = 0.375*fN & fo3 = fo1 + fo2
ph1 = (RandomU(seed2, npt)-0.5)*!pi/1.5
ph2 = (RandomU(seed3, npt)-0.5)*!pi/1.5
data = Sin(2.*!pi*Findgen(npt)*fo1*dt +ph1) $
      + Sin(2.*!pi*Findgen(npt)*fo2*dt +ph2) $
      + Sin(2.*!pi*Findgen(npt)*fo3*dt +ph1 + ph2) + noise
; Define arrays
nft = 256
wind = 0.5*(1.0-Cos(2.0*!pi*Findgen(nft)/nft)) ; if req.
df = 1./(Float(nft)*dt)
fr1 = Findgen(nft/2)*df ; X-axis freq1 array
fr2 = (Findgen(nft)-nft/2)*df ; Y-axis freq2 array
Bup = Complexarr(nft,nft) & bs1 = Bup & bs2 = Bup
bsq = Fltarr(nft/2,nft) ; b^2 matrix
Pws = Fltarr(nft) ; Summed power spectrum
; Setup index matrix {j,k} = j+k
indx = Indgen(nft)
i = Rebin(indx, nft, nft, /sample)
j = Transpose(i)
ij = i + j
; Shift (wrap) matrix > nft
cc = Where(ij GT nft)
ij[cc] = ij[cc] - nft
; Calc Auto/Cross-bispectrum
iat = 0L & ilp = 0
WHILE iat LT npt-nft DO BEGIN ; Sum over segments
  ilp = ilp+1
  da1 = data[iat:iat+nft-1] - Mean(data[iat:iat+nft-1])
  fof = FFT(Complex(da1, 0.),-1) ; Without Hann window
  IF N_Elements(data2) GT 0 THEN BEGIN ; Cross-bispec
    da2 = data2[iat:iat+nft-1] - Mean(data2[iat:iat+nft-1])
    fof2 = FFT(Complex(da2, 0.),-1)
  ENDIF
  XiXj = fof # fof ; matrix Xi*Xj & X{i+j}
  Xij = fof[ij]
  IF N_Elements(data2) GT 0 THEN Xij = fof2[ij]
  XiXjXij = XiXj * Conj(Xij) ;triple term Xi*Xj*X{i+j}
  Bup = Temporary(Bup) + XiXjXij
  bs1 = Temporary(bs1) + Abs(XiXj)^2
  bs2 = Temporary(bs2) + Abs(Xij)^2
  Pws = Temporary(Pws) + Abs(fof*Conj(fof))
  iat = iat + nstp
ENDWHILE
; Calc. bi-phase spectrum & shift
bphase = Shift(Atan(Bup[0:nft/2-1,*], /Phase), 0, nft/2)
; Normalize - bicoherence spectrum
Pws = Pws/Float(ilp)
Bup = Abs(Bup[0:nft/2-1,*])^2
bss = (bs1[0:nft/2-1,*]*bs2[0:nft/2-1,*])
mask = Where(bss NE 0.0, cnt)
IF (cnt GT 0) THEN bsq[mask] = Bup[mask]/bss[mask]
bsq[0,0] = 0
; Shift matrix back
bsq = Shift(bsq,0,nft/2)

```

*Example. 19 : Code snippet (IDL) to calculate auto/cross-bispectrum and biphas - cf. fig. 33.*

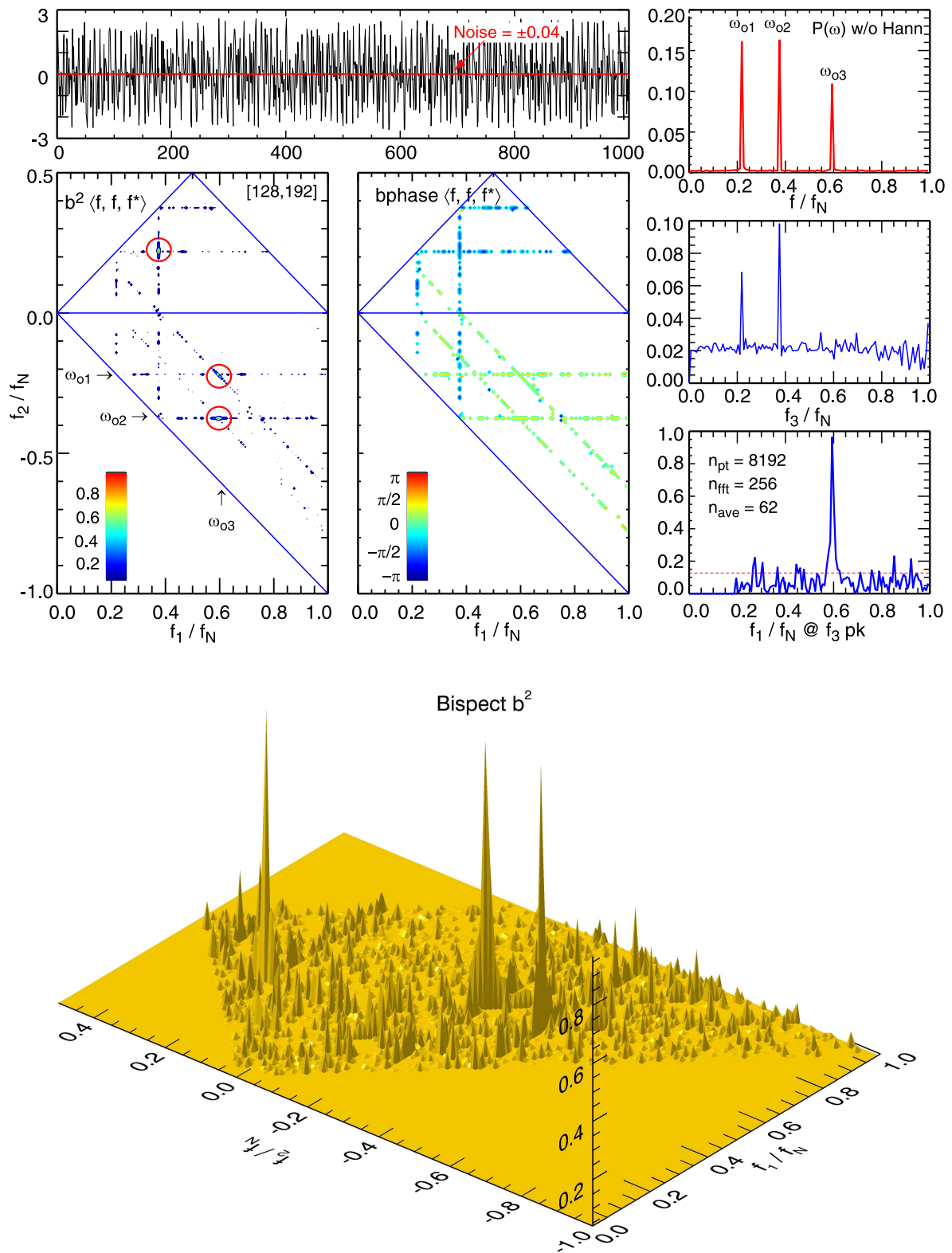


Figure. 33: Top: Bispectrum  $b^2$  and biphase spectra, and bottom: 3D perspective plot of  $b^2$  spectra for couple 3-oscillator test signal of eqn. (142) with added 4% white noise.  $f_N$  is the Nyquist frequency. Note  $f_1$ ,  $f_2$  and  $f_3 = f_1 + f_2$  are axis frequencies while  $\omega_{1,2,3}$  are the mode frequencies.

## 9.2 Triple correlations

The auto/cross-correlation is a 2nd-order statistic (cumulant) of the signal. The triple correlation, however, is a 3rd-order statistic, where one signal is correlated with a two other independent time-shifted signals. The triple correlation is useful for several reasons; firstly, unlike the 2nd order correlation which contains only the symmetric part of the Fourier phase (and thus can only give the amplitude), the triple correlation retains the phase information. The phase information gives evidence of non-linearities.<sup>99 100</sup> In addition, higher-order correlations are notably less sensitive to noise.

$$R_{xyz}(\tau_1, \tau_2) = \int_{-\infty}^{\infty} x(t)^* y(t + \tau_1) z(t + \tau_2) dt \quad (144)$$

In discrete form

$$\begin{aligned} R_{xyz}(k_1, k_2) &= E[x_n^* y_{n+k_1} z_{n+k_2}] \\ &= \frac{1}{N} \sum_{n=0}^{N-1} x_n^* y_{n+k_1} z_{n+k_2} \end{aligned} \quad (145)$$

where  $\tau_1 = k_1 \Delta t$  and  $\tau_2 = k_2 \Delta t$ . Triple auto-correlations  $R_{xxx}$ , or mixed cross-correlations and gated cross-correlations etc. are also possible - see section 12.1.3. The triple auto-correlation is often used to enhance the visibility of latent periodicities.

The (two-dimensional) Fourier transform of the triple correlation  $R(\tau_1, \tau_2)$  is the bispectrum<sup>101</sup>

$$B_{xyz}(\omega_1, \omega_2) = \langle X^*(\omega_1) Y^*(\omega_2) Z(\omega_1 + \omega_2) \rangle. \quad (146)$$

Sometimes the bispectra is also defined as

$$B_{xyz}(\omega_1, \omega_2) = \langle X(\omega_1) Y(\omega_2) Z^*(\omega_1 + \omega_2) \rangle, \quad (147)$$

the difference being simply a complex conjugate in  $B_{xyz}$ , which is not important for the modulus, but *is* important for the cross-biphase.

## 9.3 Trispectrum $T_{xyzw}$

For completeness, the general cross-trispectrum is given by:

$$\hat{T}_{xyzw}(\omega_1, \omega_2, \omega_3) = \lim_{T \rightarrow \infty} \frac{1}{T} E[X^*(\omega_1) Y^*(\omega_2) Z^*(\omega_3) W(\omega_1 + \omega_2 + \omega_3)] \quad (148)$$

$$= \langle X(\omega_1) Y(\omega_2) Z^*(\omega_3) W(\omega_4) \rangle \quad (149)$$

<sup>99</sup> Lohmann A.W. & Wirnitzer B. *Triple correlations*, Proc. IEEE **72** 889 (1984).

<sup>100</sup> Swami A. et al. *Higher-order spectral analysis toolbox, User's guide*, v.2, The MathWorks Inc. <https://www.ligo.caltech.edu/rana/mat/HOSA/HOSA.PDF> (1998) - and refs. therein.

<sup>101</sup> Nikias C.L. et al *Bispectrum estimation: A digital signal processing framework*, Proc. IEEE **75** (7) 869 (1987).

where  $X(), Y(), Z(), W()$  are the Fourier transforms of signals  $x(t), y(t), z(t)$  and  $w(t)$  at respective  $\omega_n$  with  $\omega_4 = \omega_1 \pm \omega_2 \pm \omega_3$ . As for the bicoherence, the tricoherence squared is likewise defined as

$$t_{xyzw}^2(\omega_1, \omega_2, \omega_3) = \frac{|\hat{T}_{xyzw}(\omega_1, \omega_2, \omega_3)|^2}{\langle X(\omega_1) Y(\omega_2) Z(\omega_3) \rangle \langle W(\omega_4)^2 \rangle} \quad (150)$$

where  $t^2$  ranges from 0 to 1. If  $x = y = z = w$  then this gives the auto-trispectrum. The trispectrum is also the Fourier transform of a 4th order cross-correlation <sup>100</sup>

$$R_{xyzw}(\tau_1, \tau_2, \tau_3) = \lim_{N \rightarrow \infty} \frac{1}{N} \sum_{n=0}^{N-1} x_n^* y_{n+k_1}^* z_{n+k_2}^* w_{n+k_3} + \text{correction terms.} \quad (151)$$

```

; Generate data - sinewave plus noise
npt = 8192 & seed = 104 & nlag = 60 & fr = 181.
noise = 0.01*Reform(RandomN(seed, 1, npt))
data = Cos(!pi*2.*Findgen(npt)*fr/npt) + noise
lag1 = (lag2 = Indgen(nlag))
xd = Conj(data - Total(data)/npt)
yd = data - Total(data)/npt
zd = data - Total(data)/npt
; Triple auto-correlation, un-normalized
Cor = Fltarr(nLag, nLag)
FOR k1 = 0L, nlag-1 DO BEGIN
  FOR k2 = 0L, nlag-1 DO BEGIN
    Cor[k1, k2] = Total(xd[0:npt-1-lag1[k1]-lag2[k2]] $
      * yd[lag1[k1]:npt-lag2[k2]-1] * zd[lag2[k2]:npt-lag1[k1]-1])
  ENDFOR
ENDFOR
ENDFOR

```

Example. 20 : Code snippet (IDL) to calculate triple correlation - cf. fig. 34.

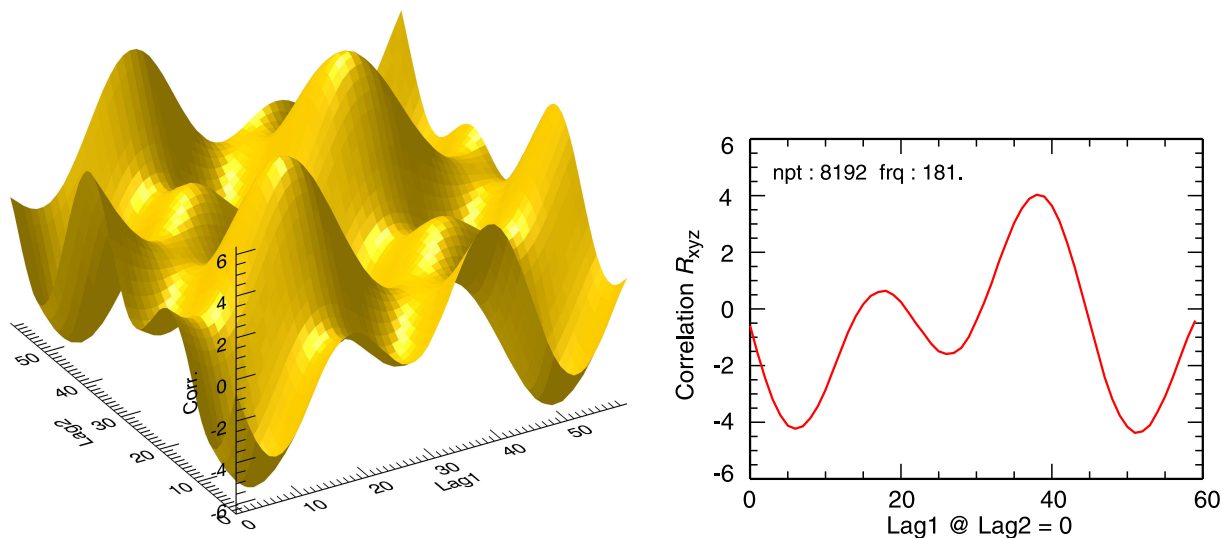


Figure. 34: Left: (Un-normalized) triple auto-correlation of a sinewave with 1% random noise vs lag1 and lag2, right: triple correlation vs positive lag1 at lag2= 0.

## 9.4 Energy transfer

While bicoherence analysis techniques can indicate the strength of nonlinear couplings within, or between regions of the turbulence domain, they do not give the direction of the energy transfer, eg. from regions of fluctuation drive to regions of dissipation. For this other techniques are employed.<sup>102</sup>

### 9.4.1 Power transfer function (PTF)

The first technique, developed by Ritz<sup>103</sup> and modified by Kim<sup>104</sup> based on a Hasegawa-Mima model (Hasegawa PRL 1977) is an energy flow equation for a single-field signal

$$\partial P_k / \partial t \approx \gamma_k P_k + \sum_{k_1, k_2} T_k(k_1, k_2), \quad (152)$$

where  $P(k) = \langle X(k)X^*(k) \rangle$  is the spectral power of some signal  $X(t, k)$  at wavenumber  $k = k_1 + k_2$ . The equation is reformed and solved for the fluctuation linear growth/damping rate  $\gamma_k$  and the nonlinear energy transfer rate  $T_k(k_1, k_2)$ , with the aid of auto and cross power spectra, auto and cross bispectra and forth order moments of  $X(k)$ . See Kim<sup>104</sup> for algorithm details.

The resulting 2D contour plot of  $T(k_1, k_2)$  visualizes the nonlinear couplings, as well as the direction of the energy transfer between different scales.

Nagashima<sup>105 106</sup>, also assuming 3-wave coupling  $k_3 = k_1 + k_2$  and  $f_3 = f_1 + f_2$ , obtained a vector energy transfer function in the frequency domain

$$F(\omega_1, \omega_2) = [k_r(\omega_1) + k_r(\omega_2)] \text{Im} \langle x_\theta(f_1) x_r(f_2) x_r^*(f_3) \rangle. \quad (153)$$

where,  $\langle x_\theta x_r x_r^* \rangle$  is essentially a poloidal-radial flow cross-bispectra.

---

<sup>102</sup> Conway G.D. et al *Geodesic acoustic modes in magnetic confinement devices* Nucl. Fusion **62** 013001 (2022).

<sup>103</sup> Ritz Ch.P. et al *Experimental measurement of three-wave coupling and energy cascading*, Phys. Fluids B **1** (1) 153 (1989).

<sup>104</sup> Kim J.S. et al *Technique for the experimental estimation of nonlinear energy transfer in fully developed turbulence*, Phys. Plasmas **3** (11) 3998 (1996).

<sup>105</sup> Nagashima Y. et al *Nonlinear energy transfer of drift-wave fluctuation in a cylindrical laboratory plasma*, Plasma Fus. Res. **3** 056 (2008).

<sup>106</sup> Nagashima Y. et al *Observation of the parametric-modulational instability between the drift-wave fluctuation and azimuthally symmetric sheared radial electric field oscillation in a cylindrical laboratory plasma*, Phys. Plasmas **16** 020706 (2009).

### 9.4.2 Amplitude Correlation Technique (ACT)

A related method by Crossley<sup>107</sup> uses the time-delayed cross correlation between two frequency bands of the same single-field signal (e.g. band pass filtered at low and high frequency ranges)

$$C(\tau) = \langle x_{\Delta f_1}^2(t).x_{\Delta f_2}^2(t + \tau) \rangle. \quad (154)$$

The sign of the correlation time delay gives the direction of energy flow between the two frequency bands  $\Delta f_1$  and  $\Delta f_2$ .

### 9.4.3 Kinetic energy transfer (KET)

The KET method, developed by Holland<sup>108</sup> is a multi-field (more than one vector field) technique which gives the evolution of the fluctuation energy  $\partial \langle |\tilde{s}(f)|^2 \rangle / \partial t$  in the frequency domain as the sum of a linear and nonlinear term involving a coupling parameter

$$T_n^y(f, f') = -\text{Re} \langle \tilde{s}^*(f) \tilde{v}_y(f - f') \partial_y \tilde{s}(f') \rangle. \quad (155)$$

The parameter  $T_y$  is essentially a cross-bispectrum which quantifies the rate at which energy is transferred between a scalar signal  $s$  at frequency  $f$  and signal gradient fluctuations ( $y$  direction) at frequency  $f'$ , mediated by a third  $y$ -direction velocity  $v_y(f - f')$  fluctuation. Holland also gives the energy transfer  $T$  in terms of the cross-bicoherence and biphas spectra  $\langle \tilde{n} v_{\perp} \nabla \tilde{n} \rangle$  and shows their equivalence (with some nuances) using experimental and simulation data. See above reference for details.

Xu derived a similar multi-field method to give kinetic energy transfer rates in the frequency domain<sup>109 110</sup> The equation form for  $T$  is similar to that of Holland, for example for  $x(t)$  the transfer rate

$$T_x(f, f_1) = -\text{Re} \langle \mathcal{F}(x^*)_f \mathcal{F}(\mathbf{u}_{\perp f_2} \cdot \nabla_{\perp}) \mathcal{F}(x)_{f_1} \rangle, \quad (156)$$

where  $f = f_1 + f_2$ ,  $\mathbf{u}_{\perp}$  is the perpendicular velocity,  $\nabla_{\perp}$  is the gradient in the perpendicular direction, and the subscript of  $\mathcal{F}(x)_f$  means the Fourier frequency component  $f$  of signal  $x$ .

<sup>107</sup> Crossley F.J. et al *Experimental study of drift-wave saturation in quadrupole geometry*, Plasma Phys. Control. Fusion **34** (2) 235 (1992).

<sup>108</sup> Holland C. et al *Zonal-flow-driven nonlinear energy transfer in experiment and simulation*, Phys. Plasmas **14** 056112 (2007).

<sup>109</sup> Xu M. et al *Study of nonlinear spectral energy transfer in frequency domain*, Phys. Plasma **16** 042312 (2009).

<sup>110</sup> Xu M. et al *Fourier-domain study of drift turbulence driven sheared flow in a laboratory plasma*, Phys. Plasma **17** 032311 (2010).

## 10 Time-frequency analysis

For time-varying or intermittent signals, different analysis methods are needed that do not assume or require signal stationarity. There are essentially two classes of techniques available: those that rely on some form of time shifting windows, and those that are based on the instantaneous frequency concept.

### 10.1 Short time or sliding FT

The most popular time-frequency spectra is the short-time Fourier transform STFT, which is essentially a simple short windowed FT, defined as

$$S(t, \omega) = \left| \int_{-\infty}^{\infty} x(t + \tau) h_K(\tau) e^{-i\omega\tau} d\tau \right|^2 \quad (157)$$

or in discrete form

$$S_K(n, j) = \frac{1}{K} \left| \sum_{k=-K/2}^{K/2-1} x_{n+k} h_K(k) e^{-i2\pi jk/K} \right|^2 \quad (158)$$

where  $h_K(k)$  is a time (index) translated window function of width  $K$  points and  $\tau = k\Delta t$ , and  $\omega = 2\pi j/\Delta t$ . The FT is usually implemented with an FFT. The simplest window is a box-car function, which, as expected from convolving the spectra with the  $H(\omega) \Leftrightarrow h(t)$  leads to leakage in the spectra and reduced frequency resolution. Window shaping or tapering is possible, and the FT is also amenable to zero padding. Nevertheless, overall the optimal window length  $K$  is always a trade-off between time resolution and frequency resolution.

### 10.2 Wigner-Ville spectrum

The Wigner-Ville (WV) spectrum or Wigner distribution (WD) is one of a class of quadratic time-frequency methods using Fourier transforms of the autocovariance function.<sup>111</sup> The WV spectrum overcomes some of the limitations of the STFT by using a forward and backward  $\pm\tau$  conjugate pair, and is defined as<sup>112</sup>

$$W(t, \omega) = \int z(t + \tau/2) z^*(t - \tau/2) h(\tau/2) h(-\tau/2) e^{-i\omega\tau} d\tau \quad (159)$$

where  $z()z^*()$  is essentially the instantaneous auto-correlation of the analytic signal  $z(t) = x(t) + i\mathcal{H}\{x(t)\}$  (see section 10.7). For numerical implementation it is easier to use the so-called (unsmoothed) pseudo-Wigner estimator

$$pW_{(2K-1)}(n, j) = 2 \sum_{k=-K+1}^{K-1} x_{n+k} x_{n-k}^* |h_K(k)|^2 e^{-i2\pi j(k/K)} \quad (160)$$

<sup>111</sup> Martin W. & Flandrin P. *Wigner-Ville spectral analysis of nonstationary processes*, IEEE Trans. Acoust. Speech **ASSP-33** (6) 1461 (1985).

<sup>112</sup> Boashash B. *Estimating and interpreting the instantaneous frequency of a signal: Part 1*, Proc. IEEE **80** (4) 520 (1992) and *Part 2*, Proc. IEEE **80** (4) 540 (1992).



where  $\omega_j = 2\pi j/K$  and  $h_K(\cdot)$  is a rectangular window function of  $2K - 1$  non-zero points.<sup>113</sup>

If  $K$  is a power of 2 then  $pW$  (over  $2K - 1$ ) points can be calculated using one FFT over  $K$  data points. Smoothed versions would include a second summation over adjacent  $pW$  estimates.<sup>114</sup>

The Wigner-Ville spectrum has some positive points - it has good frequency resolution and is time invariant. The total power/energy is also conserved, as are the marginal spectral powers  $P(t) = \int W(t, \omega) d\omega$  and  $P(\omega) = \int W(t, \omega) dt$  (unlike the STFT). However, points against include the possibility of negative  $W$  values and, perhaps more importantly, due to its quadratic nature the formation of significant cross-terms in the product of the forward and backward signal conjugate pair, leading to spectral *interference* in multi-component signals. By comparison, the product of the forward-forward conjugate pair in the STFT produces minimal cross-terms.

### 10.3 Choi-Williams distribution

The Choi-Williams distribution (CWD) is a more generalized version of the WV distribution. It is a type of so-called Reduced Interference Distribution (RID) which includes an addition exponential kernel term  $k$  to reduce the cross terms<sup>115</sup>

$$CW(t, \omega, \sigma) = \int \int \frac{1}{|\tau|} k\left(\frac{\mu - t}{\tau}, \sigma\right) x(\mu + \tau/2) x^*(\mu - \tau/2) e^{-i\omega\tau} d\mu d\tau \quad (161)$$

where the kernel  $k(t, \sigma) = (2\pi\sigma^2)^{-1/2} e^{-t^2/(2\sigma^2)}$ .  $\sigma$  is the kernel scale factor. Note  $k(\cdot) = 1$  for the WD spectrum. Translating to discrete form (which is non-trivial) gives

$$CW(n, \theta, \sigma) = 2 \sum_{\tau=-\infty}^{\infty} e^{i2\theta\tau} \left[ \sum_{\mu=-\infty}^{\infty} \frac{1}{\sqrt{4\pi\tau^2/\sigma}} \exp\left(-\frac{(\mu - n)^2}{4\tau^2/\sigma} x_n x_n^*\right) \right] \quad (162)$$

where  $n = t/\Delta t$  and  $\theta = 2\pi f\Delta t$  is the rotational angle equivalent of  $\omega$ .<sup>115 116</sup>

Positive aspects of the CWD are the reduced cross-terms in the distribution due to the kernel, but at the expense of reduced resolution. Here the trade-off is determined by  $\sigma$ . Concerning implementation, either long-hand, via FFT or via matrix methods. See reference<sup>117</sup> for examples.

<sup>113</sup> Bizarro J.P.S. & Figueiredo A.C. *The Wigner distribution as a tool for time-frequency analysis of fusion plasma signals: application to broadband reflectometry data*, Nucl. Fusion **39** (1) 61 (1999).

<sup>114</sup> Boashash B. *Note on the use of the Wigner distribution for time-frequency signal analysis*, IEEE Trans. Acoust. Speech **ASSP-36** (9) 1518 (1988).

<sup>115</sup> Choi H. & Williams W.J. *Improved Time-Frequency representation of multicomponent signals using exponential kernels*, IEEE Trans. Acoust. Speech Signal Proc. **ASSP-37** (6) (1989).

<sup>116</sup> Figueiredo A.C.A. et al *Time-frequency analysis of non-stationary signals in fusion plasmas using the Choi-Williams distribution*, 30th EPS Conf. (St.Petersburg) ECA **27A** P2.141 (2003).

<sup>117</sup> Barry D.T. *Fast calculation of the Choi-Williams time-frequency distribution*, IEEE Trans. Signal Proc. **SP-40** (2) 450 (1992).

## 10.4 S-method

The S-method (SM) is a time-frequency technique which provides a transition from the simple STFT (with minimal cross-terms) to the WD (high energy concentration). The essence of the method is to use correction terms to the base STFT spectrum. The SM spectrum is defined as <sup>118</sup>  
119

$$SM(t, \omega) = \frac{1}{\pi} \int_{-\infty}^{\infty} P(\varpi) \mathcal{F}(t, \omega + \varpi) \mathcal{F}^*(t, \omega - \varpi) d\varpi \quad (163)$$

where  $\mathcal{F}$  is the FT of  $x_n$  and  $P(\varpi)$  is a window function. If  $P(\varpi) = 1$  for all  $\varpi$  then one obtains the WD  $W(t, \omega)$ , and for  $P(\varpi) = \pi\delta(\varpi)$  one obtains the STFT  $S(t, \omega)$ .

In discrete form (without windowing)

$$SM_L(n, j) = \sum_{l=-L}^L \mathcal{F}(n, j+l) \mathcal{F}^*(n, j-l) \quad (164)$$

where  $n$  is the time index,  $j$  is the frequency index and  $L$  is the number of correction (ie. lag) terms. In the terminology used here the FT is zero centered so that the index  $l$  goes from negative to zero to positive frequency indexes. In the above equation  $P(l) = 1$ , however, a weighted form  $P(l) = 1/(2L+1)$  can be used where  $-L \leq l \leq L$  is the frequency index summation range. Reference <sup>119</sup> gives a recursive relation for

$$SM_L(n, j) = SM_{L-1}(n, j) + 2\text{Re}[\mathcal{F}(n, j+L) \mathcal{F}^*(n, j-L)]. \quad (165)$$

Essentially,  $2\text{Re}[\mathcal{F}(n, j+l) \mathcal{F}^*(n, j-l)]$  are the correction terms for each step of the recursion  $l = 1, \dots, L$ , and the initial spectrum is simply the STFT auto-spectrum  $S_N$

$$SM_0(n, j) = S_N(n, j) = \mathcal{F}(n, j) \mathcal{F}^*(n, j) = |\mathcal{F}(n, j)|^2. \quad (166)$$

Both of the Stanković references show good examples of the benefits of the S-method in action. Needless to say, the S-method is just one of many similar methods attempting to improve on the behaviour of the WD.

<sup>118</sup> Stanković L. *A method for time-frequency analysis*, IEEE Trans. Sig. Proc. **42**(1) 225 (1994).

<sup>119</sup> Stanković L. et al. *From the STFT to the Wigner distribution*, IEEE Sig. Proc. Magazine **May** 163 (2014). doi: 10.1109/MSP.2014.2301791

## 10.5 Wavelet transform

By their nature Wavelet analyses are well disposed for the study of time-dependent or intermittent signals. There is now a large body of work on the basis and application of wavelet analysis. A comprehensive and highly accessible tutorial is given by Torrence & Compo.<sup>120</sup>

The continuous wavelet transform (CWT) is defined as<sup>121</sup>

$$W(a, \tau) = \int_{-\infty}^{\infty} x(t) \psi_a(t - \tau) dt = \langle x(t) \psi_a(t - \tau) \rangle \quad (167)$$

where  $\psi_a(t) = |a|^{-1/2} \psi(t/a)$  is the scaled (by  $a$  - the dilution scale) and time translated (by  $\tau$ ) wavelet function and  $\psi(\eta)$  the scalable (or mother) wavelet base function. There are several wavelet bases in use, cf.<sup>120</sup>, the most popular perhaps being the original Morlet wavelet - a sinewave windowed by a Gaussian

$$\psi(\eta) = \pi^{-1/4} e^{-\eta^2/2} e^{i\omega_0\eta}. \quad (168)$$

Setting the scale factor  $\omega_0 \approx 2\pi$  in the 2nd exponential, allows an equivalent frequency  $\omega = 2\pi/a$  to be assigned to each scale  $a$  (note some authors use the scale number  $s = 1/a$  where the scale is comparable to a period). The corresponding frequency resolution is then  $\Delta\omega = \omega/4$  and the time resolution  $\Delta t = 2a$ , thus  $\Delta\omega \times \Delta t = \pi$ .<sup>122</sup> Torrence<sup>120</sup> gives a good overview of the pros and cons of the various wavelets, Morlet, Paul, Mexican hat and DOG  $\psi$  in terms of orthogonality, complex vs real, wavelet width and shape.

There is also a discrete wavelet function, but here the *continuous* wavelet as applied to a discrete data sequence  $x_n$  is described as

$$W_n(a) = \sum_{n'=0}^{N-1} x_n \psi^* \left( \frac{(n' - n)\delta t}{a} \right) \quad (169)$$

where the subscript  $n$  is the time integer and  $n'$  the time translation. Since eqn. (167) is essentially a convolution, the CWT can be expressed as  $W = (1/\sqrt{T}) \mathcal{F}^{-1} \{ \mathcal{F}(x) \mathcal{F}(\psi) \}$  which, in turn allows an efficient computation via FFTs (with zero padding if necessary).

The wavelet power spectrum at scale  $a$  is defined as

$$S_n^W(a) = |W_n(a)|^2. \quad (170)$$

As with the FFT power spectra the summed  $S^W$  spectra, ie. the expected value  $E[|W_n(a)|^2] = \sigma^2$  is the variance of the signal. That is,  $S^W(a)$  is the scale distribution of the signal variance. Both

<sup>120</sup> Torrence C. & Compo G.P. *A practical guide to wavelet analysis*, Bull. Amer. Meteor. Soc. **79** (1) 61 (1998).

<sup>121</sup> Jordan D. et al *Implementation of the continuous wavelet transform for digital time series analysis*, Rev. Sci. Instrum. **68** (3) 1484 (1997).

<sup>122</sup> van Milligen B.Ph. et al *Statistically robust linear and nonlinear wavelet analysis applied to plasma edge turbulence*, Rev. Sci. Instrum. **68** (1) 967 (1997).

the Morlet and Paul wavelets are complex, thus  $W_n$  is complex, allowing  $S^W$  to be expressed in terms of amplitude and phase spectra. The DOG wavelet, however, is a real function, thus  $\text{Im}(W_n) = 0$  and the phase is undefined.

As with the STFT and HHT spectra (section 10.8) one can also define marginal spectra for the WT. The **Global wavelet spectrum** is the time integrated (ie. summed over  $n$  index) marginal spectrum, defined as

$$\overline{W}^2(a) = 1/N \sum_{n=0}^{N-1} |W_n(a)|^2 \quad (171)$$

The  $W_n(a)$  spectrum can also be smoothed over adjacent time slices to increase the effective spectral dof ( $\nu$ ) and thus enhance spectral peak significance. Averaging over a scale range  $a_1 - a_2$  will give the power within that scale (range) as a function of time <sup>120</sup> (equivalent to the HHT marginals).

$$\overline{W}_n^2 \propto \sum_{j=j_1}^{j_2} |W_n(a_j)|^2 \quad (172)$$

Some interesting examples of wavelet usage in plasma physics include: Dose <sup>123</sup> who describes an application to MHD analysis, with particular emphasis on correct interpretation, Farge <sup>124</sup> on the extraction and characterization of coherent features from noisy signals, ie. filtering (with further refs. therein), and Mahdizadeh <sup>125</sup> also as a means of filtering intermittent features.

### 10.5.1 Cross-wavelet spectra

For multiple signals, corresponding wavelet cross-spectrum and cross-phase spectra can be computed <sup>126</sup>

$$W_{xy}(a) = \frac{W_x(a) W_y^*(a)}{\sigma_x \sigma_y} \quad \theta_{xy}(a) = \tan^{-1} \left\{ \frac{\text{Im } W(a)}{\text{Re } W(a)} \right\}. \quad (173)$$

However, the raw wavelet coherence is unity  $\gamma_W(a) = 1$ , unless some form of spectral smoothing or ensemble averaging is employed <sup>122</sup> which, one way or another reduces the temporal or scale resolution of the spectrum. The use of wavelets for bandpass signal filtering, in either scale or time, was also mentioned above, cf. <sup>120</sup>

<sup>123</sup> Dose V. et al *Wavelet analysis of fusion plasma transients*, Phys. Plasmas **4** (2) 323 (1997).

<sup>124</sup> Farge M et al. *Extraction of coherent bursts from turbulent edge plasma in magnetic confinement devices using orthogonal wavelets*, Phys. Plasmas **13** 042304 (2006).

<sup>125</sup> Mahdizadeh N. et al. *Investigation of intermittency in simulated and experimental turbulence data by wavelet analysis*, Phys. Plasmas **11** (8) 3932 (2004).

<sup>126</sup> Santoso S. et al *Time-series analysis of nonstationary plasma fluctuations using wavelet transforms*, Rev. Sci. Instrum. **68** (1) 898 (1997).

### 10.5.2 End effects and Confidence limits

One of the major issues of using finite data lengths is the deterioration of the wavelet spectral power at the beginning and at the end of the transform. Zero padding of the data prior to the transform can mitigate, but the effect remains. It is common to mark the affected regions of the  $|W_n(a)|^2$  plot with lines marking the so-called *Cone of influence*. See Torrence<sup>120</sup> for further details.

Another issue is the confidence interval (CI, the range of confidence about a point) and the significance level (SL, the minimum level of significance). As with the FFT spectrum, see section 6.5, their derivation depends on the assumed error distribution shape (ie. normal or Poisson etc.) and the spectral type. For the temporally averaged wavelet coherence spectrum  $\gamma_{Wxy}(a)$ , van Milligen<sup>122</sup> gives an estimate for the significance level

$$\epsilon[\hat{\gamma}_{xy}(a)] \approx 2\sqrt{\frac{\omega_s}{\omega} \frac{1}{N}} \quad (174)$$

where  $N$  is the number of samples,  $\omega_s = 2\pi/\delta t$  is the sample frequency and  $\omega = 2\pi/a$  which results from a reduced independence (dof) of the wavelet coefficients due to the finite time period of the spectra averaging.

```

; Create data : Constant freq + Chirp
npt = 2048 & nft = 256 & nst = 16
time = Findgen(npt)/Float(npt)
data = Sin(!pi*2*time*500.) + Sin(!pi*(time*16.)^2)
; STFT
freq = Findgen(nft*2)*4. ; double length & scale
sP2 = Fltarr(128,nft*2)
tims = Fltarr(128)
wind = 0.5*(1.0-Cos(2.0*!pi*Findgen(nft)/Float(nft)))
zero = Fltarr(nft) ; zero padding
j = 0
kmx = (npt - nft) / nst ; 2048-256/16 = 112
FOR k = 0, kmx DO BEGIN
  tims[k] = j+nft/2
  b = FFT([data[j:j+nft/2-1]*wind,zero],-1) ; pad, window & FFT
  sP2[k,*] = Abs(b*Conj(b))
  j = j + nst
ENDFOR
sP2 = sP2[0:k-1,0:200] & tims=tims[0:k-1] & freq=freq[0:200]
; Continuous wavelet (scale vs. time)
wave = WV_CWT(data, 'Morlet', 6, /Pad, Scale=scale)
wP2 = Abs(wave^2)/npt
FOR i = 0, npt-1 DO wP2[i,*]=wP2[i,]/scale ; rescale power
freq = Float(npt)/scale
gws = Total(wP,1)/npt ; global wavelet spec.

```

*Example. 21 : Code snippet (IDL) for STFT and Wavelet of a constant frequency sinewave plus a chirping frequency - cf. fig. 35.*

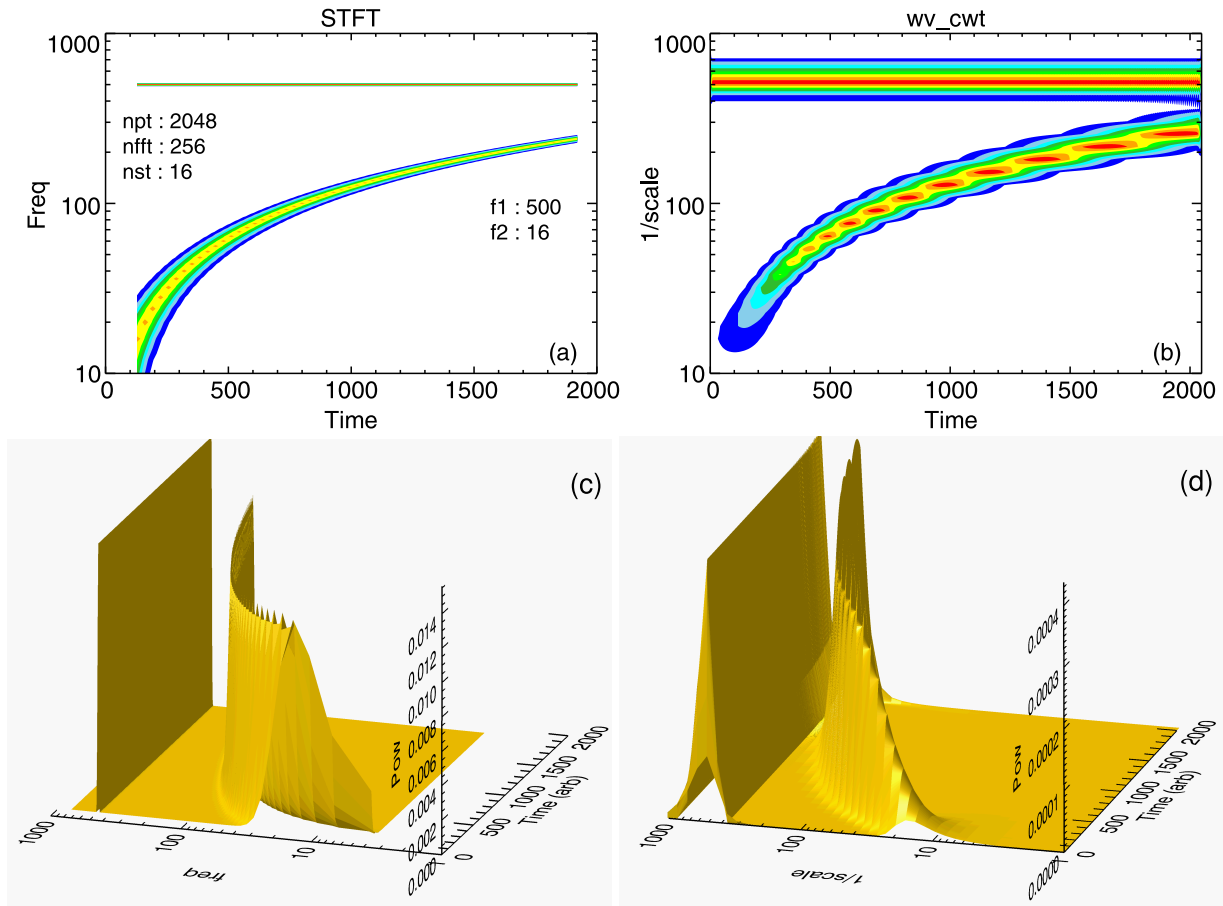


Figure. 35: Example of (left) STFT and (right) DWT spectra, as contour plots (top) and 3D perspective plots (bottom) for a constant sinewave plus a frequency chirp sinewave, using above code snippet.

### 10.5.3 Wavelet bicoherence $b_w^2$

Wavelet transforms can also be applied to bispectra and bicoherence analysis of the non-linear coupling of non-stationary signals.<sup>127</sup> The formulation is similar to that for FFT bispectral analysis - see section 9.1 - except an element of temporal integration over a finite time interval  $T$  which slides with the wavelet translation  $\tau$ . Thus the wavelet cross-bispectrum is defined as

$$B_{xy}^W(a_1, a_2) = \int_T W_x^*(a, \tau) W_y(a_1, \tau) W_y(a_2, \tau) d\tau \tag{175}$$

where  $1/a = 1/a_1 + 1/a_2$ . For a Morlet type wavelet the scale  $a = 2\pi/\omega$  translates to a frequency and one can see the equivalence to the familiar  $\omega = \omega_1 + \omega_2$ . From the bispectrum the wavelet squared bicoherence is defined as

$$b_W^2(a_1, a_2) = \frac{|B^W(a_1, a_2)|^2}{\int_T |W_x(a_1, \tau) W_y(a_2, \tau)|^2 d\tau |W_x(a)|^2} \tag{176}$$

<sup>127</sup> van Milligen B.Ph. *Wavelet bicoherence: a new turbulence analysis tool*, Phys. Plasmas **2** (8) 3017 (1995).

When using bicoherence to study 3-wave coupling (in this case, wavelets) the interpretation only really makes sense in the frequency domain. Thus, the scale conversion of a Morlet wavelet to a frequency  $\omega = 2\pi/a$  is most appropriate.

Finally, for the  $b_w$ , van Milligen gives an estimate for the significance level as

$$\Delta \hat{b}_w / \hat{b}_w \approx 2/\sqrt{N} \quad (177)$$

where  $N$  is the number of samples. This is slightly better than for the FFT based bicoherence, which might imply that the wavelet bicoherence estimate may have less uncertainty, ie. smaller variance. For an example of computation see reference <sup>127</sup>.

## 10.6 Adaptive AR modelling

Autoregressive (AR) spectra are particularly well suited to short data streams and thus can also be applied to non-stationary data. **Adaptive AR** using, for example the Least Mean Squares algorithm allows the regression coefficients  $a$  to vary or evolve or adapt with time, ie. a short-form AR spectrum. <sup>128</sup>

---

<sup>128</sup> Ulrych T.J. & Ooe M. *Chapter 3: Autoregressive and mixed autoregressive-moving average models and spectra* in *Nonlinear methods of spectral analysis*, 2nd edn. Springer (1983).

## 10.7 Hilbert transform

The Hilbert transform (HT) is a type of (linear) operator which shifts the phase angle of a signal by  $\pm 90^\circ$ , ie.  $\pi/2$ .

$$\begin{aligned}\mathcal{H}\{x(t)\} &= \frac{1}{\pi} \int_{-\infty}^{\infty} \frac{x(\tau)}{t - \tau} d\tau = x(t) * -1/(\pi t) \\ \mathcal{H}\{x_n\} &= \sum_{k=-\infty}^{\infty} x_k \frac{\sin^2[\pi(n-k)/2]}{\pi(n-k)/2} \quad (\text{cf. ref.}^{113})\end{aligned}\tag{178}$$

that is, the signal is convoluted with a linear time delay. The resulting  $\mathcal{H}\{x(t)\}$  and  $x(t)$  have the same power spectrum and auto-correlation function, but are orthogonal. Most computer code packages, such as IDL, Python etc. provide a Hilbert transform routine.

Essentially the Hilbert transform creates an imaginary component of the signal:  $\text{Im } x(\omega) = \mathcal{H}\{\text{Re } x(t)\}$ , which is used to form the **analytic signal**

$$z(t) = x(t) - i \mathcal{H}\{x(t)\} = a(t) \exp(i \theta(t))\tag{179}$$

a pair of instantaneous magnitude and phase signals. Note, the analytic signal is equivalent to the **quadrature signal** if  $a$  and  $\theta$  are separable.

### 10.7.1 Hilbert spectral analysis

The Hilbert spectral analysis (HSA) method (cf. <sup>129</sup>) uses the HT to produce a time-frequency spectrum, aka the Hilbert (amplitude) spectrum. The approach is first to decompose the signal into a set of real **Intrinsic Mode Functions** (IMF),  $c_j(t) = a_j(t) \cos(\theta_j(t))$  plus a residual  $r(t)$

$$x(t) = \sum_{j=1}^J c_j(t) + r(t)\tag{180}$$

using, for example, the **Empirical Mode Decomposition** (EMD), as used in the HHT method or the **Iterative Filtering** (IF) method - see section (10.8). The Hilbert transform is then applied to each IMF component to produce their analytic signals from which their **instantaneous frequencies**  $\omega_j(t) = d\theta_j(t)/dt$  are obtained. For discrete signals the phase gradient can be computed using forward (FFD) and backward finite difference (BFD) equations <sup>130</sup> The Hilbert spectrum is then defined (for positive  $\omega_j(t)$ ) as the sum of the IMF amplitude components

$$H(t, \omega) = \sum_{j=1}^J H_j(t, \omega) = \sum_{j=1}^J a_j(t) \quad \text{at selected } \omega = \omega_j(t)\tag{181}$$

The  $H(t, \omega)$  enables the representation of signal amplitude as a function of time and instantaneous frequency in a three-dimensional or contour plot.

<sup>129</sup> Huang N.E. et al *The empirical mode decomposition and the Hilbert spectrum for nonlinear and non-stationary time series analysis*, Proc. Royal Soc. Lon. A **454** 903-995 (1998)

<sup>130</sup> Boashash B. *Estimating and interpreting the instantaneous frequency of a signal: Part 1*, Proc. IEEE **80** (4) 520 (1992) and *Part 2*, Proc. IEEE **80** (4) 540 (1992).



### 10.7.2 Envelope detection

A particularly useful application of the HT is for extracting the amplitude envelope of a signal. One simply takes the modulus of the analytic signal and then low-pass filter or smooth

$$\text{Env}(t) = \text{LPF}\{|(x(t) + i\mathcal{H}\{x(t)\})|\} \quad (182)$$

$\text{Env}(t)$  can then be down-sampled if necessary. As an example, fig. 36 shows: (a) the envelope of a decaying sinewave, typical of a narrow band auto-correlation function, and (b) an example of the so-called **envelope detection** of amplitude modulated (AM) broad-band fluctuations. A particular application is the extraction of flow oscillations from the modulation of high frequency density turbulence.<sup>131</sup> Here,  $x(t)$  is high-pass filtered before taking the HT.

### 10.7.3 Complex demodulation

Complex, or quadrature demodulation is a well known method in the field of electrical engineering for extracting the modulation of a carrier wave. In time series analysis it offers a method of analyzing non-stationary data to obtain the phase and amplitude variation of a chosen frequency component as a function of time. Complex demodulation involves multiplying the signal with a complex exponential followed by low-pass filtering. In the 1988 Powers paper<sup>132</sup> the starting point is the assumption that the signal can be expressed as a set of  $a(t) \cos[\omega_o t + \theta(t)]$  sinusoids where  $\omega_o$  is the selected carrier frequency and  $a(t)$  and  $\theta(t)$  are its slowly time-varying amplitude and phase. Numerically the demodulation is achieved by multiplying the signal by the complex exponential  $2^{1/2} \exp(-i\omega_o t)$ , where  $\omega_o = j\Delta\omega$ , then narrow-band-pass filtering with bandwidth  $\Delta\omega$  to recover the complex demod signal  $c(\omega_j, t) = a(\omega_j, t) \exp[i\theta(\omega_j, t)]/2^{1/2}$ . The process is repeated for each  $j = 0, \dots, J - 1$ . In principle the frequencies do not need to be harmonic. The time varying (auto) power spectra is then simply  $P_{xx}(\omega_j, t) = |c(\omega_j, t)|^2 = a^2(\omega_j, t)/2$ . The method can be extended easily to cross-power spectra  $P_{xy}(\omega_j, t) = c_x(\omega_j, t) c_y^*(\omega_j, t)$ .

There is a certain duality with the HSA where the carrier modes are essentially IMFs and the complex exponent action is the HT - although with different ordering. The multiple filtering steps are computationally intensive, but an alternative approach is to inverse FT the frequency windowed, phase shifted signal.<sup>133</sup>

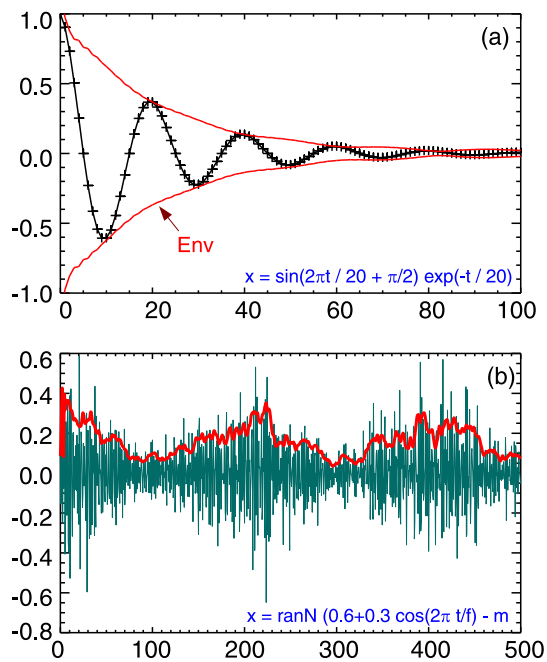


Figure. 36: HT envelope detection: (a) exponential sinewave and (b) low frequency modulation of broad-band fluctuations.

<sup>131</sup> Conway G.D. et al *Observations on edge GAM-turbulence interactions in ASDEX Upgrade*, Proc. 46th EPS Conf. on Plasma Phys. (Milan) P2.1091 (2019).

<sup>132</sup> Powers E.J et al *Spectral analysis of nonstationary plasma fluctuation data via digital complex demodulation*, Rev. Sci. Instrum. **59** (8) 1757 (1988).

<sup>133</sup> Bizarro J.P.S. et al *On complex demodulation and time-frequency distributions for broadband reflectometry signals*, Rev. Sci. Instrum. **70** (10) 1030 (1999).

## 10.8 Hilbert-Huang transform HHT

In the Hilbert-Huang transform (HHT) method the intrinsic mode functions (IMFs) are obtained using a sifting process called **Empirical Mode Decomposition** (EMD). The obtained IMFs are then Hilbert spectral analyzed, cf. section (10.7.1), to produce a time-frequency plot. <sup>134 129 135</sup>

The HHT is a somewhat unique method of analyzing non-stationary and nonlinear time series data. Here the IMFs are not predefined basis functions, like sinewaves as in the Fourier or wavelets in Wavelet transforms, but are essentially adaptively derived from the data itself using a sifting process.

The EMD procedure is as follows:

- (1) Identify alternating local extrema in the signal  $x(t)$ , ie. a maxima followed by a minima.
- (2) Fit envelopes  $e_{max}(t)$  to the local maxima and  $e_{min}(t)$  to the local minima time sequences using, for example, cubic spline fits.
- (3) Create a local running mean  $m_1(t) = (e_{max}(t) - e_{min}(t))/2$  signal.
- (4) Subtract the mean signal to create a time-varying residual  $h_1(t) = x(t) - m_1(t)$ .
- (5) Check if  $h_1(t)$  qualifies as an IMF (see note 1.)
  - if YES then  $c_1(t) = h_1(t)$  and store as an IMF
    - $r_1(t) = x(t) - c_1(t)$  : create a new reduced signal
    - Repeat from step (1) until  $r$  is a trend, ie. contains no alternating max and min.
  - if NO then Refine or sift  $h_1(t)$  until it qualifies as an IMF.
    - Take  $h_1(t)$  as input and repeat steps (2) - (5) creating new mean & residual
    - $m_{11}(t)$
    - $h_{11}(t) = h_1(t) - m_{11}(t)$
    - $c_1(t) = h_{1j}(t)$
    - Check stop criterion (note 2.) - else repeat from step (2) to create next  $c_j$ .

With each successive IMF  $c_j(t)$  the fine-structure is progressively removed from  $x(t)$  until only a trend or bit noise remains.

**Note 1:** An IMF is defined as any function (not necessarily a sinusoid) that meets the following:

- It has at least 1 maxima and 1 minima and the difference between the number of local extrema and zero crossings is 0 or 1
- The local running mean is zero.

**Note 2:** There are various criteria proposed for stopping the sifting, but the criterion proposed by Huang is to check when the standard deviation of successive  $h_k$  falls below a pre-defined value,

<sup>134</sup> Huang Y.X. et al *An amplitude-frequency study of turbulent scaling intermittency using Hilbert spectral analysis*, Euro. Phys. Lett. **84** 40010 (2008).

<sup>135</sup> Stallone A. et al. *New insights and best practices for the successful use of Empirical Mode Decomposition, Iterative Filtering and derived algorithms*, Sci. Reports **10** 15161 (2020).

such as  $\sigma_s = 0.2$

$$\sigma_j = \frac{\sum_{t=0}^T |h_{j-1}(t) - h_j(t)|^2}{\sum_{t=0}^T h_{j-1}^2(t)} < \sigma_s \quad (183)$$

Since the signal is decomposed in the time domain the length of each IMFs is the same as the original signal, that is, the HHT preserves the characteristics of the varying frequencies.

Once the IMF modes  $c_j(t)$  and residual  $r_n(t)$  are obtained the signal can be reconstructed to cross-check the accuracy of the decomposition

$$x(t) = \sum_{j=1}^J c_j(t) + r_n(t) = \text{Re} \sum_{j=1}^J A_j(t) \exp(i \theta_j(t)) \quad (184)$$

Finally, the Hilbert Transform is applied to each IMF  $c_j(t)$ , the instantaneous frequencies  $\omega_j(t)$  and amplitudes  $a_j(t)$  extracted, and then  $a_j(t)$  summed over  $j$  to give the time resolved Hilbert spectrum  $H(t, \omega)$  - as per eqn. (181). From the Hilbert spectrum one can integrate over time to give the **marginal spectrum**:

$$h(\omega) = \int_0^T H(t, \omega) dt \quad (185)$$

or integrated over frequency to give the instantaneous energy density (IE):

$$IE(t) = \int_{\omega} H^2(t, \omega) d\omega. \quad (186)$$

Huang's 1998 paper is best consulted for details of how to implement the method in practice, together with important caveats, limitations and hints.<sup>136</sup>

There are EMD codes in python readily available for those wishing to try this method.

---

<sup>136</sup> Huang N.E. *The empirical mode decomposition and the Hilbert spectrum for nonlinear and non-stationary time series analysis*, Proc. R. Soc. Lond. A **454** 903 (1998).

## 11 Non-linear structure

In the late 1980's the application of non-linear structure analysis techniques became rather popular, in particular in an attempt to characterize edge plasma turbulence in terms of fractal dimensions and strange attractors. However, techniques based on embedding dimensions were computationally intensive, and the results difficult to interpret. Nevertheless, ..

### 11.1 Increments and difference functions

Increments are used in the analysis of non-linear structure buried within the time series. The approach is to generate a time series of increments for each temporal  $\tau$  or spatial  $|l|$  scale (ie. a difference index or lag)

$$\begin{aligned}\delta x(l) &= x(r+l, t) - x(r, t) \\ \delta x(\tau) &= x(t+\tau) - x(t)\end{aligned}\tag{187}$$

and then take the pdfs of the increments  $\delta x$ . At large scale separations the pdfs tend to original/normal, but at small scales intermittency is revealed as broad tails (large Kurtosis) in the pdfs.

### 11.2 Structure functions

From the increments the structure function of order  $p$  is defined as:

$$S_p(l) = \langle (\delta x(l))^p \rangle.\tag{188}$$

The angular brackets  $\langle u \rangle$  denotes the average of  $u$ . It is usual to limit  $p < 5$ . The structure function can be used in calculating the Hurst exponent.

### 11.3 Hurst exponent

The Hurst exponent  $H$  is a popular non-linear estimator parameter which gives an indication of the rate of decay of long-term behaviour (memory) of a time sequence. In particular, it may indicate if a series has a long term trend (termed trending or persistence), or is likely to change sharply (termed mean-reverting or anti-persistence).

$H$  is usually defined in terms of the asymptotic behaviour of the (ensemble averaged) rescaled range (see below) of a time series. (For  $n$  one can read  $\tau$ .)

$$E[R(n)/S(n)] = Cn^H \quad \text{as } n \rightarrow \infty\tag{189}$$

where  $R(n)$  is the range for  $n$  time points,  $S(n)$  is their corresponding standard deviation, and  $C$  is a constant. From this one sees that the rescaled range scales with the number of points  $n$  to the power of  $H$ . Usually  $H$  is extracted from a straight-line fit to a log-log plot of  $R(n)/S(n)$  against  $n$  (at the mid to high-end of  $n$  values of course).

The interpretation of the Hurst exponent is roughly as follows:<sup>137</sup>

$H < 0.5$  : the series tends to revert back to its mean, ie. anti-persistent.

$H = 0.5$  : the series has a random or stochastic behaviour.

$H > 0.5$  : the series has a clear trend - persistent.

### 11.3.1 Generalized exponent

The basic Hurst exponent is a measure of the ‘size of changes’ (sometimes called the rate of diffusive behaviour) as a function of time lag between observations, ie. the variance of the increments  $E[|x(t+\tau) - x(t)|^2]$ . This is the so-called aggregated variance method. Here one calculates the lagged auto-covariance  $\text{var}(\tau = k\Delta t) = \langle |x_{n+k} - x_n|^2 \rangle$ . If the series  $\{x_n\}$  is random then the  $\text{var}(\tau) \sim \tau$  scales roughly with the lag  $\tau$ . However, if there is a finite autocorrelation then  $\text{var}(\tau) \sim \tau^{2H}$ . The  $H$  factor is obtained from the gradient of  $\log(\text{var}(\tau))$  vs  $\log(\tau)$ . The generalized Hurst form replaces the 2 in the exponent with the structure order  $p$ . That is the Hurst exponent is a function  $H = H(p)$  of  $p$ . Thus one is essentially analyzing the scaling behaviour of the structure function as a function of lag

$$S_p(\tau) = \langle |x(t + \tau) - x(t)|^p \rangle_t \sim \tau^{pH(p)} \quad (190)$$

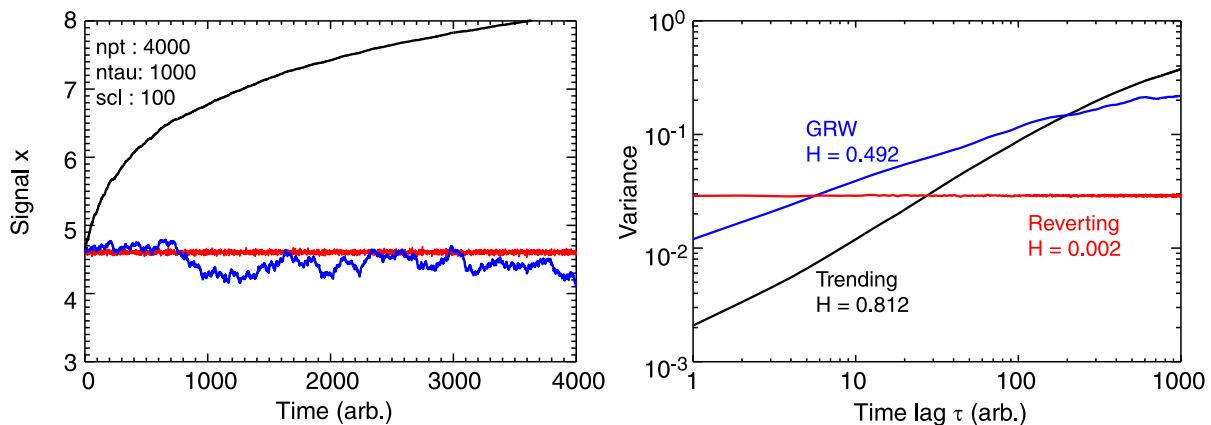


Figure. 37: Hurst analysis for 3 synthetic signals - a la Lewinson.<sup>138</sup> (Left) raw signals, (right) Log-log plot of structure function of order  $p = 2$  (variance)  $S_2(\tau)$  vs time lag  $\tau$ .

Figure 37 shows an example of calculating the Hurst exponent using eqn. (190) with order  $p = 2$ , ie. from the aggregated signal variances<sup>138</sup> using the code snippet below. Figure 37(left) shows time traces and (right) lagged variance plots for three synthetic signals: (red) mean reverting  $\ln(x)$ , (blue) a geometric random walk  $\ln(f(x))$  and (black) strongly trending  $\ln(f(x + c))$ . On the right, the log-log plot of the corresponding variance vs lag  $\tau$  allows the  $H$  factor to be extracted from a gradient fitted at moderate  $\tau$  values.

<sup>137</sup> Wikipedia - [https://en.wikipedia.org/wiki/Hurst\\_exponent](https://en.wikipedia.org/wiki/Hurst_exponent) (Dec. 2022).

<sup>138</sup> Lewinson E. *Introduction to the Hurst exponent - with code in Python*, TDC <https://towardsdatascience.com/introduction-to-the...> (2021).

```

npt = 4000 & ntau = 1000 & tau = Indgen(ntau) & scl = 100. .
x = RandomN(seed, npt) ; Random norm. signal
x1 = Alog(x*2. + scl) ; Mean reverting
x2 = Alog(Total(x, /Cumul)+ scl) ; GRW via cumulative sum
x3 = Alog(Total(x+0.8, /Cumul) + scl) ; Trending
; Aggregate variance / std dev.
V1 = Fltarr(ntau) & V2 = Fltarr(ntau) & V3 = Fltarr(ntau)
FOR it = 0, ntau-1 DO V1[it] = Stddev(x1[it:*] - x1[0:npt-it-1])
FOR it = 0, ntau-1 DO V2[it] = Stddev(x2[it:*] - x2[0:npt-it-1])
FOR it = 0, ntau-1 DO V3[it] = Stddev(x3[it:*] - x3[0:npt-it-1])
: Hurst exponent from log-log (low end) gradient divide 2.
H1 = Alog10(v1[100]/v1[1])/2.
H2 = Alog10(v2[100]/v2[1])/2.
H3 = Alog10(v3[100]/v3[1])/2.

```

Example. 22 : Code snippet (IDL) calculating Hurst exponent, eqn. (190) with  $p = 2$  - cf. fig. 37.

### 11.3.2 Rescaled range analysis

The rescaled range  $R/S$  is computed by creating from the signal  $x$  of length  $N$  a series of sub-sets of  $\{x_n\}$  of  $(0, \dots, n)$  of progressively shorter lengths  $n = N, N/2, N/4, \dots$ . Each sub-set is then converted to a cumulative deviate about their mean

$$z_t = \sum_{i=1}^t (x_i - m) \quad (191)$$

for  $t = 1, \dots, n$  and  $m = (1/n) \sum_{i=1}^n x_i$ . The range  $R(n)$  for each sub-set is computed as

$$R(n) = \max(z_1, \dots, z_n) - \min(z_1, \dots, z_n) \quad (192)$$

which is then normalized by its corresponding standard deviation  $S(n) = [1/n \sum_{i=1}^n (x_i - m)^2]^{1/2}$ . As  $n$  is progressively halved the number of sub-sets  $m$  of length  $n$  doubles, and these sub-sets can of course be averaged to give  $R(n)/S(n) = \langle R(n)/S(n) \rangle_m$ .

### 11.3.3 Fractal dimension D

The Fractal dimension  $D$  is more familiar as an index of pattern complexity and the topic of self-similarity. In the context of a time or spatial series the fractal dimension  $1 < D < 2$  is related to the Hurst exponent such that  $D = 2 - H$  where  $H$  varies between 0 and 1. Higher values of  $D$  indicate smoother trends. There are a large number of publications on the dimensionality of edge plasma turbulence in tokamaks etc. As a starting point see <sup>139</sup> for an example of anomalously high  $D$  factors from density fluctuations in JET, and <sup>140</sup> for a detailed discussion and references. Also multifractal analysis of MAST and Tore Supra edge turbulence via embedding functions <sup>141</sup>

<sup>139</sup> van den Brink A.M. et al *Dimensionality analysis of chaotic density fluctuations in tokamaks*, 17th EPS Conf. **14B** I-199 (1990).

<sup>140</sup> Budaev V. et al. *Effect of rotating helical magnetic field on the turbulence fractal structure and transport in the tokamak edge plasma*, Nucl. Fusion **44** S108 (2004).

<sup>141</sup> Rajković M. et al *Characterization of local turbulence in magnetic confinement devices*, Nucl. Fusion **48** 024016 (2008).

## 11.4 Nonlinear correlation functions

When dealing with signals with a non-linear amplitude behaviour, aside from data transformations, there are a few non-linear correlation methods reported in the literature - ranging from fairly simple to the exceedingly complicated. One of the simpler approaches uses the concept of partial moments.<sup>142</sup> Here, 4 partial moments are defined:  $C_L$  and  $C_U$  are the lower and upper co-partial moments, for when the data  $x(i)$  is below a target value  $l$  or above some target  $h$  value, and  $D_L$  and  $D_U$  are the corresponding divergent partial moments. Which moment is appropriate depends on whether  $x$  and  $y$  are above or below their respective targets - see reference<sup>142</sup> for further details.

$$C_L(p, h_x|h_y, x|y) = 1/N \sum_{n=1}^N [\max(0, h_x - x_n)^p \cdot \max(0, h_y - y_n)^p] \quad (193)$$

$$C_U(q, l_x|l_y, x|y) = 1/N \sum_{n=1}^N [\max(x_n - l_x, 0)^q \cdot \max(0, y_n - l_y)^q] \quad (194)$$

$$D_L(q|p, h_x|h_y, x|y) = 1/N \sum_{n=1}^N [\max(x_n - h_x, 0)^q \cdot \max(0, h_y - y_n)^p] \quad (195)$$

$$D_U(p|q, h_x|h_y, x|y) = 1/N \sum_{n=1}^N [\max(0, h_x - x_n)^p \cdot \max(y_n - h_y, 0)^q] \quad (196)$$

where  $p$  and  $q$  are the degree. In practice one might set  $p = q = 2$  and the targets  $h = l$  to either zero or the  $x$  and  $y$  signal means etc. The correlation, linear or nonlinear, is then given by

$$\rho_{xy} = \frac{C_L(p, h_x|h_y, x|y) - D_L(p, h_x|h_y, x|y) - D_U(p, h_x|h_y, x|y) + C_U(p, h_x|h_y, x|y)}{C_L(p, h_x|h_y, x|y) + D_L(p, h_x|h_y, x|y) + D_U(p, h_x|h_y, x|y) + C_U(p, h_x|h_y, x|y)} \quad (197)$$

The second approach are algorithms based on reconstructing vector dynamics from scalar time series. They generally draw upon the formalism of strange attractors and involve the construction of finite phase spaces from the signals using embedding techniques.<sup>143</sup>

One such algorithm was developed by Ding<sup>144</sup> with the aim of resolving or quantify long-range couplings and the ‘direction’ of coupling when the correlation is weak. The formulation is rather complex and is not reproduced here and you are referred to the paper for further details.

Other useful references include<sup>145</sup> which presents an alternative parametric approach to the dynamical reconstruction.

<sup>142</sup> Violen F. *Deriving nonlinear correlation coefficients from partial moments*, SSRN 2148522 ([https://papers.ssrn.com/sol3/papers.cfm?abstract\\_id=2148522](https://papers.ssrn.com/sol3/papers.cfm?abstract_id=2148522)) (2012).

<sup>143</sup> Takens F. *Dynamical systems and turbulence* Springer-Verlag, N.Y. p366 (1981).

<sup>144</sup> Ding W.X. *Nonlinear radial correlation of electrostatic fluctuations in the STOR-M tokamak*, Phys. Rev. Lett. **79** (13) 2458 (1997).

<sup>145</sup> Aguirre L.A. *A nonlinear correlation function for selecting the delay time in dynamical reconstructions*, Phys. Lett. A **203** 88 (1995).

## 12 Conditional and non-stationary analysis

Many non-stationary signals exhibit strong, repetitious but intermittent events. Such features commonly arise in the edge/scrape-off layer of fusion plasmas due to filaments and density blobs, or the semi-regular ELM events in the H-mode edge plasma. The non-stationarity of these signals, observed in the non-time-invariance of the amplitude pdfs  $p(x)$ , means that event dominated signals require special treatment and analysis techniques. The basic characterization of intermittency falls into two categories: (1) the variability or distribution in event amplitude, and (2) the timing or frequency of events, and specifically the **clustering** of events. It is understood that both parameters maybe spatially dependent. The section starts with a review of conditional, or event triggered techniques for amplitude analysis, followed by event timing characterisation methods.

### 12.1 Event amplitude

Concerning event amplitude, the basic signal pdf and its moments are good indicators of intermittency, specifically the pdf skewness as well as pdf tails resulting in large kurtosis values - as well as non-time-invariance, that is the moments vary with time. Inter-dependencies between the moments, such as between skewness and kurtosis behaviour are also commonly observed.

#### 12.1.1 Conditional probability density functions

The conditional pdf is best described as the pdf of  $x$ , given condition  $y$ <sup>146</sup> and is often defined as the ratio of two unconditional probabilities, the joint probability, see section 3, over the condition probability (which must be greater than zero).

$$p(x | y) = \frac{p(x, y)}{p(y)} \quad (198)$$

$$= \frac{p(x \cap y)}{p(y)} \quad (p(y) > 0). \quad (199)$$

The condition can take many forms, such as the signal being greater than a certain value, or its statistics satisfying a certain requirement.<sup>147</sup> Such conditionals are straightforward to implement.

#### 12.1.2 Conditional (box-car) averaging

Conditional averaging (CA), aka box-car averaging, defines an event “averaged” signal as

$$\langle x(\tau) \rangle_{ca} = \langle x(t + \tau) | y(t) = y_c \rangle \quad (200)$$

$$= \frac{1}{N} \sum_{n=1}^N [x_{n+\tau} | y_n = y_c] \quad (201)$$

where  $\tau$  is a time lag and  $y_c$  is the condition (eg. a trigger threshold) signal.

<sup>146</sup> Hájek A. *What conditional probability could not be*, Synthese **137** 273 (2003).

<sup>147</sup> Bengston R.D. et al. *Study of plasma edge turbulence via conditional probability density functions*, Proc. 22nd EPS Conf. vol.1-281 (1995).



The trigger condition can be a simple threshold crossing, or quite complicated, such as a range, or a gate, eg.  $y_{c1} \leq y(t) \leq y_{c2} \wedge \delta y(t) \geq 0$ . The trigger can be derived from the signal  $x(t)$  itself, or from a separate measured signal from a different location, or even from an external trigger signal.<sup>148 149 150</sup> The CA essentially returns the average temporal evolution of a coherent feature in the signal, but loses, of course, the dynamics of individual events. In addition, the resulting ‘shape’ may depend on the trigger gate threshold/window range, thus scanning the window may be needed to recover the event shape dependences on the event (peak) amplitude.

The analysis procedure is usually a two-step process:

(1) Build the trigger signal by scanning the conditional signal for positions where the condition  $y(t) = y_c$  is met, marking the trigger positions  $t_c$  with a 1.

(2) Sum the data segments around each trigger region of interest:  $t_c \pm \tau_{\max}$ .

Particular attention is generally required to handle overlapping segments and multiple triggers within an ensemble data segment etc.

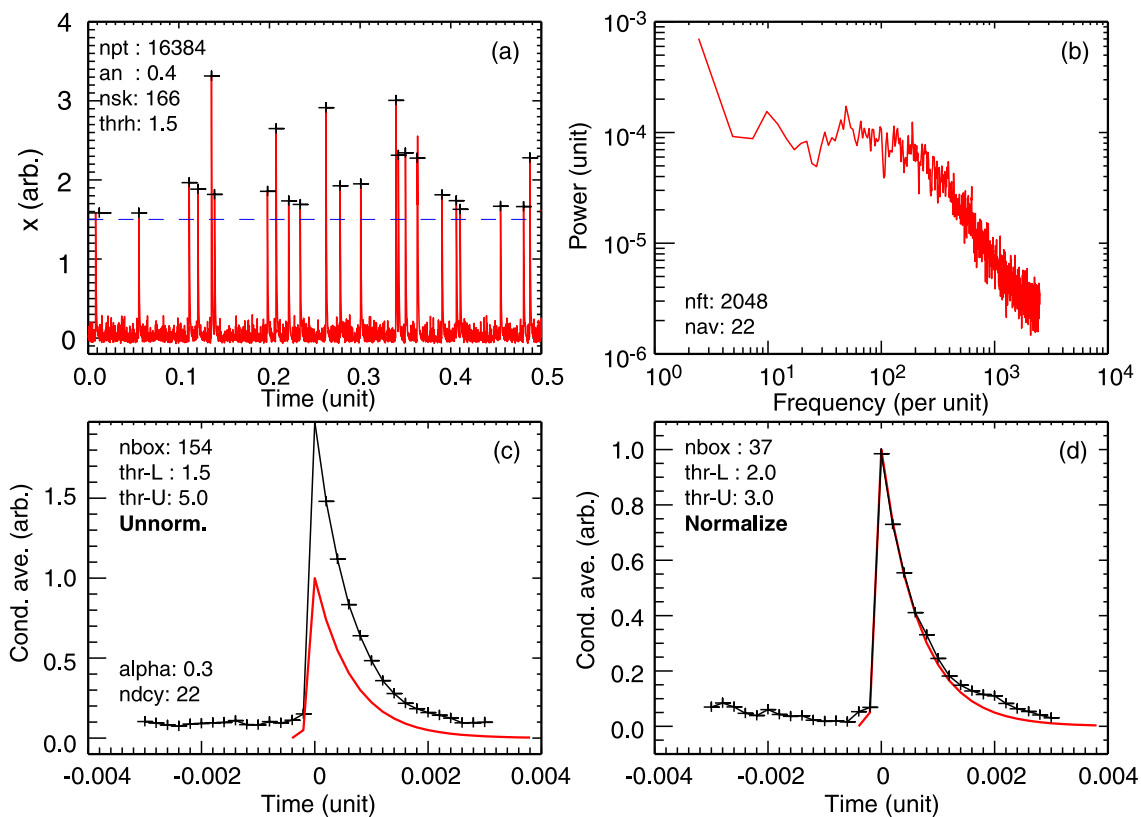


Figure 38: Example of box-car averaging of (a) an ELM-like intermittent signal with (b) spectra and (c) wide threshold gate and non-normalized ensembles, and (d) a narrower gate with normalized ensembles.

<sup>148</sup> Huld T. et al *Coherent structures in two-dimensional plasma turbulence*, Phys. Fluids B **3** (7) 1609 (1991).

<sup>149</sup> Filippas A.V. et al *Conditional analysis of floating potential fluctuations at the edge of the Texas Experimental Tokamak Upgrade (TEXT-U)*, Phys. Plasmas **2** (3) 839 (1995).

<sup>150</sup> Teliban I. et al *Improved conditional averaging technique for plasma fluctuation diagnostics*, Plasma Phys. Control. Fusion **49** 485-497 (2007).

Figure 38 shows examples of box-car averaging using the ELM-like signal of section 2.6 and the approach illustrated in the code snippet below. In fig. 38(c) and (d) two different upper and lower threshold gates are used to select the events to be averaged. Much care needs to be given in designing the threshold condition code to avoid false triggers, that is, to ensure that only the real peaks are selected and not all points within the gate range. In the code snippet the extra `shift(sig,1)` command in the condition statement tries to ensure that only upward transitions across the lower threshold are included.

The main difference between the two cases shown is that in (c) the detected peak traces are ensemble averaged as they are, while in (d) each peak is normalized to unity before adding to the ensemble. The result is that in (c) more weight is given to the larger peaks while in (d) all peaks have equal weight. Nevertheless, in both case the ELM shape is faithfully recovered because each peak has the same  $\alpha$  decay rate. In real signals this can not be assumed and scanning the gate conditions will be necessary. Note also in the normalized case the background level prior to the peak is affected by the tail of close (almost overlapping) preceding peaks.

```

bsz = 15 & nb2 = 2 & dt = 1/5000.           ; +/- box-car size      .
tLo = 1.8 & tHi = 3.0                       ; thresholds
itc = Where(sig GT tLo AND shift(sig,1) LT tLo AND sig LT tHi, nbc)
bca = Fltarr(2*bsz+1)
FOR i = 2, nbc-5 DO BEGIN                     ; sum events nb. -5 to
  mxbc = Max(sig[itc[i]-bsz:itc[i]+bsz])     ; prevent overrun
  bca = bca + sig[itc[i]-bsz:itc[i]+bsz] / mxbc
  nb2 += 1
ENDFOR
bca /= Float(nb2)                            ; normalize no. events
ibc = Findgen(2*bsz+1)-bsz                   ; time lag array
; plot ibc*dt, bca

```

*Example. 23 : Code snippet (IDL) demonstrating box-car detection and averaging - cf. fig. 38.*

### 12.1.3 Conditional & Gated correlations

Not just conditional averaging is possible, but also conditional correlation, or gated correlation. Here, the cross-correlation is of the form:

$$\langle R_{xy}(\tau) \rangle_c = \langle x(t)y(t+\tau) | z(t) = z_c \rangle \quad (202)$$

$$R_{xy}(n) = \frac{1}{N'} \sum_{n=0}^{N'-1} [x_n y_{n+\tau} | z_i = z_c] \quad (203)$$

where again the trigger signal can be the same as either one of the correlated signals (auto or cross) etc. One can also think of the gated-correlation as a triple correlation with the 3rd signal the gate (0 or 1) signal. The length of the correlation  $N' < N$  is restricted to around the event, as well as the correlation lag  $\tau = k\Delta t$ , where  $K < N'$  etc. to reveal time delays/propagation of event appearance in spatially separated channels, or the mutation of propagating events, such as filaments.

Another analysis option is the time dependent covariance - eqn. (45) - which would give a cor-

relation or covariance map vs delay  $\tau = k\Delta t$  at signal time  $t = n\Delta t$  between  $x(t)$  and  $y(t)$ . As above, the expectation (summation) would be performed over a narrow temporal window which is moved/slid through the data window.

### 12.1.4 Wavelet analysis

If the basic temporal shape of the event is fairly constant (like a heartbeat for example) but is variable in amplitude and time dilation, then one could form a wavelet basis function from the event shape (which should satisfy the basic wavelet criteria) - see section 10.5 and footnotes - to investigate the spectral / time distribution of the events. This of course also leads on to the wider topic of pattern recognition etc.

## 12.2 Event timing

### 12.2.1 Level crossing

Signal level crossing, and specifically zero crossing, is an easy method to analyze event timing. Here, the approach is to remove the effect of the signal amplitude and measure just the frequency of events by via the points where the signal crosses some threshold. The amplitude and frequency can be separated by transforming the signal into a telegraphic approximation (TA) - also called a binary approx. BA - signal consisting of only zeros or ones,<sup>151 152</sup>

$$\xi(t) = \frac{1}{2} \left( \frac{x'(t)}{|x'(t)|} + 1 \right) \tag{204}$$

where  $x'_n = (x_n - m_l)$  with  $m_l$  is the threshold level. In the case of zero-crossing  $m_l$  would be the signal mean.

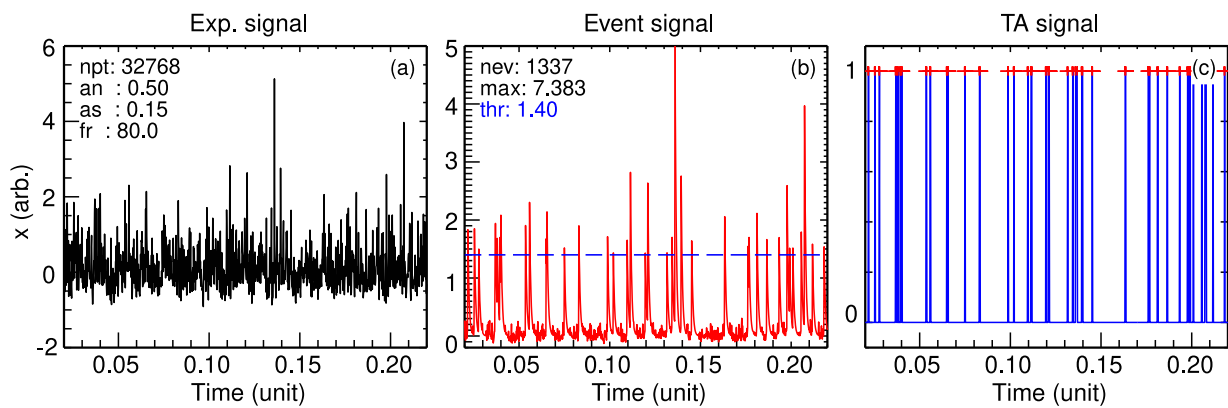


Figure. 39: Example of the (c) telegraphic approximation (TA) signal created from (b) a semi-intermittent ELM-like signal driven by (a) an exponential signal from Gaussian white noise plus a coherent sinewave. Red crosses in (c) mark binary transitions.

<sup>151</sup> Sreenivasan K.R. et al *Clustering properties in turbulent signals*, J. Stat. Phys. **125**(5/6) 1145 (2006).

<sup>152</sup> Bershadskii A. et al “*Clusterization*” and intermittency of temperature fluctuations in turbulent convection, Phys. Rev. E **69** 056314 (2004).

Figure 39 shows an example of the TA signal (right) created from a semi-intermittent event signal (ELM-like: middle) formed from an exponential signal plus a low frequency sinewave (left) - as described in section 2.6. In this example the trigger threshold is set fairly high ( $\text{thr} = 1.4$ ) to create a sequence of large amplitude, low frequency ELM-like events. The sinewave amplitude is also adjusted such that a degree of semi-periodicity is imposed on some of the, otherwise random, events. The behaviour of the TA  $\xi(t)$  signal contains much novel information.

### 12.2.2 Spectral exponent

One of the first tests is the spectral exponent.<sup>153 154</sup> Figure 40 shows the results of a collection of analysis methods for two threshold event generation levels. Top row: small, frequent events ( $\text{thr} = 0.6$ ) and bottom row large, infrequent, semi-periodic events ( $\text{thr} = 1.6$ ). Panels (b) & (d) show respective event  $x(t)$  signal and TA  $\xi(t)$  signal spectra. For a stochastic process the spectral indexes for the event and TA signals are expected to be related by

$$m_{TA} = (m_{ev} - 1)/2. \quad (205)$$

In fig. 40(d) the observed  $m_{ev} \approx -5/3$  would thus translate to a  $m_{TA} = -4/3$ , which is indeed the case for small events ( $\text{thr} = 0.6$ ) were the signals are less skewed and closer to random. For the more periodic event case ( $\text{thr} = 1.6$ ), the TA spectral index seems closer to  $m_{TA} \approx -3/3$ .

### 12.2.3 Clustering exponent

Clustering of events is a common feature of intermittency and can be quantified by the number of (zero-crossing) transitions  $n_\tau$  in the interval  $\tau$ , ie. by counting when  $\xi(t) = 0 \rightarrow 1$  and  $\xi(t) = 1 \rightarrow 0$ . The clustering information is in the standard deviation of  $n_\tau$  about a running mean. In the code snippet below the data stream is broken into many, successively shorter, windows of lengths  $\tau$ , over which variations in the zero-crossing count  $\delta n_\tau = n_\tau - \langle n_\tau \rangle$  about the local (running) mean are squared, ensemble summed and then square-rooted to give the standard deviation as a function of window length  $\tau$

$$\langle \delta n_\tau^2 \rangle^{1/2} \sim \tau^\alpha \quad (206)$$

which is expected to scale with  $\tau$  to some exponent  $\alpha$ , called the **clustering exponent**. Thus plotting  $\log \langle \delta n_\tau^2 \rangle^{1/2}$  vs  $\log \tau$  should give a straight line over some  $\tau$  region or other. For Gaussian white noise  $\alpha = -1/2$ , ie. no clustering. Deviations from  $-1/2$ , specifically a smaller  $\alpha$  would indicate a preferential clustering of events at that  $\tau$  region. (Crudely speaking one could think of  $\tau$  as a proxy for scale.) Absence of clustering would thus indicate events are purely random. In fig. 41 one sees that for both low and high event threshold values that the  $\langle \delta n_\tau^2 \rangle^{1/2}$  scales solidly as  $\alpha = -0.5$  over several orders of  $\tau$  magnitude. Thus there is no evidence of event clustering in this synthetic data, even for the weakly semi-periodic event case.

<sup>153</sup> Rajković M. et al *Level-crossing function in the analysis of edge plasma turbulence*, Nucl. Fusion **49** 095016 (2009).

<sup>154</sup> Huang K.Y. et al *Velocity and temperature dissimilarity in the surface layer uncovered by the Telegraphic Approximation*, Boundary-Layer Metrol. **180** 385 (2021).

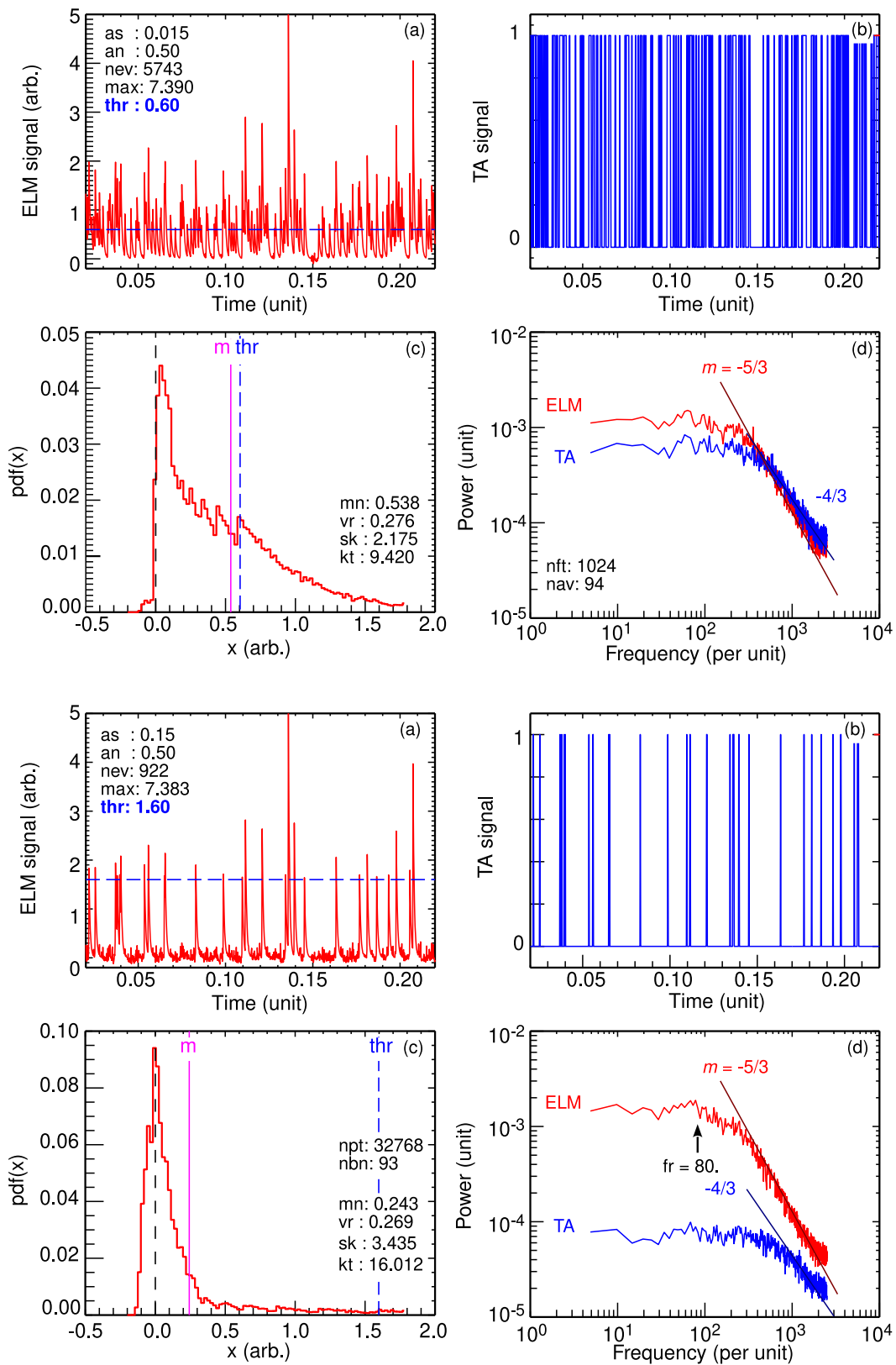


Figure. 40: Top row: Event threshold set for small events ( $thr = 0.6$ ) and bottom row: large events ( $thr = 1.6$ ). (a) Event/ELM signal, (b) TA signal, (c) amplitude pdf of ELM/event signal, and (d) ELM and TA power spectra.

```

npt = 32768L & dt = 1./5000. & thr = 1.2 & dfac = 1.35 .
tau = Fltarr(100) & sta = Fltarr(100) & dta = Fltarr(36000)
; Make TA and up-down transition signal
mlev = thr
xi = 0.5*((sig2-mlev)/Abs(sig2-mlev) +1.) ; TA
gxi = Abs(xi - shift(xi,-1))
itg = Where(gxi GT 0.9, ntl) ; Plot as crosses
; Clustering exponent
dtau = Float(npt)
idt = 0
WHILE dtau GT 2. DO BEGIN
  dtau = dtau/dfac ; Each loop / by dfac
  tau[idt] = dtau
  idtau = Fix(dtau)
  itp = 0
  ilp = 0
  WHILE itp LT npt-idtau-1 DO BEGIN
    b = Where(gxi[itp:itp+idtau-1] GT 0.9, ntlp)
    dta[ilp] = Float(ntlp)
    itp += idtau
    ilp += 1
  ENDWHILE
  sta[idt] = Stddev(dta[0:ilp-1]/dtau)
  idt += 1
ENDWHILE
tau = tau[0:idt-1] & sta = sta[0:idt-1] ; Trim arrays
; Waiting times
ite = Where(sig2 GT thr, ndt)
dtg = Float(ite - Shift(ite, 1))
ddn = 0. & ddx = Max(dtg[1:idt-1]) & nbin = 10>ddx
dx = Abs(ddx-ddn)/Float(nbin)
xbind = (Findgen(nbind)*dx+ddn)*dt
ybind = Histogram(dtg, Max=ddx, Min=ddn, Binsize=dx)/Float(ndt)

```

Example. 24 : Code snippet (IDL) for generating TA signal and for analyzing intermittent signals - cf. figs. 39, 40, 41 & 42.

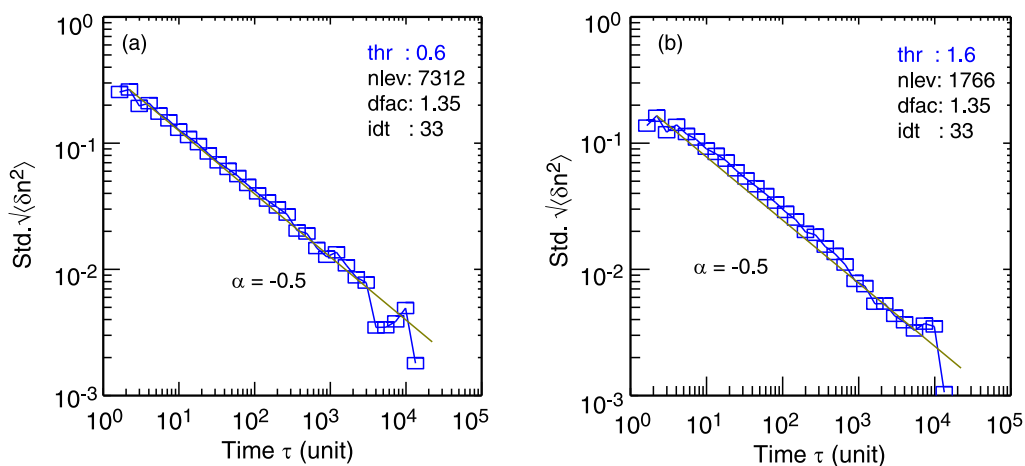


Figure. 41: Clustering exponent plot of ELM-like event signal for (a) event threshold set for small, frequent events ( $\text{thr} = 0.6$ ), and (b) large, infrequent events ( $\text{thr} = 1.6$ ).

### 12.2.4 Waiting times

Concerning event timing, one characterization technique is to measure the distribution and statistics of the event *waiting times*, the inter-event periods  $\Delta t_i = t_{i+1} - t_i$ , where  $t_i$  are the event times. One can use either a threshold detection on the raw event signal to identify and mark the onset time of each event, or use the TA signal and mark rising edges. The pdf of  $\Delta t_i$  is related to the event persistence, ie. the unchangingness of the signal.

Figure 42 shows the waiting time pdf of the ELM-like test signal as (a) a log-norm plot and (b) a log-log plot. If the pdf is *log-normal* - a characteristic of white noise - then the signal is termed passive.<sup>154</sup> On the other hand, if the pdf is more *log-log*, ie. it has a power-law scaling, then the fluctuation (turbulence) is classified as active. In fig. 42 the waiting time pdf (for the large threshold case) is clearly seen to be log-norm, as may be expected for random noise driven events. The red line in fig. 42(a) is an empirical fit to the log-norm plot:  $\log(\text{pdf}(\Delta t_i)) = \Delta t_i^{-0.5} * \Delta t_s / \text{thr}$  where  $\Delta t_s$  is the signal sample period and *thr* the signal amplitude threshold level. Note the  $-0.5$  in the  $\Delta t_i$  exponent.

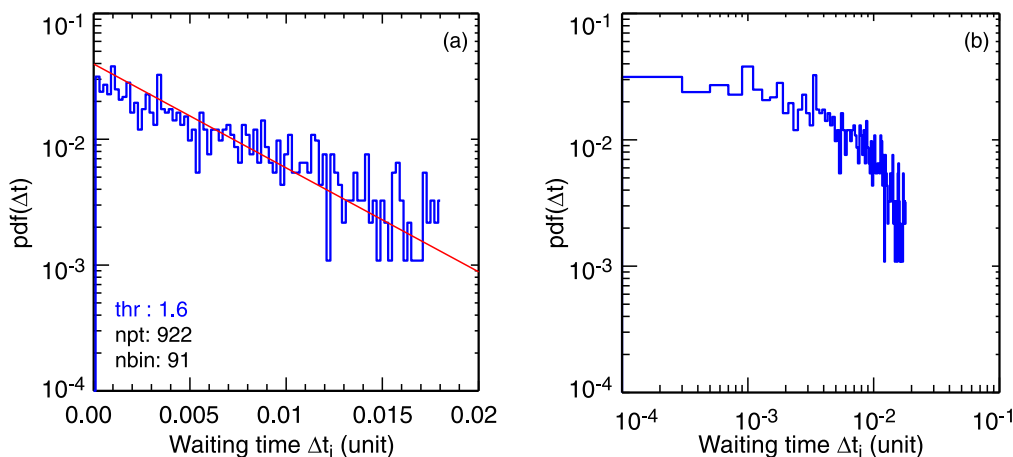


Figure. 42: Pdf plots of waiting times of ELM-like signal with event threshold for large, infrequent events ( $\text{thr} = 1.6$ ) with (a) log-normal and (b) log-log axes.

An extension to the waiting time analysis is of course to build the event interval  $\text{pdf}(\Delta t_i)$  as a function of the event amplitude - as in fig. 41 for the clustering exponent. However, as with the cluster exponent, the waiting time pdf in this test case was generally invariant with event amplitude since the events were all generated from the same random exponential signal.

There are many publications in this area. Some recent plasma turbulence references include<sup>155</sup>  
156 157

<sup>155</sup> Cipciar D. et al *Statistical properties of ion and electron temperature fluctuations in the edge of the COMPASS tokamak*, Plasma Phys. Control. Fusion **64** 055021 (2022).

<sup>156</sup> Theodorsen A. et al *Statistical properties of a filtered Poisson process with additive random noise: distributions, correlations and moment estimation*, Phys. Scr. **92** 054002 (2017).

<sup>157</sup> van Milligen B.P. & Sanchez R. *Complex features of plasma turbulence (Ch.3 of Analysis of Turbulence in Fusion Plasmas)*, IOP Publishing UK (2022).

## General further reading

- [1] Bendat J.S. and Piersol A.G. *Random data: Analysis and measurement procedures*, 4th edn. Wiley, N.Y. (2010). First published in 1971
- [2] Bevington P.R. et al *Data reduction and error analysis for the physical sciences*, McGraw-Hill, 2nd ed. (1992).
- [3] Chatfield C. *The analysis of time series: An introduction*, 6th edn. Chapman & Hall, New York (2004)
- [4] Haykin S. and others *Nonlinear methods of spectral analysis*, 2nd edn. Springer (1983)
- [5] Marple, S.L. *Digital spectral analysis with applications*, Prentice-Hall (1987)
- [6] Meko D.M. *GEOS585A, Applied time series analysis : Course notes*, Univ. of Arizona (2019). Download from <http://www.ltrr.arizona.edu/~dmeko/>
- [7] Mitra S.K. *Digital signal processing*, 2nd edn. McGraw-Hill (2001)
- [8] Oppenheim A.V. & Schafer R.W. *Discrete-time signal processing*, Prentice Hall (1989)
- [9] Otnes R.K. & Enochson L. *Applied time series analysis*, Wiley N.Y. (1978)
- [10] Proakis J.G. & Mandakis D.G. *Digital signal processing*, 4th edn. Pearson (2006)
- [11] Press W. et al *Numerical recipes in Fortran - The art of scientific computing*, 2nd edn. Cambridge Uni. Press (1992)
- [12] Shumway R.H. & Stoffer D.S. *Time series analysis and its applications, with R examples*, 3rd edn. Free texts in Statistics (2014). Subsequently published as *Time series, A data analysis approach using R*, Chapman & Hall (2019)
- [13] Takens F. *Dynamical systems and turbulence* Springer-Verlag, N.Y. p366 (1981)
- [14] van Milligen B.P. & Sanchez R. *Analysis of Turbulence in Fusion Plasmas*, IOP Publishing UK (2022)