

Auto-correlation function estimation considering noise reduction

A numerical study on (damped) sinusoidal signals in noise

Jakob Harden (Graz University of Technology)

21st of December, 2024

In traditional signal analysis, signal power and signal-to-noise ratio are estimated using the signal's stochastic properties. Those are related to a signal's auto-correlation function (ACF). Assuming that signal and noise are uncorrelated wide-sense stationary stochastic processes, the signal power can be obtained by subtracting the noise power from the signal's power in noise. This demands auxiliary data exclusively consisting of noise.

However, there are situations where signal areas that only contain noise are difficult to identify or only little data is available. In such cases, considerable losses in accuracy may have to be accepted when estimating the signal power.

The method introduced here promises to significantly reduce the influence of noise in ACF estimation without the need for auxiliary noise data. The technique involves replacing the ACF magnitudes around the Center-lag area with a regression polynomial. In its initial design, the method applies to damped, sinusoidal signals.

The numerical study presented below yields promising results. It demonstrates that the method can provide reliable signal power estimates for typical applications, making it a practical and valuable tool.

► WHAT

- estimate ACF of damped, sinusoidal signals
- reduce impact of noise in the ACF without using auxiliary noise data

► WHY

- noise-only signal sequence cannot be clearly identified
- little or no auxiliary noise data available

► HOW

- estimate ACF, unbiased estimator
- replace ACF samples around the center-lag with a discrete-sampled polynomial

► USAGE

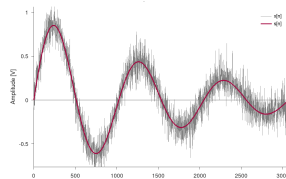
- signal power estimates
- signal-to-noise ratio estimates

► HIGHLIGHT **no auxiliary noise data necessary**

► Materials

- damped, sinusoidal signals in noise (synthetic signals)
- $x[n] = s[n] + \nu[n]$, $\nu \dots$ i.i.d. noise (GWN)
- $s[n] = A \cdot \sin(2 \pi \frac{n}{N_1}) \cdot e^{-DF \frac{n}{N}}$
- $N = N_1 N_{cy}$, $n = 0, \dots, N - 1$
- $A \dots$ signal amplitude, Volts
- $N_1 \dots$ signal period, samples
- $N_{cy} \dots$ number of cycles
- $DF \dots$ exponential decay factor

Damped, sinusoidal signal in noise



► Methods

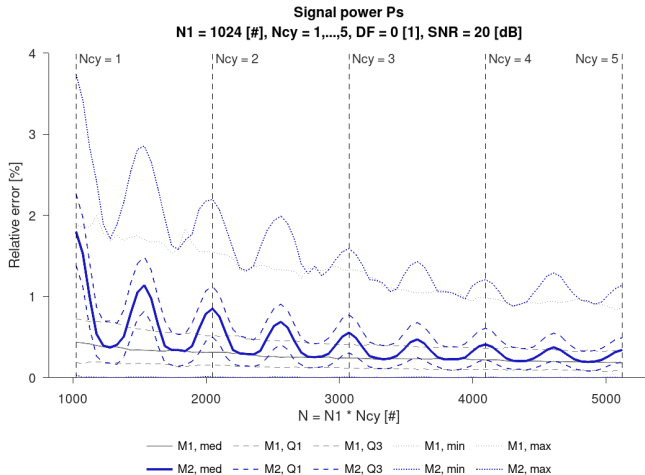
- ACF estimation (unbiased estimator)
- Regression polynomials of 2nd order
- Parameter variation (period, cycles, decay, SNR, noise)
- Monte-Carlo test to justify the impact of random noise
- Method 1 (M1): estimate signal power using auxiliary noise data (classic approach)
- **Method 2 (M2): estimate signal power using the reduced-noise auto-correlation function (ACF-RN)**
- Comparison of M1 and M2 w.r.t. signal power estimates

Note: Fundamental idea and algorithm of method 2 are detailed in the Appendix on page 22ff. Signal power and SNR estimates, please find in the Appendix on page 30 and 31.

To study the algorithm, a parameter variation was carried out using the following parameters.

► Numerical study - parameter variation

- Constant: $A = 1$ [V]
- Constant: unbiased ACF estimator
- Variation 1: $N_1 = (64, 512, 1024, 4096)$ samples
- Variation 2: $SNR = (5, 10, 15, 20)$ dB
- Variation 3: $DF = (0, 2, 4)$
- Variation 4: $N_{cy} = (1, \dots, 5)$ samples, subdivided into 81 steps
- Variation 5: $N_{mc} = 500$ turns, Monte-Carlo test

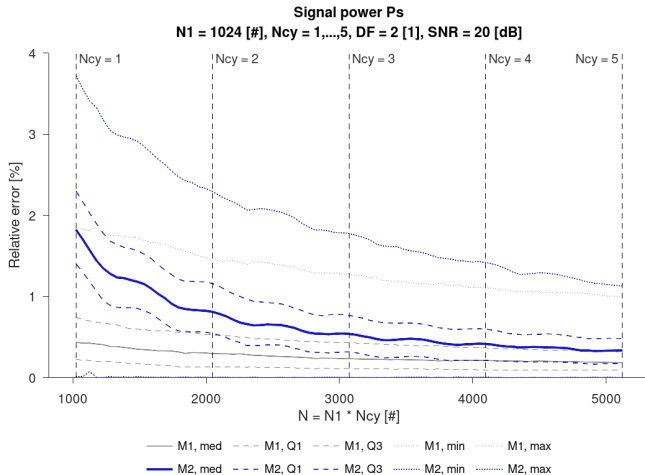


Test signals

Sinusoidal signals, low SNR, no decay

Observations

- ▶ significant ripple in M2 (ACF-RN) due to non-coherent sampling
- ▶ Error becomes smaller with increasing N
- ▶ M2 is higher but converging against M1 with increasing N

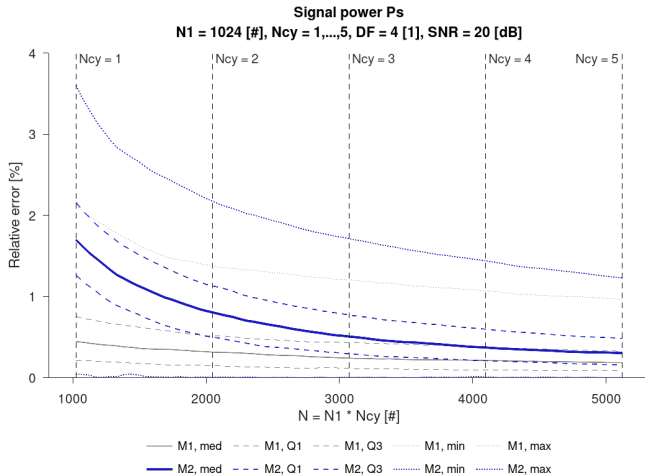


Test signals

Damped sinusoidal signals,
low SNR, moderate decay

Observations

- ▶ Ripple in M2 vanishes with increasing decay due to the attenuation of the signal's end.
- ▶ Convergence behavior remains

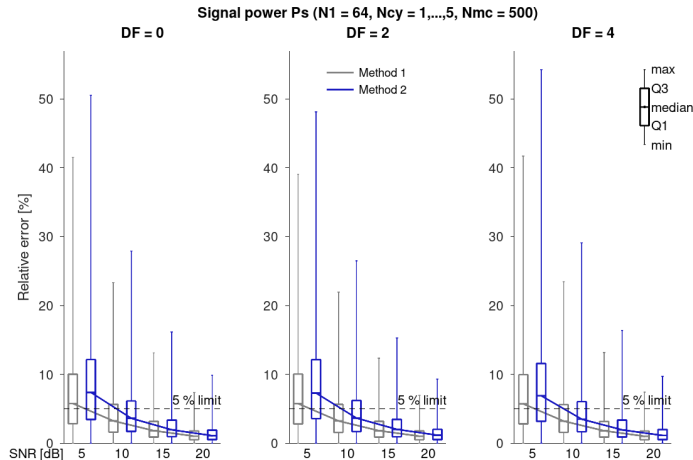


Test signals

Damped sinusoidal signals,
low SNR, strong decay

Observations

- ▶ Ripple in M2 is almost gone
- ▶ Convergence behavior remains

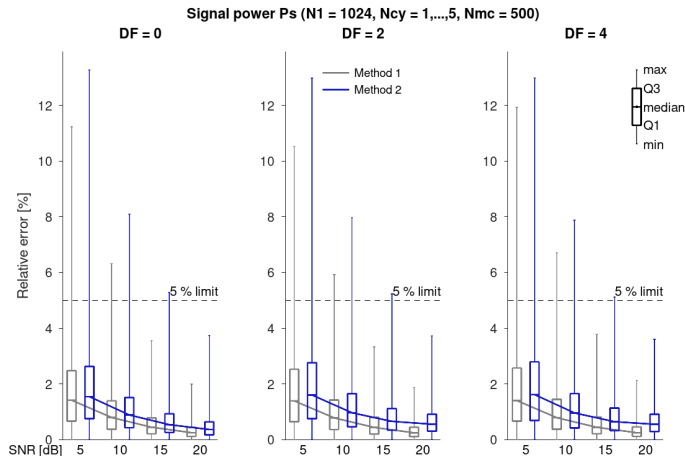


Test signals

Small amount of signal data

Observations

- ▶ M1 and M2 show similar behavior
- ▶ Error of M2 is always above error of M1
- ▶ $\text{Max} \gg \text{Q3}$
- ▶ Decay: minor impact
- ▶ SNR: significant impact



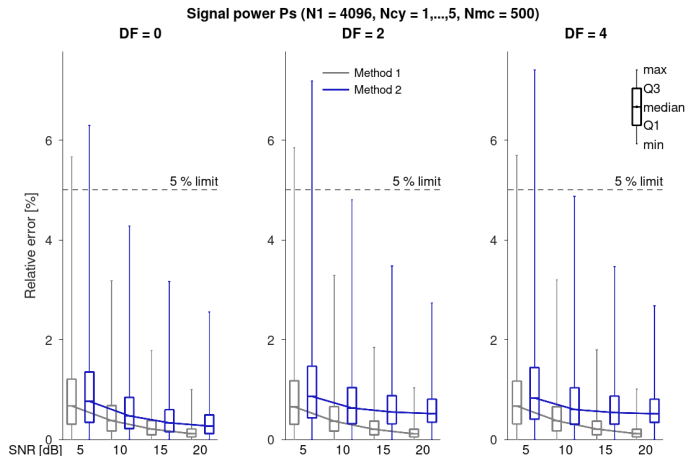
CC BY-4.0 Jakob Harden (Graz University of Technology), 2024

Test signals

Moderate amount of signal data

Observations

- ▶ Error decreases with increasing period N_1 and SNR
- ▶ Distribution spread: $M_2 > M_1$
- ▶ Q3 of error drops below 5 % limit



Test signals

High amount of signal data

Observations

- ▶ Max error drops below 5% for $SNR \geq 10$ [dB]
- ▶ The main part of all distributions (Q1-Q3) is closer to the Min than to the Max

		Range									N1 [#]		
		0 - 5			5 - 10			> 10			[%]		
SNR [dB]	64			256			1024			4096			DF [1]
	0	2	4	0	2	4	0	2	4	0	2	4	
5	50.5	48.1	54.2	24.2	25.0	24.8	13.3	13.0	13.0	6.3	7.2	7.4	
10	27.9	26.5	29.1	13.8	14.2	14.0	8.1	8.0	7.9	4.3	4.8	4.9	
15	16.2	15.3	16.4	8.4	8.6	8.4	5.3	5.2	5.1	3.2	3.5	3.5	
20	9.9	9.3	9.7	5.5	5.6	5.4	3.7	3.7	3.6	2.6	2.7	2.7	

Method 2 - ACF-RN

- Maximum estimation error

Method 1 - classic

64			256			1024			4096		
0	2	4	0	2	4	0	2	4	0	2	4
41.5	39.0	41.7	25.0	28.8	27.9	11.2	10.5	11.9	5.7	5.8	5.7
23.3	22.0	23.4	14.1	16.2	15.7	6.3	5.9	6.7	3.2	3.3	3.2
13.1	12.3	13.2	7.9	9.1	8.8	3.6	3.3	3.8	1.8	1.8	1.8
7.4	6.9	7.4	4.5	5.1	5.0	2.0	1.9	2.1	1.0	1.0	1.0

		Range												
		0 - 5			5 - 10			> 10			[%]			
SNR [dB]	64			256			1024			4096			N1 [#]	
	0	2	4	0	2	4	0	2	4	0	2	4	DF [1]	
5	20.5	20.7	20.3	9.9	10.2	10.2	5.0	5.2	5.0	2.5	2.7	2.7		
10	10.6	10.9	10.8	5.5	5.8	5.8	2.9	3.1	3.1	1.6	2.0	1.9		
15	5.9	6.1	6.0	3.2	3.5	3.4	1.8	2.1	2.1	1.2	1.6	1.6		
20	3.4	3.6	3.6	2.0	2.3	2.3	1.3	1.7	1.7	1.1	1.5	1.5		

Method 2 - ACF-RN

- 95%-percentiles of the estimation error

Method 1 - classic

64			256			1024			4096		
0	2	4	0	2	4	0	2	4	0	2	4
17.9	10.1	10.1	9.3	9.2	9.4	4.7	4.6	4.5	2.3	2.2	2.1
10.1	10.2	10.2	5.2	5.2	5.3	2.6	2.6	2.5	1.3	1.3	1.2
5.7	5.7	5.7	2.9	2.9	3.0	1.5	1.4	1.4	0.7	0.7	0.7
3.2	3.2	3.2	1.6	1.6	1.7	0.8	0.8	0.8	0.4	0.4	0.4

Range												
0 - 5			5 - 10			> 10			[%]			
SNR [dB]	64			256			1024			4096		
	0	2	4	0	2	4	0	2	4	0	2	4
N1 [#]	DF [1]											
5	12.2	12.1	11.6	5.5	5.6	5.6	2.6	2.8	2.8	1.4	1.5	1.4
10	6.1	6.2	6.1	3.0	3.1	3.1	1.5	1.7	1.7	0.8	1.0	1.0
15	3.4	3.5	3.4	1.7	1.8	1.8	0.9	1.1	1.1	0.6	0.9	0.9
20	1.9	2.0	2.0	1.0	1.2	1.2	0.6	0.9	0.9	0.5	0.8	0.8

Method 2 - ACF-RN

- 75%-percentiles (Q3) of the estimation error

Method 1 - classic

64			256			1024			4096		
0	2	4	0	2	4	0	2	4	0	2	4
10.0	10.1	10.0	5.1	5.2	5.1	2.5	2.5	2.6	1.2	1.2	1.2
5.6	5.7	5.6	2.9	2.9	2.9	1.4	1.4	1.4	0.7	0.7	0.7
3.2	3.2	3.2	1.6	1.6	1.6	0.8	0.8	0.8	0.4	0.4	0.4
1.8	1.8	1.8	0.9	0.9	0.9	0.4	0.4	0.5	0.2	0.2	0.2

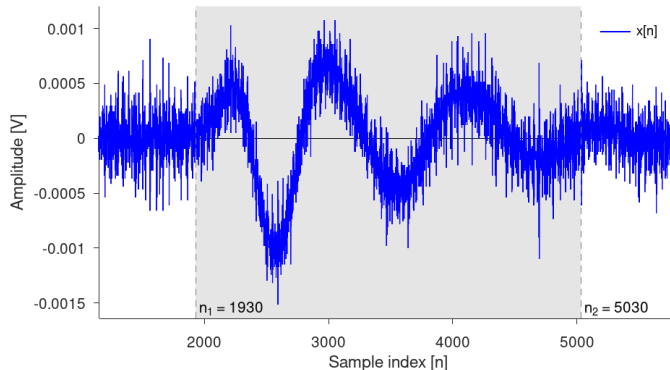
► Ultrasonic pulse transmission tests

- Test series 1 - Cement paste tests[3]
- Test method: ultrasonic pulse transmission method (UPTM)
- Data set: ts1_wc040_d70_1.oct
- Channel: P-wave (primary wave, compression wave)
- Sensors: piezoelectric contact probes, longitudinal characteristic
- Data acquisition: sampling rate = 10 MHz, vertical resolution = 16 bit

► Analysis

- signal power and signal-to-noise ratio estimates
- comparison between method 1 and method 2

Note: Auxiliary noise data is available from the first 1000 samples of each signal. The samples in this section are recorded before pulse excitation. Therefore, they consist of noise only. The signal power and signal-to-noise ratio estimation is explained in the Appendix on page 31.



CC BY-4.0 Jakob Harden (Graz University of Technology), 2024

UPTM signal

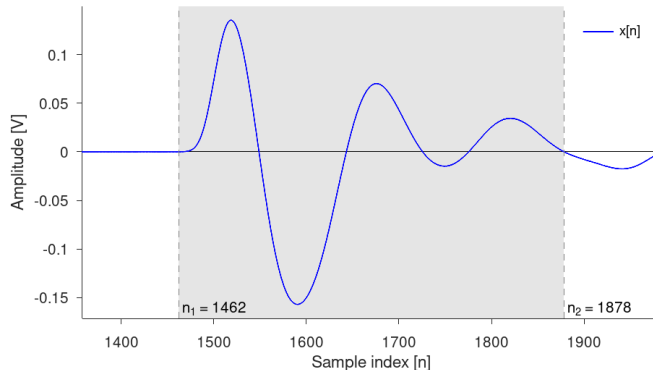
- ▶ Signal ID: 20
- ▶ Maturity: 102 [Min]

SNR estimates

- ▶ $\hat{SNR}_1 \approx 12.67$ [dB]
- ▶ $\hat{SNR}_2 \approx 12.14$ [dB]

Power estimates

- ▶ $\hat{P}_{s,1} = 1.75 \cdot 10^{-7}$ [V²]
- ▶ $\hat{P}_{s,2} = 1.55 \cdot 10^{-7}$ [V²]



CC BY-4.0 Jakob Harden (Graz University of Technology), 2024

UPTM signal

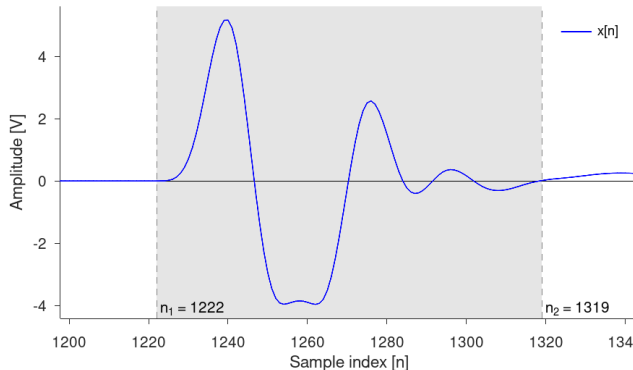
- ▶ Signal ID: 50
- ▶ Maturity: 252 [Min]

SNR estimates

- ▶ $\hat{SNR}_1 \approx 64.48$ [dB]
- ▶ $\hat{SNR}_2 \approx 64.51$ [dB]

Power estimates

- ▶ $\hat{P}_{s,1} = 4.73 \cdot 10^{-3}$ [V²]
- ▶ $\hat{P}_{s,2} = 4.76 \cdot 10^{-3}$ [V²]



CC BY-4.0 Jakob Harden (Graz University of Technology), 2024

UPTM signal

- ▶ Signal ID: 288
- ▶ Maturity: 1,442 [Min]

SNR estimates

- ▶ $\hat{SNR}_1 \approx 74.03$ [dB]
- ▶ $\hat{SNR}_2 \approx 74.06$ [dB]

Power estimates

- ▶ $\hat{P}_{s,1} = 5.21$ [V²]
- ▶ $\hat{P}_{s,2} = 5.25$ [V²]

With respect to the chosen parameter ranges, the results allow for the following conclusions:

- ▶ As expected, for both methods the power estimation error decreases with
 - increasing amount of signal data
 - increasing signal-to-noise ratio
- ▶ Method 1 (classic) and method 2 (ACF-RN) return comparable results
- ▶ ACF-RN power estimation error is slightly higher than of the classic method
- ▶ ACF-RN is influenced by coherent/non-coherent sampling
- ▶ **Highlights:** ACF-RN returns usable results **without using auxiliary noise data**.
Therefore, it is a suitable candidate for fully automated signal analysis approaches.

The ACF-RN method's basic principles have been defined. However, some questions remain unanswered and require further consideration.

- ▶ Adapt ACF-RN for other signal models
- ▶ Examine the difference between biased and unbiased ACF estimator
- ▶ Find ways to reduce the influence of coherent/non-coherent sampling
- ▶ Investigate the influence of the center-lag 'noise-peak' on the polynomial regression
- ▶ Impact of the ACF-RN on other problems (e.g. Wiener FIR filter design)

Note: To support the future development of the proposed method, the supplementary GNU Octave code[1] and the $\text{\LaTeX}$ code[2] of this presentation is made available publicly under open licenses.

- [1] Jakob Harden. *Auto-correlation function estimation considering noise reduction - supplementary GNU Octave code*. Dec. 2024. DOI: 10.3217/rvnf8-z5r12. URL: <https://doi.org/10.3217/rvnf8-z5r12>.
- [2] Jakob Harden. *Auto-correlation function estimation considering noise reduction - supplementary LaTeX code*. Dec. 2024. DOI: 10.3217/zkj4t-30097. URL: <https://doi.org/10.3217/zkj4t-30097>.
- [3] Jakob Harden. *Ultrasonic Pulse Transmission Tests: Datasets - Test Series 1, Cement Paste at Early Stages*. Version 1.0. Graz University of Technology, July 2023. DOI: 10.3217/bhs4g-m3z76. URL: <https://doi.org/10.3217/bhs4g-m3z76>.

The derivation shown below is trying to sketch the idea of the proposed method.

A common signal model for discrete-time signals in noise is

$$x[n] = s[n] + \nu[n], \quad n = 0, \dots, N-1 \quad (1)$$

where x is the signal in noise, s is the signal and ν the uncorrelated additive noise.

Under the assumption that signal and noise are wide sense stationary stochastic processes, the signal power can be estimated from the power of the signal in noise (variance σ_x^2) and the power of auxiliary noise data (variance σ_ν^2). This is the classical method which requires auxiliary noise data.

$$P_s = P_x - P_\nu = \underbrace{\frac{1}{N} \sum_{n=0}^{N-1} x[n]}_{\sigma_x^2} - \underbrace{\frac{1}{N} \sum_{n=0}^{N-1} \nu[n]}_{\sigma_\nu^2} \quad (2)$$

The formulation from above can also be written in terms of the auto-correlation function $R[m]$ at the center lag $m = 0$.

$$R_{ss}[0] = R_{xx}[0] - R_{\nu\nu}[0] = \sigma_x^2 - \sigma_\nu^2 \quad (3)$$

The fundamental idea of the proposed method is to avoid using auxiliary noise data in favour of estimating the signal power P_s directly from an approximation of the auto-correlation function of the signal $R_{xx}^{(rn)}$. Furthermore, $R_{xx}^{(rn)}$ is constructed from R_{xx} by replacing the magnitudes of R_{xx} in a close vicinity of the center-lag by a regression polynomial of second order.

$$P_s = R_{ss}[0] \approx R_{xx}^{(rn)}[0], \quad R_{xx} \mapsto R_{xx}^{(rn)} \quad (4)$$

Considering sinusoidal signals, the following explanations show why it makes sense to use second-order polynomials to construct $R_{xx}^{(rn)}$.

For a sinusoidal time-signal $s(t) = A \sin(t)$ the auto-correlation function R_{ss} is

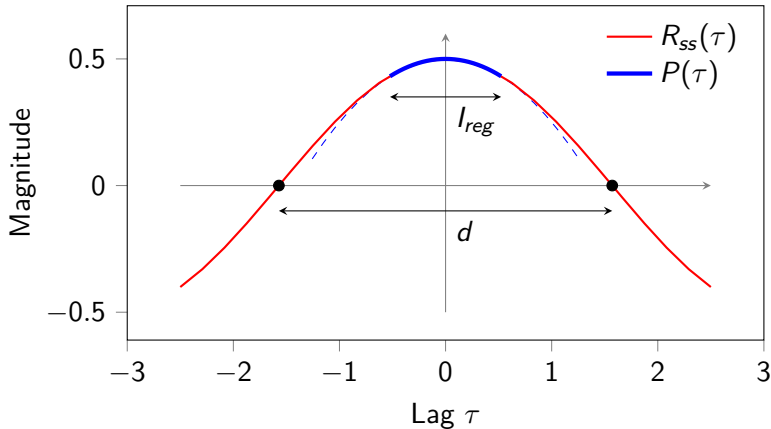
$$R_{ss}(\tau) = \int_{-\infty}^{+\infty} A \sin(t) \sin(t - \tau) dt = \frac{A^2}{2} \cos(\tau) \quad (5)$$

The Taylor expansion developed at the support $\tau = 0$ (center-lag of the ACF) is

$$\hat{R}_{ss}(\tau) = \frac{A^2}{2} \left(1 - \frac{\tau^2}{2} + \frac{\tau^4}{24} + \dots \right) \quad (6)$$

and the 2nd-order polynomial $P(\tau)$ that approximates R_{ss} in a close vicinity of $\tau = 0$ is

$$P(\tau) = \frac{A^2}{2} \left(1 - \frac{\tau^2}{2} \right) \quad (7)$$



Auto-correlation function of a sine ($A = 1$)

- ▶ $R_{ss}(\tau) = \frac{1}{2} \cos(\tau)$
- ▶ $P(\tau) = \frac{1}{2} - \frac{1}{4} \tau^2$
- ▶ Distance d between x axis crossings
- ▶ Symmetric interval l_{reg} with length $L_{reg} = \frac{d}{3}$
- ▶ $P(\tau)$ approximates $R_{ss}(\tau)$ in a close vicinity of $\tau = 0$

The algorithm, referred to as **reduced-noise auto-correlation function (ACF-RN)** is an approximation of the problem discussed before.

It consists of the following four steps:

- 1 Estimate unbiased $R_{xx}[m]$ of the signal in noise $x[n]$
- 2 Choose a symmetric interval I_{reg} around the center-lag $m = 0$
- 3 Fit 2nd-order polynomial to $R_{xx}[m]$ within I_{reg} by minimizing the mean-square error
- 4 Within I_{reg} , replace ACF samples with the sampled polynomial, $R_{xx} \mapsto R_{ss}^{(rn)}$

A realization of the algorithm is shown on page 28.

The algorithm does not claim to be more than an approximation because the following boundary conditions are not considered yet.

- ▶ Signals are usually not true sine functions (e.g. damped sinusoidal signals).
- ▶ Non-coherent sampling and limited amount of available signal data are significantly influencing the shape of the auto-correlation function.
- ▶ The impact of the center-lag's 'noise-peak' on the polynomial regression is increasing with decreasing signal-to-noise ratio and decreasing length of the regression interval.

$$n, m, N \in \mathbb{Z}$$

▷ index variables

$$P[m, c_P] \in P_2(\mathbb{R})$$

▷ discrete-sampled polynomial, 2nd order**Require:** $3 \leq m_1 \leq -N + 1$

▷ x axis crossings must be detectable

$$1: R_{xx}[m] \leftarrow \frac{1}{N-|m|} \sum_n x[n] \cdot x[n-m], m = -N+1, \dots, N-1$$

▷ ACF, unbiased

$$2: m_1 \leftarrow \sup_m \left(\arg(R_{xx}[m] < 0) \right), m = -N+1, \dots, 0$$

▷ lag of first x axis crossing

$$3: k_x \leftarrow R_{xx}[m_1 + 1] - R_{xx}[m_1]$$

▷ slope of line crossing the x axis

$$4: m_x \leftarrow m_1 - \frac{R_{xx}[m_1]}{k_x}$$

▷ estimate x axis crossing by linear interpolation

$$5: d \leftarrow 2 m_x + 1$$

▷ distance between first x axis crossings next to zero-lag

continued on next page ...

- 6: $R_{xx}^{(rn)} \leftarrow R_{xx}$
- 7: $e \leftarrow \lceil \frac{d}{6} \rceil, e \in \mathbb{Z}$ ▷ half length of regression interval (rounded to infinity)
- 8: $l_{reg} \leftarrow [-e, -e + 1, \dots, e], L_{reg} = 2e + 1$ ▷ regression interval with length L_{reg}
- 9: $c_P \leftarrow \arg \min_{c_0, c_1, c_2} (\frac{1}{L_{reg}} \sum_{m \in l_{reg}} (R_{xx}[m] - P[m, c_P])^2)$ ▷ see note below
- 10: $R_{xx}^{(rn)}[m] \leftarrow P[m, c_P], m \in l_{reg}$ ▷ replace ACF magnitudes with polynomial
- 11: **return** $R_{xx}^{(rn)}$

See also: GNU Octave code[1], function files *tool_acfrn.m* and *tool_est_acf_period.m*.

Note: In step 9, the sampled polynomial $P[m, c_P]$ is fitted to $R_{xx}[m]$ by minimizing the mean-square error. The result is the array of the polynomial's constants $c_P = [c_0, c_1, c_2]$, where $c_1 = 0$ as R_{xx} and l_{reg} are symmetric.

$$\hat{P}_x \leftarrow \frac{1}{N} \sum_{n=0}^{N-1} x[n]^2$$

▷ power of signal in noise, [V^2]

$$\hat{P}_\nu \leftarrow \frac{1}{N} \sum_{n=0}^{N-1} \nu[n]^2$$

▷ power of reference noise, [V^2]

$$\hat{P}_{s,1} \leftarrow \hat{P}_x - \hat{P}_\nu$$

▷ Method 1, signal power, [V^2]

$$\hat{P}_{s,2} \leftarrow R_{xx}^{(m)}[0]$$

▷ Method 2, signal power, [V^2]

$$\text{err}(\hat{P}_{s,1}) \leftarrow \frac{\hat{P}_{s,1} - P_s}{P_s} \cdot 100$$

▷ Method 1, relative signal power error, [%]

$$\text{err}(\hat{P}_{s,2}) \leftarrow \frac{\hat{P}_{s,2} - P_s}{P_s} \cdot 100$$

▷ Method 2, relative signal power error, [%]

See also: GNU Octave program code[1], function file *wrapper_acfrn_pvar.m*.

$$\nu[n] \leftarrow x[n], n = [0, \dots, N-1], N_\nu = 1000$$

▷ auxiliary noise data

$$\hat{P}_\nu \leftarrow \frac{1}{N_\nu} \sum_{n=0}^{N_\nu-1} \nu[n]^2$$

▷ auxiliary noise power, $[V^2]$

$$w[n] \leftarrow [n_1, \dots, n_2], L_w = n_2 - n_1 + 1$$

▷ measuring window, length L_w

$$\hat{P}_x \leftarrow \frac{1}{L_w} \sum_{n=n_1}^{n_2} x[n]^2$$

▷ power of windowed signal in noise, $[V^2]$

$$\hat{P}_{s,1} \leftarrow \hat{P}_x - \hat{P}_\nu$$

▷ Method 1, signal power estimate

$$\hat{P}_{s,2} \leftarrow R_{xx}^{(rn)}[0]$$

▷ Method 2, signal power estimate

$$\hat{SNR}_1 \leftarrow 10 \cdot \log_{10}(\hat{P}_{s,1}/\hat{P}_\nu)$$

▷ Method 1, signal-to-noise ratio estimate

$$\hat{SNR}_2 \leftarrow 10 \cdot \log_{10}(\hat{P}_{s,2}/\hat{P}_\nu)$$

▷ Method 2, signal-to-noise ratio estimate

See also: GNU Octave code[1], function file *test_acfrn.m*, sub-function *test_acfrn_nat(...)*.

First name Jakob
Last name Harden
Affiliation Graz University of Technology (Graz, Austria)
Website jakobharden.at
Email jakob.harden@tugraz.at, jakob.harden@student.tugraz.at, office@jakobharden.at
ORCID 0000-0002-5752-1785
LinkedIn jakobharden



Copyright © 2024 Jakob Harden (Graz University of Technology (Graz, Austria))

This document is licensed under a Creative Commons Attribution 4.0 International license.

See also: CC BY 4.0, license deed

The above license applies to the entire content of this document. Deviations from this license are explicitly marked.