Synthetic Signal Shaping

Analytic-gradient Phase Optimization over Cross-Spectral Density Constraints and Arbitrary Memoryless Functionals

Luc Holtkamp

SenseMagic, Netherlands

luc@sensemagic.nl

Cover: x-y Trajectories of three partially coherent, two-channel blocks with cross-kurtosis profiles, generated by the method of this paper.

Abstract

We present a phase-domain method for generating multi-input synthetic vibration signals that match a prescribed cross-spectral density (CSD) together with a user-selected set of higher-order diagonal *and cross* moments. The CSD is enforced structurally by pre-multiplying random-phase, unit-modulus, frequency-domain signals with a Cholesky factor of the target CSD, following the frequency-domain synthesis of Shinozuka and Jan [1] and its extensions to partially coherent and non-Gaussian MIMO outputs [2–4]; this leaves the remaining phase degrees of freedom unconstrained. We show that this freedom is enough to shape the channelwise moments – extending the IFFT phase-manipulation idea of Steinwolf [5–7] – and, in addition, the joint cross-moments (co-skewness, co-kurtosis) that govern coincident-peak excursions and hence multi-axial fatigue damage [8]. The key result is a single Fourier-domain identity: the gradient of any differentiable memoryless functional of the synthesized signal, with respect to the free phases, reduces to one forward Fourier transform of its pointwise derivative, projected through the conjugated unit-modulus source spectrum u^* and the conjugated frequency response H^*. Every target used in this paper – diagonal and cross moments, endpoint values, a crest-factor surrogate – is an instance of this one identity, at a cost of at most m FFTs per gradient evaluation for a target of order m, independent of the number of channels; the MIMO case therefore requires no machinery beyond the single-channel one. The resulting problem is smooth and is solved with SciPy's L-BFGS-B [9, 10] using the analytic gradient directly, found in a head-to-head comparison to need an order of magnitude fewer loss evaluations than NLopt's CCSAQ [11, 12] on the warm-started continuation used for crest minimization, while matching or improving the achieved optimum. When the target statistics are estimated from a measured multi-axial record the targets are realizable by construction – the record itself is a witness – which is the primary operating regime for the method; we demonstrate it on a 12-channel wheel-hub force/moment record measured on a test track, reproducing all 30 moment targets alongside the full CSD, and show empirically that the per-evaluation cost scales tractably to 32 channels and 560 simultaneous targets. A complete, unit-tested reference implementation – including Slepian multitaper CSD estimation and analytic-gradient verification against central differences – is open-sourced at `https://github.com/luchp/mimoshape`; every figure and table in this paper is regenerated from it with fixed seeds.

Introduction

Frequency-domain synthesis fixes the second-order statistics of a multichannel signal by construction: pre-multiplying a random phase, unit-modulus spectrum with a Cholesky factor of the target cross-spectral density (CSD) reproduces any prescribed power spectrum and coherence structure exactly, for any number of channels. This leaves the phases themselves free – and, for a signal of any practical length, there are far more free phases than there are second-order constraints. We show that this remaining freedom is sufficient to simultaneously match an arbitrary set of higher-order diagonal and cross moments – skewness, kurtosis, co-skewness, co-kurtosis, or any other smooth memoryless statistic – while leaving the prescribed CSD untouched, and that the required gradients are available in closed form at a cost of a handful of FFTs per target, independent of the number of channels.

Contact: luc@sensemagic.nl • www.sensemagic.nl

This paper makes four contributions.

1. A unified phase-domain formulation for MIMO signal synthesis that enforces the target CSD structurally and reserves the remaining phase freedom for higher-order shaping.
2. A general Fourier-domain identity for the exact gradient of any differentiable memoryless functional evaluated on same-time signal samples (Proposition 1): one forward Fourier transform of its pointwise derivative per argument, at a cost independent of channel count. Every gradient used in this paper – diagonal moments, joint cross-moments, endpoint-continuity constraints, a crest-factor surrogate – is a single instance of this identity.
3. Extension of the target set from per-channel moments to *joint* cross-moments across channels – co-skewness, co-kurtosis – which govern coincident-peak behavior and hence multi-axial fatigue damage, together with an empirical account of how the per-evaluation cost scales with channel count (up to 32 channels, 560 simultaneous targets).
4. Validation on synthetic problems and on a measured 12-channel automotive road-load record, including a multimodel (section-wise) treatment of non-stationary records; a complete, unit-tested, open-source reference implementation regenerates every figure and table in this paper from fixed random seeds.

The optimization problem is generally non-convex; we do not provide a global convergence guarantee or a closed-form characterization of the feasible target set. Empirical evidence for both – an ensemble-based feasibility check, the residual loss of an infeasible run, and restart sensitivity across independent seeds – is given below. The related-work section contrasts the method with Gaussian CSD synthesis and Steinwolf's single-channel phase manipulation; the definitions section states the signal model and proves the central identity; the optimization section gives the algorithm and its cost; the examples section validates the method; the appendices collect material that supports but does not gate the main argument – the sampled-versus-physical-signal distinction near Nyquist, endpoint continuity for streamed blocks, and a benchmark against dedicated crest-factor optimizers.

Related work and positioning

Frequency-domain synthesis of stationary Gaussian random processes with a prescribed spectrum dates back to Rice [13] and was cast in its modern digital form by Shinozuka and Jan [1]. The MIMO extension – pre-multiplying random-phase spectra with a Cholesky factor of the target CSD – is due to Smallwood and Paez [2], with follow-on work on partially coherent and non-Gaussian outputs [3, 4, 14]. This structural approach is now the standard basis of MIMO random-vibration controllers.

Non-Gaussian shaping falls into two broad families. *Amplitude-domain warping* transforms a Gaussian signal through a memoryless nonlinearity chosen to hit a target skew/kurtosis; the Hermite polynomial transform of Winterstein [15] is the canonical example. It is simple and closed-form but distorts the spectrum. *Phase manipulation* keeps $|U(f)|$ fixed and manipulates $\arg U(f)$ so that a functional of the inverse-FFT time signal reaches a target, leaving the (auto-)spectrum invariant by construction. Three lines of work take this route for different functionals and have so far been treated as separate techniques: the IFFT phase-manipulation method of Steinwolf [5–7], single-channel and restricted to per-channel skewness and kurtosis; single-channel crest factor minimization [16–19]; and extensions of phase manipulation to multi-axial excitation with a synthetic road profile in Saathoff *et al.* [8], which motivates the joint-moment target set used below. The phase-only, magnitude-fixed nature of the problem also places it in the broad family of phase-retrieval algorithms [20, 21], although here the constraint set is on time-domain moments rather than on a second image plane.

The present paper treats these as instances of a single construction rather than as separate techniques. Once the CSD is enforced structurally on a multi-channel spectrum, the remaining phase freedom can be optimized against *any* differentiable memoryless functional of the same-time signal samples – evaluated per channel or jointly across channels – through the one closed-form gradient identity of Proposition 1 below. Steinwolf's per-channel skewness and kurtosis, Guillaume's ℓ_p crest objective [18], and the joint co-skewness/co-kurtosis targets that multi-axial fatigue damage requires [8] are then three particular choices of that functional, not three different algorithms; Appendix C evaluates the framework against dedicated crest-factor optimizers on exactly this basis, and the MIMO extension with prescribed CSD, joint cross-moments, and closed-form analytic gradients for gradient-based optimization is what the remaining sections develop.

Related nonlinear identification techniques (Volterra series, parallel cascades, block-oriented models) and their use inside iterative test-control loops [22–24] motivate the multi-axial setup but are not the subject of this paper; a companion note develops that connection separately. The framework presented here is not a purely academic construction: variants of it have been deployed by the author in several industrial projects for customers, in particular for model-estimation campaigns driven by minimum-kurtosis excitation, where concentrating the available actuator headroom into a dense, low-crest drive signal maximizes the signal-to-noise ratio of the estimated response models.

Definitions of signals and moments

Throughout, N_j denotes the number of channels and N_t the number of time samples per synthesis block; f and g index frequency bins $(0, \dots, N_t - 1)$; q indexes the channel whose phase a derivative is taken with respect to; and a boldface tuple $\mathbf{i} = (i_1, \dots, i_m)$ is an ordered list of (possibly repeated) channel indices identifying one joint-moment target of order m.

Let $\mathbf{u}$ be a complex-valued frequency-domain signal with N_j channels and unit modulus. The signal $\mathbf{v}$ is the result of filtering $\mathbf{u}$ with the complex frequency response function $\mathbf{H}$ with magnitude $\mathbf{h}$ and phase θ.

The result signal in the time domain is the channel wise inverse Fourier transform of $\mathbf{v}$.

$$H_{kjf} = h_{kjf}\, \mathrm{e}^{\mathrm{i}\theta_{kjf}} \tag{0.1}$$

$$u_{kf} = \mathrm{e}^{\mathrm{i}\psi_{kf}} \tag{0.2}$$

$$v_{kf} = \sum_{j=0}^{N_j-1} H_{kjf}\, \mathrm{e}^{\mathrm{i}\psi_{jf}} = \sum_{j=0}^{N_j-1} h_{kjf}\, \mathrm{e}^{\mathrm{i}\theta_{kjf}}\ \mathrm{e}^{\mathrm{i}\psi_{jf}} \tag{0.3}$$

$$\mathbf{x_k} = \mathcal{F}^{-1}(\mathbf{v_k}) \tag{0.4}$$

Assume further that $\mathbf{x}$ has N_t samples. We can then expand x_t as:

$$x_{kt} = \mathcal{F}_t^{-1}(\mathbf{v_k}) \tag{0.5}$$

$$= \frac{1}{N_t} \sum_{f=0}^{N_t-1} \sum_{j=0}^{N_j-1} h_{kjf}\, \mathrm{e}^{\mathrm{i}(\frac{2\pi t f}{N_t} + \theta_{kjf} + \psi_{jf})} \tag{0.6}$$

We will require that $\mathbf{x}$ is real and has zero mean, and zero energy at the Nyquist frequency (the latter is both a mathematical convenience and a physical necessity, discussed in Appendix A). Realness means that for $1 \le f < N_t/2$

$$\theta_{kjf} = -\theta_{kj(N_t-f)} \tag{0.7}$$

$$h_{kjf} = h_{kj(N_t-f)} \tag{0.8}$$

$$\psi_{jf} = -\psi_{j(N_t-f)} \tag{0.9}$$

The zero-mean and zero-Nyquist conditions are imposed through the magnitudes, $h_{kj0} = h_{kj(N_t/2)} = 0$, so the $f = 0$ and $f = N_t/2$ terms drop from every sum, and we can write for $\mathbf{x}$:

$$x_{kt} = \frac{1}{N_t} \sum_{f=0}^{N_t-1} \sum_{j=0}^{N_j-1} h_{kjf}\, \mathrm{e}^{\mathrm{i}(\frac{2\pi t f}{N_t} + \theta_{kjf} + \psi_{jf})} \tag{0.10}$$

$$= \frac{1}{N_t} \sum_{f=1}^{N_t/2-1} \sum_{j=0}^{N_j-1} (h_{kjf}\, \mathrm{e}^{\mathrm{i}(\frac{2\pi t f}{N_t} + \theta_{kjf} + \psi_{jf})} + h_{kj(N_t-f)}\, \mathrm{e}^{\mathrm{i}(\frac{2\pi t (N_t-f)}{N_t} + \theta_{kj(N_t-f)} + \psi_{j(N_t-f)})}) \tag{0.11}$$

$$= \frac{1}{N_t} \sum_{f=1}^{N_t/2-1} \sum_{j=0}^{N_j-1} (h_{kjf}\, \mathrm{e}^{\mathrm{i}(\frac{2\pi t f}{N_t} + \theta_{kjf} + \psi_{jf})} + h_{kjf}\, \mathrm{e}^{-\mathrm{i}(\frac{2\pi t f}{N_t} + \theta_{kjf} + \psi_{jf})}) \tag{0.12}$$

$$= \frac{2}{N_t} \sum_{f=1}^{N_t/2-1} \sum_{j=0}^{N_j-1} h_{kjf}\ \cos(\frac{2\pi t f}{N_t} + \theta_{kjf} + \psi_{jf}) \tag{0.13}$$

$$\tag{0.14}$$

Now define, for any ordered tuple of channel indices $\mathbf{i} = (i_1, \ldots, i_m)$ with entries in $[0, \ldots, N_j - 1]$ and length $|\mathbf{i}| = m \ge 1$, the raw joint moment

$$P_{\mathbf{i}} = \frac{1}{N_t} \sum_{t=0}^{N_t-1} \prod_{a=1}^{m} x_{i_a,t} \tag{0.15}$$

The tuple $\mathbf{i} = k^{(m)} \equiv \underbrace{(k, \ldots, k)}_{m}$ recovers the channelwise diagonal moment $P_{k^{(m)}} = \frac{1}{N_t} \sum_t x_{k,t}^m$; the tuple (k, k) gives the channel variance $P_{(k,k)}$; and (i, j) with $i \neq j$ gives the time-averaged cross-covariance. The second-order tuples are fixed structurally by the CSD constraint through $\mathbf{H}$ and are therefore excluded from the target set below.

The normalized joint moment is

$$M_{\mathbf{i}} \;=\; \frac{P_{\mathbf{i}}}{\prod_{a=1}^{m} \sqrt{P_{(i_a, i_a)}}} \qquad (m \geq 3) \tag{0.16}$$

with the convention $M_{(k)} = 0$ (zero-mean constraint).

Special cases: $M_{k^{(3)}}$ is the channelwise skewness, $M_{k^{(4)}}$ the channelwise kurtosis, $M_{(i,j,k)}$ with distinct indices a *co-skewness*, and $M_{(i,i,j,j)}$ a *co-kurtosis* – the latter governing coincident-peak behavior and hence multi-axial fatigue damage [8]. Steinwolf's original construction [5] shapes $M_{k^{(3)}}$ and $M_{k^{(4)}}$ only; the added value of the present formulation is the joint-index case.

Logical vs physical moments: a Nyquist-tail caveat

The moments $P_{\mathbf{i}}$ and $M_{\mathbf{i}}$ above are moments of the discrete signal $x_{k,t}$ on the sample grid $t = 0, \ldots, N_t - 1$ – a deliberate departure from Steinwolf's formulation [5, 7], which evaluates the kurtosis on the continuous periodic inverse-Fourier reconstruction. The two can diverge substantially near Nyquist: a single flat-spectrum tone shows a worst-case kurtosis mismatch of ≈33% at the Nyquist bin itself.

This is why we require $h_{kj,\,N_t/2} = 0$; no realizable reconstruction filter passes the Nyquist bin cleanly, so energy placed there is physically distorted before it reaches the plant.

In practice, we recommend a guard band above $f_s/4$ (half the Nyquist frequency) with a target CSD of zero. This keeps the mismatch below ∼1 % for spectra of practical interest. Appendix A develops the distinction in full, including the two operating regimes – measured versus analytically prescribed targets – in which it matters differently.

The loss function

The target set $\mathcal{S}$ is a user-selected collection of tuples with $|\mathbf{i}| \geq 3$, each carrying a target value $\mu_{\mathbf{i}}$ and optional weight $w_{\mathbf{i}}$. Typical choices include the diagonal tuples $\langle k^{(3)},\, k^{(4)} :\; k = 0, \ldots, N_j - 1\rangle$ for per-channel skew and kurtosis, and the cross tuples $\langle (i, i, j, j) :\; i < j\rangle$ for pair co-kurtosis. The loss is the weighted sum of squared deviations

$$\Xi \;=\; \frac{1}{2} \sum_{\mathbf{i} \in \mathcal{S}} w_{\mathbf{i}} \left(M_{\mathbf{i}} - \mu_{\mathbf{i}}\right)^2 \tag{0.17}$$

Contact: luc@sensemagic.nl • www.sensemagic.nl

Gradient of the loss

By the chain rule,

$$\frac{\partial \Xi}{\partial \psi_{qg}} \;=\; \sum_{\mathbf{i}\in\mathcal{S}} w_{\mathbf{i}}\,(M_{\mathbf{i}}-\mu_{\mathbf{i}})\,\frac{\partial M_{\mathbf{i}}}{\partial \psi_{qg}} \tag{0.18}$$

So we need $\partial M_{\mathbf{i}}/\partial\psi_{qg}$, which in turn depends on $\partial P_{\mathbf{i}}/\partial\psi_{qg}$ and $\partial P_{(i_a,i_a)}/\partial\psi_{qg}$. All of these, and every other target gradient used in this paper – endpoint values, the crest surrogate, the MIMO variance coupling below – are instances of a single identity. It is the central technical result on which the rest of the paper rests, so we state and prove it once, in full generality, before returning to the moment gradients above.

Proposition 1 (FFT form of the phase gradient). *Let $\ell:\mathbb{R}^m\to\mathbb{R}$ be a differentiable per-sample (memoryless) functional, let $\mathbf{i}=(i_1,\ldots,i_m)$ be a tuple of channel indices, and consider the time-averaged target*

$$Q(\mathbf{x}) \;=\; \frac{1}{N_t}\sum_{t=0}^{N_t-1}\ell(x_{i_1,t},\,x_{i_2,t},\,\ldots,\,x_{i_m,t}) \tag{0.19}$$

Then, with $\mathcal{F}_g[y]=\sum_t y_t\,\mathrm{e}^{-\mathrm{i}2\pi tg/N_t}$ the forward Fourier transform evaluated at bin g,

$$\frac{\partial Q}{\partial \psi_{qg}} \;=\; \frac{2}{N_t^2}\,\operatorname{Im}\left\{u_{qg}^*\sum_{a=1}^{m}H_{i_aqg}^*\,\mathcal{F}_g\left[\frac{\partial \ell}{\partial x_{i_a,t}}\right]\right\} \tag{0.20}$$

Proof. Using $\partial x_{k,t}/\partial\psi_{qg}=-\frac{2}{N_t}\,h_{kqg}\sin(\frac{2\pi tg}{N_t}+\theta_{kqg}+\psi_{qg})$ and the chain rule,

$$\frac{\partial Q}{\partial \psi_{qg}} \;=\; \frac{1}{N_t}\sum_{t=0}^{N_t-1}\sum_{a=1}^{m}\frac{\partial \ell}{\partial x_{i_a,t}}(x_{i_1,t},\ldots,x_{i_m,t})\,\frac{\partial x_{i_a,t}}{\partial \psi_{qg}} \tag{0.21}$$

$$\;=\; -\frac{2}{N_t^2}\sum_{a=1}^{m}\sum_{t=0}^{N_t-1}\frac{\partial \ell}{\partial x_{i_a,t}}(x_{i_1,t},\ldots,x_{i_m,t})\;h_{i_aqg}\;\sin(\frac{2\pi tg}{N_t}+\theta_{i_aqg}+\psi_{qg}) \tag{0.22}$$

Writing $\sin z=\operatorname{Im}\left\{\mathrm{e}^{\mathrm{i}z}\right\}$ and collecting the phase factors as $h_{i_aqg}\,\mathrm{e}^{\mathrm{i}\theta_{i_aqg}}=H_{i_aqg}$ and $\mathrm{e}^{\mathrm{i}\psi_{qg}}=u_{qg}$,

$$\frac{\partial Q}{\partial \psi_{qg}} \;=\; -\frac{2}{N_t^2}\,\operatorname{Im}\left\{u_{qg}\sum_{a=1}^{m}H_{i_aqg}\sum_{t=0}^{N_t-1}\frac{\partial \ell}{\partial x_{i_a,t}}(x_{i_1,t},\ldots,x_{i_m,t})\,\mathrm{e}^{\mathrm{i}\frac{2\pi tg}{N_t}}\right\}. \tag{0.23}$$

The values $\frac{\partial\ell}{\partial x_{i_a,t}}$ are real for every t, since ℓ is real-valued; conjugating a real sequence leaves it untouched, so conjugating $\mathcal{F}_g$ only flips the sign of its exponent – exactly the sign difference between the inner sum above and the defining sum of $\mathcal{F}_g$. The inner sum over t is therefore itself the complex conjugate of the forward transform of the pointwise derivative:

$$\sum_{t=0}^{N_t-1}\frac{\partial \ell}{\partial x_{i_a,t}}(x_{i_1,t},\ldots,x_{i_m,t})\,\mathrm{e}^{\mathrm{i}\frac{2\pi tg}{N_t}} \;=\; \Big(\sum_{t=0}^{N_t-1}\frac{\partial \ell}{\partial x_{i_a,t}}(x_{i_1,t},\ldots,x_{i_m,t})\,\mathrm{e}^{-\mathrm{i}\frac{2\pi tg}{N_t}}\Big)^* \tag{0.24}$$

$$\;=\; \mathcal{F}_g\left[\frac{\partial \ell}{\partial x_{i_a,t}}\right]^* \tag{0.25}$$

Substituting turns the whole time-domain sum into one already-familiar spectral object, conjugated:

Contact: luc@sensemagic.nl • www.sensemagic.nl

$$\frac{\partial Q}{\partial \psi_{qg}} = -\frac{2}{N_t^2}\ \mathrm{Im}\left\{u_{qg}\sum_{a=1}^{m} H_{i_a qg}\,\mathcal{F}_g\left[\frac{\partial \ell}{\partial x_{i_a,t}}\right]^*\right\} \tag{0.26}$$

It remains to move that conjugate off the transform and onto the phasors $u_{qg}, H_{i_a qg}$, where it belongs physically: the transform is a spectral quantity computed once per bin g, while the phasors are the per-phase quantities the optimizer differentiates against. For any $z \in \mathbb{C}$, $\mathrm{Im}\{z\} = -\mathrm{Im}\{z^*\}$; and therefore:

$$\mathrm{Im}\left\{u_{qg}\sum_{a=1}^{m} H_{i_a qg}\,\mathcal{F}_g\left[\frac{\partial \ell}{\partial x_{i_a,t}}\right]^*\right\} = -\,\mathrm{Im}\left\{u^*_{qg}\sum_{a=1}^{m} H^*_{i_a qg}\,\mathcal{F}_g\left[\frac{\partial \ell}{\partial x_{i_a,t}}\right]\right\} \tag{0.27}$$

The conjugate has moved from the transform onto u_{qg} and $H_{i_a qg}$, and the extra minus sign this introduces cancels the prefactor's minus sign, turning $-2/N_t^2$ into $+2/N_t^2$ and giving the stated identity. □

The phase-gradient of any memoryless functional of the inverse-Fourier signal is itself a Fourier transform – specifically, of the pointwise partial derivative $\frac{\partial \ell}{\partial x_{i_a,t}}$ with respect to the a-th argument, projected through u^*_{qg} and $H^*_{i_a qg}$ and averaged over the arguments $a = 1, \ldots, m$. One FFT per distinct argument of ℓ suffices; repeated arguments share their FFT. Every target-gradient in the paper is an instance of this identity, so the optimizer needs only two building blocks: the pointwise derivatives $\frac{\partial \ell}{\partial x_{i_a,t}}$, and one length-$N_t$ FFT of each.

Instance: joint moments. With $\ell(y_1, \ldots, y_m) = \prod_{a=1}^{m} y_a$, so $\frac{\partial \ell}{\partial x_{i_a,t}} = \prod_{b\neq a} y_b$, Proposition 1 gives

$$\frac{\partial P_{\mathbf{i}}}{\partial \psi_{qg}} = \frac{2}{N_t^2}\ \mathrm{Im}\left\{u^*_{qg}\sum_{a=1}^{m} H^*_{i_a qg}\,\mathcal{F}_g\left[\prod_{b\neq a} x_{i_b,\cdot}\right]\right\} \tag{0.28}$$

Instance: diagonal moment. For $\mathbf{i} = k^{(m)}$ all m partial products equal $x^{m-1}_{k,\cdot}$ and all m frequency factors equal H^*_{kqg}, so

$$\frac{\partial P_{k^{(m)}}}{\partial \psi_{qg}} = \frac{2m}{N_t^2}\ \mathrm{Im}\left\{u^*_{qg}\,H^*_{kqg}\,\mathcal{F}_g\left[x_k^{m-1}\right]\right\} \tag{0.29}$$

recovering the single-channel Steinwolf-style gradient [5] as a special case.

Instance: co-skewness. For $\mathbf{i} = (i, j, k)$ with distinct entries,

$$\frac{\partial P_{(i,j,k)}}{\partial \psi_{qg}} = \frac{2}{N_t^2}\ \mathrm{Im}\left\{u^*_{qg}\Big(H^*_{iqg}\,\mathcal{F}_g\left[x_j x_k\right] + H^*_{jqg}\,\mathcal{F}_g[x_i x_k] + H^*_{kqg}\,\mathcal{F}_g\left[x_i x_j\right]\Big)\right\} \tag{0.30}$$

Instance: co-kurtosis. For $\mathbf{i} = (i, i, j, j)$ the partial products collapse pairwise so only two FFTs are needed:

$$\frac{\partial P_{(i,i,j,j)}}{\partial \psi_{qg}} = \frac{4}{N_t^2}\ \mathrm{Im}\left\{u^*_{qg}\Big(H^*_{iqg}\,\mathcal{F}_g\left[x_i x_j^2\right] + H^*_{jqg}\,\mathcal{F}_g\left[x_j x_i^2\right]\Big)\right\} \tag{0.31}$$

Instance: variance. Taking $\ell(y) = y^2$ so $\frac{\partial \ell}{\partial_y} = 2y$, and using $\mathcal{F}_g[x_k] = v_{kg}$,

$$\frac{\partial P_{(k,k)}}{\partial \psi_{qg}} = \frac{4}{N_t^2}\ \mathrm{Im}\left\{u^*_{qg}\,H^*_{kqg}\,\mathcal{F}_g[x_k]\right\} = \frac{4}{N_t^2}\ \mathrm{Im}\left\{u^*_{qg}\,H^*_{kqg}\,v_{kg}\right\}. \tag{0.32}$$

Contact: luc@sensemagic.nl • www.sensemagic.nl

The v_{kg} are computed once at the start of every iteration (they are needed to obtain $x_{k,t}$ in the first place), so the variance gradient is free.

Chain rule on the normalization.

$$\frac{\partial M_{\mathbf{i}}}{\partial \psi_{qg}} = \frac{1}{\prod_{a=1}^{m} \sqrt{P_{(i_a,i_a)}}} \left[\frac{\partial P_{\mathbf{i}}}{\partial \psi_{qg}} - \frac{P_{\mathbf{i}}}{2} \sum_{a=1}^{m} \frac{1}{P_{(i_a,i_a)}} \frac{\partial P_{(i_a,i_a)}}{\partial \psi_{qg}} \right]. \tag{0.33}$$

MIMO variance coupling. In the single-channel case the per-channel variance $P_{(k,k)}$ is invariant under changes of its own phases (a direct consequence of the FFT identity, or equivalently of $\mathrm{Im}\left\{|H_{qqg} u_{qg}|^2\right\} = 0$ when H is diagonal), so it factors out of the loss gradient cleanly. In the MIMO case with off-diagonal H_{pqg} the variance of channel k depends on the phases of other channels through the cross-coupling and no longer factors out – raising the concern that MIMO might require fundamentally new machinery beyond the single-channel case. Proposition 1 removes that concern: the variance derivative is itself an instance of the same FFT identity, contributing one already-computed pass (v_{kg}) per channel to the total gradient. Every additional target – diagonal moment, cross-moment, variance, endpoint value, or any user-supplied smooth memoryless functional – adds at most $|\mathbf{i}|$ FFTs to the iteration and nothing conceptually new.

Cost. Each target tuple contributes at most $|\mathbf{i}|$ length-N_t FFTs to the gradient (fewer when partial products repeat). The variance needed for normalization are shared across all targets referencing channel k; one pass per channel suffices per iteration. For N_j output channels with all per-channel skew/kurtosis plus all pair co-kurtosis in $\mathcal{S}$, the count is one FFT per diagonal skew and kurtosis target ($2N_j$) plus two per co-kurtosis pair ($N_j(N_j-1)$): $N_j(N_j+1)$ FFTs per gradient evaluation in total, quadratic in the channel count. The scaling rows of Table 3 measure this growth empirically up to $N_j = 32$.

Feasibility. When the target statistics $\mu_{\mathbf{i}}$ are estimated from a measured multi-channel record (see the multitaper subsection below), a signal with those statistics exists – the record itself – and in our experience the optimizer reaches them within the tolerance set by the finite block length; this is the primary operating regime of the method. For analytically prescribed targets no closed-form realizability bounds are known in the general case, and we rely on two cheap numerical diagnostics.

First, a *random-phase ensemble check* (provided by the reference implementation): evaluate each functional $M_{\mathbf{i}}$ over an ensemble of random-phase draws (64 draws suffice) to estimate its expected value $\mathbb{E}[M_{\mathbf{i}}]$ and standard deviation $\sigma(M_{\mathbf{i}})$ under an unshaped, uniform phase distribution. A target within a few ensemble standard deviations of the mean is already close to what unshaped random phases produce, so the optimizer reaches it in few iterations.

The converse does not hold – the ensemble concentrates near the Gaussian point by the central limit theorem, so a strongly non-Gaussian target such as kurtosis 5 sits tens of ensemble standard deviations out, yet is realizable in practice (it is the feasible target used throughout the Examples section, converging in roughly 100 evaluations) – so the check locates the easy region, not the feasibility boundary.

Second, the *residual loss* of a converged run measures the infeasibility gap directly: as the convergence example below shows, an infeasible target produces no erratic behavior, only a smooth descent onto a plateau whose height quantifies how far the target lies outside the achievable set.

Beyond polynomial moments. Proposition 1 places no restriction on ℓ beyond differentiability, but it does

Contact: luc@sensemagic.nl • www.sensemagic.nl

require ℓ to act on *same-time* samples: any smooth functional of the form $\frac{1}{N_t}\sum_t \ell(x_{i_1,t},\ldots,x_{i_m,t})$ is a legitimate target for the same optimizer without new machinery.

This covers absolute-value moments $\ell(y) = |y|^m$ (with $m|y|^{m-1}\operatorname{sgn} y$ as the pointwise derivative) and coincident-excursion probabilities via smoothed indicator products; the user provides ℓ and $\frac{\partial\ell}{\partial x_{i_a,t}}$ as pointwise functions and the optimizer handles the rest. The reference implementation exposes single-channel instances as `FunctionTarget` (on the raw signal) and `ScaledFunctionTarget` (on the normalized signal $z_k = x_k/s_k$, with the variance chain rule handled analytically); the ℓ_p crest objective of Guillaume *et al.* [18] is then the one-liner $\ell(z) = |z|^p$, put to work in the benchmark of Appendix C.

Functionals that couple *different* time indices fall outside the stated form: peak and sign-crossing counts involve neighboring samples $x_{t\pm1}$, and rainflow-type fatigue dosage functionals are history-dependent, not memoryless. The extension to lagged products $g(x_{i_1,t},\, x_{i_2,t+\ell})$ appears mechanical – each argument acquires its own frequency-domain shift factor – but is not derived here.

Endpoint continuity for streamed blocks

Live streaming concatenates independent blocks – each with fresh random phases – on the fly; unless consecutive blocks share their head value and slope, every block boundary produces a jump or slope kink that shows up as broadband spectral leakage.

The synthesizer fixes the block's head value and slope to prescribed constants (zero by default) as two extra terms in the loss. Because this is point evaluation at $t = 0$ rather than a sample average, it is the degenerate case of Proposition 1 in which the "FFT of the pointwise gradient" collapses to a Kronecker delta: no additional FFT is needed, only a single complex multiply–add per phase per constrained channel.

Appendix B gives the constraint and its gradient in full, together with the periodic FRF estimation use case, where the same head-value constraint removes the start-up transient without any seam constraint being needed between repetitions.

Direct minimization: a smooth crest-factor surrogate

For model estimation and endurance testing one often wants the *opposite* of heavy tails: minimum kurtosis and minimum crest factor $\max_t|x_{kt}|/s_k$ (with $s_k = \sqrt{P_{(k,k)}}$), concentrating the signal energy and maximizing the drive level attainable within actuator limits. Crest-factor minimization for single-channel multisines is a classical subject [16–19]; the treatment below recovers this special case within the present MIMO framework.

The crest factor itself is a poor optimization objective: its gradient is supported only at the extremal sample, so a gradient step relocates the peak elsewhere in the block and the optimizer hops between candidate peaks without converging.

Minimizing the kurtosis $M_{k^{(4)}}$ instead is smooth but weights peaks only polynomially and stalls at a crest well above the achievable floor; higher even moments ($M_{k^{(8)}}$, $M_{k^{(16)}}$) push further but exhibit the same plateau behavior.

A smooth surrogate with *exponential* peak weighting resolves this. For channel k define

$$\Xi_{\text{crest}} = \frac{w}{\beta} \ln\left(\frac{1}{N_t} \sum_{t=0}^{N_t-1} \cosh z_{kt} \right), \qquad z_{kt} = \frac{\beta\, x_{kt}}{s_k} \tag{0.34}$$

which approaches $w \max_t |x_{kt}|/s_k$ from below as $\beta \to \infty$, with bias of order $\ln(2N_t)/\beta$. This term is added directly to the loss (it has no target value) and its gradient is one more instance of Proposition 1 :

$$C = \frac{1}{N_t} \sum_{t=0}^{N_t-1} \cosh z_{kt} \tag{0.35}$$

$$\frac{\partial \Xi_{\text{crest}}}{\partial \psi_{qg}} = \frac{w}{C} \left[\frac{1}{s_k} \frac{2}{N_t^2} \operatorname{Im}\left\{ u_{qg}^* \, H_{kqg}^* \, \mathcal{F}_g[\sinh z_k] \right\} - \frac{1}{2 s_k^3} \frac{\partial P_{(k,k)}}{\partial \psi_{qg}} \frac{1}{N_t} \sum_{t=0}^{N_t-1} x_{kt} \sinh z_{kt} \right] \tag{0.36}$$

The cost is one FFT for the memoryless factor $\sinh z$ plus the already-computed variance gradient; the second term is required because in the MIMO case the normalization s_k depends on the phases of all channels through the off-diagonal H. Numerically the surrogate is evaluated in shifted-exponential form (cosh and sinh scaled by $e^{-\max_t |z_{kt}|}$) so that large β does not overflow.

Large β makes the landscape stiff, and starting a high-β optimization from random phases stalls at a mediocre crest. A *β-continuation* resolves this: optimize at a small β, then double β and warm-start from the previous optimum. Each stage after the first converges in a small fraction of the initial stage's iterations, and the continuation reached lower crest factors than single-stage runs at every budget we tested (see the example below).

The staging mirrors the classical ℓ_p-norm (Polya) continuation of Guillaume *et al.* [18], with two differences: the logcosh gradient factor $\tanh z$ is bounded, so no numerical care is needed at high stiffness where the ℓ_p factor $p\,|x|^{p-1}$ overflows, and the surrogate slots unchanged into the MIMO framework alongside CSD structure, cross-moment and endpoint targets.

Estimating targets from measured data

The synthesizer requires a target CSD $\mathbf{G}(f)$ (which determines $\mathbf{H}$ via Cholesky) and a set of higher-order moment targets $\langle \mu_{\mathbf{i}} \rangle_{\mathbf{i} \in \mathcal{S}}$. When these are estimated from a measured multi-axial record – the primary operating regime for this method – the choice of estimator matters, because a biased or non-PSD CSD estimate contaminates the Cholesky factor and propagates structurally into every synthesized block.

Why not the periodogram or Welch. The single-taper periodogram has bias controlled by the window's spectral leakage: rectangular windows leak wildly, Hann or Hamming tapers reduce leakage at the cost of resolution but still concentrate all estimator energy on a single window. Welch averaging reduces variance at the cost of frequency resolution and, on records with large dynamic range across frequencies, introduces persistent leakage bias that cannot be averaged out. For cross-spectra the situation is worse: the coherence estimate acquires a positive bias that vanishes only asymptotically.

Multitaper with Slepian sequences. The Thomson multitaper estimator [25] replaces the single window with a family of K orthonormal tapers whose energy is maximally concentrated in a chosen bandwidth $[-W, W]$.

These tapers are the discrete prolate spheroidal sequences (DPSS) of Slepian [26]; the first $K \approx 2N_t W - 1$ are effectively leakage-free. The multitaper CSD estimate for channels p, q is

$$\hat{G}_{pq}(f) \quad = \quad \frac{1}{K}\sum_{k=0}^{K-1} Y_p^{(k)}(f)\, Y_q^{(k)}(f)^*, \qquad Y_p^{(k)}(f) = \mathcal{F}\left[w_t^{(k)}\, y_{p,t}\right], \tag{0.37}$$

where $w^{(k)}$ is the k-th DPSS taper and $y_{p,t}$ is the measured record on channel p. The estimator has approximately $2K$ degrees of freedom, low bias by design, and is manifestly Hermitian positive semi-definite because it is an average of rank-one outer products. Percival and Walden [27] give the standard univariate treatment; Walden [28] extends it to the MIMO case with jackknife error bars on cross-spectral phase and coherence.

The PSD property is important here: Cholesky factorization $\mathbf{G}(f) = \mathbf{H}^*\mathbf{H}$ requires $\hat{\mathbf{G}}(f)$ to be PSD at every frequency bin. Estimators that are not manifestly PSD – windowed cross-periodograms, some parametric schemes, ad hoc coherence-weighted spectra – require a projection onto the PSD cone before Cholesky [29], which introduces its own bias. The multitaper estimator sidesteps this entirely. Positive *semi*-definite is not positive definite, however: $\hat{\mathbf{G}}(f)$ is rank-deficient whenever fewer than N_j independent taper–segment products enter the average at a bin, and near-singular when two channels are almost coherent there. This is the rule rather than the exception in measured multi-axial records: left/right wheel-hub forces at low frequency, or colocated force-moment pairs, routinely drive pairwise coherences above 0.99 and bin condition numbers to 10^8. The Cholesky factorization then fails or amplifies estimation noise; the reference implementation raises an explicit error rather than regularizing silently. A robust estimation package must therefore ship an explicit regularization strategy – more tapers or segments, a small diagonal load $\hat{\mathbf{G}}+\epsilon\mathbf{I}$ with documented ϵ, or restricting the synthesis band to bins where the estimate is well-conditioned – as a first-class, reported step rather than a silent fallback.

Higher-order targets. The moment targets $\mu_{\mathbf{i}}$ are estimated directly from the record as sample statistics $\hat{P}_{\mathbf{i}} = \frac{1}{N_t}\sum_t \prod_a y_{i_a,t}$, with the corresponding normalization. Their variance decreases as $1/N_t$ for stationary records; in practice we recommend segmenting a long record and averaging the per-segment $\hat{P}_{\mathbf{i}}$, or bootstrapping from the multitaper eigencoefficients as in Percival and Walden [27].

Non-stationary records and the multimodel approach. Measured field records are rarely stationary over their full length. The practical remedy is to partition the record into segments with similar statistics, group those segments, and estimate one CSD-plus-moment target set per group – yielding a small library of models rather than a single compromise model. The multitaper estimator is particularly well suited to this regime: its low variance per short segment ($\approx 2K$ degrees of freedom from a single block) makes reliable per-group estimates possible where Welch averaging would demand data lengths the group does not have. Synthesis then proceeds per model, and the models can be sequenced or blended to reproduce the non-stationary character of the original environment. We have applied this *multimodel* approach in industrial practice; a minimal worked validation closes the examples section, and an extended treatment (grouping strategies, model libraries) is deferred to a companion note.

Contact: luc@sensemagic.nl • www.sensemagic.nl

Optimization

The loss $\Xi(\psi)$ is smooth in the free phases ψ_{qg} and its analytic gradient is available at the cost of a handful of FFTs per iteration (see the cross-moment cost analysis above). This makes the problem well-suited to gradient-based optimization. We use SciPy's L-BFGS-B [9, 10], the limited-memory quasi-Newton method of Byrd, Lu, Nocedal and Zhu restricted to box constraints; it is compared below against NLopt's CCSAQ [11, 12] and SciPy's SLSQP and found both simpler to depend on – SciPy is already a required dependency – and consistently more efficient. The phase parameters are unbounded periodic, and no explicit constraints are required beyond the CSD-derived spectral structure that is already enforced through $\mathbf{H}$; the box bounds $\psi \in [-\pi, \pi]$ are imposed purely for numerical convenience.

Algorithm (synthesis of one block).

1. Estimate or prescribe the target CSD $\mathbf{G}(f)$; factor it per bin as $\mathbf{H}^*(f)\,\mathbf{H}(f) = \mathbf{G}(f)$ (Cholesky), with $\mathbf{H}(0) = \mathbf{H}(N_t/2) = \mathbf{0}$.
2. Select the target set $\mathcal{S}$: moment tuples with values $\mu_{\mathbf{i}}$ and weights $w_{\mathbf{i}}$, optional endpoint values, optional crest surrogate.
3. Draw the free phases $\psi_{qg} \sim \mathcal{U}(-\pi, \pi)$ i.i.d. (or warm-start from a previous stage).
4. Iterate L-BFGS-B: form $u = \mathrm{e}^{\mathrm{i}\psi}$, $v = Hu$, $x = \mathcal{F}^{-1}[v]$; evaluate Ξ and its gradient via Proposition 1; update ψ.
5. Stop at the loss threshold or budget; x is the synthesized block. For crest minimization, double β and warm-start from the current ψ until the final stiffness is reached.

Comparison with generic gradient-based optimizers

The loss and its analytic gradient make no solver-specific assumptions, allowing the same objective to be handed directly to any bound-constrained gradient-based optimizer—bypassing costly finite-difference approximations while preserving numerical precision.

Figure 1 compares SciPy's L-BFGS-B – the framework's default – against NLopt's CCSAQ and SciPy's SLSQP on the reference single-channel kurtosis-5 problem, using the identical gradient, the same box bounds $\psi \in [-\pi, \pi]$, and a matched early-stop threshold ($\Xi < 10^{-10}$). All three reach the threshold. L-BFGS-B needs about a third of the evaluations CCSAQ does at $N_{\text{free}} = 2047$ free phases (Table 2) and is faster than CCSAQ at every size tested. SLSQP reaches the threshold in a comparable number of evaluations but solves a dense quadratic subproblem every iteration, so its per-block cost grows far faster with problem size: at $N_{\text{free}} = 2047$ it is ≈75× slower than CCSAQ, which rules it out at the phase counts used throughout this paper (up to ≈16,000 in the channel-scaling study below).

The advantage compounds under the warm-started β-continuation used for crest minimization. Table 1 runs the same continuation as Table 6 below on all five Retzler setups with both optimizers: L-BFGS-B matches or slightly improves on CCSAQ's achieved crest in every setup, at ≈34× fewer loss evaluations on average. We

Contact: luc@sensemagic.nl • www.sensemagic.nl

therefore use SciPy's L-BFGS-B as the default optimizer throughout this paper; NLopt's CCSAQ remains a drop-in alternative given the identical gradient and is retained here as the comparison baseline.

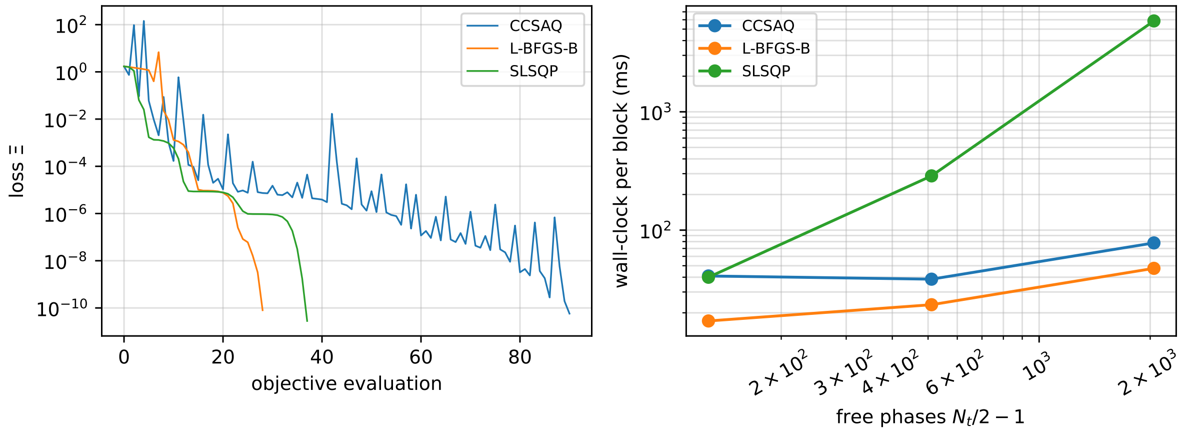

Figure 1: SciPy's L-BFGS-B versus NLopt's CCSAQ and SciPy's SLSQP, identical analytic gradient throughout. Left: loss versus evaluation for the reference kurtosis-5 problem, one seed, matched early-stop threshold. Right: wall-clock time per block versus free-phase count, log-log, averaged over 8 seeds.

	CCSAQ		L-BFGS-B	
setup	crest	evals	crest	evals
A	1.4228	31366	1.4142	826
B	1.4728	34765	1.4591	942
C	2.0172	7253	2.0138	217
D	2.0265	21883	2.0077	612
E	1.3887	34802	1.3864	1444

Table 1: CCSAQ versus L-BFGS-B on the actual β-continued crest surrogate: same five Retzler setups and continuation ladder as Table 6, averaged over 10 independent random starts per setup.

Software architecture and verification

The reference implementation[1] is a small NumPy/SciPy package: one module of stateless numerics (moments, cross-moments, crest surrogate, and their analytic gradients) and one module that assembles them into a synthesis problem and drives SciPy's L-BFGS-B optimizer.

Every analytic gradient in this paper – including the cross-moment and crest-surrogate derivatives – is verified by the unit-test suite against central-difference numerical gradients on randomized problems (random spectra, channel counts, moment tuples and surrogate stiffnesses), and the suite runs on every change. Every figure and table in this paper is regenerated by a single script with fixed random seeds.

[1] `https://github.com/luchp/mimoshape`

	$N_{free} = 127$		$N_{free} = 511$		$N_{free} = 2047$	
method	ms/block	evals	ms/block	evals	ms/block	evals
CCSAQ	41.0	155	38.5	101	78.0	108
L-BFGS-B	17.1	27	23.4	30	47.5	36
SLSQP	40.0	29	287.4	31	5862.9	39

Table 2: Wall-clock time and evaluation count to reach the matched threshold $\Xi < 10^{-10}$, averaged over 8 independent random starts, for the reference kurtosis-5 problem at growing free-phase count.

Examples

All examples in this section are produced by the reference implementation.[2]

Single channel: kurtosis shaping. Figure 2 shows two blocks with the identical flat magnitude spectrum ($N_t = 4096$): one with random phases (kurtosis ≈3, Gaussian by the central limit theorem) and one with phases shaped to the target kurtosis 5, together with zero head value and slope. The spectrum is untouched by construction; only the phase distribution differs.

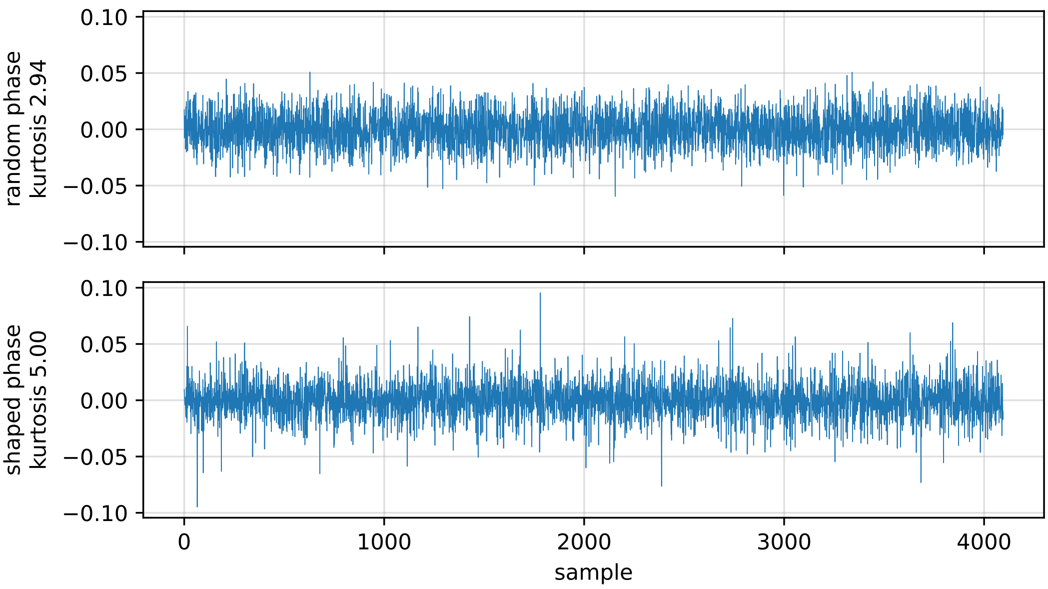

Figure 2: Identical flat magnitude spectrum, different phases. Top: random phases (Gaussian, kurtosis 2.94). Bottom: shaped phases (kurtosis 5.00).

Convergence and infeasible targets. Figure 3 shows loss trajectories for the single-channel problem above. With the feasible target (kurtosis 5) the optimizer reaches the stop value of 10^{-10} in 35 objective evaluations, each costing a handful of FFTs; the brief plateaus are step-size trial evaluations inside L-BFGS-B's line search. The second trajectory requests an *infeasible* target – kurtosis 1.0, below the ≈1.13 floor achievable for this spectrum – and shows no gradient stagnation or erratic behavior: the loss descends smoothly onto a plateau at

[2] Every figure and table is regenerated, with fixed random seeds, by a single script (`scripts/make_figures.py`) in the accompanying repository; the analytic gradients are verified against central-difference numerical gradients by the unit-test suite.

the feasibility boundary, and the residual loss directly measures how far the requested target lies outside the achievable set. This makes an infeasible request cheap to detect in practice, complementing the random-phase ensemble check of the feasibility paragraph.

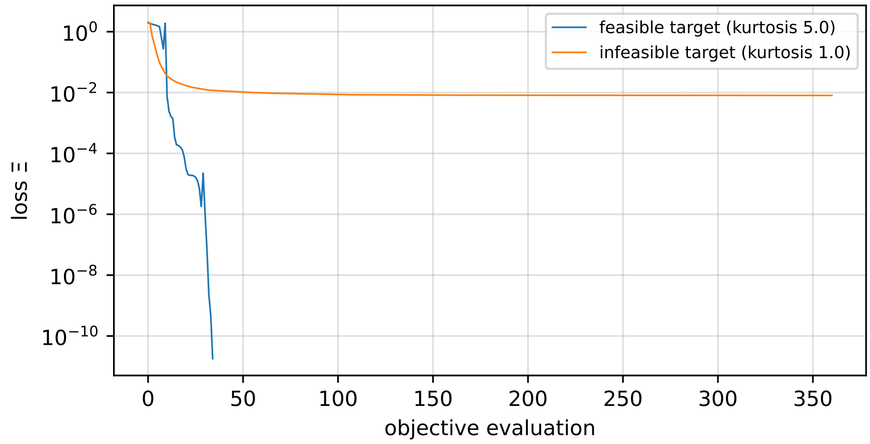

Figure 3: Loss Ξ versus objective evaluation for the single-channel kurtosis problem (L-BFGS-B, analytic gradients): a feasible target converges to the stop value; an infeasible target settles smoothly onto the feasibility gap.

Seed sensitivity. Phase-domain optimizers are sometimes suspected of trapping in poor local minima seeded by the initial random phases. Figure 4 shows the distribution of the final loss for the same feasible problem over 64 independent random restarts: median $10^{-6.0}$, worst $10^{-4.0}$. All restarts land many orders of magnitude below any practically relevant tolerance: with thousands of free phases and a handful of scalar targets, this instance is massively underdetermined. We observe the same benign behavior on every problem in this paper (including the MIMO example below, where all 32 blocks converge from independent seeds), but a systematic study of the loss landscape at scale – many channels, many coupled cross-moment targets, or stiff crest surrogates, where multimodality is more plausible – is outside the scope of this paper.

Wall-clock cost. Table 3 reports single-core wall-clock times per optimized block on a single core of an AMD Ryzen AI 9 HX 370 laptop (32 GB RAM). Moment-and-endpoint problems cost well under 0.1 s per block even at $N_t = 16384$; the evaluation cost scales with the FFT length as expected, while the evaluation *count* is set by the objective mix, not the block size. The MIMO scaling rows grow the channel count at fixed $N_t = 1024$ with the full per-channel skew/kurtosis set plus all pair co-kurtoses (targets set to their jointly Gaussian values, so the set is feasible by the central limit theorem), up to $N_j = 32$ channels with 560 simultaneous moment targets. A log–log fit of the per-evaluation cost over $N_j \in \langle 4, 8, 16, 32 \rangle$ gives ms/eval $\propto N_j^{2.16}$: the quadratic $N_j(N_j+1)$ FFT count of the cost analysis (fitted slope ≈1.9 over this range) plus a measurable contribution from the N_j^2-per-bin mixing product and memory traffic. The evaluation *count*, meanwhile, stays essentially flat in the channel count, so the block cost tracks the per-evaluation cost – convergence effort does not deteriorate as channels and targets are added. The crest surrogate is the expensive case – not per evaluation, but because a stiff surrogate needs thousands of evaluations per continuation stage. Block synthesis on desktop-class hardware is faster than real-time playback – roughly a factor of three for the 12-channel road-record block below – which

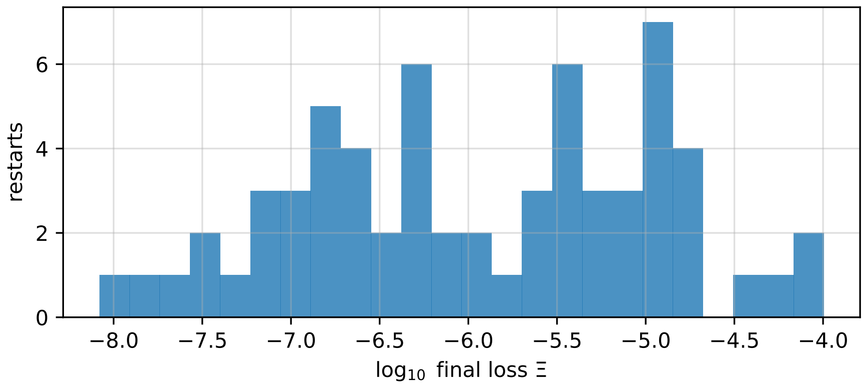

Figure 4: Final loss over 64 independent random restarts of the single-channel kurtosis problem ($N_t = 4096$, stop value 10^{-10}, budget 5 s per restart).

matches the intended deployment: the optimizer runs on the host PC and streams blocks to the real-time target; feasibility on embedded DSP-class hardware has not been assessed.

problem	N_j	N_t	phases	evals	s/block	ms/eval
SISO skew+kurt+endpoint	1	1024	511	22	0.02	0.70
SISO skew+kurt+endpoint	1	4096	2047	26	0.03	1.05
SISO skew+kurt+endpoint	1	16384	8191	53	0.12	2.26
SISO crest ($\beta = 80$)	1	4096	2047	1640	1.24	0.75
MIMO 2ch, CSD + 5 (cross-)moments	2	1024	1022	85	0.08	0.88
MIMO scaling, 14 moment targets	4	1024	2044	10	0.02	2.49
MIMO scaling, 44 moment targets	8	1024	4088	16	0.13	7.88
MIMO scaling, 152 moment targets	16	1024	8176	26	0.92	35.32
MIMO scaling, 560 moment targets	32	1024	16352	27	6.04	223.86

Table 3: Wall-clock cost per optimized block (single core, AMD Ryzen AI 9 HX 370). "Evals" counts objective/gradient evaluations to the stop criterion of the corresponding example; the MIMO scaling rows use per-channel skew/kurtosis plus all pair co-kurtoses at their jointly Gaussian values.

Minimum crest factor. Figure 5 shows the crest surrogate of the previous section at work on the flat-spectrum single-channel problem ($N_t = 4096$), for the full band and for a half-band spectrum with a zero upper tail. The β-continuation (doubling from 5 to 320, warm-started) reaches a sampled crest factor of ≈1.14 for the full band and ≈1.33 for the half band, consistently across random seeds; direct single-stage runs at high β plateau near 1.16 despite several times the iteration budget. Minimum crest and minimum kurtosis are *different* signals: the crest-optimal blocks settle at kurtosis ≈1.17, visibly above the minimum achievable kurtosis of ≈1.13 for the same spectrum.

The sampled numbers must, however, be read against the logical-versus-physical caveat, and here the effect is dramatic rather than cosmetic. Re-evaluating the optimized blocks on an 8× zero-padded reconstruction – the ideal band-limited interpolation, a proxy for the continuous DAC output; the ZOH droop and reconstruction-filter response of a real playback chain are separate linear effects, see Appendix A – gives a physical crest of ≈2.65 for the full-band signal versus ≈1.67 for the half-band signal. With near-Nyquist content the optimizer exploits the sample grid: it parks the waveform's peaks *between* the samples, and the sampled crest of 1.14 – impossibly below the $\sqrt{2}$ of a constant-envelope signal – is entirely an artifact. The half-band signal, whose reconstruction cannot oscillate between samples, degrades only mildly and its physical crest of ≈1.67 is the number that reaches the test article.

The same reasoning extends *inside* the band: a brickwall band edge forces sinc-like ringing into every reconstruction, and smoothing the edge removes it – replacing the brickwall by a raised-cosine roll-off over the top 10% of the band lowers the physical crest further to ≈1.59, while the sampled crest is indifferent to the edge shape. For crest-critical drive signals the spectrum should therefore keep a zero Nyquist tail, exactly as Appendix A prescribes for kurtosis, and preferably a tapered band edge; the resulting waveform is a constant-envelope, noise-like block with no visible chirp structure, at a physical crest competitive with classical single-channel multisine results obtained at far lower tone counts – a claim Appendix C makes quantitative.

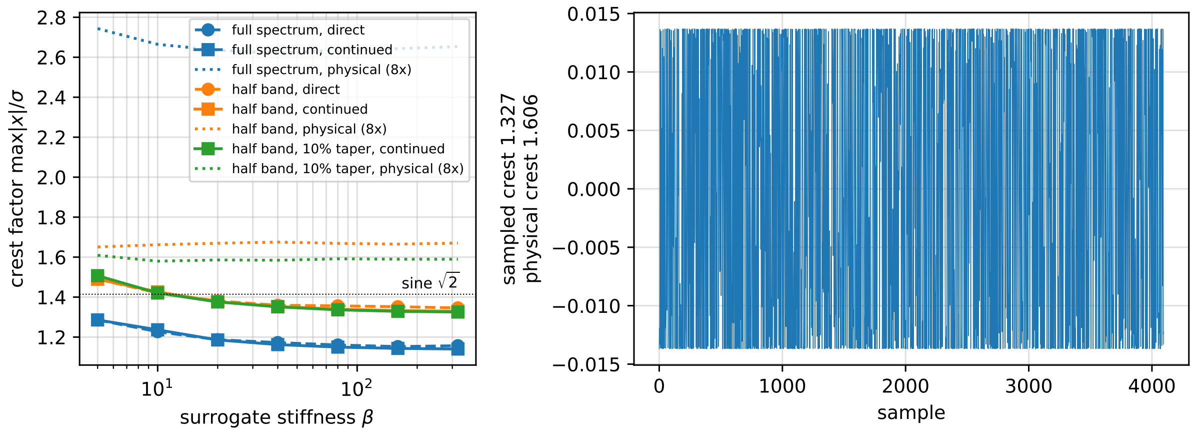

Figure 5: Crest-factor minimization with the logcosh surrogate. Left: achieved crest versus surrogate stiffness β, seed-averaged; warm-started β-continuation (solid) versus direct single-stage runs (dashed), with the physical crest of the continued optimum measured on an 8× oversampled reconstruction (dotted). A raised-cosine taper over the top 10% of the band removes the edge ringing and lowers the physical crest further. Right: the resulting tapered minimum-crest block.

Appendix C benchmarks this surrogate, and its ℓ_p-subsumption variant, against five dedicated published crest-factor optimizers and against Steinwolf's and Ojarand's reported numbers.

MIMO with a surrogate record. A two-channel surrogate record is synthesized as colored, correlated, non-Gaussian noise: channel 0 is lowpass weighted and memorylessly distorted (skewed, heavy-tailed); channel 1 shares a common source with channel 0 through a resonance at 0.2 f_s. Targets are estimated per the previous section: multitaper CSD ($NW = 4$, $K = 7$ tapers, 16 segments of 1024 samples) factored per bin by Cholesky, and sample moments for the tuple set $\langle(0,0,0), (1,1,1), (0,0,0,0), (1,1,1,1), (0,0,1,1)\rangle$. Table 4 lists targets and achieved values (mean ± standard deviation over 32 independently optimized blocks); Figure 6 shows one synthesized block, and Figure 7 compares the target CSD with a multitaper estimate over the ensemble of synthesized blocks. PSD, coherence and cross phase are all reproduced. Estimating both sides with the same estimator matters here: per block the off-diagonal cross-spectrum is a single rank-one sample that carries a random phase factor $e^{i(\psi_{0f}-\psi_{1f})}$, so only the ensemble converges – and at the low coherence of this record the phase scatter of a low-degrees-of-freedom estimate is large, of order $\sqrt{(1/\gamma^2-1)/2N}$.

target	tuple $\mathbf{i}$	$\mu_{\mathbf{i}}$	achieved (mean ± std)
skewness ch. 0	$(0,0,0)$	3.081	3.067 ± 0.012
skewness ch. 1	$(1,1,1)$	-0.004	-0.005 ± 0.005
kurtosis ch. 0	$(0,0,0,0)$	17.246	17.246 ± 0.006
kurtosis ch. 1	$(1,1,1,1)$	2.939	2.940 ± 0.005
co-kurtosis $(0,0,1,1)$	$(0,0,1,1)$	1.314	1.315 ± 0.003

Table 4: Surrogate-record MIMO example: moment targets estimated from the record versus achieved values, mean ± standard deviation over 32 blocks ($N_t = 1024$).

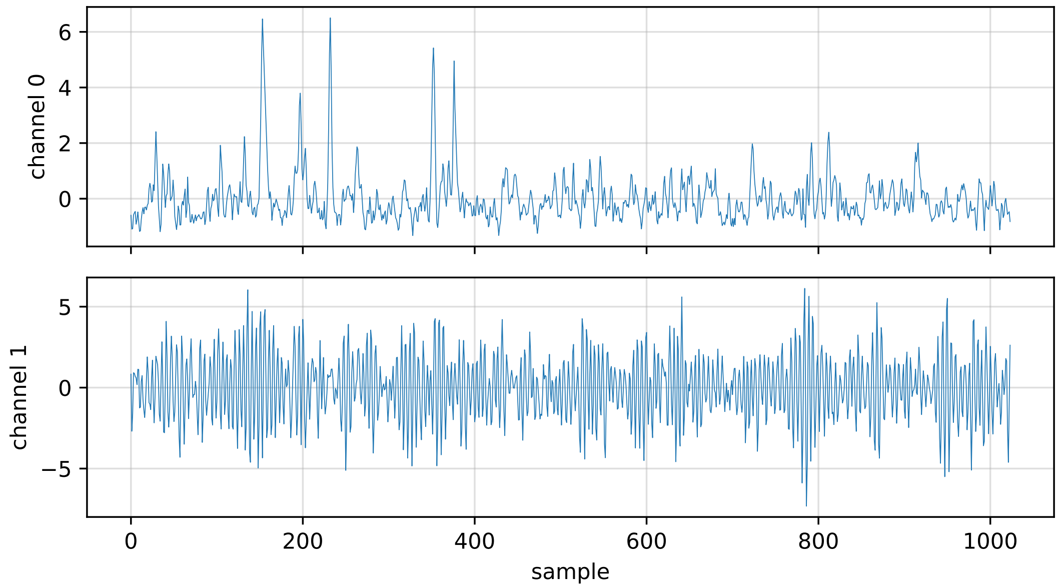

Figure 6: One synthesized MIMO block. Channel 0 reproduces the strong skew and kurtosis of the record; channel 1 stays near-Gaussian, as its targets dictate.

MIMO with a measured road record. The measured-target regime is demonstrated on a 12-channel road-load record: forces F_x, F_y, F_z and moments M_x, M_y, M_z from instrumented measuring hubs on the left and right rear wheels of a passenger vehicle, sampled at 300 Hz over 218 s of test track (the anonymized record ships with the reference implementation, so this example regenerates like all others). The record is distinctly non-Gaussian – channel kurtosis up to 5.68, crest factors up to 8.70 – which is precisely the regime where Gaussian synthesis under-tests. Nonparametric stationarity checks (reverse-arrangements and runs tests on 32-segment mean-square and kurtosis sequences, provided by the implementation) reject weak-sense stationarity on 3 of 12 channels at $\alpha = 0.01$, with borderline scores rather than distinct regimes: the file was compiled from road

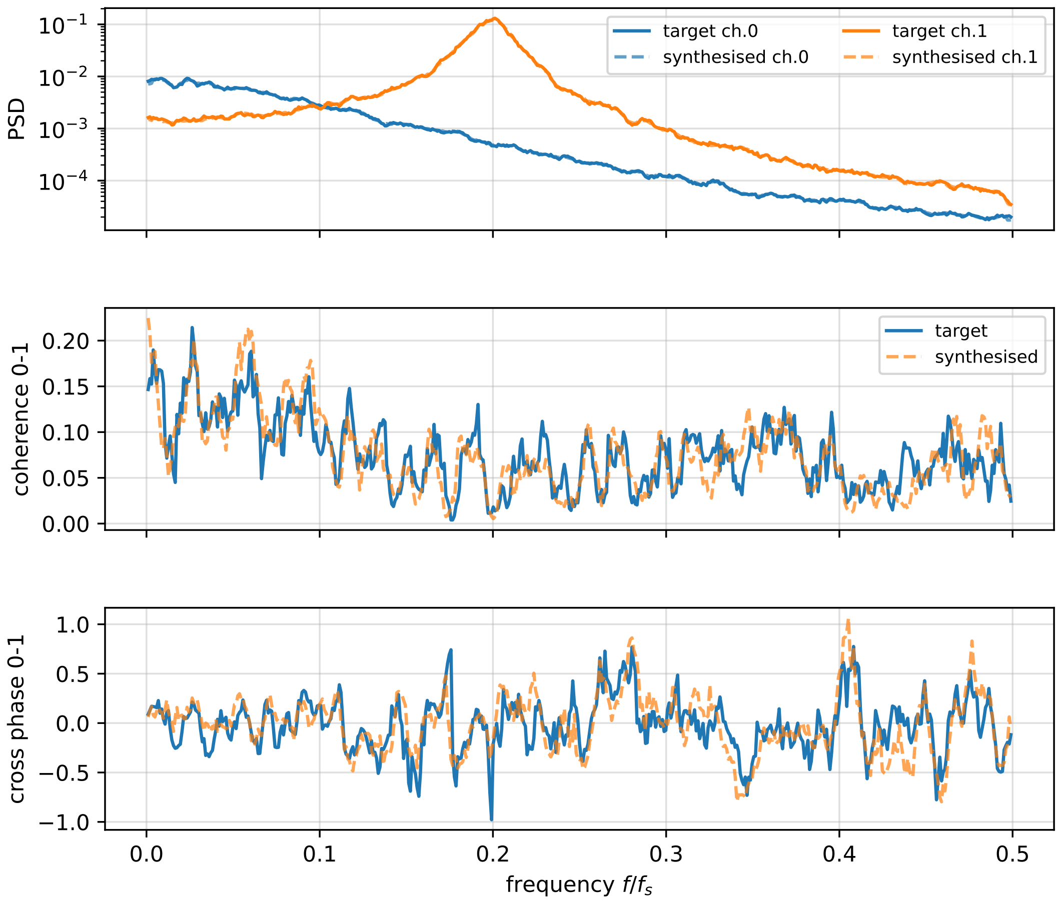

Figure 7: Target CSD versus a multitaper estimate over 32 shaped blocks: PSD per channel (top), coherence (middle) and cross phase (bottom).

sections of similar character, so a single spectral model is adequate here.

Records that mix distinct surfaces call for one model per section; this *multimodel* synthesis (sectionwise estimation with crossfaded or endpoint-matched joints, also part of the implementation) is validated below.

Targets are estimated as before (multitaper CSD, $NW = 4$, 64 segments of 1024 samples; skewness and kurtosis per channel plus the six left/right co-kurtosis tuples $(k, k, k{+}1, k{+}1)$ – 30 moment targets in total). Table 5 lists all targets and achieved values over 32 blocks; every moment is reproduced to three decimals. Figure 8 compares a measured excerpt with a synthesized block for the least and most heavy-tailed channels, and Figure 9 shows the CSD reproduction for the strongly coupled F_z left/right pair: the low-frequency coherence peak of 0.96 and the cross phase are tracked, with phase scatter confined to bins of vanishing coherence, as the estimator variance predicts.

Two engineering numbers from this example quantify earlier claims. First, conditioning: the maximum pairwise coherence over all bins is 0.9956 and the worst bin condition number of $\hat{\mathbf{G}}$ is $\sim 4 \times 10^7$ – Cholesky still succeeds, but this record sits close to the regularization regime discussed in the estimation section. Second,

Contact: luc@sensemagic.nl • www.sensemagic.nl

throughput: one optimized 12-channel block of 1024 samples (3.4 s of drive signal) takes ~383 ms on the workstation of Table 3 – roughly three times faster than real time.

channel	$\hat{\mu}_{(k,k,k)}$	achieved	$\hat{\mu}_{(k,k,k,k)}$	achieved
FXMRHL	0.301	0.301 ± 0.0000	3.620	3.620 ± 0.0000
FXMRHR	0.667	0.667 ± 0.0000	4.868	4.868 ± 0.0000
FYMRHL	-0.404	-0.404 ± 0.0000	3.507	3.507 ± 0.0000
FYMRHR	0.449	0.449 ± 0.0000	3.775	3.775 ± 0.0000
FZMRHL	-0.075	-0.075 ± 0.0000	2.843	2.843 ± 0.0000
FZMRHR	0.184	0.184 ± 0.0000	3.479	3.479 ± 0.0000
MXMRHL	-0.362	-0.362 ± 0.0000	4.777	4.777 ± 0.0000
MXMRHR	0.418	0.418 ± 0.0000	3.848	3.848 ± 0.0000
MYMRHL	0.089	0.089 ± 0.0000	4.331	4.331 ± 0.0000
MYMRHR	-0.162	-0.162 ± 0.0000	4.834	4.834 ± 0.0000
MZMRHL	0.203	0.203 ± 0.0000	5.680	5.680 ± 0.0000
MZMRHR	0.065	0.065 ± 0.0000	3.906	3.906 ± 0.0000

left/right pair	$\hat{\mu}_{\mathrm{i}}$	achieved
FXMRHL/FXMRHR	1.315	1.315 ± 0.0000
FYMRHL/FYMRHR	1.155	1.155 ± 0.0000
FZMRHL/FZMRHR	1.413	1.413 ± 0.0000
MXMRHL/MXMRHR	1.382	1.382 ± 0.0000
MYMRHL/MYMRHR	1.391	1.391 ± 0.0000
MZMRHL/MZMRHR	1.161	1.161 ± 0.0000

Table 5: Road-record MIMO example ($N_j = 12$, $N_t = 1024$): per-channel skewness and kurtosis targets (top) and left/right co-kurtosis targets (bottom) versus achieved values, mean ± standard deviation over 32 blocks.

Multimodel synthesis of a two-regime record. The multimodel claim of the estimation section gets a minimal worked validation on a record with a genuinely non-stationary structure: a Gaussian variant of the two-channel surrogate of the MIMO example (same coloring and shared-source correlation), whose second half is roughened – level ×2.5 with a mild memoryless tail-heavying – so the two halves have kurtosis 2.9/4.8. The reverse-arrangements test on the 32-segment mean-square sequence rejects stationarity decisively ($|z| = 4.3$). A single model estimated over the full record reproduces what it was fed – the *pooled* statistics: its synthesis is stationary by construction ($|z| = 1.1$) and its two halves land at kurtosis 7.9/7.8, matching neither regime.

This is the variance-mixture effect: pooling two Gaussian-like regimes of different level manufactures heavy tails (≈ 7.5 here) that no section of the real environment possesses, and the synthesizer then faithfully reproduces the artifact everywhere. Splitting the record into 8 sections, estimating one CSD-plus-moment model each (multitaper, $N_{\mathrm{fft}} = 2048$) and synthesizing four crossfaded blocks per section restores the regime structure: the segment mean-square profile tracks the measured step (Figure 10, $|z| = 5.0$) and the per-half kurtosis returns to 3.0/4.8. The block rotation used by the crossfade is a circular shift – a pure linear phase, statistically neutral – so the joints add no spectral or moment distortion beyond the fade itself.

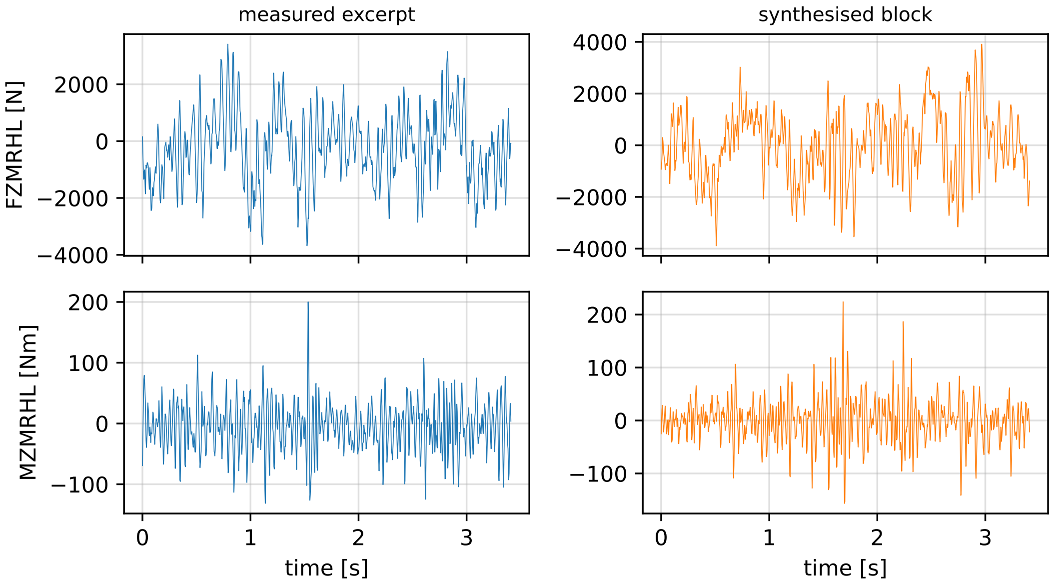

Figure 8: Measured road record versus synthesized block for the vertical force (top, kurtosis 2.8) and the aligning moment (bottom, kurtosis 5.7): the synthesis reproduces the bursty, heavy-tailed character of the measured loads.

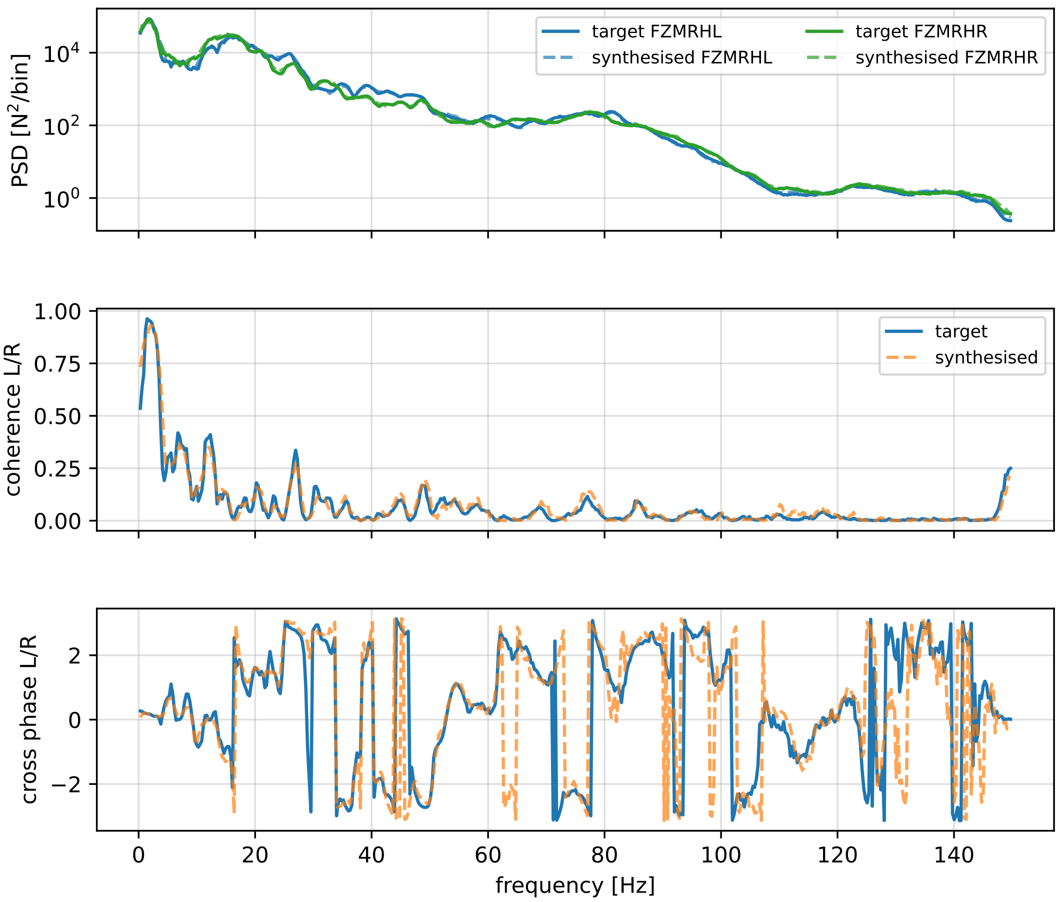

Figure 9: Road-record CSD reproduction for the F_z left/right pair: PSD (top), coherence (middle) and cross phase (bottom); target versus a multitaper estimate over 32 shaped blocks.

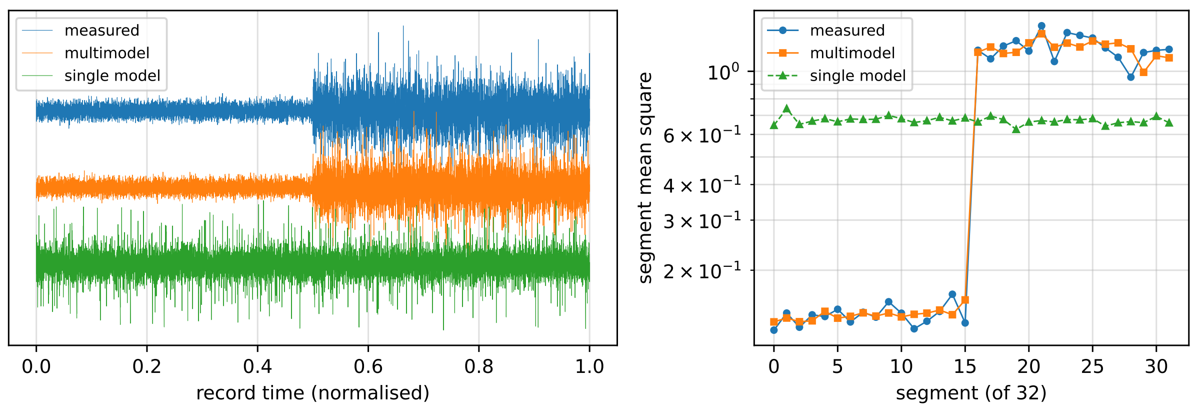

Figure 10: Multimodel validation on a two-regime composite record (channel 0 shown). Left: measured record versus multimodel and single-model syntheses (offset for clarity); the single compromise model spreads the pooled heavy tails uniformly. Right: 32-segment mean-square profiles; the multimodel synthesis (8 sections, 4 crossfaded blocks each) tracks the regime step, the single model cannot.

Contact: luc@sensemagic.nl • www.sensemagic.nl

Appendix A: Logical vs physical moments – a Nyquist-tail caveat

The moments $P_{\mathbf{i}}$ and $M_{\mathbf{i}}$ defined above are moments of the discrete signal $x_{k,t}$ evaluated on the sample grid $t = 0, \ldots, N_t - 1$. This is a deliberate departure from Steinwolf's formulation [5, 7], which evaluates the kurtosis on the *continuous* periodic inverse Fourier reconstruction; the present method works entirely with the discrete periodic FFT sequence, and the distinction matters for the reasons developed in this appendix. The signal that physically reaches the device under test is the continuous-time reconstruction $x_k(\tau)$ produced by the DAC and its analog reconstruction filter. These two signals coincide asymptotically as spectral content is pushed below Nyquist, but diverge for content that approaches Nyquist – and the divergence can be substantial.

A single-line example makes this concrete. Let x_k have all its energy in one spectral bin f_0 with amplitude A. The physical continuous-time signal is a sine wave with kurtosis $M_{k^{(4)},\text{phys}} = 3/2$ regardless of f_0. The sampled sequence, however, is $A\cos(2\pi t f_0/N_t + \psi)$ with $t \in \langle 0, \ldots, N_t - 1\rangle$; for $f_0 \ll N_t/2$ this samples the sine densely and the sample-moment kurtosis $M_{k^{(4)}} \to 3/2$, but as $f_0 \to N_t/2$ the sampled sequence becomes the two-valued alternation $[+A, -A, +A, -A, \ldots]$ whose sample-moment kurtosis is 1. The worst-case discrete-versus-physical mismatch is a full ≈33% in kurtosis, achieved precisely at Nyquist.

This is the engineering reason for the requirement $h_{kj,N_t/2} = 0$ imposed at the outset: no realizable reconstruction filter can pass content at the Nyquist bin cleanly, so any energy the optimizer placed there would be physically distorted by the analog path before reaching the plant, and the sample-domain moment computed there does not correspond to any physical reality. Setting the Nyquist bin to zero is thus both a mathematical convenience – it removes a special-case real-only frequency bin from the cosine expansion – and a physical necessity for any signal that will be played out through a DAC. In practice, we go further and recommend a guard band above $f_s/4$ (half the Nyquist frequency) with a target CSD of zero. This keeps the mismatch below ∼1 % for spectra of practical interest.

Two operating regimes result. In the *measured-target regime* the mismatch is benign by cancellation: target moments are sample moments of an ADC record, and the synthesizer produces sample moments of the same statistical character. If the measurement ADC and the playback DAC use the same sample rate and comparable anti-alias / reconstruction filtering, the Nyquist-tail distortion is applied identically on both sides and cancels; the physical playback then matches the physical original in moment even though the intermediate quantities being matched live in the sampled domain. This is exactly the practical outcome the end user cares about. In the *analytic-target regime* the user specifies $\mu_{\mathbf{i}}$ from theory or a spec, and should either band-limit the target CSD as above, or interpret the shaped moments as digital-domain quantities that the digital controller consumes directly – a legitimate interpretation, since the controller sees the sampled signal and most plant dynamics of practical interest live well below Nyquist.

A related linear effect – the $|\text{sinc}(f/f_s)|$ amplitude taper of a zero-order-hold DAC together with the amplitude response of the analog reconstruction and anti-alias filters – distorts the delivered $|H(f)|$ and hence downstream physical moments even in the measured-target regime. This is a rig-integration matter, handled either by compensating in the reconstruction filter or by pre-shaping H; it is outside the scope of the synthesizer proper.

Appendix B: Endpoint continuity for streamed blocks

Two common use cases require control over block boundaries: (i) *periodic FRF estimation*, where a single block is repeated N times and the first M repetitions are discarded so the response settles into a periodic steady state before averaging; and (ii) *live streaming*, where independent blocks – each with fresh random phases – are concatenated on the fly for extended shaker runs.

Case (i) is free of seam artifacts, but not of start-up and shut-down ones. The inverse-FFT signal is periodic with period N_t by construction: $x_{k,N_t} = x_{k,0}$, and the derivative is likewise periodic. Repeating one block produces a smooth concatenation on the ring $\mathbb{Z}/N_t\mathbb{Z}$; the transient observed at the test rig is entirely the plant's response to a periodic input, which is exactly what one wishes to average out. In practice, however, the *first* and *last* blocks still start and end from rest, where an arbitrary boundary value $x_{k,0}$ produces a discontinuity when the drive is switched on or off. One therefore still wants defined start and end values (typically zero) together with short tapers (or the explicit boundary constraints of Case (ii) below), even though the periodic repetitions themselves splice seamlessly.

Case (ii) requires a fringe constraint. Successive blocks B and $B{+}1$ share their seam at $t = N_t \equiv 0$. Unless $x^{B+1}_{k,0} = x^{B}_{k,N_t} \equiv x^{B}_{k,0}$ (and the analogous condition on the slope), the stream has a jump or a slope kink at every block boundary, showing up in the effective spectrum as broadband leakage.

The synthesizer fixes the block's head value and slope to prescribed constants $c^{(0)}_k$, $c^{(1)}_k$ for every block; consecutive blocks then splice with C^1 continuity by construction. Zero is the natural default and is used throughout the reference implementation, giving a signal that starts and returns to rest at every block boundary.

The head value and its time derivative follow directly from the cosine expansion of $x_{k,t}$:

$$x_{k,0} \;=\; \frac{2}{N_t}\sum_{f=1}^{N_t/2-1}\sum_{j=0}^{N_j-1} h_{kjf}\,\cos(\theta_{kjf}+\psi_{jf}) \;=\; \frac{2}{N_t}\sum_{f,j}\mathrm{Re}\left\{H_{kjf}\,u_{jf}\right\}, \tag{0.38}$$

$$\dot{x}_{k,0} \;=\; -\frac{2}{N_t}\sum_{f=1}^{N_t/2-1}\sum_{j=0}^{N_j-1} \omega_f\, h_{kjf}\,\sin(\theta_{kjf}+\psi_{jf}) \;=\; -\frac{2}{N_t}\sum_{f,j}\omega_f\;\mathrm{Im}\left\{H_{kjf}\,u_{jf}\right\}, \tag{0.39}$$

with $\omega_f = 2\pi f/N_t$. The synthesizer adds the endpoint terms to the loss,

$$\Xi_{\mathrm{ep}} \;=\; \frac{1}{2}\sum_{k=0}^{N_j-1}\left[w^{(0)}_k\left(x_{k,0}-c^{(0)}_k\right)^2 + w^{(1)}_k\left(\dot{x}_{k,0}-c^{(1)}_k\right)^2\right], \tag{0.40}$$

and the gradients follow immediately:

$$\frac{\partial x_{k,0}}{\partial \psi_{qg}} \;=\; -\frac{2}{N_t}\;\mathrm{Im}\left\{H_{kqg}\,u_{qg}\right\}, \tag{0.41}$$

$$\frac{\partial \dot{x}_{k,0}}{\partial \psi_{qg}} \;=\; -\frac{2\omega_g}{N_t}\;\mathrm{Re}\left\{H_{kqg}\,u_{qg}\right\}. \tag{0.42}$$

No additional FFT is required: the endpoint contribution to the total gradient is a single complex multiply–add per (q,g) per constrained channel. This is the degenerate case of Proposition 1 with the sample-average replaced by point evaluation at $t{=}0$; the "FFT of the pointwise gradient" collapses to a Kronecker delta and evaluates trivially. Constraining the block's tail is redundant given the block's built-in periodicity and given that fixing every block's head simultaneously fixes the tail to which each previous block will connect.

The endpoint constraint removes $2N_j$ degrees of freedom from the $N_j(N_t/2-1)$-dimensional phase space – a fraction that vanishes as $N_t \to \infty$ – and so interacts benignly with the moment targets at any practically useful block length.

Appendix C: Benchmark against dedicated crest-factor optimizers

Minimum-crest multisine design has a four-decade literature of specialized algorithms, and the five single-channel setups collected by Retzler *et al.* [19] – flat and shaped amplitude ladders, sparse and dense harmonic grids, block lengths from 800 to 8192 – are the sharpest published yardstick, with min/avg sampled-crest statistics over 1000 random starts for four dedicated methods.

Table 6 repeats the experiment with the present framework's default optimizer (SciPy's L-BFGS-B), from 25 independent random starts per setup, with two objective routes. The first is the native β-continued logcosh surrogate of the main text; it lands within a few percent of the dedicated optimizers on every setup: average crest 0.1–2% above the best published method (the widest gap on the shaped-amplitude setup B), and on the sparse-grid setups C and D the 25-run minima reproduce the published 1000-run minima to three decimals, while on the dense flat gratings A and E the specialists retain a ≈1.5% edge at the minimum.

The second route makes the subsumption point instead: Guillaume's ℓ_p objective itself, entered as the scaled-function target $g(z) = |z|^p$ with p doubling from 4 to 512 – no crest-specific code at all, just the two pointwise functions. It reproduces the same character but trails by a wider margin on the flat gratings (up to ≈7.5% average crest on setup E): the far steeper, more ill-conditioned gradient of $|z|^p$ at $p = 512$ makes the coarse eight-stage continuation a less reliable guide for L-BFGS-B's line search than it is for the smooth logcosh surrogate.

Substituting NLopt's CCSAQ for this route alone recovers the earlier accuracy (setup E: average crest 1.3928, ≈1.8% above published) – a concrete instance of the two optimizers, despite sharing the identical analytic gradient, suiting different regions of the objective landscape. A run takes costs 0.1 to 1 s single-core depending on setup and route. We conclude that a general moment-shaping framework buys entry to the specialists' benchmark at the cost of the last few percent of crest – and that anyone needing that percent can graft the dedicated polishers [18, 19] onto our optimum, or, for steep high-order objectives, swap in CCSAQ, since all methods agree on the character of the solution.

The comparison closes a historical loop. The same 31- and 100-tone flat multisines (setups A and E) served Steinwolf [5] to demonstrate crest reduction by *closed-form* phase selection: his analytic assignment reaches a crest of ≈1.5, and sequential minimization of even moments up to order 128 reaches ≈1.4, both quoted on multi-block quasi-random realizations rather than one optimized period, so the numbers are indicative rather than directly comparable. They are nonetheless consistent with where both of our objective routes land before the dedicated Chebyshev-type machinery buys the final percent.

In the bioimpedance literature, where excitation amplitude is capped by tissue linearity, Ojarand and Min [30] report 1.365 for 26 flat consecutive tones as the lowest published crest of its class, with tabulated small-tone-

Contact: luc@sensemagic.nl • www.sensemagic.nl

count optima in [31] – the same regime as setup A, reached there by exhaustive phase search.

method	A	B	C	D	E
Van der Ouderaa *et al.* [17] (clipping)	1.4637/1.5468	1.4579/1.5372	2.096/2.0968	2.076/2.1944	1.4857/1.5564
Guillaume *et al.* [18] (ℓ_p Chebyshev)	1.3563/1.4085	1.4042/1.437	2.0139/2.0142	1.9877/2.0063	1.3565/1.3727
Retzler *et al.* [19] (nonlinear opt.)	1.3513/1.4041	1.4004/1.4316	2.011/2.0123	1.9815/1.9961	1.3512/1.3683
Janeiro *et al.* [32] (bee colony)	1.5409/1.6314	1.4181/1.4978	2.011/2.0124	1.9862/2.0257	1.7858/1.8947
this work, crest surrogate (β-continued)	1.3727/1.4130	1.4062/1.4597	2.0112/2.0152	1.9818/2.0050	1.3720/1.3888
this work, ℓ_p scaled-function target	1.4192/1.4846	1.4225/1.4923	2.0258/2.0324	2.0232/2.0538	1.4483/1.4712

Table 6: Sampled crest factor (min/avg over random starts) on the five multisine benchmark setups of Retzler *et al.* [19]. Published rows: 1000 starts, quoted from [19]. Our rows: 25 starts per setup, warm-started continuation (β doubled 5–2560 for the surrogate, p doubled 4–512 for ℓ_p), SciPy's L-BFGS-B with analytic gradients throughout.

References

[1] M. Shinozuka and C.-M. Jan. Digital simulation of random processes and its applications. *Journal of Sound and Vibration*, 25(1):111–128, 1972. doi: 10.1016/0022-460X(72)90600-1.

[2] David O. Smallwood and Thomas L. Paez. A frequency domain method for the generation of partially coherent normal stationary time domain signals. *Shock and Vibration*, 1(1):45–53, 1993. doi: 10.3233/SAV-1993-1106. Sandia National Laboratories.

[3] David O. Smallwood. Generation of stationary non-Gaussian time histories with a specified cross-spectral density. *Shock and Vibration*, 4(5-6):361–377, 1997. doi: 10.1155/1997/713593. Sandia National Laboratories.

[4] Xuli Cui, Huaihai Chen, Xudong He, and Shuangyan Jiang. Matrix power control algorithm for multi-input multi-output random vibration test. *Chinese Journal of Aeronautics*, 24(6):741–748, 2011. doi: 10.1016/S1000-9361(11)60087-3.

[5] Alexander Steinwolf. Random vibration testing with kurtosis control by IFFT phase manipulation. *Mechanical Systems and Signal Processing*, 28:561–573, 2012. doi: 10.1016/j.ymssp.2011.11.001. Published online 2011.

[6] Alexander Steinwolf. Add "Time Variable Replication" to your TWR shaker testing. *Sound and Vibration*, 2011. AST Consulting, Ltd., Auckland, New Zealand.

[7] Alexander Steinwolf. Modeling stationary non-Gaussian vibrations in digital systems. In *Analysis and Simulation of Non-Gaussian Random Vibrations*, chapter 3, pages 62–74. Naukova Dumka, Kiev, 1993. In Russian; chapter and page numbers from an unofficial English translation.

[8] Andreas Saathoff, Franz-Josef Stolze, and Harald Zenner. Creation of a synthetic road for multiaxial fatigue limit simulation at automobiles. *Materialprüfung*, 48(4):164–167, 2006. doi: 10.3139/120.100722.

[9] Richard H. Byrd, Peihuang Lu, Jorge Nocedal, and Ciyou Zhu. A limited memory algorithm for bound constrained optimization. *SIAM Journal on Scientific Computing*, 16(5):1190–1208, 1995. doi: 10.1137/0916069.

[10] Pauli Virtanen, Ralf Gommers, Travis E. Oliphant, Matt Haberland, Tyler Reddy, David Cournapeau, Evgeni Burovski, Pearu Peterson, Warren Weckesser, Jonathan Bright, Stéfan J. van der Walt, Matthew Brett, Joshua Wilson, K. Jarrod Millman, Nikolay Mayorov, Andrew R. J. Nelson, Eric Jones, Robert Kern, Eric Larson, C. J. Carey, İlhan Polat, Yu Feng, Eric W. Moore, Jake VanderPlas, Denis Laxalde, Josef Perktold, Robert Cimrman, Ian Henriksen, E. A. Quintero, Charles R. Harris, Anne M. Archibald, Antônio H. Ribeiro, Fabian Pedregosa, Paul van Mulbregt, and SciPy 1.0 Contributors. SciPy 1.0: fundamental algorithms for scientific computing in Python. *Nature Methods*, 17:261–272, 2020. doi: 10.1038/s41592-019-0686-2.

[11] Krister Svanberg. A class of globally convergent optimization methods based on conservative convex separable approximations. *SIAM Journal on Optimization*, 12(2):555–573, 2002. doi: 10.1137/S1052623499362822.

[12] Steven G. Johnson. The NLopt nonlinear-optimization package. `https://github.com/stevengj/nlopt`, 2007–present.

[13] S. O. Rice. Mathematical analysis of random noise. *Bell System Technical Journal*, 23(3):282–332, 1944. doi: 10.1002/j.1538-7305.1944.tb00874.x.

[14] Keng D. Hsueh and Roger P. Hamernik. A generalized approach to random noise synthesis: Theory and computer simulation. *Journal of the Acoustical Society of America*, 87(3):1207–1217, 1990. doi: 10.1121/1.398795.

[15] Steven R. Winterstein. Nonlinear vibration models for extremes and fatigue. *Journal of Engineering Mechanics*, 114(10):1772–1790, 1988. doi: 10.1061/(ASCE)0733-9399(1988)114:10(1772).

[16] Manfred R. Schroeder. Synthesis of low-peak-factor signals and binary sequences with low autocorrelation. *IEEE Transactions on Information Theory*, 16(1):85–89, 1970. doi: 10.1109/TIT.1970.1054411.

[17] Edwin Van der Ouderaa, Johan Schoukens, and Jean Renneboog. Peak factor minimization using a time-frequency domain swapping algorithm. *IEEE Transactions on Instrumentation and Measurement*, 37(1): 145–147, 1988. doi: 10.1109/19.2684.

[18] Patrick Guillaume, Johan Schoukens, Rik Pintelon, and István Kollár. Crest-factor minimization using nonlinear Chebyshev approximation methods. *IEEE Transactions on Instrumentation and Measurement*, 40 (6):982–989, 1991. doi: 10.1109/19.119778.

[19] András Retzler, Bence Cseppentő, Jan Swevers, Joris Gillis, László Leitold, and Zsolt Kollár. Improved crest factor minimization of multisine excitation signals using nonlinear optimization. *Automatica*, 146: 110654, 2022. doi: 10.1016/j.automatica.2022.110654.

[20] R. W. Gerchberg and W. O. Saxton. A practical algorithm for the determination of phase from image and diffraction plane pictures. *Optik*, 35:237–246, 1972.

[21] J. R. Fienup. Phase retrieval algorithms: a comparison. *Applied Optics*, 21(15):2758–2769, 1982. doi: 10.1364/AO.21.002758.

[22] Julius S. Bendat. *Nonlinear System Techniques and Applications*. Wiley-Interscience, New York, 1998.

[23] Rik Pintelon and Johan Schoukens. *System Identification: A Frequency Domain Approach*. Wiley-IEEE Press, 2nd edition, 2012.

[24] Johan Schoukens, Rik Pintelon, Tadeusz Dobrowiecki, and Yves Rolain. Identification of linear systems with nonlinear distortions. *Automatica*, 41(3):491–504, 2005. doi: 10.1016/j.automatica.2004.10.004.

[25] David J. Thomson. Spectrum estimation and harmonic analysis. *Proceedings of the IEEE*, 70(9):1055–1096, 1982. doi: 10.1109/PROC.1982.12433.

[26] David Slepian. Prolate spheroidal wave functions, fourier analysis, and uncertainty — V: The discrete case. *Bell System Technical Journal*, 57(5):1371–1430, 1978.

[27] Donald B. Percival and Andrew T. Walden. *Spectral Analysis for Physical Applications: Multitaper and Conventional Univariate Techniques*. Cambridge University Press, 1993.

[28] Andrew T. Walden. A unified view of multitaper multivariate spectral estimation. *Biometrika*, 87(4): 767–788, 2000.

[29] Nicholas J. Higham. Computing a nearest symmetric positive semidefinite matrix. *Linear Algebra and its Applications*, 103:103–118, 1988.

[30] Jaan Ojarand and Mart Min. Recent advances in crest factor minimization of multisine. *Elektronika ir Elektrotechnika*, 23(2):59–62, 2017. doi: 10.5755/j01.eie.23.2.18001.

[31] Jaan Ojarand, Mart Min, and Paul Annus. Crest factor optimization of the multisine waveform for bioimpedance spectroscopy. *Physiological Measurement*, 35(6):1019–1033, 2014. doi: 10.1088/0967-3334/35/6/1019.

[32] Fernando M. Janeiro, Yifan Hu, and Pedro M. Ramos. Peak factor optimization of multi-harmonic signals using artificial bee colony algorithm. *Measurement*, 150:107040, 2020. doi: 10.1016/j.measurement.2019.107040.