

Applying Spectral Representation to Simulation of Stochastic Processes: Scaling, Units, Sidedness

Brian Bingham

Abstract—Physics-based simulation is a critical capability for engineering design because it enables evaluating early-stage prototypes through testing in a variety of virtual environmental conditions. To provide quantitative results, the simulated conditions must be derived from accurate descriptions of the physical environment. Such environments are often modeled as stochastic processes and described by power spectral density functions, e.g., empirical wind and wave spectra. While the relationship between a spectral description and a time series sampled from that description is mathematically well defined, generating sample time series consistent with the physical description requires attention to the details of spectra normalization scaling, engineering unit conventions, and sidedness of the spectral representation. This report synthesizes the basic mathematical background, presented with the intent of providing a self-consistent set of conventions for the practitioner. A reference implementation is provided, implementing both summation of cosines and fast Fourier transform methods, to illustrate the importance of accounting for scaling, units, and sidedness when generating synthetic virtual environments.

Index Terms—time series, stochastic processes, simulation, spectral representation, wave generation

I. INTRODUCTION

Many of the environmental influences on an engineered system can be modeled as stochastic processes where the particular environmental conditions are described by the power spectral density (PSD) of the process. For maritime vessels, waves and wind have important influences on control and perception. Using spectral representations of these influences improves the connection between the simulated environment and field conditions. This study is motivated by our recent work in simulation of environmental influences upon ocean robotic systems at and below the water surface [1], [2], [3].

As pointed out in [4] the variety of nomenclature, conventions and definitions in spectral analysis leads to considerable confusion and often improper use when applying these techniques.

This article describes techniques to generate a time series, $y(t)$, that is a sample function (a realization) of the one dimensional, univariate (1D-1V) stochastic process, $y_o(t)$, with a given one-sided PSD function $S_{y_o}(\omega)$, where ω is angular frequency in rad/s. Methods for simulation of stochastic process include covariance decomposition, auto-regressive moving-average (ARMA), noise shower, scale refinement, turning

band and spectral representation [5]. This report discusses the spectral representation method.

A. Background

Foundational work on spectral representation of a stochastic process was done by Rice [6], [7]. Some of the original approaches to generating a time series with given PSD were based on the design of a linear filter for which white noise input would generate an output with the desired PSD. Sum-of-cosines methods were introduced in [8], [9]. The methods covered in this report include the following:

- Linear filtering and spectral factorization.
- Spectral representation - sum of cosines
- Spectral representation - FFT/IFFT

For general time series analysis see [10] and the references therein.

B. Frequency, Sidedness, Conventions, Notation and Scaling

Our goal is to generate time series realizations of stochastic processes described by spectral representations. Spectral representation is a concise way to describe a physical phenomena (wave, wind, etc.) based on thorough empirical analysis, allowing the simulated time series to embody authentic physical scenarios. Because of this connection between the spectral representation and the time series, it is important to address the following issues:

- Expression of the frequency variable in angular or oscillation frequency.
- Scaling of the Fourier transform and Fourier series resulting from the choice of frequency units.
- Use of the two-sided or one-sided PSD.

Often the details of the conventions used are ambiguous because these issues are not of primary importance in mathematics and physics domains, or in applications where both the transform and inverse transform are used. However, for our engineering applications, maintaining the physical relevance between the expression of the PSD (in either frequency units and either sidedness) and the time series with physically meaningful units is important for the purposes of generating environmental influences that are statistically representative of the underlying spectral descriptions. The contribution of this report is to provide a complete and consistent synthesis of the simulation generation process with explicit notation to make use of the various conventions for expressing PSD sidedness, frequency units and Fourier transform scaling.

B. Bingham is a Professor in the Department of Mechanical and Aerospace Engineering, Naval Postgraduate School, Monterey, CA 93943, USA (ORCID: 0009-0007-8261-3108).

This report was originally prepared as NPS technical report NPS-MAE-23-002, dated 1 April 2023. The present version has been reformatted in the IEEEtran style without changes to the technical content.

II. PRELIMINARIES

A. Integration by Substitution

Given the functional relation $u = \phi(x)$, the following is true for finite, single variable integrals¹:

$$\int_{\phi(a)}^{\phi(b)} y(u)du = \int_a^b y(\phi(x)) \left(\frac{d\phi(x)}{dx} \right) dx. \quad (1)$$

This will be useful when changing variables, e.g., between oscillation and angular frequency.

B. Fourier Transform

1) *Oscillation Frequency*: f [Hz]: When using the oscillation frequency in Hertz, the continuous time Fourier and inverse Fourier transform exhibit symmetry

$$X(f) = \int_{-\infty}^{\infty} x(t)e^{-j(2\pi f)t} dt = \mathcal{F}\{x(t)\} \quad (2a)$$

$$x(t) = \int_{-\infty}^{\infty} X(f)e^{j(2\pi f)t} df = \mathcal{F}^{-1}\{X(f)\} \quad (2b)$$

where we use the notation $\mathcal{F}\{\cdot\}$ for the Fourier transform operator and $\mathcal{F}^{-1}\{\cdot\}$ for the inverse Fourier transform operator.

2) *Angular Frequency*: ω [rad/s]: However, expressing the transforms in terms of angular frequency, $\omega = 2\pi f$, destroys this symmetry². To make matters worse, there are a variety of conventions when it comes to writing the definition of the transform with angular frequency, hence the ‘‘plethora of rival conventions of the definition of the Fourier transform’’³. If we use integration by substitution (1) and the definition (2) results in

$$X(\omega) = \int_{-\infty}^{\infty} x(t)e^{-j\omega t} dt \quad (3)$$

$$x(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} X(\omega)e^{j\omega t} d\omega \quad (4)$$

We can also write the transform pair as

$$X_2(\omega) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} x(t)e^{-j\omega t} dt \quad (5)$$

$$x(t) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} X_2(\omega)e^{j\omega t} d\omega \quad (6)$$

which is similar to the Fourier-Stieltjes transform used commonly in probability theory. Another possible convention is

$$X_3(\omega) = \frac{1}{2\pi} \int_{-\infty}^{\infty} x(t)e^{-j\omega t} dt \quad (7)$$

$$x(t) = \int_{-\infty}^{\infty} X_3(\omega)e^{j\omega t} d\omega \quad (8)$$

3) *Einstein-Wiener-Khinchin Theorem*: The theorem states that, ‘‘the autocorrelation function of a wide-sense-stationary random process has a spectral decomposition given by the power spectrum of that process’’⁴.

Consider a zero-mean, real-valued, wide-sense stationary⁵ $x(t)$. The autocorrelation is defined as

$$R_{xx}(\tau) = E[x(t + \tau)x(t)]$$

where $E(\cdot)$ is the expectation operator. Under reasonable assumptions of continuity and differentiability, the autocorrelation and PSD are Fourier transform pairs. In units of oscillation frequency this is expressed as

$$S_{xx}(f) = \int_{-\infty}^{\infty} R_{xx}(\tau)e^{-j(2\pi f)\tau} d\tau = \mathcal{F}\{R_{xx}(\tau)\} \quad (9a)$$

$$R_{xx}(\tau) = \int_{-\infty}^{\infty} S_{xx}(f)e^{j(2\pi f)\tau} df = \mathcal{F}^{-1}\{S_{xx}(f)\} \quad (9b)$$

It is important to note $S_{xx}(f)$ refers to the **two-sided** PSD. For real-valued processes, the PSD is symmetric about $f = 0$, i.e., $S_{xx}(f) = S_{xx}(-f)$. Often times the **one-sided** PSD, $\Sigma_{xx}(f)$ is used for just positive frequencies. The two are related by

$$\Sigma_{xx}(f) = 2S_{xx}(f), \quad \forall f \geq 0, \quad (10)$$

and the integrals involving the one-sided spectrum have a lower limit of 0 rather than $-\infty$.

This relation allows us to illustrate the relation between PSD and power. The expected power in the process $x(t)$, equivalent to the variance of the process (σ^2), is

$$\begin{aligned} E[x^2(t)] &= \sigma^2 = R_{xx}(\tau = 0) = \int_{-\infty}^{\infty} S_{xx}(f)e^{j(2\pi f)(\tau=0)} df \\ &= \int_{-\infty}^{\infty} S_{xx}(f) df \end{aligned} \quad (11b)$$

$$= 2 \int_0^{\infty} S_{xx}(f) df \quad (11c)$$

$$= \int_0^{\infty} \Sigma_{xx}(f) df \quad (11d)$$

which is the area under the PSD curve.

4) *Power Spectral Density and Angular Frequency*: Expressing the PSD in angular frequency (ω [rad/s]) can be error prone. We define the two-sided PSD in angular frequency as a separate function, $G_{xx}(\omega)$. This expression must also be

¹https://en.wikipedia.org/wiki/Integration_by_substitution

²<http://mathworld.wolfram.com/FourierTransform.html>

³https://en.wikipedia.org/wiki/Fourier_transform#Units_and_duality

⁴https://en.wikipedia.org/wiki/Wiener-Khinchin_theorem

⁵Wide-sense stationarity is a weaker form of strict sense stationarity. For a wide-sense stationary stochastic process the mean and the autocorrelation are time invariant, and the average power is finite.

consistent with the expected power relationships (11), so using the integration by substitution (1) we find

$$\begin{aligned}
E[x^2(t)] &= R_{xx}(\tau = 0) = \int_{-\infty}^{\infty} S_{xx}(f) df & (12a) \\
&= \int_{-\infty}^{\infty} S_{xx}(f = \omega/(2\pi)) \left(\frac{df}{d\omega} \right) d\omega & (12b) \\
&= \frac{1}{2\pi} \int_{-\infty}^{\infty} S_{xx}(f = \omega/(2\pi)) d\omega & (12c) \\
&= \frac{1}{\pi} \int_0^{\infty} S_{xx}(f = \omega/(2\pi)) d\omega & (12d) \\
&= \frac{1}{2\pi} \int_0^{\infty} \Sigma_{xx}(f = \omega/(2\pi)) d\omega & (12e)
\end{aligned}$$

Equations (12) are applicable when provided with a PSD function expressed in oscillation frequency, but carrying out the integration in angular frequency. If the PSD is expressed in angular frequency, the intention may be that

$$G_{xx}(\omega) = \frac{1}{2\pi} S_{xx}(f = \omega/(2\pi)) \quad (13)$$

where $G_{xx}(\omega)$ is the two-sided PSD in angular frequency and the corresponding one-sided PSD in angular frequency is

$$\begin{aligned}
\Gamma_{xx}(\omega) &= 2G_{xx}(\omega) \\
&= \frac{1}{\pi} S_{xx}(f = \omega/(2\pi)) \\
&= \frac{1}{2\pi} \Sigma_{xx}(f = \omega/(2\pi)), \quad \forall \omega \geq 0, & (14)
\end{aligned}$$

which implies that

$$E[x^2(t)] = R_{xx}(\tau = 0) = \int_{-\infty}^{\infty} G_{xx}(\omega) d\omega \quad (15a)$$

$$= 2 \int_0^{\infty} G_{xx}(\omega) d\omega \quad (15b)$$

$$= \int_0^{\infty} \Gamma_{xx}(\omega) d\omega. \quad (15c)$$

III. GENERATING TIME SERIES

A. Summation of Cosines Method

Using the explicit notation above, the simulation formula from [11] can be applied to generate the time series $y^{(i)}(t)$ that is the i -th realization of the stochastic process $y(t)$ described by the two-sided PSD $G_{yy}(\omega)$ expressed in angular frequency. This simulation formula converges to the series as $N \rightarrow \infty$:

$$y^{(i)}(t) = \sqrt{2} \sum_{n=0}^{N-1} A_n \cos(\omega_n t + \phi_n^{(i)}) \quad (16)$$

where

$$A_n = (2 G_{yy}(\omega_n) \Delta\omega)^{1/2}, \quad n = 0, 1, 2, \dots, N-1 \quad (17a)$$

$$\omega_n = n\Delta\omega, \quad n = 0, 1, 2, \dots, N-1 \quad (17b)$$

$$\Delta\omega = \omega_u/N. \quad (17c)$$

The frequency sampling is $\Delta\omega$, the upper cut-off frequency, ω_u , is the frequency beyond which the PSD may be assumed to be zero, $\phi_0^{(i)}, \phi_1^{(i)}, \dots, \phi_{N-1}^{(i)}$ are the i -th realizations of random phase angles distributed uniformly over the interval $[0, 2\pi)$ and

to avoid aliasing the time step (sampling time) is constrained by

$$\Delta t \leq \frac{2\pi}{2\omega_u} = \frac{\pi}{\omega_u} \quad (18a)$$

$$\leq \frac{1}{2} \left(\frac{2\pi}{\Delta\omega} \right) \frac{1}{N} = \frac{\pi}{\Delta\omega N} \quad (18b)$$

$$\leq \frac{1}{2} \left(\frac{1}{\Delta f} \right) \frac{1}{N} = \frac{1}{2\Delta f N} \quad (18c)$$

with the sampling frequency expressed as both angular and oscillation frequencies. The condition

$$A_0 = 0 \text{ or } G_{yy}(\omega_0 = 0) = 0 \quad (19)$$

is necessary and must be forced if $G_{yy}(0) \neq 0$. The simulated time series is periodic with period

$$T_0 = \frac{2\pi}{\Delta\omega}. \quad (20)$$

1) *Periodicity, Simulation Error and Convergence:* By examining the error between the time series ensemble autocorrelation and the desired autocorrelation, we can quantify the quality of the time series reproduction of the target PSD. In [11] the authors show, using [12], that for the simulation equations (16) and (17) the rate of convergence is proportional to $1/N$. It is worth noting that the frequency sampling (17b) of the summation in (16) is analogous to integral approximation by a left Riemann sum, which is consistent with an error function that decreases with $1/N$.

An alternative approach is to use frequency sampling analogous to a middle Riemann sum method by sampling of the spectrum according to

$$\omega_n = n\Delta\omega + \Delta\omega/2, \quad n = 0, 1, 2, \dots, N-1 \quad (21)$$

instead of (17b). The analogy holds that as would be expected for a middle Riemann sum, the convergence when using (21) is proportional to $1/N^2$. Also the periodicity of the resulting time series is doubled to

$$T_o = 4\pi/\Delta\omega. \quad (22)$$

A disadvantage of this approach is that it cannot be implemented with the FFT technique below.

Note that [13] present a slight modification of the summation of cosines formulation that converges on the order of $1/N^4$. The modification only affects the first two terms.

B. Fast Fourier Transform (FFT) Method

The FFT algorithm can be applied to the summation in (16) to improve the computational efficiency of generating the complete sample function at once [14]. The discrete-time Fourier series (DTFS) for finite series⁶ is defined by the forward and inverse transform pair

$$X[k] = \sum_{m=0}^{M-1} x[m] e^{-j\left(\frac{2\pi}{M}\right)k m} \quad (23a)$$

$$x[n] = \frac{1}{M} \sum_{k=0}^{M-1} X[k] e^{j\left(\frac{2\pi}{M}\right)k n} \quad (23b)$$

⁶https://www.princeton.edu/~cuff/ele201/kulkarni_text/frequency.pdf and <https://web.eecs.umich.edu/~fessler/course/451/l/pdf/c5.pdf>

for a time series $x[n]$ and a frequency series $X[k]$ of length M

$$x[n]: n = 0, 1, \dots, M - 1 \quad (24a)$$

$$X[k]: k = 0, 1, \dots, M - 1. \quad (24b)$$

To make use of the computation efficiency of the FFT technique, the summation of cosines simulation formula (16), with frequency sampling (17b), is rewritten in a form consistent with the DTFS as

$$y^{(i)}(p\Delta t) = \text{Re} \left\{ \sum_{n=0}^{M-1} B_n e^{j(n\Delta\omega)(p\Delta t)} \right\} \quad (25a)$$

$$= \text{Re} \left\{ \sum_{n=0}^{M-1} B_n e^{j\left(\frac{2\pi}{M}\right)pn} \right\} \quad (25b)$$

$$p = 0, 1, \dots, M - 1 \quad (25c)$$

where Re indicates the real part and B_n is

$$B_n = \sqrt{2}A_n e^{j\phi_n^{(i)}}, \quad n = 0, 1, \dots, M - 1 \quad (26)$$

and, consistent with (17a),

$$A_n = (2G_{yy}(\omega_n = n\Delta\omega)\Delta\omega)^{1/2}, \quad n = 0, 1, 2, \dots, M - 1 \quad (27)$$

which is equivalent to the spectrum evaluations from (17b) with

$$\Delta\omega = \omega_u/N. \quad (28)$$

Comparing the expressions (17a) and (27) we notice that the former (summation of cosines) produces N coefficients and the latter produces M coefficients. Because the spectrum is negligible for $\omega = n\Delta\omega \geq \omega_u$, the condition

$$B_n = 0, \quad \forall N \leq n \leq M - 1 \quad (29)$$

is imposed. Also, to be consistent with (19), the condition

$$B_0 = 0 \quad (30)$$

is imposed.

The generated time series $y^{(i)}(p\Delta t)$ is periodic with $T_0 = 2\pi/\Delta\omega$ as demonstrated in the summation of cosines (20). Time and frequency sampling are related by

$$T_0 = \frac{2\pi}{\Delta\omega} = M\Delta t \quad (31a)$$

$$\omega_u = \frac{2\pi}{\Delta t} = M\Delta\omega \quad (32)$$

C. Computational Comparison

Generating physically meaningful time series for engineering applications is an important part of an overall *simulation framework*. The simulation framework refers to total simulation scenario being put to use and can contain such aspects as physics engines, sensor and actuator models, visual rendering, communication interfaces, etc. To support engineering analysis in this context, the simulation method should have the following characteristics:

- 1) The error between the PSD (or autocorrelation) of the generated sample function and that of the original stochastic process should be small.

- 2) The periodicity and temporal sampling of the time series should be consistent with the simulation framework.
- 3) The computational requirements of the time series generation should be sufficiently small as to not affect the overall simulation framework. For example, some simulation scenarios require running in real-time.
- 4) The data storage requirements for the time series should be reasonable for the simulation framework.

The numerical convergence ($1/N$, $1/N^2$, or $1/N^4$) affects both the computation and data storage requirements as illustrated by the examples presented in [13].

IV. ILLUSTRATIVE EXAMPLE

A. Equivalent Sampling

For the purposes of making direct, fair comparisons between methods we would like to generate the same sample functions by each method. The summation of cosines method, as expressed in (16) and (17) uses N samples divided equally between zero and the upper limit of ω_u . The temporal sampling is independent, with P time samples based on the choice of Δt . For the FFT method (25) the frequency sample size ($\Delta\omega$) is kept the same, but M samples are used with $M \geq 2N$ in order to prevent aliasing, hence the condition for padding zeros in the FFT series (29). Unlike the summation of cosines method, the FFT method results in an equivalent number of time and frequency samples. This is summarized in Table I.

We can adjust the summations of cosines method to achieve the same frequency and time sampling for both methods. By setting frequency upper limit to $2\omega_u$ we can set $N = M$ without causing aliasing in the FFT method. To achieve equivalent temporal sampling we need to set $P = M$ or equivalently choose

$$\Delta t = \frac{T_0}{M} \quad (33)$$

which satisfies the constraint (18). Combining (18), (31a) and (33) yields

$$\Delta\omega = \frac{2\omega_u}{M}. \quad (34)$$

B. MATLAB Implementation

We consider the numerical example from [11]

$$G_{yy}(\omega) = \frac{1}{4}\sigma^2 b^3 \omega^2 e^{-b|\omega|}, \quad -\infty < \omega < \infty \quad (35)$$

where constants are chosen as $\sigma = 1$ and $b = 1$ s. The cutoff frequency is $\omega_u = 4\pi$ rad/s. We choose the maximum frequency for all methods to be $\omega_{max} = 2\omega_u$ and choose $N = M = 2^{12}$ which results in a $T_0 = 1024$ s and $\Delta t = 0.25$ s. We generate a series of M random phase values, $\phi_n^{(i)}$ for use in all methods.

When implementing the DTFS method (25) with an FFT algorithm, it is important to be explicit about which FFT convention is being used, because conventions for expressing (23) vary across implementations. The MATLAB `fft()` implementation corresponds to the convention in (23a) terms of scaling and the sign of the imaginary exponential⁷. To make

⁷<https://www.mathworks.com/help/signal/ug/discrete-fourier-transform.html>

TABLE I
FREQUENCY AND TIME SAMPLING

	Summation of Cosines (16) and (17)	FFT (25)
Number of Frequency Sample	N	M
Frequency Sample Size	$\Delta\omega = \frac{\omega_u}{N}$	$\Delta\omega = \frac{\omega_u}{N}$
Frequency Upper Limit	ω_u	$M \Delta\omega \geq 2\omega_u$
Number of Time Samples	$P = \frac{T_0}{\Delta t}$	M
Time Upper Limit	T_0	T_0

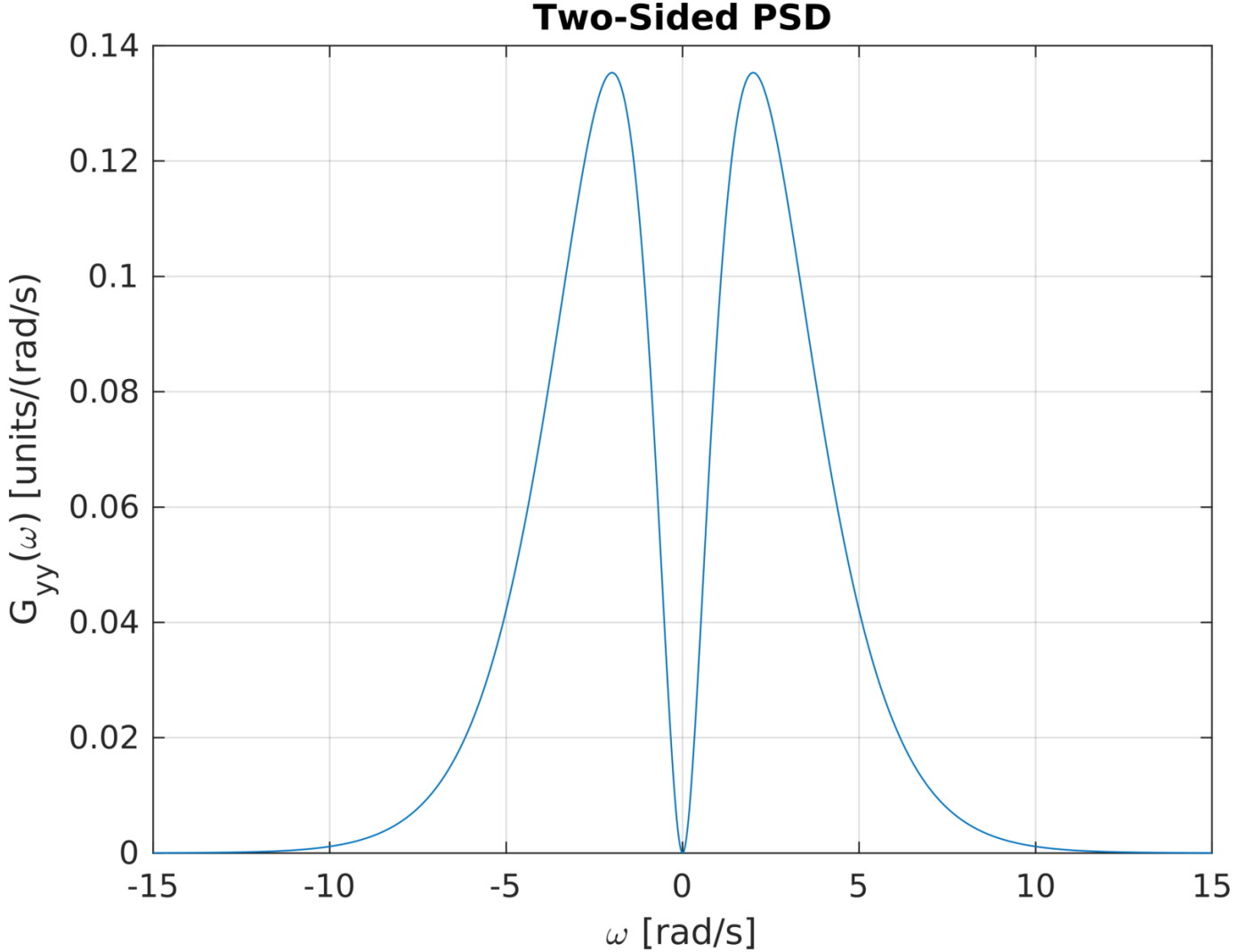


Fig. 1. Two-sided PSD from (35).

use of the FFT convention, we express the series B_n with negative imaginary values

$$\beta_n = \sqrt{2}A_n e^{-j\phi_n^{(i)}}, \quad n = 0, 1, \dots, M-1 \quad (36)$$

which allows us to use MATLAB's FFT of the β_n series.

We implement both the sum of cosines and FFT methods for generating the sample function with the same time/frequency samples sizes of $N = M$ and the same set of M random phase values. The resulting time series, one of infinitely many sample functions, is shown in Figure 2. The figure illustrates

that the sample functions generated by the two methods are equivalent to within numerical precision.

To verify that the generated sample functions are consistent with the original target PSD, we estimate the PSD from each sample function based on Welch's method with 1024 point FFT, Hamming windowing and 44% overlap. The estimated PSDs and the original target PSD are shown in Figure 3. Note that the estimated PSDs are the one-sided PSDs, $\Gamma(\omega)$, while the original PSD from (35) and Figure 1 is two-sided. The figure illustrates that both time series have equivalent estimated PSDs and that they approximate the target PSD. The FFT

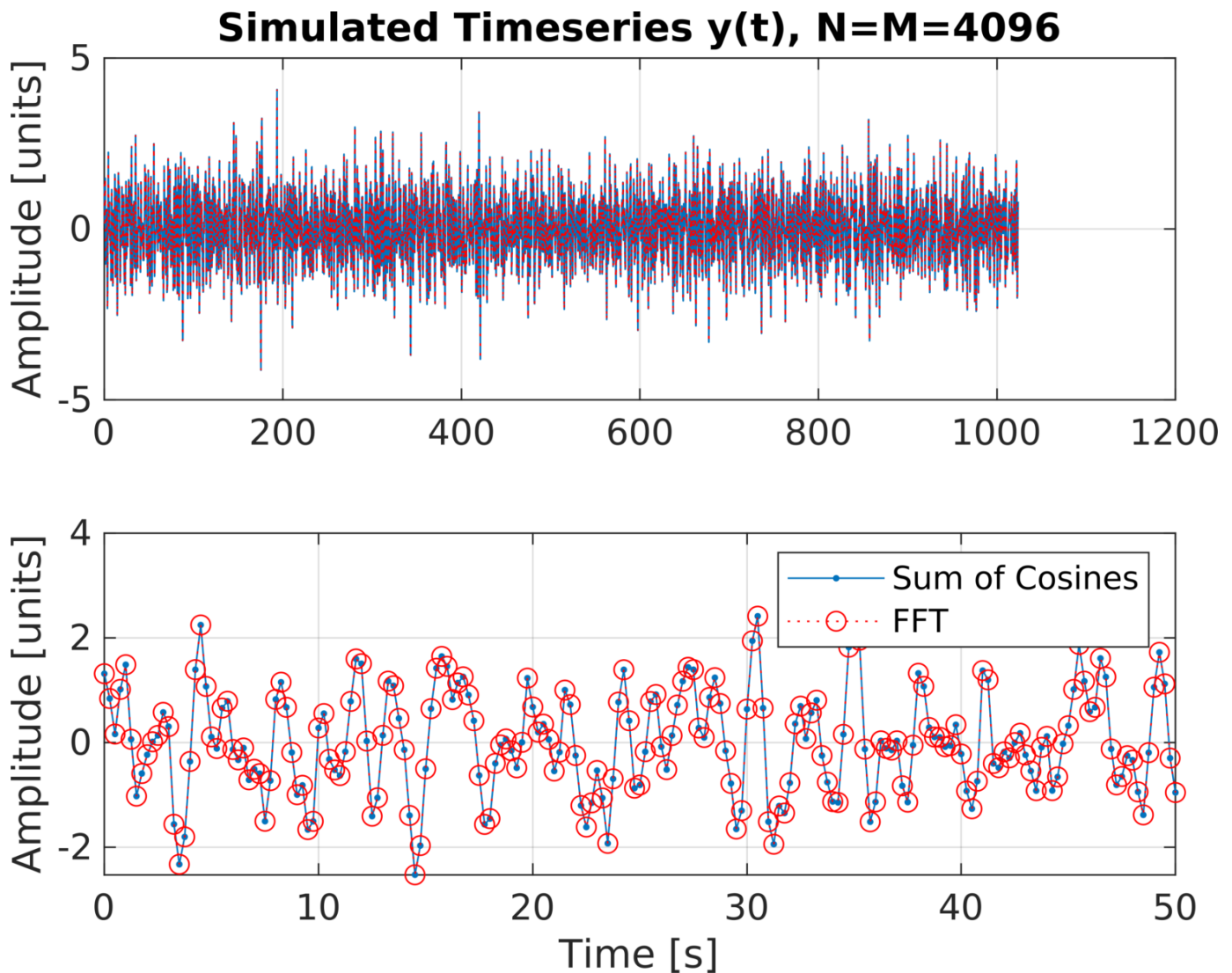


Fig. 2. Two example time series generated by summation of cosines and FFT

method ran ≈ 60 times faster than the summation of cosines method: 0.14 s (summation of cosines) vs. 0.0024 s (FFT). Repeating the tests for various values of M suggests that the speed-up ratio for the FFT method is $T_{soc}/T_{fft} = 0.04M$ where T_{soc}/T_{fft} is the ratio of computation time for the summation of cosines method vs. the FFT method.

The MATLAB source code for this example is co-located with this report at <https://github.com/bsb808/carrel/tree/main/posts/wave-spectra-sim/spectra-from-psd/src>.

V. CONCLUSION

This report documents the implementation details necessary to generate time series samples of physical processes described via an empirical PSD. It is important to maintain consistency with the scaling and conventions employed in order to generate synthetic samples statistically representative of the target phenomena. These details are put into practice in a MATLAB reference implementation to illustrate both summation of cosines and FFT methods for generating equivalent time series

and to show that the simulated time series are statistically representative of the underlying PSD.

REFERENCES

- [1] B. Bingham, C. Agüero, M. McCarrin, J. Klamo, J. Malia, K. Allen, T. Lum, M. Rawson, and R. Waqar, "Toward maritime robotic simulation in Gazebo," in *OCEANS 2019 MTS/IEEE SEATTLE*, 2019, pp. 1–10.
- [2] M. M. Zhang, W.-S. Choi, J. Herman, D. Davis, C. Vogt, M. McCarrin, Y. Vijay, D. Dutia, W. Lew, S. Peters, and B. Bingham, "DAVE aquatic virtual environment: Toward a general underwater robotics simulator," in *2022 IEEE/OES Autonomous Underwater Vehicles Symposium (AUV)*, 2022, pp. 1–8.
- [3] W. Choi, B. Bingham, and R. Camilli, "Faster-than-real-time Hybrid Autonomous Underwater Glider Simulation for Ocean Mapping," *Journal of the Korean Society of Marine Environment and Safety*, vol. 28, pp. 441–450, 2022. [Online]. Available: <http://www.kosomes.or.kr/journalarticle.php?code=83649>
- [4] A. J. Barbour and R. L. Parker, "Normalization of power spectral density estimates," *Computers & Geosciences Research*, vol. 34, pp. 51–68, 2015.
- [5] Y. Cao, H. Xiang, and Y. Zhou, "Simulation of stochastic wind velocity field on long-span bridges," *Journal of Engineering Mechanics*, vol. 126, no. 1, pp. 1–6, 2000.
- [6] S. O. Rice, "Mathematical analysis of random noise," *The Bell System Technical Journal*, vol. 23, no. 3, pp. 282–332, July 1944.

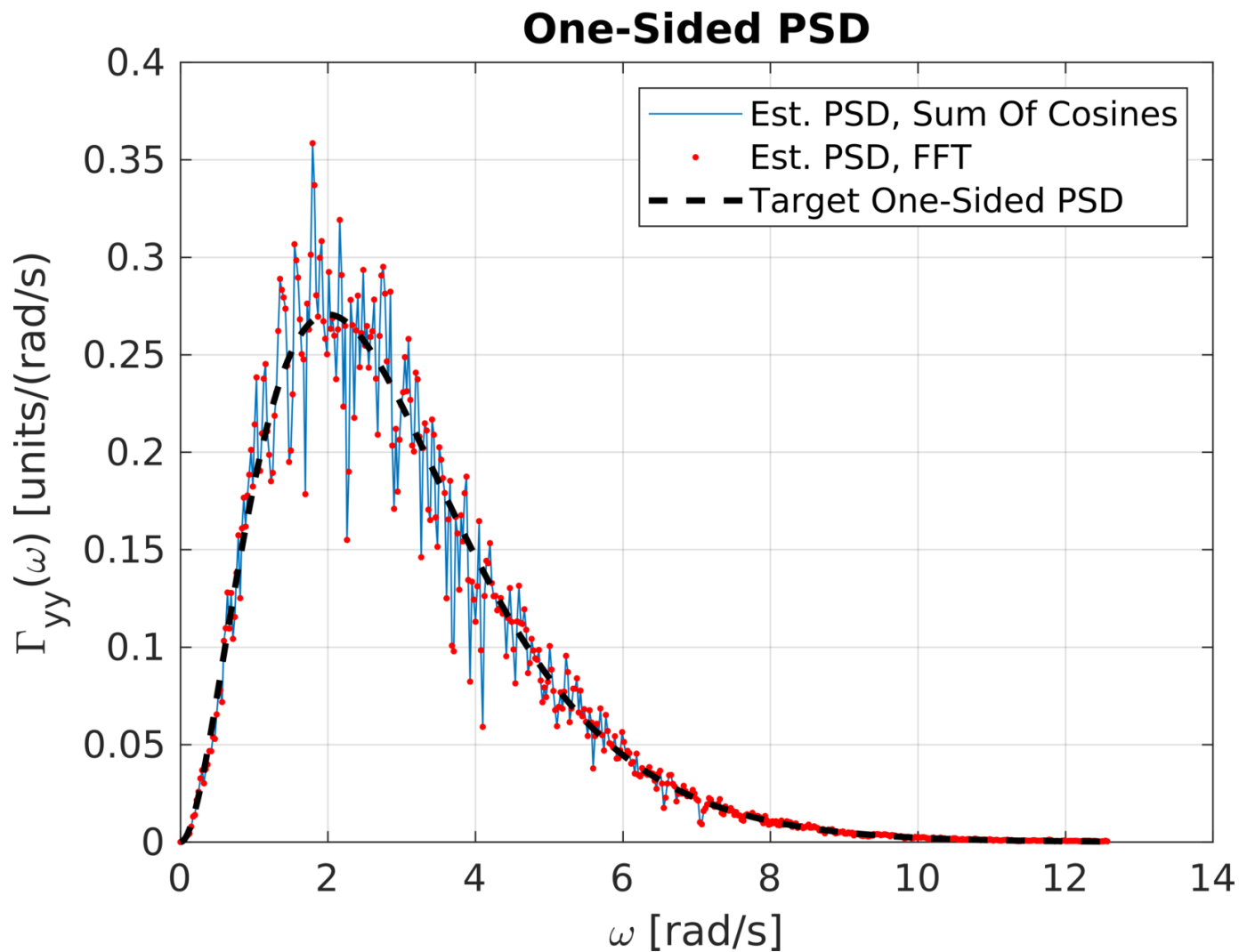


Fig. 3. Estimates of the one-sided PSDs for the two time series from Figure 2 shown alongside the original PSD from Figure 1.

- [7] —, “Mathematical analysis of random noise,” *The Bell System Technical Journal*, vol. 24, no. 1, pp. 46–156, Jan 1945.
- [8] L. E. Borgman, “Ocean wave simulation for engineering design,” *Journal of the Waterways and Harbors Division*, vol. 95, no. 4, p. 557586, 1969.
- [9] M. Shinozuka, “Monte Carlo solution of structural dynamics,” *Computers & Structures*, vol. 2, no. 5, pp. 855 – 874, 1972. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/0045794972900430>
- [10] O. Marchal, “Notes of time series analysis,” Woods Hole Oceanographic Institution, Department of Geology & Geophysics, techreport, 2015. [Online]. Available: <https://www.whoi.edu/fileserver.do?id=225689&pt=2&p=232010>
- [11] M. Shinozuka and G. Deodatis, “Simulation of Stochastic Processes by Spectral Representation,” *Applied Mechanics Reviews*, vol. 44, no. 4, pp. 191–204, 04 1991. [Online]. Available: <https://doi.org/10.1115/1.3119501>
- [12] S. D. Conte and C. De Boor, *Elementary numerical analysis: an algorithmic approach*. SIAM, 2017.
- [13] B. Hu and W. Schiehlen, “On the simulation of stochastic processes by spectral representation,” *Probabilistic Engineering Mechanics*, vol. 12, no. 2, pp. 105 – 113, 1997. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0266892096000392>
- [14] J.-N. Yang, “Simulation of random envelope processes,” *Journal of Sound and Vibration*, vol. 21, no. 1, pp. 73 – 85, 1972. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/0022460X72902076>