Fast and Exact Synthesis of Stationary Multivariate Gaussian Time Series Using Circulant Embedding

Hannes Helgason[a], Vladas Pipiras[b,*], Patrice Abry[a,**]

[a] *Laboratoire de Physique (CNRS UMR 5672), École Normale Supérieure de Lyon (France)*
[b] *CEMAT - Instituto Superior Técnico (Portugal) and University of North Carolina (USA)*

Abstract

A fast and exact procedure for the numerical synthesis of stationary multivariate Gaussian time series with a priori prescribed and well controlled auto- and cross-covariance functions is proposed. It is based on extending the Circulant Embedding technique to the multivariate case and can be viewed as a modification and variation around the Chan and Wood algorithm proposed earlier to solve the same problem. The procedure is shown to yield time series possessing exactly the desired covariance structure, when sufficient conditions are satisfied. Such conditions are discussed theoretically and examined on several examples of multivariate time series models. Issues related to prescribing a priori the spectral structure rather than the covariance one are also discussed. MATLAB routines implementing this procedure are publicly available at `http://www.hermir.org`.

Keywords: Numerical Synthesis, Multivariate Gaussian Series, Circulant Embedding, Stationarity, Time Reversibility.

1. Introduction

This contribution aims at proposing a *fast* and *exact* simulation procedure for synthesizing numerically stationary (P component) multivariate correlated Gaussian time series (of length N), with a priori prescribed auto- and cross-covariance sequences.

By 'exact', we mean that the covariance structure of the synthesized series is exactly the prescribed one (up to the usual machine precision and the generation of pseudorandom numbers – see [1] for discussion), and by 'fast', the computational complexity of the synthesis is $O(P^2 N \log N)$, which is close to the best one could hope for.

1.1. Motivation

The availability of such a procedure is of major relevance in modern signal processing applications. With the steadfast decrease of the costs of collecting and storing data, an increasing number of real-world applications require the processing and analysis of

*Work supported in part by the NSF grants DMS-0505628 and DMS-060866.
**Work supported by the *Young Research Team* award granted by Foundation del Duca, Académie des Sciences, Institut de France, 2007

multiple-channel time series. Typical examples are provided by sensor network monitoring, nowadays increasingly used, e.g., in environmental surveillance [2, 3, 4]. Other examples are provided by functional magnetic resonance imaging, where neural activity is studied via the joint analysis of the time course of multiple neuron responses [5, 6], human body monitoring in medicine [7], or the Internet, where information about traffic flow might be collected jointly at numerous points of presence [8, 9]. Many multivariate time series arise and are studied extensively within time series econometrics [10].

When dealing with multivariate time series, the statistical performance of estimation or test procedures is also more difficult to assess theoretically. This is well illustrated in the context of Internet traffic anomaly detection, where, when aggregated at a high enough resolution, data can be regarded as multivariate Gaussian. Yet, their dependencies are such that qualifying the performance of anomaly detection procedures cannot be done analytically [8]. In such situations, one can resort to Monte Carlo simulations, where fast synthesis of multivariate Gaussian time series with a priori controlled and prescribed dependence structures is crucial.

1.2. State of the Art

In the univariate case $P = 1$, simulation of stationary Gaussian time series has been studied extensively and a number of methods are available. One standard method is based on the Cholesky factorization of the covariance matrix. Though exact, the method has complexity of order $O(N^3)$ and hence is computationally too expensive for simulation of long time series. Other popular methods are based on approximations of the series through the spectral domain and the use of FFT. Though of low computational complexity $O(N \log N)$, these methods are only approximate in general. The method that is both exact and fast, easy to implement, and now often preferable, is based on circulant embedding procedure of the covariance matrix, which is assumed to be known and given [11, 12, 13, 14, 1]. The only drawback of this method is that circulant embedding does not always lead to a valid covariance structure. However, this seems of minor relevance in practice and, moreover, circulant embedding has been justified theoretically for a number of time series models of practical interest. All these and other simulation methods are reviewed and compared nicely in [15].

In the multivariate case $P > 1$, simulation of stationary Gaussian series has been explored much less. Chambers' method [16] extends the univariate spectral methods to the multivariate case starting from a given spectral density matrix. Though fast computationally, it again provides only an approximation to the series. In [17], the univariate circulant embedding approach has been extended theoretically to the very general case of the synthesis of stationary Gaussian vector fields, that is, P-variate series taking values in d-dimension. The shortness of the article (with little or no proofs), with apparent focus on random fields rather than on multivariate series, and the complexity of notation, does however not provide much intuition for why the algorithm actually works or, for that matter, into the way it should be implemented in practice. General comments on the relationship of this work to [17] are made in Section 1.3 below, and a detailed comparison is postponed to Appendix B.

1.3. Contributions and Outline

The extension from the univariate to the multivariate case, the central topic of this contribution, is obtained by a joint circulant embedding of the auto- and cross-covariances

of the original multivariate series, and is described in Section 2. The proposed algorithm works given that circulant embedding leads to a valid covariance structure, to which we refer as *ND assumption* (for "nonnegative definite"). This assumption is studied theoretically and practically in Section 3, along with computational costs and memory requirements, several examples of models of multivariate stationary Gaussian series, and the case where the spectral density matrix rather than the covariance structure is given.

Despite obvious similarities, the present contribution differs from [17]. While both algorithms turn out to be identical in the particular case of time-reversibility (see Section 2), they are in general different in their embedding structures (as detailed in Appendix B). In particular, our algorithm allows powers of 2 for embedding size in general whereas powers of 3 were used in [17]. The approach proposed here may also appear more natural to practitioners as it is based on the univariate matrix embedding. Moreover, sufficient conditions for the algorithm to work are given and checked on several models of practical interest.

For better readability, technical proofs are moved to Appendix A. MATLAB routines are publicly available at `http://perso.ens-lyon.fr/patrice.abry/software.html`.

2. New Multivariate Circulant Embedding Method

2.1. Definitions

Consider a real-valued multivariate stationary Gaussian series

$$X[n] = (X_1[n], X_2[n], \ldots, X_P[n])^T, \quad n \in \mathbb{Z}, \tag{1}$$

with a prescribed covariance structure

$$EX_p[0]X_{p'}[n] =: r_{p,p'}[n], \quad n \in \mathbb{Z},\ 1 \leq p, p' \leq P. \tag{2}$$

(Without loss of generality, we assume that the series has zero mean.) When $p = p'$, we will also write $r_p = r_{p,p}$. Let

$$R[n] = EX[0]X[n]^T = (r_{p,p'}[n])_{1 \leq p,p' \leq P} \tag{3}$$

be the $P \times P$ auto-covariance matrices.

The goal here is to synthesize N samples $X[n]$, $n = 0, \ldots, N-1$, of the series using the covariance structure in Eq. (2). Since we will use ideas from the univariate case, it is useful to write univariate series $\{X_p[n] : n = 0, \ldots, N-1\}$ as column vectors X_p, $p = 1, \ldots, P$. Let

$$\Sigma_{p,p'} = EX_pX_{p'}^T = (r_{p,p'}[n' - n])_{1 \leq n,n' \leq N}$$

be the auto- or cross-covariance matrix of X_p and $X_{p'}$. When $p = p'$, we will also write $\Sigma_p = \Sigma_{p,p}$. The matrices $\Sigma_{p,p'}$, $1 \leq p, p' \leq P$, fully determine the covariance structure of the sequence $X[n]$, $n = 0, \ldots, N-1$. Note also that these matrices are Toeplitz, the matrices Σ_p are symmetric, and the matrices $\Sigma_{p,p'}$ are not symmetric in general. We should also add that we are interested in the case when $N \gg P$.

The synthesis procedure developed here is based on tailoring the circulant embedding technique –now well known and used for the univariate case– to the multivariate problem.

2.2. Derivation of Synthesis Algorithm

2.2.1. Circulant Embedding of Covariance Matrices

Let $\Sigma = (r[n'-n])_{1\le n,n'\le N}$ be a Toeplitz matrix, standing for $\Sigma_{p,p'}$. The aim is to construct a $2M \times 2M$ circulant matrix $\tilde{\Sigma}$ which has Σ appearing as an $N \times N$ block-matrix in the upper-left corner. (M will be taken as N below though other choices of M are also possible in special cases – see Section 2.4 below.) Let

$$\tilde{r}[n] = \begin{cases} r[n], & \text{for } 0 \le n \le M, \\ r[-(2M-n)], & \text{for } M+1 \le n \le 2M-1, \end{cases} \tag{4}$$

and set a circulant matrix $\tilde{\Sigma} = (\tilde{\Sigma}[n,n'])_{1\le n,n'\le 2M}$ as

$$\tilde{\Sigma}[n,n'] = \tilde{r}[(n'-n) \bmod 2M], \tag{5}$$

that is, the first row of $\tilde{\Sigma}$ is $(\tilde{r}[0],\dots,\tilde{r}[2M-1])$ and each row is a cyclic right-shift of the row above it. A sufficient condition for the matrix $\tilde{\Sigma}$ to contain Σ in its upper-left corner is to take $M \ge N$. The matrix $\tilde{\Sigma}$ will be called the *circulant embedding* of the matrix Σ.

The circulant embedding in Eq. (5) is also motivated by the following two reasons. First, when r is the auto-covariance function of a univariate stationary series, we have $r[n] = r[-n]$ and hence the first row of the circulant embedding matrix $\tilde{\Sigma}$ is $(r[0]\cdots r[M-1]r[M]r[M-1]\cdots r[1])$. This is the type of circulant embedding used and studied in the univariate case [13], and hence results from that case could be readily applied to our case. Second, the size $2M$ of the circulant embedding matrix $\tilde{\Sigma}$ is even and, for a suitable choice of M, will become a power of 2. The latter is important when synthesizing longer time series, since our algorithm relies on FFT.

For the embedding of $\Sigma_{p,p'}$, we can use circulant embedding with $M = N$, based on the sequence $\tilde{r}$ becoming

$$(r_{p,p'}[0]\cdots r_{p,p'}[N-1]r_{p,p'}[N]r_{p,p'}[-N+1]\cdots r_{p,p'}[-1]) \tag{6}$$

and being of length $2M = 2N$. For $1 \le p \le p' \le P$, denote the circulant embeddings of the matrices $\Sigma_{p,p'}$ by $\tilde{\Sigma}_{p,p'}$. Again, for $p = p'$, also write $\tilde{\Sigma}_p = \tilde{\Sigma}_{p,p}$. For $1 \le p' < p \le P$, on the other hand, set $\tilde{\Sigma}_{p,p'} = \tilde{\Sigma}^T_{p',p}$, which is also circulant and contains $\Sigma_{p,p'}$ in its upper-left corner. In the latter regard, it is important to note that the definition of $\tilde{\Sigma}_{p,p'}$, $p' < p$, does not correspond in general to that of the circulant embedding of $\Sigma_{p,p'}$, $p' < p$. Indeed, the first row of $\tilde{\Sigma}^T_{p',p}$

$$(r_{p',p}[0]\cdots r_{p',p}[-N+1]r_{p',p}[N]\cdots r_{p',p}[1]),$$

which, by using $r_{p',p}[n] = r_{p,p'}[-n]$, is equal to

$$(r_{p,p'}[0]\cdots r_{p,p'}[N-1]r_{p,p'}[-N]\cdots r_{p,p'}[-1]).$$

The $(N+1)$th entry $r_{p,p'}[-N]$ is in general different from the $(N+1)$th entry $r_{p,p'}[N]$ in the first row (6) of the circulant embedding $\tilde{\Sigma}_{p,p'}$, $p' < p$.

We shall suppose next that the circulant embeddings $\tilde{\Sigma}_{p,p'}$ satisfy the following assumption. See Section 3.2 below for sufficient conditions on this assumption.

ND assumption: The matrices $\tilde{\Sigma}_{p,p'}$, $1 \leq p, p' \leq P$, constitute a valid covariance structure, that is, the block matrix $(\tilde{\Sigma}_{p,p'})_{1\leq p,p'\leq P}$ is nonnegative definite.

Because the discrete Fourier basis diagonalizes circulant matrices, one can write:

$$\tilde{\Sigma}_p = F^* \Lambda_p F, \tag{7}$$

$$\tilde{\Sigma}_{p,p'} = F^* \Lambda_{p,p'} F, \tag{8}$$

where $\Lambda_p = diag(\lambda_p[0], \ldots, \lambda_p[2M-1])$ and $\Lambda_{p,p'} = diag(\lambda_{p,p'}[0], \ldots, \lambda_{p,p'}[2M-1])$ are diagonal, * indicates the conjugate transpose, and F^* is the $2M \times 2M$ Fourier matrix with jth column

$$e_j = \frac{1}{\sqrt{2M}} \left(1, e^{-\imath \frac{2\pi j}{2M}}, \ldots, e^{-\imath \frac{2\pi j(2M-1)}{2M}}\right)^T.$$

The diagonal elements of matrices $\Lambda_{p,p'}$ satisfy

$$\lambda_{p,p'}[m] = \sum_{j=0}^{2M-1} \tilde{r}_{p,p'}[j] e^{-\imath \frac{2\pi jm}{2M}} \tag{9}$$

and can therefore be rapidly computed using FFT. For the diagonal elements of Λ_p, note that, since $r_p[n] = r_p[-n]$,

$$\lambda_p[m] = r_p[0] + r_p[M](-1)^m + 2 \sum_{j=1}^{M-1} r_p[j] \cos\left(\frac{\pi jm}{M}\right).$$

Then, by construction, the diagonal elements of Λ_p (or the eigenvalues of $\tilde{\Sigma}_p$) are real. By ND assumption, $\tilde{\Sigma}_p$ is a covariance matrix and hence the diagonal elements of Λ_p are assumed to be nonnegative. On the other hand, the diagonal elements of $\Lambda_{p,p'}$ are just complex-valued in general. Note also that $\Lambda_{p,p'} = \Lambda^*_{p',p}$ for $p' < p$, since we set $\tilde{\Sigma}_{p,p'} = \tilde{\Sigma}^T_{p',p}$ and since $F\tilde{\Sigma}_{p',p}F^* = \Lambda_{p',p}$ implies $F\tilde{\Sigma}^T_{p',p}F^* = \Lambda^*_{p',p}$ or $\tilde{\Sigma}_{p,p'} = F^*\Lambda^*_{p',p}F$.

2.2.2. Univariate Components

Following the univariate circulant embedding method, we intend to synthesize the random vectors X_p, $p = 1, \ldots, P$, by keeping the first N entries of the real and imaginary parts of

$$\tilde{X}_p := F^* \Lambda_p^{1/2} Z_p, \tag{10}$$

where Z_p is a column vector of size $2M$, such that $Z_p = Z_p^0 + \imath Z_p^1$, with Z_p^0 and Z_p^1 being two independent $\mathcal{N}(0, I_{2M})$ random vectors. The covariance structure of Z_p^0 and Z_p^1 can equivalently be determined from:

$$\left.\begin{array}{ll} EZ_pZ_p^T & = 0, \\ EZ_pZ_p^* & = 2I_{2M}. \end{array}\right\} \tag{11}$$

2.2.3. Dependence Structure of Z_p's

If Eq. (10) is used to synthesize each univariate component of the time series, the cross-covariance will come from the dependence of the random vectors Z_p, $p = 1, \ldots, P$. To obtain the targeted cross-covariance $\Sigma_{p,p'}$ for X_p's, we will consider the conditions on the matrix $EZ_pZ_{p'}^*$ so that $E\tilde{X}_p\tilde{X}_{p'}^* = 2\tilde{\Sigma}_{p,p'}$.

By using Eq. (10), one obtains, for $p \neq p'$:

$$2\tilde{\Sigma}_{p,p'} = E\tilde{X}_p\tilde{X}_{p'}^* = F^*\Lambda_p^{1/2}E[Z_pZ_{p'}^*]\Lambda_{p'}^{1/2}F.$$

Combining this with Eq. (8), and since Λ_p, $\Lambda_{p,p'}$ are diagonal, one gets:

$$\begin{aligned} EZ_pZ_{p'}^* &= 2\,\Lambda_p^{-1/2}\Lambda_{p,p'}\Lambda_{p'}^{-1/2} \\ &=: diag(d_{p,p'}[0], \ldots, d_{p,p'}[2M-1]) =: D_{p,p'}. \end{aligned} \tag{12}$$

We also write $D_{p,p} = 2I_{2M}$, since $EZ_pZ_p^* = 2I_{2M}$.

Remark. Under ND assumption, Λ_p can have diagonal elements of value zero and hence $\Lambda_p^{-1/2}$ appearing in (12) is not defined in principle. However, the rest of the paper continues to be valid even with diagonal elements of value zero if (12) is interpreted in a suitable way. More precisely, set

$$EZ_p[m]Z_{p'}[m]^* = d_{p,p'}[m] = 0, \tag{13}$$

if $\lambda_p[m] = 0$ or $\lambda_{p'}[m] = 0$. The logic with this choice is the following. If say $\lambda_p[m] = 0$, then the mth element $\lambda_p[m]^{1/2}Z_p[m]$ of $\Lambda_p^{1/2}Z_p$ is also zero and, in this sense, the choice of $Z_p[m]$ is irrelevant in (10). On the other hand, the condition (13) can be readily satisfied by $Z_p[m]$ by taking the latter independent of $Z_{p'}[m]$, that is, the choice (13) does not impose any additional restrictions.

2.2.4. Constructing Z_p's

We want to generate the random vectors Z_p, $p = 1, \ldots, P$, jointly so that they satisfy conditions (11) and (12) simultaneously:

$$\begin{aligned} &\text{(i)} \quad EZ_pZ_p^T = 0, \\ &\text{(ii)} \quad EZ_pZ_{p'}^* = D_{p,p'}, \quad \text{for } 1 \leq p, p' \leq P. \end{aligned}$$

Since the matrices $D_{p,p'}$ are diagonal, one can construct vectors $\underline{Z}_m = (Z_1[m] \cdots Z_P[m])^T$ independently for each $m = 0, \ldots, 2M-1$, and replace (i), (ii) above by the conditions:

$$\begin{aligned} &\text{(i')} \quad E\underline{Z}_m\underline{Z}_m^T = 0, \\ &\text{(ii')} \quad E\underline{Z}_m\underline{Z}_m^* = (D_{p,p'}[m,m])_{1\leq p,p'\leq P} =: C_m. \end{aligned}$$

Note that C_m are Hermitian matrices since $D_{p,p'} = D_{p',p}^*$, which follows from the construction of $\tilde{\Sigma}_{p,p'}$, $p' < p$, above. The following result shows in particular, that ND assumption implies nonnegative definiteness of the matrices C_m. For later use, we also formulate the result in terms of matrices $\widehat{R}_m := (\lambda_{p,p'}[m])_{1\leq p,p'\leq P}$.

Theorem 2.1. *ND assumption holds if and only if the matrices C_m, $m = 0, \ldots, 2M-1$, and Λ_p, $p = 1, \ldots, P$, are nonnegative definite, and also if and only if $\widehat{R}_m$, $m = 0, \ldots, 2M-1$, are nonnegative definite.*

Since C_m is Hermitian nonnegative definite under ND assumption, it can be factorized as

$$C_m = A_m A_m^*. \tag{14}$$

In our algorithm, we use A_m from a Cholesky factorization. Other possibilites such as Schur decomposition, though are also possible. Now set

$$\underline{Z}_m = A_m W_m, \tag{15}$$

where $W_m = W_m^0 + \imath W_m^1$ with two independent vectors W_m^0 and W_m^1 being $\mathcal{N}(0, I_P/2)$. (As said earlier, all this is carried out independently for $m = 0, \ldots, 2M-1$, and W_m are independent here.) With this choice, since $EW_m W_m^* = I_P$, one has:

$$E\underline{Z}_m \underline{Z}_m^* = A_m A_m^* = C_m,$$

that is, the condition (ii') above holds. Similarly, since $EW_m W_m^T = 0$,

$$E\underline{Z}_m \underline{Z}_m^T = A_m E[W_m W_m^T] A_m^T = 0, \tag{16}$$

that is, the condition (i') above also holds.

2.2.5. Cross-Covariance Structure

Finally, the desired vectors X_p, $p = 1, \ldots, P$, are obtained as the first N entries of the real (or imaginary) part of the vectors $\tilde{X}_p$ in Eq. (10). The following result shows that the real and imaginary parts of $\tilde{X}_p$ have the covariance structure determined by the circulant embeddings of the original series. See Appendix A.2 for a proof.

Lemma 2.2. *For any $1 \le p, p' \le P$,*

$$E\Re(\tilde{X}_p)\Re(\tilde{X}_{p'})^T = E\Im(\tilde{X}_p)\Im(\tilde{X}_{p'})^T = \tilde{\Sigma}_{p,p'}.$$

Moreover, $\Re(\tilde{X}_p)$ and $\Im(\tilde{X}_{p'})$ are independent.

Lemma 2.2 ensures the correct auto- and cross-covariance of X_p's, since $\tilde{\Sigma}_{p,p'}$ contains $\Sigma_{p,p'}$ in the upper-left corner.

2.3. Synthesis Algorithm

The synthesis algorithm derived in Section 2.2 above can be summarized and organized as follows.

Algorithm (Synthesis of Gaussian Stationary Multivariate Time Series)

I **Embedding Phase:**

1) **Covariance Embedding.** For $1 \le p \le p' \le P$, form the sequences $\tilde{r}_{p,p'}[n]$, $n = 0, \ldots, 2M-1$, based on (6).

2) **Transfer to Spectral Domain.** Apply FFT to $\tilde{r}_{p,p'}$ to calculate $\lambda_{p,p'}[m]$, $m = 0, \ldots, 2M-1$, in Eq. (9). Calculate the diagonal elements $d_{p,p'}[m]$, $m = 0, \ldots, 2M-1$, in Eq. (12).

3) **Cholesky Factorization.** Form the $P \times P$ matrices C_m, $m = 0, \ldots, 2M-1$, in (ii'). Perform their Cholesky factorization in Eq. (14) and store the matrices A_m, $m = 0, \ldots, 2M-1$.

II **Synthesis Phase:**

4) **Noise Generation.** For each $m = 0, \ldots, 2M-1$, generate independently $W_m = W_m^0 + \imath W_m^1$, with two independent vectors W_m^0 and W_m^1 being $\mathcal{N}(0, I_P/2)$. Compute $\underline{Z}_m = (Z_1[m] \cdots Z_P[m])^T = A_m W_m$ using Eq. (15) (with A_m from step 3) above).

5) **Spectral Synthesis.** For $p = 1, \ldots, P$, apply FFT to the sequence $\lambda_p^{1/2}[m] Z_p[m]$, $m = 0, \ldots, 2M-1$, multiplied by $1/\sqrt{2M}$, to calculate the series $\tilde{X}_p$ in Eq. (10). Set the desired vectors X_p, $p = 1, \ldots, P$, as the first N entries of the real part of $\tilde{X}_p$'s.

The algorithm is clearly divided into two phases: the embedding phase that needs to be performed only once for a fixed N and P, regardless of the number of independent realizations that are produced; and the synthesis phase which can be repeated for the synthesis of multiple realizations of the series in Eq. (1). This explains why steps 3) and 4) are split, when they could obviously be performed in the same loop on m.

Note that in step 5) we could also have set the vector X_p as the first N entries of the imaginary part of $\tilde{X}_p$, thanks to Lemma 2.2. Hence, to synthesize B independent realizations, the synthesis phase needs to be run only $B/2$ times.

It is important to emphasize that the last step 5) in the algorithm involves the discrete Fourier transform (DFT) but not the inverse discrete Fourier transform (IDFT). (Using IDFT is quite common in the univariate circulant embedding method.) Using the IDFT instead would in general not give the correct covariance structure. To see this, let $\tilde{Y}_p = F \Lambda_p^{1/2} Z_p$, where multiplying by the matrix F corresponds to using IDFT in the last step. Then, $E\tilde{Y}_p \tilde{Y}_p^T = 0$ and

$$E\tilde{Y}_p \tilde{Y}_{p'}^* = F \Lambda_{p,p'} F^*. \tag{17}$$

Using the same arguments as in Appendix A.2, $\tilde{Y}_p$ can be used instead of $\tilde{X}_p$ only if this last expression is equal to $2\tilde{\Sigma}_{p,p'}$, $1 \le p, p' \le P$. Since the Fourier matrices are symmetric, we have $F^* = \overline{F}$, where $\overline{F}$ denotes the complex conjugate of F. Taking the complex conjugate of both sides of $2\tilde{\Sigma}_{p,p'} = F^* \Lambda_{p,p'} F = \overline{F} \Lambda_{p,p'} F$ gives

$$2\tilde{\Sigma}_{p,p'} = F \overline{\Lambda}_{p,p'} \overline{F} = F \Lambda_{p,p'}^* F^*,$$

since $\Lambda_{p,p'}$ is diagonal. Therefore, the right-hand side of Eq. (17) is equal to $2\tilde{\Sigma}_{p,p'}$ if and only if $\Lambda_{p,p'} = \Lambda_{p,p'}^*$. But since $\Lambda_{p,p'}$ is diagonal, this condition is equivalent to $\Lambda_{p,p'}$ being real-valued, or in other words, $\tilde{\Sigma}_{p,p'}$ being symmetric. Using the IDFT instead of

the DFT in the last step of the algorithm will only give the correct covariance structure if the series is *time-reversible* (see definition in Section 2.4).

2.4. Time-Reversible Series

The synthesis procedure above can be further tuned to the special case of time-reversible series (e.g., [18, 19, 20]).

Definition 1: A time series $\{X[n] : n \in \mathbb{Z}\}$ is said to be *time-reversible* if the series $\{X[n] : n \in \mathbb{Z}\}$ and $\{X[-n] : n \in \mathbb{Z}\}$ have the same finite-dimensional distributions.

In the Gaussian case, time-reversibility is equivalent to

$$R[n] = R[-n] = R[n]^T, \quad n \geq 1, \tag{18}$$

where the last equality follows from stationarity. Thus, if the series is time-reversible, then the matrices $\Sigma_{p,p'}$ are symmetric and $r_{p,p'}[n] = r_{p,p'}[-n]$. Conversely, if $\Sigma_{p,p'}$ are symmetric for all N, p, p', then the series is time-reversible.

For the embedding of $\Sigma_{p,p'}$ in the time-reversible case (step 1) in the algorithm), we can use circulant embedding with $M = N - 1$, instead of $M = N$ for the general case. Hence, the sequence $\tilde{r}$ (in Eq. (4)) becomes

$$(r_{p,p'}[0] \cdots r_{p,p'}[N-1] r_{p,p'}[N-2] \cdots r_{p,p'}[1]). \tag{19}$$

The embedding size $2M = 2N - 2$ is a power of 2 with $N = 2^K + 1$.

There is a priori little reason why real data should be time-reversible. However, a number of potentially interesting models, such as those described in Section 3.3 below, often possess this property by definition.

3. Discussion and Comments

3.1. Computational Complexity

We study the computational costs and memory requirements of Algorithm for the Embedding and Synthesis phases separately. Recall that complexity is envisaged for $N \gg P$ and that $M \simeq N$.

Embedding Phase. The cost of finding the diagonal matrices in step 2) is $O(P^2 M \log M)$; i.e., one application of FFT per matrix, applied $O(P^2)$ times. Since the computational complexity of a single Cholesky factorization in step 3) is $O(P^3)$, the total computational complexity is $O(P^3 M)$ for finding all the lower-triangular matrices A_m. With the assumption $M \gg P$, the total cost for the Embedding Phase is therefore $O(P^2 M \log M)$.

Synthesis Phase. The M matrix multiplications in step 4) consist of $O(P^2 M)$ complex-number multiplications. Step 5) requires one multiplication of a diagonal matrix and one application of the FFT for each random vector, $\tilde{X}_p$, i.e., $O(PM \log M)$ operations.

Global Costs. In total, $O(P^2 M \log M)$ operations are needed for generating the $P \times N$ entries of the series $\{X[n], n = 0, \ldots, N-1\}$. Thus, the complexity is roughly P^2 times that needed for the synthesis of a *single* univariate Gaussian series, which, keeping in mind the assumption $M \gg P$, is essentially of the same order. For generating multiple realizations of the same series, the $P \times M$ complex-numbers $\lambda_p[m]$ need to be stored, along with the M matrices A_m, which is equivalent to storing a complex-valued matrix of size $O(P^2 M)$.

Comparing to using Cholesky factorization of the full covariance structure of the series, $O(P^3 N^3)$ operations would be required for the factorization and $O(P^2 N^2)$ for the synthesis. For multiple realizations, a matrix of size $O(P^2 N^2)$ would need to be stored.

3.2. Sufficient Conditions for ND Assumption

We examine here conditions under which ND assumption from Section 2.2.1 holds.

On the practical side, ND assumption is checked by verifying numerically whether the matrices Λ_p and C_m are nonnegative definite. For the matrices Λ_p, this requires $\lambda_p[m]$'s to be nonnegative and, for the matrices C_m, their nonnegative definiteness is checked when performing the Cholesky factorization. When ND assumption fails, one general recommendation is to repeat the algorithm for larger M (or N). This is justified theoretically below. Alternatively, an approximate solution could be used by setting negative $\lambda_p[m]$'s to zero and by approximating C_m's, which are not nonnegative definite, by nonnegative definite matrices. The latter could be achieved for example, by setting to zero the negative eigenvalues in the Schur decomposition of the matrices C_m. These approximating C_m's would then be used in the Cholesky factorization step in Eq. (14) and would also allow one to numerically calculate the actual covariance of the series being synthesized. The resulting covariance structure can then be easily compared to the targeted one.

On the theoretical side, we provide here sufficient conditions for ND assumption to hold, in terms of the covariance structure of a time series model. Recall from Theorem 2.1 that this is equivalent to matrices $\widehat{R}_m = (\lambda_{p,p'}[m])_{1 \le p,p' \le P}$ being nonnegative definite for all $m = 0, \ldots, 2M-1$. In the univariate case $P = 1$, basic sufficient conditions were proposed in [13], and then further investigated in [21, 22, 23].

The first result is a multivariate counterpart of Theorem 1 and Corollary 2 in [13]. In a slight contrast with [13], we work with discrete time series directly and also reformulate main assumptions. (See also Proposition in [17].)

Theorem 3.1. *Suppose that the covariance matrices $R[n]$, $n \in \mathbb{Z}$, defined in Eq. (3), are absolutely summable and that the spectral density matrix $\widehat{R}(\omega) = \sum_{j \in \mathbb{Z}} R[j] e^{-\imath j \omega}$ is positive definite for all $\omega \in [0, 2\pi]$. Then, for large enough M, the matrices $\widehat{R}_m$ are nonnegative definite for all $m = 0, \ldots, 2M-1$, and hence ND assumption holds.*

The second result is a natural multivariate extension of Theorem 2 in [13]. It concerns only the time-reversible case (see Eq. (18)). Set $\Delta R[n] = R[n] - R[n+1]$, and $\Delta^2 R[n] = \Delta R[n] - \Delta R[n+1] = R[n] - 2R[n+1] + R[n+2]$.

Theorem 3.2. *Consider the time-reversible case and the time-reversible embedding based on* (19). *If* $R[n]$, $\Delta R[n]$ *and* $\Delta^2 R[n]$ *are nonnegative definite, then the matrices* $\widehat{R}_m$ *are nonnegative definite for all* $m = 0, \ldots, 2M-1$, *and hence ND assumption holds.*

The proofs of Theorems 3.1 and 3.2 are postponed to Appendix A.3 and Appendix A.4 respectively. Note that the sufficient conditions in Theorem 3.2 above reduce to those in Theorem 2 of [13] when $P = 1$, and that they are expectedly more difficult to verify when $P > 1$. These conditions will be examined below in several examples of multivariate series.

We should stress here the different messages conveyed by these theorems. On the one hand, Theorem 3.1 states that, under mild assumptions, the algorithm will always work (i.e., ND assumption will hold) as long as M is taken large enough (without precisely stipulating the required value M that should be taken). This justifies that the algorithm should work under quite general assumptions and, if it does not, the common recommendation is to increase M as mentioned above. On the other hand, Theorem 3.2 provides sufficient conditions for the algorithm to work for any finite M. For a number of time series models, these conditions can be verified theoretically and a priori (see Section 3.3 below).

3.3. Examples

Several examples of stationary multivariate series are provided. Conditions for ND assumption above to hold are theoretically and numerically investigated on these examples.

3.3.1. Bivariate Geometrically Decaying Covariances

Consider a bivariate Gaussian stationary time series with the covariance structure given by

$$R[n] = \begin{pmatrix} \varphi_1^n & c\varphi_3^n \\ c\varphi_3^n & \varphi_2^n \end{pmatrix}, \tag{20}$$

where $n \geq 0$, $0 < \varphi_1, \varphi_2 < 1$ and $c, \varphi_3 \in \mathbb{R}$. If $0 \leq c \leq 1$, $|\varphi_3| \leq \min(\varphi_1, \varphi_2)$ and $1 - (1 - \max(\varphi_1, \varphi_2))/\sqrt{c} \leq \varphi_3 \leq \min(\varphi_1, \varphi_2)$, it can be shown easily that $R[n]$, $\Delta R[n]$, and $\Delta^2 R[n]$ are nonnegative definite and hence ND assumption holds by Theorem 3.2.

3.3.2. Multivariate AR(1) Series

Consider the series defined by

$$X[n] = \Phi X[n-1] + \epsilon[n], \ n \in \mathbb{Z},$$

where Φ is a $P \times P$ matrix whose eigenvalues are smaller than 1 in absolute value, and $\epsilon[n]$ are i.i.d. Gaussian vectors of length P with mean zero and covariance $E\epsilon[n]\epsilon[n]^T = \Sigma_\epsilon$ (e.g. [10]). If Σ_ϵ and Φ commute, and Φ is symmetric, it can be seen that the series $X[n]$ is time-reversible with

$$R[n] = (\Sigma_\epsilon \sum_{k=0}^{\infty} \Phi^{2k})\Phi^n = R[0]\Phi^n = R[n-1]\Phi, \tag{21}$$

where $n \geq 1$. Note that $\Delta R[n] = R[n](I - \Phi)$ and $\Delta^2 R[n] = R[n](I - \Phi)^2$. Then, $R[n]$, $\Delta R[n]$, and $\Delta^2 R[n]$ are nonnegative definite if Φ and $I - \Phi$ are nonnegative definite.

3.3.3. Multivariate FARIMA$(0, D, 0)$ Series

Consider a $P \times P$ matrix D whose (possibly complex) eigenvalues d_k satisfy $|d_k| < 1/2$, and suppose that D is diagonalizable as $D = W\Lambda W^{-1}$ with diagonal $\Lambda = diag(d_1, \ldots, d_P)$. Define a multivariate FARIMA$(0, D, 0)$ series as

$$(I - B)^D X[n] = \epsilon[n]$$

or $X[n] = (I-B)^{-D}\epsilon[n] = \sum_{k=0}^{\infty} C_D[k]\epsilon[n-k]$, where B is the usual backshift operator, $C_D[k]$ are the matrix coefficients in the Taylor expansion $(1-z)^{-D} = \sum_{k=0}^{\infty} C_D[k]z^k$ and the innovations $\epsilon[n]$ are i.i.d. Gaussian vectors of length P with mean zero and covariance $E\epsilon[n]\epsilon[n]^T = \Sigma_\epsilon$. The series above is well-defined by writing it as $X[n] = W(I - B)^{-\Lambda}W^{-1}\epsilon[n] =: W(I-B)^{-\Lambda}\eta[n] =: WX_\Lambda[n]$, where $X_\Lambda[n]$ is a FARIMA$(0, \Lambda, 0)$ series with the innovations $\eta[n]$ having the covariance $\Sigma_\eta = (\Sigma_{\eta,p,p'})_{1\le p,p'\le P} = W^{-1}\Sigma_\epsilon(W^{-1})^*$. The FARIMA$(0, \Lambda, 0)$ series $X_\Lambda[n]$ with diagonal Λ are considered in, for example, [16, 24, 25]. The covariance structure of the series $X[n]$ is given by

$$R[n] = WR_\Lambda[n]W^*, \tag{22}$$

where $n \ge 0$ and the entries of $R_\Lambda[n]$ are

$$R_\Lambda[n]_{p,p'} = \Sigma_{\eta,p,p'}\frac{(-1)^n\Gamma(1 - d_p - \overline{d}_{p'})}{\Gamma(1 + n - \overline{d}_{p'})\Gamma(1 - n - d_p)}$$

by using the formula in Proposition 2 of [16] (and correcting the typo where the term $(1 - d_j - d_k)$ should be removed from the denominator in that formula).

In view of (22) and for general Σ_ϵ, FARIMA$(0, D, 0)$ series is time-reversible only when the eigenvalues $d_k = d$ are all the same. In this case, $D = dI$ and $R[n] = r[n]\Sigma_\epsilon$, where $r[n]$ is the auto-covariance function of a univariate FARIMA$(0, d, 0)$ series. Then, $R[n]$, $\Delta R[n]$, and $\Delta^2 R[n]$ are always nonnegative definite by using the univariate results for $r[n]$ as in, for example, [23].

3.3.4. Multivariate Fractional Gaussian Noise (FGN)

Define multivariate FGN as the series

$$X[n] = B_H(n+1) - B_H(n),\ n \in \mathbb{Z},$$

where $B_H(t),\ t \in \mathbb{R}$, is the so-called P-variate operator fractional Brownian motion (OFBM). For a matrix H, the process B_H is called OFBM if it is zero-mean, Gaussian, has stationary increments, and is operator self-similar in the sense that the processes $B_H(ct)$ and $c^H B_H(t)$ have the same finite-dimensional distributions for any $c > 0$, where $c^H = e^{(\ln c)H} = \sum_{k=0}^{\infty}(\ln c)^k H^k/k!$. OFBMs generalize univariate FBMs to the multivariate case, and have been studied extensively in [26]. We shall focus here only on time-reversible OFBMs B_H (and such FGNs) for and only for which the covariance $\Sigma_H(s,t) = EB_H(s)B_H(t)^*,\ s,t \in \mathbb{R}$, can be written as

$$\Sigma_H(s,t) = \frac{1}{2}(|t|^H\Sigma|t|^{H^*} + |s|^H\Sigma|s|^{H^*} - |t-s|^H\Sigma|t-s|^{H^*})$$

where $\Sigma = EB_H(1)B_H(1)^*$ (see [26]). For such OFBM, the resulting FGN series $X[n]$ has the covariance structure

$$\begin{aligned} R[n] &= \frac{1}{2}(|n+1|^H\Sigma|n+1|^{H^*} + |n-1|^H\Sigma|n-1|^{H^*} \\ &\quad -2|n|^H\Sigma|n|^{H^*}). \end{aligned} \tag{23}$$

The matrix Σ in (23) is positive definite but not every positive definite matrix Σ would lead to a valid covariance structure $R[n]$. One way to choose a valid Σ is to use a parametrization through spectral representations of OFBM derived in [26]. Suppose the matrix H has its eigenvalues h_k such that $0 < h_k < 1$. Then, as shown in [26], OFBM B_H has an integral representation

$$B_H(t) \stackrel{d}{=} \int_{\mathbb{R}} \frac{e^{itx}-1}{ix}(x_+^{-(H-\frac{1}{2}I)}A + x_-^{-(H-\frac{1}{2}I)}\overline{A})\widetilde{B}(dx),$$

where A is a matrix with complex-valued entries, $x_+ = \max\{x, 0\}$, $x_- = \max\{-x, 0\}$, and $\widetilde{B}(dx)$ is a suitable multivariate complex-valued Gaussian measure satisfying $E\widetilde{B}(dx)\widetilde{B}(dx)^* = dx$. OFBM is time-reversible if and only if AA^* is a real-valued matrix (see [26]). Moreover, in this case, the matrix $\Sigma = EB_H(1)B_H(1)^*$ is given by

$$\Sigma = \int_{\mathbb{R}} \frac{|e^{ix}-1|^2}{|x|^2}|x|^{-(H-\frac{1}{2}I)}AA^*|x|^{-(H^*-\frac{1}{2}I)}dx \tag{24}$$

and hence is parameterized by H, A.

The matrix Σ in (24) can be computed explicitly in the case when H is diagonalizable. Thus suppose that $H = W\Lambda W^{-1}$ with diagonal $\Lambda = diag(h_1, \ldots, h_P)$. Then,

$$\begin{aligned} \Sigma &= 8W\int_0^\infty \frac{\sin^2\frac{x}{2}}{x^2}x^{-(\Lambda-\frac{1}{2}I)}Cx^{-(\Lambda^*-\frac{1}{2}I)}dxW^* \\ &=: 8W\Sigma_0W^*, \end{aligned} \tag{25}$$

where $C = (c_{p,p'})_{1\le p,p'\le P} = W^{-1}AA^*(W^{-1})^*$. The entries $\Sigma_{0,p,p'}$ of Σ_0 can be written as (using Fomula 3.823 in [27])

$$\begin{aligned} \Sigma_{0,p,p'} &= c_{p,p'}\int_0^\infty \sin^2(\frac{x}{2})x^{-(h_p+\overline{h}_{p'})-1}dx \\ &= -\frac{c_{p,p'}}{2}\Gamma(-h_p-\overline{h}_{p'})\cos\left(\frac{(h_p+\overline{h}_{p'})\pi}{2}\right). \end{aligned} \tag{26}$$

Finally, sufficient conditions for $R[n]$, $\Delta R[n]$, and $\Delta^2 R[n]$ to be nonnegative definite can be studied but are less immediate than in the examples above, and go beyond the scope of this work.

3.3.5. Numerical Experiments

We report here on the performance of our algorithm on the examples of multivariate series above. The basic goal was to check numerically whether and in which cases the algorithm fails, or equivalently ND assumption is not satisfied, when the time series models above are not necessarily time-reversible (and hence not covered by Theorem 3.2). Since the series in Example 1) above is time-reversible by construction and we have defined it only when ND assumption holds, this example is excluded from the discussion below.

Multivariate AR(1) Series. Simulations were done for the bivariate case for the matrix Φ of the form $\Phi = W diag(h) W^{-1}$, $h = (h_1, h_2)$ with $-1 < h_1, h_2 < 1$. For each trial, the four entries of W were independent samples of $\mathcal{N}(0,1)$ variables; h_1 and h_2 were independently and uniformly sampled from the interval $(-1, 1)$. The matrix $\Sigma_\epsilon = (c_{p,p'})_{1 \le p,p' \le 2}$ was generated at random such that $c_{1,1} = 1$ and $c_{2,2}$ was sampled uniformly at random from the interval $(0, 2)$. Then $c_{1,2} (= c_{2,1})$ was sampled uniformly at random from the interval $(-\sqrt{c_{1,1}c_{2,2}}, \sqrt{c_{1,1}c_{2,2}})$. This generation does not force Φ to be symmetric nor Σ_ϵ and Φ to commute.

The algorithm was run for $B = 1000$ trials for each length $N = 2^J$, $J = 10, 11, 12, 13, 14$. The only cases it did not succeed was when the absolute value of either h_1 or h_2 was very close to 1 (greater than 0.99) or if the generated matrix W was very close to being singular. For the larger lengths N, h_1 and h_2 could be taken closer to 1, which is not surprising in light of Theorem 3.1.

Multivariate FARIMA$(0, D, 0)$ *Series.* Simulations were done for the bivariate case and the parameters W, Σ_ϵ and $\Lambda = diag(d_1, d_2)$ generated at random for each trial. The matrices W and Σ_ϵ were generated as the same parameters in the multivariate AR(1) example. The parameters d_1 and d_2 were independently and uniformly sampled from the interval $(-1/2, 1/2)$. The simulations indicated that ND assumption holds (numerically) if the ratio $|\Sigma_{\eta,1,2}|/\sqrt{\Sigma_{\eta,1,1}\Sigma_{\eta,2,2}}$ is not close to one.

Multivariate Fractional Gaussian Noise (FGN). Simulations were done for the bivariate case and the parameters W, A and $\Lambda = diag(h_1, h_2)$ generated at random for each trial: the four entries of W and A were taken to be independent $\mathcal{N}(0,1)$ variables; h_1 and h_2 were independently and uniformly sampled from the interval $(0, 1)$. The simulations indicated that ND assumption holds (numerically) if the ratio $|c_{1,2}|/\sqrt{c_{1,1}c_{2,2}}$ is not close to one (about 0.95 or smaller for $N = 2^{10}$), where $C = (c_{p,p'})_{1 \le p,p' \le 2} = W^{-1} A A^* (W^{-1})^*$.

In conclusion, for the models considered above, the algorithm works, or equivalently ND assumption holds, for most values of parameters of the models; that is, the Cholesky factorization step in the algorithm could be performed, or equivalently, as discussed in the beginning of Section 3.2, the matrices Λ_p and C_m were positive definite.

3.4. Synthesis Given the Spectrum Instead of Covariance

The synthesis procedure proposed in Section 2 is for the situation when the covariance structure of the Gaussian stationary multivariate series is available explicitly. One can often be given instead the spectral density matrix $\widehat{R}(\omega)$, which is related to the covariance by

$$R[n] = \frac{1}{2\pi} \int_0^{2\pi} \widehat{R}(\omega) e^{\imath n \omega} d\omega, \quad n \in \mathbb{Z}, \tag{27}$$

and where the latter integral cannot or is not known to be computed explicitly. In such situation, one option is to synthesize the series approximately by using Chambers' method [16] (see also [28, 29], for the univariate case). In a simplified, univariate case, the method starts by sampling $\widehat{R}(\omega)$ at discrete frequencies $\omega_k = 2\pi k/(2K)$, $k = 0, \ldots, 2K$, and forming a vector

$$\widehat{R}_D = (\widehat{R}(\omega_0), \ldots, \widehat{R}(\omega_K), \widehat{R}(\omega_{K-1}), \ldots, \widehat{R}(\omega_1))^T.$$

Then, a sequence $\tilde{X}$ is generated as FFT of the sequence $K^{-1/2}(\widehat{R}_D[k])^{1/2}Z[k]$, where $Z = Z^0 + \imath Z^1$, with Z^0 and Z^1 being two independent $\mathcal{N}(0, I_{2K})$ random vectors. An approximate realization of the desired series is obtained as the first K entries of $\Re(\tilde{X})$ (or $\Im(\tilde{X})$).

The covariance sequence of this simulated series is

$$r_{approx}[n] = \frac{1}{2K} \sum_{k=0}^{2K-1} \widehat{R}_D[k] e^{\imath \frac{2\pi n k}{2K}}, \quad n = 0, \ldots, K-1,$$

which is a Riemann-sum approximation of the integral on the right-hand side of Eq. (27). As is well-known for such integrals, the oscillatory integrand will cause the sequence $r_{approx}[n]$ to converge very slowly to the exact covariance and, where the approximation worsens the larger n is. Therefore, only a small portion of the generated series can be used, that is, one needs to take $K = C \cdot N$, with C being a big constant, to generate a sample of length N accurately. This is very unfortunate for Monte Carlo simulations where one needs to synthesize multiple realizations of the same series. Also, due to memory constraints, this limits the length of the series that can be produced.

Another problem arises for synthesis of long-range dependent series, which are of significant practical interest (e.g., [30]). Such series are characterized by a power law behavior of the spectrum at the origin of the form $\widehat{R}(\omega) \sim |\omega|^{-\alpha}$, where $0 < \alpha < 1$. The singularity at $\omega = 0$ creates obvious difficulties in sampling of the spectral density at low frequencies and a Riemann-sum approximation becomes even less appropriate than before.

A way to overcome these problems is to compute the integral in Eq. (27) numerically and to apply the algorithm proposed in Section 2. Computation of the integral in Eq. (27) can be performed numerically with high precision using integration schemes such as the trapezoidal rule (see [31]). Singularities of the spectrum can be dealt with using change of variables prior to the numerical integration; in the example above, one could use $\omega = u^{1/(1-\alpha)}$. For repeated Monte Carlo simulations, the covariance sequence only needs to be calculated once.

4. Conclusions and Perspectives

We have provided a simple and fast simulation procedure for generating realizations (of length N) of stationary multivariate Gaussian time series (with P components), whose auto- and cross-covariance sequences are exactly controlled and a priori prescribed. It is based on an extension of the univariate circulant embedding method to the multivariate case. A practical algorithm has been provided with detailed analyses of the roles and relations of all steps. It has been shown to yield realizations possessing exactly the prescribed covariance. A MATLAB software package, implementing the proposed algorithm, together with the covariances for the examples described in Section 3.3 is publicly available at `http://perso.ens-lyon.fr/patrice.abry/software.html`.

This work can be further developed to address related issues, of major practical relevance in a number of real world applications. For example, it can be of interest to consider the situation where a new component needs to be added to P components of length N that have already been synthesized. Also, in some situations where P components, or

actual real data, are available, it can be useful to be able to create numerically (one or more) additional components, that not only possess the same auto- and cross-covariance but also are actually correlated with the available data. Both of these questions can receive a solution by turning the proposed algorithm into a sequential version that adds iteratively components correlated to the existing ones. Such issues are notably of interest in large sensor network deployment and/or monitoring: the number of sensors is indeed likely to vary (increase or decrease) either because of failures or of monitoring optimization policies. Also, the benefits and relevance of a further deployment of sensors can be evaluated a priori, thanks to the analysis of artificially added extra components obtained by numerical synthesis [2, 4]. Furthermore, this synthesis algorithm can be incorporated into a more general procedure aiming at synthesizing second-order stationary multivariate non-Gaussian series, for which both the auto- and cross-covariance functions and the marginal distributions are a priori and jointly prescribed (within the class of Gaussian related processes). This is also of major practical importance in a number of applications, such as Internet traffic monitoring [9, 32] for instance. These different issues are under current investigation.

Appendix A.

Appendix A.1. Proof of Theorem 2.1

We will use the following notation. For a collection $\{A_{p,p'}, 1 \leq p, p' \leq P\}$, of matrices of equal size $2M \times 2M$, denote the corresponding block matrix by $[A_{p,p'}]$. Write $diag_P(B_p)$ for the $P \times P$ block matrix with $2M \times 2M$ matrices $B_1, \ldots, B_P$, on the diagonal and zero elsewhere. Write $diag_P(B)$ in the case where $B = B_1 = \cdots = B_P$.

Using this notation, ND assumption is equivalent to the block matrix $[\tilde{\Sigma}_{p,p'}]$ being nonnegative definite. The decomposition

$$[\Lambda_{p,p'}] = diag_p(F^*)[\tilde{\Sigma}_{p,p'}]diag_p(F)$$

shows that $[\Lambda_{p,p'}]$ is nonnegative definite if and only if $[\tilde{\Sigma}_{p,p'}]$ is nonnegative definite. Take a row vector $\beta = (\beta_1, \ldots, \beta_P)$, where $\beta_p = (\beta_p[0], \ldots, \beta_p[2M-1])$ is a row vector with complex-valued entries $\beta_p[m]$. Then, $[\Lambda_{p,p'}]$ is nonnegative definite if and only if the following sum is nonnegative for all vectors β:

$$\begin{aligned} \beta[\Lambda_{p,p'}]\beta^* &= \sum_{p,p'=1}^{P} \beta_p \Lambda_{p,p'} \beta_{p'}^* \\ &= \sum_{p,p'=1}^{P} \sum_{m=0}^{2M-1} \beta_p[m] \beta_{p'}^*[m] \lambda_{p,p'}[m], \end{aligned}$$

since the matrices $\Lambda_{p,p'}$ are diagonal. Writing $\alpha_p[m] = 2^{-1/2}\lambda_p[m]^{1/2}\beta_p[m]$, we get

$$\beta[\Lambda_{p,p'}]\beta^* = \sum_{m=0}^{2M-1} \sum_{p,p'=1}^{P} \alpha_p[m] \alpha_{p'}^*[m] C_m[p,p'].$$

(If $\lambda_p[m] = 0$, then $C_m[p,p'] = 0$ by construction (13) and hence the choice of $\alpha_p[m]$ does not affect the latter sum.) This shows that $[\Lambda_{p,p'}]$ is nonnegative definite if and only if all the matrices C_m are nonnegative definite, which finishes the proof of the first statement of the theorem. The second statement is proved similarly.

Appendix A.2. Proof of Lemma 2.2

Pick any pair (p, p') such that $1 \le p, p' \le P$. It follows from Eq. (16) that the vectors Z_p and $Z_{p'}$ satisfy $EZ_pZ_{p'}^T = 0$. Therefore,

$$E\tilde{X}_p\tilde{X}_{p'}^T = F^*\Lambda_p^{1/2}E[Z_pZ_{p'}^T]\Lambda_{p'}^{1/2}(F^*)^T = 0.$$

Decomposing $\tilde{X}_p$ and $\tilde{X}_{p'}$ into real and imaginary parts gives

$$E[\Re(\tilde{X}_p)\Re(\tilde{X}_{p'})^T - \Im(\tilde{X}_p)\Im(\tilde{X}_{p'})^T] = 0,$$
$$E[\Re(\tilde{X}_p)\Im(\tilde{X}_{p'})^T + \Im(\tilde{X}_p)\Re(\tilde{X}_{p'})^T] = 0.$$

By construction, we have $E\tilde{X}_p\tilde{X}_{p'}^* = 2\tilde{\Sigma}_{p,p'}$, which is real-valued, so that

$$E[\Re(\tilde{X}_p)\Re(\tilde{X}_{p'})^T + \Im(\tilde{X}_p)\Im(\tilde{X}_{p'})^T] = 2\tilde{\Sigma}_{p,p'},$$
$$E[\Re(\tilde{X}_p)\Im(\tilde{X}_{p'})^T - \Im(\tilde{X}_p)\Re(\tilde{X}_{p'})^T] = 0.$$

Therefore,

$$E\Re(\tilde{X}_p)\Re(\tilde{X}_{p'})^T = E\Im(\tilde{X}_p)\Im(\tilde{X}_{p'})^T = \tilde{\Sigma}_{p,p'}$$

and

$$E\Re(\tilde{X}_p)\Im(\tilde{X}_{p'})^T = E\Im(\tilde{X}_p)\Re(\tilde{X}_{p'})^T = 0.$$

Appendix A.3. Proof of Theorem 3.1

The proof is similar to those of Theorem 1 and Corollary 2 in [13]. We consider a slightly more involved case of non-time-reversible series, and the non-time-reversible embedding based on (6). Note first that, in this case, when $p \le p'$,

$$\lambda_{p,p'}[m] \quad = \quad \sum_{j=-(N-1)}^{N-1} r_{p,p'}[j]e^{-\imath\frac{2\pi jm}{2N}} + r_{p,p'}[N](-1)^m$$

and, as we set, $\lambda_{p,p'}[m] = \lambda_{p',p}[m]^*$ for $p > p'$. Then,

$$\widehat{R}_m = \sum_{j=-(N-1)}^{N-1} R[j]e^{-\imath\frac{2\pi jm}{2N}} + R_0[N](-1)^m,$$

where

$$R_0[N] = (r_{0,p,p'}[N])_{1\le p,p'\le P} \tag{A.1}$$

with

$$r_{0,p,p'}[N] = \begin{cases} r_{p,p'}[N], & \text{if } p \le p', \\ r_{p,p'}[-N], & \text{if } p' < p. \end{cases}$$

By the assumptions of the theorem, the series $\sum_{j\in\mathbb{Z}} R[j]e^{-\imath j\omega}$ converges uniformly and the limit is a positive definite matrix for all $\omega \in [0, 2\pi]$. Then, for large enough N, the Hermitian symmetric matrices

$$\sum_{j=-(N-1)}^{N-1} R[j]e^{-\imath j\omega} + R_0[N]\cos(Nw),$$

are also positive definite for all $\omega \in [0, 2\pi]$. Taking $\omega = \pi m/N$ now yields the desired result in view of the expression for $\widehat{R}_m$ above.

Appendix A.4. Proof of Theorem 3.2

In the time-reversible case, we can write

$$\widehat{R}_m = R[0] - R[N-1](-1)^m + 2\sum_{j=1}^{N-1} R[j]\cos\left(\frac{2\pi mj}{2N-2}\right)$$

and the proof follows exactly as in that of Theorem 2 of [13] by using the assumption that $R[n]$, $\Delta R[n]$ and $\Delta^2 R[n]$ are nonnegative definite.

Appendix B. Comparison to Chan and Wood's Algorithm

Since both our and Chan and Wood's [17] (CW, in short) algorithms are based on a circulant matrix embedding, it is natural to expect that the two algorithms are closely related. In fact, we will show here that the two algorithms are essentially equivalent for time-reversible series, and indicate how they are different for non-time-reversible series.

It is more instructive to show first the equivalence of our and CW algorithms for time-reversible series. Using our notation, CW algorithm considers a block circulant matrix of the form,

$$\mathrm{circ}\{R[0],\ldots,R[N-1],R[N-2],\ldots,R[1]\} =: \mathrm{circ}\{\tilde{R}[0],\ldots,\tilde{R}[2M-1]\} \tag{B.1}$$

and, for $m = 0,\ldots,2M-1$, defines $P\times P$ matrices

$$H[m] = (H_{p,p'}[m])_{1\le p,p'\le P} = \sum_{j=0}^{2M-1} \tilde{R}[j]e^{-\imath\frac{2\pi jm}{2M}}. \tag{B.2}$$

Observe that, as in (9),

$$H_{p,p'}[m] = \lambda_{p,p'}[m]$$

and that

$$H[m] = 2^{-1}G_m^{1/2}C_m(G_m^{1/2})^*, \tag{B.3}$$

where C_m are defined in Section 2.2.4 and $G_m = \mathrm{diag}\{\lambda_1[m],\ldots,\lambda_P[m]\}$. CW algorithm proceeds by factorizing the Hermitian symmetric matrices $H[m]$ as[1]

$$H[m] = B[m]B[m]^*. \tag{B.4}$$

In view of (B.3), this is equivalent to the factorization $C_m = A_mA_m^*$ in (14) with

$$A_m = 2^{1/2}G_m^{-1/2}B[m] \quad \text{or} \quad B[m] = 2^{-1/2}G_m^{1/2}A_m. \tag{B.5}$$

For each m, CW algorithm then sets

$$Y[m] = (Y_1[m],\ldots,Y_P[m])^T = B[m](V^{(1)} + \imath V^{(2)}),$$

[1]We slightly rewrite CW algorithm by making it more suitable for comparison.

where $V^{(1)}$ and $V^{(2)}$ are two independent $\mathcal{N}(0, I_P)$ vectors. In view of (B.5), this is equivalent to setting

$$Y[m] = G_m^{1/2} \underline{Z}_m, \tag{B.6}$$

where $\underline{Z}_m$ are defined in (15). Letting $Y_p = (Y_p[0], \ldots, Y_p[2M-1])^T$, Eq. (B.6) can be rewritten as

$$Y_p = \Lambda_p^{1/2} Z_p, \tag{B.7}$$

where the notation Λ_p and Z_p is used in Section 2.2. Finally, CW algorithm considers

$$\tilde{X}_p = F^* Y_p \tag{B.8}$$

and takes the first N entries of either $\Re(\tilde{X}_p)$ or $\Im(\tilde{X}_p)$ for the desired univariate series X_p. Note that, in view of (B.7), (B.8) is precisely Eq. (10) used in our algorithm. This shows that our and CW algorithms are indeed equivalent for time-reversible series.

We now turn to the case of non-time-reversible series, which is the general case considered in this work. Our and CW algorithms are different in this case. The key difference lies in the types of embeddings used. CW algorithm is based on a block circulant matrix of the form

$$\text{circ}\{R[0], \ldots, R[N-1], R[-(N-1)], \ldots, R[-1]\} =: \text{circ}\{\tilde{R}[0], \ldots, \tilde{R}[K-1]\},$$

where $K = 2N-1$ is the associated *embedding size* (see Eq. (B.1)). As in Eq. (B.2), the matrices $H[m]$ are obtained from FFT of size K. Note that the embedding size K here is necessarily odd. CW algorithm suggests using $K = 3^n$ for a suitable integer n. In CW algorithm, the embedding $\text{circ}\{R[0], \ldots, R[N-1], R[N], R[-(N-1)], \ldots, R[-1]\}$ of even size $2N$ cannot be considered in the non-time-reversible case because the matrices $H[m]$ would not be Hermitian symmetric (and hence not factorizable as in (B.4)).

In contrast, in our algorithm, we are able to take embedding of even size $K = 2^n$. How is that possible? The answer lies in our choice of circulant matrices $\tilde{\Sigma}_{p,p'}$. When $p' < p$, we set $\tilde{\Sigma}_{p,p'} = \tilde{\Sigma}_{p',p}^T$, instead of using the same construction as in the case $p \leq p'$ (Section 2.2.1). In fact, one could show that this choice "corresponds" to taking the block circulant matrix in CW algorithm as

$$\text{circ}\{R[0], \ldots, R[N-1], R_0[N], R[-(N-1)], \ldots, R[-1]\}, \tag{B.9}$$

where the matrix $R_0[N]$ is the same as defined by (A.1) in the proof of Theorem 3.1 in Appendix A.3, and is neither $R[N]$ nor $R[-N]$.

In this work, we could alternatively have started with CW formulation of the algorithm and have modified it according to (B.9). We decided to focus on our approach and postpone the discussion on connection to CW algorithm for several reasons. Being based on the univariate circulant matrix embedding, our approach is more natural, revealing, and clarifies CW algorithm. The embedding (B.9) also appears less ad hoc in our approach. In particular, note again that it would not be possible to define this embedding just by using multivariate auto-covariance matrices $R[n]$. In addition, our contributions go beyond CW in the following: we provide sufficient conditions for our algorithm to work and check them on several models, we test our algorithm on several examples, and the code behind our algorithm is publicly available.

References

[1] A. Wood, G. Chan, Simulation of stationary Gaussian processes in $[0,1]^d$, Journal of Computational and Graphical Statistics 3 (4) (1994) 409–432.

[2] G. Barrenetxea, F. Ingelrest, G. Schaefer, M. Vetterli, The hitchhiker's guide to successful wireless sensor network deployments, in: Proc. of the ACM SenSys Conference, 2008.

[3] S. Matsuura, H. Ishizuka, H. Ochiai, S. Doi, S. Ishida, M. Nakayama, H. Esaki, H. Sunahara., Live E! project: establishment of infrastructure sharing environmental information, in: Applications and the Internet Workshops, 2007. SAINT Workshops 2007. International Symposium on, Jan. 2007, pp. 67–67.

[4] Z.-Q. Luo, M. Gastpar, J. Liu, A. Swami, Distributed signal processing in sensor networks, IEEE Signal Proc. Mag. 23 (4).

[5] P. Ciuciu, P. Abry, C. Rabrait, H. Wendt, Log wavelet leaders cumulant based multifractal analysis of EVI fMRI time series: evidence of scaling in ongoing and evoked brain activity, IEEE J. of Selected Topics in Signal Proces. 2 (6) (2008) 929–943.

[6] S. Achard, D. Bassett, A. Meyer-Lindenberg, E. Bullmore, Fractal connectivity of long-memory networks, Phys. Rev. E 77 (3) (2008) 036104.

[7] J. Goldberger, Sympathovagal balance: how should we measure it ?, Am. J. Physiol., 45 (1999) 1273–1280.

[8] B. Krishnamurthy, S. Sen, Y. Zhang, Y. Chen, Sketch-based change detection: Methods, evaluation, and applications, in: IMC'03, 2003, pp. 234–247.

[9] P. Abry, R. Baraniuk, P. Flandrin, R. Riedi, D. Veitch, Multiscale nature of network traffic, IEEE Signal Proc. Mag. 19 (3) (2002) 28–46.

[10] H. Lütkepohl, New introduction to multiple time series analysis, Springer-Verlag, Berlin, 2005.

[11] R. Davies, D. Harte, Tests for Hurst effect, Biometrika 74 (4) (1987) 95–101.

[12] A. Dembo, C. Mallows, L. Shepp, Embedding nonnegative definite Toeplitz matrices in nonnegative definite circulant matrices, with applications to covariance estimation, IEEE Trans. Inform. Theory 35 (1989) 1206–1212.

[13] C. Dietrich, G. Newsam, Fast and exact simulation of stationary Gaussian processes through circulant embedding of the covariance matrix, SIAM J. Sci. Comput. 18 (4) (1997) 1088–1107.

[14] D. Percival, Exact simulation of complex-valued Gaussian stationary processes via circulant embedding, Signal Process. 86 (7) (2006) 1470–1476.

[15] J. Bardet, G. Lang, G. Oppenheim, A. Philippe, M. Taqqu, Long-range dependence: theory and applications, Birkhäuser, 2003, Ch. "Generators of long-range dependent processes: A survey", pp. 579–623.

[16] M. Chambers, The simulation of random vector time series with given spectrum, Mathl. Comput. Modelling 22 (2) (1995) 1–6.

[17] G. Chan, A. Wood, Simulation of stationary Gaussian vector fields, Statistics and Computing 9 (4) (1999) 265–268.

[18] Q. Cheng, On time-reversibility of linear processes, Biometrika 86 (2) (1999) 483–486.

[19] M. Rosenblatt, Gaussian and Non-Gaussian Linear Time Series and Random Fields, Springer-Verlag, Berlin, 2000.

[20] H. Tong, Z. Zhang, On time-reversibility of multivariate linear processes, Statistica Sinica 15 (2005) 495–504.

[21] J. Chilés, P. Delfiner, Geostatistics: modeling spatial uncertainty, Wiley, New York, 1999.

[22] T. Gneiting, Power-law correlations, related models for long-range dependence and their simulation, Journal of Applied Probability 37 (4) (2000) 1104–1109.

[23] P. F. Craigmile, Simulating a class of stationary Gaussian processes using the Davies-Harte algorithm, with application to long memory processes, Journal of Time Series Analysis 24 (5) (2003) 505–511.

[24] C.-F. Chung, Sample means, sample autocovariances, and linear regression of stationary multivariate long memory processes, Econometric Theory 18 (2002) 51–78.

[25] K. Shimotsu, Gaussian semiparametric estimation of multivariate fractionally integrated processes, Journal of Econometrics 137 (2007) 277–310.

[26] G. Didier, V. Pipiras, Integral representations and properties of operator fractional Brownian motions, preprint. To appear in Bernoulli 2010.

[27] I. S. Gradshteyn, I. M. Ryzhik, Table of integrals, series, and products, sixth Edition, Academic Press Inc., 2000.

[28] R. Thompson, Generation of stochastic processes with given spectrum, Util. Math. 3 (1973) 127–137.
[29] D. Percival, Simulating Gaussian random processes with specified spectra, Computing Science and Statistics 4 (1992) 534–538.
[30] P. Doukhan, G. Oppenheim, M. S. Taqqu, Theory and applications of long-range dependence, Birkhäuser, Boston, 2003.
[31] W. Press, S. Teukolsky, W. Vetterling, B. Flannery, Numerical recipes in C. The art of scientific computing, 2nd Edition, Cambridge University Press, 1992.
[32] H. Wendt, A. Scherrer, P. Abry, S. Achard, Testing fractal connectivity in multivariate long memory processes, in: Proc. IEEE Int. Conf. Acoust., Speech, and Signal Proc. (ICASSP), Taipei, Taiwan, 2009.