A note on spatio-temporal random fields on meshed surfaces defined from advection-diffusion SPDEs

Mike Pereira

A NOTE ON SPATIO-TEMPORAL RANDOM FIELDS ON MESHED SURFACES DEFINED FROM ADVECTION-DIFFUSION SPDES

Mike PEREIRA
Department of Geosciences and Geoengineering
Mines Paris – PSL University, Fontainebleau, France.
mike.pereira@minesparis.psl.eu

ABSTRACT

The aim of this work is to propose a statistical model for spatio-temporal data on meshed surfaces based on the SPDE modeling approach. To do so, we consider a class of advection-diffusion SPDEs defined on smooth compact orientable closed Riemannian manifolds of dimension 2, and their discretization using a Galerkin approach. We show how this approach allows to easily propose scalable algorithms for the simulation and prediction of Gaussian random fields that are solutions to the discretized SPDE.

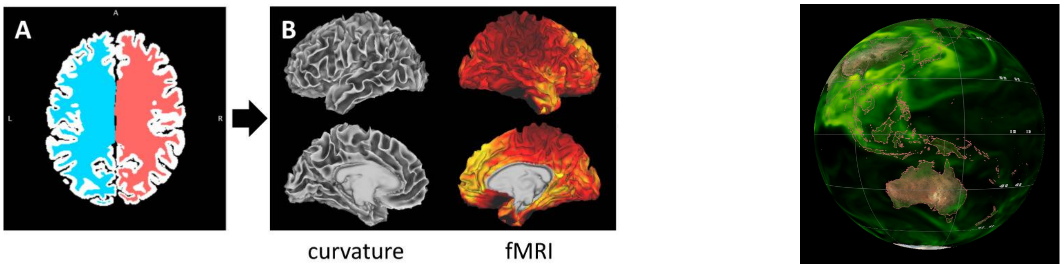
Note: This document contains links to animated figures colored in dark red. All these animations are collected at: https://mike-pereira.github.io/STRF/.

1 Introduction

In Geostatistics, when modeling spatio-temporal data, the observed variable is seen as (a realization of) a space-time Gaussian random field (GRF), so that the mere characterization of its mean and covariance functions suffices to fully describe its statistical properties. In particular, these two functions are chosen to mimic the spatio-temporal variability and structure observed in the data [33]. This probabilistic framework has many advantages. On the one hand, it allows to perform simulations of random fields with the same spatio-temporal structure as the one observed in the data, and predictions of at unobserved locations. On the other hand, uncertainties can be quantified both on the variable behavior at unobserved locations (using so-called conditional simulations) or on the model parameters (through Bayesian approaches) [8].

Of particular interest in this work is the setting where the spatial domain on which the data lie is not Euclidean, but rather represents a meshed surface. For instance, this is the case when dealing fMRI data in neuroimaging applications, in which case the data lie on the cortical surface (i.e.. the surface of the brain): see eg. [20] and the illustration in Figure 1.1A. This is also the case when considering global data in environmental applications, for which the data lie on a sphere representing our planet (see eg. [28]), and on this surface transport phenomena (due to winds and currents for instance) can affect the structure of the data (see eg. Figure 1.1B). Hence the main motivations of this work: proposing models for spatio-temporal GRFs flexible enough to represent complex patterns of correlations in data lying on compact meshed surfaces, and but simple enough so that numerically efficient algorithms for their inference, simulation and prediction can be derived.

ACKNOWLEDGMENT The author would like to thank Nicolas Desassis and Lucia Clarotto for their insightful comments and advices.



(A) Volumetric (left) and surface (right) representations of fMRI data. In the right picture, both the curvature of the surface and the BOLD response from the fMRI volume are represented. (Source: [20])

(B) Simulation of the presence of sulfate in the Earth atmosphere (Source: NASA Global Modeling and Assimilation Office)

Figure 1.1: Examples of spatio-temporal data distributed on surfaces.

2 Context and state-of-the art

When it comes to defining and building spatio-temporal GRFs to model data, there exists two main approaches: either through the definition of valid covariance functions (which are then “fitted” on the data), or through dynamical models describing the evolution in space and time of the GRFs. Let us review the principle and limits of both approaches.

2.1 The covariance-based approach

Since we consider GRFs, tasks such as sampling from (un)-conditional distributions, predictions (through conditional expectations),and likelihood-based inference can all be performed by solving linear systems or adequately factorizing covariance matrices of the field [33]. Hence, the most straightforward (and classical) approach to spatio-temporal geostatistical modeling consists in fitting valid space-time covariance functions on the data, so that these covariance matrices may be built. Consequently, extensive literature on which covariance functions may be used to model spatio-temporal data, even with complex correlation patterns, is available (see [7, 26, 27] for recent reviews).

Nonetheless, the covariance-based approach has two main drawbacks. First, the matrix factorizations required in sampling, prediction and inference tasks have a complexity that scales as the cube of the number of observations and/or target points, thus making them unfeasible when this last number is large. To circumvent this, simplifying assumptions on the covariance model must be made, such as the separability of space and time dependencies or stationarity. This in turn may result in a lack of realism of the model. Secondly, since they rely on Euclidean or arc-length distances, most of the covariance models available in the literature are restricted to the setting where the spatial domain is either Euclidean or the sphere . Hence, they hardly generalize to other surfaces.

2.2 The dynamic approach

As foretold by its name, this approach relies on models of the dynamic evolution of the GRF in time and space. These models take the form of stochastic partial differential equations (SPDE), the solutions of which are GRFs. This “SPDE approach” to GRF modeling has been popularized by Lindgren et al. [17], and builds on a result from Whittle [34] which states that isotropic GRFs Z on $\mathbb{R}^d$ ($d \in \mathbb{N}$) with a Matérn covariance function are stationary solutions of the SPDE given by

$$(\kappa^2 - \Delta)^{\alpha/2} Z = \tau \mathcal{W}, \tag{1}$$

where $\kappa > 0$, $\alpha > d/2$, $\tau > 0$, and $(\kappa^2 - \Delta)^{\alpha/2}$ is a pseudo-differential operator (defined as $(\kappa^2 - \Delta)^{\alpha/2}[\cdot] = \mathcal{F}^{-1}[\boldsymbol{w} \mapsto (\kappa^2 + \|\boldsymbol{w}\|^2)^{\alpha/2}\mathcal{F}[\cdot](\boldsymbol{w})]$), and $\mathcal{W}$ is a Gaussian white noise on $\mathbb{R}^d$. Solving numerically this SPDE using stochastic finite elements allows to directly obtain an expression for the precision matrix (i.e. the inverse of the covariance matrix) of a GRF with Matérn covariance. Then, at the price of a minor approximation (called mass lumping), these expressions yield a Gaussian Markov random field representation of the GRF, characterized by a sparse precision matrix [18, 29]. This in turn results in significant computational gains since sparse matrix algorithms can be used to deal with the matrix factorizations and linear system solving involved when performing sampling, prediction and inference [13, 17].

The SPDE approach has been extensively used to model spatial data on Euclidean domains (see [18] for a recent review), and extended to model spatial data on surfaces (see eg. [3, 11, 20]) and more generally on Riemannian manifolds (see eg. [15, 16]) by replacing the Laplace operator $-\Delta$ in SPDE (1) by a Laplace–Beltrami operator.

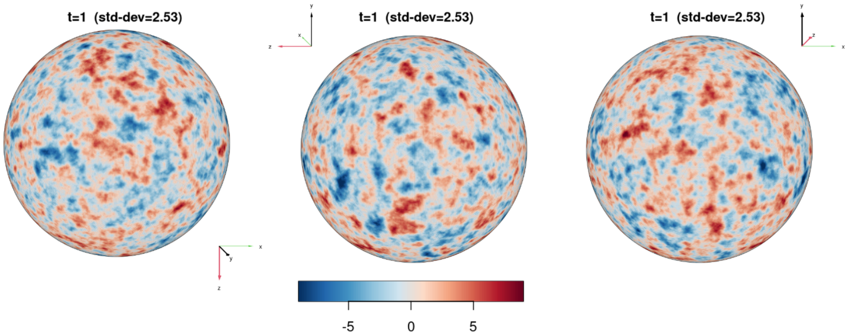


Figure 2.1: Simulation a spatio-temporal diffusion SPDE on the sphere represented from three different viewpoints on the surface (t represents the time step, std-dev the standard deviation of the field value across the surface).

Extensions of the SPDE approach to the spatio-temporal setting have also been proposed. Cameletti et al. [6] propose an approach where the spatial SPDE is coupled with an AR(1) process in time, thus yielding a separable model. Non-separable models based on a direct generalization of SPDE (1) have been proposed by Bakka et al. [1] and Rayner et al. [28], who consider the solutions of diffusion SPDE defined on Euclidean domains and on surfaces by

$$\frac{\partial Z}{\partial t} + (\kappa^2 - \Delta)^{\alpha/2} Z = \tau \mathcal{W}_T \otimes \mathcal{W}_S,$$

where $\mathcal{W}_T$ denotes a temporal white noise and $\mathcal{W}_S$ denotes either a white or colored noise in space. For reference, we provide in Figure 2.1 a simulation of a solution to this SPDE on the sphere. But these models result in random fields with even covariance functions, meaning that changing the sign of the spatial or temporal lag at which the covariance is evaluated does not change the value of the covariance. Consequently, these models are incapable of accounting for transport effects such as advection phenomena (which are intrinsically asymmetrical in time). Note however that in the Euclidean setting, extensions of the SPDE approach allowing to deal with asymmetries in the covariance structure have been proposed by Clarotto et al. [9], Liu et al. [19], Sigrist et al. [32]. However, to the best of our knowledge, the generalization of such models to more complex geometries is left open.

2.3 Proposed approach

The aim of this work is to propose new models for spatio-temporal data on meshed surfaces based on the SPDE modeling approach. To do so, we generalize the approach proposed by Clarotto et al. [9] to model spatio-temporal data on Euclidean domains using an advection-diffusion SPDE, to compact smooth orientable Riemannian manifolds of dimension 2. Then, following the framework proposed in [16], we define a counterpart to the advection-diffusion SPDE on a meshed triangulation of the manifold using Galerkin approximations of the differential operators. We show how this approach allows to easily propose scalable algorithms for the simulation and prediction of Gaussian random fields that are solutions to the resulting SPDE.

3 Advection-diffusion SPDE on a Riemannian manifold

We start by defining the advection-diffusion SPDE on a Riemannian manifold that will be considered on this work. Let $(\mathcal{M}, g)$ denote a compact orientable smooth Riemannian manifold of dimension $d = 2$, without boundary. Let $T > 0$, we denote by $-\Delta_{\mathcal{M}}$ the Laplace–Beltrami operator of the surface We consider the following advection-diffusion SPDE on the domain $[0, T] \times \mathcal{M}$:

$$\frac{\partial \mathcal{Z}}{\partial t} + \frac{1}{c}\big(P(-\Delta_{\mathcal{M}})\mathcal{Z} + \operatorname{div}(\mathcal{Z}\gamma)\big) = \frac{\tau}{\sqrt{c}} \mathcal{W}_T \otimes \mathcal{Y}_S, \tag{2}$$

where

- P is a polynomial taking positive values on $\mathbb{R}_+$,
- $s \in \mathcal{M} \mapsto \gamma(s)$ is a smooth vector field on $T\mathcal{M}$ (the tangent bundle of $\mathcal{M}$),
- $c > 0$ is a time-scaling parameter, and $\tau > 0$ is a variance-scaling parameter,

- $\mathcal{W}_T \otimes \mathcal{Y}_S$ is a space-time separable stochastic forcing given as the product of a time-dependent Gaussian white noise $\mathcal{W}_T$ and a space-dependent colored noise $\mathcal{Y}_S = f_S(-\Delta_{\mathcal{M}})\mathcal{W}_S$ where $f_S : \mathbb{R}_+ \rightarrow \mathbb{R}$ is a bounded function (cf. Appendix A.1 for a definition of colored noise).

In particular, we assume that the vector field γ and the solution $\mathcal{Z}$ are smooth enough so that the divergence term in (2) may be rewritten according to the Leibniz rule as

$$\operatorname{div}(\mathcal{Z}\gamma)(t,s) = \operatorname{div}(\gamma(s))\mathcal{Z}(t,s) + g_s(\gamma(s), \nabla\mathcal{Z}(t,s)), \quad s \in \mathcal{M},\ t \in [0,T]. \tag{3}$$

The space-time forcing is defined as a generalized random field acting on functions of $L^2([0,T]) \times L^2(\mathcal{M})$. Let $\langle\cdot,\cdot\rangle$ (resp. $\langle\cdot,\cdot\rangle_T$) denote the usual inner product on $L^2(\mathcal{M})$ (resp. $L^2([0,T])$). Then for any $(\phi_T, \phi_S), (\varphi_T, \varphi_S) \in L^2([0,T]) \times L^2(\mathcal{M})$, $\mathcal{W}_T \otimes \mathcal{Y}_S(\phi_T, \phi_S)$ and $\mathcal{W}_T \otimes \mathcal{Y}_S(\varphi_T, \varphi_S)$ are centered Gaussian random variables, and

$$\operatorname{Cov}\Big[\mathcal{W}_T \otimes \mathcal{Y}_S(\phi_T, \phi_S), \mathcal{W}_T \otimes \mathcal{Y}_S(\varphi_T, \varphi_S)\Big] = \langle\phi_T, \varphi_T\rangle_T \langle f_S(-\Delta_{\mathcal{M}})\phi_S, f_S(-\Delta_{\mathcal{M}})\varphi_S\rangle.$$

Note that this space-time forcing term can be identified with a cylindrical Wiener process $\{\widetilde{\mathcal{W}}_t\}_{t\in[0,T]}$ in $L^2(\mathcal{M})$ through

$$\widetilde{\mathcal{W}}_t(\phi_S) = (\mathcal{W}_T \otimes \mathcal{Y}_S)(\mathbb{1}_{[0,t]}, \phi_S), \quad \phi_S \in L^2(\mathcal{M}),\ t \in [0,T],$$

where $\mathbb{1}_{[0,t]}$ denotes the indicator function of the segment $[0,t]$ [4]. As such, we have (almost-surely) the following decomposition of $\widetilde{\mathcal{W}}_t$

$$\widetilde{\mathcal{W}}_t = \sum_{j\in\mathbb{N}} f_S(\lambda_j)\beta_j(t)e_j, \quad t \in [0,T], \tag{4}$$

where $\{e_j\}_{j\in\mathbb{N}}$ denotes an orthonormal basis of $L^2(\mathcal{M})$ composed eigenfunctions of the Laplace–Beltrami operator $-\Delta_{\mathcal{M}}$, and $\{\lambda_j\}_{j\in\mathbb{N}}$ their associated eigenvalues. This identification allows in turn to write

$$\mathcal{W}_T \otimes \mathcal{Y}_S(\phi_T, \phi_S) = \langle \int_0^T \phi_T \mathrm{d}\widetilde{\mathcal{W}}_t, \phi_S\rangle, \quad (\phi_T, \phi_S) \in L^2([0,T]) \times L^2(\mathcal{M}).$$

where the integral term is given by

$$\int_0^T \phi_T \mathrm{d}\widetilde{\mathcal{W}}_t = \sum_{j\in\mathbb{N}} f_S(\lambda_j)\Big(\int_0^T \phi_T \mathrm{d}\beta_j(t)\Big)e_j.$$

Hence, we can interpret the forcing term $\mathcal{W}_T \otimes \mathcal{Y}_S$ as the (time) derivative of the cylindrical Wiener process $\{\widetilde{\mathcal{W}}_t\}_{t\in[0,T]}$. This analogy allows in particular to rewrite SPDE (2) in the perhaps more familiar form for the readers used to stochastic differential equations (SDE) in infinite dimensions [12]

$$\mathrm{d}Z = -\frac{1}{c}\big(P(-\Delta_{\mathcal{M}})\mathcal{Z} + \operatorname{div}(\mathcal{Z}\gamma)\big)\mathrm{d}t + \frac{\tau}{\sqrt{c}}\mathrm{d}\widetilde{\mathcal{W}}_t.$$

We conclude this section with a few words about the vector field γ. A natural way to define (and parametrize) a smooth vector field γ on a arbitrary manifold $\mathcal{M}$ is to assume that it is the gradient of a smooth scalar function $\xi : \mathcal{M} \rightarrow \mathbb{R}$, i.e. for any $s \in \mathcal{M}$,

$$\gamma(s) = \nabla\xi(s) \in T_s\mathcal{M}. \tag{5}$$

In this setting, the function ξ can be seen as a potential whose spatial variations locally define the direction of the advection.

Note that this decomposition is not general enough to parametrize all smooth vector fields. For instance, similarly to the Euclidean case, the curl of vector fields defined as in (5) will be zero. A complete characterization of smooth vector fields is given by Helmholtz-Hodge decomposition theorem, which states that vector fields may be uniquely decomposed as the sum of a irrotational component (whose curl is zero), a divergence-free component (whose div is zero) and a harmonic component [2]. In some specific cases, this decomposition can be easily parametrized. For instance, if $\mathcal{M} = \mathbb{S}^2$ is the 2-sphere, we can decompose any tangent vector field γ as

$$\gamma(s) = \nabla\xi(s) + \vec{n}(s) \times \nabla\chi(s) \in T_s\mathbb{S}^2, \tag{6}$$

for some scalar functions $\xi, \chi : \mathbb{S}^2 \rightarrow \mathbb{R}$, and with $\vec{n}(s)$ denoting the vector normal to $\mathbb{S}^2$ and pointing outwards [22].

4 Advection-diffusion SPDE on a meshed surface

4.1 Definition and discretization of the SPDE

Let $\mathcal{M}_h$ be a discretization of the manifold $\mathcal{M}$ into a polyhedral surface with mesh size $h > 0$ (by triangulation). Let $\{\psi_1, \dots, \psi_N\} \subset H^1(\mathcal{M}_h)$ be the the *linear* finite element basis associated with $\mathcal{M}_h$, where N is the number of nodes of the triangulation. Let then $V_N = \operatorname{span}\{\psi_k : 1 \le k \le N\}$.

In order to formulate an advection-diffusion SPDE on $\mathcal{M}_h$, we look for an approximation of the solution $\mathcal{Z}$ of (2) that can be expressed as a V_N-valued random variable (at any time). We obtain it by approximating each term in the SPDE by operators or variables that "live" on V_N. This is done in two steps. First, the colored noise $\mathcal{Y}_S$ is approximated by a V_N-valued random variable Y_S defined by

$$Y_S = f_S(-\Delta_N)W_S = \sum_{k=1}^{n} w_k f_S(\lambda_k^{(N)}) e_k^{(N)}, \tag{7}$$

where $\{w_k\}_{1\le k\le N}$ a sequence of independent standard Gaussian variable. Note that this definition holds for any bounded f_S, and is equivalent to the definition of colored noise previously introduced, after replacing H by V_N and the Laplace–Beltrami operator $-\Delta_{\mathcal{M}}$ by its Galerkin approximation $-\Delta_N$. Besides, when either of the decompositions (5)-(6) is used, we assume that the potential functions ξ and χ are also taken in V_N, which yields a piecewise constant approximation of the vector field. Details about the computations related to such vector fields can be found in [25].

Let $-\Delta_N$ be the Galerkin approximation of $-\Delta_{\mathcal{M}}$ over V_N (cf. Appendix A.2) and let $\{\lambda_k^{(N)}\}_{1\le k\le N}$ denote its eigenvalues, and $\{e_k^{(N)}\}_{1\le k\le N}$ be a set of associated eigenfunctions forming an orthonormal basis of V_N. Then, we rewrite (2) by replacing $\mathcal{Z}(t,\cdot)$ by an approximation $Z(t,\cdot) \in V_N$, and replacing $-\Delta_{\mathcal{M}}$ by its approximation $-\Delta_N$, thus giving

$$\frac{\partial Z}{\partial t} + \frac{1}{c}\Big(P(-\Delta_N)Z + \operatorname{div}(\gamma Z)\Big) = \frac{\tau}{\sqrt{c}}\mathcal{W}_T \otimes Y_S, \quad t \in [0,T], \tag{8}$$

where $\mathcal{W}_T \otimes Y_S$ is defined in the same way as its counterpart $\mathcal{W}_T \otimes \mathcal{Y}_S$, i.e. as a generalized random field acting on functions of $L^2([0,T]) \times V_N$. Note in particular that the gradients and integrals now involved in the definition of (8) are now taken on the polyhedral surface.

Remark 4.1. *Once again, we can be identify $\mathcal{W}_T \otimes Y_S$ with a cylindrical Wiener process $\{\widetilde{W}_t\}_{t\in[0,T]}$ in V_N, which can be decomposed as* (4) *where the eigenfunctions $\{e_j\}_{j\in\mathbb{N}}$ and eigenvalues $\{\lambda_j\}_{j\in\mathbb{N}}$ of $-\Delta_{\mathcal{M}}$, by the eigenfunctions $\{e_j^{(N)}\}_{1\le j\le N}$ and eigenvalues $\{\lambda_j^{(N)}\}_{1\le j\le N}$ of $-\Delta_N$.*

We now discretize (8) in time, by applying an implicit Euler scheme with time step $\delta t > 0$. We start by discretizing the time interval $[0,T]$ into $K+1$ regular time steps of size $\delta t = T/K$, and write $t_k = k\delta t$ for $k \in \{0,\dots,K\}$. Let then $Z^{(k)} = Z(t_k,\cdot)$ denote the approximation of the spatial trace of the solution to SPDE (8) at time t_k. Starting from an initial condition $Z^{(0)} = Z(0,\cdot) \in V_N$ (taken for instance as the projection onto V_N of the initial condition $\mathcal{Z}(0,\cdot)$ of the original SPDE (2)), we have the recursion

$$Z^{(k+1)} - Z^{(k)} + \frac{\delta t}{c}\Big(P(-\Delta_N)Z^{(k+1)} + \operatorname{div}(\gamma Z^{(k+1)})\Big) = \tau\sqrt{\frac{\delta t}{c}}Y^{(k+1)}, \quad k \in \mathbb{N}_0, \tag{9}$$

where $\{Y^{(k)}\}_{k\in\mathbb{N}}$ is a sequence of independent samples of Y_S.

The next proposition presents explicit formulas to compute this recursion. First, let us introduce some standard finite element discretization matrices. We denote by $\boldsymbol{C}$, $\boldsymbol{R}$ and $\boldsymbol{B}$ the matrices whose entries are respectively given by

$$C_{ij} = \langle \psi_i, \psi_j \rangle, \quad R_{ij} = \langle \nabla\psi_i, \nabla\psi_j \rangle, \quad B_{ij} = \langle \psi_i, \operatorname{div}(\gamma\psi_j) \rangle, \quad 1 \le i,j \le N, \tag{10}$$

and let $\sqrt{\boldsymbol{C}} \in \mathbb{R}^{N\times N}$ such that $\boldsymbol{C} = \sqrt{\boldsymbol{C}}(\sqrt{\boldsymbol{C}})^T$. We also introduce the scaled matrices $\tilde{\boldsymbol{R}}$ and $\tilde{\boldsymbol{B}}$ defined by

$$\widetilde{\boldsymbol{R}} = (\sqrt{\boldsymbol{C}})^{-1}\boldsymbol{R}(\sqrt{\boldsymbol{C}})^{-T}, \quad \widetilde{\boldsymbol{B}} = (\sqrt{\boldsymbol{C}})^{-1}\boldsymbol{B}(\sqrt{\boldsymbol{C}})^{-T}.$$

Proposition 4.2. *For $0 \le k \le K$, let $\boldsymbol{z}^{(k)} = (z_1^{(k)}, \dots, z_N^{(k)})^T$ be the the (random) vector such that*

$$Z^{(k)} = \sum_{j=1}^{N} z_j^{(k)} \psi_j \tag{11}$$

We also denote by $\mathbf{\Gamma}$ *the matrix defined by*

$$\mathbf{\Gamma} = \boldsymbol{I} + \frac{\delta t}{c}\big(P(\widetilde{\boldsymbol{R}}) + \widetilde{\boldsymbol{B}}\big). \tag{12}$$

Let $\boldsymbol{x}^{(0)} = (\sqrt{\boldsymbol{C}})^T \boldsymbol{z}^{(0)}$. *Then we have the following recursion for* $0 \leq k < K$

$$\begin{cases} \mathbf{\Gamma}\boldsymbol{x}^{(k+1)} = \boldsymbol{x}^{(k)} + f_{\delta t}(\widetilde{\boldsymbol{R}})\boldsymbol{w}^{(k+1)}, \\ \boldsymbol{z}^{(k+1)} = (\sqrt{\boldsymbol{C}})^{-T}\boldsymbol{x}^{(k+1)}, \end{cases} \tag{13}$$

where $\{\boldsymbol{w}^{(k)}\}_{k\in\mathbb{N}}$ *is a sequence of independent centered Gaussian vectors with covariance matrix* $\boldsymbol{I}$*, and* $f_{\delta t}$ *is the function defined by*

$$f_{\delta t}(\lambda) = \tau\sqrt{\frac{\delta t}{c}} f_S(\lambda), \quad \lambda \geq 0,$$

and the matrix function $f_{\delta t}(\widetilde{\boldsymbol{R}})$ *is defined in Appendix A.3.*

Proof. We write for $k \in \mathbb{N}_0$, $\boldsymbol{\alpha}^{(k)} = (\alpha_1^{(k)}, \dots, \alpha_N^{(k)})^T \in \mathbb{R}^N$, $\boldsymbol{y}^{(k)} = (y_1^{(k)}, \dots, y_N^{(k)})^T$, where

$$P(-\Delta_N)Z^{(k)} = \sum_{j=1}^{N} \alpha_j^{(k)}\psi_j, \quad Y^{(k)} = \sum_{j=1}^{N} y_j^{(k)}\psi_j.$$

Besides, let $\boldsymbol{\xi}^{(k)}$ be the vector defined by $\boldsymbol{\xi}^{(k)} = \big(\langle Z^{(k)}, e_1^{(N)}\rangle, \dots, \langle Z^{(k)}, e_N^{(N)}\rangle\big)^T$.

Firstly, note that, following [16, Theorem 3.4], we can take

$$\boldsymbol{y}^{(k)} = (\sqrt{\boldsymbol{C}})^{-T} f_S(\widetilde{\boldsymbol{R}})\boldsymbol{w}^{(k)}, \quad k \in \mathbb{N},$$

where $\{\boldsymbol{w}^{(k)}\}_{k\in\mathbb{N}}$ is a sequence of independent centered Gaussian vectors with covariance matrix $\boldsymbol{I}$.

Then, by testing (9) against ψ_i (for $i \in \{1, \dots, N\}$), and injecting (11), we get the following linear system of equations

$$\boldsymbol{C}\boldsymbol{z}^{(k+1)} - \boldsymbol{C}\boldsymbol{z}^{(k)} + \frac{\delta t}{c}\bigg(\boldsymbol{C}\boldsymbol{\alpha}^{(k+1)} + \boldsymbol{B}\boldsymbol{z}^{(k+1)}\bigg) = \tau\sqrt{\frac{\delta t}{c}}\boldsymbol{C}\boldsymbol{y}^{(k+1)} \tag{14}$$

On the one hand, following the definition of the map E in (27), we have

$$Z^{(k+1)} = E\bigg((\sqrt{\boldsymbol{C}})^T\boldsymbol{z}^{(k+1)}\bigg) \quad \text{and} \quad P(-\Delta_N)Z^{(k+1)} = E\bigg((\sqrt{\boldsymbol{C}})^T\boldsymbol{\alpha}^{(k+1)}\bigg) \tag{15}$$

On the other hand, following the definition of the basis $\{e_j^{(N)}\}_{1\leq j\leq N}$,

$$Z^{(k+1)} = \sum_{j=1}^{N}\langle Z^{(k+1)}, e_j^{(N)}\rangle E(\boldsymbol{v}_j) = E\bigg(\sum_{j=1}^{N}\langle Z^{(k+1)}, e_j^{(N)}\rangle \boldsymbol{v}_j\bigg) = E\big(\boldsymbol{V}\boldsymbol{\xi}^{(k+1)}\big)$$

Hence, since E is invertible, we have $(\sqrt{\boldsymbol{C}})^T\boldsymbol{z}^{(k+1)} = \boldsymbol{V}\boldsymbol{\xi}^{(k+1)}$.

Similarly, by definition of $P(-\Delta_N)$, we have

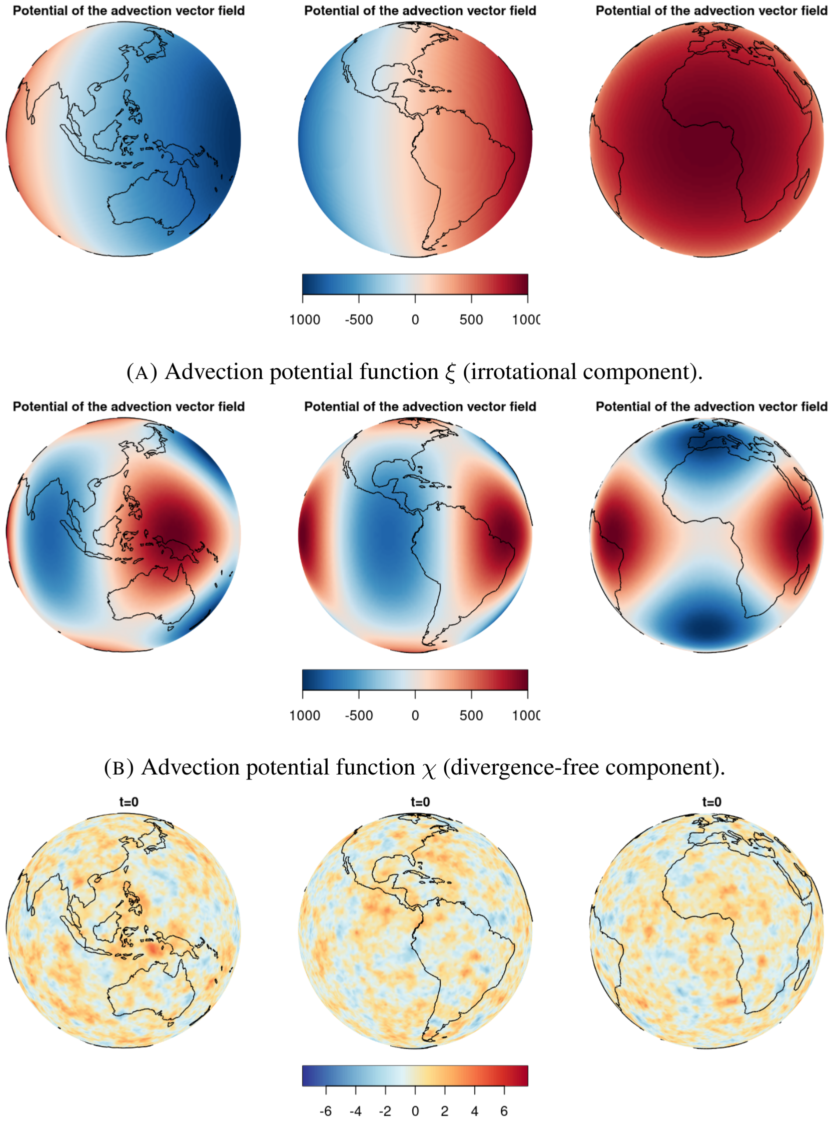
$$P(-\Delta_N)Z^{(k+1)} = \sum_{j=1}^{N} P(\lambda_j^{(N)})\langle Z^{(k+1)}, e_j^{(N)}\rangle e_j^{(N)} = E\big(\boldsymbol{V}P(\mathbf{\Lambda}^{(N)})\boldsymbol{\xi}^{(k+1)}\big) = E\big(P(\widetilde{\boldsymbol{R}})\boldsymbol{V}\boldsymbol{\xi}^{(k+1)}\big)$$

Therefore, we have $P(-\Delta_N)Z^{(k+1)} = E\big(P(\widetilde{\boldsymbol{R}})(\sqrt{\boldsymbol{C}})^T\boldsymbol{z}^{(k+1)}\big)$, and using (15) and the fact that E is invertible, we can deduce that $\boldsymbol{\alpha}^{(k+1)} = (\sqrt{\boldsymbol{C}})^{-T}P(\widetilde{\boldsymbol{R}})(\sqrt{\boldsymbol{C}})^T\boldsymbol{z}^{(k+1)}$.

In conclusion, we can now rewrite (14) as

$$\bigg(\boldsymbol{C} + \frac{\delta t}{c}(\sqrt{\boldsymbol{C}})P(\widetilde{\boldsymbol{R}})(\sqrt{\boldsymbol{C}})^T + \frac{\delta t}{c}\boldsymbol{B}\bigg)\boldsymbol{z}^{(k+1)} = \boldsymbol{C}\boldsymbol{z}^{(k)} + \tau\sqrt{\frac{\delta t}{c}}\boldsymbol{C}\boldsymbol{y}^{(k+1)}.$$

By then introducing the vectors $\boldsymbol{x}^{(k)} = (\sqrt{\boldsymbol{C}})^T\boldsymbol{z}^{(k)}$, we retrieve the recursion (13). □


Potential of the advection vector field
Potential of the advection vector field
Potential of the advection vector field
1000
-500
0
500
1000
Potential of the advection vector field
Potential of the advection vector field
Potential of the advection vector field
1000
-500
0
500
1000
t=0
t=0
t=0
-6
-4
-2
0
2
4
6

(A) Advection potential function ξ (irrotational component).

(B) Advection potential function χ (divergence-free component).

(C) Numerical solution of the SPDE.

Figure 4.1: Simulation of a solution to the advection-diffusion SPDE on a meshed sphere, represented from three different viewpoints on the surface.

Remark 4.3. *In the case where there is no advection ($\gamma = 0$), an exact time integrator can be proposed for* (8)*. This resulting numerical scheme can be applied to uneven time discretization steps, and relies solely on matrix functions of the scaled matrix $\widetilde{\boldsymbol{R}}$. More details are provided in Appendix C.*

We present in Figure 4.1 an example of simulation to the advection-diffusion SPDE on a meshed sphere. We took P to be a polynomial of degree 1 and f_S to be the inverse of a polynomial of degree 1. The vector field was parametrized thanks to two scalar functions as in (6). As seen in the simulation, the resulting random fields seems to be able to recreate complex convection and diffusion phenomena.

4.2 Stabilization

Similarly to the Euclidean case, the Galerkin approximation of advection-diffusion SPDEs on surfaces are subject to instabilities when the advection term dominates the diffusion term [21]. This phenomenon can be observed in Figure 4.2B, where a GRF on a cortical surface is simulated using the scheme described in Proposition 4.4: as one can notice in the animation, the values taken by the field quickly explode and tend to oscillate. These instabilities are due to the fact that in advection-dominated cases, the non-symmetric part of $\mathbf{\Gamma}$ in (12) dominates and causes the linear systems involving $\mathbf{\Gamma}$ to become ill-conditioned. To avoid such instabilities, it is usual to either sufficently decrease the time step δt, or to introduce a stabilization term to advection-diffusion SPDEs [10].

In the latter case, following the approach of Clarotto et al. [9], we use the Streamline Diffusion stabilization method [5]. This method consists in adding n additional term in the variational formulation of SPDE (8), namely the bilinear form defined on V_N by

$$a_S(v_1, v_2) = h\langle g(\gamma, \nabla v_1), \frac{1}{\sqrt{g(\gamma,\gamma))}} g(\gamma, \nabla v_2)\rangle, \quad v_1, v_2 \in V_N,$$

where h denotes the size of the triangulation mesh. This term effectively adds a small diffusion along the local direction of the advection γ. Note that scaling this term by h allows to minimize its impact on the SPDE solution: for instance, in the Euclidean setting, the resulting errors due to the introduction of this term are of order $\mathcal{O}(h)$.

In practice, when adding the stabilization term SPDE (8), the derivation of the recursion remains the same. The only change is the introduction of an additional matrix in the definition (12) of $\mathbf{\Gamma}$, which now reads:

$$\mathbf{\Gamma} = \boldsymbol{I} + \frac{\delta t}{c}\big(P(\widetilde{\boldsymbol{R}}) + \widetilde{\boldsymbol{B}} + \widetilde{\boldsymbol{S}}\big), \tag{16}$$

where $\widetilde{\boldsymbol{S}} = (\sqrt{\boldsymbol{C}})^{-1}\boldsymbol{S}(\sqrt{\boldsymbol{C}})^{-T}$ is the scaled stabilization matrix, and the entries of $\boldsymbol{S}$ are given by

$$S_{ij} = h\langle g(\gamma, \nabla\psi_i), \frac{1}{\sqrt{g(\gamma,\gamma))}} g(\gamma, \nabla\psi_j)\rangle, \quad 1 \le i, j \le N.$$

As seen Figure 4.2C, this method allowed us to get rid of the instabilities in the simulation.

4.3 Covariance structure

The recursion in Proposition 4.2 can be used to derive, for any $K \in \mathbb{N}$, the expression of the precision matrix of the vectors $\boldsymbol{z}^{(0)}, \dots, \boldsymbol{z}^{(K)}$. Let us first introduce some notations. For $\mathbf{\Theta}_1 \in \mathbb{R}^{N\times N}$, $\mathbf{\Theta}_2$ invertible, let $\boldsymbol{L}(\mathbf{\Theta}_1, \mathbf{\Theta}_2) \in \mathbb{R}^{(K+1)N\times(K+1)N}$ and $\boldsymbol{D}(\mathbf{\Theta}_1, \mathbf{\Theta}_2) \in \mathbb{R}^{(K+1)N\times(K+1)N}$ be the block matrices given by

$$\boldsymbol{L}(\mathbf{\Theta}_1, \mathbf{\Theta}_2) = \begin{pmatrix} \boldsymbol{I} & & & \\ -\mathbf{\Theta}_2 & \mathbf{\Theta}_1 & & \\ & \ddots & \ddots & \\ & & -\mathbf{\Theta}_2 & \mathbf{\Theta}_1 \end{pmatrix}, \quad \text{and} \quad \boldsymbol{D}(\mathbf{\Theta}_1, \mathbf{\Theta}_2) = \begin{pmatrix} \mathbf{\Theta}_1 & & & \\ & \mathbf{\Theta}_2 & & \\ & & \ddots & \\ & & & \mathbf{\Theta}_2 \end{pmatrix}. \tag{17}$$

The next proposition gives the expression of the precision matrix of the coefficients $\boldsymbol{z}^{(0)}, \dots, \boldsymbol{z}^{(K)}$ obtained through the recursion (13).

Proposition 4.4. *Let us assume that the initial condition of SPDE* (8) *can be expressed as*

$$Z(0, \cdot) = f_0(-\Delta_N)Y_N$$

for some function $f_0 : \mathbb{R}_+ \to \mathbb{R}$ that is bounded takes positive values. Let then $\boldsymbol{X} = ((\boldsymbol{x}^{(0)})^T, \dots, (\boldsymbol{x}^{(K)})^T)^T$ be the vector obtained by concatenating the vectors $\boldsymbol{x}^{(0)}, \dots, \boldsymbol{x}^{(K)}$ defined in Proposition 4.2, and similarly, let $\boldsymbol{Z} = ((\boldsymbol{z}^{(0)})^T, \dots, (\boldsymbol{z}^{(K)})^T)^T$.

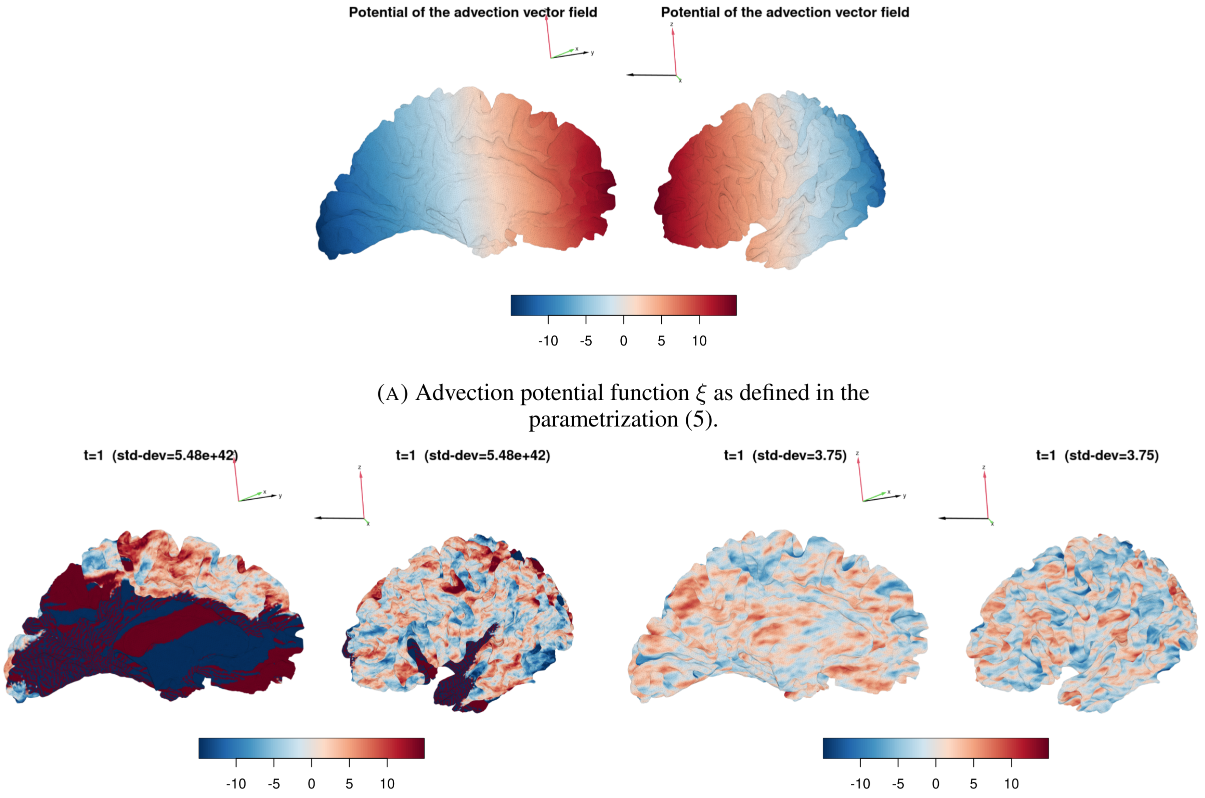
Let $\boldsymbol{Q_X}$ (resp. $\boldsymbol{Q_Z}$) the precision matrix of $\boldsymbol{X}$ (resp. $\boldsymbol{Z}$). Then we have the following relations

$$\begin{cases} \boldsymbol{Q_X} = \boldsymbol{L}(\mathbf{\Gamma}, \boldsymbol{I})^T\, \boldsymbol{D}\big(f_0^{-2}(\widetilde{\boldsymbol{R}}),\, f_{\delta t}^{-2}(\widetilde{\boldsymbol{R}})\big)\, \boldsymbol{L}(\mathbf{\Gamma}, \boldsymbol{I}), \\ \boldsymbol{Q_Z} = \boldsymbol{D}\big(\sqrt{\boldsymbol{C}},\, \sqrt{\boldsymbol{C}}\big)\, \boldsymbol{Q_X}\, \boldsymbol{D}\big((\sqrt{\boldsymbol{C}})^T,\, (\sqrt{\boldsymbol{C}})^T\big), \end{cases} \tag{18}$$

Proof. First, note that by definition of $Z(0, \cdot)$ and f_0, we can write $\boldsymbol{z}^{(0)} = (\sqrt{\boldsymbol{C}})^{-T} f_0(\widetilde{\boldsymbol{R}})\boldsymbol{w}^{(0)}$ where $\boldsymbol{w}^{(0)} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$ (cf. proof of Proposition 4.2), and the function $f_0^{-1} = (1/f_0)$ is well-defined on $\mathbb{R}_+$.

Let $\boldsymbol{W} = ((\boldsymbol{w}^{(0)})^T, \dots, (\boldsymbol{w}^{(K)})^T)^T$ be defined from the vectors $\boldsymbol{w}^{(0)})$ in Proposition 4.2. In particular, its precision matrix is $\boldsymbol{Q_W} = \boldsymbol{I}$. We can then rewrite (13) in matrix form as

$$\boldsymbol{L}(\mathbf{\Gamma}, \boldsymbol{I})\boldsymbol{X} = \boldsymbol{D}\big(f_0(\widetilde{\boldsymbol{R}}),\, f_{\delta t}(\widetilde{\boldsymbol{R}})\big)\boldsymbol{W}.$$



(A) Advection potential function ξ as defined in the parametrization (5).

(B) Numerical solution of the SPDE, without stabilization.

(C) Numerical solution of the SPDE, with stabilization.

Figure 4.2: Simulation of the spatio-temporal advection-diffusion SPDE on a cortical surface, represented from three different viewpoints on the surface (t represents the time step, std-dev the standard deviation of the field values across the surface).

Taking the covariance of both side of this inequality, we get

$$\boldsymbol{L}(\boldsymbol{\Gamma}, \boldsymbol{I})\boldsymbol{Q}_{\boldsymbol{X}}^{-1}\boldsymbol{L}(\boldsymbol{\Gamma}, \boldsymbol{I})^T = \boldsymbol{D}\big(f_0^2(\widetilde{\boldsymbol{R}}),\, f_{\delta t}^2(\widetilde{\boldsymbol{R}})\big),$$

where we used the fact that $\boldsymbol{D}\big(f_0(\widetilde{\boldsymbol{R}}),\, f_{\delta t}(\widetilde{\boldsymbol{R}})\big)$ is block diagonal and that $\boldsymbol{Q}_{\boldsymbol{W}} = \boldsymbol{I}$. By then inverting both sides, we then retrieve our claim on $\boldsymbol{Q}_{\boldsymbol{X}}$. Finally, note that by definition of $\boldsymbol{x}^{(k)}$ $(k \geq 0)$ in Proposition 4.2, we have $\boldsymbol{X} = \boldsymbol{D}\big((\sqrt{\boldsymbol{C}})^T,\, (\sqrt{\boldsymbol{C}})^T\big)\boldsymbol{Z}$. By once again taking the covariance and then inverting both sides of this equality, we retrieve our claim on $\boldsymbol{Q}_{\boldsymbol{Z}}$. □

Remark 4.5. *The explicit computation of the precision matrix $\boldsymbol{Q}_{\boldsymbol{Z}}$ results in a block tri-diagonal matrix given by*

$$\boldsymbol{Q}_{\boldsymbol{Z}} = \begin{pmatrix} \boldsymbol{\Gamma}_0 & -\boldsymbol{\Gamma}_2 & & & \\ -\boldsymbol{\Gamma}_2^T & \boldsymbol{\Gamma}_1 & - & & \\ \boldsymbol{\Gamma}_2 & & & & \\ & \ddots & \ddots & \ddots & \\ & & -\boldsymbol{\Gamma}_2^T & \boldsymbol{\Gamma}_1 & -\boldsymbol{\Gamma}_2 \\ & & & -\boldsymbol{\Gamma}_2^T & \boldsymbol{\Gamma}_3 \end{pmatrix},$$

where

$$\boldsymbol{\Gamma}_0 = (\sqrt{\boldsymbol{C}})\big(f_0^{-2}(\widetilde{\boldsymbol{R}}) + f_{\delta t}^{-2}(\widetilde{\boldsymbol{R}})\big)(\sqrt{\boldsymbol{C}})^T, \qquad \boldsymbol{\Gamma}_1 = (\sqrt{\boldsymbol{C}})\big(\boldsymbol{\Gamma}^T f_{\delta t}^{-2}(\widetilde{\boldsymbol{R}})\boldsymbol{\Gamma} + f_{\delta t}^{-2}(\widetilde{\boldsymbol{R}})\big)(\sqrt{\boldsymbol{C}})^T,$$

$$\boldsymbol{\Gamma}_2 = (\sqrt{\boldsymbol{C}})f_{\delta t}^{-2}(\widetilde{\boldsymbol{R}})\boldsymbol{\Gamma}(\sqrt{\boldsymbol{C}})^T, \qquad \boldsymbol{\Gamma}_3 = (\sqrt{\boldsymbol{C}})\boldsymbol{\Gamma}^T f_{\delta t}^{-2}(\widetilde{\boldsymbol{R}})\boldsymbol{\Gamma}(\sqrt{\boldsymbol{C}})^T.$$

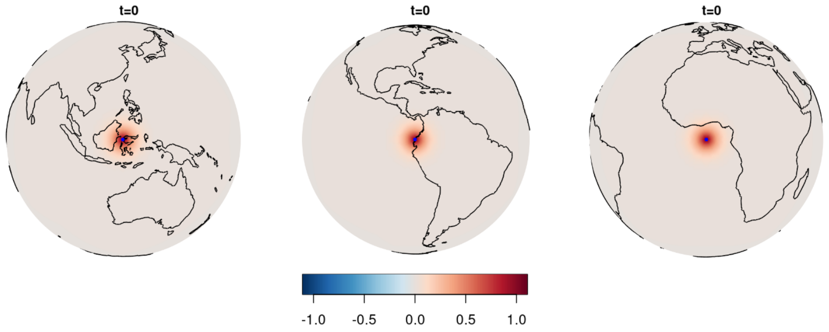


Figure 4.3: Spatio-temporal evolution of the covariance between a reference point (in blue) and the rest of the points in the domain, for the spatio-temporal model simulated in Figure 4.1. The color (red to blue) represent the value of the covariance.

As an illustration, starting from the same model as the one simulated in Figure 4.1, we represent in Figure 4.3 the spatio-temporal evolution of the covariance between the value of the field at time $t=0$ at three reference points (in blue), and the values of the field elsewhere and at later times. These covariance are computed using the formula in Proposition 4.4 for the spatio-temporal precision matrix of the field. As one can note in the animation, the zone of high-correlation "moves" along the advection direction, as expected in an advection problem.

The covariance matrix $\boldsymbol{Q_Z}$ of the Euler discretization of Z can be expressed as the product of block diagonal and block bi-diagonal matrices. This particular structure allows to propose scalable algorithms to perform products with vectors, solving linear systems and computing the log-determinant, all of which will prove useful when tackling inference and prediction problems.

For instance, the log-determinants of $\boldsymbol{Q_X}$ and $\boldsymbol{Q_Z}$ can be deduced from the log-determinants of the diagonal block entries of the matrices $\boldsymbol{L}(\boldsymbol{\Gamma}, \boldsymbol{I})$, $\boldsymbol{D}\big(f_0^{-2}(\widetilde{\boldsymbol{R}}),\, f_{\delta t}^{-2}(\widetilde{\boldsymbol{R}})\big)$ and $\boldsymbol{D}\big(\sqrt{\boldsymbol{C}},\, \sqrt{\boldsymbol{C}}\big)$. We obtain in particular the relations

$$\begin{cases} \log|\boldsymbol{Q_X}| = K\log|\boldsymbol{\Gamma}^T\boldsymbol{\Gamma}| + \log|f_0^{-2}(\widetilde{\boldsymbol{R}})| + K\log|f_{\delta t}^{-2}(\widetilde{\boldsymbol{R}})| = K\log|\boldsymbol{\Gamma}^T\boldsymbol{\Gamma}| - \log|f_0^2(\widetilde{\boldsymbol{R}})| - K\log|f_{\delta t}^2(\widetilde{\boldsymbol{R}})|, \\ \log|\boldsymbol{Q_Z}| = 2(K+1)\log|\sqrt{\boldsymbol{C}}| + \log|\boldsymbol{Q_X}|, \end{cases}$$

Hence, in order to compute its log-determinant, we do not need to build and store the matrix $\boldsymbol{Q_Z}$, but rather only need to store the matrices $\widetilde{\boldsymbol{R}}$, $\boldsymbol{\Gamma}$ and $\sqrt{\boldsymbol{C}}$. This ensures that the storage needs for that computation remain the same as the number of time steps considered K increases (which is not the case when using directly the matrix $Q_{\boldsymbol{Z}} \in \mathbb{R}^{(K+1)N\times(K+1)N}$).

As for the matrix-vector product $\boldsymbol{F} = ((\boldsymbol{f}^{(0)})^T, \dots, (\boldsymbol{f}^{(K)})^T)^T = \boldsymbol{Q_Z E}$ (for $\boldsymbol{E} = ((\boldsymbol{e}^{(0)})^T, \dots, (\boldsymbol{e}^{(K)})^T)^T \in \mathbb{R}^{(K+1)N}$), it can be computed iteratively while requiring only products between the matrices $\widetilde{\boldsymbol{R}}$, $\boldsymbol{\Gamma}$ and $\sqrt{\boldsymbol{C}}$ and vectors (cf. Algorithm 1). Finally, the linear system $\boldsymbol{L}(\boldsymbol{\Theta}_1, \boldsymbol{\Theta}_2)\boldsymbol{E} = \boldsymbol{F}$ can be also solved iteratively by substitution, by leveraging the bi-diagonal structure of the matrix $\boldsymbol{L}(\boldsymbol{\Gamma}, \boldsymbol{I})$. This iterative approach only requires to compute products between functions of the matrix $\widetilde{\boldsymbol{R}}$ and vectors, and to be able to solve linear systems involving $\boldsymbol{\Gamma}$ (cf. Algorithm 4).

5 Prediction and inference from data

Let $T>0$. We assume that the time interval $[0,T]$ is discretized into $K+1$ regular time steps of size $\delta t = T/K$. We hence write $t_k = k\delta t$ for $k \in \{0, \dots, K\}$.

We consider a statistical model with fixed and random effects to model observations of some variable u in the spatiotemporal domain $[0,T]\times\mathcal{M}_h$. The fixed effects correspond to a regression of a set of q covariates, and the random effects are modeled as the solution , to which an independent measurement noise is added. More precisely, for each $k \in \{0, \dots, K\}$, we assume that we have n_k observations of u at some locations $s_1^{(k)}, \dots, s_{n_k}^{(k)} \in \mathcal{M}_h$. We then denote by $\boldsymbol{u}^{(k)} \in \mathbb{R}^{n_k}$ the vector containing these observations, i.e. $\boldsymbol{u}^{(k)} = \big(u\big(t_k, s_1^{(k)}\big), \dots, u\big(t_k, s_{n_k}^{(k)}\big)\big)^T$. Finally, we denote by $\boldsymbol{U} \in \mathbb{R}^{N_o}$ the vector containing all the $N_o = n_0 + \dots + n_K$ observations, i.e. $\boldsymbol{U} = \big(\big(\boldsymbol{u}^{(0)}\big)^T, \dots, \big(\boldsymbol{u}^{(K)}\big)^T\big)^T$. Let $\boldsymbol{\varepsilon}$ be a vector of N_o independent standard Gaussian variables and $\sigma>0$. Then the statistical model for the observations takes the form

$$\boldsymbol{U} = \boldsymbol{\eta b} + \boldsymbol{A}^T\boldsymbol{Z} + \sigma\boldsymbol{\varepsilon},$$

where $\boldsymbol{b} \in \mathbb{R}^q$ is the vector of q fixed effects, $\boldsymbol{\eta} \in \mathbb{R}^{N_o \times q}$ is a matrix of covariates, $\boldsymbol{Z}$ is the vector containing the weights defining the solution of SPDE (8) as in Proposition 4.4, and $\boldsymbol{A} \in \mathbb{R}^{(K+1)N \times N_o}$ is the block diagonal matrix whose k-th block ($k \in \{0, \dots, K\}$) $\boldsymbol{A}^{(k)} \in \mathbb{R}^{N \times n_k}$ is defined by

$$[\boldsymbol{A}^{(k)}]_{ij} = \psi_i(s_j^{(k)}), \quad i \in \{1, \dots, N\},\ j \in \{1, \dots, n_k\}.$$

5.1 Prediction by kriging

Predictions of the spatio-temporal field are tackled using conditional expectations (and variances), following the approach outlined by Clarotto et al. [9], Pereira et al. [24]. The next proposition provides explicit formulas for the computation of the conditional expectation and variance of the field $\boldsymbol{Z}$ given the observations $\boldsymbol{U}$.

Proposition 5.1. *The conditional expectation (also called kriging predictor) $\mathbb{E}[\boldsymbol{Z}|\boldsymbol{U}]$ of $\boldsymbol{Z}$ given $\boldsymbol{U}$ is given*

$$\mathbb{E}[\boldsymbol{Z}|\boldsymbol{U}] = \mathcal{K}_{\boldsymbol{Q}_{\boldsymbol{Z}},\sigma^2}(\boldsymbol{U} - \boldsymbol{\eta}\boldsymbol{b}) = (\boldsymbol{Q}_{\boldsymbol{Z}} + \sigma^{-2}\boldsymbol{A}\boldsymbol{A}^T)^{-1}\boldsymbol{A}(\boldsymbol{U} - \boldsymbol{\eta}\boldsymbol{b}),$$

and where the map $\mathcal{K}$ is defined in (21). *Besides, the conditional variance* $\mathrm{Var}[\boldsymbol{Z}|\boldsymbol{U}]$ *is given by*

$$\mathrm{Var}[\boldsymbol{Z}|\boldsymbol{U}] = (\boldsymbol{Q}_{\boldsymbol{Z}} + \sigma^{-2}\boldsymbol{A}\boldsymbol{A}^T)^{-1}.$$

Proof. This result is a direct consequence of the fact that the vector $(\boldsymbol{Z}^T, \boldsymbol{U}^T)^T$ is Gaussian, and a complete proof is given in [24, Proposition 3.1]. □

For $k \in \{0, \dots, K\}$, the spatial prediction $Z^*(t_k, p)$ of the field $Z(t_k, \cdot)$ at any location $p \in \mathcal{M}_h$ is deduced from the conditional expectation $\mathbb{E}[\boldsymbol{Z}|\boldsymbol{U}]$ by leveraging the linearity of the (conditional) expectation, thus giving:

$$Z^*(t_k, p) = \mathbb{E}[Z(t_k, p)|\boldsymbol{U}] = \begin{pmatrix} \psi_1(p) \\ \vdots \\ \psi_N(p) \end{pmatrix}^T \mathbb{E}[\boldsymbol{Z}|\boldsymbol{U}]. \tag{19}$$

Time extrapolation at times t_k, $k > K$, can be tackled by similarly. Indeed, by taking the conditional expectation $\mathbb{E}[\ \cdot\ |\boldsymbol{U}]$ on both sides of the recursion (13) we get

$$\mathbb{E}[\boldsymbol{z}^{(k+1)}|\boldsymbol{U}] = (\sqrt{\boldsymbol{C}})^{-T}\boldsymbol{\Gamma}^{-1}(\sqrt{\boldsymbol{C}})^T\mathbb{E}[\boldsymbol{z}^{(k)}|\boldsymbol{U}], \quad k \geq K, \tag{20}$$

where we recall that $\boldsymbol{z}^{(k)}$ is the weight vector defining the solution Z at time t_k (cf. Proposition 4.2), and $\mathbb{E}[\boldsymbol{z}^{(K)}|\boldsymbol{U}]$ corresponds to the N last rows of $\mathbb{E}[\boldsymbol{Z}|\boldsymbol{U}]$. Then spatial predictions at any locations can one again be obtained using (19).

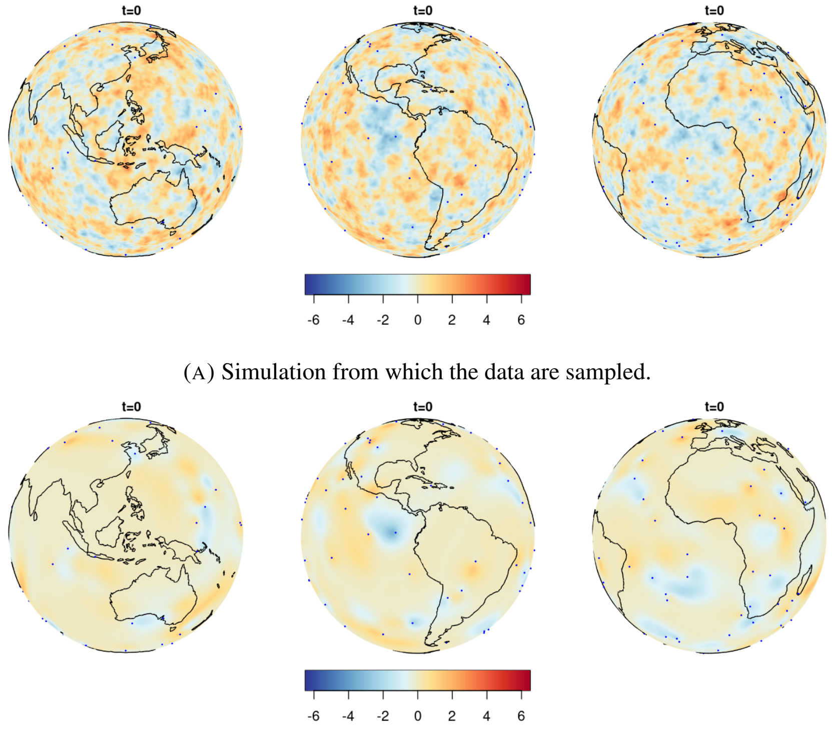
From now on, let us denote by $\mathcal{K}_{\boldsymbol{Q},s^2}$ the linear map given by

$$\mathcal{K}(\boldsymbol{v}\,;\,\boldsymbol{Q}, s^2) = s^{-2}(\boldsymbol{Q} + s^{-2}\boldsymbol{A}\boldsymbol{A}^T)^{-1}\boldsymbol{A}\boldsymbol{v}, \quad \boldsymbol{v} \in \mathbb{R}^{N_o}, \tag{21}$$

where $s > 0$ and $\boldsymbol{Q} \in \mathbb{R}^{(K+1)N \times (K+1)N}$ is a positive definite matrix. As seen in the proposition above, we have $\mathbb{E}[\boldsymbol{Z}|\boldsymbol{U}] = \mathcal{K}(\boldsymbol{U} - \boldsymbol{\eta}\boldsymbol{b}\,;\,\boldsymbol{Q}_{\boldsymbol{Z}}, \sigma^2)$. Therefore, efficient algorithms to evaluate $\mathcal{K}$ are fundamental to perform predictions. A first approach to compute (21) consists in factorizing the matrix $\boldsymbol{M} = (\boldsymbol{Q} + s^{-2}\boldsymbol{A}A^T)$ (using for instance a Cholesky decomposition), and then using the factorization to efficiently solve the linear system in (21). However, since the matrix $\boldsymbol{M}$ has size $N(K+1) \times N(K+1)$, building, storing and factorizing can become computationally prohibitive in some applications where either N or K (or both) are large.

An alternative approach to evaluate $\mathcal{K}$ consists in solving the linear system in (21) using a matrix-free iterative algorithm [30]. Such algorithms yield an approximate solution of the linear system using through an iterative process which requires at each iterations products between $\boldsymbol{M}$ and vectors. Such products can in turn be evaluated without having to explicitly build the matrix $\boldsymbol{M}$, but using instead the diagonal block structure of $\boldsymbol{A}$. In this setting, only the "spatial" matrices $\boldsymbol{\Gamma}$, $\widetilde{\boldsymbol{R}}$ and $\sqrt{\boldsymbol{C}}$ and $\boldsymbol{A}^{(k)}$ ($0 \leq k \leq K$), which are sparse and of size at most $N \times N$, are stored.

Note that linear systems of the form of (21) are classically encountered in regularized least-square problems, and are known to be ill-conditioned when s^2 is small. This can make the convergence of classical iterative algorithms (e.g Conjugate gradient) very slow. To circumvent this problem, appropriate preconditioning should be applied to the system. For instance, Clarotto et al. [9] propose to use the Gauss–Siedel preconditioner, which in our case takes the



(A) Simulation from which the data are sampled.

(B) Kriging predictor.

Figure 5.1: Spatio-temporal kriging predictor from data sampled from a simulation of the SPDE model and with a measurement noise of standard-deviation $\sigma = 0.1$. The blue points locate the places where the observations are taken.

form of lower-triangular block matrix. An alternative method consists in noting that the solution $\boldsymbol{x}$ of the linear system $(\boldsymbol{Q} + s^{-2}\boldsymbol{A}\boldsymbol{A}^T)\boldsymbol{x} = \boldsymbol{A}\boldsymbol{v}$ satisfies is (part of) the solution of the augmented system

$$\begin{pmatrix} \boldsymbol{Q} & \boldsymbol{A} \\ \boldsymbol{A}^T & -s^2\boldsymbol{I} \end{pmatrix} \begin{pmatrix} \boldsymbol{x} \\ \boldsymbol{y} \end{pmatrix} = \begin{pmatrix} \boldsymbol{v} \\ \boldsymbol{0} \end{pmatrix}, \tag{22}$$

which in turn is equivalent (after applying a block diagonal left and right preconditioners) to the system

$$\begin{pmatrix} \boldsymbol{I} & (\boldsymbol{L}_{\boldsymbol{Q}}^{-1}\boldsymbol{A}) \\ (\boldsymbol{L}_{\boldsymbol{Q}}^{-1}\boldsymbol{A})^T & -s^2\boldsymbol{I} \end{pmatrix} \begin{pmatrix} \widehat{\boldsymbol{x}} \\ \widehat{\boldsymbol{y}} \end{pmatrix} = \begin{pmatrix} \boldsymbol{L}_{\boldsymbol{Q}}^{-1}\widehat{\boldsymbol{v}} \\ \boldsymbol{0} \end{pmatrix}, \tag{23}$$

where $\boldsymbol{L}_{\boldsymbol{Q}}^{-1} = \boldsymbol{D}\big(\sqrt{\boldsymbol{C}}, \sqrt{\boldsymbol{C}}\big)\boldsymbol{L}(\boldsymbol{\Gamma}, \boldsymbol{I})^T\boldsymbol{D}\big(f_0^{-1}(\widetilde{\boldsymbol{R}}), f_{\delta t}^{-1}(\widetilde{\boldsymbol{R}})\big)$, $\widehat{\boldsymbol{x}} = \boldsymbol{L}_{\boldsymbol{Q}}^T\boldsymbol{x}$, and $\widehat{\boldsymbol{y}} = \boldsymbol{y}$. Systems of the form (22)-(23) are known as saddle-point systems and specific algorithms have been devised to tackle them, such as the TriMR algorithm [23]. In the numerical applications presented in this work, the evaluations of the map $\mathcal{K}$ are done using the Julia implementation of the TriMR algorithm in the Krylov.jl to solve the system (23).

As an illustration, we present in Figure 5.1 the results of a kriging prediction based on data sampled from a simulation of the solution to the advection-diffusion model. For the SPDE, we use the same model as the one in Figure 4.1. The observations are assumed to be at the same spatial locations at each time step, and a measurement noise of standard-deviation $\sigma = 0.1$ is considered. When comparing the kriging predictor to the actual simulation, we can notice that the predictor enforces the transport phenomena modeled by the SPDE.

Finally, note that the conditional expectation can also be used to generate conditional simulations at time steps t_k, $0 \leq k \leq K$ of the field Z, by leveraging the fact that the conditional variance does not depend explicitly on the conditioning data $\boldsymbol{U}$. This approach, presented in more details in [9, Section 3.3], is recalled below:

1. Compute a non-conditional simulation $\boldsymbol{Z}_{NC}$ by running the recursion in Proposition 4.2.
2. Generate new observations by computing $\boldsymbol{U}_{NC} = \boldsymbol{\eta}\boldsymbol{b} + \boldsymbol{A}^T\boldsymbol{Z}_{NC} + \sigma\boldsymbol{\varepsilon}_{NC}$.

3. Compute the residuals $\boldsymbol{r}_{NC} = \boldsymbol{Z}_{NC} - \mathbb{E}[\boldsymbol{Z}_{NC}|\boldsymbol{U}_{NC}]$.
4. Return the conditional simulation $\boldsymbol{Z}_C = \mathbb{E}[\boldsymbol{Z}|\boldsymbol{U}] + \boldsymbol{r}_{NC}$.

Conditional simulations at further time steps are then obtained using once again the recursion in Proposition 4.2.

5.2 A few words about inference

Let $\boldsymbol{\theta}$ be the vector containing the parameters of (8), and let $\boldsymbol{\nu} = (\boldsymbol{\theta}^T, \boldsymbol{b}^T, \sigma)^T$ be the vector containing all the parameters of the statistical model. The next proposition, proven in [9, Section 3.1] gives the likelihood of the observations.

Proposition 5.2. *The likelihood function of the vector of observations $\boldsymbol{U}$ is given by*

$$\mathcal{L}(\boldsymbol{\psi}) = -\frac{N_o}{2}\log 2\pi + \frac{1}{2}\log|\boldsymbol{Q}_{\boldsymbol{U}}(\boldsymbol{\psi})| - \frac{\sigma^{-2}}{2}\Big(\|\boldsymbol{U}-\boldsymbol{\eta}\boldsymbol{b}\|_2^2 - (\boldsymbol{U}-\boldsymbol{\eta}\boldsymbol{b})^T\boldsymbol{A}^T\mathcal{K}_{\boldsymbol{Q}_{\boldsymbol{Z}}(\boldsymbol{\theta}),\sigma^2}(\boldsymbol{U}-\boldsymbol{\eta}\boldsymbol{b})\Big), \tag{24}$$

where $\boldsymbol{Q}_{\boldsymbol{U}}(\boldsymbol{\psi}) = (\boldsymbol{A}^T\boldsymbol{Q}_{\boldsymbol{Z}}(\boldsymbol{\theta})^{-1}\boldsymbol{A} + \sigma^2\boldsymbol{I})^{-1}$, and

$$\log|\boldsymbol{Q}_{\boldsymbol{U}}(\boldsymbol{\psi})| = -N_o\log\sigma^2 + \log|\boldsymbol{Q}_{\boldsymbol{Z}}(\boldsymbol{\theta})| - \log|\boldsymbol{Q}_{\boldsymbol{Z}}(\boldsymbol{\theta}) + \sigma^{-2}\boldsymbol{A}\boldsymbol{A}^T|.$$

The evaluation of the log-likelihood (24) can be tackled using the approach described in [24, Section 4] and [9, Section 3.1]. Indeed, the map $\mathcal{K}$ can be evaluated using either a preconditioned Conjugate gradient algorithm or a saddle point algorithm (cf. Section 5.1). And the log-determinant terms can be evaluated using either Cholesky decompositions of the matrices (when their size allow it) or using matrix-free approaches based on Hutchinson trace estimators [14]. The maximization of the log-likelihood can then be tackled using gradient-based optimization (based on finite-difference approximations of the gradient) or the Nelder-Meade algorithm (which only requires evaluations of the cost function).

References

[1] H. Bakka, E. Krainski, D. Bolin, H. Rue, and F. Lindgren. The diffusion-based extension of the Matérn field to space-time. 2020.

[2] H. Bhatia, G. Norgard, V. Pascucci, and P.-T. Bremer. The helmholtz-hodge decomposition—a survey. *IEEE Transactions on visualization and computer graphics*, 19(8):1386–1404, 2012.

[3] A. Bonito, D. Guignard, and W. Lei. Numerical approximation of Gaussian random fields on closed surfaces. *arXiv preprint arXiv:2211.13739*, 2022.

[4] C.-E. Bréhier. A short introduction to stochastic pdes. 2014.

[5] A. N. Brooks and T. J. Hughes. Streamline upwind/petrov-galerkin formulations for convection dominated flows with particular emphasis on the incompressible navier-stokes equations. *Computer methods in applied mechanics and engineering*, 32(1-3):199–259, 1982.

[6] M. Cameletti, F. Lindgren, D. Simpson, and H. Rue. Spatio-temporal modeling of particulate matter concentration through the SPDE approach. *AStA Advances in Statistical Analysis*, 97:109–131, 2013.

[7] W. Chen, M. G. Genton, and Y. Sun. Space-time covariance structures and models. *Annual Review of Statistics and Its Application*, 8:191–215, 2021.

[8] J.-P. Chiles and P. Delfiner. *Geostatistics: Modeling Spatial Uncertainty*, volume 497. John Wiley & Sons, 2009.

[9] L. Clarotto, D. Allard, T. Romary, and N. Desassis. The spde approach for spatio-temporal datasets with advection and diffusion. *arXiv preprint arXiv:2208.14015*, 2022.

[10] R. Codina. On stabilized finite element methods for linear systems of convection–diffusion-reaction equations. *Computer Methods in Applied Mechanics and Engineering*, 188(1-3):61–82, 2000.

[11] S. Coveney, C. Corrado, C. H. Roney, R. D. Wilkinson, J. E. Oakley, F. Lindgren, S. E. Williams, M. D. O'Neill, S. A. Niederer, and R. H. Clayton. Probabilistic interpolation of uncertain local activation times on human atrial manifolds. *IEEE Transactions on Biomedical Engineering*, 67(1):99–109, 2019.

[12] G. Da Prato and J. Zabczyk. *Stochastic equations in infinite dimensions*. Cambridge university press, 2014.

[13] T. A. Davis. *Direct Methods for Sparse Linear Systems*, volume 2. SIAM, 2006.

[14] I. Han, D. Malioutov, and J. Shin. Large-scale log-determinant computation through stochastic chebyshev expansions. In *International Conference on Machine Learning*, pages 908–917. PMLR, 2015.

[15] L. Herrmann, K. Kirchner, and C. Schwab. Multilevel approximation of Gaussian random fields: fast simulation. *Mathematical Models and Methods in Applied Sciences*, 30(01):181–223, 2020.

[16] A. Lang and M. Pereira. Galerkin–Chebyshev approximation of Gaussian random fields on compact Riemannian manifolds. *arXiv preprint arXiv:2107.02667*, 2021.

[17] F. Lindgren, H. Rue, and J. Lindström. An explicit link between Gaussian fields and Gaussian markov random fields: the stochastic partial differential equation approach. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 73(4):423–498, 2011.

[18] F. Lindgren, D. Bolin, and H. Rue. The SPDE approach for Gaussian and non-Gaussian fields: 10 years and still running. *Spatial Statistics*, 50:100599, 2022.

[19] X. Liu, K. Yeo, and S. Lu. Statistical modeling for spatio-temporal data from stochastic convection-diffusion processes. *Journal of the American Statistical Association*, 117(539):1482–1499, 2022.

[20] A. F. Mejia, Y. Yue, D. Bolin, F. Lindgren, and M. A. Lindquist. A Bayesian general linear modeling approach to cortical surface fMRI data analysis. *Journal of the American Statistical Association*, 115(530):501–520, 2020.

[21] G. T. Mekuria, J. A. Rao, et al. Adaptive finite element method for steady convection-diffusion equation. *American Journal of Computational Mathematics*, 6(03):275, 2016.

[22] J. Molina and R. M. Slevinsky. A rapid and well-conditioned algorithm for the helmholtz–hodge decomposition of vector fields on the sphere. *arXiv preprint arXiv:1809.04555*, 2018.

[23] A. Montoison and D. Orban. Tricg and trimr: Two iterative methods for symmetric quasi-definite systems. *SIAM Journal on Scientific Computing*, 43(4):A2502–A2525, 2021.

[24] M. Pereira, N. Desassis, and D. Allard. Geostatistics for large datasets on riemannian manifolds: A matrix-free approach. *Journal of Data Science*, 20(4):512–532, 2022. ISSN 1680-743X. doi:10.6339/22-JDS1075.

[25] K. Poelke and K. Polthier. Boundary-aware hodge decompositions for piecewise constant vector fields. *Computer-Aided Design*, 78:126–136, 2016.

[26] E. Porcu, A. Alegria, and R. Furrer. Modeling temporally evolving and spatially globally dependent data. *International Statistical Review*, 86(2):344–377, 2018.

[27] E. Porcu, R. Furrer, and D. Nychka. 30 years of space–time covariance functions. *Wiley Interdisciplinary Reviews: Computational Statistics*, 13(2):e1512, 2021.

[28] N. A. Rayner, R. Auchmann, J. Bessembinder, S. Brönnimann, Y. Brugnara, F. Capponi, L. Carrea, E. M. Dodd, D. Ghent, E. Good, et al. The EUSTACE project: delivering global, daily information on surface air temperature. *Bulletin of the American Meteorological Society*, 101(11):E1924–E1947, 2020.

[29] H. Rue and L. Held. *Gaussian Markov Random Fields: Theory and Applications*. CRC press, 2005.

[30] Y. Saad. *Iterative methods for sparse linear systems*. SIAM, 2003.

[31] S. Särkkä and A. Solin. *Applied stochastic differential equations*, volume 10. Cambridge University Press, 2019.

[32] F. Sigrist, H. R. Künsch, and W. A. Stahel. Stochastic partial differential equation based modelling of large space-time data sets. *Journal of the Royal Statistical Society: Series B: Statistical Methodology*, pages 3–33, 2015.

[33] H. Wackernagel. *Multivariate Geostatistics: An Introduction with Applications*. Springer Science & Business Media, 2003.

[34] P. Whittle. On stationary processes in the plane. *Biometrika*, pages 434–449, 1954.

A Mathematical tools

A.1 Functions of the Laplacian and colored noise

Let $\{\lambda_k\}_{k\in\mathbb{N}}$ denote the set of eigenvalues of the Laplace– Beltrami operator $-\Delta_{\mathcal{M}}$ on $(\mathcal{M}, g)$, and $\{e_k\}_{k\in\mathbb{N}}$ denote the associated eigenfunctions. In particular, $\{e_k\}_{k\in\mathbb{N}}$ form a basis of the space $L^2(\mathcal{M})$. For $f:\mathbb{R}_+\rightarrow\mathbb{R}$, we define $\mathcal{D}_f\subset L^2(\mathcal{M})$ as

$$\mathcal{D}_f = \{\phi\in L^2(\mathcal{M}) : \sum_{j\in\mathbb{N}} f(\lambda_j)^2\langle\phi, e_j\rangle^2 < \infty\}.$$

Note that if f is bounded, then $\mathcal{D}_f = L^2(\mathcal{M})$. Then, the *function of the Laplacian* $f(-\Delta_{\mathcal{M}})$ is the operator $f(-\Delta_{\mathcal{M}}) : \mathcal{D}_f \rightarrow L^2(\mathcal{M})$ defined by

$$f(-\Delta_{\mathcal{M}})\phi = \sum_{j\in\mathbb{N}} f(\lambda_j)\langle\phi, e_j\rangle e_j, \quad \phi\in L^2(\mathcal{M}).$$

Let then $f_S : \mathbb{R}_+ \to \mathbb{R}$ be a bounded function, and $\{w_j\}_{j\in\mathbb{N}}$ be a sequence of independent standard Gaussian variable. We call *colored noise* the linear functional $f_S(-\Delta_{\mathcal{M}})\mathcal{W}_S$ defined by

$$f_S(-\Delta_{\mathcal{M}})\mathcal{W}_S : \phi \in H \mapsto \langle f_S(-\Delta_{\mathcal{M}})\mathcal{W}_S, \phi\rangle = \sum_{j\in\mathbb{N}} w_j f_S(\lambda_j)\langle \phi, e_j\rangle.$$

Note that for any $\phi \in H$, the series $\langle f_S(-\Delta_{\mathcal{M}})\mathcal{W}_S, \phi\rangle$ converges in quadratic mean since $\mathbb{E}[\langle f_S(-\Delta_{\mathcal{M}})\mathcal{W}_S, \phi\rangle] = 0$, and by independence of the variables w_j,

$$\mathbb{E}[|\langle f_S(-\Delta_{\mathcal{M}})\mathcal{W}_S, \phi\rangle|^2] = \sum_{j\in\mathbb{N}} |f_S(\lambda_j)|^2 |\langle \phi, e_j\rangle|^2 \leq (\sup_{\mathbb{R}_+} |f_S|)^2 \|\phi\|_S^2 < \infty.$$

Besides, using the same arguments, we have for any $\phi_1, \phi_2 \in H$,

$$\operatorname{Cov}[\langle f_S(-\Delta_{\mathcal{M}})\mathcal{W}_S, \phi_1\rangle, \langle f_S(-\Delta_{\mathcal{M}})\mathcal{W}_S, \phi_2\rangle] = \sum_{j\in\mathbb{N}} |f_S(\lambda_j)|^2 \langle \phi_1, e_j\rangle\langle \phi_2, e_j\rangle = \langle f_S(-\Delta_{\mathcal{M}})\phi_1, f_S(-\Delta_{\mathcal{M}})\phi_2\rangle,$$

Hence, in the case where $f_S(\lambda) = 1$ for any $\lambda \geq 0$, $f_S(-\Delta_{\mathcal{M}})\mathcal{W}_S = \mathcal{W}_S$ corresponds to the definition of the spatial Gaussian white noise on $L^2(\mathcal{M})$. Also, whenever f_S satisfies $f_S(\lambda) = \mathcal{O}_{\lambda\to\infty}(\lambda^{-\alpha})$ with $\alpha > d/4$, $f_S(-\Delta_{\mathcal{M}})\mathcal{W}_S$ can be identified with a square-integrable H-valued random variable, and decomposed as [16, Proposition 2.7]:

$$f_S(-\Delta_{\mathcal{M}})\mathcal{W}_S = \sum_{j\in\mathbb{N}} w_j f_S(\lambda_j) e_j.$$

For instance, if $f_S(\lambda) = |\kappa^2 + \lambda|^{-\alpha}$, then $\mathcal{Y}_S = f_S(-\Delta_{\mathcal{M}})\mathcal{W}_S$ can be seen as a solution of the Whittle-Matérn SPDE

$$(\kappa^2 - \Delta_{\mathcal{M}})^\alpha \mathcal{Y}_S = \mathcal{W}_S \tag{25}$$

and can thus be seen as a Whittle-Matérn random field on $\mathcal{M}$.

A.2 Galerkin approximation

For $N \in \mathbb{N}$, let $V_N = \operatorname{span}\{\psi_k : 1 \leq k \leq N\}$ where $\psi_1, \dots, \psi_N \in H^1(\mathcal{M}_h)$ are linearly independent functions. Let $\boldsymbol{C}$ and $\boldsymbol{R}$ be the matrices whose entries are respectively given by

$$C_{ij} = \langle \psi_i, \psi_j\rangle, \quad R_{ij} = \langle \nabla\psi_i, \nabla\psi_j\rangle. \tag{26}$$

The Galerkin approximation of $-\Delta_{\mathcal{M}}$ over V_N is the linear operator $-\Delta_N : V_N \to V_N$ satisfying, for any $\phi, v \in V_N$,

$$\langle -\Delta_N \phi, v\rangle = \langle \nabla\phi, \nabla v\rangle.$$

As defined, $-\Delta_N$ is a symmetric endomorphism, and as such is diagonalizable. Let $\{\lambda_k^{(N)}\}_{1\leq k\leq N}$ denote its eigenvalues, and let $\{e_k^{(N)}\}_{1\leq k\leq N}$ be a set of associated eigenfunctions forming an orthonormal basis of V_N. Note that, following [16, Corollary 3.2], $\{\lambda_k^{(N)}\}_{1\leq k\leq N}$ are also the eigenvalues of the matrix $\widetilde{\boldsymbol{R}}$. Besides, the map

$$E : \boldsymbol{v} \in \mathbb{R}^N \mapsto \sum_{k=1}^N \left[(\sqrt{\boldsymbol{C}})^{-T}\boldsymbol{v}\right]_k \psi_k \in V_N \tag{27}$$

is an isomorphism that maps the eigenvectors of $\widetilde{\boldsymbol{R}}$ to the eigenfunctions of $-\Delta_N$, and an isometry between $(\mathbb{R}^N, \|\cdot\|_2)$ and $(V_N, \|\cdot\|_S)$.

A.3 Matrix functions

Let $\boldsymbol{S} \in \mathbb{R}^{N\times N}$ be a real symmetric matrix and let $f : \mathbb{R} \to \mathbb{R}$. In particular let us denote by $\lambda_1, \dots, \lambda_N$ the eigenvalues of $\boldsymbol{S}$ and let $\boldsymbol{V} \in \mathbb{R}^{N\times N}$ be an orthogonal matrix such that

$$\boldsymbol{S} = \boldsymbol{V} \begin{pmatrix} \lambda_1 & & \\ & \ddots & \\ & & \lambda_N \end{pmatrix} \boldsymbol{V}^T. \tag{28}$$

Then, the matrix function $f(\boldsymbol{S}) \in \mathbb{R}^{N\times N}$ is the matrix defined by

$$f(\boldsymbol{S}) = \boldsymbol{V} \begin{pmatrix} f(\lambda_1) & & \\ & \ddots & \\ & & f(\lambda_N) \end{pmatrix} \boldsymbol{V}^T.$$

Note in particular that this definition is independent of the choice of matrix $\boldsymbol{V}$ in (28), and that when f is a polynomial, $f(\boldsymbol{S})$ coincides with the usual notion of matrix polynomial.

B Algorithms

We expose in this section a few algorithms that are necessary to perform matrix-free predictions.

Algorithm 1 Matrix-vector product by $\boldsymbol{Q_Z}$

```
    **Depends on:** Matrices $(\sqrt{\boldsymbol{C}}), \widetilde{\boldsymbol{R}}, \boldsymbol{\Gamma}$ in (18).
    **Input:** Vector $\boldsymbol{X} = ((\boldsymbol{x}^{(0)})^T, \dots, (\boldsymbol{x}^{(K)})^T)^T \in \mathbb{R}^{(K+1)N}$.
    **Output:** Vector $\boldsymbol{Y} = ((\boldsymbol{y}^{(0)})^T, \dots, (\boldsymbol{y}^{(K)})^T)^T = \boldsymbol{Q_Z}\boldsymbol{X}$.

for $k = 0$ to $K$ do
    Initialize $\boldsymbol{y}^{(k)} = (\sqrt{\boldsymbol{C}})^T \boldsymbol{x}^{(k)}$
    Initialize $\boldsymbol{z}^{(k)} = \boldsymbol{0}$
end for
Set $\boldsymbol{z}^{(0)} \leftarrow \boldsymbol{y}^{(0)}$
for $k = 1$ to $K$ do
    Set $\boldsymbol{z}^{(k)} \leftarrow \boldsymbol{\Gamma}\boldsymbol{y}^{(k)} - \boldsymbol{y}^{(k-1)}$
end for
Set $\boldsymbol{z}^{(0)} \leftarrow f_0^{-2}(\widetilde{\boldsymbol{R}})\boldsymbol{z}^{(0)}$
for $k = 1$ to $K$ do
    Set $\boldsymbol{z}^{(k)} \leftarrow f_{\delta t}^{-2}(\widetilde{\boldsymbol{R}})\boldsymbol{z}^{(k)}$
end for
Set $\boldsymbol{y}^{(K)} \leftarrow \boldsymbol{\Gamma}^T\boldsymbol{z}^{(K)}$
for $k = K-1$ to $1$ do
    Set $\boldsymbol{y}^{(k)} \leftarrow \boldsymbol{\Gamma}^T\boldsymbol{z}^{(k)} - \boldsymbol{z}^{(k+1)}$
end for
Set $\boldsymbol{y}^{(0)} \leftarrow \boldsymbol{z}^{(0)} - \boldsymbol{z}^{(1)}$
for $k = 0$ to $K$ do
    Set $\boldsymbol{y}^{(k)} \leftarrow (\sqrt{\boldsymbol{C}})\boldsymbol{y}^{(k)}$
end for
return $\boldsymbol{Y} = ((\boldsymbol{y}^{(0)})^T, \dots, (\boldsymbol{y}^{(K)})^T)^T$.
```

Algorithm 2 Solve a linear system defined by $\boldsymbol{L}(\boldsymbol{\Theta}_1, \boldsymbol{\Theta}_2)$

Depends on: Matrices $\boldsymbol{\Theta}_1, \boldsymbol{\Theta}_2$ in (17).
Input: Vector $\boldsymbol{X} = ((\boldsymbol{x}^{(0)})^T, \dots, (\boldsymbol{x}^{(K)})^T)^T \in \mathbb{R}^{(K+1)N}$.
Output: Vector $\boldsymbol{Y} = ((\boldsymbol{y}^{(0)})^T, \dots, (\boldsymbol{y}^{(K)})^T)^T = \boldsymbol{L}(\boldsymbol{\Theta}_1, \boldsymbol{\Theta}_2)^{-1}\boldsymbol{X}$.

```
for $k = 0$ to $K$ do
    Initialize $\boldsymbol{y}^{(k)} = \boldsymbol{0}$
end for
Set $\boldsymbol{y}^{(0)} \leftarrow \boldsymbol{x}^{(0)}$
for $k = 1$ to $K$ do
    Set $\boldsymbol{y}^{(k)} \leftarrow \boldsymbol{\Theta}_1^{-1}\big(\boldsymbol{\Theta}_2\boldsymbol{y}^{(k-1)} + \boldsymbol{x}^{(k)}\big)$
end for
return $\boldsymbol{Y} = ((\boldsymbol{y}^{(0)})^T, \dots, (\boldsymbol{y}^{(K)})^T)^T$.
```

Algorithm 3 Solve a linear system defined by $\boldsymbol{L}(\boldsymbol{\Theta}_1, \boldsymbol{\Theta}_2)^T$

Depends on: Matrices $\boldsymbol{\Theta}_1, \boldsymbol{\Theta}_2$ in (17).
Input: Vector $\boldsymbol{X} = ((\boldsymbol{x}^{(0)})^T, \dots, (\boldsymbol{x}^{(K)})^T)^T \in \mathbb{R}^{(K+1)N}$.
Output: Vector $\boldsymbol{Y} = ((\boldsymbol{y}^{(0)})^T, \dots, (\boldsymbol{y}^{(K)})^T)^T = \boldsymbol{L}(\boldsymbol{\Theta}_1, \boldsymbol{\Theta}_2)^{-T}\boldsymbol{X}$.

```
for $k = 0$ to $K$ do
    Initialize $\boldsymbol{y}^{(k)} = \boldsymbol{0}$
end for
Set $\boldsymbol{y}^{(K)} \leftarrow \boldsymbol{\Theta}_1^{-T}\boldsymbol{x}^{(K)}$
for $k = K - 1$ to $1$ do
    Set $\boldsymbol{y}^{(k)} = \boldsymbol{\Theta}_1^{-T}\big(\boldsymbol{\Theta}_2^T\boldsymbol{y}^{(k+1)} + \boldsymbol{x}^{(k)}\big)$
end for
Set $\boldsymbol{y}^{(0)} \leftarrow \boldsymbol{\Theta}_2^T\boldsymbol{y}^{(1)} + \boldsymbol{x}^{(0)}$
return $\boldsymbol{Y} = ((\boldsymbol{y}^{(0)})^T, \dots, (\boldsymbol{y}^{(K)})^T)^T$.
```

Algorithm 4 Solve a linear system defined by $\boldsymbol{Q}_{\boldsymbol{Z}}$

Depends on: Matrices $(\sqrt{\boldsymbol{C}}), \widetilde{\boldsymbol{R}}, \boldsymbol{\Gamma}$ in (18).
Input: Vector $\boldsymbol{X} = ((\boldsymbol{x}^{(0)})^T, \dots, (\boldsymbol{x}^{(K)})^T)^T \in \mathbb{R}^{(K+1)N}$.
Output: Vector $\boldsymbol{Y} = ((\boldsymbol{y}^{(0)})^T, \dots, (\boldsymbol{y}^{(K)})^T)^T = \boldsymbol{Q}_{\boldsymbol{Z}}^{-1}\boldsymbol{X}$.

```
for $k = 0$ to $K$ do
    Initialize $\boldsymbol{y}^{(k)} = (\sqrt{\boldsymbol{C}})^{-1}\boldsymbol{x}^{(k)}$
end for
Set $\boldsymbol{Y} \leftarrow \boldsymbol{L}(\boldsymbol{\Gamma}, \boldsymbol{I})^{-T}\boldsymbol{Y}$ using Algorithm 3
Set $\boldsymbol{y}^{(0)} \leftarrow f_0^2(\widetilde{\boldsymbol{R}})\boldsymbol{y}^{(0)}$
for $k = 1$ to $K$ do
    Set $\boldsymbol{y}^{(k)} \leftarrow f_{\delta t}^2(\widetilde{\boldsymbol{R}})\boldsymbol{y}^{(k)}$
end for
Set $\boldsymbol{Y} \leftarrow \boldsymbol{L}(\boldsymbol{\Gamma}, \boldsymbol{I})^{-1}\boldsymbol{Y}$ using Algorithm 2
for $k = 0$ to $K$ do
    Set $\boldsymbol{y}^{(k)} \leftarrow (\sqrt{\boldsymbol{C}})^{-T}\boldsymbol{x}^{(k)}$
end for
return $\boldsymbol{Y} = ((\boldsymbol{y}^{(0)})^T, \dots, (\boldsymbol{y}^{(K)})^T)^T$.
```

C Diffusion-only case

In the absence of advection term (i.e. when $\gamma = 0$), an alternative time discretization method for the SPDE (8) can be proposed. Indeed, note that in this case, SPDE (8) takes the form

$$\frac{\partial Z}{\partial t}(t,\cdot) + \frac{1}{c}P(-\Delta_N)Z(t,\cdot) = \frac{\tau}{\sqrt{c}}\mathcal{W}_T \otimes Y_S, \quad t \in [0,T], \tag{29}$$

Let us write for any $j \in \{1,\dots,N\}$ and $t \in [0,T]$, $\xi_j(t) = \langle Z(t,\cdot), e_j^{(N)}\rangle$. By testing (29) against $e_j^{(N)}$, and following the link between the stochastic forcing term $\mathcal{W}_T \otimes Y_S$ and cylindrical Wiener processes, we get that ξ_j satisfies the following SDE:

$$\mathrm{d}\xi_j(t) = -P\big(\lambda_j^{(N)}\big)\xi_j(t)\mathrm{d}t + \frac{\tau}{\sqrt{c}}f_S\big(\lambda_j^{(N)}\big)\mathrm{d}\beta_j(t), \quad 1 \le j \le N, \quad t \in [0,T], \tag{30}$$

where $\{\beta_j\}_{j\in\mathbb{N}}$ denotes a sequence of independent Brownian motions. The SDE (30) models an Ornstein–Uhlenbeck process, whose analytical solution is known [31]. In particular, ξ_j is a Gaussian process and for any $t_0 \ge 0$ and $h > 0$, the conditional distribution $\pi(\xi_j(t_0+h)|x_j(t_0))$ of $\xi_j(t_0+h)$ given $\xi_j(t_0)$ is Gaussian and given by

$$\pi(\xi_j(t_0+h)|\xi_j(t_0)) = \mathcal{N}\Big(m_h\big(\lambda_j^{(N)}\big)\xi_j(t_0), \sigma_h\big(\lambda_j^{(N)}\big)^2\Big),$$

where m_h and σ_h are the functions defined by

$$m_h(\lambda) = e^{-hP(\lambda)}, \quad \lambda \ge 0,$$

and

$$\sigma_h(\lambda) = \frac{\tau}{\sqrt{c}}\frac{f_S(\lambda)}{\sqrt{2P(\lambda)}}\sqrt{1-e^{-2hP(\lambda)}}, \quad \lambda \ge 0.$$

For $t \in [0,T]$, let $\boldsymbol{\xi}(t) = (\xi_1(t),\dots,\xi_N(t))^T$ be the vector containing the N Ornstein–Uhlenbeck processes. Using the fact that the entries of $\boldsymbol{\xi}$ are independent (since the Brownian motions β_j are independent), we have

$$\pi(\boldsymbol{\xi}(t_0+h)|\boldsymbol{\xi}(t_0)) = \mathcal{N}\Big(m_h\big(\boldsymbol{\Lambda}^{(N)}\big)\boldsymbol{\xi}(t_0),\ \sigma_h^2\big(\boldsymbol{\Lambda}^{(N)}\big)\Big).$$

In turn, we introduce for $t \in [0,T]$, the vector $\boldsymbol{z}(t) = (z_1(t),\dots,z_N(t))^T$ such that

$$Z(t,\cdot) = \sum_{j=1}^{N} z_j(t)\psi_j \tag{31}$$

and let $\boldsymbol{x}(t) = (\sqrt{\boldsymbol{C}})^T\boldsymbol{z}(t)$. Note that using the same arguments as the ones used in proof of Proposition 4.2, we can deduce the relation

$$\boldsymbol{x}(t) = (\sqrt{\boldsymbol{C}})^T\boldsymbol{z}(t) = \boldsymbol{V}\boldsymbol{\xi}(t), \quad t \in [0,T].$$

Hence, we have

$$\pi(\boldsymbol{x}(t_0+h)|\boldsymbol{x}(t_0)) = \mathcal{N}\Big(\boldsymbol{V}m_h\big(\boldsymbol{\Lambda}^{(N)}\big)\boldsymbol{V}^T\boldsymbol{x}(t_0), \boldsymbol{V}\sigma_h^2\big(\boldsymbol{\Lambda}^{(N)}\big)\boldsymbol{V}^T\Big).$$

which gives, by definition of the matrix functions,

$$\pi(\boldsymbol{x}(t_0+h)|\boldsymbol{x}(t_0)) = \mathcal{N}\Big(m_h\big(\widetilde{\boldsymbol{R}}\big)\boldsymbol{x}(t_0),\ \sigma_h^2\big(\widetilde{\boldsymbol{R}}\big)\Big). \tag{32}$$

This last property can be used to derive the joint distribution of observations of the vector $\boldsymbol{x}(t)$ (and therefore of the vector $\boldsymbol{z}(t)$) at irregular time steps, and for instance sample them. Indeed, let $0 \le t_0 \le \dots \le t_K = T$, then using the Markov property of the process $\boldsymbol{x}(t)$ (inherited from the process $\boldsymbol{\xi}(t)$), we have

$$\pi(\boldsymbol{x}(t_0),\dots,\boldsymbol{x}(t_K)) = \pi(\boldsymbol{x}(t_K)|\boldsymbol{x}(t_{K-1}))\cdots\pi(\boldsymbol{x}(t_1)|\boldsymbol{x}(t_0))\pi(\boldsymbol{x}(t_0)).$$

This decomposition implies that (jointly) sampling the vectors $(\boldsymbol{x}(t_0),\dots,\boldsymbol{x}(t_K))$ can be done by first sampling $\boldsymbol{x}(t_0)$ and then, for $k \ge 0$, by drawing $\boldsymbol{x}(t_{k+1})$ from the distribution $\pi(\boldsymbol{x}(t_{k+1})|\boldsymbol{x}(t_k))$ given in (32). hence the following proposition.

Proposition C.1. *Let $K \in \mathbb{N}$, and $0 \leq t_0 \leq \cdots \leq t_K = T$. For $k \in \{0, \dots, K\}$, let $\boldsymbol{z}(t_k)$ be the vector defining the solution of the diffusion* (29) *at time t_k as in* (31) *and let $\delta t_k = t_{k+1} - t_k$.*

Let $\boldsymbol{x}(t_0) = (\sqrt{\boldsymbol{C}})^T \boldsymbol{z}(t_0)$. Then we have the following recursion for $k \geq 0$,

$$\begin{cases} \boldsymbol{x}(t_{k+1}) = m_{\delta t_k}(\widetilde{\boldsymbol{R}})\boldsymbol{x}(t_k) + \sigma_{\delta t_k}(\widetilde{\boldsymbol{R}})\boldsymbol{w}^{(k+1)}, \\ \boldsymbol{z}(t_{k+1}) = (\sqrt{\boldsymbol{C}})^{-T}\boldsymbol{x}(t_{k+1}), \end{cases} \tag{33}$$

where $\{\boldsymbol{w}^{(k)}\}_{1 \leq k \leq K}$ is a sequence of independent centered Gaussian vectors with covariance matrix $\boldsymbol{I}$.

Contrary to the implicit Euler scheme introduced in Proposition 4.2, the present scheme is exact since it results from the exact solution (in distribution) to the Ornstein–Uhlenbeck processes that define the SPDE solution. Once again, the precision matrix of the vectors can be derived, using the same approach as for Proposition 4.4.

Proposition C.2. *Let us assume that the initial condition of SPDE* (8) *can be expressed as*

$$Z(t_0, \cdot) = f_0(-\Delta_N)Y_N,$$

for some function $f_0 : \mathbb{R}_+ \to \mathbb{R}$ that is bounded takes positive values. Let then $\boldsymbol{X} = (\boldsymbol{x}(t_0)^T, \dots, \boldsymbol{x}(t_K)^T)^T$ be the vector obtained by concatenating the vectors $\boldsymbol{x}(t_0), \dots, \boldsymbol{x}(t_K)$ defined in Proposition C.1, and similarly, let $\boldsymbol{Z} = (\boldsymbol{z}(t_0)^T, \dots, \boldsymbol{z}(t_K)^T)^T$.

Let $\boldsymbol{Q}_{\boldsymbol{X}}$ (resp. $\boldsymbol{Q}_{\boldsymbol{Z}}$) the precision matrix of $\boldsymbol{X}$ (resp. $\boldsymbol{Z}$). Then we have the following relations

$$\begin{cases} \boldsymbol{Q}_{\boldsymbol{X}} = \boldsymbol{L}_m^T \, \boldsymbol{D}_\sigma \, \boldsymbol{L}_m, \\ \boldsymbol{Q}_{\boldsymbol{Z}} = \boldsymbol{D}\big(\sqrt{\boldsymbol{C}}, \sqrt{\boldsymbol{C}}\big) \, \boldsymbol{Q}_{\boldsymbol{X}} \, \boldsymbol{D}\big((\sqrt{\boldsymbol{C}})^T, (\sqrt{\boldsymbol{C}})^T\big), \end{cases}$$

where

$$\boldsymbol{L}_m = \begin{pmatrix} \boldsymbol{I} & & & \\ -m_{\delta t_0}(\widetilde{\boldsymbol{R}}) & \boldsymbol{I} & & \\ & \ddots & \ddots & \\ & & -m_{\delta t_{K-1}}(\widetilde{\boldsymbol{R}}) & \boldsymbol{I} \end{pmatrix}, \quad \textit{and} \quad \boldsymbol{D}_\sigma = \begin{pmatrix} f_0^{-2}(\widetilde{\boldsymbol{R}}) & & & \\ & \sigma_{\delta t_0}^{-2}(\widetilde{\boldsymbol{R}}) & & \\ & & \ddots & \\ & & & \sigma_{\delta t_{K-1}}^{-2}(\widetilde{\boldsymbol{R}}) \end{pmatrix}.$$

Note that once again, products with vectors, solving linear systems and computing the log-determinant can be done without requiring to build the matrix $\boldsymbol{Q}_{\boldsymbol{Z}}$. Indeed, the log-determinants are now given by

$$\begin{cases} \log|\boldsymbol{Q}_{\boldsymbol{X}}| = \log|f_0^{-2}(\widetilde{\boldsymbol{R}})| + \sum_{k=0}^{K-1} \log|\sigma_{\delta t_k}^{-2}(\widetilde{\boldsymbol{R}})| = \log|f_0^{2}(\widetilde{\boldsymbol{R}})| + \sum_{k=0}^{K-1} \log|\sigma_{\delta t_k}^{-2}(\widetilde{\boldsymbol{R}})|, \\ \log|\boldsymbol{Q}_{\boldsymbol{Z}}| = 2(K+1)\log|\sqrt{\boldsymbol{C}}| + \log|\boldsymbol{Q}_{\boldsymbol{X}}|, \end{cases}$$

and Algorithms 1 and 4 used earlier to compute matrix-vector products and to solve linear systems can be straightforwardly adapted to the new expression of $\boldsymbol{Q}_{\boldsymbol{Z}}$.