Journal of the American Statistical Association

ISSN: (Print) (Online) Journal homepage: https://www.tandfonline.com/loi/uasa20

Highly Scalable Bayesian Geostatistical Modeling via Meshed Gaussian Processes on Partitioned Domains

Michele Peruzzi , Sudipto Banerjee & Andrew O. Finley

To cite this article: Michele Peruzzi , Sudipto Banerjee & Andrew O. Finley (2020): Highly Scalable Bayesian Geostatistical Modeling via Meshed Gaussian Processes on Partitioned Domains, Journal of the American Statistical Association, DOI: 10.1080/01621459.2020.1833889

To link to this article: https://doi.org/10.1080/01621459.2020.1833889

JOURNAL OF THE AMERICAN STATISTICAL ASSOCIATION
2020, VOL. 00, NO. 0, 1–14: Theory and Methods
https://doi.org/10.1080/01621459.2020.1833889

Highly Scalable Bayesian Geostatistical Modeling via Meshed Gaussian Processes on Partitioned Domains

Michele Peruzzi[a,b], Sudipto Banerjee[c], and Andrew O. Finley[a]

[a]Department of Forestry, Michigan State University, East Lansing, MI; [b]Department of Statistical Science, Duke University, Durham, NC; [c]Department of Biostatistics, UCLA Fielding School of Public Health, Los Angeles, CA

ABSTRACT

We introduce a class of scalable Bayesian hierarchical models for the analysis of massive geostatistical datasets. The underlying idea combines ideas on high-dimensional geostatistics by partitioning the spatial domain and modeling the regions in the partition using a sparsity-inducing directed acyclic graph (DAG). We extend the model over the DAG to a well-defined spatial process, which we call the meshed Gaussian process (MGP). A major contribution is the development of an MGPs on tessellated domains, accompanied by a Gibbs sampler for the efficient recovery of spatial random effects. In particular, the cubic MGP (Q-MGP) can harness high-performance computing resources by executing all large-scale operations in parallel within the Gibbs sampler, improving mixing and computing time compared to sequential updating schemes. Unlike some existing models for large spatial data, a Q-MGP facilitates massive caching of expensive matrix operations, making it particularly apt in dealing with spatiotemporal remote-sensing data. We compare Q-MGPs with large synthetic and real world data against state-of-the-art methods. We also illustrate using Normalized Difference Vegetation Index data from the Serengeti park region to recover latent multivariate spatiotemporal random effects at millions of locations. The source code is available at *github.com/mkln/meshgp*. Supplementary materials for this article are available online.

ARTICLE HISTORY
Received March 2020
Accepted October 2020

KEYWORDS
Bayesian; Domain partitioning; Graphical models; Large *n*; Sparsity; Spatial

1. Introduction

Collecting large quantities of spatial and spatiotemporal data is now commonplace in many fields. In ecology and forestry, massive datasets are collected using satellite imaging and other remote sensing instruments such as LiDAR that periodically record high-resolution images. Unfortunately, clouds frequently obstruct the view resulting in large regions with missing information. Figure 1 shows this phenomenon in Normalized Difference Vegetation Index (NDVI) data from the Serengeti region. Filling such gaps in the data is an important goal as is quantifying uncertainty in predictions. This goal is achieved through stochastic modeling of the underlying phenomenon, which involves the specification of a spatial or spatiotemporal process characterizing dependence from a finite realization. Gaussian processes (GPs) are a customary choice to characterize spatial dependence, but their implementation is notoriously burdened by their $O(n^3)$ computational complexity. Consequently, intense research has been devoted in recent years to developing scalable models for large spatial datasets—see detailed reviews by Sun, Li, and Genton (2011) and Banerjee (2017).

Computational complexity can be reduced by considering low-rank models; among these, knot-based methods motivated by "kriging" ideas enjoy some optimality properties but oversmooth the estimates of spatial random effects unless the number of knots is large, and require corrections to avoid overestimation of the nugget (Banerjee et al. 2008; Cressie and Johannesson 2008; Banerjee et al. 2010; Guhaniyogi et al. 2011; Finley, Banerjee, and Gelfand 2012). Other methods reduce the computational burden by introducing sparsity in the covariance matrix; strategies include tapering (Furrer, Genton, and Nychka 2006; Kaufman, Schervish, and Nychka 2008) or partitioning of the spatial domain into regions with a typical assumption of independence across regions (Sang and Huang 2012; Stein 2014). These can be improved by considering a recursive partitioning scheme, resulting in a multi-resolution approximation (MRA; Katzfuss 2017). Other assumptions on conditional independence assumptions also have a good track record in terms of scalability to large spatial datasets: Gaussian random Markov random fields (GMRF; Rue and Held 2005), composite likelihood methods (Eidsvik et al. 2014), and neighbor-based likelihood approximations (Vecchia 1988) belong to this family.

The recent literature has witnessed substantial activity surrounding the so called Vecchia approximation (Vecchia 1988). This approximation can be regarded as a special case of the GMRF approximations with a simplified neighborhood structure motivated from a directed acyclic graphical (DAG) representation of a GP likelihood. Extensions leading to well-defined spatial processes to accommodate inference at arbitrary locations by extending the DAG representation to the entire domain include nearest neighbor Gaussian processes (NNGPs; Datta,

CONTACT Sudipto Banerjee *sudipto@ucla.edu* Department of Biostatistics, UCLA Fielding School of Public Health, 650 Charles E. Young Drive South, Los Angeles, CA 90095.

Supplementary materials for this article are available online. Please go to *www.tandfonline.com/r/JASA*.

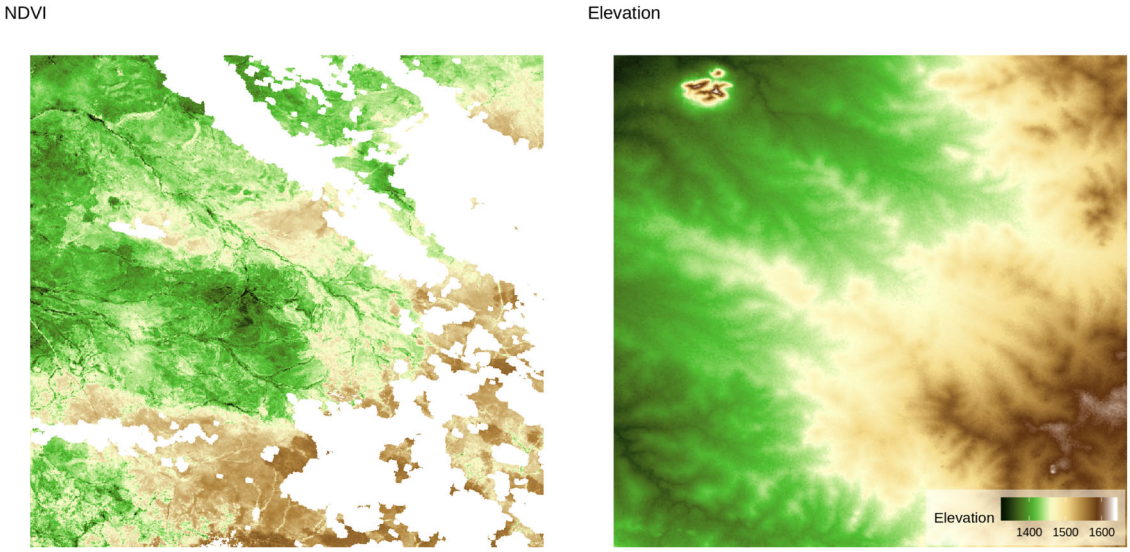

Figure 1. Left: NDVI in the Serengeti region on 2016-12-17. White areas correspond to missing data due to cloud cover. Right: Elevation data for the same region.

Banerjee, Finley, Gelfand, et al. 2016; Datta, Banerjee, Finley, Hamm, et al. 2016) and further generalizations by constructing DAGs over the augmented space of outcomes and spatial effects (Katzfuss and Guinness 2017). These approaches render computational scalability by introducing sparsity in the precision matrix. The DAG relies upon a specific topological ordering of the locations, which also determine the construction of neighborhood sets, and certain orderings tend to deliver improved performance of such models (Katzfuss and Guinness 2017; Guinness 2018).

When inference on the latent process is sought, Bayesian inference has the benefits of providing direct probability statements based upon the posterior distribution of the process. Inference based on asymptotic approximations are avoided, but there remain challenges in computing the posterior distribution given that inference is sought on a very high-dimensional parameter space (including the realizations of the latent process). One possibility, available for Gaussian first-stage likelihoods, is to work with a collapsed or marginalized likelihood by integrating out the spatial random effects. However, Gibbs samplers and other MCMC algorithms for the collapsed models can be inexorably slow and are impractical when data are in the millions. A sequential Gibbs sampler that updates the latent spatial effects (Datta, Banerjee, Finley, Gelfand, et al. 2016) is faster in updating the parameters but suffers from high autocorrelation and slow mixing. Another possibility emerges when interest lies in prediction or imputation of the outcome variable only and not the latent process. Here, a so called "response" model that models the outcome itself using an NNGP can be constructed. This model is much faster and enjoys superior convergence properties, but we lose inference on the latent process and its predictive performance tends to be inferior to the latent process model. Furthermore, these options are unavailable in non-Gaussian first-stage hierarchical models or when the focus is not uniquely on prediction. A detailed comparison of different approaches for computing Bayesian NNGP models is presented in Finley et al. (2019).

Our current contribution introduces a class of *meshed Gaussian process* (MGP) models for Bayesian hierarchical modeling of large spatial datasets. This class builds upon the aforementioned works that build upon Vecchia (1988) and other DAG based models. The inferential focus remains within the context of massive spatial datasets over very large domains. We exploit the demonstrated benefits of the DAG based models, but we now adapt them to partitioned domains. We describe dependence across regions of a partitioned domain using a small, patterned DAG which we refer to as a *mesh*. Within each region, some locations are selected as *reference* and collectively mapped to a single node in the DAG. Relationships among nodes are governed by kriging ideas. In the resulting MGP, regions in the spatial domain depend on each other through the reference locations. Realizations at all other locations are assumed independent, conditional upon the reference locations. This construction leads to a valid standalone spatial process.

As a particular subclass of MGPs, we propose a novel partitioning and graph design based on domain tessellations. Unlike methods that build sparse DAGs by limiting dependence to m nearest neighbors, our approach shapes the underlying DAG with a known, repeating pattern corresponding to the chosen tessellation geometry. The underlying sparse DAG enables scaling computations to large data settings and its known pattern guarantees the availability of block-parallel sampling schemes; furthermore, large computational savings can be achieved at no additional approximation cost if data are collected on patterned lattices. Finally, extensions to spatiotemporal and/or multivariate data are straightforward once a suitable covariance function has been defined. We use axis-parallel domain partitioning and the corresponding cubic DAG—resulting in cubic MGPs or Q-MGPs—to show substantial improvements in computational time and inferential performance relative to other models with data sizes ranging from the thousands to the several millions, for both spatial and spatiotemporal data and using multivariate spatial processes.

The present work may appear to share similarities with the block-NNGP model of Quiroz, Prates, and Dey (2019), who advocate building sparse DAGs on grouped locations based on their ordering and subsequent identification of m "past" neighbors. Unlike block-NNGPs, our tessellated GPs

consider the domain tessellation as generating the DAG; the number of parents of any node is thus fixed and depends on the geometry of the chosen tessellation rather than on a user-defined parameter. Inclusion of more distant locations in the parent set of any location will, therefore, not proceed by increasing the number of neighbors m, but rather by increasing the regions' size and/or modifying their shape. Central to tessellated GPs is the idea of forcing a DAG with known *coloring* on the data, resulting in guaranteed efficiencies when recovering the latent spatial effects. This strategy is analogous in spirit to multi-resolution approximations (Gramacy and Lee 2008; Katzfuss 2017), which also force a DAG on the data, resulting in conditional independence patterns that are known in advance and that can be used to improve computations. However, while multi-resolution approximations are defined by branching graphs associated to recursive domain partitioning, tessellated GPs use a single domain partition, with each region connected in the DAG only to its immediate neighbors. Compared to treed graphs, tessellated GPs are associated to DAGs with fewer conditionally independent groups and whose repeated patterns facilitate the identification of redundant matrix operations arising when one or more coordinate margins are gridded. We also note that while the idea of partitioning domains to create approximations is not new, construction of the DAG-based approximation over partitioned domains has received considerably less attention. Finally, our focus here is in developing tessellated GPs as a methodology that enables the efficient recovery of the latent spatial random effects and the Bayesian estimation of covariance parameters via MCMC; we are thus not focusing on alternative computational algorithms (see, e.g., Finley et al. 2019), which have been developed for NNGPs but can nonetheless all be adapted to general MGP models.

The balance of this article proceeds as follows. Section 2 introduces our general framework for hierarchical Bayesian modeling of spatial processes using networks of grouped spatial locations. The MGP is outlined in Section 3, where we provide a general, scalable computing algorithm in Section 3.1. Tessellation-based schemes and the specific case of Q-MGPs are outlined in Section 4, which highlights their properties and computational advantages. We illustrate the performance of our proposed approach in Section 5 using simulation experiments and an application on a massive dataset with millions of spatiotemporal locations. We conclude the article with a discussion and pointers to further research. Supplementary materials accompanying this article as an appendix are available online and contain further comparisons of Q-MGPs with several state-of-the-art methods for spatial data.

2. Spatial Processes on Partitioned Domains

A $q \times 1$ spatial process assigns a probability law on $\{\boldsymbol{w}(\boldsymbol{\ell}) : \boldsymbol{\ell} \in \mathcal{D}\}$, where $\boldsymbol{w}(\boldsymbol{\ell})$ is a $q \times 1$ random vector with elements $w_i(\boldsymbol{\ell})$ for $i = 1, 2, \ldots, q$. In the following general discussion we will not distinguish between spatial ($\mathcal{D} \subset \Re^d$) and spatiotemporal domains ($\mathcal{D} \subset \Re^{d+1}$), and denote spatial or spatiotemporal locations as $\boldsymbol{\ell}, \boldsymbol{s}$, or $\boldsymbol{u}$.

For any finite set of spatial locations $\{\boldsymbol{\ell}_1, \boldsymbol{\ell}_2, \ldots, \boldsymbol{\ell}_{n_\mathcal{L}}\} = \mathcal{L} \subset \mathcal{D}$ of size $n_\mathcal{L}$, let $P(\cdot)$ denote the probability law of the $n_\mathcal{L} q \times 1$ random vector $\boldsymbol{w}_\mathcal{L} = (\boldsymbol{w}(\boldsymbol{\ell}_1)^\top, \boldsymbol{w}(\boldsymbol{\ell}_2)^\top, \ldots, \boldsymbol{w}(\boldsymbol{\ell}_{n_\mathcal{L}})^\top)^\top$ with probability density $p(\cdot)$. The joint density of $\boldsymbol{w}_\mathcal{L}$ can be expressed as a DAG (or a Bayesian network model) with respect to the ordered set of locations $\mathcal{L}$ as

$$p(\boldsymbol{w}_\mathcal{L}) = \prod_{i=1}^{n_\mathcal{L}} p(\boldsymbol{w}(\boldsymbol{\ell}_i) \mid \boldsymbol{w}(\boldsymbol{\ell}_1), \ldots, \boldsymbol{w}(\boldsymbol{\ell}_{i-1})), \tag{1}$$

where the conditional set for each $\boldsymbol{w}(\boldsymbol{\ell}_i)$ can be interpreted as the set of its parents in a large, dense Bayesian network. Defining a simplified valid joint density on $\mathcal{L}$ by reducing the size of the conditioning sets is a popular strategy for fast likelihood approximations in the context of large spatial datasets. One typically limits dependence to "past" neighboring locations with respect to the ordering in (1) (Vecchia 1988; Stein, Chi, and Welty 2004; Gramacy and Apley 2015; Datta, Banerjee, Finley, Gelfand, et al. 2016; Katzfuss and Guinness 2017). The neighbors are defined and fixed and model performance may benefit from the addition of some distant locations (Stein, Chi, and Welty 2004). The ordering in $\mathcal{L}$ is also fixed and inferential performance may benefit from the use of some fixed permutations (Guinness 2018). The result of shrinking the conditional sets to a smaller set of neighbors from the past yields a sparse DAG or Bayesian network, which yields potentially massive computational gains.

We proceed in a similar manner, but instead of defining a sparse DAG at the level of each individual location, we map entire groups of locations to nodes in a much smaller graph; the same graph will be used to model the dependence between any location in the spatial domain and, therefore, to define a spatial process. Let $\mathcal{P} = \{\mathcal{D}_1, \ldots, \mathcal{D}_M\}$ be a partition of $\mathcal{D}$ into M mutually exclusive subsets so that $\mathcal{D} = \cup_{i=1}^M \mathcal{D}_i$ and $\mathcal{D}_i \cap \mathcal{D}_j = \emptyset$ whenever $i \neq j$. Similar to the nomenclature in the NNGP, we fix a *reference set* $\mathcal{S} = \{\boldsymbol{s}_1, \ldots, \boldsymbol{s}_{n_\mathcal{S}}\} \subset \mathcal{D}$, which itself is partitioned using $\mathcal{P}$ by letting $\mathcal{S}_j = \mathcal{D}_j \cap \mathcal{S}$. The set of nonreference locations is similarly partitioned with $\mathcal{U}_j = \mathcal{D}_j \setminus \mathcal{S}_j$ so that $\mathcal{D}_j = \mathcal{S}_j \cup \mathcal{U}_j$ for each $j = 1, 2, \ldots, M$. We now construct a DAG to model dependence within and between $\mathcal{S}$ and $\mathcal{U}$. Let $\mathcal{G} = \{\boldsymbol{V}, \boldsymbol{E}\}$ be a graph with nodes $\boldsymbol{V} = \boldsymbol{A} \cup \boldsymbol{B}$, where we refer to $\boldsymbol{A} = \{\boldsymbol{a}_1, \ldots, \boldsymbol{a}_M\}$ as the *reference* nodes and to $\boldsymbol{B} = \{\boldsymbol{b}_1, \ldots, \boldsymbol{b}_M\}$ as the *nonreference*, or simply "other", nodes. Let $\boldsymbol{A} \cap \boldsymbol{B} = \emptyset$. We introduce a map $\eta : \mathcal{D} \to \boldsymbol{V}$ such that

$$\eta(\boldsymbol{\ell}) = \begin{cases} \boldsymbol{a}_j \in \boldsymbol{A} \text{ if } \boldsymbol{\ell} \in \mathcal{S}_j, \\ \boldsymbol{b}_j \in \boldsymbol{B} \text{ if } \boldsymbol{\ell} \in \mathcal{U}_j. \end{cases} \tag{2}$$

This surjective many-to-one map links each location in $\mathcal{S}_j$ and $\mathcal{U}_j$ to a node in $\mathcal{G}$. The edges connecting nodes in $\mathcal{G}$ are $\boldsymbol{E} = \{\text{Pa}[\boldsymbol{v}_1], \ldots, \text{Pa}[\boldsymbol{v}_{2M}]\}$ where $\text{Pa}[\boldsymbol{v}] \subset \boldsymbol{V}$ denotes the set of parents of any $\boldsymbol{v} \in \boldsymbol{V}$ and, hence, identifies the directed edges pointing to $\boldsymbol{v}$. We let $\mathcal{G}$ be acyclic, that is, there is no chain $\{\boldsymbol{v}_{i_1} \to \boldsymbol{v}_{i_2} \to \cdots \to \boldsymbol{v}_{i_t}\}$ of elements of $\boldsymbol{V}$ such that $\boldsymbol{v}_{i_j} \in \text{Pa}[\boldsymbol{v}_{i_{j+1}}]$ and $\boldsymbol{v}_{i_{j+1}} \in \text{Pa}[\boldsymbol{v}_{i_1}]$. Crucially, we assume that $\text{Pa}[\boldsymbol{v}] \subset \boldsymbol{A}$ for all $\boldsymbol{v} \in \boldsymbol{V}$, that is, that only reference nodes have children, to distinguish the reference nodes $\boldsymbol{A}$ from the other nodes $\boldsymbol{B}$. Apart from the assumption that $\boldsymbol{a}_j \in \text{Pa}[\boldsymbol{b}_j]$, we refrain from defining the parents of a node, thereby retaining flexibility. In general, however, all locations in $\mathcal{U}_j$ will share the same parent set. In Section 4, we will consider meshes associated to domain tessellations.

Consider the enumeration $\mathcal{S}_i = \{\boldsymbol{s}_{i_1}, \ldots, \boldsymbol{s}_{i_{n_i}}\}$, where $\{i_1, i_2, \ldots, i_{n_i}\} \subset \{1, 2, \ldots, n_{\mathcal{S}}\}$, and let $\boldsymbol{w}_i = (\boldsymbol{w}(\boldsymbol{s}_{i_1})^\top, \boldsymbol{w}(\boldsymbol{s}_{i_2})^\top, \ldots, \boldsymbol{w}(\boldsymbol{s}_{i_{n_i}})^\top)^\top$ be the $n_i q \times 1$ random vector listing elements of $\boldsymbol{w}(\boldsymbol{s})$ for each $\boldsymbol{s} \in \mathcal{S}_i$. We now rewrite (1) as a product of M conditional densities

$$p(\boldsymbol{w}_{\mathcal{S}}) = p(\boldsymbol{w}_1, \boldsymbol{w}_2, \ldots, \boldsymbol{w}_M) = \prod_{i=1}^{M} p(\boldsymbol{w}_i \mid \boldsymbol{w}_1, \ldots, \boldsymbol{w}_{i-1}). \quad (3)$$

The conditioning sets are then reduced based on the graph $\mathcal{G}$:

$$\widetilde{p}(\boldsymbol{w}_{\mathcal{S}}) = \prod_{i=1}^{M} p(\boldsymbol{w}_i \mid \boldsymbol{w}_{[i]}) \,, \quad (4)$$

where we denote $\boldsymbol{w}_{[i]} = \{\boldsymbol{w}_j : \boldsymbol{a}_j \in \mathrm{Pa}[\boldsymbol{a}_i]\}$, and $\mathrm{Pa}[\boldsymbol{a}_i] \subset \{\boldsymbol{a}_1, \ldots, \boldsymbol{a}_{i-1}\} \subset \boldsymbol{A}$. This is a proper multivariate joint density since the graph is acyclic (Lauritzen 1996). It is instructive to note how the above approximation behaves when the size of the parent set shrinks, for a given domain partitioning scheme. To this end, we adapt a result in Banerjee (2020) and show that sparser DAGs correspond to a larger Kullback–Leibler (KL) divergence from the base density p. This result has been proved earlier for Gaussian likelihoods by Guinness (2018), but the argument given below is free of distributional assumptions and is linked to the submodularity of entropy and the "information never hurts" principle (see, e.g., Cover and Thomas 1991).

Consider random vector $\boldsymbol{w}$ and some partition of the domain $\mathcal{P}$ corresponding to nodes $\boldsymbol{V} = \{\boldsymbol{v}_1, \ldots, \boldsymbol{v}_M\}$ via map η. Let the base process correspond to graph $\mathcal{G}_0 = \{\boldsymbol{V}, \boldsymbol{E}_0\}$ where $\boldsymbol{E}_0 = \{\mathrm{Pa}_0[\boldsymbol{v}_1], \ldots, \mathrm{Pa}_0[\boldsymbol{v}_M]\}$. Then, let $\mathcal{G}_1 = \{\boldsymbol{V}, \boldsymbol{E}_1\}$ where $\boldsymbol{E}_1 = \{\mathrm{Pa}_1[\boldsymbol{v}_1], \ldots, \mathrm{Pa}_1[\boldsymbol{v}_M]\}$ and $\mathrm{Pa}_1[\boldsymbol{v}_i] \subseteq \mathrm{Pa}_0[\boldsymbol{v}_i]$ for all $i \in \{1, \ldots, M\}$. Finally construct $\mathcal{G}_2 = \{\boldsymbol{V}, \boldsymbol{E}_2\}$ by letting $\mathrm{Pa}_2[\boldsymbol{v}_{i^*}] = \mathrm{Pa}_1[\boldsymbol{v}_{i^*}] \backslash \{\boldsymbol{v}^*\}$ for some $\boldsymbol{v}^* \in \mathrm{Pa}_1[\boldsymbol{v}_{i^*}]$. In other words, graph $\mathcal{G}_2$ is obtained by removing the directed edge $\boldsymbol{v}^* \to \boldsymbol{v}_{i^*}$ from $\mathcal{G}_1$. We approximate p using densities p_1 and p_2 based on $\mathcal{G}_1$ and $\mathcal{G}_2$, respectively, obtaining

$$\frac{p_1(\boldsymbol{w})}{p_2(\boldsymbol{w})} = \prod_{i=1}^{M} \frac{p(\boldsymbol{w}_i \mid \boldsymbol{w}_{[i]_1})}{p(\boldsymbol{w}_i \mid \boldsymbol{w}_{[i]_2})} = \frac{p(\boldsymbol{w}_{i^*} \mid \boldsymbol{w}_{[i^*]_1})}{p(\boldsymbol{w}_{i^*} \mid \boldsymbol{w}_{[i^*]_2})}. \quad (5)$$

Considering the KL divergence of each density from p, and denoting $\boldsymbol{V}^* = \boldsymbol{V} \setminus \{\{i^*\} \cup \mathrm{Pa}_1[i^*]\}$, we find

$$\begin{aligned}
&\mathrm{KL}(p_2 \| p) - \mathrm{KL}(p_1 \| p) \\
&= \int \left\{ \log\left(\frac{p(\boldsymbol{w})}{p_2(\boldsymbol{w})}\right) - \log\left(\frac{p(\boldsymbol{w}}{p_1(\boldsymbol{w}}\right) \right\} p(\boldsymbol{w}) d\boldsymbol{w} \\
&= \int \log\left(\frac{p_1(\boldsymbol{w})}{p_2(\boldsymbol{w})}\right) p(\boldsymbol{w}) d\boldsymbol{w} \\
&= \int \log\left(\frac{p(\boldsymbol{w}_{i^*} \mid \boldsymbol{w}_{[i^*]_1})}{p(\boldsymbol{w}_{i^*} \mid \boldsymbol{w}_{[i^*]_2})}\right) p(\boldsymbol{w}) d\boldsymbol{w} \quad (6) \\
&= \int \log\left(\frac{p(\boldsymbol{w}_{i^*} \mid \boldsymbol{w}_{[i^*]_1})}{p(\boldsymbol{w}_{i^*} \mid \boldsymbol{w}_{[i^*]_2})}\right) p(\boldsymbol{w}_{i^*}, \boldsymbol{w}_{[i^*]_1}) d\boldsymbol{w}_{i^*} d\boldsymbol{w}_{[i^*]_1} \\
&= \int \left\{ \int \log\left(\frac{p(\boldsymbol{w}_{i^*} \mid \boldsymbol{w}_{[i^*]_1})}{p(\boldsymbol{w}_{i^*} \mid \boldsymbol{w}_{[i^*]_2})}\right) p(\boldsymbol{w}_{i^*} \mid \boldsymbol{w}_{[i^*]_1}) d\boldsymbol{w}_{i^*} \right\} \\
&\quad \times p(\boldsymbol{w}_{[i^*]_1}) d\boldsymbol{w}_{[i^*]_1} \geq 0,
\end{aligned}$$

where we use (5), the fact that $\boldsymbol{V}^*$ and $\{i^*\} \cup \mathrm{Pa}_1[i^*]$ are disjoint, and Jensen's inequality. This result implies that larger parent sets are preferrable as they correspond to better approximations to the full model; the choice of sparser graphs will be driven by computational considerations—see Section 3.2.

We construct the spatial process over arbitrary locations by enumerating other locations as $\mathcal{U} = \{\boldsymbol{u}_1, \ldots, \boldsymbol{u}_{n_{\mathcal{U}}}\} \subset \mathcal{D} \setminus \mathcal{S}$ and extending (4) to the nonreference locations. Given the partition of $\mathcal{U}$ defined earlier with components $\mathcal{U}_j$ for $j = 1, 2, \ldots, M$, for each $\boldsymbol{u} \in \mathcal{U}_j$ we set $\eta(\boldsymbol{u}) = \boldsymbol{b}_j$ and recall that $\mathrm{Pa}[\boldsymbol{b}_i] \subset \boldsymbol{A}$ by construction. For each $i = 1, \ldots, n_{\mathcal{U}}$, we denote $\boldsymbol{w}_{[\boldsymbol{u}_i]} = \{\boldsymbol{w}_j : \boldsymbol{a}_j \in \mathrm{Pa}[\eta(\boldsymbol{u}_i)]\} \subset \boldsymbol{w}_{\mathcal{S}}$ and define the conditional density of $\boldsymbol{w}_{\mathcal{U}}$ given $\boldsymbol{w}_{\mathcal{S}}$ as

$$\widetilde{p}(\boldsymbol{w}_{\mathcal{U}} \mid \boldsymbol{w}_{\mathcal{S}}) = \prod_{\boldsymbol{u}_i \in \mathcal{U}} p(\boldsymbol{w}(\boldsymbol{u}_i) \mid \boldsymbol{w}_{[\boldsymbol{u}_i]}) = \prod_{j=1}^{M} p(\boldsymbol{w}_{\mathcal{U}_j} \mid \boldsymbol{w}_{[\boldsymbol{b}_j]}). \quad (7)$$

Therefore, for any finite subset of spatial locations $\mathcal{L} \subset \mathcal{D}$ we can let $\mathcal{U} = \mathcal{L} \setminus \mathcal{S}$ and obtain

$$\widetilde{p}(\boldsymbol{w}_{\mathcal{L}}) = \int \widetilde{p}(\boldsymbol{w}_{\mathcal{U}} \mid \boldsymbol{w}_{\mathcal{S}}) \widetilde{p}(\boldsymbol{w}_{\mathcal{S}}) \prod_{\boldsymbol{s}_i \in \mathcal{S} \setminus \mathcal{L}} d(\boldsymbol{w}(\boldsymbol{s}_i)) \,.$$

We show (see Appendix A, available online) that this is a well-defined process by verifying the Kolmogorov consistency conditions. This new process can be built starting from a base process, a fixed reference set, domain partition $\mathcal{P}$ and a graph $\mathcal{G}$. Next, we elucidate with GPs.

3. Meshed Gaussian Processes

Let $\{\boldsymbol{w}(\boldsymbol{\ell}) : \boldsymbol{\ell} \in \mathcal{D}\}$ be a q-variate multivariate GP, denoted as $\boldsymbol{w}(\boldsymbol{\ell}) \sim \mathrm{GP}(\boldsymbol{0}, \boldsymbol{C}(\cdot, \cdot \mid \boldsymbol{\theta}))$. The *cross-covariance* $\boldsymbol{C}(\cdot, \cdot \mid \boldsymbol{\theta})$ indexed by parameters $\boldsymbol{\theta}$ is a function $\boldsymbol{C} : \mathcal{D} \times \mathcal{D} \to \mathcal{M}_{q \times q}$, where $\mathcal{M}_{q \times q}$ is a subset of $\Re^{q \times q}$ (the space of all $q \times q$ real matrices) such that the (i, j)th entry of $\boldsymbol{C}(\boldsymbol{\ell}, \boldsymbol{\ell}' \mid \boldsymbol{\theta})$ evaluates the covariance between the ith and jth elements of $\boldsymbol{w}(\boldsymbol{\ell})$ at $\boldsymbol{\ell}$ and $\boldsymbol{\ell}'$, respectively, that is, $\mathrm{cov}(w_i(\boldsymbol{\ell}), w_j(\boldsymbol{\ell}'))$. We omit dependence on $\boldsymbol{\theta}$ to simplify notation. The cross-covariance function itself needs to be neither symmetric nor positive-definite, but must satisfy the following two properties: (i) $\boldsymbol{C}(\boldsymbol{\ell}, \boldsymbol{\ell}') = \boldsymbol{C}(\boldsymbol{\ell}', \boldsymbol{\ell})^\top$; and (ii) $\sum_{i=1}^{n} \sum_{j=1}^{n} \boldsymbol{z}_i^\top \boldsymbol{C}(\boldsymbol{\ell}_i, \boldsymbol{\ell}_j) \boldsymbol{z}_j > 0$ for any integer n and any finite collection of points $\{\boldsymbol{\ell}_1, \boldsymbol{\ell}_2, \ldots, \boldsymbol{\ell}_n\}$ and for all $\boldsymbol{z}_i \in \Re^q \backslash \{\boldsymbol{0}\}$. See Genton and Kleiber (2015) for a review of cross-covariance functions for multivariate processes. The (partial) realization of the multivariate process over any finite set $\mathcal{L}$ has a multivariate normal distribution $\boldsymbol{w}_{\mathcal{L}} \sim N(0, \boldsymbol{C}_{\mathcal{L}})$ where $\boldsymbol{w}_{\mathcal{L}}$ is the $qn_{\mathcal{L}} \times 1$ column vector and $\boldsymbol{C}_{\mathcal{L}}$ is the $qn_{\mathcal{L}} \times qn_{\mathcal{L}}$ block matrix with the $q \times q$ matrix $\boldsymbol{C}(\boldsymbol{\ell}_i, \boldsymbol{\ell}_j)$ as its (i, j) block for $i, j = 1, \ldots, n_{\mathcal{L}}$.

We construct the MGP from a base, or *parent*, multivariate GP for $\boldsymbol{w}(\boldsymbol{\ell})$ and then, using the graph $\mathcal{G}$ defined in Section 2, represent the joint density at the reference set $\mathcal{S}$ as

$$\widetilde{p}(\boldsymbol{w}_{\mathcal{S}}) = \prod_{j=1}^{M} N(\boldsymbol{w}_j \mid \boldsymbol{H}_j \boldsymbol{w}_{[j]}, \boldsymbol{R}_j), \quad (8)$$

where $\boldsymbol{H}_1 = \boldsymbol{O}_{n_1 \times 1}$, $\boldsymbol{R}_1 = \boldsymbol{C}_{\mathcal{S}_j}$ and for $j > 1$, $\boldsymbol{H}_j = \boldsymbol{C}_{\mathcal{S}_j, \mathcal{S}_{[j]}} \boldsymbol{C}_{\mathcal{S}_{[j]}}^{-1}$ and $\boldsymbol{R}_j = \boldsymbol{C}_{\mathcal{S}_j} - \boldsymbol{C}_{\mathcal{S}_j, \mathcal{S}_{[j]}} \boldsymbol{C}_{\mathcal{S}_{[j]}}^{-1} \boldsymbol{C}_{\mathcal{S}_{[j]}, \mathcal{S}_j}$. The resulting joint density $\widetilde{p}(\boldsymbol{w}_{\mathcal{S}})$ is multivariate normal with covariance $\widetilde{\boldsymbol{C}}_{\mathcal{S}}$ and a precision matrix $\widetilde{\boldsymbol{C}}_{\mathcal{S}}^{-1}$. The precision matrix for Gaussian

graphical models is easily derived using customary linear model representations for each conditional regression. Consider the DAG in (4). Each $\boldsymbol{w}_i$ is $n_iq \times 1$ and let $J_i = |\text{Pa}[\boldsymbol{a}_i]|$ be the number of parents for $\boldsymbol{a}_i$ in the graph $\mathcal{G}$. Furthermore, let $\boldsymbol{C}_{i,j}$ be the $n_iq \times n_jq$ covariance matrix between $\boldsymbol{w}_i$ and $\boldsymbol{w}_j$, $\boldsymbol{C}_{i,[i]}$ be the $n_iq \times J_iq$ covariance matrix between $\boldsymbol{w}_i$ and $\boldsymbol{w}_{[i]}$, and $\boldsymbol{C}_{[i],[i]}$ be the $J_iq \times J_iq$ covariance matrix between $\boldsymbol{w}_{[i]}$ and itself. Representing each conditional density in (4) as a linear regression on $\boldsymbol{w}_i$, we get

$$\boldsymbol{w}_1 = \boldsymbol{\omega}_1 \sim N(\boldsymbol{0}, \boldsymbol{R}_1)\,; \quad \boldsymbol{w}_i = \sum_{\{j:\boldsymbol{a}_j \in \text{Pa}[\boldsymbol{a}_i]\}} \boldsymbol{H}_{ij}\boldsymbol{w}_j + \boldsymbol{\omega}_i\,, \quad i = 2, 3, \ldots, M\,, \tag{9}$$

where each $\boldsymbol{H}_{ij}$ is an $n_iq \times n_jq$ is a coefficient matrix representing the multivariate regression of $\boldsymbol{w}_j$ given $\boldsymbol{w}_{[i]}$, $\boldsymbol{\omega}_i \overset{\text{ind}}{\sim} N(\boldsymbol{0}, \boldsymbol{R}_i)$ for $i = 1, 2, \ldots, M$, and each $\boldsymbol{R}_i$ is an $n_iq \times n_iq$ residual covariance matrix. We set $\boldsymbol{H}_{ii} = \boldsymbol{O}$ and $H_{ij} = \boldsymbol{O}$, where $\boldsymbol{O}$ is the matrix of zeros, whenever $j \in \{j : \boldsymbol{a}_j \notin \text{Pa}[\boldsymbol{a}_i]\}$. For $j \in \{j : \boldsymbol{a}_j \in \text{Pa}[\boldsymbol{a}_i]\}$, let $\{j_1, j_2, \ldots, j_{J_i}\}$ be the indices in $\text{Pa}[\boldsymbol{a}_i]$ and let $\boldsymbol{H}_{i,[i]} = \left[\boldsymbol{H}_{i,j_1}, \boldsymbol{H}_{i,j_2}, \ldots, \boldsymbol{H}_{i,j_{J_i}}\right]$ be the $n_iq \times (\sum_{k=1}^{J_i} n_{j_k})q$ block matrix formed by stacking $\boldsymbol{H}_{i,j_k}$ side by side for each $\boldsymbol{a}_{j_k} \in \text{Pa}[\boldsymbol{a}_i]$. Since $\text{E}[\boldsymbol{w}_i \mid \boldsymbol{w}_{[i]}] = \boldsymbol{H}_{i,[i]}\boldsymbol{w}_{[i]} = \boldsymbol{C}_{i,[i]}\boldsymbol{C}^{-1}_{[i][i]}\boldsymbol{w}_{[i]}$, we obtain $\boldsymbol{H}_{i,[i]} = \boldsymbol{C}_{i,[i]}\boldsymbol{C}^{-1}_{[i][i]}$ and each $\boldsymbol{H}_{ij_K}$ can be obtained from the respective submatrix of $\boldsymbol{H}_{i[i]}$. We also obtain $\boldsymbol{R}_i = \text{var}\{\boldsymbol{w}_i \mid \boldsymbol{w}_{[i]}\} = \boldsymbol{C}_{i,i} - \boldsymbol{C}_{i,[i]}\boldsymbol{C}^{-1}_{[i][i]}\boldsymbol{C}_{[i],i}$. Therefore, all the $\boldsymbol{H}_{ij}$'s and $\boldsymbol{R}_i$'s can be computed from the base cross-covariance function.

The distribution of $\boldsymbol{w} = [\boldsymbol{w}_1^\top, \boldsymbol{w}_2^\top, \ldots, \boldsymbol{w}_M^\top]^\top$ can be obtained by noting that $\boldsymbol{w} = \boldsymbol{H}\boldsymbol{w} + \boldsymbol{\omega}$, where $\boldsymbol{H} = \{\boldsymbol{H}_{ij}\}$ is the $(\sum_{i=1}^M n_iq) \times (\sum_{i=1}^M n_iq)$ block matrix with $\{\boldsymbol{H}_{ij}\}$ as (i,j)th block. Therefore, $\widetilde{\boldsymbol{C}}_{\mathcal{S}} = \text{var}(\boldsymbol{w}) = (\boldsymbol{I} - \boldsymbol{H})^{-1}\boldsymbol{R}(\boldsymbol{I} - \boldsymbol{H})^{-\top}$, where $\boldsymbol{R}$ is block-diagonal with $\boldsymbol{R}_i$ as the (i,i)th block. Note that $\boldsymbol{I} - \boldsymbol{H}$ is block lower-triangular with 1's on the diagonal, hence nonsingular. Also, the precision matrix $\widetilde{\boldsymbol{C}}_{\mathcal{S}}^{-1} = (\boldsymbol{I} - \boldsymbol{H})^\top\boldsymbol{R}^{-1}(\boldsymbol{I} - \boldsymbol{H})$ is sparse because of $\boldsymbol{H}_{ij} = \boldsymbol{O}$ whenever $\boldsymbol{a}_j \notin \text{Pa}[\boldsymbol{a}_i]$. Block-sparsity of $\widetilde{\boldsymbol{C}}_{\mathcal{S}}^{-1}$ can be induced by building $\mathcal{G}$ with few, carefully placed directed edges among nodes in $\boldsymbol{A}$; Appendix B, available online, contains a more in-depth treatment. We extend (8) to the collection of nonreference locations $\mathcal{U} \subset \mathcal{D} \setminus \mathcal{S}$:

$$\begin{aligned}\widetilde{p}(\boldsymbol{w}_{\mathcal{U}} \mid \boldsymbol{w}_{\mathcal{S}}) &= \prod_{j=1}^{M} N(\boldsymbol{w}_{\mathcal{U}_j} \mid \boldsymbol{H}_{\mathcal{U}_j}\boldsymbol{w}_{[\boldsymbol{b}_j]}, \boldsymbol{R}_{\mathcal{U}_j}) \\ &= N(\boldsymbol{w}_{\mathcal{U}} \mid \boldsymbol{H}_{\mathcal{U}}\boldsymbol{w}_{\mathcal{S}}, \boldsymbol{R}_{\mathcal{U}}),\end{aligned} \tag{10}$$

where $\boldsymbol{H}_{\mathcal{U}_j} = \boldsymbol{C}_{\mathcal{U}_j,\mathcal{S}_{[\boldsymbol{b}_j]}}\boldsymbol{C}^{-1}_{\mathcal{S}_{[\boldsymbol{b}_j]}}$ and $\boldsymbol{R}_{\mathcal{U}_j} = \boldsymbol{C}_{\mathcal{U}_j} - \boldsymbol{C}_{\mathcal{U}_j,\mathcal{S}_{[\boldsymbol{b}_j]}}\boldsymbol{C}^{-1}_{\mathcal{S}_{[\boldsymbol{b}_j]}}\boldsymbol{C}_{\mathcal{S}_{[\boldsymbol{b}_j]},\mathcal{U}_j}$, analogously to (8), while $\boldsymbol{H}_{\mathcal{U}}$ and $\boldsymbol{R}_{\mathcal{U}}$ are analogous to $\boldsymbol{H}_{\mathcal{S}}$ and $\boldsymbol{R}_{\mathcal{S}}$. Clearly, given that all the $\widetilde{p}$ densities are Gaussian, all finite dimensional distributions will also be Gaussian. We have constructed a GP with the following cross-covariance function for any two locations $\boldsymbol{\ell}_1, \boldsymbol{\ell}_2 \in \mathcal{D}$

$$\text{cov}_{\widetilde{p}}(\boldsymbol{w}(\boldsymbol{\ell}_1), \boldsymbol{w}(\boldsymbol{\ell}_2)) = \begin{cases} \widetilde{\boldsymbol{C}}_{\boldsymbol{s}_i,\boldsymbol{s}_j} & \text{if } \boldsymbol{\ell}_1 = \boldsymbol{s}_i, \boldsymbol{\ell}_2 = \boldsymbol{s}_j \text{ and } \boldsymbol{s}_i, \boldsymbol{s}_j \in \mathcal{S}, \\ \boldsymbol{H}_{\boldsymbol{\ell}_1}\widetilde{\boldsymbol{C}}_{\mathcal{S}_{[\boldsymbol{\ell}_1]},\boldsymbol{s}_j} & \text{if } \boldsymbol{\ell}_1 \in \mathcal{D} \setminus \mathcal{S}, \boldsymbol{\ell}_2 = \boldsymbol{s}_j \text{ and } \boldsymbol{s}_j \in \mathcal{S}, \\ \delta_{(\boldsymbol{\ell}_1 = \boldsymbol{\ell}_2)}\boldsymbol{R}_{\boldsymbol{\ell}_1} & \\ \quad + \boldsymbol{H}_{\boldsymbol{\ell}_1}\widetilde{\boldsymbol{C}}_{\mathcal{S}_{[\boldsymbol{\ell}_1]},\mathcal{S}_{[\boldsymbol{\ell}_2]}}\boldsymbol{H}_{\boldsymbol{\ell}_2}^\top & \text{otherwise.} \end{cases}$$

For a given base Gaussian covariance function $\boldsymbol{C}$, domain partitioning $\mathcal{P}$, mesh $\mathcal{G}$, and reference set $\mathcal{S}$, we denote the corresponding MGP as $\text{MGP}(\mathcal{G}, \mathcal{P}, \mathcal{S}, \boldsymbol{C})$.

3.1. Bayesian Hierarchical Model and Gibbs Sampler

Meshed GPs produce block-sparse precision matrices that are constructed cheaply from their block-sparse Cholesky factors by solving small linear systems. General purpose sparse-Cholesky algorithms (Davis 2006; Chen et al. 2008) can then be used to obtain collapsed samplers as in Finley et al. (2019). Unfortunately, these algorithms can only be used on Gaussian first stage models and are computationally impracticable for data in the millions. Hence, we develop a more general scalable Gibbs sampler for the recovery of spatial random effects in hierarchical MGP models that entirely circumvents large matrix computations.

Consider a multivariate spatiotemporally varying regression model at $\boldsymbol{\ell} \in \mathcal{D} \subset \Re^{d+1}$,

$$\boldsymbol{y}(\boldsymbol{\ell}) = \boldsymbol{X}(\boldsymbol{\ell})^\top\boldsymbol{\beta} + \boldsymbol{Z}(\boldsymbol{\ell})^\top\boldsymbol{w}(\boldsymbol{\ell}) + \boldsymbol{\varepsilon}(\boldsymbol{\ell}), \tag{11}$$

where $\boldsymbol{y}(\boldsymbol{\ell}) \in \Re^l$ is the multivariate point-referenced outcome, $\boldsymbol{X}(\boldsymbol{\ell})^\top = \text{blockdiag}\{\boldsymbol{x}_i(\boldsymbol{\ell})^\top\}_{i=1}^l$ is a $l \times p = l \times \sum p_i$ matrix of spatially referenced predictors linked to constant coefficients $\boldsymbol{\beta}$, $\boldsymbol{w}(\boldsymbol{\ell})$ is the spatial process, $\boldsymbol{Z}(\boldsymbol{\ell})$ is a $l \times q$ design matrix, $\boldsymbol{\varepsilon}(\boldsymbol{\ell})$ is measurement error such that $\boldsymbol{\varepsilon}(\boldsymbol{\ell}) \overset{\text{iid}}{\sim} N(0, \boldsymbol{D})$ and $\boldsymbol{D} = \text{diag}(\tau_1^2, \ldots, \tau_l^2)$. A simple univariate regression model with a spatially varying intercept can be obtained with $l = 1$, $\boldsymbol{Z}(\boldsymbol{\ell}) = 1$. For observed locations $\mathcal{T} = \{\boldsymbol{\ell}_1, \ldots, \boldsymbol{\ell}_n\}$, we write the above model compactly $\boldsymbol{y} = \boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{Z}\boldsymbol{w} + \boldsymbol{\varepsilon}$, where $\boldsymbol{y} = (\boldsymbol{y}(\boldsymbol{\ell}_1)^\top, \ldots, \boldsymbol{y}(\boldsymbol{\ell}_n)^\top)^\top$, $\boldsymbol{w}$ and $\boldsymbol{\varepsilon}$ are similarly defined, $\boldsymbol{X} = [\boldsymbol{X}(\boldsymbol{\ell}_1) : \cdots : \boldsymbol{X}(\boldsymbol{\ell}_n)]^\top$, $\boldsymbol{Z} = \text{blockdiag}(\{\boldsymbol{Z}(\boldsymbol{\ell}_i)^\top\}_{i=1}^n)$, and $\boldsymbol{D}_n = \text{blockdiag}(\{\boldsymbol{D}\}_{i=1}^n)$.

For subsets $\{\boldsymbol{\ell}_1, \ldots, \boldsymbol{\ell}_{n_{\mathcal{A}}}\} = \mathcal{A} \subset \mathcal{T}$, let $\boldsymbol{y}(\mathcal{A}) = (\boldsymbol{y}(\boldsymbol{\ell}_1)^\top, \ldots, \boldsymbol{y}(\boldsymbol{\ell}_{n_{\mathcal{A}}})^\top)^\top$, with analogous definitions for $\boldsymbol{w}(\mathcal{A})$ and $\boldsymbol{\varepsilon}(\mathcal{A})$, $\boldsymbol{X}(\mathcal{A}) = [\boldsymbol{X}(\boldsymbol{\ell}_1) : \ldots : \boldsymbol{X}(\boldsymbol{\ell}_{n_{\mathcal{A}}})]^\top$, $\boldsymbol{Z}_{\mathcal{A}} = \text{blockdiag}(\{\boldsymbol{Z}(\boldsymbol{\ell}_i)^\top\}_{i=1}^{n_{\mathcal{A}}})$ and $\boldsymbol{D}_{\mathcal{A}} = \text{blockdiag}(\{\boldsymbol{D}\}_{i=1}^{n_{\mathcal{A}}})$. After fixing a reference set $\mathcal{S}$, we obtain $\mathcal{S}^* = \mathcal{T} \cap \mathcal{S}$ and $\mathcal{U} = \mathcal{T} \setminus \mathcal{S}$. We partition the domain as above to obtain $\mathcal{S}_j, \mathcal{S}_j^*, \mathcal{U}_j$ for $j = 1, \ldots, M$ and model $\boldsymbol{w}(\boldsymbol{\ell})$ using the MGP which yields $\boldsymbol{w} \sim N(\boldsymbol{0}, \widetilde{\boldsymbol{C}}_{\mathcal{S}}^{-1})$. We complete the model specification by assigning $\boldsymbol{\beta} \sim N(\boldsymbol{\beta} \mid \boldsymbol{\mu}_\beta, \boldsymbol{\Sigma}_\beta)$, $\tau_j^2 \sim \text{Inv.Gamma}(\tau_j^2 \mid a_{\tau_j}, b_{\tau_j})$, $\boldsymbol{\theta} \sim p(\boldsymbol{\theta})$.

The resulting full conditional distribution for $\boldsymbol{\beta}$ is $N(\boldsymbol{\Sigma}_\beta^*\boldsymbol{\mu}_\beta^*, \boldsymbol{\Sigma}_\beta^*)$, where $\boldsymbol{\Sigma}_\beta^* = (\boldsymbol{\Sigma}_\beta^{-1} + \boldsymbol{X}^\top\boldsymbol{D}_n^{-1}\boldsymbol{X})^{-1}$, $\boldsymbol{\mu}_\beta^* = \boldsymbol{\Sigma}_\beta^{-1}\boldsymbol{\mu}_\beta + \boldsymbol{X}^\top\boldsymbol{D}_n^{-1}(\boldsymbol{y} - \boldsymbol{Z}\boldsymbol{w})$. For τ_r^2, $r = 1, \ldots, q$, the full conditional is Inverse-Gamma with parameters $a_{\tau_r} + n/2$ and $b_{\tau_r} + \frac{1}{2}\boldsymbol{E}_r^\top\boldsymbol{E}_r$ where $\boldsymbol{E}_r = \boldsymbol{y}_{\cdot r} - \boldsymbol{X}_{\cdot r}\boldsymbol{\beta} - \boldsymbol{Z}_{\cdot r}\boldsymbol{w}$ and $\boldsymbol{y}_{\cdot r}, \boldsymbol{X}_{\cdot r}, \boldsymbol{Z}_{\cdot r}$ are the subsets of $\boldsymbol{y}, \boldsymbol{X}, \boldsymbol{Z}$ corresponding to outcome r (out of q).

The Gibbs update of the $\boldsymbol{w}_{\mathcal{U}}$ components can proceed simultaneously as all blocks in $\mathcal{U}$ have no children and their parents are in $\mathcal{S}$. The full conditional for $\boldsymbol{w}_{\mathcal{U}_j}$ for $j = 1, \ldots, M$ is thus $N(\boldsymbol{\Sigma}_{\mathcal{U}_j}^*\boldsymbol{\mu}_{\mathcal{U}_j}^*, \boldsymbol{\Sigma}_{\mathcal{U}_j}^*)$ where $\boldsymbol{\Sigma}_{\mathcal{U}_j}^* = (\boldsymbol{Z}(\mathcal{U}_j)\boldsymbol{D}^{-1}\boldsymbol{Z}(\mathcal{U}_j)^\top + \boldsymbol{R}_{\mathcal{U}_j}^{-1})^{-1}$ and $\boldsymbol{\mu}_{\mathcal{U}_j}^* = \boldsymbol{Z}(\mathcal{U}_j)\boldsymbol{D}^{-1}(\boldsymbol{y}(\mathcal{U}_j) - \boldsymbol{X}(\mathcal{U}_j)^\top\boldsymbol{\beta}) + \boldsymbol{R}_{\mathcal{U}_j}^{-1}\boldsymbol{H}_{\mathcal{U}_j}\boldsymbol{w}_{[\boldsymbol{b}_j]}$, where $\boldsymbol{w}_{[\boldsymbol{b}_j]}$ is the spatial process at locations corresponding to the parents of $\boldsymbol{b}_j \in \boldsymbol{B} \subset \boldsymbol{V}$.

We update $\boldsymbol{w}_{\mathcal{S}_j} = \boldsymbol{w}_j$ for $j = 1, \ldots, M$ via its full conditional $N(\boldsymbol{\Sigma}_j^* \boldsymbol{\mu}_j^*, \boldsymbol{\Sigma}_j^*)$. Let $\mathbf{1}_j = (In(s_1 \in \mathcal{S}_j^*), \ldots, In(s_{n_j} \in \mathcal{S}_j^*))^\top$ be the vector of indicators that identify locations with nonmissing outputs, and let $\boldsymbol{a}_j \in \boldsymbol{V}$ be the node in $\mathcal{G}$ corresponding to $\mathcal{S}_j$. Then,

$$\begin{aligned}\boldsymbol{\Sigma}_j^{*-1} &= \boldsymbol{Z}_j^\top \widetilde{\boldsymbol{D}}_{n_j}^{-1} \boldsymbol{Z}_j + \boldsymbol{R}_j^{-1} + \sum_{i=1}^{|\mathrm{Ch}[\boldsymbol{a}_j]|} \boldsymbol{H}_i^{[j]\top} \boldsymbol{R}_i^{[j]-1} \boldsymbol{H}_i^{[j]}, \\ \boldsymbol{\mu}_j^* &= \boldsymbol{R}_j^{-1} \boldsymbol{H}_j \boldsymbol{w}_{[j]} + \boldsymbol{Z}_j^\top \widetilde{\boldsymbol{D}}_{n_j}^{-1} \widetilde{\boldsymbol{y}}_j + \sum_{i=1}^{|\mathrm{Ch}[\boldsymbol{a}_j]|} \boldsymbol{H}_i^{[j]\top} \boldsymbol{R}_i^{[j]-1} w_i^{[j]}, \end{aligned} \tag{12}$$

where $\widetilde{\boldsymbol{D}}_{n_j}^{-1} = \boldsymbol{I}_j \odot \boldsymbol{D}_{n_j}^{-1}$ with $\boldsymbol{I}_j = \mathbf{1}_j \mathbf{1}_j^\top$, and $\widetilde{\boldsymbol{y}}_j = \mathbf{1}_j \odot (\boldsymbol{y}_j - \boldsymbol{X}_j \beta)$ and $\odot$ denotes the Hadamard or Schur (element-by-element) product. Finally, $\boldsymbol{\theta}$ is updated via a Metropolis step with target density $p(\boldsymbol{\theta}) N(\boldsymbol{w}_{\mathcal{S}} \mid \mathbf{0}, \widetilde{\boldsymbol{C}}_{\mathcal{S}}) N(\boldsymbol{w}_{\mathcal{U}} \mid \boldsymbol{H}_{\mathcal{U}} \boldsymbol{w}_{\mathcal{S}}, \boldsymbol{R}_{\mathcal{U}})$ using (8) and (10). The Gibbs sampling algorithm will iterate across the above steps and, upon convergence, will produce samples from $p(\boldsymbol{\beta}, \{\tau_j^2\}_{j=1}^q, \boldsymbol{w} \mid \boldsymbol{y})$.

We obtain posterior predictive inference at arbitrary $\boldsymbol{\ell} \in \mathcal{D}$ by evaluating $p(\boldsymbol{y}(\boldsymbol{\ell}) \mid \boldsymbol{y})$. If $\boldsymbol{\ell} \in \mathcal{S} \cup \mathcal{U}$, then we draw one sample of $\boldsymbol{y}(\boldsymbol{\ell}) \sim N(\boldsymbol{X}(\boldsymbol{\ell})^\top \boldsymbol{\beta} + \boldsymbol{Z}(\boldsymbol{\ell})^\top \boldsymbol{w}(\boldsymbol{\ell}), \boldsymbol{D})$ for each draw of the parameters from $p(\boldsymbol{\beta}, \{\tau_j^2\}_{j=1}^q, \boldsymbol{w} \mid \boldsymbol{y})$. Otherwise, considering that $\boldsymbol{\ell} \in \mathcal{D}_j$ for some j and thus $\eta(\boldsymbol{\ell}) = \boldsymbol{b}_j$, with parent nodes $\mathrm{Pa}[\boldsymbol{b}_j]$ and children $\mathrm{Ch}[\boldsymbol{b}_j] = \emptyset$, we sample $\boldsymbol{w}(\boldsymbol{\ell})$ from the full conditional $N(\boldsymbol{\Sigma}_{\boldsymbol{\ell}}^* \boldsymbol{\mu}_{\boldsymbol{\ell}}^*, \boldsymbol{\Sigma}_{\boldsymbol{\ell}}^*)$, where $\boldsymbol{\Sigma}_{\boldsymbol{\ell}}^* = (\boldsymbol{Z}(\boldsymbol{\ell}) \boldsymbol{D}^{-1} \boldsymbol{Z}(\boldsymbol{\ell})^\top + \boldsymbol{R}_{\boldsymbol{\ell}}^{-1})^{-1}$ and $\boldsymbol{\mu}_{\boldsymbol{\ell}}^* = \boldsymbol{Z}(\boldsymbol{\ell}) \boldsymbol{D}^{-1} (\boldsymbol{y}(\boldsymbol{\ell}) - \boldsymbol{X}(\boldsymbol{\ell})^\top \boldsymbol{\beta}) + \boldsymbol{R}_{\boldsymbol{\ell}}^{-1} \boldsymbol{H}_{\boldsymbol{\ell}} \boldsymbol{w}_{[\boldsymbol{b}_j]}$, then draw $\boldsymbol{y}(\boldsymbol{\ell}) \sim N(\boldsymbol{X}(\boldsymbol{\ell})^\top \boldsymbol{\beta} + \boldsymbol{Z}(\boldsymbol{\ell})^\top \boldsymbol{w}(\boldsymbol{\ell}), \boldsymbol{D})$.

3.2. Nonseparable Multivariate Spatiotemporal Covariances

We provide an account of the computational cost of general MGPs as a starting point to motivate the introduction of more efficient tessellated MGPs, and specifically Q-MGPs, in Section 4. We consider (11) and take $l = 1$ to simplify our exposition. In the resulting model, β is the regression coefficient on the p point-referenced regressors with a static effect on the outcome, whereas the q-variate spatiotemporal process $\boldsymbol{w}(\cdot)$ captures the dynamic effect of the $\boldsymbol{Z}$ regressors. Typically in geostatistical modeling p and q are small, hence sampling $\boldsymbol{\beta}$ and τ^2 carries a negligible computational cost. The cost of each Gibbs iteration is dominated by updates of $\boldsymbol{\theta}$ and $\boldsymbol{w}$. Let us assume, solely for expository purposes, that each of the M blocks comprise the same number of locations, that is, $|\mathcal{S}_j| = |\mathcal{U}_j| = m$, for all $j = 1, \ldots, M$. Thus, $m = \frac{n}{2M}$ and the graph nodes have J or fewer parents and L or fewer children.

The evaluation of $N(\boldsymbol{w}_{\mathcal{S}} \mid \mathbf{0}, \widetilde{\boldsymbol{C}}_{\mathcal{S}}) = \prod_{j=1}^M N(\boldsymbol{w}_j \mid \boldsymbol{H}_j \boldsymbol{w}_{[j]}, \boldsymbol{R}_j)$ and $N(\boldsymbol{w}_{\mathcal{U}} \mid \boldsymbol{H}_{\mathcal{U}} \boldsymbol{w}_{\mathcal{S}}, \boldsymbol{R}_{\mathcal{U}}) = \prod_{j=1}^M N(\boldsymbol{w}_{\mathcal{U}_j} \mid \boldsymbol{H}_{\mathcal{U}_j} \boldsymbol{w}_{[\boldsymbol{b}_j]}, \boldsymbol{R}_{\mathcal{U}_j})$ dominates the computation. Each term in the product entails $\boldsymbol{R}_j^{-1}$ and $\boldsymbol{R}_{\mathcal{U}_j}^{-1}$, both of size $qm \times qm$, and their determinants. These require $\boldsymbol{C}_{[j]}^{-1}$ of size $Jqm \times Jqm$ or less, resulting in $O(2M(q^3 m^3 + J^3 q^3 m^3)) = O(2Mq^3 m^3 (J^3 + 1)) \approx O(2Mq^3 m^3 J^3) = O(\frac{n^3 q^3 J^3}{M^2})$ flops via Cholesky decomposition. Reasonably, J and m are fixed so M may grow linearly with sample size and the cost is $O(nq^3 J^3)$ considering $M \propto n$. The total computing time is $\sim O(\frac{nq^3 J^3}{K})$ with K processors for computing the $2M$ densities. Sampling $\boldsymbol{w}_{\mathcal{S}}$ and $\boldsymbol{w}_{\mathcal{U}}$ from their full conditional distributions requires $O(2Mq^3 m^3 + MLq^2 m^2 + Mq^2 m^2)$ flops, assuming $\boldsymbol{R}_j^{-1}$ and $\boldsymbol{R}_{\mathcal{U}_j}^{-1}$ are stored in the previous step. The first term in the complexity order is due to the Cholesky decomposition of covariance matrices, the second is due to sampling the reference nodes, and the third comes from sampling other nodes. Without further assumptions, parallelization reduces complexity to $O(\frac{2Mq^3 m^3}{K} + \frac{Mq^2 m^2}{K} + MLq^2 m^2)$, since the covariances can be computed beforehand and the M components of $\boldsymbol{w}_{\mathcal{U}}$ are independent given $\boldsymbol{w}_{\mathcal{S}}$. With fixed block size m, the overall complexity for a Gibbs iteration is $O(\frac{2}{K} Mq^3 m^3 (J^3 + 1) + \frac{1}{K} 2Mq^3 m^3 + \frac{1}{K} Mq^2 m^2 + MLq^2 m^2) \approx O(\frac{1}{K} J^3 q^3 n + q^2 n) \approx O(n)$, linear in the sample size and cubic in J, highlighting the computational speedup of sparse graphs (J small), the negative impact of large q, and the serial sampling of $\boldsymbol{w}_{\mathcal{S}}$.

In terms of storage, $\boldsymbol{H}_j$ and $\boldsymbol{R}_j$ correspond to a storage requirement of $O(4Mq^2 m^2) = O(q^2 n)$. The matrix $\boldsymbol{Z}$ of size $qn \times qn$ can be represented as a list of $2M$ block-diagonal (hence sparse) $\boldsymbol{Z}_j$ matrices. Furthermore, computing $\boldsymbol{Zw}$ (dimension $n \times 1$) can be vectorized as the row-wise sum of $\boldsymbol{Z}^* \odot \boldsymbol{w}^*$ where $\boldsymbol{Z}^*$ and $\boldsymbol{w}^*$ are $n \times q$ matrices with jth column representing the jth space-time varying predictor. The cost of storing $\boldsymbol{Z}$ is thus $O(2qn)$.

Complexity is further reduced by considering a graph with small J or a finer partition resulting in large M and small m, whereas the overall time can be reduced by distributing computations on K processors. Possible choices for $\mathcal{G}$ include nearest-neighbor graphs and multiresolution trees. In settings with large q, adjusting J and M may be insufficient to reduce the computational burden. Covariance functions that are separable in the variables (but perhaps nonseparable in space and time) bring the cost of Choesky factorizations of $Jqm \times Jqm$ matrices from $O(J^3 q^3 m^3)$ to $O(J^3 m^3 + q^3)$ because $\boldsymbol{C}^{-1} = (\boldsymbol{C}_{\boldsymbol{h},u} \otimes \boldsymbol{C}_{\boldsymbol{v}})^{-1} = \boldsymbol{C}_{\boldsymbol{h},u}^{-1} \otimes \boldsymbol{C}_{\boldsymbol{v}}^{-1}$, where $\boldsymbol{C}_{\boldsymbol{h},u}$ is the $Jm \times Jm$ space-time component of the cross-covariance, and $\boldsymbol{C}_{\boldsymbol{v}}$ the $q \times q$ variable component. Savings accrue when evaluating the likelihood and in sampling from the full-conditionals at the cost of realism in describing the spatial process.

The next section develops a novel MGP design based on domain tessellations or tiling—that is, partitions of the domain forming repeated patterns—to which we associate similarly patterned meshes. If observations are also located in patterns, the bulk of the largest linear solvers will be redundant, resulting in a significant reduction in computational time. In either scenario, sampling $\boldsymbol{w}_{\mathcal{S}}$ will also proceed in parallel with improved mixing.

4. MGPs Based on Domain Tessellation or Tiling

We construct MGPs based on a tessellation or tiling of the domain. For spatial domains ($d = 2$, Figure 2), regular tiling results in triangular, quadratic, or hexagonal tessellations; mixed designs are also possible. These partition schemes can be linked to a DAG $\mathcal{G}$ by drawing directed outgoing edges starting from an originating node/tile. The same fixed pattern can be repeated over a surface of any size. In dimensions $d > 2$, which may

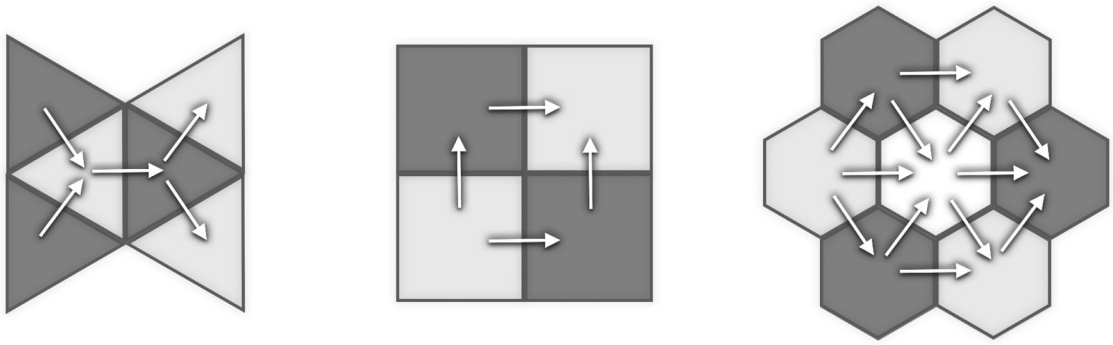

Figure 2. Regular tessellation base units and corresponding MGP graphs for spatial domains.

include time, space-filling tessellations or honeycombs can be constructed analogously, along with their corresponding meshes. Constructions of MGPs based on these ideas simply requires partitioning the locations $\mathcal{S}$ into subsets based on the chosen tessellation.

This subclass of MGP models corresponds by design to graphs with known *coloring*, with each color linked to a subgraph conditionally independent of all nodes of other colors, regardless of the dimension of the domain. This feature enables large-scale parallel sampling of $\boldsymbol{w}_{\mathcal{S}}$ and improves mixing without the need to implement heuristic graph-coloring algorithms. Furthermore, regions in a tessellated domain are typically translations and/or rotations of a single geometric shape. Carefully choosing $\mathcal{S}$, it will be possible to avoid computing the bulk of linear solvers, resulting in substantial computational gains. Subsequently, we focus on axis-parallel partitioning (quadratic or cubic tessellation) and cubic meshes, but analogous constructions and the same properties hold with other tessellation schemes.

A cubic MGP (Q-MGP) is constructed by partitioning each coordinate axis into intervals. In $d+1$ dimensions, splitting each axis into L intervals results in L^{d+1} regions. Consider a spatiotemporal domain $\mathcal{D} = \bigtimes_{r=1}^{d+1} \mathcal{D}^{(r)}$, where $\mathcal{D}^{(d+1)}$ is the time dimension. We partition each coordinate axis into L_r disjoint sets: $\mathcal{D}^{(r)} = \mathcal{I}_{r,1} \cup \cdots \cup \mathcal{I}_{r,L_r}$, where $\mathcal{I}_{r,j} \cap \mathcal{I}_{r,k} = \emptyset$ if $j \neq k$ and $\mathcal{I}_{r,s}$ denotes the sth interval in the rth coordinate axis. Solely for exposition, and without loss of generality, assume that $\mathcal{D}^{(r)} = \mathcal{I} = [0,1]$ and $L_r = L$ for $r = 1, \ldots, d+1$. Any location $\boldsymbol{\ell} = (\ell_1, \ldots, \ell_{d+1}) \in \mathcal{D}$ will be such that $\boldsymbol{\ell} \in \mathcal{I}_{1,i_1} \times \cdots \times \mathcal{I}_{d+1,i_{d+1}} = \mathcal{D}_j$ for some $i_1, \ldots, i_{d+1}$ and with $j = 1, \ldots, M$, where $M = L^{d+1}$. We refer to this axis-parallel partition scheme as a cubic tessellation and denote it by $\boldsymbol{T} = \{\mathcal{I}_{r,s}\}_{r=1,\ldots,d+1}^{s=1,\ldots,L}$. We use $\boldsymbol{T}$ to partition the reference set $\mathcal{S}$ as $\mathcal{S}_j = \mathcal{D}_j \cap \mathcal{S}$ for $j = 1, \ldots, L^{d+1}$.

Next, we define $\eta(\boldsymbol{\ell}) = (\eta_1(\boldsymbol{\ell}), \ldots, \eta_L(\boldsymbol{\ell})) \in \{1, \ldots, L\}^{d+1}$, where $\eta_j = \eta_j(\boldsymbol{\ell}) = r$ if $\ell_j \in \mathcal{I}_{j,r}$. Then, let $\mathcal{Q} = (\boldsymbol{V}, \boldsymbol{E})$ be a DAG with $\boldsymbol{V} = \boldsymbol{A} \cup \boldsymbol{B}$ and reference nodes $\boldsymbol{A} = \{\boldsymbol{a}_1, \ldots, \boldsymbol{a}_{L^{d+1}}\}$. Therefore, for any $j = 1, \ldots, L^{d+1}$ if $\boldsymbol{s} \in \mathcal{S}_j$ then $\eta(\boldsymbol{s}) = \boldsymbol{a}_j \in \boldsymbol{A} \subset \boldsymbol{V}$. We write each node $\boldsymbol{v} \in \boldsymbol{V}$ as $\boldsymbol{v} = (v_{\eta_1}, \ldots, v_{\eta_L}) \in \{1, \ldots, L\}^{d+1}$. The directed edges are constructed using a "line-of-sight" strategy. Suppose $\mathrm{Pa}[\boldsymbol{v}] = \{\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(d+1)}\}$. The hth parent of $\boldsymbol{v}$ is defined as $\boldsymbol{x}^{(h)} = (a_{\eta_1}, \ldots, a_{\eta_h} - k, \ldots, a_{\eta_L}) \cap \{1, \ldots, d+1\}^{d+1}$, where $k \geq 1$ is the smallest integer such that $\boldsymbol{x}^{(h)} \in \boldsymbol{A}$. Consequently $\boldsymbol{x}^{(h)} = \emptyset$ if $a_h = 1$. Thus, the parents of node $\boldsymbol{v} = \eta(\boldsymbol{\ell})$ are the ones that precede it along each of the $d+1$ coordinates. If $\boldsymbol{\ell} \in \mathcal{D}_j \setminus \mathcal{S}_j$, then $\eta(\boldsymbol{\ell}) = \boldsymbol{b}_j \in \boldsymbol{B}$ and $\mathrm{Pa}[\boldsymbol{b}_j] = \{\boldsymbol{a}_j\} \cup \mathrm{Pa}[\boldsymbol{a}_j]$ where $\boldsymbol{a}_j \in \boldsymbol{A}$ is a reference node. To avoid $\mathrm{Pa}[\boldsymbol{b}_j] = \emptyset$ we set $\mathrm{Pa}[\boldsymbol{b}_j] = \{\boldsymbol{x}_1^{(1)}, \boldsymbol{x}_2^{(1)}, \ldots, \boldsymbol{x}_1^{(d+1)}, \boldsymbol{x}_2^{(d+1)}\}$. The two parents along the hth dimension are $\boldsymbol{x}_1^{(h)} = a_{\eta_h} + k_1$, $\boldsymbol{x}_2^{(h)} = a_{\eta_h} - k_2$ where k_i is the smallest positive integer such that $\boldsymbol{x}_i^{(h)} \in \boldsymbol{A}$, $i = 1, 2$. In this setting $J = 2(d+1)$. The construction is finalized by fixing the cross-covariance function $\boldsymbol{C}(\boldsymbol{\ell}, \boldsymbol{\ell}')$; Figure 3 shows that the same basic structure can be immediately extended to higher dimensions, including time.

4.1. Caching Redundant Expensive Matrix Operations

The key computational bottleneck for the Gibbs sampler in Section 3.2 is calculating, for $j = 1, \ldots, 2M$, of (i) $\boldsymbol{C}_{[j]}^{-1}$ ($2MJ^3q^3m^3$ flops) and (ii) $\boldsymbol{R}_j^{-1}$, $\boldsymbol{\Sigma}_j^{*-1}$ ($4Mq^3m^3$ flops). The former is costlier than the latter by a factor of $J^3/2$. Q-MGPs are designed to greatly reduce this cost. We start with an axis-parallel tessellation of the domain in equally sized regions $\mathcal{D}_1, \ldots, \mathcal{D}_M$, storing observed locations in $\mathcal{U}$ to create $\mathcal{U}_1, \ldots, \mathcal{U}_M$, which we assume, for simplicity, to be no larger than m in size. Taking a stationary base-covariance function $\boldsymbol{C}$, implies that $\boldsymbol{C}(\mathcal{L}_1, \mathcal{L}_2) = \boldsymbol{C}(\mathcal{L}_1 + \boldsymbol{h}, \mathcal{L}_2 + \boldsymbol{h})$, where $\boldsymbol{h} \in \Re^{d+1}$ is used to shift all locations in the sets. Recall that the reference set $\mathcal{S}$ of MGPs can include unobserved locations. Hence, we can build $\mathcal{S}$ on a lattice of regularly spaced locations. Since domain partitions have the same size, we have $\mathcal{S}_j = \mathcal{S}^* + \boldsymbol{h}_j$ for $j = 1, \ldots, M$, where $\mathcal{S}^*$ is a single "prototype set" using which one can locate all other reference subsets. Also, since $\mathrm{Pa}[\boldsymbol{a}_j] \subset \mathrm{Pa}[\boldsymbol{b}_j]$, there will be $4(d+1)$ prototype sets for parents, that is, $\mathcal{S}_{\mathrm{Pa}[\boldsymbol{v}_j]} = \mathcal{S}_r^* + \boldsymbol{h}_j$ for some $r \in \{1, \ldots, 4(d+1)\}$ and $j = 1, \ldots, 2M$. Then, we can build maps $\xi_{\mathcal{S}} : \{1, \ldots, M\} \to \{1, \ldots, 4(d+1)\}$ and $\xi_{\mathcal{U}} : \{1, \ldots, M\} \to \{1, \ldots, 4(d+1)\}$ linking each of $\mathcal{S}_j$ and $\mathcal{U}_j$ to a parent prototype. This ensures that $\boldsymbol{C}_{[j]}^{-1} = \boldsymbol{C}_{\mathcal{S}_r^*}^{-1}$ for each $j = 1, \ldots, 2M$. One only needs the maps $\xi_{\mathcal{S}}$ and $\xi_{\mathcal{U}}$, *cache* the r unique inverses, and reuse them. The same method applies to cache $\boldsymbol{R}_{\mathcal{S}_j}^{-1} = \boldsymbol{R}_{\mathcal{S}_r^*}^{-1}$ on reference sets, but not on other locations since no redundancy arises in $\boldsymbol{C}_{\mathcal{U}_j}$ for $j = 1, \ldots, M$. See Figure 4 for an illustration. Compared to general MGPs (see Table 1), the number of large linear system solvers is now constant with sample size and $(d+1) \ll M$ significantly reduces computational cost.

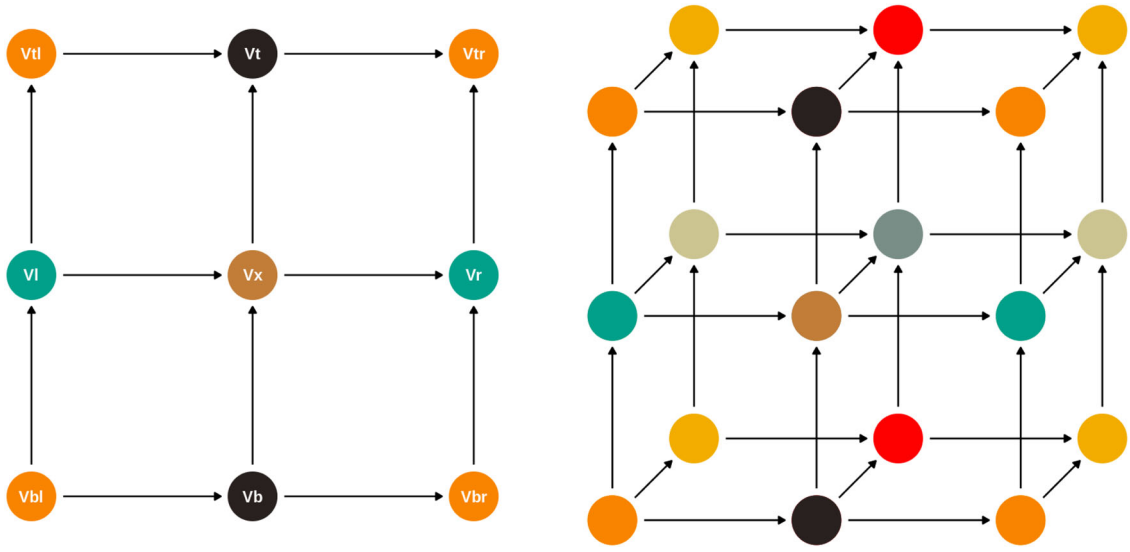

Figure 3. Q-MGP meshes used for spatial data on $d = 2$ (left) can be extended for use on spatiotemporal data $d = 3$ (right). Node colors correspond to Gibbs sampler blocks.

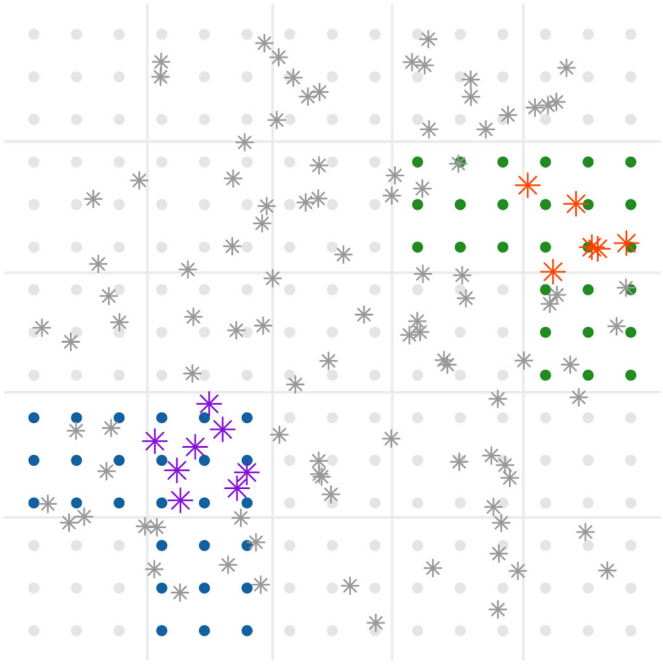

Figure 4. Visualizing redundancies: a spatial domain is partitioned in $M = 25$ regions and linked to a quadratic mesh. The reference set $\mathcal{S}$ is fixed on a regular grid, with $m = 9$. Parent locations of the orange (resp. purple) are in green (resp. blue). Using a stationary covariance, $\boldsymbol{C}_{\text{blue,blue}} = \boldsymbol{C}_{\text{green,green}}$. Therefore, only one inversion is necessary; this can be replicated at no cost across 9 of the 16 regions.

Furthermore, Q-MGPs automatically adjust to settings where observed locations $\mathcal{T}$ are on partly regular lattices, that is, they are located at patterns repeating in space or time which emerge after initial inspections of the data. Appendix G, available online, outlines a simple algorithm to identify such patterns and create maps $\xi_{\mathcal{S}}$ and $\xi_{\mathcal{U}}$. In such cases, we fix $\mathcal{S} \supseteq \mathcal{T}$ and $\mathcal{U} = \emptyset$. In addition to the above mentioned savings, we now do not have to compute $\boldsymbol{R}_{\mathcal{U}_j}^{-1}$ and $\boldsymbol{\Sigma}_{\mathcal{U}_j}^{*-1}$. If $\mathcal{T}$ is not a regular lattice over the whole domain, $4(d+1)$ is a lower bound and in general there are $M^* \ll M$ inverses to compute. If $\mathcal{T}$ is a fully observed regular lattice and if $\boldsymbol{Z}(\boldsymbol{\ell}) = I$ (a varying intercept model), then we save in computing the full conditional covariances as well, since all $\boldsymbol{D}_j = I$. See Appendix C, available online, for details on choosing $\mathcal{S}$ and $\mathcal{U}$.

4.2. Improved Mixing via Parallel Sampling

With caching, a much larger proportion of time is spent on sampling; parallelization may in general be achieved via appropriate node coloring algorithms (see, e.g., Molloy and Reed 2002; Gonzalez et al. 2011; Lewis 2016), but this step is unnecessary in Q-MGPs as the colors in $\mathcal{Q}$ are set in advance independently of the data and result in efficient parallel sampling of the latent effects. Reference nodes $\boldsymbol{A}$ of $\mathcal{Q}$ are colored to achieve independence conditional on realizations of nodes of all other colors. For example, we partition spatial domains ($d = 2$) into $M_1 \times M_2$ regions and link each region to a reference node in a quadratic

Table 1. Summary of computational cost of general MGPs and Q-MGPs.

	$\boldsymbol{C}_{[j],[j]}^{-1}$	$\boldsymbol{R}_{\mathcal{S}_j}^{-1}$	$\boldsymbol{R}_{\mathcal{U}_j}^{-1}$	$\boldsymbol{\Sigma}_{\mathcal{S}_j}^{*-1}$	$\boldsymbol{\Sigma}_{\mathcal{U}_j}^{*-1}$	Sampling $\boldsymbol{w}_{\mathcal{S}}, \boldsymbol{w}_{\mathcal{U}}$
MGPs (all cases)	$2MJ^3q^3m^3$	Mq^3m^3	Mq^3m^3	Mq^3m^3	Mq^3m^3	$MLq^2m^2 + Mq^2m^2$
Q-MGPs						
Irregular locations	$4(d+1)J^3q^3m^3$	$4(d+1)q^3m^3$	Mq^3m^3	Mq^3m^3	Mq^3m^3	$MLq^2m^2 + Mq^2m^2$
Pattern lattice w/missing	$2M^*J^3q^3m^3$	$2M^*q^3m^3$		Mq^3m^3		MLq^2m^2
Lattice w/ missing	$4(d+1)J^3q^3m^3$	$4(d+1)q^3m^3$		Mq^3m^3		MLq^2m^2
Full lattice and $\boldsymbol{Z}(\boldsymbol{\ell}) = I_q$	$4(d+1)J^3q^3m^3$	$4(d+1)q^3m^3$		$2^{(d+2)}(d+1)q^3m^3$		MLq^2m^2

NOTE: Rows are sorted from most expensive (top) to least expensive (bottom).

mesh. A "central" reference node $\boldsymbol{v}_+$ will have two parents and two children, that is, $\text{Pa}[\boldsymbol{v}_+] = \{\boldsymbol{v}_l, \boldsymbol{v}_b\}$ and $\text{Ch}[\boldsymbol{v}_+] = \{\boldsymbol{v}_r, \boldsymbol{v}_t\}$, with l, b, r, t, respectively, denoting *left*, *bottom*, *right*, *top*—refer to Figure 3 (left). We have $\text{Pa}[\boldsymbol{v}_t] = \{\boldsymbol{v}_+, \boldsymbol{v}_{tl}\}$ and $\text{Pa}[\boldsymbol{v}_r] = \{\boldsymbol{v}_+, \boldsymbol{v}_{br}\}$. The Markov blanket of $\boldsymbol{v}_+$, denoted as $\texttt{mb}(\boldsymbol{v}_+)$, is the set of neighbors of $\boldsymbol{v}_+$ in the undirected "moral" graph $\mathcal{Q}^{\mathcal{M}}$, hence $\texttt{mb}(\boldsymbol{v}_+) = \text{Pa}[\boldsymbol{v}_+] \cup \text{Ch}[\boldsymbol{v}_+] \cup \{\boldsymbol{v}_{tl}, \boldsymbol{v}_{br}\}$. The corresponding spatial process is such that $p(\boldsymbol{w}_+ \mid \boldsymbol{w} \setminus \boldsymbol{w}_+) = p(\boldsymbol{w}_+ \mid \boldsymbol{w}_{\texttt{mb}(\boldsymbol{v}_+)})$. Denoting $\boldsymbol{v}_{bl} = \text{Pa}[v_l] \cap \text{Pa}[v_b]$ and $\boldsymbol{v}_{tr} = \text{Ch}[v_r] \cap \text{Ch}[v_t]$, we note that $\{\boldsymbol{v}_{bl}, \boldsymbol{v}_{tr}\} \cap \texttt{mb}(\boldsymbol{v}_+) = \emptyset$. We partition reference nodes $\boldsymbol{A}$ into four groups $\{\boldsymbol{A}^{(1)}, \boldsymbol{A}^{(2)}, \boldsymbol{A}^{(3)}, \boldsymbol{A}^{(4)}\}$, such that $\{\boldsymbol{v}_+\} \subset \boldsymbol{A}^{(1)}$, $\{\boldsymbol{v}_b, \boldsymbol{v}_t\} \subset \boldsymbol{A}^{(2)}$, $\{\boldsymbol{v}_l, \boldsymbol{v}_r\} \subset \boldsymbol{A}^{(3)}$, and $\{\boldsymbol{v}_{tl}, \boldsymbol{v}_{tr}, \boldsymbol{v}_{bl}, \boldsymbol{v}_{br}\} \subset \boldsymbol{A}^{(4)}$. This 3×3 pattern is repeated over the whole graph. Then, if $\boldsymbol{v} \in \boldsymbol{A}^{(j)}$, $\texttt{mb}(\boldsymbol{v}) \cap \boldsymbol{A}^{(j)} = \emptyset$. Denoting by $\mathscr{D}$ the other variables in the Gibbs sampler, we get:

$$p(\boldsymbol{w}_j \mid \boldsymbol{w}_{-j}, \mathscr{D}) = p(\boldsymbol{w}_j \mid \boldsymbol{w}_{\texttt{mb}(\boldsymbol{v}_j)}, \mathscr{D}) = \prod_{\boldsymbol{v}_i \in \boldsymbol{A}^{(j)}} p(\boldsymbol{w}_i \mid \boldsymbol{w}_{\boldsymbol{A}^{(-j)}}, \mathscr{D}).$$

Since parallelization is possible within each of the groups, only be four serial steps are needed; time savings are due to $M/4$ typically being orders of magnitude larger than the number of available processors. Extensions to other tessellation schemes and higher dimensional domains and the associated graphs follow analogously.

5. Data Analysis

Satellite imaging and remote sensing data are nowadays frequently collected in large quantities and processed to be used in geology, ecology, forestry, and other fields, but clouds and atmospheric conditions obstruct aerial views and corrupt the data creating gaps. Recovery of the underlying signal and quantification of the associated uncertainty are thus the major goals to enable practitioners in the natural sciences to fully exploit these data sources. Several scalable geostatistical models based on GPs have been implemented on tens or hundreds of thousands of data points, with few exceptions. In considering larger data sizes, one must either have a large time budget—usually several days—or reduce model flexibility and richness. Scalability concerns become the single most important issue in multivariate spatiotemporal settings. In fact, repeated collection of aerial images and multiple spatially referenced predictors modeled to have a variable effect on the outcome have a multiplicative effect on data size. With no separability assumptions, the dimension of the latent spatial random effects that one may wish to recover will be huge even when individual images would be manageable when considered individually.

The lack of software to implement scalable models for spatiotemporal data makes it difficult to compare our proposed approach with others in these settings. On the other hand, a recent article (Heaton et al. 2019) pins many state-of-the-art models against each other in a spatial ($d = 2$) prediction contest. On the same data, we show in Appendix E, available online, that Q-MGPs can outperform all competitors in terms of predictive performance and coverage while using a similar computational budget.

5.1. Nonseparable Multivariate Spatiotemporal Base Covariance

In our analyses, we choose a class of multivariate space-time cross-covariances that models the covariance between variables i and j at the $(\boldsymbol{h}, u) \in \Re^{d+1}$ space-time lags as

$$\boldsymbol{C}_{ij}(\boldsymbol{h}, u) = \frac{\sigma^2}{\left(\psi_1\left(\frac{|u|^2}{\psi_2\left(\delta_{ij}^2\right)}\right)\right)^{d/2}\left(\psi_2\left(\delta_{ij}^2\right)\right)^{1/2}}\phi_1 \times \left(\frac{\|\boldsymbol{h}\|^2}{\psi_1\left(\frac{|u|^2}{\psi_2\left(\delta_{ij}^2\right)}\right)}\right), \tag{13}$$

where $\delta_{ij} > 0$ (and with $\delta_{ij} = \delta_{ji}$) is the latent dissimilarity between variables i and j. In the resulting cross-covariance function $\boldsymbol{C}(\boldsymbol{h}, u, \boldsymbol{v})$ in $\Re^{d+1+k}$, each component of the q-variate spatial process is represented by a point in a k-dimensional latent space, $k \leq q$. Refer to Apanasovich and Genton (2010) for a more in-depth discussion. We set $\phi_1(x) = \exp(-cx)$ and $\psi_j(x) = (a_j x^{\alpha_j} + 1)^{\beta_j}$, $j = 1, 2$; see Gneiting (2002) for alternatives. We also fix $\alpha_1 = \alpha_2 = \frac{1}{2}$, and seek to estimate $\boldsymbol{\theta} = (\sigma^2, c, a_1, \beta_1, a_2, \beta_2, \{\delta_{ij}\}_{i<j, j=1,\ldots,q})$ a posteriori. The usual exponential covariance arises in univariate spatial settings.

5.2. Synthetic Data

We mimick real world satellite imaging data analyzed later in Section 5.3 at a much smaller scale by generating 81 datasets from the model $\boldsymbol{y}(\boldsymbol{\ell}) = \boldsymbol{Z}(\boldsymbol{\ell})^\top \boldsymbol{w}(\boldsymbol{\ell}) + \boldsymbol{\varepsilon}(\boldsymbol{\ell})$, where $\boldsymbol{\varepsilon}(\boldsymbol{\ell}) \sim N(0, \tau^2)$ with $\boldsymbol{\ell} \in \mathcal{T}$ and $\mathcal{T}$ is a regular grid of size $40 \times 40 \times 10$, resulting in $n_{\text{all}} = 16{,}000$ total locations. We take $\boldsymbol{w} \sim \text{GP}(\boldsymbol{0}, \boldsymbol{C})$ where $\boldsymbol{C}$ is as in (13), $\psi_2 \equiv 1$ and $\sigma^2 = 1$. We generate one dataset for each combination of $\tau^2 \in \{1/1000, 1/20, 1/10\}$, temporal range $\alpha \in \{5, 50, 500\}$, space-time separability $\beta \in \left\{1/20, 1/2, 1 - \frac{1}{20}\right\}$, and spatial range $c \in \{1, 5, 25\}$.

We compare Q-MGPs with the similarly targeted Gapfill method of Gerber et al. (2018) as implemented in the R package `gapfill`. We create "synthetic clouds" of radius $\sqrt{0.1}$ and with center $(c_{1,t}, c_{2,t}) \in [0, 1/20]^2$ where $c_{1,t}, c_{2,t} \overset{\text{iid}}{\sim} U[0, 1]$ to cover the outcomes at six randomly selected times for each of the 81 datasets. Outcomes at two of the remaining four time periods were then randomly selected to be completely unobserved at all but 10 locations to avoid errors from `gapfill`. Refer to Figure 5 for an illustration.

A Q-MGP model with $M = 500$ was fit by partitioning each spatial axis into 10 intervals and the time axis into 5 intervals. The priors were $\tau^2 \sim \text{Inv.G.}(2, 1)$, $\sigma^2 \sim \text{Inv.G.}(2, 1)$, $\beta \sim U(0, 1)$, $\alpha \sim U(0, 10^4)$, $c \sim U(0, 10^4)$; 7000 iterations of Gibbs sampling were run, of which 5000 used for burn-in and thinning the remaining 2000 to obtain a posterior sample of size 1000. For each of the 81 datasets we calculate the mean absolute prediction error (MAE) and the root mean squared prediction error (RMSE). Figure 6 compares Gapfill's 90% intervals with 90% posterior equal-tailed credible intervals for the Q-MGP predictions obtained from 1000 posterior samples. In terms of

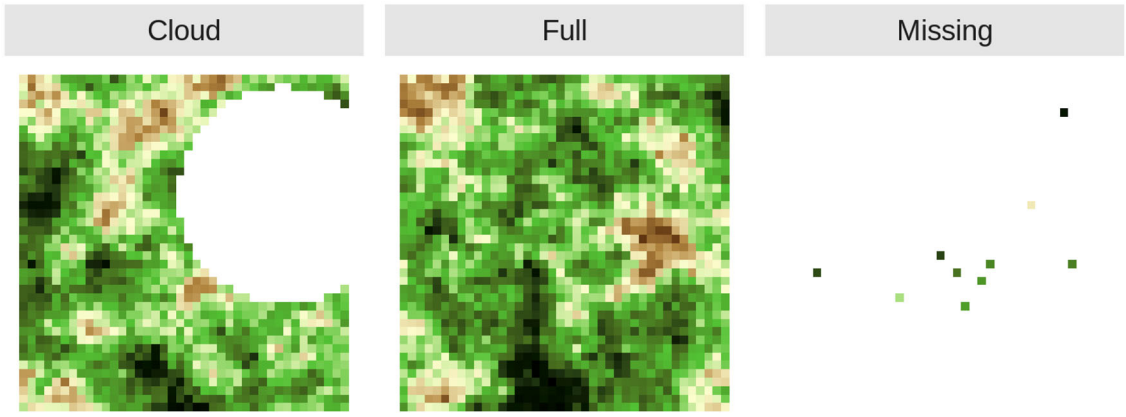

Figure 5. Artificial cloud covering in synthetic data.

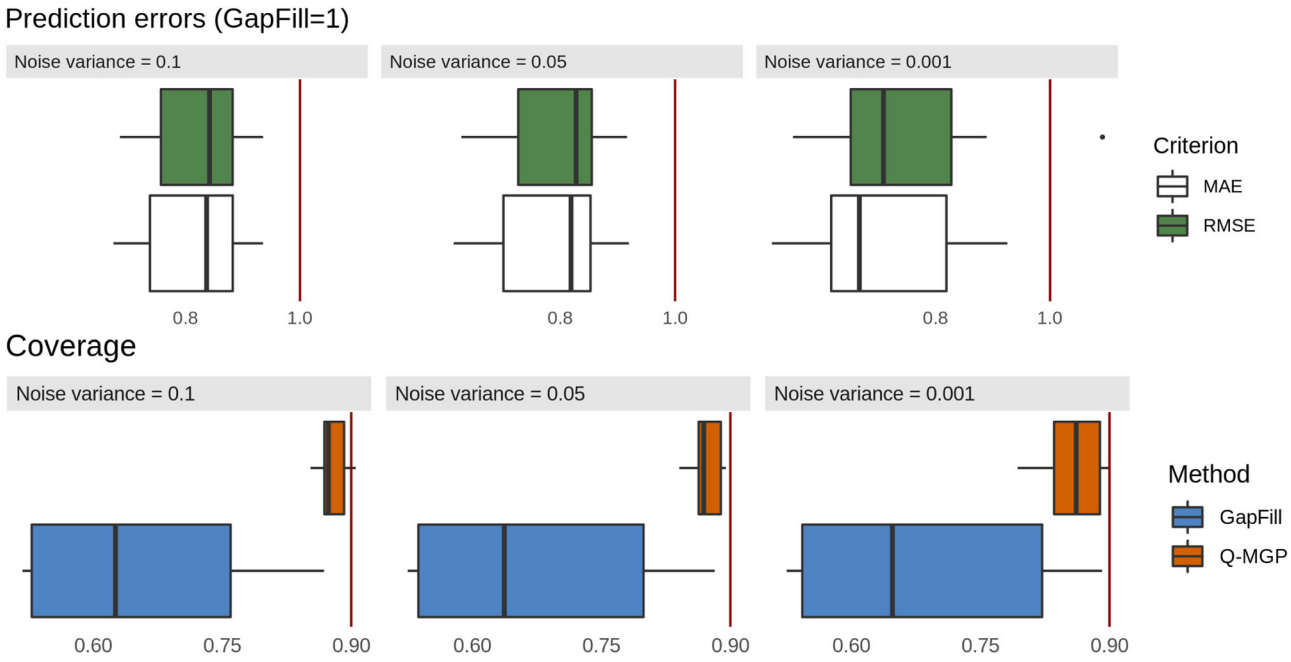

Figure 6. Performance of Q-MGP and Gapfill in out-of-sample predictions in 81 spatiotemporal datasets, at the three tested levels of noise variance τ^2.

MAE, the Q-MGP model outperformed Gapfill in all datasets; in terms of RMSE, it outperformed Gapfill in all but one dataset. The average MAE of Q-MGP was 0.4094 against Gapfill's 0.5366; the average RMSE was 0.5308 against Gapfill's 0.6820. The Q-MGP also yielded improved coverage of the prediction intervals, although some under-coverage was observed possibly due to the large M. This comparison may favor Q-MGPs as the data were generated from a GP. Appendix K, available online, confirms similar findings on non-Gaussian data (a GIF image).

5.3. NDVI Data From the Serengeti Ecosystem

Time series of NDVI derived from satellite imagery are used to understand spatial patterns in vegetation phonology. For such studies, image pixel-level NDVI values are observed over time to assess seasonal trends in vegetation green-up, growing season length and peak, and senescence. These analyses typically require NDVI values for all pixels over the region and time period of interest. As noted in the beginning of this section, atmospheric conditions, for example, cloud cover, and sensor malfunction cause missing NDVI pixel values and hence predicting such values, that is, gap-filling, is of key interest to the remote sensing community. Here, we consider NDVI data derived from the LandSat 8 sensor (which provides a $\sim 30 \times 30$ m spatial resolution pixel) taken over Serengeti National Park, Tanzania, Africa. These data were part of a larger study conducted by Desanker, Dahlin, and Finley (2020) that looked at environmental drivers in vegetation phonology change. The data cover an area of 30 km $\times$ 30 km and 34 months, and correspond to 64 images of size 1000×1000 collected at 16-day intervals. Data on NDVI are complemented with elevation and soil moisture data, for a total of three spatially referenced predictors. We are thus interested in understanding their varying effect in space and time, in addition to predicting NDVI output at missing locations. We achieve both these goals by implementing model (11), where $\boldsymbol{Z}(\boldsymbol{\ell}) = \boldsymbol{X}(\boldsymbol{\ell})$ includes the intercept and three predictors; their varying effect will be represented by $\boldsymbol{w}(\boldsymbol{\ell})$, which we recover by implementing Q-MGP models. Storing posterior samples of the multivariate spatially varying coefficients for the

full data with $q = 4$ is impossible using our computing resources as each sample would be of size $1000 \times 1000 \times 64 \times 4 = 2.56$ E+8. For this reason, we consider two feasible setups. Denote by n_{all} the number of observed and missing locations. In model (1), we subsample each image to obtain 64 frames of size 500 $\times$ 500, and fit a regression model with $\boldsymbol{Z}(\boldsymbol{\ell}) = 1$ resulting in a spatially varying intercept model on $n_{\text{obs}} = 12{,}582{,}484$ observed locations, a total of $n_{\text{all}} = 16{,}000{,}000$ locations for prediction, and a latent spatial random effect $\boldsymbol{w}$ of the same size. The Q-MGP was fit using $M = 328{,}125$ space-time regions of size ~48.

The base covariance of (13) becomes a univariate nonseparable spatiotemporal covariance as in Gneiting (2002). In model (2), we aim to estimate the varying effect of elevation on NDVI. We subsample each image to obtain 64 frames of size 278 $\times$ 278, each covering an area of 25 km $\times$ 25 km, and take $\boldsymbol{Z}(\boldsymbol{\ell}) = (1\ \boldsymbol{X}_{\text{elev}}(\boldsymbol{\ell}))$ resulting in $q = 2$ and targeting the recovery of latent effects of size 9,892,352. Considering the additional computational burden of the multivariate latent effects, in this case we used $M = 156{,}800$, corresponding to smaller space-time regions of average size ~31. In this model, there is a single unknown δ_{ij} in (13) which corresponds to the latent dissimilarity between the intercept and elevation. We thus consider $\psi_2 = (a_2\delta_{ij} + 1)^{\beta_2}$ as the unknown parameter. We assign priors $\beta_r \sim N(0, 100)$ for $r = 1, \ldots, q$, $\sigma^2 \sim$ Inv.G.$(2, 1)$, $\tau^2 \sim$ Inv.G.$(2, 1)$, and uniform priors to other covariance parameters (their support is reported in Table 2).

In both cases, approximate posterior samples of the latent random effects and the other unknown parameters were obtained by running the proposed Gibbs sampler for a total of 25,000 iterations. A posterior sample of size 500 was obtained by using the first 22,000 iterations as burn-in, and thinning the remaining 3000 by a factor of 6. Additional computational details are at Appendix F, available online. Posterior summaries of the unknown parameters for these models are reported in Table 2, along with RMSE in predicting NDVI at 10,000 left-out locations, 95% posterior coverage at those locations, and run times. Both models achieved similar out-of-sample predictive performance and coverage. Figure 7 shows the NDVI predictions of model (2) at one of the 64 time points. This reveals that the varying effect of elevation on NDVI output (see, e.g., Figure 8) is credibly different from zero at 42.54% of the space-time locations (95% CI). In particular, it highlights the extent to which higher elevation reduces vegetation. The spatial range is approximately 4 km; the time range is about 8 days. The large estimated ψ_2 indicates that the correlation between the two covariates of the latent random process is very small at all spatial and temporal lags. The predicted NDVI and latent spatiotemporal effects are supplied as animated GIF images in the supplementary materials.

Table 2. Posterior summaries of Q-MGP models implemented on the Serengeti data.

	Q-MGP model (1)	Q-MGP model (2)
n_{all}	16,000,000	4,946,176
n_{obs}	12,755,856	3,961,715
M	328,125	156,800
q	1	2
$\beta_{\text{elevation}}$	$0.0017_{(0.0014,0.0021)}$	$0.0415_{(0.0398,0.0432)}$
$\beta_{\text{topoindex}}$	$5.54\text{e-}4_{(4.72\text{e-}4,6.30\text{e-}4)}$	$-0.0011_{(-0.0012,-0.0008)}$
β_{accum}	$-4.84\text{e-}4_{(-5.66\text{e-}4,-4.02\text{e-}4)}$	$7.88\text{e-}4_{(6.94\text{e-}4,9.06\text{e-}4)}$
σ^2	$0.0585_{(0.0583,0.0587)}$	$0.0728_{(0.0711,0.0749)}$
τ^2	$1.05\text{e-}4_{(1.05\text{e-}4,1.05\text{e-}4)}$	$1.27\text{e-}4_{(1.21\text{e-}4,1.32\text{e-}4)}$
$c \sim U(0, 1\text{e+}6)$	$7.0331_{(7.0146,7.0519)}$	$3.0710_{(3.0562,3.0846)}$
$a_1 \sim U(0, 1\text{e+}6)$	$433.98_{(429.67,439.50)}$	$3857.6_{(3492.6,4154.7)}$
$\beta_1 \sim U(0, 1)$	$0.0694_{(0.0690,0.0697)}$	$0.1058_{(0.1043,0.1080)}$
$\psi_2 \sim U(0, 1\text{e+}6)$	–	$221.36_{(198.09,240.57)}$
95% coverage	94.96	95.66
RMSE	0.0175	0.0253
Time/it. (sec)	6.18	4.53
Time (hr)	42.9	31.5

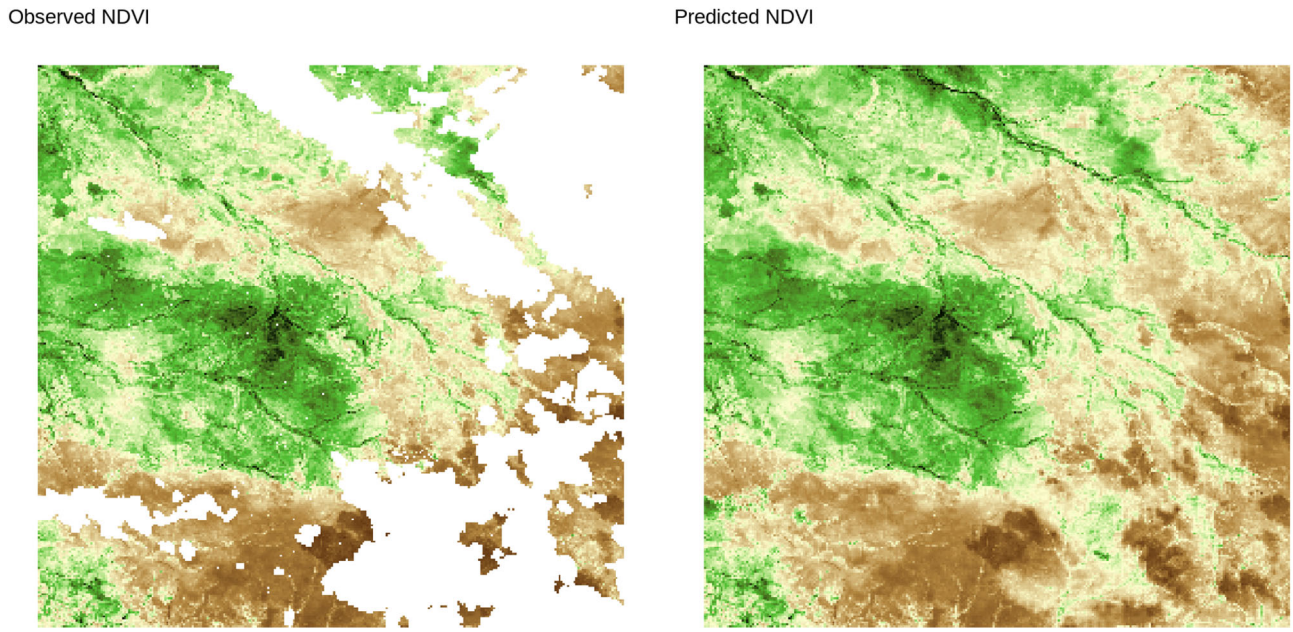

Figure 7. NDVI predictions from Q-MGP model (2) at time 60 (2016-12-17).

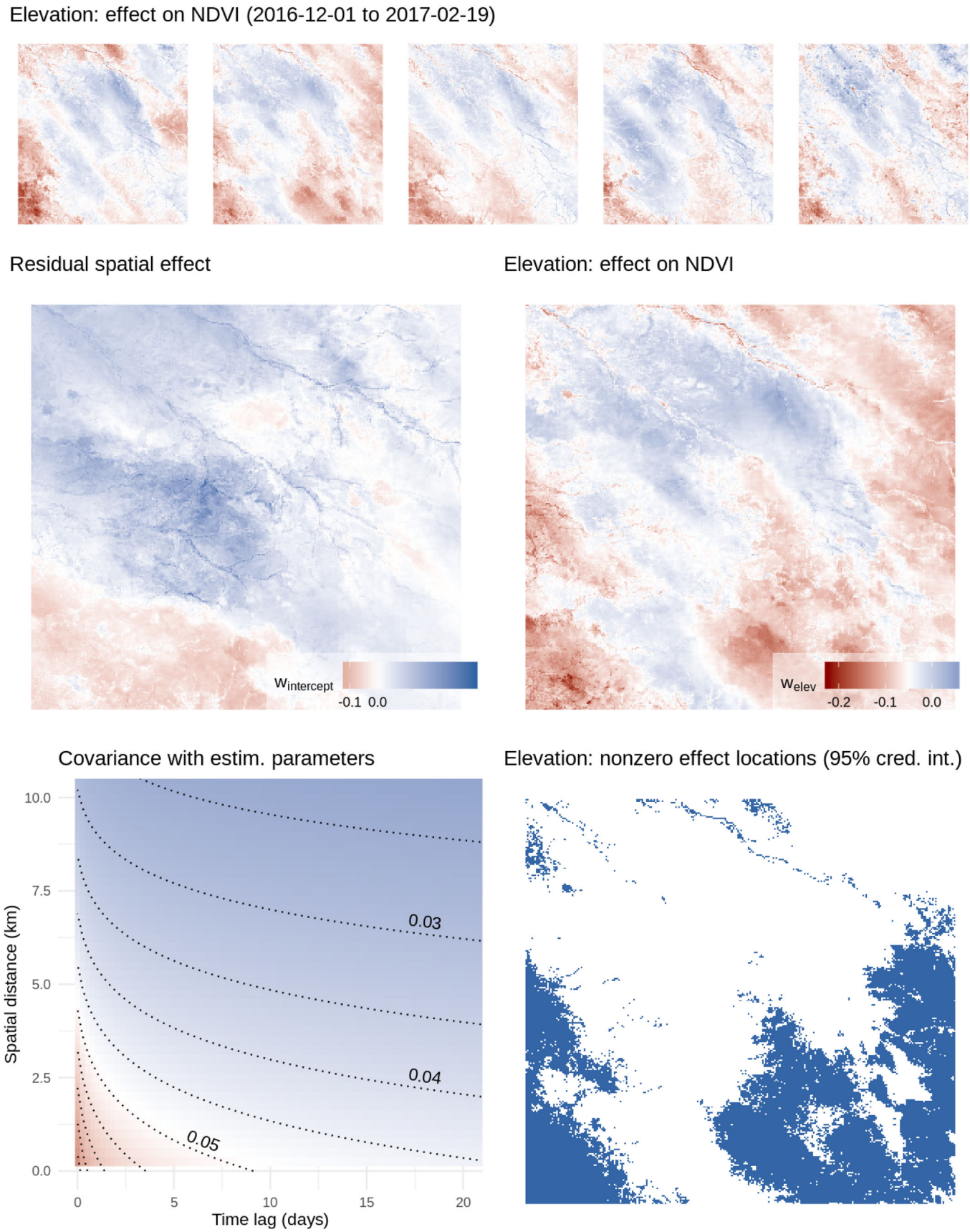

Figure 8. Top: the effect of elevation on NDVI output, evolving over five time periods. Middle left: effect on NDVI not explained by elevation; right: effect on NDVI attributable to elevation. Bottom left: Estimated covariance at different space-time lags; right, in blue: locations with credibly nonzero effect of elevation on NDVI output.

6. Discussion

We have developed a class of Bayesian hierarchical models for large spatial and spatiotemporal datasets based on linking domain partitions to DAGs. These models can be tailored for specific algorithmic needs, and we have demonstrated the advantages of using a cubic tessellation scheme (Q-MGP) when targeting the efficient recovery of spatial random effects in Bayesian hierarchical models using Gibbs samplers.

When considering alternative computational strategies, the proposed Q-MGP may not be optimal. For example, Gaussian first stage models enable marginalization of the latent spatial effects; posterior sampling of unknown covariance parameters via MCMC is typically associated by better mixing. Future research may thus focus on identifying "optimal" DAGs for collapsed samplers. Furthermore, the blocked conditional independence structure of Q-MGPs may be suboptimal as it corresponds to possibly restrictive conditional independence assumptions in neighboring locations. While we have not focused on the effect of different tessellations or partitioning choices in this article, alternative tessellation schemes (e.g., hexagonal) may be associated to less stringent assumptions and possibly better performance, while retaining all the desirable features of Q-MGP.

Other natural extensions to high-dimensional spatiotemporal statistics include settings where there are a very large number of spatiotemporal outcomes in addition to a large number of spatial and temporal locations. Here there are a few different avenues. One approach is in the same spirit of joint modeling pursued here, but instead of modeling the cross-covariance functions explicitly, as has been done here, we pursue dimension reduction using factor models (see, e.g., Christensen and Amemiya 2003; Lopes, Salazar, and Gamerman 2008; Ren and Banerjee 2013; Taylor-Rodriguez et al. 2019). The aforementioned references have attempted to model the factor models using spatial processes some of which have used scalable low-rank predictive processes or the NNGP. We believe that modeling latent factors using spatiotemporal MGPs will impart some of the computational and inferential benefits seen here. However, this will need further development especially with regard to identifiability of loading matrices (Lopes, Salazar, and Gamerman 2008; Ren and Banerjee 2013) and process parameters.

A different approach to multivariate spatial modeling has relied upon conditional or hierarchical specifications. This has been well articulated in the text by Cressie and Wikle (2011); see also Royle and Berliner (1999) and the recent developments in Cressie and Zammit-Mangion (2016). An advantage of the hierarchical approach is that the multivariate processes are valid stochastic processes, essentially by construction and without requiring spectral representations, and can also impart considerable computational benefits. It will be interesting to extend the ideas in Cressie and Zammit-Mangion (2016) to augmented spaces of DAGs to further introduce conditional independence, and therefore sparsity, in MGP models with high-dimensional outcomes.

Finally, it is worth pointing out that alternate computational algorithms, particularly tuned for high-dimensional Bayesian models, should also be explored. Recent developments on algorithms based upon classes of piecewise deterministic Markov processes (see, e.g., Fearnhead et al. 2018; Bierkens, Fearnhead, and Roberts 2019, and references therein) that avoid Gibbs samplers and even reversible MCMC algorithms are being shown to be increasingly effective for high-dimensional Bayesian inference. Adapting such algorithms to MGP and Q-MGP for scalable Bayesian spatial process models will constitute natural extensions of our current offering.

Supplementary Materials

The online supplement includes additional theoretical and computational details on Meshed Gaussian Processes, along with discussions on tessellation designs; the choice of the reference set and partition sizes; an application to multivariate outcomes; and and a comparison with other state-of-the-art scalable methods for large spatial data.

Funding

Banerjee was supported by the NSF grants DMS-1513654, IIS-1562303, and DMS-1916349; and by the National Institute of Health grants NIEHS-R01ES027027 and NIEHS-R01ES030210. Finley and Peruzzi were supported by National Science Foundation (NSF) EF-1253225 and DMS-1916395, and National Aeronautics and Space Administration's Carbon Monitoring System project. Peruzzi was supported in part by 1R01ES028804 of the National Institute of Environmental Health Sciences of the National Institutes of Health and European Union project 856506.

References

Apanasovich, T. V., and Genton, M. G. (2010), "Cross-Covariance Functions for Multivariate Random Fields Based on Latent Dimensions," *Biometrika*, 97, 15–30, DOI: 10.1093/biomet/asp078. [9]

Banerjee, S. (2017), "High-Dimensional Bayesian Geostatistics," *Bayesian Analysis*, 12, 583–614, DOI: 10.1214/17-BA1056R. [1]

——— (2020), "Modeling Massive Spatial Datasets Using a Conjugate Bayesian Linear Modeling Framework," *Spatial Statistics* (in press), DOI: 10.1016/j.spasta.2020.100417. [4]

Banerjee, S., Finley, A. O., Waldmann, P., and Ericsson, T. (2010), "Hierarchical Spatial Process Models for Multiple Traits in Large Genetic Trials," *Journal of American Statistical Association*, 105, 506–521, DOI: 10.1198/jasa.2009.ap09068. [1]

Banerjee, S., Gelfand, A. E., Finley, A. O., and Sang, H. (2008), "Gaussian Predictive Process Models for Large Spatial Data Sets," *Journal of the Royal Statistical Society*, Series B, 70, 825–848, DOI: 10.1111/j.1467-9868.2008.00663.x. [1]

Bierkens, J., Fearnhead, P., and Roberts, G. (2019), "The Zig-Zag Process and Super-Efficient Sampling for Bayesian Analysis of Big Data," *The Annals of Statistics*, 47, 1288–1320, DOI: 10.1214/18-AOS1715. [13]

Chen, Y., Davis, T. A., Hager, W. W., and Rajamanickam, S. (2008), "Algorithm 887: CHOLMOD, Supernodal Sparse Cholesky Factorization and Update/Downdate," *ACM Transactions on Mathematical Software*, 35, 1–14, DOI: 10.1145/1391989.1391995. [5]

Christensen, W. F., and Amemiya, Y. (2003), "Modeling and Prediction for Multivariate Spatial Factor Analysis," *Journal of Statistical Planning and Inference*, 115, 543–564, DOI: 10.1016/S0378-3758(02)00173-8. [13]

Cover, T. M., and Thomas, J. A. (1991), *Elements of Information Theory*, Wiley Series in Telecommunications and Signal Processing, New York: Wiley-Interscience. [4]

Cressie, N. A., and Johannesson, G. (2008), "Fixed Rank Kriging for Very Large Spatial Data Sets," *Journal of the Royal Statistical Society*, Series B, 70, 209–226, DOI: 10.1111/j.1467-9868.2007.00633.x. [1]

Cressie, N. A., and Wikle, C. K. (2011), *Statistics for Spatio-Temporal Data*, Wiley Series in Probability and Statistics, Hoboken, NJ: Wiley. [13]

Cressie, N. A., and Zammit-Mangion, A. (2016), "Multivariate Spatial Covariance Models: A Conditional Approach," *Biometrika*, 103, 915–935, DOI: 10.1093/biomet/asw045. [13]

Datta, A., Banerjee, S., Finley, A. O., and Gelfand, A. E. (2016), "Hierarchical Nearest-Neighbor Gaussian Process Models for Large Geostatistical Datasets," *Journal of the American Statistical Association*, 111, 800–812, DOI: 10.1080/01621459.2015.1044091. [2,3]

Datta, A., Banerjee, S., Finley, A. O., Hamm, N. A. S., and Schaap, M. (2016), "Nonseparable Dynamic Nearest Neighbor Gaussian Process Models for Large Spatio-Temporal Data With an Application to Particulate Matter Analysis," *The Annals of Applied Statistics*, 10, 1286–1316, DOI: 10.1214/16-AOAS931. [2]

Davis, T. A. (2006), *Direct Methods for Sparse Linear Systems*, Philadelphia, PA: SIAM. [5]

Desanker, G., Dahlin, K. M., and Finley, A. O. (2020), "Environmental Controls on Landsat-Derived Phenoregions Across an East African Megatransect," *Ecosphere*, 11, e03143. [10]

Eidsvik, J., Shaby, B. A., Reich, B. J., Wheeler, M., and Niemi, J. (2014), "Estimation and Prediction in Spatial Models With Block Composite Likelihoods," *Journal of Computational and Graphical Statistics*, 23, 295–315, DOI: 10.1080/10618600.2012.760460. [1]

Fearnhead, P., Bierkens, J., Pollock, M., and Roberts, G. O. (2018), "Piecewise Deterministic Markov Processes for Continuous-Time Monte Carlo," *Statistical Science*, 33, 386–412, DOI: 10.1214/18-STS648. [13]

Finley, A. O., Banerjee, S., and Gelfand, A. E. (2012), "Bayesian Dynamic Modeling for Large Space-Time Datasets Using Gaussian Predictive Processes," *Journal of Geographical Systems*, 14, 29–47, DOI: 10.1007/s10109-011-0154-8. [1]

Finley, A. O., Datta, A., Cook, B. D., Morton, D. C., Andersen, H. E., and Banerjee, S. (2019), "Efficient Algorithms for Bayesian Nearest Neighbor Gaussian Processes," *Journal of Computational and Graphical Statistics*, 28, 401–414, DOI: 10.1080/10618600.2018.1537924. [2,3,5]

Furrer, R., Genton, M. G., and Nychka, D. (2006), "Covariance Tapering for Interpolation of Large Spatial Datasets," *Journal of Computational and Graphical Statistics*, 15, 502–523, DOI: 10.1198/106186006X132178. [1]

Genton, M. G., and Kleiber, W. (2015), "Cross-Covariance Functions for Multivariate Geostatistics," *Statistical Science*, 30, 147–163, DOI: 10.1214/14-STS487. [4]

Gerber, F., Furrer, R., Schaepman-Strub, G., de Jong, R., and Schaepman, M. E. (2018), "Predicting Missing Values in Spatio-Temporal Remote Sensing Data," *IEEE Transactions on Geoscience and Remote Sensing*, 56, 2841–2853, DOI: 10.1109/TGRS.2017.2785240. [9]

Gneiting, T. (2002), "Nonseparable, Stationary Covariance Functions for Space-Time Data," *Journal of the American Statistical Association*, 97, 590–600, DOI: 10.1198/016214502760047113. [9,11]

Gonzalez, J., Low, Y., Gretton, A., and Guestrin, C. (2011), "Parallel Gibbs Sampling: From Colored Fields to Thin Junction Trees," in *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, Proceedings of Machine Learning Research (Vol. 15), eds. G. Gordon, D. Dunson, and M. Dudík, PMLR, Fort Lauderdale, FL, USA, pp. 324–332. [8]

Gramacy, R. B., and Apley, D. W. (2015), "Local Gaussian Process Approximation for Large Computer Experiments," *Journal of Computational and Graphical Statistics*, 24, 561–578, DOI: 10.1080/10618600.2014.914442. [3]

Gramacy, R. B., and Lee, H. K. H. (2008), "Bayesian Treed Gaussian Process Models With an Application to Computer Modeling," *Journal of the American Statistical Association*, 103, 1119–1130, DOI: 10.1198/016214508000000689. [3]

Guhaniyogi, R., Finley, A. O., Banerjee, S., and Gelfand, A. E. (2011), "Adaptive Gaussian Predictive Process Models for Large Spatial Datasets," *Environmetrics*, 22, 997–1007, DOI: 10.1002/env.1131. [1]

Guinness, J. (2018), "Permutation and Grouping Methods for Sharpening Gaussian Process Approximations," *Technometrics*, 60, 415–429, DOI: 10.1080/00401706.2018.1437476. [2,3,4]

Heaton, M. J., Datta, A., Finley, A. O., Furrer, R., Guinness, J., Guhaniyogi, R., Gerber, F., Gramacy, R. B., Hammerling, D., Katzfuss, M., Lindgren, F., Nychka, D. W., Sun, F., and Zammit-Mangion, A. (2019), "A Case Study Competition Among Methods for Analyzing Large Spatial Data," *Journal of Agricultural, Biological and Environmental Statistics*, 24, 398–425, DOI: 10.1007/s13253-018-00348-w. [9]

Katzfuss, M. (2017), "A Multi-Resolution Approximation for Massive Spatial Datasets," *Journal of the American Statistical Association*, 112, 201–214, DOI: 10.1080/01621459.2015.1123632. [1,3]

Katzfuss, M., and Guinness, J. (2017), "A General Framework for Vecchia Approximations of Gaussian Processes," arXiv no. 1708.06302. [2,3]

Kaufman, C. G., Schervish, M. J., and Nychka, D. W. (2008), "Covariance Tapering for Likelihood-Based Estimation in Large Spatial Data Sets," *Journal of the American Statistical Association*, 103, 1545–1555, DOI: 10.1198/016214508000000959. [1]

Lauritzen, S., L. (1996), *Graphical Models*, Oxford: Clarendon Press. [4]

Lewis, R. (2016), *A Guide to Graph Colouring*, Cham: Springer. [8]

Lopes, H. F., Salazar, E., and Gamerman, D. (2008), "Spatial Dynamic Factor Analysis," *Bayesian Analysis*, 3, 759–792, DOI: 10.1214/08-BA329. [13]

Molloy, M., and Reed, B. (2002), *Graph Colouring and the Probabilistic Method*, Berlin, Heidelberg: Springer-Verlag. [8]

Quiroz, Z. C., Prates, M. O., and Dey, D. K. (2019), "Block Nearest Neighbor Gaussian Processes for Large Datasets," arXiv no. 1908.06437. [2]

Ren, Q., and Banerjee, S. (2013), "Hierarchical Factor Models for Large Spatially Misaligned Data: A Low-Rank Predictive Process Approach," *Biometrics*, 69, 19–30, DOI: 10.1111/j.1541-0420.2012.01832.x. [13]

Royle, J. A., and Berliner, L. M. (1999), "A Hierarchical Approach to Multivariate Spatial Modeling and Prediction," *Journal of Agricultural, Biological, and Environmental Statistics*, 1, 29–56, DOI: 10.2307/1400420. [13]

Rue, H., and Held, L. (2005), *Gaussian Markov Random Fields: Theory and Applications*, Boca Raton, FL: Chapman & Hall/CRC. [1]

Sang, H., and Huang, J. Z. (2012), "A Full Scale Approximation of Covariance Functions for Large Spatial Data Sets," *Journal of the Royal Statistical Society*, Series B, 74, 111–132, DOI: 10.1111/j.1467-9868.2011.01007.x. [1]

Stein, M. L. (2014), "Limitations on Low Rank Approximations for Covariance Matrices of Spatial Data," *Spatial Statistics*, 8, 1–19, DOI: 10.1016/j.spasta.2013.06.003. [1]

Stein, M. L., Chi, Z., and Welty, L. J. (2004), "Approximating Likelihoods for Large Spatial Data Sets," *Journal of the Royal Statistical Society*, Series B, 66, 275–296, DOI: 10.1046/j.1369-7412.2003.05512.x. [3]

Sun, Y., Li, B., and Genton, M. (2011), "Geostatistics for Large Datasets," in *Advances and Challenges in Space-Time Modelling of Natural Events*, eds. J. Montero, E. Porcu, and M. Schlather, Berlin, Heidelberg: Springer-Verlag, pp. 55–77. [1]

Taylor-Rodriguez, D., Finley, A. O., Datta, A., Babcock, C., Andersen, H. E., Cook, B. D., Morton, D. C., and Banerjee, S. (2019), "Spatial Factor Models for High-Dimensional and Large Spatial Data: An Application in Forest Variable Mapping," *Statistica Sinica*, 29, 1155–1180, DOI: 10.5705/ss.202018.0005. [13]

Vecchia, A. V. (1988), "Estimation and Model Identification for Continuous Spatial Processes," *Journal of the Royal Statistical Society*, Series B, 50, 297–312, DOI: 10.1111/j.2517-6161.1988.tb01729.x. [1,2,3]