

Spatial meshing for general Bayesian multivariate models

Michele Peruzzi

*Department of Biostatistics
University of Michigan
Ann Arbor, MI 48109-2029, USA*

PERUZZI@UMICH.EDU

David B. Dunson

*Department of Statistical Science
Duke University
Durham, NC 27708-0251, USA*

DUNSON@DUKE.EDU

Editor: Debdeep Pati

Abstract

Quantifying spatial and/or temporal associations in multivariate geolocated data of different types is achievable via spatial random effects in a Bayesian hierarchical model, but severe computational bottlenecks arise when spatial dependence is encoded as a latent Gaussian process (GP) in the increasingly common large scale data settings on which we focus. The scenario worsens in non-Gaussian models because the reduced analytical tractability leads to additional hurdles to computational efficiency. In this article, we introduce Bayesian models of spatially referenced data in which the likelihood or the latent process (or both) are not Gaussian. First, we exploit the advantages of spatial processes built via directed acyclic graphs, in which case the spatial nodes enter the Bayesian hierarchy and lead to posterior sampling via routine Markov chain Monte Carlo (MCMC) methods. Second, motivated by the possible inefficiencies of popular gradient-based sampling approaches in the multivariate contexts on which we focus, we introduce the simplified manifold preconditioner adaptation (SiMPA) algorithm which uses second order information about the target but avoids expensive matrix operations. We demonstrate the performance and efficiency improvements of our methods relative to alternatives in extensive synthetic and real world remote sensing and community ecology applications with large scale data at up to hundreds of thousands of spatial locations and up to tens of outcomes. Software for the proposed methods is part of R package `meshed`, available on CRAN.

Keywords: Multivariate models, Directed acyclic graph, Gaussian process, non-Gaussian data, Markov chain Monte Carlo, Langevin algorithms.

1. Introduction

Geolocated data are routinely collected in many fields and motivate the development of geostatistical models based on Gaussian processes (GPs). GPs are appealing due to their analytical tractability, their flexibility via a multitude of covariance or kernel choices, and their ability to effectively represent and quantify uncertainty. When Gaussian distributional assumptions are appropriate, GPs may be used directly as correlation models for the multivariate response. Otherwise, flexible models of multivariate spatial association can in

principle be built via assumptions of conditional independence of the outcomes on a latent GP encoding space- and/or time-variability, regardless of data type. The poor scalability of naïve implementations of GPs to large scale data is addressed in a growing body of literature. Sun et al. (2011), Banerjee (2017) and Heaton et al. (2019) review and compare methods for big data geostatistics. Methods include low-rank approaches (Banerjee et al., 2008; Cressie and Johannesson, 2008), covariance tapering (Furrer et al., 2006; Kaufman et al., 2008), domain partitioning (Sang and Huang, 2012; Stein, 2014), local approximations (Gramacy and Apley, 2015), and composite likelihood approximations (Stein et al., 2004). In particular, a popular strategy is to assume sparsity in the Gaussian precision matrix via Gaussian random Markov fields (GRMF; Rue and Held, 2005) which can be represented as sparse undirected graphical models. Proper joint densities are a result of using directed acyclic graphs (DAG), leading to Vecchia’s approximation (Vecchia, 1988), nearest-neighbor GPs (NNGPs; Datta et al., 2016a), and generalizations (see e.g. Katzfuss, 2017; Katzfuss and Guinness, 2021). DAGs can be designed by taking a small number of “past” neighbors after choosing an arbitrary ordering of the data. In models of the response and in the conditionally-conjugate latent Gaussian case, posterior computations rely on sparse-matrix routines for scalability (Finley et al., 2019; Jurek and Katzfuss, 2020), enabling fast cross-validation (Shirota et al., 2019; Banerjee, 2020). Alternatives to sparse-matrix algorithms involve Gibbs samplers whose efficiency improves by prespecifying a DAG defined on domain partitions, resulting in spatially meshed GPs (MGPs; Peruzzi et al., 2022). These perspectives are reinforced when considering multivariate outcomes (see e.g. Zhang and Banerjee 2022; Dey et al. 2021; Peruzzi and Dunson 2022).

The literature on scalable GPs predominantly relies on Gaussian assumptions on the outcomes, but in many applied contexts these assumptions are restrictive, inflexible, or inappropriate. For example, vegetation phenologists may wish to characterize the life cycle of plants in mountainous regions using remotely sensed Leaf Area Index (LAI, a count variable) and relate it to snow cover during 8 day periods (SC, a discrete variable whose values range from 0 to 8—see e.g., Figure 1). Similarly, community ecologists are faced with spatial patterns when considering counts or dichotomous presence/absence data of several animal species (Figure 2). In this article, we address this key gap in the literature, which is how to construct arbitrary Bayesian multivariate geostatistical models which (1) may include non-Gaussian components, (2) lead to efficient computation for massive datasets.

There are considerable challenges in these contexts for efficient Bayesian computation when avoiding Gaussian distributional assumptions on the outcomes. General purpose Markov chain Monte Carlo (MCMC) methods can in principle be used to draw samples from the posterior distribution of the latent process by making local proposals within accept/reject schemes. However, due to the huge dimensionality of the parameter space, poor mixing and slow convergence are likely. For instance, random-walk Metropolis proposals are cheaply computed but lack in efficiency as they overlook the local geometry of the high dimensional posterior. Alternatively, one may consider gradient-based MCMC methods such as the Metropolis-adjusted Langevin algorithm (MALA; Roberts and Stramer 2002), Hamiltonian Monte Carlo (HMC; Duane et al. 1987; Neal 2011; Betancourt 2018) and others such as MALA and HMC on the Riemannian manifold (Girolami and Calderhead, 2011) or the no-U-turn sampler (NUTS; Hoffman and Gelman, 2014) used in the `Stan` probabilistic programming language (Carpenter et al., 2017). These methods are appealing because they

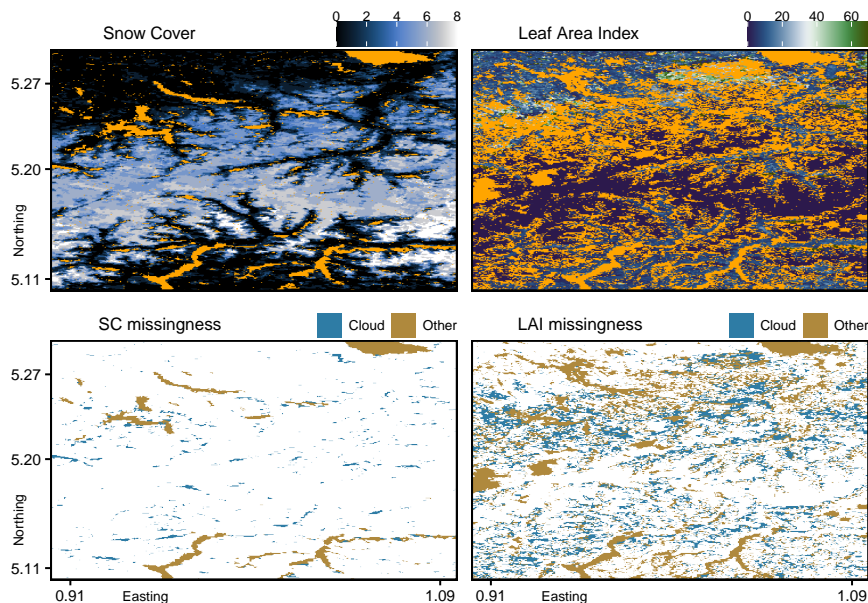


Figure 1: Snow cover (left) and Leaf Area Index, as measured by the MODIS-TERRA satellite. Missing data are in orange. Bottom maps detail the extents of cloud cover and other phenomena negatively impacting data quality.

modulate proposal step sizes using local gradient and/or higher order information of the target density. Unfortunately, their performance very rapidly drops with parameter dimension (Dunson and Johndrow, 2020). Although it is common in other contexts to rely on subsamples to cheaply approximate gradients, Johndrow et al. (2020) show that such approximate MCMC algorithms are either slow or have large approximation error. Such issues can be tackled by considering low-rank models, which facilitate the design of more efficient proposals as they involve parameters of greatly reduced dimension. Certain low-rank models endowed with conjugate full conditional distributions (Bradley et al., 2018, 2019) lead to always-accepted Gibbs proposals. However, excessive dimension reduction—which may be necessary for acceptable MCMC performance—may lead to oversmoothing of the spatial surface, overlooking the small-range variability that frequently occurs in big spatial data (Banerjee et al., 2010). Alternative dimension reduction strategies via divide-and-conquer methods that combine posterior samples obtained via MCMC from data subsets typically rely on assumptions of independence that are inappropriate in the highly correlated data settings in which we are interested (Neiswanger et al., 2014; Wang and Dunson, 2014; Wang et al., 2015b; Nemeth and Sherlock, 2018; Blomstedt et al., 2019; Mesquita et al., 2020) or have only considered univariate Gaussian likelihoods (Guhaniyogi and Banerjee, 2018).

The poor practical performance of MCMC in high dimensional settings has motivated the development of MCMC-free methods for posterior computation that take advantage of Laplace approximations (Sengupta and Cressie, 2013; Zilber and Katzfuss, 2020). In particular, the integrated nested Laplace approximation (INLA; Rue et al., 2009) iterates between Gaussian approximations of the conditional posterior of the latent effects, and numerical integrations over the hyperparameters. INLAs are accurate because of the non-negligible

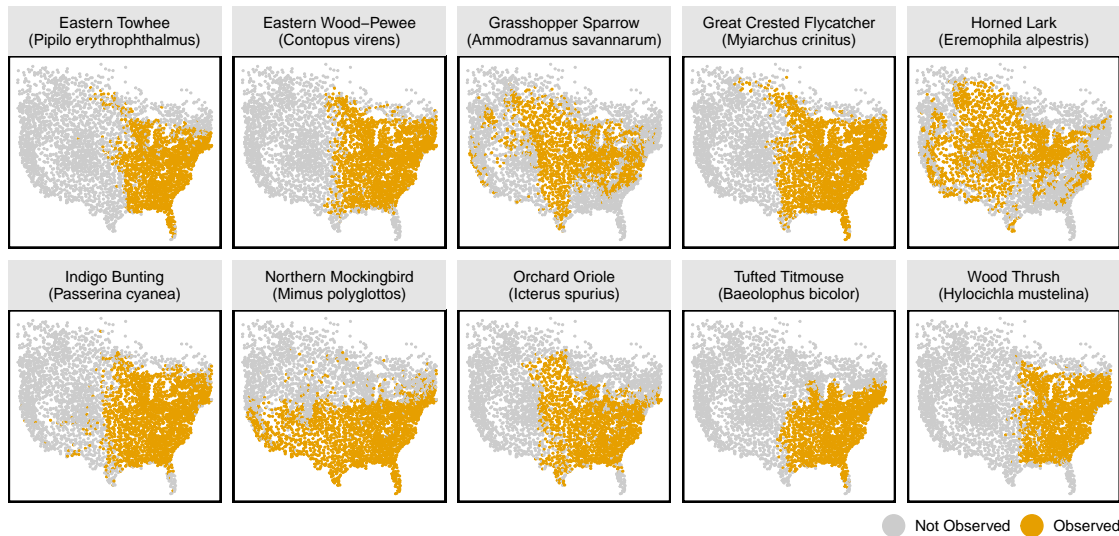


Figure 2: An extract of dichotomized North American Breeding Bird Survey data. Orange points correspond to locations at which at least 1 individual has been observed.

impact the Gaussian prior on the latent process has on its posterior; they achieve scalability to big spatial data by forcing sparsity on the Gaussian precision matrix via a GMRF assumption (Lindgren et al., 2011). INLAs are reliable alternatives to MCMC methods in several settings, but may be outperformed by carefully-designed MCMC methods in terms of accuracy or uncertainty quantification (Taylor and Diggle, 2014). Furthermore, the practical reliance of INLAs on Matérn covariance models with small dimensional hyperparameters for fast numerical integration makes them less flexible than MCMC methods in multivariate contexts or whenever special-purpose parametric covariance functions are required.

In this article, we introduce methodological and computational innovations for scalable posterior computations for general non-Gaussian spatial models. Our contributions include a class of Bayesian hierarchical models of multivariate outcomes of possibly different types based on spatial meshing of a latent multivariate process. In our treatment, outcomes can be misaligned—i.e., not all measured at all spatial locations—and relatively large in number, and there is no Gaussian assumption on the latent process. We maintain this perspective when developing posterior sampling methods. In particular, we develop a new Langevin algorithm which, based on ideas related to manifold MALA, adaptively builds a preconditioner but also avoids cubic-cost operations, leading to efficiency improvements in the contexts in which we focus. Our methods enable computations on data of size 10^5 or more. Unlike low-rank methods, we do not require restrictive dimensionality reduction at the level of the latent process. Unlike INLA, our computational methods are exact (upon convergence) for a class of valid spatial processes which is not restricted to latent GPs with Matérn covariances; furthermore, our methods are hit by a smaller computational penalty in higher-dimensional multivariate settings. Our methods are generally applicable to models of spatially referenced data, but we highlight the connections between Langevin methods and the Gibbs sampler available for Gaussian outcomes, and we develop new results for latent coregionalization models using MGPs. In applications, we consider Student-t processes,

HMC and NUTS, and other cross-covariance models as methodological and computational alternatives to latent GPs, Langevin algorithms, and coregionalization models, respectively. Software for the proposed methods and the related posterior sampling algorithms is available as part of the `meshed` package for R, available on CRAN.

The article proceeds as follows. Section 2 outlines our model for spatially-referenced multivariate outcomes of different types and introduces general purpose methods and algorithms for scaling computations to high dimensional spatial data. Section 3 outlines Langevin methods for posterior sampling of the latent process and introduces a novel algorithm for multivariate spatial models. Section 4 translates the proposed methodologies for the latent Gaussian model of coregionalization. The remaining sections highlight algorithmic efficiency in applications on large synthetic and real world datasets motivated by remote sensing and spatial community ecology. The supplementary material includes alternative constructions of our proposed methods based on latent grids, Student-t processes, and NUTS for posterior computations, in addition to proofs, practical guidelines, additional simulations, and a real world application of our methods in the context of spatial multi-species N-mixture models.

2. Meshed Bayesian multivariate models for non-Gaussian data

We introduce our model for multivariate outcomes of possibly different types (e.g. continuous and counts) which also allows for misalignment. Let $\mathcal{G} = \{\mathbf{A}, \mathbf{E}\}$ be a DAG with nodes $\mathbf{A} = \{a_1, \dots, a_M\}$ and edges $\mathbf{E} = \{\text{Par}(a) : a \in \mathbf{A}\}$, where $\text{Par}(a) \subset \mathbf{A}$ is referred to as the parent set of a . Let \mathcal{D} be the input domain and $\mathcal{S} \subset \mathcal{D}$ denote a user-specified set of “knots” or “reference locations.” We partition \mathcal{S} into subsets $\mathcal{S}_i \subset \mathcal{S}$ such that $\mathcal{S}_i \cap \mathcal{S}_j = \emptyset$ if $i \neq j$ and $\cup_{i=1}^M \mathcal{S}_i = \mathcal{S}$. Then, we set up our hierarchical model for multivariate outcomes as:

$$\begin{aligned} y_j(\boldsymbol{\ell}) \mid \eta_j(\mathbf{x}_j(\boldsymbol{\ell}), w_j(\boldsymbol{\ell})), \gamma_j &\sim F_j(\eta_j(\mathbf{x}_j(\boldsymbol{\ell}), w_j(\boldsymbol{\ell})), \gamma_j), \\ \boldsymbol{\beta}_j, \gamma_j &\sim \pi(\boldsymbol{\beta}_j, \gamma_j) \quad \boldsymbol{\theta} \sim \pi(\boldsymbol{\theta}), \quad \mathbf{w}(\cdot) \sim \Pi_{\mathcal{G}, \boldsymbol{\theta}} \end{aligned} \tag{1}$$

where F_j is the probability distribution of the j th outcome, parametrized by an unknown constant γ_j and spatially-varying function $\eta_j(\mathbf{x}_j(\boldsymbol{\ell}), w_j(\boldsymbol{\ell}))$, which includes a p_j -dimensional vector of covariates specific for the j th outcome, denoted by $\mathbf{x}_j(\boldsymbol{\ell})$, whereas $w_j(\boldsymbol{\ell})$ is the j th element of the random vector $\mathbf{w}(\boldsymbol{\ell})$, for $j = 1, \dots, q$. A common linear assumption leads to $\eta_j(\boldsymbol{\ell}) = \mathbf{x}_j(\boldsymbol{\ell})^\top \boldsymbol{\beta}_j + w_j(\boldsymbol{\ell})$. Given a set of locations $\mathcal{L} \subset \mathcal{D}$ of size $n_{\mathcal{L}}$ we denote $\mathbf{w}_{\mathcal{L}} = (\mathbf{w}(\boldsymbol{\ell}_1)^\top, \mathbf{w}(\boldsymbol{\ell}_2)^\top, \dots, \mathbf{w}(\boldsymbol{\ell}_{n_{\mathcal{L}}})^\top)^\top$. We assume $\mathbf{w}_{\mathcal{L}}$ is the finite realization at \mathcal{L} of an infinite-dimensional latent process $\mathbf{w}(\cdot)$, with law $\Pi_{\mathcal{G}}$ and density $\pi_{\mathcal{G}}$, which characterizes spatial/temporal dependence between outcomes. We construct such a process by enforcing conditional independence assumptions encoded in \mathcal{G} onto the law Π of a q -variate spatial process (also referred to as the *base* or *parent* process). For locations $\boldsymbol{\ell} \in \mathcal{S}$, we make the assumption that $\pi_{\mathcal{G}}$ factorizes according to \mathcal{G} . This means $\pi_{\mathcal{G}}(\mathbf{w}_{\mathcal{S}} \mid \boldsymbol{\theta}) = \prod_{a_i \in \mathbf{A}} \pi(\mathbf{w}_i \mid \mathbf{w}_{[i]}, \boldsymbol{\theta})$, where we denote $\mathbf{w}_i = \mathbf{w}_{\mathcal{S}_i}$ and $\mathbf{w}_{[i]}$ is the vector of $\mathbf{w}(\cdot)$ at locations $\boldsymbol{\ell} \in \cup_{a_j \in \text{Par}(a_i)} \mathcal{S}_j$ – i.e. the set of locations mapped to parents of a_i . For locations $\boldsymbol{\ell} \in \mathcal{U} = \mathcal{D} \setminus \mathcal{S}$, we assume conditional independence given a set of parents $[l] \subset \mathbf{A}$, which means $\pi_{\mathcal{G}}(\mathbf{w}_{\mathcal{U}} \mid \mathbf{w}_{\mathcal{S}}, \boldsymbol{\theta}) = \prod_{\boldsymbol{\ell} \in \mathcal{U}} \pi(\mathbf{w}(\boldsymbol{\ell}) \mid \mathbf{w}_{[l]}, \boldsymbol{\theta})$ where $\mathbf{w}_{[l]}$ is a vector collecting realizations of $\mathbf{w}(\cdot)$ at locations $\mathcal{S}_{[l]} = \cup_{a_i \in [l]} \mathcal{S}_i$.

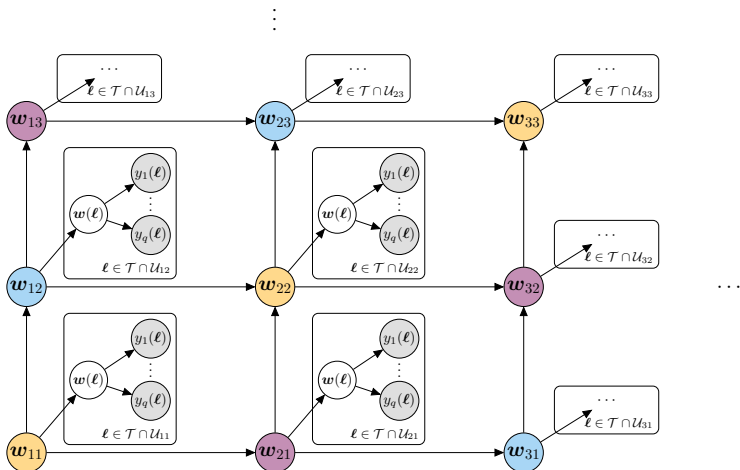


Figure 3: Directed acyclic graph representing a special case of model (1). For simplicity, we omit the directed edges from (β_j, γ_j) to each $y_j(\ell)$, $\ell \in \mathcal{T}$. If $y_j(\ell)$ is unobserved and therefore $\ell \notin \mathcal{T}_j$, the corresponding node is missing.

2.1 DAG and partition choice

We refer to the method of building spatial processes via sparse DAGs associated to domain partitioning as spatial meshing. Several options for constructing \mathcal{G} and populating and partitioning \mathcal{S} are available, but sparsity assumptions on \mathcal{G} are necessary to avoid computational bottlenecks in using $\Pi_{\mathcal{G}}$. Specifically, we restrict our focus on sparse DAGs such that $|\text{mb}(a)| \leq \bar{m}$ for all $a \in \mathbf{A}$, where $\text{mb}(a)$ is the Markov blanket of a , and \bar{m} is a small number. The Markov blanket of a node in a DAG is the set $\text{mb}(a) = \text{Par}(a) \cup \text{Chi}(a) \cup \text{Copar}(a)$ which enumerates the parents of a along with the set of children of a , $\text{Chi}(a) = \{b \in \mathbf{A} : a \in \text{Par}(b)\}$, and the set of co-parents of a , $\text{Copar}(a) = \{c \in \mathbf{A} : c \neq a \text{ and } \{a, c\} \subset \text{Par}(b) \text{ for some } b \in \text{Chi}(a)\}$ —this is the set of a 's children's other parents. We additionally assume that the undirected moral graph $\bar{\mathcal{G}}$ obtained by adding pairwise edges between co-parents has a small number of colors; if node a has color c , then no elements of $\text{mb}(a)$ have the same color. Because our assumptions on the size of the Markov blanket lead to large scale conditional independence, the spatially meshed process $\Pi_{\mathcal{G}}$ has a simpler dependence structure than the parent process Π from which it originates. The “screening” effect (Stein, 2002) makes these assumptions appealing in geostatistical contexts. Furthermore, if the Markov blanket of nodes in \mathcal{G} can be built to cover their spatial neighbors, then $\Pi_{\mathcal{G}}$ can provably accurately approximate Π in some settings (Zhu et al., 2022). If Π is a GP, the i, j entry of the resulting precision matrix is nonzero if the corresponding nodes are in their respective Markov blankets. In the context of Gibbs-like samplers that visit each node of \mathcal{G} , a small Markov blanket bounds the compute time for each step of the algorithm; we take advantage of our assumptions on step 4 of Algorithm 1. Refer to Algorithm 3 and Section D.3 in the supplement for an account of computational complexity in the coregionalized GP setting.

Figure 3 visualizes (1) when implemented on a “cubic” spatial DAG using row-column indexing of the nodes resulting in $M = M_{\text{row}} \cdot M_{\text{col}}$ and $\mathcal{S} = \cup_{i=1}^{M_{\text{row}}} \cup_{j=1}^{M_{\text{col}}} S_{ji}$. Even though

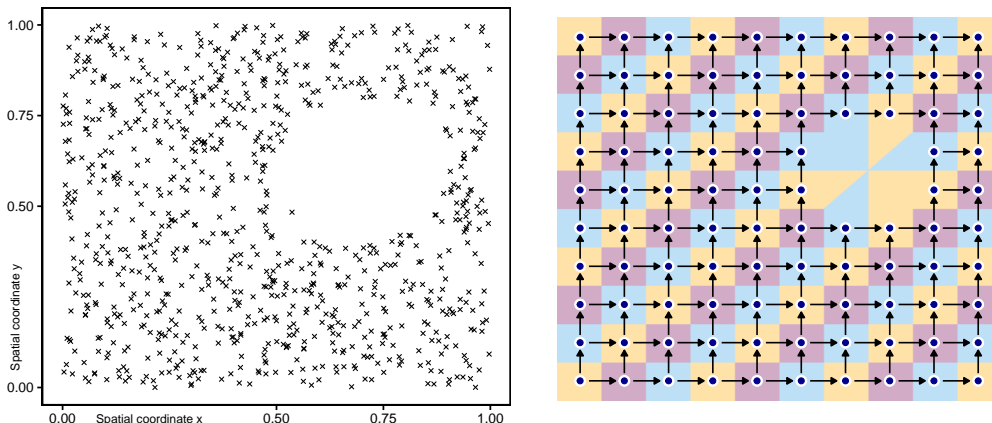


Figure 4: Visualizing cubic DAG and associated domain partitioning. Left: scatter of \mathcal{S} locations. Right: \mathcal{G} overlaid to partitions of the domain with colors matching those of $\bar{\mathcal{G}}$.

DAGs are abstract representations of conditional independence assumptions, nodes of the DAG in Figure 3 conform to a single pattern (i.e., edges from left and bottom nodes, and to right and top nodes). As a consequence, the moral graph $\bar{\mathcal{G}}$ only adds undirected edges between $a_{i+1,j}$ and $a_{i,j+1}$ for all $i = 1, \dots, M_{\text{row}} - 1$ and $j = 1, \dots, M_{\text{col}} - 1$, leading to cliques of size 3 and 3 colors, irrespective of input data. We refer to this kind of DAG as a cubic DAG as it naturally extends to a hypercube structure in $d > 2$ dimensions.

Once a sparse DAG has been set, one needs to associate each node to a partition \mathcal{S}_i of \mathcal{S} . With cubic DAGs, the i th node of \mathcal{G} can be associated to the i th domain partition found via axis-parallel tiling, or via Voronoi tessellations using a grid of centroids. These two partitioning strategies are equivalent when data have no gaps; otherwise, the latter strategy simplifies the proposal in Peruzzi et al. (2022) and can be used to guarantee that every domain partition includes observations, see e.g. Figure 4. Suppose \mathcal{D}_i , $i = 1, \dots, M$ is the chosen domain tessellation. Then, the parent set $[\ell]$ for a location $\ell \in \mathcal{U}$ can be as simple as letting $[\ell] = \mathcal{S}_i$ if $\ell \in \mathcal{U}_i = \mathcal{D}_i \setminus \mathcal{S}_i$.

This general methodology can be used to construct other processes. For instance, dropping the sparsity assumptions on \mathcal{G} , one can recover the base process itself.

Proposition 1 *If \mathcal{G} is such that for all $a_i \in \mathbf{A}$, $\text{Par}(a_i) = \{a_1, \dots, a_{i-1}\}$ then $\Pi_{\mathcal{G}} = \Pi$ at \mathcal{S} , i.e. $\pi_{\mathcal{G}}(\mathbf{w}_{\mathcal{S}}) = \pi(\mathbf{w}_{\mathcal{S}})$. The same result holds if $M = 1$.*

Proof Omitting θ for clarity, $\pi_{\mathcal{G}}(\mathbf{w}_{\mathcal{S}}) = \prod_{a_i \in \mathbf{A}} \pi(\mathbf{w}_i | \mathbf{w}_{[i]}) = \pi(\mathbf{w}_1) \prod_{i=2}^M \pi(\mathbf{w}_i | \mathbf{w}_1, \dots, \mathbf{w}_{i-1}) = \pi(\mathbf{w}_1, \dots, \mathbf{w}_M) = \pi(\mathbf{w}_{\mathcal{S}})$. If $M = 1$ then $\mathbf{A} = \{a_1\}$ and $\mathcal{S} = \mathcal{S}_1$, $\mathbf{E} = \{\emptyset\}$, and the result is immediate. \blacksquare

Several other spatial process models based on Vecchia's approximation can be derived similarly (Vecchia, 1988; Banerjee et al., 2008; Datta et al., 2016a; Katzfuss, 2017; Katzfuss and Guinness, 2021; Peruzzi and Dunson, 2022, and others) and any of these can be used in place of $\Pi_{\mathcal{G}}$. For example, a Vecchia approximation can be obtained by partitioning $\mathcal{S} = \{\ell_1, \dots, \ell_{n_{\mathcal{S}}}\}$ into sets of size 1; the sparse DAG is then generated by finding the m nearest neighbors of ℓ_i from the set $\{\ell_1, \dots, \ell_{i-1}\}$. Heuristic graph coloring algorithms can be used to ensure a degree of parallelization in Algorithm 1. Unlike in the cubic DAG

setting, the number of colors cannot be determined in advance because it is bounded below by clique size, which depends on the order of elements in \mathcal{S} and their values, and m . A larger number of colors corresponds to smaller sampling blocks and may correspond to lower MCMC efficiency when sampling latent surfaces with strong spatial correlations.

DAG and partition choice both relate to the restrictiveness of spatial conditional independence assumptions. Relative to the same partition, adding edges to a DAG brings $\Pi_{\mathcal{G}}$ closer to Π in a Kullback-Leibler (KL) sense (Peruzzi et al., 2022, Section 2), and similar reasoning informs placement of knots in recursive treed DAGs (Peruzzi and Dunson, 2022). Here, we consider a cubic DAG and alternative nested partitions. Proposition 2 shows that coarser partitions lead to smaller KL divergence of $\Pi_{\mathcal{G}}$ from the base process Π .

Proposition 2 *Consider a 2×1 domain partition $\mathbf{w} = (\mathbf{w}_1^\top, \mathbf{w}_2^\top)^\top$ and suppose \mathcal{G}_1 is a DAG with nodes $\mathbf{A}_1 = \{a_1, a_2\}$ and the edge $a_1 \rightarrow a_2$. Take a finer 3×1 partition nested in the first, i.e. we write $\mathbf{w}_2 = (\mathbf{w}_{21}^\top, \mathbf{w}_{22}^\top)^\top$, and DAG \mathcal{G}_2 such that $\mathbf{A}_2 = \{a_1, a_{21}, a_{22}\}$, edges $a_1 \rightarrow a_{21}$ and $a_{21} \rightarrow a_{22}$. Then, $KL(\pi \| \pi_{\mathcal{G}_1}) \leq KL(\pi \| \pi_{\mathcal{G}_2})$.*

Proof Since $\pi_{\mathcal{G}_1} = \pi(\mathbf{w}_1)\pi(\mathbf{w}_2 | \mathbf{w}_1) = \pi(\mathbf{w}_1)\pi(\mathbf{w}_{21} | \mathbf{w}_1)\pi(\mathbf{w}_{22} | \mathbf{w}_{21}, \mathbf{w}_1)$, the coarser partition model can be equivalently written in terms of the finer partition using the DAG \mathcal{G}_1^* with nodes $\mathbf{A}_1^* = \mathbf{A}_2$ and the additional edge $a_1 \rightarrow a_{22}$. Then, \mathcal{G}_2 is sparser than \mathcal{G}_1^* and therefore $KL(\pi \| \pi_{\mathcal{G}_1}) \leq KL(\pi \| \pi_{\mathcal{G}_2})$. \blacksquare

We provide a discussion in the supplement relating to KL comparisons between non-nested partitioning schemes.

2.2 Posterior distribution and sampling

After introducing the set $\mathcal{T}_j = \{\ell \in \mathcal{T} : y_j(\ell) \text{ is observed}\}$, we obtain $\mathcal{T}_1 \cup \dots \cup \mathcal{T}_q = \mathcal{T} = \{\ell_1, \dots, \ell_n\}$ as the set of locations at which at least one outcome is observed. Then, we denote as $\bar{\mathcal{T}} = \mathcal{T} \setminus \mathcal{S}$ the set of non-reference locations with at least one observed outcome. The posterior distribution of (1) is

$$\pi(\{\beta_j, \gamma_j\}_{j=1}^q, \mathbf{w}_{\mathcal{S}}, \mathbf{w}_{\bar{\mathcal{T}}}, \boldsymbol{\theta} | \mathbf{y}_{\mathcal{T}}) \propto \pi(\boldsymbol{\theta})\pi_{\mathcal{G}}(\mathbf{w}_{\mathcal{S}} | \boldsymbol{\theta})\pi_{\mathcal{G}}(\mathbf{w}_{\bar{\mathcal{T}}} | \mathbf{w}_{\mathcal{S}}, \boldsymbol{\theta}) \prod_{j=1}^q \pi(\beta_j, \gamma_j) \prod_{\ell \in \mathcal{T}_j} dF_j(y_j(\ell) | w_j(\ell), \beta_j, \gamma_j). \quad (2)$$

Sampling (2) may proceed via Algorithm 1, where we denote as \mathbf{y}_i the vector of observed outcomes at \mathcal{S}_i and as $\mathbf{w}_{\text{mb}(i)}$ the vector of latent effects at the Markov blanket of \mathbf{w}_i , which includes parents, children, coparents of $a_i \in A$, and all locations $\ell \in \mathcal{U}$ such that \mathbf{w}_i is part of $\mathbf{w}_{[\ell]}$. Algorithm 1 has the structure of a Gibbs sampler, as the Bayesian hierarchy is expanded to include the spatial DAG \mathcal{G} : at each step of the MCMC loop, the goal is to sample from a full conditional distribution of one random component, conditioning on the most recent value of all the others. Upon convergence, one obtains correlated samples from the target joint posterior density. The lack of conditional conjugacy at steps 1–5, which is expected given our avoidance of simplifying assumptions on F_j 's and the base process Π , implies that 1–5 will require accept/reject steps in which updating parameter \mathbf{z} proceeds by generating a move to \mathbf{z}^* via a proposal distribution $q(\cdot | \mathbf{z})$ and then accepting such move with probability $\min\{1, \frac{p(\mathbf{z}^* | -)q(\mathbf{z} | \mathbf{z}^*)}{p(\mathbf{z} | -)q(\mathbf{z}^* | \mathbf{z})}\}$ where $p(\mathbf{z} | -)$ is the target distribution to be

sampled from. Steps 1 and 2 are generally not a concern in the setting on which we focus due to the independence of (β_j, γ_j) on (β_i, γ_i) for $i \neq j$ given the latent process and the fact that the number of covariates for each outcomes is typically small relative to the data size.

Algorithm 1 Posterior sampling of spatially meshed model (1) and predictions.

Initialize $\beta_j^{(0)}$ and $\gamma_j^{(0)}$ for $j = 1, \dots, q$, $\mathbf{w}_S^{(0)}$, $\mathbf{w}_{\overline{\mathcal{T}}}^{(0)}$, and $\theta^{(0)}$
for $t \in \{1, \dots, T^*, T^* + 1, \dots, T^* + T\}$ **do** ▷ sequential MCMC loop
 1: **for** $j = 1, \dots, q$, sample $\beta_j^{(t)} \mid \mathbf{y}_{\mathcal{T}}, \mathbf{w}_{\mathcal{T}}^{(t-1)}, \gamma_j^{(t-1)}$
 2: **for** $j = 1, \dots, q$, sample $\gamma_j^{(t)} \mid \mathbf{y}_{\mathcal{T}}, \mathbf{w}_{\mathcal{T}}^{(t-1)}, \beta_j^{(t)}$
 3: sample $\theta^{(t)} \mid \mathbf{w}_{\overline{\mathcal{T}}}^{(t-1)}, \mathbf{w}_S^{(t-1)}$
 for $c \in \text{Colors}(\mathcal{G})$ **do** ▷ sequential
 for $i \in \{i : \text{Color}(a_i) = c\}$ **do in parallel**
 4: sample $\mathbf{w}_i^{(t)} \mid \mathbf{w}_{\text{mb}(i)}^{(t)}, \mathbf{y}_i, \theta^{(t)}, \{\beta_j^{(t)}, \gamma_j^{(t)}\}_{j=1}^q$ ▷ reference sampling
 for $\ell \in \overline{\mathcal{T}}$ **do in parallel**
 5: sample $\mathbf{w}(\ell)^{(t)} \mid \mathbf{w}_{[\ell]}^{(t-1)}, \mathbf{y}(\ell), \theta^{(t)}, \{\beta_j^{(t)}, \gamma_j^{(t)}\}_{j=1}^q$ ▷ non-reference sampling
 Assuming convergence has been attained after T^* iterations:
 discard $\{\beta_j^{(t)}, \gamma_j^{(t)}\}_{j=1}^q, \mathbf{w}_S^{(t)}, \mathbf{w}_{\overline{\mathcal{T}}}^{(t)}, \theta^{(t)}$ for $t = 1, \dots, T^*$
Output: Correlated sample of size T with density

$$\{\beta_j^{(t)}, \gamma_j^{(t)}\}_{j=1}^q, \mathbf{w}_S^{(t)}, \mathbf{w}_{\overline{\mathcal{T}}}^{(t)}, \theta^{(t)} \sim \pi_{\mathcal{G}}(\{\beta_j, \gamma_j\}_{j=1}^q, \mathbf{w}_S^{(t)}, \mathbf{w}_{\overline{\mathcal{T}}}^{(t)}, \theta \mid \mathbf{y}_{\mathcal{T}}).$$

Predict at $\ell^* \in \mathcal{U}$: for $t = 1, \dots, T$ and $j = 1, \dots, q$, sample from $\pi(\mathbf{w}_{\ell^*}^{(t)} \mid \mathbf{w}_{[\ell^*]}^{(t)}, \theta^{(t)})$, then from $F_j(w_j(\ell^*)^{(t)}, \beta_j^{(t)}, \gamma_j^{(t)})$

It is also typical in these settings to choose a reference set \mathcal{S} which includes all locations with at least one observed outcome, implying that $\overline{\mathcal{T}} = \emptyset$; when this is the case, step 5 is not performed in Algorithm 1. We consider alternative strategies to restore flexibility in choosing \mathcal{S} in the supplementary material. Our sparsity assumptions encoded in $\Pi_{\mathcal{G}}$ via \mathcal{G} facilitate computations at steps 3 and 4, which would otherwise be the two major computational bottlenecks. Specifically, in step 3 and assuming $\overline{\mathcal{T}} = \emptyset$, a proposal θ^* generated from a distribution $q(\cdot \mid \theta)$ is accepted with probability α

$$\alpha = \min \left\{ 1, \frac{\pi(\theta^*) \prod_{i=1}^M \pi(\mathbf{w}_i \mid \mathbf{w}_{[i]}, \theta^*) q(\theta \mid \theta^*)}{\pi(\theta) \prod_{i=1}^M \pi(\mathbf{w}_i \mid \mathbf{w}_{[i]}, \theta) q(\theta^* \mid \theta)} \right\}, \quad (3)$$

whose computation is likely expensive when \mathbf{w}_i and $\mathbf{w}_{[i]}$ are high dimensional because the base law Π models pairwise dependence of elements of \mathbf{w}_i based on their spatial distance. As an example, a GP assumption on Π leads to $\pi(\mathbf{w}_i \mid \mathbf{w}_{[i]}, \theta) = N(\mathbf{w}_i; \mathbf{H}_i, \mathbf{R}_i)$ where $\mathbf{H}_i = \mathbf{C}_{i,[i]} \mathbf{C}_{[i]}^{-1}$ and $\mathbf{R}_i = \mathbf{C}_i - \mathbf{H}_i \mathbf{C}_{[i]}$, whose computation has complexity $O(\min\{n_i^3 q^3, n_{[i]}^3 q^3\})$. If n_i or the number of parent locations $n_{[i]}$ are large, such density evaluation is computationally prohibitive. Partitioning of \mathcal{S} ensures that n_i is small for all i , and the assumed small Markov blankets of nodes in \mathcal{G} ensure that the number of parents, and thus $n_{[i]}$, is small.

Step 4 updates the latent process at each partition and is performed in two loops. The outer loop is sequential with a number of sequential steps equalling the number of colors of $\overline{\mathcal{G}}$,

which is small by construction. The inner loop can be performed in parallel or, equivalently, all partitions of the same color can be updated as a single block. In step 4, the lack of conditional conjugacy implies that proposals for \mathbf{w}_i^* for all $i = 1, \dots, M$ need to be designed and then accepted with probability α_i

$$\alpha_i = \min \left\{ 1, \frac{\pi(\mathbf{w}_i^* | \text{---}) dF(\mathbf{y}_i | \mathbf{w}_i^*, \text{---}) q(\mathbf{w}_i | \mathbf{w}_i^*)}{\pi(\mathbf{w}_i | \text{---}) dF(\mathbf{y}_i | \mathbf{w}_i, \text{---}) q(\mathbf{w}_i^* | \mathbf{w}_i)} \right\}, \quad (4)$$

where we denote the full conditional distribution of \mathbf{w}_i as $\pi(\mathbf{w}_i | \text{---})$ and the outcome densities $dF(\mathbf{y}_i | \mathbf{w}_i^*, \text{---}) = \prod_{j=1}^q \prod_{\ell_i \in \mathcal{S}_i \cap \mathcal{T}_j} dF_j(y_j(\ell_i) | w_j(\ell), \boldsymbol{\beta}_j, \gamma_j)$. Here, it is desirable to increase the size of each \mathbf{w}_i : in proposition 2 we showed that a coarser partitioning of \mathcal{S}_i leads to less restrictive spatial conditional independence assumptions. Furthermore, we may expect a smaller number of larger blocks to lead to improved sampling efficiency at step 4. However, several roadblocks appear when \mathbf{w}_i is high dimensional. Firstly, evaluating $\pi(\mathbf{w}_i^* | \text{---})/\pi(\mathbf{w}_i | \text{---})$ becomes expensive. Secondly, it is difficult to design an efficient proposal distribution $q(\cdot | \mathbf{w}_i)$ in high dimensions. A random-walk Metropolis (RWM) proposal proceeds by letting $\mathbf{w}_i^* = \mathbf{w}_i + \mathbf{g}_i$ where we let $\mathbf{g}_i \sim N(\mathbf{0}, \mathbf{G}_i)$, but the $n_i q \times n_i q$ matrix \mathbf{G}_i must be specified by the user for all i , making a RWM proposal unlikely to achieve acceptable performance in practice if n_i is large, especially if one were to take \mathbf{G}_i as diagonal matrices. Manual specification of \mathbf{G}_i 's can be circumvented via Adaptive Metropolis (AM) methods, which build \mathbf{G}_i dynamically based on past acceptances and rejections (see e.g., Haario et al., 2001; Andrieu and Thoms, 2008; Vihola, 2012), or via gradient-based schemes such as HMC, which use local information about the target distribution. However, when the dimension of \mathbf{w}_i is large the Markov chain will only make small steps and thus negatively impact overall efficiency and convergence regardless of the proposal scheme. The above mentioned issues worsen when q is larger, because spatial meshing via partitioning and a sparse DAG only operates at the level of the spatial domain.

Finally, while it is easier to specify smaller dimensional proposals, reducing the size of each \mathbf{w}_i will lead to more restrictive spatial conditional independence assumptions and poorer sampling performance due to high posterior correlations in the spatial nodes. Therefore, proposal mechanisms for updating \mathbf{w}_i should (1) be inexpensive to compute and allow for the number of outcomes to increase without overly restrictive spatial conditional independence assumptions, and (2) use local target information with minimal or no user input or tuning.

We begin detailing novel computational approaches in the next section, maintaining a general perspective. We implement our proposals on Gaussian coregionalized meshed process models and detail Algorithm 3 with an account of computational cost in terms of flops and clock time.

3. Gradient-based sampling of spatially meshed models

Algorithm 1 is essentially a Metropolis-within-Gibbs sampler for updating the latent effects $\mathbf{w}_{\mathcal{T}}$ in $M + |\overline{\mathcal{T}}|$ small dimensional substeps. The setup and tuning of efficient proposals for updating \mathbf{w}_i remains a challenge and we consider several update schemes below. Given our assumption that $\overline{\mathcal{T}} = \emptyset$, we only need to sample all \mathbf{w}_i 's conditional on their Markov blanket

(step 4). The target full conditional density, for $i = 1, \dots, M$, is

$$p(\mathbf{w}_i | \text{---}) \propto \pi(\mathbf{w}_i | \mathbf{w}_{[i]}, \boldsymbol{\theta}) \prod_{j \in \{i \rightarrow j\}} \pi(\mathbf{w}_j | \mathbf{w}_i, \mathbf{w}_{[j] \setminus \{i\}}, \boldsymbol{\theta}) \prod_{\substack{j=1, \dots, q, \\ \boldsymbol{\ell} \in \mathcal{S}_i \\ y_j(\boldsymbol{\ell}) \text{ is observed}}} dF_j(y_j(\boldsymbol{\ell}) | w_j(\boldsymbol{\ell}), \boldsymbol{\beta}_j, \gamma_j), \quad (5)$$

which takes the form $p(\mathbf{w}_i | \text{---}) \propto [i\text{'s parents}] \times [i\text{'s children}] \times [\text{data at } i]$ and where the last term is a product of one-dimensional densities due to conditional independence of the outcomes given the latent process. The update of \mathbf{w}_i proceeds by proposing a move $\mathbf{w}_i \rightarrow \mathbf{w}_i^*$ using density $q(\cdot | \mathbf{w}_i)$; then, \mathbf{w}_i^* is accepted with probability $\min\{1, \alpha\}$ where $\alpha = \frac{p(\mathbf{w}_i^* | \text{---})q(\mathbf{w}_i | \mathbf{w}_i^*)}{p(\mathbf{w}_i | \text{---})q(\mathbf{w}_i^* | \mathbf{w}_i)}$. We consider gradient-based update schemes that are accessible due to the sparsity of \mathcal{G} and the low dimensional terms in (5).

3.1 Langevin methods for meshed models

Updating $\mathbf{w}_{\mathcal{S}}$ in spatial models via a Metropolis-adjusted Langevin algorithm proceeds in general by proposing a move to \mathbf{w}_i^* for each $i = 1, \dots, M$ via

$$\begin{aligned} q(\mathbf{w}_i^* | \mathbf{w}_i) &= N(\mathbf{w}_i + \varepsilon_i^2 \mathbf{M} \nabla_{\mathbf{w}_i} \log p(\mathbf{w}_i | \text{---}) / 2, \varepsilon_i^2 \mathbf{M}), \\ \text{i.e. } \mathbf{w}_i^* &= \mathbf{w}_i + \frac{\varepsilon_i^2}{2} \mathbf{M} \nabla_{\mathbf{w}_i} \log p(\mathbf{w}_i | \text{---}) + \varepsilon_i \mathbf{M}^{\frac{1}{2}} \mathbf{u}, \end{aligned} \quad (6)$$

where $\mathbf{u} \sim N(\mathbf{0}, I_{n_i})$, I_{n_i} is the identity matrix of dimension n_i , $\nabla_{\mathbf{w}_i} p(\mathbf{w}_i | \text{---})$ denotes the gradient of the full conditional log-density $\log p(\mathbf{w}_i | \text{---})$ with respect to \mathbf{w}_i , and ε_i is a step size specific to node i which can be chosen adaptively via dual averaging (see, e.g., the discussion in Hoffman and Gelman, 2014). With (5) as the target, let \mathbf{f}_i be the $n_i q \times 1$ vector that stacks n_i blocks of size $q \times 1$; each of the n_i blocks has $\frac{\delta}{\delta w_j(\boldsymbol{\ell})} \log dF(y_j(\boldsymbol{\ell}) | w_j(\boldsymbol{\ell}), \boldsymbol{\beta}_j, \gamma_j)$ as its j th element, for $\boldsymbol{\ell} \in \mathcal{S}_i$, and zeros if $y_j(\boldsymbol{\ell})$ is unobserved. Then, we obtain

$$\nabla_{\mathbf{w}_i} \log p(\mathbf{w}_i | \text{---}) = \mathbf{f}_i + \frac{\delta}{\delta \mathbf{w}_i} \log p(\mathbf{w}_i | \mathbf{w}_{[i]}, \boldsymbol{\theta}) + \sum_{j \rightarrow \{i \rightarrow j\}} \frac{\delta}{\delta \mathbf{w}_i} \log p(\mathbf{w}_j | \mathbf{w}_i, \mathbf{w}_{[j] \setminus \{i\}}, \boldsymbol{\theta}). \quad (7)$$

The matrix \mathbf{M} in (6) is a preconditioner also referred to as the mass matrix (Neal, 2011). In the simplest setting, one sets $\mathbf{M} = I_{n_i}$ to obtain a MALA update (Roberts and Tweedie, 1996). If we assume that gradients can be computed with linear cost, MALA iterations run very cheaply in $O(qn_i)$ flops. However, we may conjecture that taking into account the geometry of the target beyond its gradient might be advantageous when seeking to formulate efficient updates. Complex update schemes that achieve this goal may operate on the Riemannian manifold (Girolami and Calderhead, 2011), but lead to an increase in the computational burden relative to simpler schemes. A special case of manifold MALA corresponding to relatively small added complexity uses a position-dependent preconditioner $\mathbf{M}_{\mathbf{w}_i} = \mathbf{G}_{\mathbf{w}_i} = \left(-E \left[\frac{\delta^2}{\delta \mathbf{w}_i^2} \log p(\mathbf{w}_i | \text{---}) \right] \right)^{-1}$. Let \mathbf{F}_i be the $n_i q \times n_i q$ diagonal matrix whose diagonal $\text{diag}(\mathbf{F}_i)$ is a $n_i q \times 1$ vector that stacks n_i blocks of size $q \times 1$; each of the n_i blocks has $-E \left[\frac{\delta^2}{\delta^2 w_j(\boldsymbol{\ell})} \log dF(y_j(\boldsymbol{\ell}) | w_j(\boldsymbol{\ell}), \boldsymbol{\beta}_j, \gamma_j) \right]$ as its j th element, for $\boldsymbol{\ell} \in \mathcal{S}_i$, and zeros if $y_j(\boldsymbol{\ell})$

is unobserved. For a target taking the form of (5) we find

$$\mathbf{G}_{\mathbf{w}_i}^{-1} = \mathbf{F}_i - \frac{\delta^2}{\delta \mathbf{w}_i^2} \log p(\mathbf{w}_i | \mathbf{w}_{[i]}, \boldsymbol{\theta}) - \sum_{j \rightarrow \{i \rightarrow j\}} \frac{\delta^2}{\delta \mathbf{w}_i^2} \log p(\mathbf{w}_j | \mathbf{w}_i, \mathbf{w}_{[j] \setminus \{i\}}, \boldsymbol{\theta}); \quad (8)$$

this choice leads to an interpretation of (6) as a *simplified* manifold MALA proposal (SM-MALA) in which the curvature of the target $p(\mathbf{w}_i | \text{---})$ is assumed constant. We make a connection between a modified SM-MALA update and the Gibbs sampler available when the latent process and all outcomes are Gaussian.

Proposition 3 *In the hierarchical model $\boldsymbol{\alpha} \sim N_k(\boldsymbol{\alpha}; \mathbf{m}_\alpha, \mathbf{V}_\alpha)$, $\mathbf{x} | \boldsymbol{\alpha}, S \sim N_n(\mathbf{x}; A\boldsymbol{\alpha}, S)$, consider the following proposal for updating $\boldsymbol{\alpha} | \mathbf{x}, S$:*

$$\boldsymbol{\alpha}^* = \boldsymbol{\alpha} + \frac{\varepsilon_1^2}{2} \mathbf{G}_\alpha \nabla_\alpha \log p(\boldsymbol{\alpha} | \text{---}) + \varepsilon_2 \mathbf{G}_\alpha^{\frac{1}{2}} \mathbf{u},$$

where $\mathbf{u} \sim N_n(0, I_n)$, and we set $\varepsilon_1 = \sqrt{2}$, $\varepsilon_2 = 1$. Then, $q(\boldsymbol{\alpha}^* | \boldsymbol{\alpha}) = p(\boldsymbol{\alpha}^* | \mathbf{x}, S)$, i.e. this modified SM-MALA proposal leads to always accepted Gibbs updates.

The proof is in the supplement, Section C. A corollary of this proposition in the context of spatially meshed models is that when $F_j(y_j(\boldsymbol{\ell}); w_j(\boldsymbol{\ell}), \boldsymbol{\beta}_j, \gamma_j) = N(y_j(\boldsymbol{\ell}); w_j(\boldsymbol{\ell}) + \mathbf{x}_j(\boldsymbol{\ell})^\top \boldsymbol{\beta}_j, \gamma_j^2)$ for all $j = 1, \dots, q$, an algorithm based on the modified SM-MALA proposal with unequal step sizes for updating \mathbf{w}_i is a Gibbs sampler. In other words, SM-MALA updates are related to a generalization of Gibbs samplers that have been shown to scale to big spatial data analyses (Datta et al., 2016a,b; Finley et al., 2019; Peruzzi et al., 2022; Peruzzi and Dunson, 2022; Peruzzi et al., 2021). With non-Gaussian outcomes, the probability of accepting the proposed \mathbf{w}_i^* depends on the ratio $q(\mathbf{w}_i | \mathbf{w}_i^*)/q(\mathbf{w}_i^* | \mathbf{w}_i)$. Computing this ratio requires $O(2q^3 n_i^3)$ floating point operations since the dimension of \mathbf{w}_i and \mathbf{w}_i^* is qn_i and one needs to compute both $\mathbf{G}_{\mathbf{w}_i}^{-\frac{1}{2}}$ and $\mathbf{G}_{\mathbf{w}_i^*}^{-\frac{1}{2}}$, e.g. via Cholesky or QR factorizations. For these reasons, SM-MALA proposals may lead to unsatisfactory performance with larger q due to their steeper compute costs relative to simpler MALA updates. We propose a novel adaptive algorithm below to overcome these issues.

3.2 Simplified Manifold Preconditioner Adaptation

Using a dense, constant preconditioner \mathbf{M} in (6) rather than the identity matrix leads to a computational cost of $O(q^2 n_i^2)$ per MCMC iteration; this cost is larger than MALA updates, but “good” choices of \mathbf{M} might improve overall efficiency. Relative to position-dependent SM-MALA updates, a constant \mathbf{M} might be convenient if q and/or n_i are large, but it is unclear how \mathbf{M} can be fixed from the outset in the context of Algorithm 1. In the context of model (1), we cannot take \mathbf{M}^{-1} as the expected Fisher information evaluated at the mode due to the high dimensionality of the latent variables and their dependence on unknown hyperparameters. Adaptive methods may build a preconditioner (or its inverse) by starting from an initial guess $\mathbf{M}_{(0)}$, then applying smaller and smaller changes to $\mathbf{M}_{(m)}$ at iteration m to get $\mathbf{M}_{(m+1)}$. Past values of \mathbf{w}_i can be used to build a preconditioner: see, e.g., Haario et al. (2001), Andrieu and Thoms (2008), Marshall and Roberts (2012) for adaptive Metropolis, and Atchadé (2006) for MALA. These methods are not immediately

advantageous because adaptation using past acceptances may be slow and lead to poor performance, especially in the within-Gibbs contexts in which we operate. Because $O(q^3 n_i^3)$ updates must be performed each time $\mathbf{M}_{(m)}$ or its inverse are updated due to the need to compute a matrix square root (e.g., Cholesky), slow adaptation methods become increasingly unappealing compared to simpler methods, like MALA, or methods that systematically construct a position-dependent preconditioner, like RM-MALA.

Algorithm 2 The m th iteration of Simplified Manifold Preconditioner Adaptation.

Setup and inputs: d -dimensional random vector $X \in \mathcal{X} \subseteq \mathfrak{R}^d$, $X \sim P$ whose density $p(\cdot) > 0$ is continuous with respect to the Lebesgue measure, assume K is a compact subset of \mathcal{X} , fix the constants $D \gg 0$, $\kappa > 0$, $0 < T^{\text{adapt}} < \infty$, step size $0 < \varepsilon < D$, denote $\mathbf{g}_{\mathbf{x}} = \nabla_{\mathbf{x}} \log p(\mathbf{x})$, $\tilde{\mathbf{g}}_{\mathbf{x}} = \mathbf{g}_{\mathbf{x}} \cdot \min \left\{ \frac{D}{\max_i \{g_{\mathbf{x}}[i]\}}, 1 \right\}$, $\mathbf{G}_{\mathbf{x}}^{-1} = -\frac{\delta^2}{\delta \mathbf{x}^2} \log p(\mathbf{x})$, $\tilde{\mathbf{G}}_{\mathbf{x}}^{-1} = \mathbf{G}_{\mathbf{x}}^{-1} \cdot \min \left\{ \frac{D}{\max_i \{\mathbf{G}_{\mathbf{x}}^{-1}[i,i]\}}, 1 \right\}$, let the sequence $(\gamma_m, m \in \mathbb{N})$ be such that $\gamma_m > 0$, $\gamma_m \downarrow 0$.

function SiMPA:

- 1: Sample $z \sim U(0, 1)$, $v \sim U(0, 1)$, $\mathbf{u} \sim N(\mathbf{0}, I_d)$.
 - 2: Let $\boldsymbol{\mu}_{(\text{new})} = \mathbf{x}_{(m-1)} + \frac{\varepsilon^2}{2} \mathbf{M}_{(m-1)} \tilde{\mathbf{g}}_{\mathbf{x}_{(m-1)}}$ and propose $\mathbf{x}_{(\text{new})} = \boldsymbol{\mu}_{(\text{new})} + \varepsilon \mathbf{M}_{(m-1)}^{\frac{1}{2}} \mathbf{u}$.
 - 3: Let $\boldsymbol{\mu}_{(\text{back})} = \mathbf{x}_{(\text{new})} + \frac{\varepsilon^2}{2} \mathbf{M}_{(m-1)} \tilde{\mathbf{g}}_{\mathbf{x}_{(\text{new})}}$.
 - 4: Compute

$$\alpha = \frac{p(\mathbf{x}_{(\text{new})})}{p(\mathbf{x}_{(m-1)})} \cdot \frac{N(\mathbf{x}_{(m-1)}; \boldsymbol{\mu}_{(\text{back})}, \varepsilon^2 \mathbf{M}_{(m-1)})}{N(\mathbf{x}_{(\text{new})}; \boldsymbol{\mu}_{(\text{new})}, \varepsilon^2 \mathbf{M}_{(m-1)})}.$$
 - 5: **if** $\alpha < v$ **and** $\|\mathbf{x}_{(\text{new})} - \mathbf{x}_{(m)}\| < D$, ▷ proposal accepted
 - 6: Set $\mathbf{x}_{(m)} = \mathbf{x}_{(\text{new})}$.
 - 7: **else:** set $\mathbf{x}_{(m)} = \mathbf{x}_{(m-1)}$. ▷ proposal rejected
 - if** $z < \gamma_m$ **and** $(\mathbf{x}_{(m)} \in K$ or $m < T^{\text{adapt}})$: ▷ adapting
 - 8: Set $\mathbf{M}_{(m)}^{-1} = \mathbf{M}_{(m-1)}^{-1} + \kappa(\tilde{\mathbf{G}}_{\mathbf{x}_{(m)}}^{-1} - \mathbf{M}_{(m-1)}^{-1})$ and compute $\mathbf{M}_{(m)}^{\frac{1}{2}}$.
 - else:** ▷ not adapting
 - 9: Set $\mathbf{M}_{(m)} = \mathbf{M}_{(m-1)}$.
-

To resolve these issues, we outline our Simplified Manifold Preconditioner Adaptation (SiMPA) as Algorithm 2. We present SiMPA in general terms as it operates independently of spatial meshing. The main feature of our algorithm is that it uses the negative Hessian matrix $\mathbf{G}_{\mathbf{x}}^{-1}$ to adaptively build a (position-independent) preconditioner. In spatially meshed models and corresponding within-Gibbs posterior sampling algorithms, $\mathbf{G}_{\mathbf{x}}$ can be computed easily using (8), also see Appendix D.2. Comparatively, an adaptive algorithm similar to Atchadé (2006), which we label YA-MALA, replaces step 8 in Algorithm 2 with $\bar{\mathbf{x}}_{(m)} = \bar{\mathbf{x}}_{(m-1)} + \kappa(\mathbf{x}_{(m)} - \bar{\mathbf{x}}_{(m-1)})$ and $\mathbf{M}_{(m)} = \mathbf{M}_{(m-1)} + \kappa(\boldsymbol{\Gamma}_m - \mathbf{M}_{(m-1)})$, where $\boldsymbol{\Gamma}_m = (\mathbf{x}_{(m)} - \bar{\mathbf{x}}_{(m-1)})(\mathbf{x}_{(m)} - \bar{\mathbf{x}}_{(m-1)})^\top + 10^{-6} I_d$ and leaves everything else the same. We show the benefits of adapting via SiMPA compared to YA-MALA in Section E.3.

In SiMPA, we reduce the number of iterations with $O(q^3 n_i^3)$ complexity by applying fixed changes to $\mathbf{M}_{(m)}$ with probability $\gamma_m \rightarrow 0$ as $m \rightarrow \infty$. As a consequence, the (expected) cost at iteration m is $O(q^2 n_i^2 + \gamma_m q^3 n_i^3)$ rather than $O(q^3 n_i^3)$. In the context of spatially meshed models, n_i is small, and the quadratic cost on q can be further reduced via coregionalization (we do so in Section 4). In our applications, we use $\gamma_m = \mathbf{1}_{(m \leq \bar{T})} + \mathbf{1}_{(m > \bar{T})} (m - \bar{T})^{-a}$, where $\mathbf{1}_A$ is the indicator for the occurrence of A , $\bar{T} < \infty$ is the number of initial iterations during

which adaptation always occurs, and $a > 0$ is the rate at which the probability of adaptation decays after \bar{T} . Small values of the parameter κ lead to $\mathbf{M}_{(m)}$ having long memory of the past.

We conservatively choose $\bar{T} = 500$, $a = 1/3$, $\kappa = 1/100$ as these values allowed ample burn-in time for all spatial nodes in all our applications. Preliminary analyses with $\bar{T} = 1000$ led to an increase in compute time with no advantage in estimation, prediction, or efficiency. On the other hand, $\bar{T} = 100$ or $a = 1/2$ resulted in lower compute times at the cost of overall performance: letting γ_m decay too quickly may lead to an inability to capture the appropriate geometry of the target density.

Because its update does not result in any increase in computational complexity, the step size ε can be changed at each step, for example via dual averaging (DA) as in Algorithms 5 and 6 of Hoffman and Gelman (2014). We use the same DA scheme when comparing SiMPA to other gradient-based sampling methods. DA involves updates to ε at each iteration $m < T^{\text{adapt}}$ and none afterwards. Because $T^{\text{adapt}} < \infty$, DA has no impact on the eventual convergence of the chain. Finally, the constant D is used to limit the jump size of the proposals as well as bound the index set for adaptation. We need D as well as additional conditions on the algorithmic behavior near the boundary of K to satisfy the containment or bounded convergence condition (Roberts and Rosenthal, 2007, 2009) that allows SiMPA to provably converge in total variation distance to the target distribution P even when the state space is not compact. Intuitively, outside of the compact K we stop adapting after iteration T^{adapt} , whereas we perform an infinite (diminishing) adaptation inside it, in order to satisfy the conditions of Theorem 21 of Craiu et al. (2015).

Proposition 4 *Suppose π is everywhere non-zero and twice differentiable so that $\mathbf{g}_{\mathbf{x}}$ and $\mathbf{G}_{\mathbf{x}}$ are well defined. Let $\varepsilon > 0$, $K \subset \mathbb{R}^d$, $D > 0$. Additionally assume that if $\mathbf{x}_{(m)} \in K$ with $\text{dist}(\mathbf{x}_{(m)}, K^c) = u$ with $0 \leq u \leq 1$ then the proposal is changed to $\mathbf{x}_{(\text{new})} \sim N(\mathbf{x}_{(m)} + \frac{\varepsilon^2}{2} \mathbf{M}_{(T^{\text{adapt}})} \tilde{\mathbf{g}}_{\mathbf{x}_{(m)}}, \varepsilon^2 \mathbf{M}_{(T^{\text{adapt}})})$. Then, Algorithm 2 converges in distribution to P .*

The proof is in the supplement, Section C. The containment condition would hold without introducing K and without specifying the behavior of the algorithm near and outside K by assuming that \mathcal{X} itself is compact, which is in principle a restrictive assumption. In practice, K can be fixed large enough so that the chain essentially never leaves it. The SiMPA preconditioner will not in general correspond to the negative Hessian computed at the mode of the target density; rather, by a law of large numbers argument it will converge to the expectation of the negative Hessian of the target density.

4. Gaussian coregionalization of multi-type outcomes

We have so far outlined general methods and sampling algorithms for big data Bayesian models on multivariate multi-type outcomes. In this section, we remain agnostic on the outcome distributions, but specify a Gaussian model of latent dependency based on coregionalization. GPs are a convenient and common modeling option for characterizing latent cross-variability. We now assume the base process law $\Pi_{\boldsymbol{\theta}}$ is a q -variate GP, i.e. $\mathbf{w}(\boldsymbol{\ell}) \sim GP(\mathbf{0}, \mathbf{C}_{\boldsymbol{\theta}}(\cdot, \cdot))$. The matrix-valued *cross-covariance* function $\mathbf{C}_{\boldsymbol{\theta}}(\cdot, \cdot)$ is parametrized by $\boldsymbol{\theta}$ and is such that $\mathbf{C}_{\boldsymbol{\theta}}(\cdot, \cdot) = [\text{cov}\{w_i(\boldsymbol{\ell}), w_j(\boldsymbol{\ell}')\}]_{i,j=1}^q$, the $q \times q$ matrix with (i, j) th element given by the covariance between $w_i(\boldsymbol{\ell})$ and $w_j(\boldsymbol{\ell}')$. $\mathbf{C}_{\boldsymbol{\theta}}(\cdot, \cdot)$ must be such that $\mathbf{C}_{\boldsymbol{\theta}}(\boldsymbol{\ell}, \boldsymbol{\ell}') = \mathbf{C}_{\boldsymbol{\theta}}(\boldsymbol{\ell}', \boldsymbol{\ell})^\top$

and $\sum_{i=1}^n \sum_{j=1}^n \mathbf{z}_i^\top \mathbf{C}_\theta(\ell_i, \ell_j) \mathbf{z}_j > 0$ for any integer n and any finite collection of points $\{\ell_1, \ell_2, \dots, \ell_n\}$ and for all $\mathbf{z}_i \in \mathbb{R}^q \setminus \{\mathbf{0}\}$ (see, e.g., Genton and Kleiber, 2015).

4.1 Coregionalized cross-covariance functions

The challenges in constructing valid cross-covariance functions can be overcome by considering a linear model of coregionalization (LMC; Matheron, 1982; Wackernagel, 2003; Schmidt and Gelfand, 2003). A stationary LMC builds q -variate processes via linear combinations of k univariate processes, i.e. $\mathbf{w}(\ell) = \sum_{h=1}^k \boldsymbol{\lambda}_h v_h(\ell) = \boldsymbol{\Lambda} \mathbf{v}(\ell)$, where $\boldsymbol{\Lambda} = [\boldsymbol{\lambda}_1, \dots, \boldsymbol{\lambda}_k]$ is a $q \times k$ full (column) rank matrix with (i, j) th entry λ_{ij} , whose i th row is denoted $\boldsymbol{\lambda}_{[i, \cdot]}$, and each $v_j(\ell)$ is a univariate spatial process with correlation function $\rho_j(\ell, \ell') = \rho(\ell, \ell'; \phi_j)$, and therefore $\boldsymbol{\theta} = (\text{vec}(\boldsymbol{\Lambda})^\top, \boldsymbol{\Phi}^\top)^\top$ where $\boldsymbol{\Phi} = (\phi_1^\top, \dots, \phi_k^\top)^\top$. Independence across the $k \leq q$ components of $\mathbf{v}(\ell)$ implies $\text{cov}\{v_j(\ell), v_h(\ell')\} = 0$ whenever $h \neq j$, and therefore $\mathbf{v}(\ell)$ is a multivariate process with diagonal cross-correlation $\boldsymbol{\rho}(\ell, \ell'; \boldsymbol{\Phi})$. As a consequence, the q -variate $\mathbf{w}(\cdot)$ process cross-covariance is defined as $\mathbf{C}_\theta(\ell, \ell') = \boldsymbol{\Lambda} \boldsymbol{\rho}(\ell, \ell'; \boldsymbol{\Phi}) \boldsymbol{\Lambda}^\top = \sum_{h=1}^k \boldsymbol{\lambda}_h \boldsymbol{\lambda}_h^\top \rho(\ell, \ell', \phi_h)$. If $\|\ell - \ell'\| = 0$, then $\mathbf{C}_\theta(\mathbf{0}) = \boldsymbol{\Lambda} \boldsymbol{\rho}(\mathbf{0}; \boldsymbol{\Phi}) \boldsymbol{\Lambda}^\top = \boldsymbol{\Lambda} \boldsymbol{\Lambda}^\top$ since $\boldsymbol{\rho}(\mathbf{0}; \boldsymbol{\Phi}) = \mathbf{I}_k$. Therefore, when $k = q$, $\boldsymbol{\Lambda}$ is identifiable e.g. as a lower-triangular matrix with positive diagonal entries corresponding to the Cholesky factorization of $\mathbf{C}_\theta(\mathbf{0})$ (see e.g., Finley et al., 2008; Zhang and Banerjee, 2022, and references therein for Bayesian LMC models). When $k < q$, a coregionalization model is interpretable as a latent spatial factor model. For a set $\mathcal{L} = \{\ell_1, \dots, \ell_n\}$ of locations, we let $\boldsymbol{\rho}_{\boldsymbol{\Phi}, \mathcal{L}}$ be the $kn \times kn$ block-matrix whose (i, j) block is $\boldsymbol{\rho}(\ell_i, \ell_j, \boldsymbol{\phi})$ —which has zero off-diagonal elements—and thus $\mathbf{C}_{\theta, \mathcal{L}} = (\mathbf{I}_n \otimes \boldsymbol{\Lambda}) \boldsymbol{\rho}_{\boldsymbol{\Phi}, \mathcal{L}} (\mathbf{I}_n \otimes \boldsymbol{\Lambda}^\top)$. Notice that the $qn \times 1$ vector $\mathbf{w}_{\mathcal{L}}$ can be represented by a $n \times q$ matrix \mathbf{W} whose j th column includes realizations of the j th margin of the q -variate process. Assuming a GP, we find $\mathbf{w}_{\mathcal{L}} = \text{vec}(\mathbf{W}^\top) \sim N(\mathbf{0}, \mathbf{C}_{\theta, \mathcal{L}})$. We can also equivalently represent process realizations by outcome rather than by location: if we let $\tilde{\mathbf{w}}_{\mathcal{L}} = \text{vec}(\mathbf{W})$ then $\tilde{\mathbf{w}}_{\mathcal{L}} \sim N(\mathbf{0}, Q \mathbf{C}_{\theta, \mathcal{L}} Q^\top)$ where Q is a permutation matrix that appropriately reorders rows of $\mathbf{C}_{\theta, \mathcal{L}}$ (and thus, Q^\top reorders its columns). We can write $Q \mathbf{C}_{\theta, \mathcal{L}} Q^\top = \tilde{\mathbf{C}}_{\theta, \mathcal{L}} = (\boldsymbol{\Lambda}^\top \otimes \mathbf{I}_n) \tilde{\boldsymbol{\rho}}_{\boldsymbol{\Phi}, \mathcal{L}} (\boldsymbol{\Lambda} \otimes \mathbf{I}_n) = (\boldsymbol{\Lambda}^\top \otimes \mathbf{I}_n) J \boldsymbol{\rho}_{\boldsymbol{\Phi}, \mathcal{L}} J^\top (\boldsymbol{\Lambda} \otimes \mathbf{I}_n)$ where J is a $nk \times nk$ permutation matrix that operates similarly to Q but on the k components of the LMC. Here, $\tilde{\boldsymbol{\rho}}_{\boldsymbol{\Phi}, \mathcal{L}}$ is a block-diagonal matrix whose j th diagonal block is $\rho_{j, \mathcal{L}}$, i.e. the j th LMC component correlation matrix at all locations. This latter representation clarifies that prior independence (i.e., a block diagonal $\tilde{\boldsymbol{\rho}}_{\boldsymbol{\Phi}, \mathcal{L}}$) does not translate to independence along the q outcome margins once the loadings $\boldsymbol{\Lambda}$ are taken into account (in fact, $\mathbf{C}_{\theta, \mathcal{L}}$ is dense).

4.2 Latent GP hierarchical model

In practice, LMCs are advantageous in allowing one to represent dependence across q outcomes via $k \ll q$ latent spatial factors. We build a multi-type outcome spatially meshed model by specifying Π in (1) as a latent Gaussian LMC model with MGP factors

$$\begin{aligned} y_j(\ell) | \eta_j(\ell), \gamma_j &\sim F_j(\eta_j(\ell), \gamma_j), \\ \eta_j(\ell) &= \mathbf{x}_j(\ell)^\top \boldsymbol{\beta}_j + \boldsymbol{\lambda}_{[j, \cdot]} \mathbf{v}(\ell), & v_h(\cdot) &\sim \text{MGP}_{\mathcal{G}}(\mathbf{0}, \rho_h(\cdot, \cdot)), h = 1, \dots, k \end{aligned} \quad (9)$$

whose posterior distribution is

$$\pi(\{\boldsymbol{\beta}_j^{(t)}, \gamma_j^{(t)}\}_{j=1}^q, \mathbf{v}_{\mathcal{T}}, \boldsymbol{\Phi}, \boldsymbol{\Lambda} \mid \mathbf{y}_{\mathcal{T}}) \propto \pi(\boldsymbol{\Phi}) \prod_{h=1}^k \prod_{i=1}^M \pi(\mathbf{v}_{h,i} \mid \mathbf{v}_{h,[i]} \phi_h) \cdot \prod_{j=1}^q \left(\pi(\boldsymbol{\beta}_j, \gamma_j) \cdot \prod_{\ell \in \mathcal{T}_j} dF_j(y_j(\ell) \mid v_j(\ell), \boldsymbol{\lambda}_{[j,:]}, \boldsymbol{\beta}_j, \gamma_j) \right). \quad (10)$$

The LMC assumption on $\mathbf{w}(\cdot)$ using MGP margins leads to computational simplifications in evaluating the density of the latent factors. For each of the M partitions, we now have a product of k independent Gaussian densities of dimension n_i rather than a single density of dimension qn_i .

4.3 Spatial meshing of Gaussian LMCs

When seeking to achieve scalability of LMCs to large scale data via spatial meshing, it is unclear whether one should act directly on the q -variate spatial process $\mathbf{w}(\cdot)$ obtained via coregionalization, or independently on each of the k LMC component processes. We now show that the two routes are equivalent with MGPs if a single DAG and a single domain partitioning scheme are used.

Algorithm 3 Posterior sampling and prediction of LMC model (1) with MGP priors.

Initialize $\boldsymbol{\beta}_j^{(0)}$, $\boldsymbol{\Lambda}^{(0)}$ and $\gamma_j^{(0)}$ for $j = 1, \dots, q$, $\mathbf{v}_S^{(0)}$, and $\boldsymbol{\Phi}^{(0)}$
for $t \in \{1, \dots, T^*, T^* + 1, \dots, T^* + T\}$ **do** ▷ sequential MCMC loop
 for $j = 1, \dots, q$, **do in parallel**
 1: use SiMPA to update $\boldsymbol{\beta}_j^{(t)}, \boldsymbol{\lambda}_{[j,:]}^{(t)} \mid \mathbf{y}_{\mathcal{T}}, \mathbf{v}_S^{(t-1)}, \gamma_j^{(t-1)}$ ▷ $O(nq(p+k)^2)$
 for $j = 1, \dots, q$, **do in parallel**
 2: use Metropolis-Hastings to update $\gamma_j^{(t)} \mid \mathbf{y}_{\mathcal{T}}, \mathbf{v}_S^{(t-1)}, \boldsymbol{\beta}_j^{(t)}, \boldsymbol{\lambda}_{[j,:]}^{(t)}$ ▷ $O(nq)$
 3: use Metropolis-Hastings to update $\boldsymbol{\Phi}^{(t)} \mid \mathbf{v}_S^{(t-1)}$ ▷ $O(nkd^3m^2)$
 for $c \in \text{Colors}(\mathcal{G})$ **do** ▷ sequential
 for $i \in \{i : \text{Color}(a_i) = c\}$ **do in parallel**
 4: use SiMPA to update $\mathbf{v}_i^{(t)} \mid \mathbf{v}_{\text{mb}(i)}^{(t)}, \mathbf{y}_i, \boldsymbol{\Lambda}^{(t)}, \boldsymbol{\Phi}^{(t)}, \{\boldsymbol{\beta}_j^{(t)}, \gamma_j^{(t)}\}_{j=1}^q$ ▷ $O(nmk^2)$
 Assuming convergence has been attained after T^* iterations:
 discard $\{\boldsymbol{\beta}_j^{(t)}, \gamma_j^{(t)}\}_{j=1}^q, \mathbf{v}_S^{(t)}, \boldsymbol{\Lambda}^{(t)}, \boldsymbol{\Phi}^{(t)}$ for $t = 1, \dots, T^*$
Output: Correlated sample of size T with density

$$\{\boldsymbol{\beta}_j^{(t)}, \gamma_j^{(t)}\}_{j=1}^q, \mathbf{v}_S^{(t)}, \boldsymbol{\Lambda}^{(t)}, \boldsymbol{\Phi}^{(t)} \sim \pi_{\mathcal{G}}(\{\boldsymbol{\beta}_j, \gamma_j\}_{j=1}^q, \mathbf{v}_S^{(t)}, \boldsymbol{\Lambda}, \boldsymbol{\Phi}, \mid \mathbf{y}_{\mathcal{T}}).$$

Predict at $\ell^* \in \mathcal{U}$: for $t = 1, \dots, T$ and $j = 1, \dots, q$, sample from $\pi(\mathbf{v}_{\ell^*}^{(t)} \mid \mathbf{v}_{[\ell^*]}^{(t)}, \boldsymbol{\Phi}^{(t)})$, then from $F_j(w_j(\ell^*)^{(t)}, \boldsymbol{\beta}_j^{(t)}, \boldsymbol{\lambda}_{[j,:]}^{(t)}, \gamma_j^{(t)})$

If the base process Π is a q -variate coregionalized GP, then for $i = 1, \dots, M$ the conditional distributions are $\pi(\mathbf{w}_i \mid \mathbf{w}_{[i]}, \boldsymbol{\theta}) = N(\mathbf{w}_i; \mathbf{H}_i \mathbf{w}_{[i]}, \mathbf{R}_i)$ where $\mathbf{H}_i = \mathbf{C}_{i,[i]} \mathbf{C}_{[i]}^{-1}$, $\mathbf{R}_i = \mathbf{C}_i - \mathbf{H}_i \mathbf{C}_{[i],i}$, and $\mathbf{C}(\ell, \ell') = \boldsymbol{\Lambda} \rho(\ell, \ell') \boldsymbol{\Lambda}^\top$ (we omit the $\boldsymbol{\theta}$ and $\boldsymbol{\Phi}$ subscripts for

simplicity). When sampling, (5) simplifies to

$$\begin{aligned}
 p(\mathbf{w}_i | \text{---}) \propto & N(\mathbf{w}_i; \mathbf{H}_i \mathbf{w}_{[i]}, \mathbf{R}_i) \prod_{j \in \{i \rightarrow j\}} N(\mathbf{w}_j; \mathbf{H}_{i \rightarrow j} \mathbf{w}_i + \mathbf{H}_{[j] \setminus \{i\}} \mathbf{w}_{[j] \setminus \{i\}}, \mathbf{R}_j) \cdot \\
 & \prod_{\substack{j=1, \dots, q, \\ \ell \in \mathcal{S}_i \\ y_j(\ell) \text{ is observed}}} dF_j(y_j(\ell) | w_j(\ell), \boldsymbol{\beta}_j, \gamma_j), \tag{11}
 \end{aligned}$$

where the notation $i \rightarrow j$ and $[j] \setminus \{i\}$ refers to the partitioning of \mathbf{H}_j by column into $\mathbf{H}_j = [\mathbf{H}_{i \rightarrow j} \ \mathbf{H}_{[j] \setminus \{i\}}]$ and thus $\mathbf{w}_{[j] \setminus \{i\}}$ corresponds to blocks of $\mathbf{w}_{[j]}$ excluding \mathbf{w}_i (i.e. the co-parents of i relative to node j). \mathbf{H}_i and \mathbf{R}_i have dimension $qn_i \times qn_{[i]}$ and $qn_i \times qn_i$, respectively. Their dimension depends on q , and the following proposition uncovers their structure.

Proposition 5 *A q -variate MGP on a fixed DAG \mathcal{G} , a domain partition \mathbf{T} , and a LMC cross-covariance function \mathbf{C}_θ is equal in distribution to a LMC model built upon k independent univariate MGPs, each of which is defined on the same \mathcal{G} and \mathbf{T} .*

The proof proceeds by showing that if $\mathbf{w}_i = (I_{n_i} \otimes \boldsymbol{\Lambda}) \mathbf{v}_i$ then $\pi(\mathbf{w}_i | \mathbf{w}_{[i]}) = \pi(\mathbf{v}_i | \mathbf{v}_{[i]})$ and that for all $i = 1, \dots, M$ we can write $\pi(\mathbf{v}_i | \mathbf{v}_{[i]}) = \prod_{h=1}^k \pi(v_i^{(h)} | v_{[i]}^{(h)})$, concluding that $\pi_{\mathcal{G}}(\mathbf{w}_{\mathcal{S}}) = \prod_{i=1}^M \pi(\mathbf{w}_i | \mathbf{w}_{[i]}) = \prod_{i=1}^M \prod_{h=1}^k \pi(v_i^{(h)} | v_{[i]}^{(h)}) = \prod_{h=1}^k \pi_{\mathcal{G}}^{(h)}(v_{\mathcal{S}}^{(h)})$ where $\pi_{\mathcal{G}}^{(h)}$ is the density of the h th independent univariate MGP using \mathcal{G} , \mathbf{T} , and correlation function $\rho_h(\cdot, \cdot)$. The complete derivation is available in the supplement. A corollary of Proposition 5 is that a *different* spatially meshed GP can be constructed via unequal spatial meshing (i.e., different graphs and partitions) along the k margins; this result intuitively says that an MGP behaves like a standard GP with respect to the construction of multivariate processes via LMCs and in other words, there is no loss in flexibility when using MGPs compared to the full GP. The supplementary material provides details on $\nabla_{\mathbf{v}_i} \log p(\mathbf{v} | \text{---})$ and $\mathbf{G}_{\mathbf{v}_i}$ for posterior sampling of the latent meshed Gaussian LMC models via Algorithm 1.

5. Applications on bivariate non-Gaussian data

We concentrate here on a scenario in which two possibly misaligned non-Gaussian outcomes are measured at a large number of spatial locations and we aim to jointly model them. We will consider a larger number of outcomes in Section 6, in the context of community ecology. In addition to the analysis presented here, the supplement (Section E) includes (1) a comparison of methods across 750 multivariate synthetic datasets, and (2) performance assessments of multiple sampling schemes in multivariate multi-type models using latent coregionalized QMGPs.

5.1 Illustration: bivariate log-Gaussian Cox processes

When modeling spatial point patterns via log-Gaussian Cox processes with the goal of estimating the intensity surface, one typically proceeds by counting occurrences within cells in a regular grid of the spatial domain. We simulate this scenario by generating a bivariate

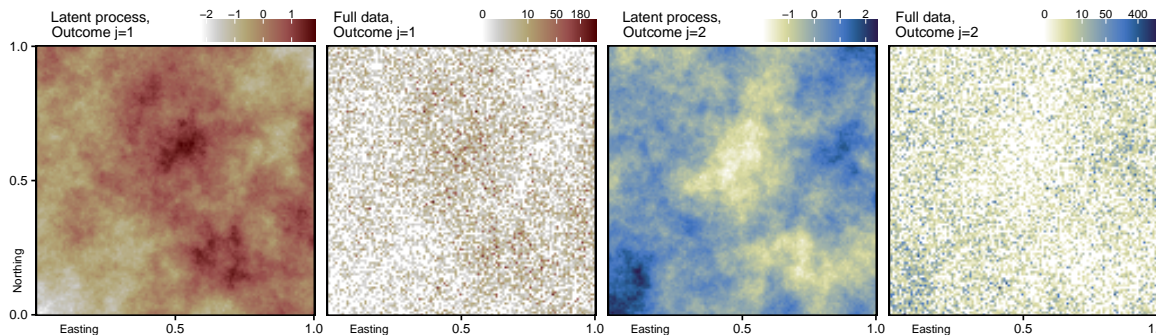


Figure 5: Latent NNGP process realization and corresponding synthetic count data at 14,400 spatial locations for correlated spatial outcomes. We omit the plots corresponding to the unrestricted GP scenario as they are visually indistinguishable.

Poisson outcome at each location of a 120×120 regular grid, for a data dimension of $qn = 28800$. In model (1), we let F_j be a Poisson distribution with intensity $\exp\{\eta_j(\boldsymbol{\ell})\}$ at $\boldsymbol{\ell} \in [0, 1]^2$, where $\eta_j(\boldsymbol{\ell}) = \mathbf{x}(\boldsymbol{\ell})^\top \boldsymbol{\beta}_j + w_j(\boldsymbol{\ell})$ is the log-intensity for count outcome j . We sample 3 correlated covariates at each location independently as $\mathbf{x}^\top(\boldsymbol{\ell}) \sim N_3(\mathbf{0}, \boldsymbol{\Sigma}_x)$ where $\boldsymbol{\Sigma}_x$ is a correlation matrix with off-diagonal elements $\sigma_{12} = 0.9, \sigma_{13} = -0.3, \sigma_{23} = -0.6$, and we let $\boldsymbol{\beta}_1 = (-0.5, -1, 0)^\top, \boldsymbol{\beta}_2 = (-1, -0.5, 0.5)^\top$. We fix the latent process Π in one scenario as a coregionalized GP and in another as a coregionalized NNGP. In both cases, $w_j(\boldsymbol{\ell}) = \boldsymbol{\lambda}_{[j,:]} \mathbf{v}(\boldsymbol{\ell})$ and $\boldsymbol{\Lambda} \boldsymbol{\Lambda}^\top = (\sigma_{ij})_{i,j=1,2}$ where $\sigma_{11} = 4, \sigma_{12} = \sigma_{21} = -1.3, \sigma_{22} = 1$, which yields a latent cross-correlation between the two outcomes of $\rho = -0.65$; the two spatial correlations used in the LMC model are $\rho_h(\|\boldsymbol{\ell} - \boldsymbol{\ell}'\|) = \exp\{-\phi_h \|\boldsymbol{\ell} - \boldsymbol{\ell}'\|\}$ and we let $\phi_1 = 1.5, \phi_2 = 2.5$. We use R package `GpGp` to generate an NNGP using maxmin ordering of the spatial locations and 10 neighbors. We depict the latent NNGP process along with the full data in Figure 5. We introduce missing values at 20% of the spatial locations, independently for each outcome. As a result, our training data are misaligned.

We investigate the comparative performance of several coregionalized QMGP variants computed via MALA, SM-MALA, SiMPA and NUTS. We also consider latent multivariate Student-t processes (Chen et al. 2020; Shah et al. 2014) using an alternative cross-covariance specification based on Apanasovich and Genton (2010)—in short “AG10”—and previously used in Peruzzi et al. (2022), which we also implement in the meshed Gaussian case. We detail the specifics of spatial meshing and gradient-based sampling for Student-t processes in Section F. To the best of our knowledge, ours is the first implementation of a scalable Student-t process using DAGs. We also compare with a data transformation method based on NNGPs: for each outcome, we use $y^* = \log(1 + y)$, then fit NNGP models of the response on each outcome independently. All MCMC-based results are based on chains of length 30,000. All gradient-based methods share the dual averaging setup for adapting the step size ε and are thus allowed the same burn-in period. Finally, we implement an MCMC-free stochastic partial differential equations method (SPDE; Lindgren et al., 2011) fit via INLA. The SiMPA-estimated posterior means for the latent process, predictions across the spatial domain, as well as the width of 95% CIs about the linear predictors are reported in Figure

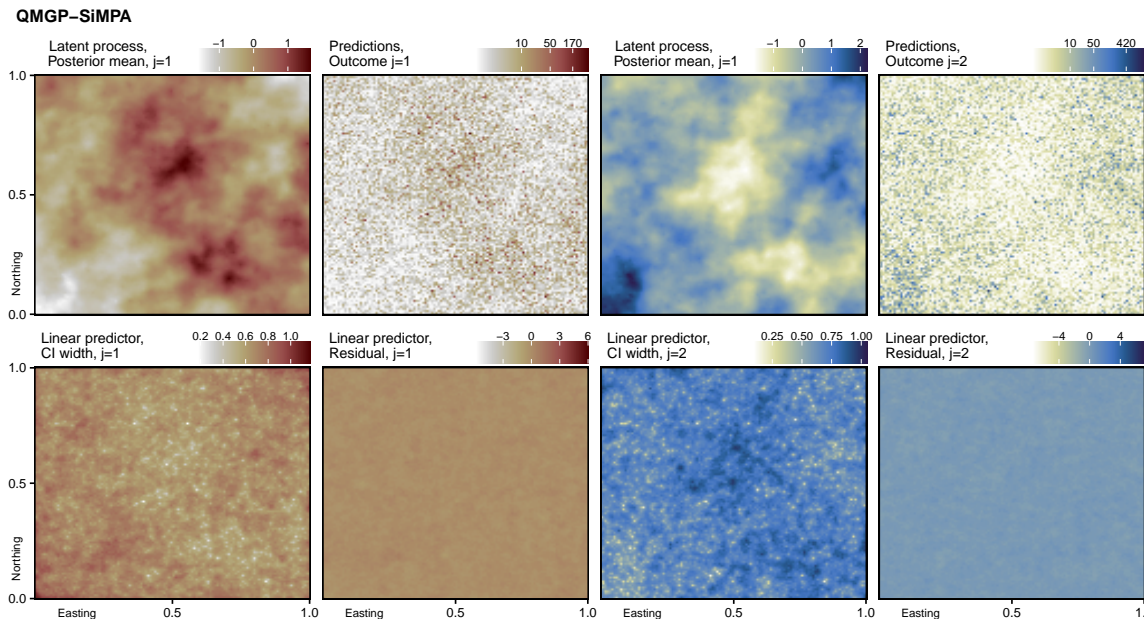


Figure 6: Output from fitting a coregionalized QMGP via SiMPA to simulated data in the latent NNGP scenario. Top row: estimated posterior mean of the spatial latent process and predictions for both outcomes. Bottom row: width of posterior credible intervals about log-intensity, and residual log-intensity.

6, where we also highlight that the lack of visible spatial patterns in the linear predictor residuals is evidence of the ability of SiMPA to capture the spatial correlation in the data.

A summary of results from all implemented methods is available in Table 1, which reports root mean square prediction error (RMSPE) and mean absolute error in prediction (MAEP) when predicting the log-intensity $\eta_{j,\text{test}}$ and the outcomes $y_{j,\text{test}}$, $j = 1, 2$ on the test set of 5740 locations, and the empirical coverage of 95% credible intervals (CI) about the log-intensity, in both scenarios. We observe that SiMPA offers excellent predictive performance and well calibrated credible intervals at a fraction of the compute cost, relative to state-of-the-art posterior sampling methods in this context. In the NNGP scenario, there is a disconnect between the fitted DAG (arising from a QMGP) and the data-generating DAG. This disconnect may explain why QMGP methods implementing the flexible AG10 cross-covariance function perform relatively better than in the GP scenario. Even in the NNGP setting, SiMPA retains excellent performance at a small compute cost.

Because the only difference between SiMPA and YA-MALA is in how the preconditioner is adapted, the relatively poor performance of YA-MALA can be attributed to it requiring a much longer burn-in period in practice. We attribute the poor performance of the implemented NNGP methods to the fact that they are unable to capture cross-variable dependence, as well as their being limited to Gaussian outcomes in R package `spNNGP`

Figure 7 expands on the analysis of empirical coverage of CIs by reporting the performance of all models at additional quantiles, relative to the oracle coverage, i.e., the empirical coverage of the model in which all unknowns are set to their true value. A value of relative coverage near 1 implies that the empirical coverage of the $Q\%$ CI is close to the coverage of

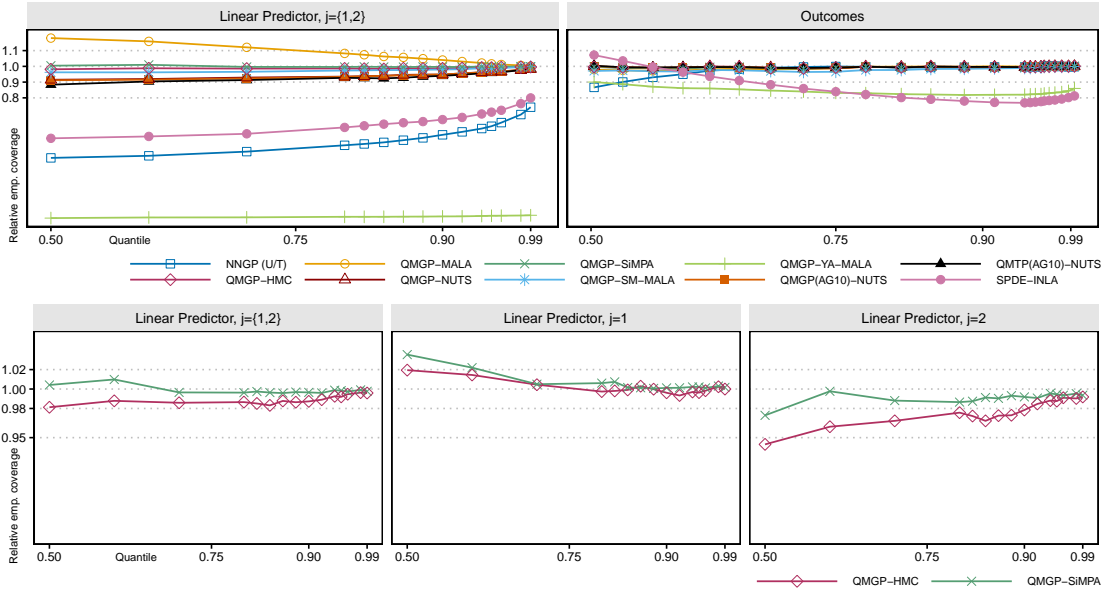
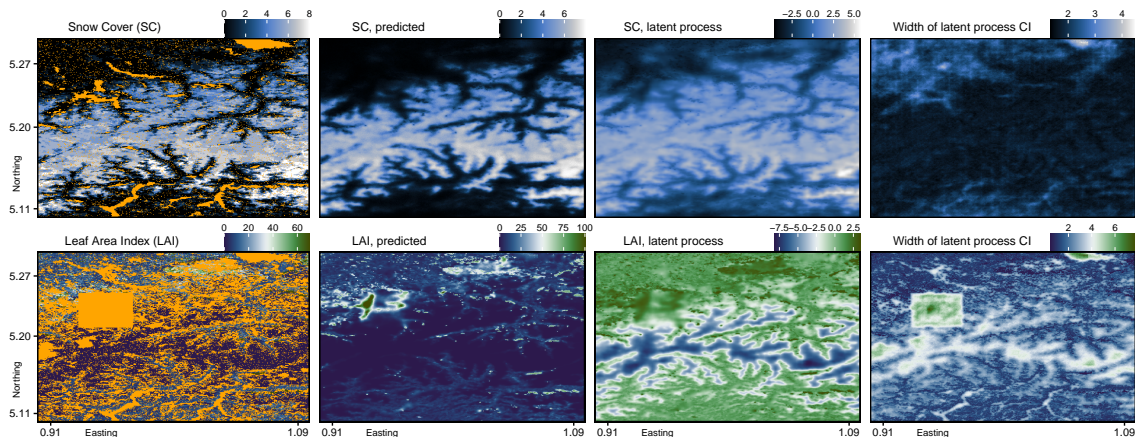


Figure 7: Top row: empirical coverage of uncertainty intervals at different quantiles, relative to the oracle model (values under 1 imply undercoverage of the interval), in the NNGP scenario. Bottom row: detailed comparison of relative coverage of SiMPA and HMC for the linear predictor of each outcome.

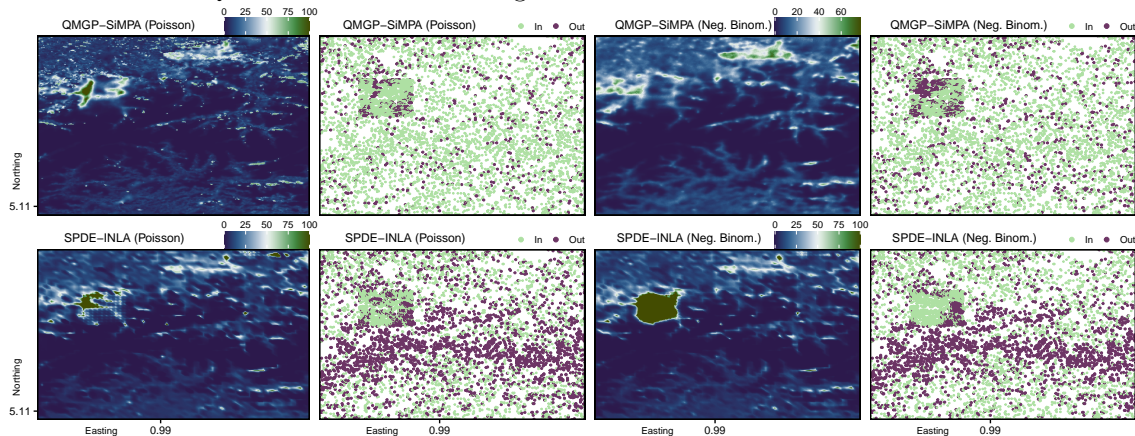
the true data generating model. From Figure 7 we observe that SiMPA outperforms other methods at this task.

Spatial model	Covariance model	Compute algorithm	j	Unrestricted GP					Nearest-neighbor GP, NN = 10						
				$y_{j,\text{test}}(\ell)$		$\eta_{j,\text{test}}(\ell)$		Time(s)	$y_{j,\text{test}}(\ell)$		$\eta_{j,\text{test}}(\ell)$		Cov _{95%}	Time(s)	
				RMSPE	MAEP	RMSPE	MAEP		Cov _{95%}	RMSPE	MAEP	RMSPE			MAEP
SPDE		INLA	1	3.01	1.39	0.32	0.25	69.48							
			2	6.14	1.90	0.33	0.26	63.44	333	2.87	1.36	0.32	0.26	69.27	334
QMGP	LMC	MALA	1	2.33	1.27	0.21	0.17	99.51	90	2.26	1.22	0.21	0.17	99.44	89
			2	4.08	1.57	0.20	0.16	94.27		3.77	1.53	0.19	0.15	93.58	
		YA-MALA	1	7.27	2.56	0.95	0.76	5.42	111	6.96	2.48	0.93	0.75	5.56	108
			2	18.98	4.92	1.28	1.01	3.92		18.85	4.89	1.30	1.03	4.20	
		SiMPA	1	2.18	1.22	0.17	0.14	95.83	117	2.16	1.19	0.17	0.14	95.21	116
			2	4.03	1.57	0.19	0.15	94.55		3.60	1.51	0.19	0.15	94.48	
	RM-MALA	1	2.19	1.22	0.18	0.15	96.28	187	2.17	1.20	0.18	0.15	94.93	183	
		2	4.20	1.56	0.23	0.18	93.37		3.97	1.56	0.23	0.18	92.57		
	HMC	1	2.18	1.23	0.17	0.14	95.73	246	2.16	1.20	0.17	0.14	94.65	359	
		2	3.96	1.56	0.19	0.15	94.76		3.54	1.51	0.19	0.15	93.82		
	NUTS	AG10	1	2.16	1.22	0.18	0.14	93.89	620	2.21	1.20	0.18	0.14	92.57	633
			2	4.07	1.57	0.20	0.16	90.66		3.56	1.51	0.19	0.15	90.97	
QMTTP		NUTS	1	2.22	1.23	0.18	0.14	92.64	501	2.13	1.19	0.17	0.14	91.39	480
			2	4.01	1.56	0.20	0.16	91.60		3.56	1.50	0.19	0.15	92.01	
NNGP	Exp	Transform & Response	1	2.19	1.23	0.18	0.14	91.77	857	2.16	1.20	0.18	0.14	90.97	841
			2	4.02	1.57	0.20	0.16	90.21		3.50	1.50	0.19	0.15	91.28	
			1	5.71	2.01	1.19	0.98	59.41	166	5.52	1.96	1.19	0.99	59.10	171
			2	15.55	3.63	1.36	1.12	58.09		15.44	3.56	1.34	1.10	58.68	

Table 1: Summary of out-of-sample results for all implemented models. Bolded values correspond to best performance.



(a) From the left: in-sample data, predictions of outcomes and latent processes, uncertainty quantification from a QMGP-SiMPA model using a Poisson distribution for LAI.



(b) Each raster image reports whole-domain predictions of all tested models, whereas each dark dot in the point clouds represents a domain location at which the 75% one-sided credible interval does not include the observed data point.

Figure 8: Performance of QMGP-SiMPA and SPDE-INLA in the MODIS data application.

5.2 MODIS data: leaf area and snow cover

The dynamics of vegetation greenness are important drivers of ecosystem processes; in alpine regions, they are influenced by seasonal snow cover. Predictive models for vegetation greenup and senescence in these settings are crucial for understanding how local biological communities respond to global change (Walker et al., 1993; Jönsson et al., 2010; Wang et al., 2015a; Xie et al., 2020). We consider remotely sensed leaf area and snow cover data from the MODerate resolution Imaging Spectroradiometer (MODIS) on the Terra satellite operated by NASA (v.6.1) at 122,500 total locations (a 350×350 grid where each cell covers a 0.25km^2 area) over a region spanning northern Italy, Switzerland, and Austria, during the 8-day period starting on December 3rd, 2019 (Figure 1). Leaf area index (LAI; number of equivalent layers of leaves relative to a unit of ground area, available as level 4 product MOD15A2H) is our primary interest and is stored as a positive integer value but is missing or unavailable at

38.2% of all spatial locations due to cloud cover or poor measurement quality. Snow cover (SC; number of days within an 8-day period during which a location is covered by snow, obtained from level 3 product MOD10A2) is measured with error and missing at 7.3% of the domain locations.

We create a test set by introducing missingness in LAI at 10,000 spatial locations, of which 5030 are chosen uniformly at random across the whole domain and 4970 belong to a contiguous rectangular region as displayed on the bottom left subplot of Figure 8a. We attempt to explain LAI based on SC by fitting (9) on the bivariate outcome $\mathbf{y}(\boldsymbol{\ell}) = (y_{\text{SC}}(\boldsymbol{\ell}), y_{\text{LAI}}(\boldsymbol{\ell}))^\top$ where we assume a Binomial distribution with 8 trials and logit link for SC, i.e. $E(y_{\text{SC}}(\boldsymbol{\ell}) \mid \mu(\boldsymbol{\ell})) = 8\mu(\boldsymbol{\ell}) = 8(1 + \exp\{-\eta_{\text{SC}}(\boldsymbol{\ell})\})^{-1}$, and a Poisson or Negative Binomial distribution for LAI. In both cases, $E(y_{\text{LAI}}(\boldsymbol{\ell}) \mid \eta_{\text{LAI}}(\boldsymbol{\ell})) = \mu_{\text{LAI}}(\boldsymbol{\ell}) = \exp\{\eta_{\text{LAI}}(\boldsymbol{\ell})\}$; for the Poisson model, $\text{Var}(y_{\text{LAI}}(\boldsymbol{\ell}) \mid \eta_{\text{LAI}}(\boldsymbol{\ell})) = \mu_{\text{LAI}}(\boldsymbol{\ell})$, whereas for the Negative Binomial model $\text{Var}(y_{\text{LAI}}(\boldsymbol{\ell}) \mid \eta_{\text{LAI}}(\boldsymbol{\ell})) = \mu_{\text{LAI}}(\boldsymbol{\ell}) + \tau\mu_{\text{LAI}}^2(\boldsymbol{\ell})$ where τ is an unknown scale parameter. We fit model (9) using latent coregionalized QMGPs with $k = 2$ on a 50×50 axis-parallel domain partition and run SiMPA for 30,000 iterations, of which 10,000 are discarded as burn-in and thinning the remaining ones with a 20:1 ratio, leading to a posterior sample of size 1,000. We compare our approaches in terms of prediction and uncertainty quantification about y_{LAI} on the test set to a SPDE-INLA approach implemented on a 60×60 mesh which led to similar compute times. As shown in Table 2, QMGP-SiMPA is competitive with or outperforms the SPDE-INLA method across all measured quantities. Figure 8b reports predictive maps of the tested models (prediction values are censored at 100 for visualization clarity), along with a visualization of 75% one-sided credible intervals which shows the SPDE-INLA method exhibiting undesirable spatial patterns, unlike QMGP-SiMPA.

Method	F_{LAI}	RMSPE	MedAE	CRPS		CI_{75}	CI_{95}	CI_{99}	Time (minutes)
				(mean)	(median)				
QMGP-SiMPA	Poisson	16.543	1.322	3.916	1.199	0.867	0.974	0.989	25.4
	Neg. Binom.	11.726	2.155	4.462	2.241	0.809	0.980	0.994	32
SPDE-INLA	Poisson	27.839	2.154	4.695	1.214	0.835	0.938	0.961	25.8
	Neg. Binom.	27019.470	2.444	54.986	1.720	0.875	0.975	0.987	86.5

Table 2: Root mean square error (RMSPE), median absolute error (MedAE), continuous ranked probability score (CRPS), and empirical coverage of one-sided intervals (CI_q), over the out-of-sample test set of 6,998 locations.

6. Applications: spatial community ecology

Ecologists seek to jointly model the spatial occurrence of multiple species, while inferring the impact of phylogeny and environmental covariates (see, e.g., Dorazio and Royle 2005; Doser et al. 2022). In order to realistically model such a scenario, we consider cases in which a relatively large number of georeferenced outcomes is observed, with the goal of predicting their realization at unobserved locations and estimating their latent correlation structure after accounting for spatial and/or temporal variability. Presence/absence information for different species is encoded as a multivariate binary outcome. Our model for multivariate binary outcomes lets $F_j(y_j(\boldsymbol{\ell}); \eta_j(\boldsymbol{\ell})) = \text{Bern}(\mu_j(\boldsymbol{\ell}))$ where $\mu_j(\boldsymbol{\ell}) = (1 + \exp\{-\eta_j(\boldsymbol{\ell})\})^{-1}$ and $v_h(\cdot) \sim \text{QMGP}(\mathbf{0}, \boldsymbol{\rho}_h(\cdot, \cdot))$, $h = 1, \dots, k$ in model (9), leading to coregionalized k -factor

QMGP methods which we fit via several Langevin methods, all of which use domain partitioning with blocks of size ≈ 36 and independent standard Gaussian priors on the lower-triangular elements of the factor loadings $\mathbf{\Lambda}$, unless otherwise noted.

We compare QMGP methods fit via our proposed Langevin algorithms to joint species distribution models (JSDM) implemented in R package `Hmsc` (Tikhonov et al., 2020), a popular software package for community ecology. `Hmsc` uses a probit link for binary outcomes, i.e. $\mu_j(\boldsymbol{\ell}) = \Phi(\eta_j(\boldsymbol{\ell}))$ where $\Phi(\cdot)$ is the Gaussian distribution function; then, non-spatial JS-DMs are implemented by letting $v_h(\boldsymbol{\ell}) \sim N(0, 1)$ independently for all $\boldsymbol{\ell}$ and $h = 1, \dots, K$, whereas NNGP-based JS-DMs assume $v_h(\cdot) \sim NNGP(\mathbf{0}, \boldsymbol{\rho}_h(\cdot, \cdot))$, $h = 1, \dots, k$. We set the number of neighbors as $m = 20$ in the NNGP specification. `Hmsc` assumes a cumulative shrinkage prior on the factor loadings (Bhattacharya and Dunson, 2011), which we set up with minimal shrinking ($a_1 = 2, a_2 = 2$) unless otherwise noted.

Section E.3 in the Supplementary Material compares our methods with alternative posterior sampling algorithm in fitting a multi-species N-mixture model for multi-species abundance data in community ecology.

6.1 Synthetic occupancy data

We generate 30 datasets by sampling $q = 10$ binary outcomes at $n = 900$ locations scattered uniformly in the domain $[0, 1]^2$: after sampling $k = 3$ independent univariate GPs $v_j(\cdot) \sim GP(\mathbf{0}, C_{\varphi_j})$ where $C_{\varphi_j}(\boldsymbol{\ell}, \boldsymbol{\ell}') = \exp\{-\varphi_j \|\boldsymbol{\ell} - \boldsymbol{\ell}'\|\}$ is the exponential covariance function with decay parameter φ_j , we construct a q -variate GP via coregionalization by letting $\mathbf{w}(\boldsymbol{\ell}) = \mathbf{\Lambda}\mathbf{v}(\boldsymbol{\ell})$ where $\mathbf{\Lambda}$ is a $q \times k$ lower-triangular matrix. We then sample the binary outcomes using a probit link, i.e. $y_j(\boldsymbol{\ell}) \sim \text{Bern}(\mu_j(\boldsymbol{\ell}))$ where $\mu_j(\boldsymbol{\ell}) = \Phi(\mathbf{x}(\boldsymbol{\ell})^\top \boldsymbol{\beta}_j + w_j(\boldsymbol{\ell}))$ for each $j = 1, \dots, q$ and where $\mathbf{x}(\boldsymbol{\ell})$ is a column vector of $p = 2$ covariates. For each of the 30 datasets, we randomly set $\varphi_j \sim U(1/2, 10)$, $j = 1, \dots, k$, $\mathbf{\Lambda}_{jj} \sim U(3/2, 2)$ for $j = 1, \dots, k$, $\mathbf{\Lambda}_{ij} \sim U(-2, 2)$ for $i < j$, and $\boldsymbol{\beta}_j \sim N(\mathbf{0}, I_2/5)$. These choices lead to a wide range of latent pairwise correlations induced on the outcomes via $\mathbf{w}(\cdot)$: letting $\boldsymbol{\Omega} = (\omega_{ij})_{i,j=1,\dots,q} = \mathbf{\Lambda}\mathbf{\Lambda}^\top$ represent the cross-covariance at zero spatial distance, we find the cross-correlations as $\boldsymbol{\Omega}_{\text{corr}} = \text{diag}(\omega_{jj}^{-1/2})\boldsymbol{\Omega}\text{diag}(\omega_{jj}^{-1/2})$. We realistically model long-range spatial dependence by choosing small values for φ_j , $j = 1, \dots, k$. Lastly, we create a test set using 20% of the locations for each outcome (missing data locations differ for each outcome).

We use the setup of QMGPs and `Hmsc` outlined above, noting that the link function used to generate the data is correctly specified for `Hmsc` but not for our models based on QMGP due to our current software implementation in R package `meshed`. MCMC for all methods was run for 10,000 iterations, of which the first 5,000 is discarded as burn-in. We compare all models based on the out-of-sample classification performance on each of the 10 outcomes as measured via the area under the receiver operating characteristic curve (AUC). Since a primary interest in these settings is to estimate latent correlations across outcomes, we compare models based on $\|\hat{\boldsymbol{\Omega}}_{\text{corr}} - \boldsymbol{\Omega}_{\text{corr}}\|_F$, i.e. the Frobenius distance between the Monte Carlo estimate of cross-correlation and its true value. Therefore, smaller values of $\|\hat{\boldsymbol{\Omega}}_{\text{corr}} - \boldsymbol{\Omega}_{\text{corr}}\|_F$ are desirable. Figure 10 shows box-plots summarising the results, whereas Table 3 reports averages along with compute times. In these settings, the non-spatial model unsurprisingly performed worst. Langevin methods for the spatial models proposed in this article – and in particular SiMPA – lead to improved classification performance, smaller

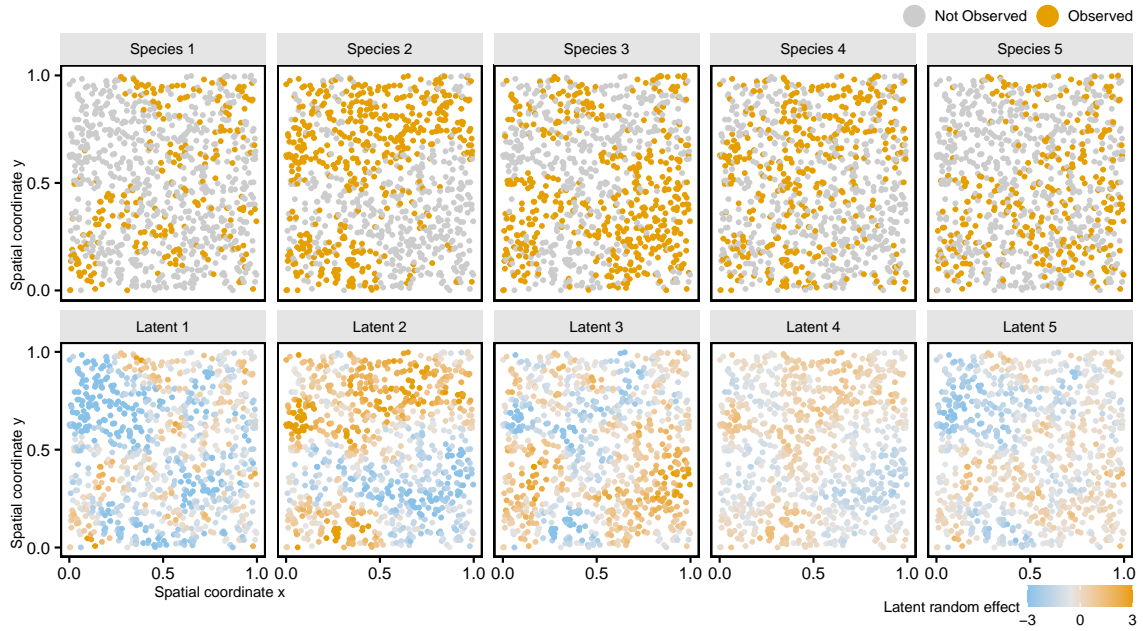


Figure 9: Synthetic dichotomous occurrence data (top row), and the spatial latent effects used to generate them (bottom row). Here, we show 5 (of 10) outcomes in 1 (of 30) simulated datasets.

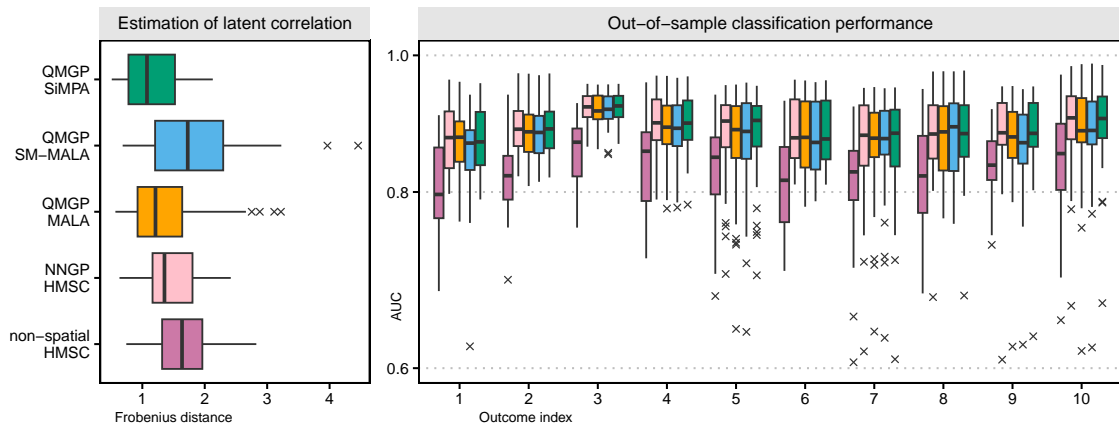


Figure 10: Box-plot summaries of estimation and classification performance over 30 datasets. Left: $\|\hat{\Omega}_{\text{corr}} - \Omega_{\text{corr}}\|_F$ for the competing methods. Right: AUC for each outcome.

errors in estimating latent correlations, and a 30-fold reduction in compute time, relative to the coregionalized NNGP method implemented via MCMC in `Hmsc`.

Method	<code>Hmsc</code>		MALA	SM-MALA	SiMPA
Prior on rand. eff.	non-spatial	NNGP	QMGP		
Avg. AUC	0.827	0.885	0.882	0.874	0.885
Min. AUC	0.573	0.608	0.392	0.530	0.609
Max. AUC	0.969	0.983	0.986	0.987	0.984
$\ \hat{\Omega}_{\text{corr}} - \Omega_{\text{corr}}\ _F$	1.66	1.43	1.46	1.91	1.14
Avg. time (minutes)	5.15	17.4	0.44	0.73	0.53

Table 3: Performance in classification, estimation, and compute time, over 30 synthetic datasets.

6.2 North American breeding bird survey data

The North American Breeding Bird Survey dataset contains avian point count data for more than 700 North American bird taxa (species, races, and unidentified species groupings). These data are collected annually during the breeding season, primarily June, along thousands of randomly established roadside survey routes in the United States and Canada.

We consider a dataset of $n = 4118$ locations spanning the continental U.S., and $q = 27$ bird species. The specific species we consider belong to the *passeriforme* order and are observed at a number of locations which is between 40% and 60% of the total number of available locations – Figure 2 shows a subset of the data. We dichotomize the observed counts to obtain presence/absence data. The effective data size is $nq = 111,186$. We implement Langevin methods using coregionalized QMGPs with $k = 2, 4, 6, 8, 10$ spatial factors using exponential correlations with decay $\phi \sim U[0.1, 10]$ *a priori*. We also test the sensitivity to the domain partitioning scheme by testing 8×4 (coarse), 16×8 (medium), and 32×16 (fine) axis-parallel domain partitioning schemes. Finer partitioning implies more restrictive spatial conditional independence assumptions. In implementing the shrinkage prior of Bhattacharya and Dunson (2011), `Hmsc` dynamically chooses the number of factors up to a maximum k_{max} : in the non-spatial `Hmsc` model, letting $k_{\text{max}} = 10$ ultimately leads to 6 or fewer factors being used during MCMC. In the spatial `Hmsc` models using NNGPs, we set $k_{\text{max}} = 2$ or $k_{\text{max}} = 5$ to restrict run times. Figure 11 reports average classification performance and run times. QMGP-MALA scales only linearly with the number of factors, but its performance is strongly negatively impacted by partition size. QMGP-SM-MALA exhibits large improvements in classification performance, however these improvements come at a large run time cost. QMGP-SiMPA outperforms all other models while providing large time savings relative to SM-MALA and being less sensitive to the choice of partition. A QMGP-SiMPA model on the 32×16 partition with $k = 4$ outperforms a spatial NNGP-`Hmsc` model in classifying the 27 bird species with a reduction in run time of over three orders of magnitude (respectively 4.1 minutes and 70.7 hours). We provide a summary of the efficiency in sampling the elements of Ω_{corr} in Table 4, where we make comparisons of ESS/s relative to the non-spatial `Hmsc` model. While efficient estimation of Ω_{corr} remains challenging, QMGP-SiMPA models show marked improvements relative to a state-of-the-art alternative.

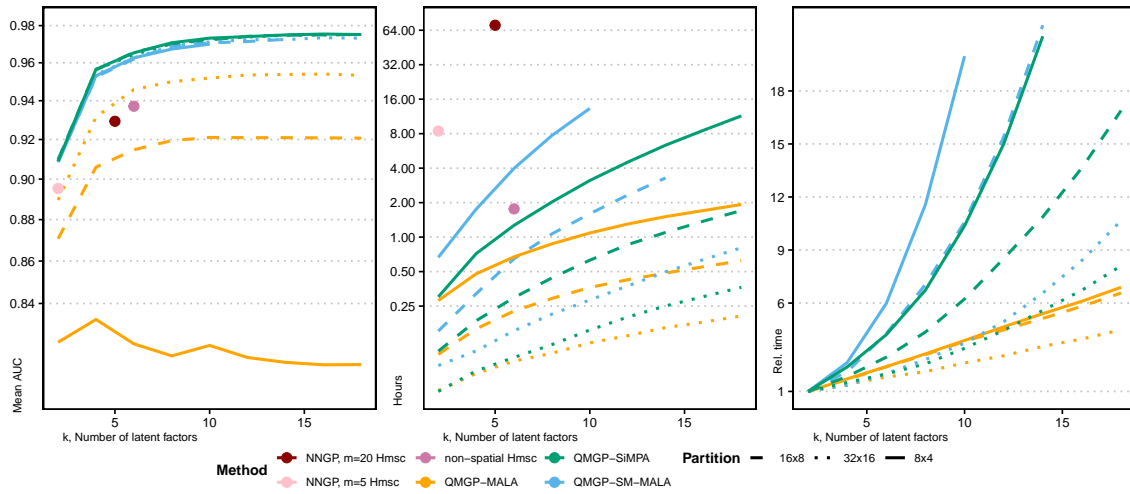


Figure 11: Left: mean AUC across the 27 bird species for different choices of k . Center: run times in hours. Right: run time of QMGP models as a proportion to the run time choosing $k = 2$.

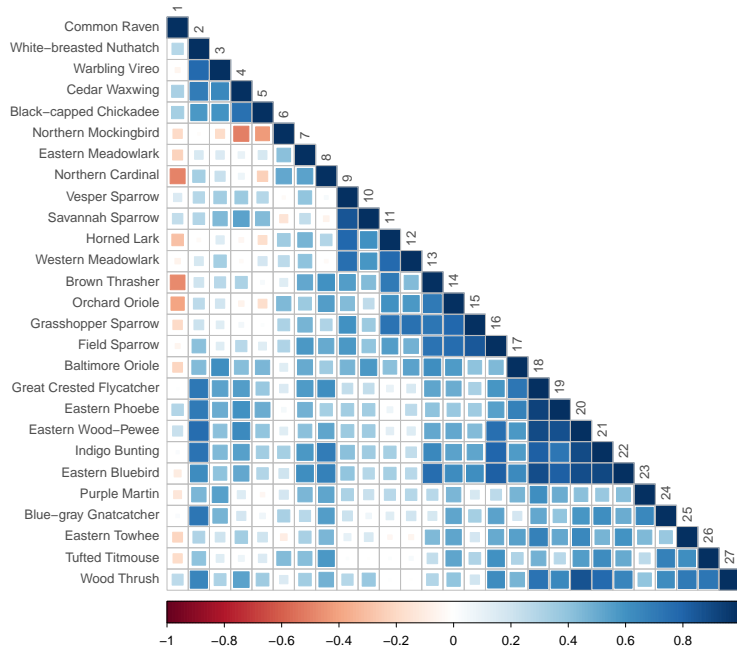


Figure 12: Lower triangular portion of Ω_{corr} , the estimated correlation among the 27 bird species.

Method	Hmsc		SiMPA			
Prior	non-spatial	NNGP	QMGP			
k	≤ 10	5	4	4	10	10
Setting		$m = 20$	32×16	8×4	32×16	8×4
Avg. AUC	0.9349	0.9293	0.9565	0.9565	0.9728	0.9732
Time (minutes)	87.45	4245.02	4.08	43.10	9.27	187.24
ESS/s for elements of Ω_{corr} (relative to Hmsc non-spatial)						
min	1	10^{-4}	0.57	0.02	0.05	0.01
median	1	0.012	2.12	0.33	0.86	0.05
mean	1	0.015	3.69	0.45	1.20	0.07
max	1	0.102	42.47	3.95	11.15	0.56

Table 4: Out-of-sample performance in classification of the 27 bird species, compute time, and efficiency in estimation of Ω_{corr} , relative to a non-spatial JDSM model.

7. Discussion

We have introduced Bayesian hierarchical models based on DAG constructions of latent spatial processes for large scale non-Gaussian multivariate multi-type data which may be misaligned, along with computational tools for adaptive posterior sampling. We illustrated our methods using applications with data sizes in the tens to hundreds of thousands, with compute times ranging from a few seconds to under 30 minutes in a single workstation. The compute time for a single SiMPA iteration for a univariate Poisson outcome observed on gridded coordinates with $n = 10^6$ is under 0.2 seconds after burn-in; in other words, our methods enable running MCMC for hundreds of thousands of iterations on massive spatial data under a total time budget of 12 hours.

We have applied our methodologies using practical cross-covariance choices such as models of coregionalization built on independent stationary covariances. However, nonstationary models are desirable in many applied settings. Recent work (Jin et al., 2021) highlights that DAG choice must be made carefully when considering explicit models of nonstationary, as spatial process models based on sparse DAGs induce nonstationarities even when using stationary covariances. Our work in this article will enable new research into nonstationary models of large scale non-Gaussian data. Furthermore, our methods can be applied for posterior sampling of Bayesian hierarchies based on more complex conditional independence models of multivariate dependence (Dey et al., 2021).

In our work, we have assumed a common DAG and partitioning for all spatial variables. In some settings, these assumptions may lead to inflexibility in modeling variables with fundamentally different dependence structures and/or spatial domain constraints (see, e.g., Jin et al. 2022). In the multivariate setting, one potentially useful direction towards building a highly flexible class of models is to infer different DAGs for different factors within a spatial factor model by extending the methods of Jin et al. (2021). Understanding how to generally build flexible and scalable spatial factor models using different DAGs and accounting for unequal domain constraints is an interesting direction for future research.

Our methodologies rely on the ability to embed the assumed spatial DAG within the larger Bayesian hierarchy and lead to drastic reductions in wall clock time compared to

models based on unrestricted GPs. Nevertheless, high posterior correlations of high dimensional model parameters may still negatively impact overall sampling efficiency in certain cases. Motivated by recent progress in improving sampling efficiency of multivariate Gaussian models (Peruzzi et al., 2021), future research will consider generalized strategies for improving MCMC performance in spatial factor models of highly multivariate non-Gaussian data. Finally, optimizing DAG choice for MCMC performance is another interesting path, and recent work on the theory of Bayesian computation for hierarchical models (Zanella and Roberts, 2021) might motivate further development for spatial process models based on DAGs.

Acknowledgements

The authors have received funding from the European Research Council (ERC) under the European Union’s Horizon 2020 research and innovation programme (grant agreement No 856506), and grants R01ES028804 and R01ES035625 of the United States National Institutes of Health (NIH).

SUPPLEMENTARY MATERIAL

Appendix A. Spatial meshing with projections

The customary setup of a DAG-based model based on spatial meshing is to let $\mathcal{S} \cap \mathcal{T} \approx \mathcal{T}$ as the resulting large overlap between knots and observed locations avoids sampling at non-reference locations. However, it is often desirable to allow flexible choices of \mathcal{S} ; for example, there are some computational advantages when \mathcal{S} is a grid and \mathcal{T} are irregularly spaced, or when the data are observed with particular patterns (Peruzzi et al., 2021). In order to let \mathcal{S} be more flexibly determined while also avoiding sampling $\mathbf{w}(\boldsymbol{\ell})$ at non-reference locations, we introduce a linear projection operator $\mathbf{H}(\boldsymbol{\ell})$ of dimension $q \times qn_{[\boldsymbol{\ell}]}$ and where $n_{[\boldsymbol{\ell}]}$ is the number of locations in $[\boldsymbol{\ell}] \subset \mathcal{S}$; after denoting $\tilde{\mathbf{w}}(\boldsymbol{\ell}) = \mathbf{H}(\boldsymbol{\ell})\mathbf{w}_{[\boldsymbol{\ell}]}$, we assume that if $\boldsymbol{\ell} \in \mathcal{S}$ then $\mathbf{H}(\boldsymbol{\ell})$ is such that $\tilde{\mathbf{w}}(\boldsymbol{\ell}) = \mathbf{w}(\boldsymbol{\ell})$. Then, we build the outcome model as

$$\begin{aligned} y_j(\boldsymbol{\ell}) \mid \eta_j(\boldsymbol{\ell}), \gamma_j &\sim F_j(\eta_j(\boldsymbol{\ell}), \gamma_j), & \eta_j(\boldsymbol{\ell}) &= \mathbf{x}_j(\boldsymbol{\ell})^\top \boldsymbol{\beta}_j + \tilde{w}_j(\boldsymbol{\ell}), \\ \mathbf{w}(\cdot) &\sim \Pi_{\mathcal{G}} \end{aligned} \tag{12}$$

where we have replaced $\mathbf{w}(\boldsymbol{\ell})$ with $\tilde{\mathbf{w}}(\boldsymbol{\ell})$. Setting $\mathbf{H}(\boldsymbol{\ell})$ such that $\tilde{\mathbf{w}}(\boldsymbol{\ell}) = E[\mathbf{w}(\boldsymbol{\ell}) \mid \mathbf{w}_{[\boldsymbol{\ell}]}]$ leads to an interpretation of (12) as a “local” predictive process (Banerjee et al., 2008). The posterior distribution for this model is:

$$\pi(\{\boldsymbol{\beta}_j, \gamma_j\}_{j=1}^q, \mathbf{w}_{\mathcal{S}}, \boldsymbol{\theta} \mid \mathbf{y}_{\mathcal{T}}) \propto \pi(\boldsymbol{\theta})\pi_{\mathcal{G}}(\mathbf{w}_{\mathcal{S}} \mid \boldsymbol{\theta}) \prod_{j=1}^q \pi(\boldsymbol{\beta}_j, \gamma_j) \prod_{\boldsymbol{\ell} \in \mathcal{T}_j} dF_j(y_j(\boldsymbol{\ell}) \mid \tilde{w}_j(\boldsymbol{\ell}), \boldsymbol{\beta}_j(\boldsymbol{\ell}), \gamma_j). \tag{13}$$

In this scenario, omitting the residual term $\mathbf{e}(\boldsymbol{\ell}) = \mathbf{w}(\boldsymbol{\ell}) - \tilde{\mathbf{w}}(\boldsymbol{\ell})$ from (12) leads to advantages in sampling, but possible oversmoothing of the latent spatial surface due to the fact that $\text{var}[\tilde{\mathbf{w}}(\boldsymbol{\ell})] < \text{var}[\mathbf{w}(\boldsymbol{\ell})]$. In the conditionally conjugate Gaussian setting, such biases can be partly corrected (Banerjee et al., 2010; Peruzzi et al., 2021). Certain ad-hoc solutions may be available by allowing spatial variation of γ_j , i.e. replacing it with $\gamma_j(\boldsymbol{\ell})$. However, we may choose to ignore the residual term because (1) it is common to assume smoother surfaces with non-Gaussian data, (2) we can choose \mathcal{S} to be very large, reducing possible biases, (3) we can revert to model (1) by setting $\mathcal{S} = \mathcal{T}$. Posterior sampling for (12) proceeds via Algorithm 4.

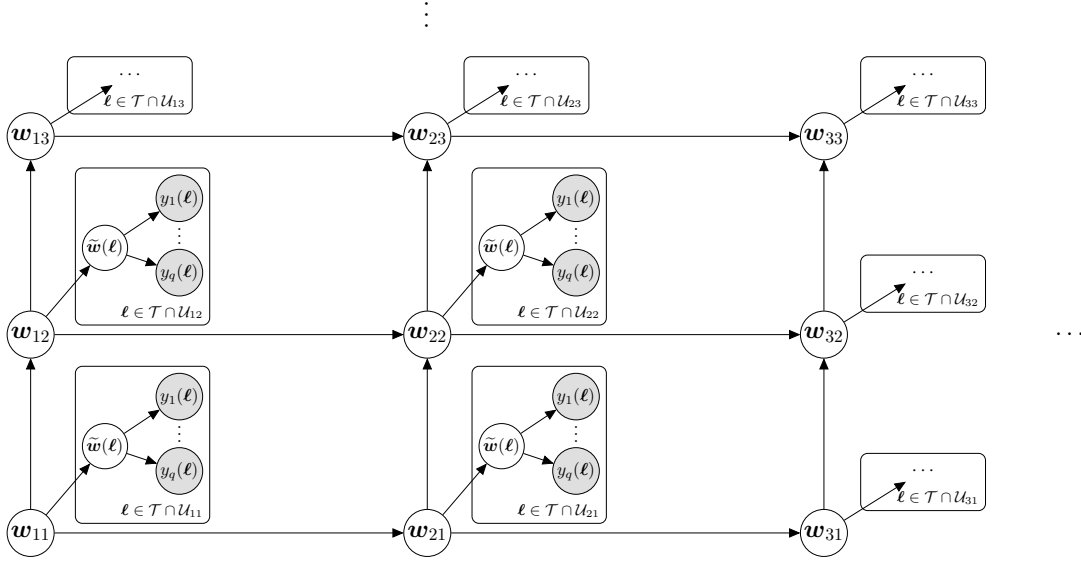


Figure 13: Directed acyclic graph representing a special case of model (12), for locations at which at least one outcome is observed. For simplicity, we consider $\mathcal{S} \cap \mathcal{T} = \emptyset$ and omit the directed edges from (β_j, γ_j) to each $y_j(\ell)$. If $y_j(\ell)$ is unobserved and therefore $\ell \notin \mathcal{T}_j$, the corresponding node is missing.

Algorithm 4 Posterior sampling of model (12).

Initialize $\beta_j^{(0)}$ and $\gamma_j^{(0)}$ for $j = 1, \dots, q$, $\mathbf{w}_S^{(0)}$, and $\boldsymbol{\theta}^{(0)}$

for $t \in \{1, \dots, T^*, T^* + 1, \dots, T^* + T\}$ **do** ▷ MCMC loop

- 1: for $j = 1, \dots, q$, sample $\beta_j^{(t)} \mid \mathbf{y}_\mathcal{T}, \tilde{\mathbf{w}}_\mathcal{T}^{(t-1)}, \gamma_j^{(t-1)}$
- 2: for $j = 1, \dots, q$, sample $\gamma_j^{(t)} \mid \mathbf{y}_\mathcal{T}, \tilde{\mathbf{w}}_\mathcal{T}^{(t-1)}, \beta_j^{(t)}$
- 3: sample $\boldsymbol{\theta}^{(t)} \mid \mathbf{y}_\mathcal{T}, \mathbf{w}_S^{(t-1)}, \{\beta_j^{(t)}, \gamma_j^{(t)}\}_{j=1}^q$
- 4: for $i = 1, \dots, M$, sample $\mathbf{w}_i^{(t)} \mid \mathbf{w}_{\text{mb}(i)}^{(t)}, \mathbf{y}_i, \boldsymbol{\theta}^{(t)}, \{\beta_j^{(t)}, \gamma_j^{(t)}\}_{j=1}^q$ ▷ reference sampling

end for

Assuming convergence has been attained after T^* iterations:
 discard $\{\beta_j^{(t)}, \gamma_j^{(t)}\}_{j=1}^q, \mathbf{w}_S^{(t)}, \boldsymbol{\theta}^{(t)}$ for $t = 1, \dots, T^*$

Output: Correlated sample of size T with density

$$\{\beta_j^{(t)}, \gamma_j^{(t)}\}_{j=1}^q, \mathbf{w}_S^{(t)}, \boldsymbol{\theta}^{(t)} \sim \pi(\{\beta_j, \gamma_j\}_{j=1}^q, \mathbf{w}_S^{(t)}, \boldsymbol{\theta} \mid \mathbf{y}_\mathcal{T}).$$

Appendix B. Choice of DAG and partition

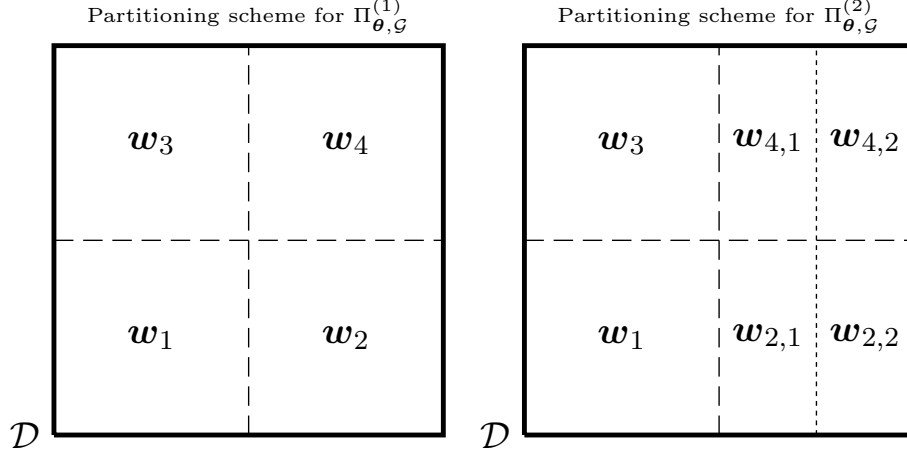


Figure 14: Illustration of the two partitioning schemes. On the right, we juxtapose the second partitioning scheme to clarify the changes relative to the scenario on the left.

Spatially meshed models on the same partition of \mathcal{S} can be compared in terms of the sparsity of \mathcal{G} . If edges are added to a sparse DAG \mathcal{G}_1 to obtain \mathcal{G}_2 , the child process $\Pi_{\mathcal{G}_2}$ is closer to the parent process Π_{θ} (in a Kullback-Leibler (KL) sense) relative to $\Pi_{\mathcal{G}_1}$ (Peruzzi et al., 2022). For treed DAGs and recursive domain partitioning, the KL divergence of $\Pi_{\mathcal{G}}$ from Π can be reduced by increasing the block size at the root nodes (Peruzzi and Dunson, 2022). Here, we analyse the modeling implications different non-nested domain partitions have, while using the same DAG structure to govern dependence between partition regions. This scenario occurs e.g. when constructing a cubic MGP model (QMGP).

We consider two partitions of the x-coordinate axis within a 2×2 axis-parallel partitioning scheme (Figure 14) and construct $\Pi_{\mathcal{G}}^{(i)}$, $i = 1, 2$ based on each partitioning scheme. According to the first partitioning scheme, $\mathbf{w}_{\mathcal{S}}$ (in short, \mathbf{w}) is partitioned as $\mathbf{w} = \{\mathbf{w}_1, \mathbf{w}_2, \mathbf{w}_3, \mathbf{w}_4\}$ whereas with the alternative we have $\mathbf{w} = \{\mathbf{w}_1^*, \mathbf{w}_{2,2}, \mathbf{w}_3^*, \mathbf{w}_{4,2}\}$ where $\mathbf{w}_1^* = \{\mathbf{w}_1, \mathbf{w}_{2,1}\}$ and $\mathbf{w}_3^* = \{\mathbf{w}_3, \mathbf{w}_{4,1}\}$. When analysing the relative KL divergence of these two models from Π , we see

$$\begin{aligned} KL(\pi \| \pi_{\mathcal{G}}^{(2)}) - KL(\pi \| \pi_{\mathcal{G}}^{(1)}) &= \int \log \frac{\pi(\mathbf{w})}{\pi_{\mathcal{G}}^{(2)}(\mathbf{w})} \pi(\mathbf{w}) d\mathbf{w} - \int \log \frac{\pi(\mathbf{w})}{\pi_{\mathcal{G}}^{(1)}(\mathbf{w})} \pi(\mathbf{w}) d\mathbf{w} \\ &= \int \log \pi_{\mathcal{G}}^{(1)}(\mathbf{w}) \pi(\mathbf{w}) d\mathbf{w} - \int \log \pi_{\mathcal{G}}^{(2)}(\mathbf{w}) \pi(\mathbf{w}) d\mathbf{w} \\ &= \int \left(\log \pi_{\mathcal{G}}^{(1)}(\mathbf{w}) - \log \pi_{\mathcal{G}}^{(2)}(\mathbf{w}) \right) \pi(\mathbf{w}) d\mathbf{w} \end{aligned}$$

Since we fix the same \mathcal{G} across partitions, we have

$$\begin{aligned} \pi_{\mathcal{G}}^{(1)}(\mathbf{w}_{\mathcal{S}}) &= \pi(\mathbf{w}_1) \pi(\mathbf{w}_2 | \mathbf{w}_1) \pi(\mathbf{w}_3 | \mathbf{w}_1) \pi(\mathbf{w}_4 | \mathbf{w}_2, \mathbf{w}_3) \\ &= \pi(\mathbf{w}_1) \pi(\mathbf{w}_{2,1} | \mathbf{w}_1) p(\mathbf{w}_{2,2} | \mathbf{w}_1, \mathbf{w}_{2,1}) \pi(\mathbf{w}_3 | \mathbf{w}_1) \pi(\mathbf{w}_{4,1} | \mathbf{w}_{2,1}, \mathbf{w}_{2,2}, \mathbf{w}_3) \cdot \end{aligned}$$

$$\begin{aligned}
 & \cdot \pi(\mathbf{w}_{4,2} | \mathbf{w}_{2,1}, \mathbf{w}_{2,2}, \mathbf{w}_3, \mathbf{w}_{4,1}) \\
 \pi_{\mathcal{G}}^{(2)}(\mathbf{w}_S) &= \pi(\mathbf{w}_1^*) \pi(\mathbf{w}_{2,2} | \mathbf{w}_1^*) \pi(\mathbf{w}_3^* | \mathbf{w}_1^*) \pi(\mathbf{w}_{4,2} | \mathbf{w}_{2,2}, \mathbf{w}_3^*) \\
 &= \pi(\mathbf{w}_1) \pi(\mathbf{w}_{2,1} | \mathbf{w}_1) p(\mathbf{w}_{2,2} | \mathbf{w}_1, \mathbf{w}_{2,1}) \pi(\mathbf{w}_3 | \mathbf{w}_1, \mathbf{w}_{2,1}) \pi(\mathbf{w}_{4,1} | \mathbf{w}_1, \mathbf{w}_{2,1}, \mathbf{w}_3) \cdot \\
 & \quad \cdot \pi(\mathbf{w}_{4,2} | \mathbf{w}_{2,2}, \mathbf{w}_3, \mathbf{w}_{4,1}),
 \end{aligned}$$

and therefore the sign of $KL(\pi || \pi_{\mathcal{G}}^{(2)}) - KL(\pi || \pi_{\mathcal{G}}^{(1)})$ depends on

$$\log \frac{\pi_{\mathcal{G}}^{(1)}(\mathbf{w})}{\pi_{\mathcal{G}}^{(2)}(\mathbf{w})} = \log \left(\frac{\pi(\mathbf{w}_3 | \mathbf{w}_1)}{\pi(\mathbf{w}_3 | \mathbf{w}_{2,1}, \mathbf{w}_1)} \frac{\pi(\mathbf{w}_{4,1} | \mathbf{w}_{2,2}, \mathbf{w}_{2,1}, \mathbf{w}_3)}{\pi(\mathbf{w}_{4,1} | \mathbf{w}_1, \mathbf{w}_{2,1}, \mathbf{w}_3)} \frac{\pi(\mathbf{w}_{4,2} | \mathbf{w}_{2,1}, \mathbf{w}_{2,2}, \mathbf{w}_3, \mathbf{w}_{4,1})}{\pi(\mathbf{w}_{4,2} | \mathbf{w}_{2,2}, \mathbf{w}_3, \mathbf{w}_{4,1})} \right),$$

where we see that the performance of $\Pi_{\mathcal{G}}^{(1)}$ relative to $\Pi_{\mathcal{G}}^{(2)}$ in approximating Π is undetermined because there is no ordering between the number of edges in $\Pi_{\mathcal{G}}^{(1)}$ and $\Pi_{\mathcal{G}}^{(2)}$. Nevertheless, the above discussion remains useful in practice when the reference set \mathcal{S} is chosen at observed locations. For example, if data are unavailable at $(2, 1)$, then $\mathbf{w}_{2,1}$ has length zero, and one would then choose $\Pi_{\mathcal{G}}^{(1)}$ over $\Pi_{\mathcal{G}}^{(2)}$ if uncertainty about $\mathbf{w}_{4,1}$ is reduced by knowledge of $\mathbf{w}_{2,2}$ more than it is reduced by knowledge of \mathbf{w}_1 .

Appendix C. Gradient-based sampling

We outline proofs for propositions of Section 3.

Proposition 3. In the hierarchical model $\alpha \sim N_k(\alpha; \mathbf{m}_\alpha, \mathbf{V}_\alpha)$, $\mathbf{x} | \alpha, S \sim N_n(\mathbf{x}; A\alpha, S)$, consider the following proposal for updating $\alpha | \mathbf{x}, S$:

$$\alpha^* = \alpha + \frac{\varepsilon_1^2}{2} \mathbf{G}_\alpha \nabla_\alpha \log p(\alpha | \text{---}) + \varepsilon_2 \mathbf{G}_\alpha^{\frac{1}{2}} \mathbf{u},$$

where $\mathbf{u} \sim N_n(0, I_n)$, and we set $\varepsilon_1 = \sqrt{2}$, $\varepsilon_2 = 1$. Then, $q(\alpha^* | \alpha) = p(\alpha^* | \mathbf{x}, S)$, i.e. this modified SM-MALA proposal leads to always accepted Gibbs updates.

Proof We compute

$$\begin{aligned}
 \nabla_\alpha \log p(\alpha | \text{---}) &= \nabla_\alpha \log p(\mathbf{x} | \alpha, S) \pi(\alpha) = \nabla_\alpha \log \{N_n(\mathbf{x}; A\alpha, S) N_k(\alpha; \mathbf{m}_\alpha, \mathbf{V}_\alpha)\} \\
 &= -\frac{1}{2} \nabla_\alpha \{(\alpha - \mathbf{m}_\alpha)^\top \mathbf{V}_\alpha^{-1} (\alpha - \mathbf{m}_\alpha) + (\mathbf{x} - A\alpha)^\top S^{-1} (\mathbf{x} - A\alpha)\} \\
 &= A^\top S^{-1} \mathbf{x} + \mathbf{V}_\alpha^{-1} \mathbf{m}_\alpha - (A^\top S^{-1} A + \mathbf{V}_\alpha^{-1}) \alpha
 \end{aligned}$$

from which we immediately find $\mathbf{G}_\alpha = (A^\top S^{-1} A + \mathbf{V}_\alpha^{-1})^{-1}$. Then, the update is

$$\begin{aligned}
 \alpha^* &= \alpha + \frac{\varepsilon_1^2}{2} (A^\top S^{-1} A + \mathbf{V}_\alpha^{-1})^{-1} (A^\top S^{-1} \mathbf{x} + \mathbf{V}_\alpha^{-1} \mathbf{m}_\alpha - (A^\top S^{-1} A + \mathbf{V}_\alpha^{-1}) \alpha) + \tilde{\mathbf{u}} \\
 &= \frac{\varepsilon_1^2}{2} (A^\top S^{-1} A + \mathbf{V}_\alpha^{-1})^{-1} (A^\top S^{-1} \mathbf{x} + \mathbf{V}_\alpha^{-1} \mathbf{m}_\alpha) - \left(1 - \frac{\varepsilon_1^2}{2}\right) \alpha + \tilde{\mathbf{u}},
 \end{aligned}$$

where $\tilde{\mathbf{u}} \sim N(\mathbf{0}, \varepsilon_2^2 (A^\top S^{-1} A + \mathbf{V}_\alpha^{-1})^{-1})$. Setting $\varepsilon_1 = \sqrt{2}$ and $\varepsilon_2 = 1$ leads to the Gibbs update one obtains from a Gaussian likelihood and a Gaussian conjugate prior. In fact,

since $q(\boldsymbol{\alpha}^* | \boldsymbol{\alpha}) = p(\boldsymbol{\alpha}^* | \mathbf{x}, S)$ then the acceptance probability for $\boldsymbol{\alpha}^*$ is $\frac{p(\boldsymbol{\alpha}^* | \mathbf{x}, S)q(\boldsymbol{\alpha} | \boldsymbol{\alpha}^*)}{p(\boldsymbol{\alpha} | \mathbf{x}, S)q(\boldsymbol{\alpha}^* | \boldsymbol{\alpha})} = 1$.
 ■

Proposition 4. Suppose π is everywhere non-zero and twice differentiable so that \mathbf{g}_x and \mathbf{G}_x are well defined. Let $\varepsilon > 0$, $K \subset \mathbb{R}^d$, $D > 0$. Additionally assume that if $\mathbf{x}_{(m)} \in K$ with $\text{dist}(\mathbf{x}_{(m)}, K^c) = u$ with $0 \leq u \leq 1$ then the proposal is changed to $\mathbf{x}_{(\text{new})} \sim N(\mathbf{x}_{(m)} + \frac{\varepsilon^2}{2}\mathbf{M}_{(T^{\text{adapt}})}\tilde{\mathbf{g}}_{\mathbf{x}_{(m)}}, \varepsilon^2\mathbf{M}_{(T^{\text{adapt}})})$. Then, Algorithm 2 converges in distribution to P .

Proof We show that SiMPA satisfies the assumptions of Theorem 21 of Craiu et al. (2015). Algorithm 2 has by construction bounded jumps, no adaptation outside K after iteration T^{adapt} and the fixed kernel outside K is bounded above by $(2\pi)^{-d/2}|\mathbf{M}_{T^{\text{adapt}}}|^{1/2}$. Because we bound \mathbf{g}_x and \mathbf{G}_x with D by using $\tilde{\mathbf{g}}_x$ and $\tilde{\mathbf{G}}_x$, the adaptive proposal kernel inside K is $Q_\delta(\mathbf{x}', \mathbf{x})$ where $\delta \in \Delta$ and Δ is a compact index set. Because outside K we use a fixed proposal kernel with continuous densities with respect to Lebesgue measure and we assumed that the target density $p(\cdot)$ is also continuous, the ε - δ condition holds (eq (6) in Craiu et al. 2015). Continuity of the target density and the proposal kernels hold by assumption. These assumptions are sufficient for the algorithm to satisfy the containment condition. The additional requirement to achieve convergence is diminishing adaptation, which holds by construction given the decreasing sequence $\{\gamma_m\}$.
 ■

Appendix D. Coregionalization of MGPs

D.1 Equivalency result

Proposition 6 A q -variate MGP on a fixed DAG \mathcal{G} , a domain partition \mathbf{T} , and a LMC cross-covariance function \mathbf{C}_θ is equal in distribution to a LMC model built upon k independent univariate MGPs, each of which is defined on the same DAG \mathcal{G} and the same domain partition \mathbf{T} .

Proof For $i = 1, \dots, M$, we want to show that the conditional densities $\pi(\mathbf{w}_i | \mathbf{w}_{[i]}) = N(\mathbf{w}_i; \mathbf{H}_i \mathbf{w}_{[i]}, \mathbf{R}_i)$ a q variate MGP based on LMC cross-covariance $\mathbf{C}(\boldsymbol{\ell}, \boldsymbol{\ell}') = \boldsymbol{\Lambda} \boldsymbol{\rho}(\boldsymbol{\ell}, \boldsymbol{\ell}') \boldsymbol{\Lambda}^\top$ (we drop $\boldsymbol{\theta}$ and Φ subscripts on \mathbf{C} and $\boldsymbol{\rho}$, respectively, for simplicity) can be obtained equivalently via a LMC in which the k margins are univariate MGPs

$$\begin{aligned} \mathbf{C}_{i,[i]} &= (I_{n_i} \otimes \boldsymbol{\Lambda}) \boldsymbol{\rho}_{i,[i]} (I_{n_{[i]}} \otimes \boldsymbol{\Lambda}^\top) & \mathbf{C}_{[i]}^{-1} &= (I_{n_{[i]}} \otimes (\boldsymbol{\Lambda}^\top)^+) \boldsymbol{\rho}_{[i]}^{-1} (I_{n_{[i]}} \otimes \boldsymbol{\Lambda}^+) \\ \mathbf{H}_i \mathbf{w}_{[i]} &= \mathbf{C}_{i,[i]} \mathbf{C}_{[i]}^{-1} \mathbf{w}_{[i]} \\ &= (I_{n_i} \otimes \boldsymbol{\Lambda}) \boldsymbol{\rho}_{i,[i]} (I_{n_{[i]}} \otimes \boldsymbol{\Lambda}^\top) (I_{n_{[i]}} \otimes (\boldsymbol{\Lambda}^\top)^+) \boldsymbol{\rho}_{[i]}^{-1} (I_{n_{[i]}} \otimes \boldsymbol{\Lambda}^+) (I_{n_{[i]}} \otimes \boldsymbol{\Lambda}) \mathbf{v}_{[i]} & (14) \\ &= (I_{n_i} \otimes \boldsymbol{\Lambda}) \boldsymbol{\rho}_{i,[i]} (I_{n_{[i]}} \otimes \boldsymbol{\Lambda}^\top (\boldsymbol{\Lambda}^\top)^+) \boldsymbol{\rho}_{[i]}^{-1} (I_{n_{[i]}} \otimes \boldsymbol{\Lambda}^+ \boldsymbol{\Lambda}) \mathbf{v}_{[i]} \\ &= (I_{n_i} \otimes \boldsymbol{\Lambda}) \boldsymbol{\rho}_{i,[i]} \boldsymbol{\rho}_{[i]}^{-1} \mathbf{v}_{[i]} = (I_{n_i} \otimes \boldsymbol{\Lambda}) \ddot{\mathbf{H}}_i \mathbf{v}_{[i]}, \end{aligned}$$

where we denoted $\ddot{\mathbf{H}}_i = \boldsymbol{\rho}_{i,[i]} \boldsymbol{\rho}_{[i]}^{-1}$ and $\boldsymbol{\Lambda}^+$ denotes the Moore-Penrose pseudoinverse $\boldsymbol{\Lambda}^+ = (\boldsymbol{\Lambda}^\top \boldsymbol{\Lambda})^{-1}$ (which exists because $\boldsymbol{\Lambda}$ is assumed of full column rank), and therefore $\boldsymbol{\Lambda}^+ \boldsymbol{\Lambda} =$

$I_k = \mathbf{\Lambda}^\top (\mathbf{\Lambda}^\top)^+$. Similarly,

$$\begin{aligned} \mathbf{R}_i &= \mathbf{C}_i - \mathbf{H}_i \mathbf{C}_{[i],i} = (I_{n_i} \otimes \mathbf{\Lambda}) \boldsymbol{\rho}_i (I_{n_i} \otimes \mathbf{\Lambda}^\top) - (I_{n_i} \otimes \mathbf{\Lambda}) \boldsymbol{\rho}_{i,[i]} \boldsymbol{\rho}_{[i]}^{-1} \boldsymbol{\rho}_{[i],i} (I_{n_i} \otimes \mathbf{\Lambda}^\top) \\ &= (I_{n_i} \otimes \mathbf{\Lambda}) \left(\boldsymbol{\rho}_i - \boldsymbol{\rho}_{i,[i]} \boldsymbol{\rho}_{[i]}^{-1} \boldsymbol{\rho}_{[i],i} \right) (I_{n_i} \otimes \mathbf{\Lambda}^\top) = (I_{n_i} \otimes \mathbf{\Lambda}) \ddot{\mathbf{R}}_i (I_{n_i} \otimes \mathbf{\Lambda}^\top). \end{aligned} \quad (15)$$

Then

$$\begin{aligned} \pi(\mathbf{w}_i | \mathbf{w}_{[i]}) &\propto |\mathbf{R}_i|^{-\frac{1}{2}} \exp \left\{ -\frac{1}{2} (\mathbf{w}_i - \mathbf{H}_i \mathbf{w}_{[i]})^\top \mathbf{R}_i (\mathbf{w}_i - \mathbf{H}_i \mathbf{w}_{[i]}) \right\} \\ &= |(I_{n_i} \otimes \mathbf{\Lambda}) \ddot{\mathbf{R}}_i (I_{n_i} \otimes \mathbf{\Lambda}^\top)|^{-\frac{1}{2}} \cdot \\ &\quad \cdot \exp \left\{ -\frac{1}{2} ((I_{n_i} \otimes \mathbf{\Lambda}) \mathbf{v}_i - (I_{n_i} \otimes \mathbf{\Lambda}) \ddot{\mathbf{H}}_i \mathbf{v}_{[i]})^\top \cdot \right. \\ &\quad \cdot \left. \left((I_{n_i} \otimes \mathbf{\Lambda}) \ddot{\mathbf{R}}_i (I_{n_i} \otimes \mathbf{\Lambda}^\top) \right)^{-1} ((I_{n_i} \otimes \mathbf{\Lambda}) \mathbf{v}_i - (I_{n_i} \otimes \mathbf{\Lambda}) \ddot{\mathbf{H}}_i \mathbf{v}_{[i]}) \right\} \\ &= |\ddot{\mathbf{R}}_i|^{-\frac{1}{2}} \exp \left\{ -\frac{1}{2} (\mathbf{v}_i - \ddot{\mathbf{H}}_i \mathbf{v}_{[i]})^\top \ddot{\mathbf{R}}_i^{-1} (\mathbf{v}_i - \ddot{\mathbf{H}}_i \mathbf{v}_{[i]}) \right\} = \pi(\mathbf{v}_i | \mathbf{v}_{[i]}). \end{aligned} \quad (16)$$

We then proceed by reordering \mathbf{v}_i , $\ddot{\mathbf{H}}_i$ and $\ddot{\mathbf{R}}_i$ by factor index (from $h = 1, \dots, k$) rather than by location (see discussion above). After letting K_i denote the appropriate permutation matrix that applies such reordering and letting $\mathbf{v}_i^{(h)}$ be the $n_i \times 1$ vector whose elements are realizations of the h th latent factor at the reference subset \mathcal{S}_i , we can write

$$\begin{aligned} K_i \mathbf{v}_i &= \begin{bmatrix} \mathbf{v}_i^{(1)} \\ \vdots \\ \mathbf{v}_i^{(k)} \end{bmatrix} & K_i \ddot{\mathbf{H}}_i \mathbf{v}_i &= \begin{bmatrix} \tilde{\mathbf{H}}_i^{(1)} \mathbf{v}_{[i]}^{(1)} \\ \vdots \\ \tilde{\mathbf{H}}_i^{(1)} \mathbf{v}_{[i]}^{(k)} \end{bmatrix} \\ K_i \ddot{\mathbf{R}}_i^{-1} K_i^\top &= \text{blockdiag} \left\{ \tilde{\mathbf{R}}_i^{(1)}, \dots, \tilde{\mathbf{R}}_i^{(k)} \right\}, \end{aligned}$$

where $\tilde{\mathbf{H}}_i^{(h)} \mathbf{v}_{[i]}^{(h)} = \rho_{i,[i]}^{(h)} \rho_{[i]}^{(h)-1} \mathbf{v}_{[i]}^{(h)}$ and $\tilde{\mathbf{R}}_i^{(h)} = \rho_i^{(h)} - \rho_{i,[i]}^{(h)} \rho_{[i]}^{(h)-1} \rho_{[i],i}^{(h)}$, with $\rho_{i,[i]}^{(h)}$ denoting the correlation function of the h th LMC component evaluated between pairs of \mathcal{S}_i and $\mathcal{S}_{[i]}$ and the other terms are defined analogously. Since reordering does not affect the joint density $\pi(\mathbf{v}_i | \mathbf{v}_{[i]})$, we obtain

$$\pi(K_i \mathbf{v}_i | \mathbf{v}_{[i]}) = \pi(\mathbf{v}_i | \mathbf{v}_{[i]}) = \prod_{h=1}^k N(\mathbf{v}_i^{(h)}; \tilde{\mathbf{H}}_i^{(h)}, \tilde{\mathbf{R}}_i^{(h)}).$$

We have shown that the density of $(\mathbf{w}_i | \mathbf{w}_{[i]})$ is the same as that of $(\mathbf{v}_i | \mathbf{v}_{[i]})$ and that it can be written as a product of independent conditional densities. Then, for $i = 1, \dots, M$:

$$\begin{aligned} \pi_{\mathcal{G}}(\mathbf{w}_{\mathcal{S}}) &= \prod_{i=1}^M \pi(\mathbf{w}_i | \mathbf{w}_{[i]}) = \prod_{i=1}^M \pi(\mathbf{v}_i | \mathbf{v}_{[i]}) = \prod_{i=1}^M \prod_{h=1}^k N(\mathbf{v}_i^{(h)}; \tilde{\mathbf{H}}_i^{(h)}, \tilde{\mathbf{R}}_i^{(h)}) \\ &= \prod_{h=1}^k \prod_{i=1}^M N(\mathbf{v}_i^{(h)}; \tilde{\mathbf{H}}_i^{(h)}, \tilde{\mathbf{R}}_i^{(h)}) = \prod_{h=1}^k \pi_{\mathcal{G}}^{(h)}(\mathbf{v}_S^{(h)}). \end{aligned}$$

We have shown that the meshed density $\pi_{\mathcal{G}}$ at \mathcal{S} is equal to the product of k independent meshed densities which are defined on the same DAG \mathcal{G} and the same partitioning of the spatial domain (i.e., k independent MGPs). \blacksquare

D.2 Langevin methods for coregionalized MGPs

We now show how Algorithm 1 is specified for the latent MGP model with LMC cross-covariance using MELANGE when targeting (11). Let K_i be the permutation matrix that reorders \mathbf{v}_i by factor, i.e. the h th block of $\tilde{\mathbf{v}}_i = K_i \mathbf{v}_i$ is the $n_i \times 1$ vector $v_i^{(h)}$, for $h = 1, \dots, k$. Then, after letting $\mathbf{H}_i = (I_{n_i} \otimes \mathbf{\Lambda}) \ddot{\mathbf{H}}_i$ and $\mathbf{R}_i = (I_{n_i} \otimes \mathbf{\Lambda}) \ddot{\mathbf{R}}_i (I_{n_i} \otimes \mathbf{\Lambda}^\top)$ and $r_j^{(h)} = v_j^{(h)} - \tilde{H}_{[j] \setminus \{i\}}^{(h)} v_{[j] \setminus \{i\}}^{(h)}$, the gradient $\nabla_{\mathbf{v}_i} p(\mathbf{v}_i | \text{---})$ can be found as we get

$$\begin{aligned} \nabla_{\mathbf{v}_i} p(\mathbf{v}_i | \text{---}) &= -\ddot{\mathbf{R}}_i \left(\mathbf{v}_i - \ddot{\mathbf{H}}_i \mathbf{v}_{[i]} \right) + \mathbf{f}_i \\ &= -K_i^\top \begin{bmatrix} \tilde{R}_i^{(1)} \left(v_i^{(1)} - \tilde{H}_i^{(1)} v_{[i]}^{(1)} \right) + \tilde{H}_{i \rightarrow j}^{(1)\top} \tilde{R}_j^{(1)-1} \left(r_j^{(1)} - \tilde{H}_{i \rightarrow j}^{(1)} v_i^{(1)} \right) \\ \vdots \\ \tilde{R}_i^{(k)} \left(v_i^{(k)} - \tilde{H}_i^{(k)} v_{[i]}^{(k)} \right) + \tilde{H}_{i \rightarrow j}^{(k)\top} \tilde{R}_j^{(k)-1} \left(r_j^{(k)} - \tilde{H}_{i \rightarrow j}^{(k)} v_i^{(k)} \right) \end{bmatrix} + \mathbf{f}_i, \end{aligned} \quad (17)$$

where, letting $\mathcal{S}_i = \{\ell_1, \dots, \ell_{n_i}\}$, we compute $\mathbf{f}_i = (\mathbf{f}_{i, \ell_1}^\top, \dots, \mathbf{f}_{i, \ell_{n_i}}^\top)^\top$ as the $n_i k \times 1$ vector whose ℓ block is

$$\mathbf{f}_{i, \ell} = \mathbf{\Lambda}^\top \begin{bmatrix} \nabla_{\mathbf{v}(\ell)} dF_1(y_1(\ell) | \mathbf{v}(\ell), \boldsymbol{\lambda}_{[1, :]}, \boldsymbol{\beta}_q, \gamma_q) \\ \vdots \\ \nabla_{\mathbf{v}(\ell)} dF_q(y_q(\ell) | \mathbf{v}(\ell), \boldsymbol{\lambda}_{[q, :]}, \boldsymbol{\beta}_q, \gamma_q) \end{bmatrix}.$$

For SM-MALA and SiMPA (Algorithm 2) we compute

$$\mathbf{G}_{\mathbf{v}_i}^{-1} = K_i^\top \left(\bigoplus \left\{ \tilde{R}_i^{(h)} + \tilde{H}_{i \rightarrow j}^{(h)\top} \tilde{R}_j^{(h)-1} \tilde{H}_{i \rightarrow j}^{(h)} \right\}_{h=1}^k + K_i \mathbf{F}_i K_i^\top \right) K_i, \quad (18)$$

where \bigoplus is the direct sum operator, $\mathbf{F}_i = \bigoplus \{ \mathbf{A}_i(\ell) \}_{\ell \in \mathcal{S}_i}$, and after letting $x_j(\ell) = \boldsymbol{\lambda}_{[j, :]} \mathbf{v}(\ell)$, we compute $\mathbf{A}_i(\ell) = -\sum_{j=1}^q \boldsymbol{\lambda}_{[j, :]}^\top \boldsymbol{\lambda}_{[j, :]} E \left[\frac{\delta^2}{\delta^2 x_j(\ell)} \log dF_j(y_j(\ell) | \mathbf{v}(\ell), \boldsymbol{\lambda}_{[j, :]}, \boldsymbol{\beta}_j, \gamma_j) \right]$.

D.3 Complexity in fitting coregionalized cubic MGPs

We now consider model (9) and replace the GP prior with an MGP based on LMCs (as in Section 4.3) using a cubic mesh (Figure 3), whose main feature is that the number of parents of each reference node is at most d when the dimension of the input space is d (in spatial settings, $d = 2$). The resulting coregionalized QMGP is implemented on k factors to model dependence across $q \geq k$ outcomes, when at n locations we observe at least one of them. We assume $\overline{\mathcal{T}} = \emptyset$, SiMPA updates at each block and let H refer to the number of available processors for parallel computations.

In the resulting Algorithm 3, step 1 requires the update of q sets of p covariates plus k factor loadings. SiMPA can be used here for an expected cost at iteration m of $O(\gamma_m n q (p + k)^3 + n q (p + k)^2)$ which is approximately $O(q n (p + k)^2)$ for large m because $\gamma_m \downarrow 0$. The

compute time is $O(\gamma_m n q (p+k)^3 / H + n q (p+k)^2 / H)$, respectively, because $(\beta_j, \lambda_{[j,:]} \perp (\beta_h, \lambda_{[h,:]}) \mid \mathbf{y}_T, \mathbf{v}_S$. Step 2 costs $O(qn)$ flops assuming a Metropolis update, and the compute time is $O(qn/H)$. Step 3 involves the evaluation of k independent sets of MGP densities, each of which is a product of M Gaussian conditional densities. We make the simplifying assumption that $n_i \approx m \approx n/M$ and $n_{[i]} \leq dm \approx dn/M$ for all $i = 1, \dots, M$ —we are taking M partitions of size m and a cubic mesh which attributes at most d parents to each node in the DAG. The cost for this update is due to computing $\tilde{\mathbf{R}}_i$ for all i , which is $O(kM(dm)^3) = O(nkd^3m^2)$ flops in $O(nkd^3m^2/H)$ time. Finally, reference sampling of \mathbf{v}_i , $i = 1, \dots, M$, whose sizes are mk , is performed via SiMPA in $O(\gamma_m n m^2 k^3 + n m k^2)$ flops and in $O(\gamma_m n m^2 k^3 / H + n m k^2 / H)$ time, respectively, assuming that each color of \mathcal{G} includes at least H nodes. In summary, the cost of a k -factor coregionalized QMGP fit via SiMPA is linear in n and q , which may be large, quadratic on k and p , which we assume relatively small, and cubic on the domain dimension d , which is 2 or 3 for the spatial and spatiotemporal settings on which we focus.

Appendix E. Applications Supplement

In all our applications, all methods are configured to use up to 16 CPU threads in a workstation with 128GB memory and an AMD Ryzen 9 5950X CPU on the Ubuntu 22.04.2 LTS operating system and using Intel MKL version 2019.5.28 for BLAS/LAPACK. R package `meshed` (v.0.2) allows one to set the number of OpenMP (Dagum and Menon, 1998) threads, whereas `Hmsc` takes advantage of parallelization via BLAS when performing expensive operations (e.g., `chol(\cdot)`). The R-INLA package used to implement SPDE-INLA methods can similarly take advantage of multithreaded operations.

E.1 Bivariate counts analysis on 750 synthetic datasets

The comparison above is based on a single dataset; we replicate the same analysis on 750 smaller datasets. We generate Poisson data on a 50×50 regular grid, for a total of 2500 observations for $y_j(\ell) \sim \text{Pois}(\exp\{\eta_j(\ell)\})$ where $\boldsymbol{\eta}(\ell) = \mathbf{A}\mathbf{v}(\ell)$ and $\mathbf{v}(\cdot)$ is a bivariate GP with independent Matérn correlations with $\nu_j = 1/2$ for $j = 1, 2$ and $\phi_2 = 2.5$. We choose $\phi_1 \in \{2.5, 12.5, 25\}$. We introduce missing values at 1/5 of spatial locations independently for each outcome. We fix the 2×2 loading matrix via

$$\mathbf{A} = \begin{bmatrix} \lambda_{11} & \lambda_{12} \\ \lambda_{21} & \lambda_{22} \end{bmatrix} = \text{chol} \left(\begin{bmatrix} \lambda_1 & 0 \\ 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & \rho \\ \rho & 1 \end{bmatrix} \cdot \begin{bmatrix} \lambda_1 & 0 \\ 0 & 1 \end{bmatrix} \right),$$

which implies $\lambda_{11} = \lambda_1$, $\lambda_{12} = 0$, and λ_{21} and λ_{22} are such the latent correlation between the first and second margin is exactly ρ . We choose $\lambda_1 \in \{\frac{\sqrt{2}}{2}, 2\}$ and $\rho \in \{-.9, -0.25, 0, 0.65, .9\}$. We generate 25 datasets for every combination of values of ϕ_1 , λ_1 and ρ . We target estimation of the latent correlation $\rho = \text{Corr}(w_1(\ell), w_2(\ell))$ in terms of absolute error and efficiency (ESS/s), along with the empirical coverage of 95% intervals for the log-intensity for both outcomes. We compare SiMPA with several other methods – all the coregionalized QMGP methods use parameter expansion as in Peruzzi et al. (2021). Figure 15 and Table 5 summarize our findings across the 750 datasets: SiMPA has low estimation error, high sampling efficiency, and excellent uncertainty quantification relative to all other tested methods.

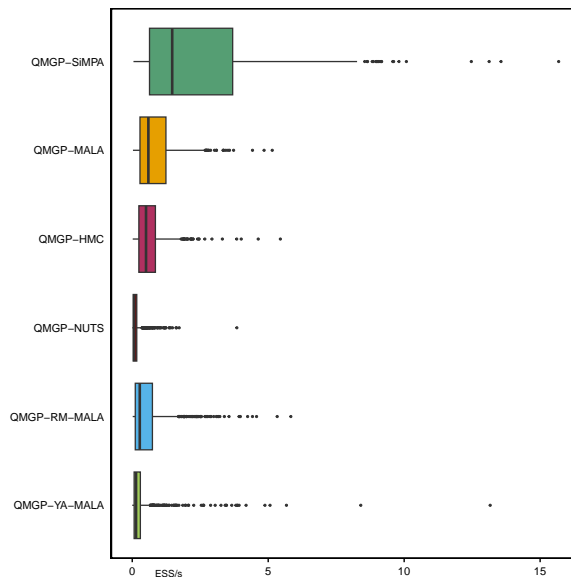


Figure 15: Efficiency in terms of ESS/s in the estimation of ρ over 750 simulated datasets.

Method	ρ		
	RMSE	RMSPE	Covg _{95%}
QMGP-SiMPA	0.08	0.45	0.95
QMGP-MALA	0.09	0.46	0.94
QMGP-HMC	0.09	0.55	0.93
QMGP-NUTS	0.16	0.46	0.93
QMGP-RM-MALA	0.37	0.86	0.87
QMGP-YA-MALA	0.42	1.83	0.07
QMGP(AG10)-NUTS	0.56	0.91	0.94
QMTP(AG10)-NUTS	0.47	0.46	0.92
SPDE-INLA	0.21	0.66	0.67

Table 5: Performance summary across 750 datasets in the estimation of the latent correlation and the linear predictor on the test sets.

E.2 Latent process sampler efficiency in multi-type data

The analysis in the previous section models both outcomes as Poisson counts. In this section, we use the same setup and $\lambda_1 \in \{2.5, 12.5\}$, but consider the following pairs of outcome types: $\{(\text{Gaussian}, \text{Poisson}), (\text{Neg. Binomial}, \text{Binomial}), (\text{Neg. Binomial}, \text{Poisson})\}$, for a total of 1500 datasets, of which 500 include a Binomial or Gaussian outcome and 1000 include a Poisson or Neg. Binomial outcome. Because we target a comparison of posterior sampling efficiency in integrating out the latent spatial effects via MCMC, we fix all unknowns ($\mathbf{\Lambda}$, ϕ_1 , ϕ_2) to their true values except for the latent process. For each dataset, we calculate ESS/s for samples of $\mathbf{w}(\ell_i)$, $\ell_i = 1, \dots, n$. After computing the median ESS/s as a summary efficiency measure for each dataset, we compute the mean of this measure over all datasets. Efficiency summary results are reported in Table 6. We also report each method’s RMSPE and coverage about $\eta_j(\ell)$, $j = 1, 2$ in Table 7. SiMPA is again more efficient than other

methods in integrating out the spatial effects, while matching or outperforming them in out-of-sample inference about model parameters.

Method	Binomial	Gaussian	Negative Binomial	Poisson
MALA	1.15	8.35	1.31	2.81
NUTS	0.25	2.15	0.32	0.68
SiMPA	4.26	17.65	9.03	8.90
SM-MALA	1.81	7.85	3.45	3.59

Table 6: Efficiency in posterior sampling of $\boldsymbol{w}(\cdot)$, in terms of ESS/s, for different types of outcomes in the bivariate synthetic data application with multi-type outcomes.

Method	RMSPE				Covg. 95%			
	Binomial	Gaussian	Neg. Bin.	Poisson	Binomial	Gaussian	Neg. Bin.	Poisson
MALA	0.450	0.332	14.206	1.198	0.897	0.942	0.899	0.930
NUTS	0.449	0.328	13.932	1.195	0.932	0.944	0.934	0.941
SiMPA	0.449	0.327	13.923	1.194	0.944	0.948	0.947	0.947
SM-MALA	0.449	0.327	13.971	1.212	0.943	0.948	0.939	0.940

Table 7: RMSPE in predicting $\eta_j(\boldsymbol{\ell})$ at locations in the test set and empirical coverage of 95% credible intervals about $\eta_j(\boldsymbol{\ell})$, $j = 1, 2$, for different types of outcomes in the bivariate synthetic data application with multi-type outcomes.

E.3 Multi-species N-mixture abundance modeling

The total number of individuals of a certain animal species in a region is known as the local abundance. Community ecologists seek to estimate abundance using spatially replicated count data of multiple species. At each spatial location, the observed counts correspond to a portion of the latent abundance of each of q species. The unobserved abundance of species j can be estimated via a model for count data that accounts for imperfect detection. See Royle (2004), Mimmagh et al. (2022) and reference therein. In the context of joint species distribution models of count data, one lets the local abundance depend on covariates and latent variables accounting for cross-species dependence through a Poisson log-linear model, with the observed abundance then having a conditional binomial likelihood. Here, we consider an extension of Mimmagh et al. (2022) to include MGP random effects:

$$\begin{aligned}
 v_h(\cdot) &\sim MGP_{\mathcal{G}}(\mathbf{0}, \rho_h(\cdot, \cdot; \boldsymbol{\theta}_h)), & h = 1, \dots, k \\
 w_j(\boldsymbol{\ell}) &= \boldsymbol{\lambda}_{[j,:]} \boldsymbol{v}(\boldsymbol{\ell}) & j = 1, \dots, q \\
 N_j(\boldsymbol{\ell}) \mid \boldsymbol{\beta}_j, \boldsymbol{\lambda}_{[j,:]}, \boldsymbol{v}(\boldsymbol{\ell}) &\sim \text{Poisson}(\mu_j(\boldsymbol{\ell})) & \mu_j(\boldsymbol{\ell}) = \exp\{\boldsymbol{x}_j(\boldsymbol{\ell})^\top \boldsymbol{\beta}_j + w_j(\boldsymbol{\ell})\} \\
 y_j(\boldsymbol{\ell}) \mid N_j(\boldsymbol{\ell}), \boldsymbol{\xi}_j &\sim \text{Binomial}(N_j(\boldsymbol{\ell}), p_j(\boldsymbol{\ell})) & p_j(\boldsymbol{\ell}) = \left[1 + \exp\{-\boldsymbol{z}_j(\boldsymbol{\ell})^\top \boldsymbol{\xi}_j\}\right]^{-1},
 \end{aligned} \tag{19}$$

where we let a set of species-specific covariates $\boldsymbol{x}_j(\boldsymbol{\ell})$ explain the latent species abundance and another set $\boldsymbol{z}_j(\boldsymbol{\ell})$ impact the detection probability and hence the observed counts $y_j(\boldsymbol{\ell})$. A latent factor model with $k \leq q$ spatial random effects is used to characterize dependence across species in abundances. Applied goals include the estimation of $\boldsymbol{\beta}_j$ and $\boldsymbol{\xi}_j$ for $j =$

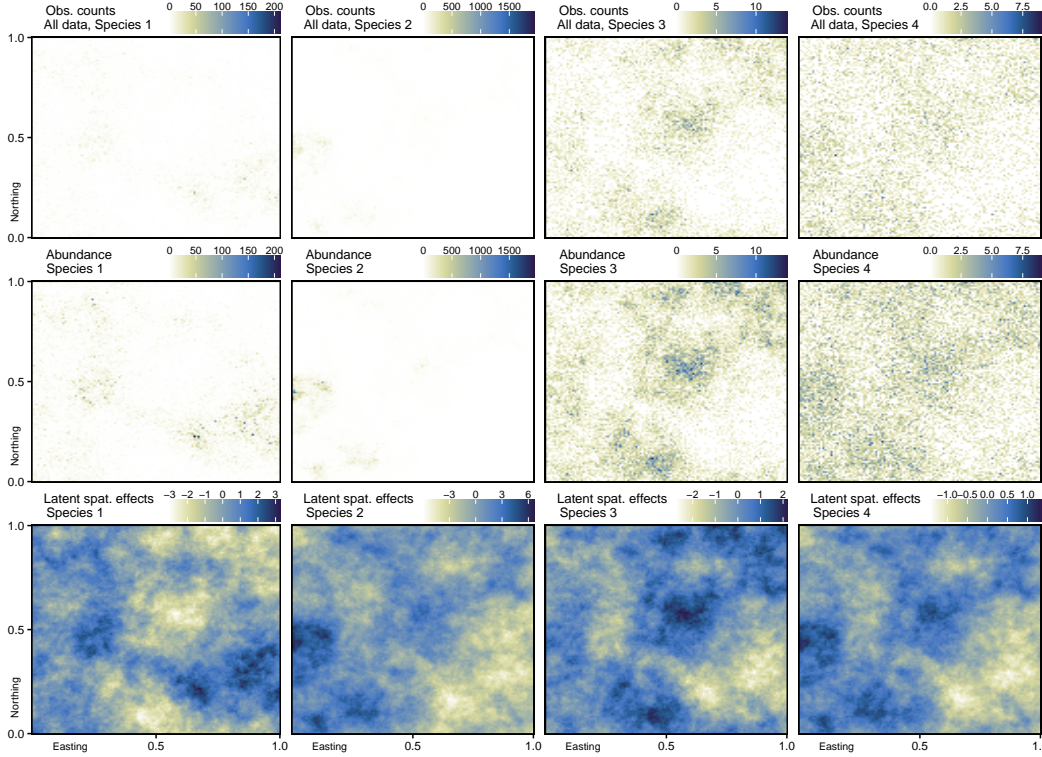


Figure 16: Simulated data on multi-species abundance. Top row: all counts observed with imperfect detection, including the 20% missing from the training data; mid row: unobserved species abundance; bottom row: realization of $w_j(\boldsymbol{\ell}) = \boldsymbol{\lambda}_{[j,:]} \mathbf{v}(\boldsymbol{\ell})$, $\boldsymbol{\ell} \in \mathcal{D}$.

$1, \dots, q$, the cross-covariance function $\mathbf{C}_\theta = \boldsymbol{\Lambda} \boldsymbol{\rho}(\boldsymbol{\ell}, \boldsymbol{\ell}', \boldsymbol{\Phi}) \boldsymbol{\Lambda}^\top$ via the estimation of $\boldsymbol{\theta} = (\text{vec}(\boldsymbol{\Lambda})^\top, \boldsymbol{\Phi}^\top)^\top$, and the local abundance of species j at $\boldsymbol{\ell}$, $N_j(\boldsymbol{\ell})$. Posterior computations for (19) simplify by marginalizing $N_j(\boldsymbol{\ell})$ from the model likelihood; the marginal model is $p(y_j(\boldsymbol{\ell}) \mid \text{---}) = \text{Poisson}(p_j(\boldsymbol{\ell}) \mu_j(\boldsymbol{\ell}))$. After collecting posterior samples of $\boldsymbol{\beta}_j, \boldsymbol{\xi}_j, \boldsymbol{\Lambda}, \mathbf{v}$, we estimate $N_j(\boldsymbol{\ell})$ using the fact that $N_j(\boldsymbol{\ell}) \mid N_j(\boldsymbol{\ell}) > y_j(\boldsymbol{\ell}) = y_j(\boldsymbol{\ell}) + \tilde{N}_j(\boldsymbol{\ell})$, where $\tilde{N}_j(\boldsymbol{\ell}) \sim \text{Poisson}([1 - p_j(\boldsymbol{\ell})] \mu_j(\boldsymbol{\ell}))$. If $y_j(\boldsymbol{\ell})$ is missing, we proceed by first sampling from $\pi(\mathbf{v}(\boldsymbol{\ell}) \mid \mathbf{v}_S, \boldsymbol{\Phi})$, then $N_j(\boldsymbol{\ell}) \sim \text{Poisson}(\mu_j(\boldsymbol{\ell}))$.

We simulate abundance data of $q = 4$ species at $n = 14,400$ spatial locations on a regular grid using model (19). We sample $k = 2$ latent factors from independent unrestricted GPs with exponential correlation and spatial decays $\phi_1 = \phi_2 = 4$. The factor loadings are set to $(\lambda_{11}, \lambda_{21}, \lambda_{31}, \lambda_{41}, \lambda_{22}, \lambda_{32}, \lambda_{42}) = (1.3, -0.65, -0.9, -0.3, 2, 0.35, 0.4)$; these values lead to latent spatial cross-species correlations ranging from $\text{corr}(w_3(\boldsymbol{\ell}), w_1(\boldsymbol{\ell})) \approx -0.93$ to $\text{corr}(w_4(\boldsymbol{\ell}), w_2(\boldsymbol{\ell})) \approx 0.95$. These correlations decrease for increasing spatial distances as modeled by the underlying exponential covariances. In order to generate the latent abundance and the observed counts at each location, we sample the covariate vector $(x(\boldsymbol{\ell}), \mathbf{z}(\boldsymbol{\ell})^\top)^\top$ independently from a Gaussian distribution with correlation matrix Σ_x whose off-diagonal elements are $\sigma_{x,z_1} = 0.8$, $\sigma_{x,z_2} = -0.3$, $\sigma_{z_1,z_2} = -0.7$. We let $(\beta_1, \beta_2, \beta_3, \beta_4) = (-1, 0.5, 0, 0)$ and $\boldsymbol{\xi}_1 = (1, -1)^\top$, $\boldsymbol{\xi}_2 = (-1, 1)^\top$, $\boldsymbol{\xi}_3 = (0.5, -0.5)^\top$, $\boldsymbol{\xi}_4 = (-1, -1)^\top$. Finally, for each of the 4 species, we introduce missingness by indepen-

dently dropping 20% of the observed count data from the training set uniformly at random; the counts of at least one species are missing at 8,480 locations. Because not all species are observed at all spatial locations, the resulting data are misaligned. This scenario mirrors a situation in which a subset of the total number of individuals of species j are counted at a subset of all locations. Figure 16 reports the full dataset (including the missing data) along with the latent variables.

Method	j	RMSE $N_j(\ell)$	MAE $N_j(\ell)$	CI ₉₅ $\mathbf{w}_j(\ell)$	ESS/s $\mathbf{w}_j(\ell)$	RMSE $(\beta, \xi, \lambda_{[j,:]})$	ESS/s $(\beta, \xi, \lambda_{[j,:]})$	Time(s)
SiMPA	1	2.156	0.443	0.949	3.16	0.0197	9.09	139
	2	5.948	0.443	0.951	3.91	0.0289	2.78	
	3	0.951	0.479	0.950	3.42	0.0569	15.8	
	4	0.828	0.494	0.959	4.11	0.0156	25.6	
MALA	1	2.164	0.447	0.950	0.80	0.0203	1.82	114
	2	5.954	0.449	0.947	1.64	0.2640	1.02	
	3	0.950	0.482	0.940	0.90	0.0749	1.86	
	4	0.828	0.495	0.948	1.36	0.0487	1.71	
SM-MALA	1	2.182	0.423	0.929	1.87	0.0473	7.24	194
	2	9.786	0.410	0.932	2.09	0.4330	1.73	
	3	0.951	0.488	0.920	2.14	0.0498	9.77	
	4	0.830	0.488	0.920	2.27	0.0596	19.5	
HMC	1	2.148	0.444	0.945	0.92	0.0421	1.97	214
	2	5.874	0.444	0.950	2.04	0.0370	1.70	
	3	0.951	0.479	0.944	1.13	0.0606	2.32	
	4	0.828	0.494	0.952	1.92	0.0140	2.51	
NUTS	1	2.158	0.442	0.946	0.17	0.0242	0.93	907
	2	5.876	0.446	0.941	0.24	0.0534	0.61	
	3	0.950	0.477	0.944	0.20	0.0563	1.34	
	4	0.828	0.495	0.949	0.27	0.0107	1.31	
YA-MALA	1	2.178	0.424	0.329	0.10	0.1073	0.09	132
	2	7.437	0.414	0.346	0.08	0.2970	0.20	
	3	0.954	0.478	0.346	0.10	0.1151	0.08	
	4	0.829	0.493	0.356	0.09	0.0644	0.10	
Ellipt-SS	1	3.204	0.486	0.937	0.79	0.2953	0.15	264
	2	15.570	0.567	0.821	0.84	0.3609	0.08	
	3	1.014	0.512	0.852	0.79	0.0906	0.33	
	4	0.834	0.509	0.874	0.80	0.0582	0.28	

Table 8: A comparison of posterior sampling methods for fitting the same model for abundance data with imperfect detection based on latent QMGPs. For the four species, we compare the root mean squared error (RMSE) as well as the mean absolute error (MAE) in estimating the latent abundance $N_j(\ell)$. For $\mathbf{w}_j(\ell)$, we report the empirical coverage of 95% credible intervals (CI₉₅) and the median effective sample size (ESS) per unit time across spatial locations. We also report the RMSE and median ESS/s in estimating the vector $(\beta, \xi, \lambda_{[j,:]})$ for each species.

We fit model (19) with a QMGP prior on the latent effects. To build the QMGP prior, we use axis-parallel partitioning to tessellate the spatial domain into $M = 400$ blocks each including 36 spatial locations. We choose this partitioning setup to ensure all sampling methods proceed swiftly and without making the overly restrictive spatial conditional independence assumptions that would result from a finer partitioning scheme. We detail the

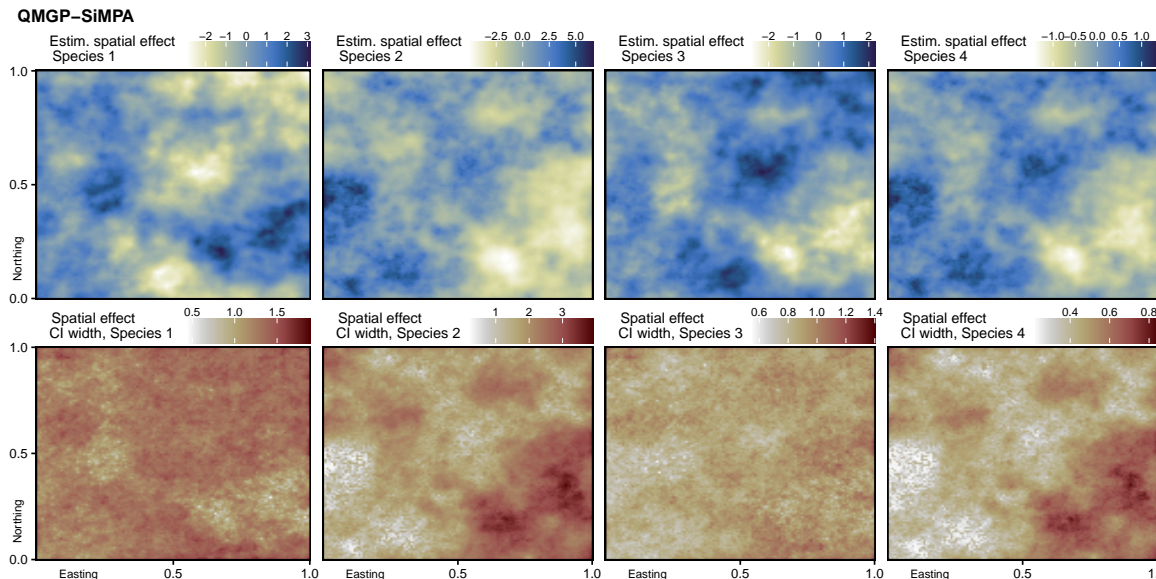


Figure 17: Estimation of the latent spatial effects in the multi-species abundance model. Top row: posterior mean of the species-specific spatial random effects $\mathbf{w}_j(\ell)$; bottom row: width of the 95% pointwise credible interval on $\mathbf{w}_j(\ell)$ as computed via SiMPA.

common posterior sampling algorithm used for fitting model (19) in Appendix E.4 as a minor modification of Algorithm 3.

We compare our proposed SiMPA with MALA, simplified Riemannian manifold MALA (RM-MALA; Girolami and Calderhead 2011), HMC and NUTS with dual averaging (Algorithms 5 and 6 in Hoffman and Gelman 2014, respectively), the elliptical slice sampler (Murray et al., 2010), and YA-MALA. All methods perform 20,000 MCMC iterations, of which we drop the first half as burn-in. All gradient-based methods use dual averaging to adapt ε for $T^{\text{adapt}} = 10,000$ iterations. Figure 17 reports the SiMPA-estimated latent effects along with uncertainty quantification. Table 8 summarises our findings: because all sampling methods target the same posterior distribution, we do not expect major discrepancies in point estimates. Our SiMPA method is on par with other state-of-the-art gradient-based methods when estimating unknown model parameters, but outperforms them in terms of sampling efficiency measured as ESS per unit time. Because the SiMPA 95% intervals on the latent effects are subjectively better calibrated than those from other methods, it more robustly quantifies uncertainty about the latent spatial effects. Finally, because SiMPA and YA-MALA adapt the preconditioner at the same iterations, we conclude that SiMPA is a much more efficient adaptation scheme for MALA preconditioning in this context.

E.4 Posterior sampling of the multi-species N-mixture model

All the tested sampling methods are used in steps 1 and 3 of Algorithm 5.

Block updating $\beta_j, \xi_j, \lambda_{[j,:]}$ using SiMPA requires second order information about the target. Because $\mathbf{x}_j(\ell)^\top \beta_j + \lambda_{[j,:]} \mathbf{v}(\ell) = (\mathbf{x}_j(\ell)^\top, \mathbf{v}(\ell)^\top) (\beta_j^\top, \lambda_{[j,:]}) = \tilde{\mathbf{x}}_j(\ell) \tilde{\beta}_j$, we can proceed without loss of generality by outlining the block-update for (β_j, ξ_j) in the model without

Algorithm 5 Posterior sampling and prediction of LMC model (19) with MGP priors.

Initialize $\beta_j^{(0)}, \xi_j^{(0)}, \Lambda^{(0)}$ for $j = 1, \dots, q$, $\mathbf{v}_S^{(0)}$, and $\Phi^{(0)}$
for $t \in \{1, \dots, T^*, T^* + 1, \dots, T^* + T\}$ **do** ▷ sequential MCMC loop
 for $j = 1, \dots, q$, **do in parallel**
1: Block-update $\beta_j^{(t)}, \xi_j^{(t)}, \lambda_{[j,:]}^{(t)} \mid \mathbf{y}_T, \mathbf{v}_S^{(t-1)}$
2: use Metropolis-Hastings to update $\Phi^{(t)} \mid \mathbf{v}_S^{(t-1)}$
 for $c \in \text{Colors}(\mathcal{G})$ **do** ▷ sequential
 for $i \in \{i : \text{Color}(a_i) = c\}$ **do in parallel**
3: Update $\mathbf{v}_i^{(t)} \mid \mathbf{v}_{\text{mb}(i)}^{(t)}, \mathbf{y}_i, \Lambda^{(t)}, \Phi^{(t)}, \{\beta_j^{(t)}, \gamma_j^{(t)}\}_{j=1}^q$

spatial effects. After letting $\pi_j(\ell) = (1 + \exp\{-\mathbf{z}_j(\ell)^\top \xi_j\})^{-1}$ and $\alpha_j(\ell) = \exp\{\mathbf{x}_j(\ell)^\top \beta_j\}$, we find

$$-\frac{\delta^2 P(y_j(\ell) = y)}{\delta^2(\beta_j, \xi_j)} = \begin{bmatrix} g_{11} & g_{12} \\ g_{12} & g_{22} \end{bmatrix},$$

where $g_{11} = \alpha_j(\ell)\pi_j(\ell)\mathbf{x}_j(\ell)\mathbf{x}_j(\ell)^\top$, $g_{12} = \alpha_j(\ell)\pi_j(\ell)(1 - \pi_j(\ell))\mathbf{x}_j(\ell)\mathbf{z}_j(\ell)^\top$, $g_{22} = \pi_j(\ell)(1 - \pi_j(\ell))(\alpha_j(\ell)(2\pi_j(\ell) - 1) - y)\mathbf{z}_j(\ell)\mathbf{z}_j(\ell)^\top$.

Appendix F. Spatial meshing of Student-t processes

GPs are desirable thanks to their convenient properties; however, a similar construction based on cross-covariances can be used to model $\mathbf{w}(\cdot)$ as a q -variate Student-t process (TP), in which case we write $\mathbf{w}(\ell) \sim TP_{\nu_0}(\mathbf{0}, \mathbf{C}(\cdot, \cdot))$ where $\nu_0 > 2 \in \mathfrak{R}$ is a degrees of freedom parameter which controls tail heaviness; similarities with GPs include closedness under marginalization and analytic forms of conditional densities. Then, for any \mathcal{L} , the random effects have a multivariate Student-t distribution, i.e. $\mathbf{w}_{\mathcal{L}} \sim MVT_{\nu_0}(0, \mathbf{C}_{\mathcal{L}})$. In the limiting case $\nu_0 \rightarrow \infty$ one obtains a GP with cross-covariance $\mathbf{C}(\cdot, \cdot)$. Shah et al. (2014) and Chen et al. (2020) introduce and consider TPs as alternatives to GPs in regression, citing improved flexibility owing to the ability of a TP to capture more extreme behavior. There are difficulties associated to using TPs in regression, notably the lack of closedness under linear combinations. This implies that spatial meshing of multivariate TPs built upon a LMC does not equate the LMC of spatially meshed univariate TPs.

The TP is closed under marginalization and conditioning, which implies that it is relatively easy to build a spatially meshed TP. Letting $\mathbf{w}_{\mathcal{L}} = \mathbf{w}$ and $\mathbf{C}_{\mathcal{L}} = \mathbf{C}$ for simplicity, the density of a zero-mean TP evaluated at \mathbf{w} , denoted as $MVT(\nu, \mathbf{0}, \mathbf{C})$, is defined as (Shah et al., 2014)

$$p(\mathbf{w} \mid \nu_0) = \frac{\Gamma(\frac{\nu+n}{2})}{((\nu-2)\pi)^{n/2}} |\mathbf{C}|^{-\frac{1}{2}} \left(1 + \frac{1}{\nu-2} \mathbf{w}^\top \mathbf{C}^{-1} \mathbf{w}\right)^{-\frac{\nu+n}{2}}.$$

The above density formulation leads to $\text{cov}(\mathbf{w}) = \mathbf{C}$. Closedness of the TP under marginalization and conditioning leads to the TP conditional densities also being multivariate t's; we find

$$\pi(\mathbf{w}_i \mid \mathbf{w}_{[i]}) \sim MVT\left(\nu + n_{[i]}, \mathbf{H}_i \mathbf{w}_{[i]}, \frac{\mathbf{b} + \nu - 2}{n_{[i]} + \nu - 2} \mathbf{R}_i\right),$$

where \mathbf{H}_i and \mathbf{R}_i are defined like in the GP, and the new term $\mathbf{b} = \mathbf{w}_{[i]}^\top \mathbf{C}_{[i]}^{-1} \mathbf{w}_{[i]}$ determines how the conditional variance of $\mathbf{w}_i | \mathbf{w}_{[i]}$ also depends on the values of $\mathbf{w}_{[i]}$. In fact, $\text{cov}(\mathbf{w}_i | \mathbf{w}_{[i]}) = \frac{\mathbf{b} + \nu - 2}{n_{[i]} + \nu - 2} \mathbf{R}_i$, where the (covariance-weighted) sum of squares \mathbf{b} is used to inform the conditional density about the observed variance in the conditioning set. In fact, if $\mathbf{b}/n_{[i]}$ is large (i.e., the conditioning set has larger spread), then the conditional variance is also larger. This intuitive behavior is missing from a GP, which we obtain in this context by letting $\nu \rightarrow \infty$ (or $n_{[i]} \rightarrow \infty$, which is uninteresting when doing spatial meshing).

Gradient based sampling for MTPs.

When building gradient-based MCMC methods for posterior sampling MTPs, we require $\nabla_{\mathbf{w}_i} \log p(\mathbf{w}_i | -) = \mathbf{f}_i + \frac{\delta}{\delta \mathbf{w}_i} \log p(\mathbf{w}_i | \mathbf{w}_{[i]}, \boldsymbol{\theta}) + \sum_{j \rightarrow \{i \rightarrow j\}} \frac{\delta}{\delta \mathbf{w}_i} \log p(\mathbf{w}_j | \mathbf{w}_i, \mathbf{w}_{[j] \setminus \{i\}}, \boldsymbol{\theta})$. In particular, letting $\mathbf{r}_i = \mathbf{w}_i - \mathbf{H}_i \mathbf{w}_{[i]}$ we find

$$\frac{\delta}{\delta \mathbf{w}_i} \log \pi(\mathbf{w}_i | \mathbf{w}_{[i]}, \boldsymbol{\theta}) = - \frac{\nu + n_i + n_{[i]}}{\nu - 2 + \mathbf{w}_{[i]}^\top \mathbf{C}_{[i]}^{-1} \mathbf{w}_{[i]} + \mathbf{r}_i^\top \mathbf{R}_i^{-1} \mathbf{r}_i} \mathbf{R}_i^{-1} \mathbf{r}_i,$$

and we proceed similarly for $\nabla_{\mathbf{w}_i} \log \pi(\mathbf{w}_j | \mathbf{w}_i, \mathbf{w}_{[j] \setminus \{i\}}, \boldsymbol{\theta})$, where $\pi(\mathbf{w}_j | \mathbf{w}_i, -)$ is a MVT density of \mathbf{w}_j but not of \mathbf{w}_i because MVT are not closed under linear combinations. We partition \mathbf{H}_j and $\mathbf{C}_{[j]}^{-1}$ as

$$\mathbf{H}_j = [A \quad B] \quad \mathbf{C}_{[j]}^{-1} = \begin{bmatrix} C & D \\ D^\top & E \end{bmatrix},$$

with A and C corresponding to blocks which refer to node $a_i \in [j]$, whereas B and E refer to nodes $[j] \setminus a_i$. Let $\tilde{\mathbf{w}}_j = \mathbf{w}_j - B \mathbf{w}_{[j] \setminus \{i\}}$, $\alpha = \frac{\nu + n_j + n_{[j]}}{2}$, $\beta = \nu - 2 + \mathbf{w}_{[j] \setminus \{i\}}^\top E [j] \setminus \{i\} \mathbf{w}_{[j] \setminus \{i\}}$, $c_1 = \mathbf{w}_i^\top C \mathbf{w}_i + 2 \mathbf{w}_i^\top D \mathbf{w}_{[j] \setminus \{i\}}$, $c_2 = (\tilde{\mathbf{w}}_j - A \mathbf{w}_i)^\top \mathbf{R}_j^{-1} (\tilde{\mathbf{w}}_j - A \mathbf{w}_i)$. Then

$$\begin{aligned} \nabla_{\mathbf{w}_i} \log \pi(\mathbf{w}_j | \mathbf{w}_i, \mathbf{w}_{[j] \setminus \{i\}}, \boldsymbol{\theta}) &= \frac{\delta}{\delta \mathbf{w}_i} \left\{ -\alpha \log \left(1 + \frac{(\tilde{\mathbf{w}}_j - A \mathbf{w}_i)^\top \mathbf{R}_j^{-1} (\tilde{\mathbf{w}}_j - A \mathbf{w}_i)}{\mathbf{w}_i^\top C \mathbf{w}_i + 2 \mathbf{w}_i^\top D \mathbf{w}_{[j] \setminus \{i\}} + \beta} \right) \right\} \\ &= \frac{2\alpha}{\beta + c_1 + c_2} \left(A^\top \mathbf{R}_j^{-1} (\mathbf{w}_j - A \mathbf{w}_i) + \frac{c_2 (C \mathbf{w}_i + D \mathbf{w}_{[j] \setminus \{i\}})}{\beta + c_1} \right). \end{aligned}$$

References

- Christophe Andrieu and Johannes Thoms. A tutorial on adaptive MCMC. *Statistics and Computing*, 18:343–373, 2008. doi:10.1007/s11222-008-9110-y.
- Tatiana V. Apanasovich and Marc G. Genton. Cross-covariance functions for multivariate random fields based on latent dimensions. *Biometrika*, 97:15–30, 2010. doi:10.1093/biomet/asp078.
- Yves F. Atchadé. An adaptive version for the Metropolis adjusted Langevin algorithm with a truncated drift. *Methodology and Computing in Applied Probability*, 8:235–254, 2006. doi:10.1007/s11009-006-8550-0.
- Sudipto Banerjee. High-dimensional Bayesian geostatistics. *Bayesian Analysis*, 12(2):583–614, 2017. doi:10.1214/17-BA1056R.
- Sudipto Banerjee. Modeling massive spatial datasets using a conjugate Bayesian linear modeling framework. *Spatial Statistics*, 37:100417, 2020. doi:10.1016/j.spasta.2020.100417.

- Sudipto Banerjee, Alan E. Gelfand, Andrew O. Finley, and Huiyan Sang. Gaussian predictive process models for large spatial data sets. *Journal of the Royal Statistical Society, Series B*, 70:825–848, 2008. doi:10.1111/j.1467-9868.2008.00663.x.
- Sudipto Banerjee, Andrew O. Finley, Patrik Waldmann, and Tore Ericsson. Hierarchical spatial process models for multiple traits in large genetic trials. *Journal of American Statistical Association*, 105(490):506–521, 2010. doi:10.1198/jasa.2009.ap09068.
- Michael Betancourt. A conceptual introduction to Hamiltonian Monte Carlo, 2018. arXiv:1701.02434.
- A. Bhattacharya and D. B. Dunson. Sparse Bayesian infinite factor models. *Biometrika*, 98(2):291–306, 2011. doi:10.1093/biomet/asr013.
- Paul Blomstedt, Diego Parente Paiva Mesquita, Jarno Lintusaari, Tuomas Sivula, Jukka Corander, and Samuel Kaski. Meta-analysis of Bayesian analyses, 2019. arXiv:1904.04484.
- Jonathan R. Bradley, Scott H. Holan, and Christopher K. Wikle. Computationally efficient multivariate spatio-temporal models for high-dimensional count-valued data (with discussion). *Bayesian Analysis*, 13(1):253–310, 2018. doi:10.1214/17-BA1069.
- Jonathan R. Bradley, Scott H. Holan, and Christopher K. Wikle. Bayesian hierarchical models with conjugate full-conditional distributions for dependent data from the natural exponential family. *Journal of the American Statistical Association*, 2019. doi:10.1080/01621459.2019.1677471.
- Bob Carpenter, Andrew Gelman, Matthew D. Hoffman, Daniel Lee, Ben Goodrich, Michael Betancourt, Jiqiang Brubaker, Marcus Guo, Peter Li, and Allen Riddell. Stan: A probabilistic programming language. *Journal of Statistical Software*, 76(1), 2017. doi:10.18637/jss.v076.i01.
- Zexun Chen, Bo Wang, and Alexander N. Gorban. Multivariate Gaussian and Student-t process regression for multi-output prediction. *Neural Computing and Applications*, 32:3005–3028, 2020. doi:10.1007/s00521-019-04687-8.
- Radu V. Craiu, Lawrence Gray, Krzysztof Łatuszyński, Neal Madras, Gareth O. Roberts, and Jeffrey S. Rosenthal. Stability of adversarial Markov chains, with an application to adaptive MCMC algorithms. *The Annals of Applied Probability*, 25(6):3592 – 3623, 2015. doi:10.1214/14-AAP1083.
- Noel Cressie and Gardar Johannesson. Fixed rank kriging for very large spatial data sets. *Journal of the Royal Statistical Society, Series B*, 70:209–226, 2008. doi:10.1111/j.1467-9868.2007.00633.x.
- Leonardo Dagum and Ramesh Menon. OpenMP: an industry standard api for shared-memory programming. *Computational Science & Engineering, IEEE*, 5(1):46–55, 1998.
- Abhirup Datta, Sudipto Banerjee, Andrew O. Finley, and Alan E. Gelfand. Hierarchical nearest-neighbor Gaussian process models for large geostatistical datasets. *Journal of the American Statistical Association*, 111:800–812, 2016a. doi:10.1080/01621459.2015.1044091.
- Abhirup Datta, Sudipto Banerjee, Andrew O. Finley, Nicholas A. S. Hamm, and Martijn Schaap. 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, 2016b. doi:10.1214/16-AOAS931.

- Debangana Dey, Abhirup Datta, and Sudipto Banerjee. Graphical Gaussian process models for highly multivariate spatial data. *Biometrika*, 2021. in press. doi:doi.org/10.1093/biomet/asab061.
- Robert M Dorazio and J. Andrew Royle. Estimating size and composition of biological communities by modeling the occurrence of species. *Journal of the American Statistical Association*, 100(470):389–398, 2005. doi:10.1198/016214505000000015.
- Jeffrey W. Doser, Andrew O. Finley, Marc Kéry, and Elise F. Zipkin. spOccupancy: An R package for single-species, multi-species, and integrated spatial occupancy models. *Methods in Ecology and Evolution*, 13(8):1670–1678, 2022. doi:10.1111/2041-210X.13897.
- Simon Duane, Kennedy A.D., Brian J. Pendleton, and Duncan Roweth. Hybrid Monte Carlo. *Physics Letters B*, 195:216–222, 1987.
- David Dunson and James E. Johndrow. The Hastings algorithm at fifty. *Biometrika*, 107(1):1–23, 2020. doi:10.1093/biomet/asz066.
- Andrew O. Finley, Sudipto Banerjee, Alan R. Ek, and Ronald E. McRoberts. Bayesian multivariate process modeling for prediction of forest attributes. *Journal of Agricultural, Biological, and Environmental Statistics*, 13:60, 2008. doi:10.1198/108571108X273160.
- Andrew O. Finley, Abhirup Datta, Bruce D. Cook, Douglas C. Morton, Hans E. Andersen, and Sudipto Banerjee. Efficient algorithms for Bayesian nearest neighbor Gaussian processes. *Journal of Computational and Graphical Statistics*, 28:401–414, 2019. doi:10.1080/10618600.2018.1537924.
- Reinhard Furrer, Marc G. Genton, and Douglas Nychka. Covariance tapering for interpolation of large spatial datasets. *Journal of Computational and Graphical Statistics*, 15:502–523, 2006. doi:10.1198/106186006X132178.
- Marc G. Genton and William Kleiber. Cross-covariance functions for multivariate geostatistics. *Statistical Science*, 30:147–163, 2015. doi:10.1214/14-STS487.
- Mark Girolami and Ben Calderhead. Riemann manifold Langevin and Hamiltonian Monte Carlo methods. *Journal of the Royal Statistical Society: Series B*, 73(2):123–214, 2011. doi:10.1111/j.1467-9868.2010.00765.x.
- Robert B. Gramacy and Daniel W. Apley. Local Gaussian process approximation for large computer experiments. *Journal of Computational and Graphical Statistics*, 24(2):561–578, 2015. doi:10.1080/10618600.2014.914442.
- Rajarshi Guhaniyogi and Sudipto Banerjee. Meta-kriging: Scalable Bayesian modeling and inference for massive spatial datasets. *Technometrics*, 60(4):430–444, 2018. doi:10.1080/00401706.2018.1437474.
- Heikki Haario, Eero Saksman, and Johanna Tamminen. An adaptive Metropolis algorithm. *Bernoulli*, 7(2):223–242, 2001. doi:10.2307/3318737.
- Matthew J. Heaton, Abhirup Datta, Andrew O. Finley, Reinhard Furrer, Joseph Guinness, Rajarshi Guhaniyogi, Florian Gerber, Robert B. Gramacy, Dorit Hammerling, Matthias Katzfuss, Finn Lindgren, Douglas W. Nychka, Furong Sun, and Andrew Zammit-Mangion. A case study competition among methods for analyzing large spatial data. *Journal of Agricultural, Biological and Environmental Statistics*, 24(3):398–425, Sep 2019. doi:10.1007/s13253-018-00348-w.

- Matthew D. Hoffman and Andrew Gelman. The no-U-turn sampler: Adaptively setting path lengths in Hamiltonian Monte Carlo. *Journal of Machine Learning Research*, 15(47):1593–1623, 2014. <https://www.jmlr.org/papers/v15/hoffman14a.html>.
- Bora Jin, Michele Peruzzi, and David B. Dunson. Bag of DAGs: Flexible & scalable modeling of spatiotemporal dependence, 2021. [arXiv:2112.11870](https://arxiv.org/abs/2112.11870).
- Bora Jin, Amy H. Herring, and David B. Dunson. Spatial predictions on physically constrained domains: Applications to arctic sea salinity data, 2022. [arXiv:2210.03913](https://arxiv.org/abs/2210.03913).
- James E. Johndrow, Natesh S. Pillai, and Aaron Smith. No free lunch for approximate MCMC, 2020. [arXiv:2010.125147](https://arxiv.org/abs/2010.125147).
- A. M. Jönsson, L. Eklundh, M. Hellström, L. Bärring, and P. Jönsson. Annual changes in MODIS vegetation indices of Swedish coniferous forests in relation to snow dynamics and tree phenology. *Remote Sensing of Environment*, 114:2719–2730, 2010. doi:10.1016/j.rse.2010.06.005.
- Marcin Jurek and Matthias Katzfuss. Hierarchical sparse Cholesky decomposition with applications to high-dimensional spatio-temporal filtering, 2020. [arXiv:2006.16901](https://arxiv.org/abs/2006.16901).
- Matthias Katzfuss. A multi-resolution approximation for massive spatial datasets. *Journal of the American Statistical Association*, 112:201–214, 2017. doi:10.1080/01621459.2015.1123632.
- Matthias Katzfuss and Joseph Guinness. A general framework for Vecchia approximations of Gaussian processes. *Statistical Science*, 36(1):124–141, 2021. doi:10.1214/19-STS755.
- Cari G. Kaufman, Mark J. Schervish, and Douglas W. Nychka. Covariance tapering for likelihood-based estimation in large spatial data sets. *Journal of the American Statistical Association*, 103:1545–1555, 2008. doi:10.1198/016214508000000959.
- Finn Lindgren, Håvard Rue, and Johan Lindström. An explicit link between Gaussian fields and Gaussian Markov random fields: the stochastic partial differential equation approach. *Journal of the Royal Statistical Society: Series B*, 73:423–498, 2011. doi:10.1111/j.1467-9868.2011.00777.x.
- Tristan Marshall and Gareth Roberts. An adaptive approach to Langevin MCMC. *Statistics and Computing*, 22:1041–1057, 2012. doi:10.1007/s11222-011-9276-6.
- G. Matheron. Pour une analyse krigéante des données régionalisées. *Technical report N.732, Centre de Géostatistique*, 1982.
- Diego Mesquita, Paul Blomstedt, and Samuel Kaski. Embarrassingly parallel MCMC using deep invertible transformations. In Ryan P. Adams and Vibhav Gogate, editors, *Proceedings of Machine Learning Research*, volume 115, pages 1244–1252, Tel Aviv, Israel, 22–25 Jul 2020. PMLR. <http://proceedings.mlr.press/v115/mesquita20a.html>.
- Niamh Mimmagh, Andrew Parnell, Estevão Prado, and Rafael de Andrade Moral. Bayesian multi-species N-mixture models for unmarked animal communities. *Environmental and Ecological Statistics*, 29:755–778, 2022. doi:10.1007/s10651-022-00542-7.
- Iain Murray, Ryan Adams, and David MacKay. Elliptical slice sampling. In Yee Whye Teh and Mike Titterton, editors, *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, volume 9 of *Proceedings of Machine Learning Research*, pages 541–548, Chia Laguna Resort, Sardinia, Italy, 13–15 May 2010. PMLR. <https://proceedings.mlr.press/v9/murray10a.html>.

- R. M. Neal. MCMC using Hamiltonian dynamics. In S. Brooks, A. Gelman, G. L. Jones, and X.-L. Meng, editors, *Handbook of Markov Chain Monte Carlo*. CRC Press, New York, 2011. doi:10.1201/b10905.
- Willie Neiswanger, Chong Wang, and Eric P. Xing. Asymptotically exact, embarrassingly parallel MCMC. In *Proceedings of the Thirtieth Conference on Uncertainty in Artificial Intelligence, UAI'14*, page 623–632, Arlington, Virginia, USA, 2014. AUAI Press. ISBN 9780974903910.
- Christopher Nemeth and Chris Sherlock. Merging MCMC subposteriors through Gaussian-process approximations. *Bayesian Analysis*, 13(2):507–530, 06 2018. doi:10.1214/17-BA1063.
- Michele Peruzzi and David B. Dunson. Spatial multivariate trees for big data Bayesian regression. *Journal of Machine Learning Research*, 23(17):1–40, 2022. <http://jmlr.org/papers/v23/20-1361.html>.
- Michele Peruzzi, Sudipto Banerjee, David B. Dunson, and Andrew O. Finley. Grid-Parametrize-Split (GriPS) for improved scalable inference in spatial big data analysis, 2021. arXiv:2101.03579.
- Michele Peruzzi, Sudipto Banerjee, and Andrew O. Finley. Highly scalable Bayesian geostatistical modeling via meshed Gaussian processes on partitioned domains. *Journal of the American Statistical Association*, 117(538):969–982, 2022. doi:10.1080/01621459.2020.1833889.
- Gareth O. Roberts and Jeffrey S. Rosenthal. Coupling and ergodicity of adaptive Markov chain Monte Carlo algorithms. *Journal of Applied Probability*, 44:458–475, 2007. doi:10.1239/jap/1183667414.
- Gareth O. Roberts and Jeffrey S. Rosenthal. Examples of adaptive MCMC. *Journal of Computational and Graphical Statistics*, 18(2):349–367, 2009. doi:10.1198/jcgs.2009.06134.
- Gareth O. Roberts and Osnat Stramer. Langevin diffusions and Metropolis-Hastings algorithms. *Methodology And Computing In Applied Probability*, 4:337–357, 2002. doi:10.1023/A:1023562417138.
- Gareth O. Roberts and Richard L. Tweedie. Exponential convergence of Langevin distributions and their discrete approximations. *Bernoulli*, 2(4):341–363, 1996.
- J.A. Royle. N-Mixture Models for Estimating Population Size from Spatially Replicated Counts. *Biometrics*, 60(1):108–115, 2004. doi:10.1111/j.0006-341X.2004.00142.x.
- Havard Rue and Leonhard Held. *Gaussian Markov Random Fields: Theory and Applications*. Chapman & Hall/CRC, 2005. doi:10.1007/978-3-642-20192-9.
- Håvard Rue, Sara Martino, and Nicolas Chopin. Approximate Bayesian inference for latent Gaussian models by using integrated nested laplace approximations. *Journal of the Royal Statistical Society: Series B*, 71:319–392, 2009. doi:10.1111/j.1467-9868.2008.00700.x.
- Huiyan Sang and Jianhua Z. Huang. A full scale approximation of covariance functions for large spatial data sets. *Journal of the Royal Statistical Society, Series B*, 74:111–132, 2012. doi:10.1111/j.1467-9868.2011.01007.x.
- Alexandra M. Schmidt and Alan E. Gelfand. A Bayesian coregionalization approach for multivariate pollutant data. *Journal of Geophysical Research*, 108:D24, 2003. doi:10.1029/2002JD002905.
- Aritra Sengupta and Noel Cressie. Hierarchical statistical modeling of big spatial datasets using the exponential family of distributions. *Spatial Statistics*, 2013. doi:10.1016/j.spasta.2013.02.002.

- Amar Shah, Andrew G. Wilson, and Zoubin Ghahramani. Student-t processes as alternatives to Gaussian processes. In *Proceedings of the 17th International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2014.
- Shinichiro Shirota, Andrew O. Finley, Bruce D. Cook, and Sudipto Banerjee. Conjugate nearest neighbor Gaussian process models for efficient statistical interpolation of large spatial data, 2019. [arXiv:1907.10109](https://arxiv.org/abs/1907.10109).
- Michael L. Stein. The screening effect in kriging. *The Annals of Statistics*, 30(1):298–323, 2002. doi:10.1214/aos/1015362194.
- Michael L. Stein. Limitations on low rank approximations for covariance matrices of spatial data. *Spatial Statistics*, 8:1–19, 2014. doi:doi:10.1016/j.spasta.2013.06.003.
- Michael L. Stein, Zhiyi Chi, and Leah J. Welty. Approximating likelihoods for large spatial data sets. *Journal of the Royal Statistical Society, Series B*, 66:275–296, 2004. doi:10.1046/j.1369-7412.2003.05512.x.
- Y. Sun, B. Li, and M. Genton. Geostatistics for large datasets. In J. Montero, E. Porcu, and M. Schlather, editors, *Advances and Challenges in Space-time Modelling of Natural Events*, pages 55–77. Springer-Verlag, Berlin Heidelberg, 2011. doi:10.1007/978-3-642-17086-7.
- Benjamin M. Taylor and Peter J. Diggle. INLA or MCMC? a tutorial and comparative evaluation for spatial prediction in log-Gaussian Cox processes. *Journal of Statistical Computation and Simulation*, 84(10):2266–2284, 2014. doi:10.1080/00949655.2013.788653.
- Gleb Tikhonov, Oystein H. Opedal, Nerea Abrego, Aleksi Lehikoinen, Melinda M. J. de Jonge, Jari Oksanen, and Otso Ovaskainen. Joint species distribution modelling with the R-package Hmsc. *Methods in Ecology and Evolution*, 11(3):442–447, 2020. doi:10.1111/2041-210X.13345.
- A. V. Vecchia. Estimation and model identification for continuous spatial processes. *Journal of the Royal Statistical Society, Series B*, 50:297–312, 1988. doi:10.1111/j.2517-6161.1988.tb01729.x.
- Matti Vihola. Robust adaptive Metropolis algorithm with coerced acceptance rate. *Statistics and Computing*, 22:997–1008, 2012. doi:10.1007/s11222-011-9269-5.
- Hans Wackernagel. *Multivariate Geostatistics: An Introduction with Applications*. Springer, Berlin, 2003. doi:10.1007/978-3-662-05294-5.
- D. A. Walker, James C. Halfpenny, Marilyn D. Walker, and Carol A. Wessman. Long-term studies of snow-vegetation interactions. *BioScience*, 43(5):287–301, 1993. doi:10.2307/1312061.
- Kun Wang, Li Zhang, Yubao Qiu, Lei Ji, Feng Tian, Cuizhen Wang, and Zhiyong Wang. Snow effects on alpine vegetation in the Qinghai-Tibetan plateau. *International Journal of Digital Earth*, 8(1):58–75, 2015a. doi:10.1080/17538947.2013.848946.
- Xiangyu Wang and David B. Dunson. Parallelizing MCMC via Weierstrass sampler, 2014. [arXiv:1312.4605](https://arxiv.org/abs/1312.4605).
- Xiangyu Wang, Fangjian Guo, Katherine A. Heller, and David B. Dunson. Parallelizing MCMC with random partition trees. In *Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 1, NIPS’15*, page 451–459, Cambridge, MA, USA, 2015b. MIT Press. [arXiv:1506.03164](https://arxiv.org/abs/1506.03164).

- Jing Xie, Tobias Jonas, Christian Rixen, Rogier de Jong, Irene Garonna, Claudia Notarnicola, Sarah Asam, Michael E. Schaepman, and Mathias Kneubühler. Land surface phenology and greenness in Alpine grasslands driven by seasonal snow and meteorological factors. *Science of The Total Environment*, 725:138380, 2020. doi:10.1016/j.scitotenv.2020.138380.
- Giacomo Zanella and Gareth Roberts. Multilevel linear models, gibbs samplers and multigrid decompositions. *Bayesian Analysis*, 2021. doi:10.1214/20-BA1242.
- Lu Zhang and Sudipto Banerjee. Spatial factor modeling: A Bayesian matrix-normal approach for misaligned data. *Biometrics*, 78(2):560–573, 2022. doi:10.1111/biom.13452.
- Yichen Zhu, Michele Peruzzi, Cheng Li, and David B. Dunson. Radial neighbors for provably accurate scalable approximations of Gaussian processes, 2022. arXiv:2211.14692.
- Daniel Zilber and Matthias Katzfuss. Vecchia-Laplace approximations of generalized Gaussian processes for big non-Gaussian spatial data, 2020. arXiv:1906.07828.