

SPATIOTEMPORAL AUTOREGRESSIVE PARTIALLY LINEAR VARYING COEFFICIENT MODELS

Shan Yu¹, Yueying Wang², Li Wang³ and Lei Gao³

¹*University of Virginia*, ²*Columbia University*
and ³*George Mason University*

Abstract: With increasingly abundant data that relate to both space and time becoming available, spatiotemporal modeling is receiving much attention in the literature. This paper study develops a class of spatiotemporal autoregressive partially linear varying-coefficient models that are sufficiently flexible to simultaneously capture the spatiotemporal dependence and nonstationarity often encountered in practice. When spatial observations are observed over time and exhibit dynamic and nonstationary behaviors, our models become particularly useful. We develop a numerically stable and computationally efficient estimation procedure, using the tensor-product splines over triangular prisms to approximate the coefficient functions. The estimators of both the constant coefficients and the varying coefficients are consistent. We also show that the estimators of the constant coefficients are asymptotically normal, which enables us to construct confidence intervals and make inferences. The method's performance is evaluated using Monte Carlo experiments, and applied to model and forecast the spread of COVID-19 at the county level in the United States.

Key words and phrases: Partially linear models, penalized splines, semiparametric regression, spatiotemporal dependence, triangular prismatic partitions, varying coefficient models.

1. Introduction

The wide availability of data observed over time and space has stimulated studies in a variety of disciplines, such as economics, environmental science, epidemiology, and many areas of health studies. At the same time, spatiotemporal data are generated at scales and levels of complexity far beyond what could have been imagined previously. For example, there are many large-scale economic studies based on panels of data collected at the census tract, city, or county level with an implicit, but complex spatial structure. The observations in data can be regularly or irregularly distributed in space or time. Complex data call for

Corresponding author: Li Wang, Department of Statistics, George Mason University, Fairfax, VA 22030, USA. E-mail: lwang41@gmu.edu.

statistical models that are sufficiently flexible to adapt to underlying signals, allowing scientists to discover unknown patterns and predict the time evolution of the variables of interest. Although spatiotemporal models have gained in popularity in recent years, enhancing the capability of spatiotemporal modeling and analysis remains a long-standing challenge.

In the statistical literature, there is a long history of using spatial data in regression analyses to investigate covariate effects on response variables in the presence of spatial correlation. The literature on spatial data modeling is overwhelmingly dominated by “global” regression models, which intrinsically assume that relationships between the regressors and the regressand are homogeneous (stationary) over space and/or time. Classical spatial autoregressive (SAR) models, which fit linear models with autoregressive errors and spatial lags of the dependent and independent variables, have attracted much attention in spatial data analysis and its application in many fields. For instance, Kelejian and Prucha (2010) developed a method of inference for SARs that allows for the possibility of heteroskedasticity. Lee (2004) investigated the asymptotic properties of the quasi-maximum likelihood estimator. Pace et al. (1998) and Lee and Yu (2010) explored SAR models under a spatiotemporal framework. Xu, Wang and Shin (2020) consider a dynamic spatial autoregressive quantile model using predetermined network information.

However, in practice, many data exhibit clearly heterogeneous and nonstationary features; see the discussions in Cressie and Wikle (2011), Fotheringham, Brunson and Charlton (2002), and Zhang and Wang (2015). The assumption of stationarity or structural stability over time and space is generally unrealistic, because the processes tend to vary over the study area and time period. For instance, in real estate applications, spatially varying-coefficient models are useful for capturing the spatial heterogeneity in housing prices and accounting for local features (Helbich and Griffith (2016)). Neglecting these features might have serious consequences for model estimation, such as biased regression coefficients, resulting in inappropriate conclusions (LeSage (2008)). To incorporate nonstationarity in the regression models, Hoover et al. (1998), Cai (2007), and Chen, Li and Li (2015) studied time-varying-coefficient models (TVCM) with correlated errors. Fotheringham, Brunson and Charlton (2002) and Gelfand et al. (2003) introduced spatially varying-coefficient models (SVCMs) to explore the spatial nonstationarity of a regression relationship. In this study, we illustrate how spatiotemporal dependence and nonstationarity can be modeled simultaneously in a regression analysis framework.

Suppose there are n space-time observations $\mathcal{A}_n = \{(\mathbf{S}_1, T_1), \dots, (\mathbf{S}_n, T_n)\}$,

where $T_i \in \mathcal{T}$ and $\mathbf{S}_i \equiv (S_{i1}, S_{i2})^\top \in \Omega$ are the time and the spatial location, respectively, of the i th observation. Let Y_i be the response variable, and let $\mathbf{Z}_i = (1, Z_{i1}, \dots, Z_{ip_1})^\top$ and $\mathbf{X}_i = (X_{i1}, \dots, X_{ip_2})^\top$ be the explanatory variables. We consider the following spatiotemporal autoregressive partially linear varying-coefficient model (STAR-PLVCM):

$$Y_i = \alpha_0 \sum_{j=1}^n w_{ij} Y_j + \sum_{\ell=0}^{p_1} Z_{i\ell} \eta_{0\ell} + \sum_{k=1}^{p_2} X_{ik} \beta_{0k}(S_{i1}, S_{i2}, T_i) + \epsilon_i, \quad (1.1)$$

where α_0 is a global parameter, w_{ij} is the weight of the neighbor effects, satisfying $w_{ii} = 0$ and $\sum_{j \neq i} w_{ij} = 1$, for any $i = 1, \dots, n$, $\eta_{0\ell}$ are unknown coefficient parameters, $\beta_{0k}(\cdot, \cdot, \cdot)$ are unknown varying-coefficient functions, and ϵ_i 's are independent and identically distributed (i.i.d) random noises, with $E(\epsilon_i) = 0$ and $\text{Var}(\epsilon_i) = \sigma_0^2$, and ϵ_i is independent of \mathbf{Z}_i and \mathbf{X}_i . In the rest of this paper, we denote $\mathbf{W} = (w_{ij})$ as the $n \times n$ weight matrix. The STAR-PLVCM accounts for both spatiotemporal nonstationarity and autocorrelation simultaneously. In addition, it offers greater flexibility in assessing varying effects at different times and locations than do current global models in the literature. At the same time, it preserves the simplicity and efficiency when some of the coefficients are indeed constants.

The STAR-PLVCM encompasses many existing models as special cases, such as the spatiotemporal autoregressive (STAR) model, when all β_{0k} are assumed to be constant (Pace et al. (1998)); the binary treatment model with spatial interactions, when X_{ik} consists of a constant term only; the semiparametric SAR model, when X_{ik} consists of a constant term only and its coefficient effect is assumed to be spatially dependent only (Su and Jin (2010)); the partially linear varying-coefficient model (Li and Liang (2008)), when there is no neighbor effect in the model, that is, $\alpha_0 = 0$; the TVCM (Fan and Zhang (2008); Park et al. (2015); Yang et al. (2006)), when only the time index is included in the coefficient functions; and the SVCML in Fotheringham, Brunson and Charlton (2002) Gelfand et al. (2003), and Mu, Wang and Wang (2018), when only the spatial index is included and neighbor effects are not considered.

The coefficient estimators play an important role in reflecting the spatiotemporal nonstationarity of the regression relationship and, thus, largely determine the analysis results. Huang, Wu and Barry (2010) and Fotheringham, Crespo and Yao (2015) developed the geographically and temporally weighted regression (GTWR) method to deal with both spatial and temporal nonstationarity simultaneously by incorporating the temporal effects into the standard SVCML.

Wu, Li and Huang (2014) proposed a geographically and temporally weighted autoregressive (GTWAR) model to further account for correlation among the observations. The GTWAR assumes that all coefficients are spatially varying. However, in reality, some covariates may have homogenous effects, while others have heterogeneous effects across locations. The STAR-PLVCM is a parsimonious special case of the GTWAR.

Estimating the STAR-PLVCM is challenging. There are suitable methods for spatiotemporal time modeling, such as the kriging or kernel smoothing methods (Müller, Stadtmüller and Tabnak (1997)), when sufficient information is available in both dimensions and the data are regularly distributed over a rectangular domain. However, in many cases, the observations can be dense at some locations or time intervals, while sparse at others, and the shape of the domain may not be regular or show gaps and holes; see Sangalli, Ramsay and Ramsay (2013) and Wood, Bravington and Hedley (2008), for example. As pointed out in Wang and Ranalli (2007) and Ramsay (2002), many traditional smoothing tools, such as kriging and kernel smoothing, perform badly when used to smooth data over such complex domains, because they smooth inappropriately across the boundary features (referred to as the “leakage” problem in the literature). Thus, we propose using penalized *tensor product splines* over *triangular prismatic partitions* (TPST), which are the tensor products of bivariate splines and univariate splines, to overcome these challenges. We prefer the TPST, owing to their (i) computational efficiency, (ii) ability to handle sparse designs, and (iii) convenient representations with flexible degrees and various smoothness. To estimate the proposed model in (1.1), we use the profile maximum likelihood (ML) method, which is a popular method for (semi)parametric SAR models. Under some regularity conditions, we obtain the asymptotically normal distribution of the estimators of the constant coefficients in the linear part, and derive the convergence rates of the estimators of the varying-coefficient functions.

The rest of the paper is organized as follows. In Section 2, we describe our model, briefly review univariate splines and bivariate splines over triangulations, and introduce the penalized estimation method. Section 3 provides the asymptotic properties of the estimators of the linear coefficients and the coefficient functions. Section 4 discusses how to implement the proposed methodology in practice. In Section 5, we conduct simulation studies to evaluate the finite-sample performance of the proposed method. In Section 6, we apply our method to model and forecast COVID-19 infection counts and death counts in all counties in the United States. Concluding remarks are given in Section 7. Proofs of the main results are deferred to Sections A, B, and C in the Supplementary Material.

Additional simulation results and a description of related COVID-19 data set are given in Section D of the Supplementary Material.

2. Methodology

Denote n -dimensional vectors $\mathbf{Y} = (Y_1, \dots, Y_n)^\top$, $\boldsymbol{\epsilon} = (\epsilon_1, \dots, \epsilon_n)^\top$, and $\boldsymbol{\mu}_0 = (\mu_{01}, \dots, \mu_{0n})^\top$, where $\mu_{0i} = \mathbf{Z}_i^\top \boldsymbol{\eta}_0 + \mathbf{X}_i^\top \boldsymbol{\beta}_0(S_{i1}, S_{i2}, T_i)$, $\boldsymbol{\eta}_0 = (\eta_{00}, \eta_{01}, \dots, \eta_{0p_1})^\top$, and $\boldsymbol{\beta}_0 = (\beta_{01}, \dots, \beta_{0p_2})^\top$. Let $\mathbf{W} = (w_{ij})$ be an $n \times n$ weight matrix. Model (1.1) can be written in the following matrix form: $\mathbf{Y} = \alpha_0 \mathbf{W} \mathbf{Y} + \boldsymbol{\mu}_0 + \boldsymbol{\epsilon}$. For any value of α , denote $\boldsymbol{\Xi}(\alpha) = \mathbf{I}_n - \alpha \mathbf{W}$. Then the equilibrium vector \mathbf{Y} is $\mathbf{Y} = \{\boldsymbol{\Xi}(\alpha_0)\}^{-1}(\boldsymbol{\mu}_0 + \boldsymbol{\epsilon})$. Denote $\mathbf{Y}(\alpha) = \boldsymbol{\Xi}(\alpha) \mathbf{Y}$. For simplicity of notation, we denote $\boldsymbol{\Xi}_0 = \boldsymbol{\Xi}(\alpha_0)$ in the rest of our paper. If the noise term $\boldsymbol{\epsilon}$ is assumed to follow a Gaussian distribution with mean zero and variance $\sigma^2 \mathbf{I}_n$, then the log-likelihood function is

$$L_n(\alpha, \boldsymbol{\eta}, \boldsymbol{\beta}, \sigma^2) = -\frac{n}{2} \log(2\pi) - \frac{n}{2} \log(\sigma^2) + \log(|\boldsymbol{\Xi}(\alpha)|) - \frac{1}{2\sigma^2} \{\mathbf{Y}(\alpha) - \boldsymbol{\mu}\}^\top \{\mathbf{Y}(\alpha) - \boldsymbol{\mu}\}, \quad (2.1)$$

where $\boldsymbol{\mu} = (\mu_1, \dots, \mu_n)^\top$, with $\mu_i = \mathbf{Z}_i^\top \boldsymbol{\eta} + \mathbf{X}_i^\top \boldsymbol{\beta}(S_{i1}, S_{i2}, T_i)$. We propose using the profile log-likelihood approach to estimation the model. For each fixed α , we maximize (2.1) with respect to $\boldsymbol{\eta}$ and $\boldsymbol{\beta}(s_1, s_2, t)$ to obtain the estimators $\hat{\boldsymbol{\eta}}(\alpha)$ and $\hat{\boldsymbol{\beta}}(s_1, s_2, t; \alpha)$, respectively, which are functions of α . Next, to estimate α , we plug $\hat{\boldsymbol{\eta}}(\alpha)$ and $\hat{\boldsymbol{\beta}}(\cdot, \cdot, \cdot; \alpha)$ into (2.1), and obtain the estimators $\hat{\alpha}$ by maximizing (2.1) with respect to α .

2.1. Tensor-product splines over triangular prismatic partition

For the estimation of the coefficient functions $\beta_k(\cdot)$, we assume $\beta_k(\cdot)$ are defined over a 3D domain $\Omega \times \mathcal{T}$, where Ω is a polygon on the spatial plane, \mathcal{T} is an interval on the time dimension, and without loss of generality, we assume $\mathcal{T} = [t_1, t_2]$ throughout the paper. We propose approximating $\beta_k(\cdot)$ using the tensor-product basis of bivariate splines and univariate splines over triangular prismatic partitions, detailed below.

2.2. Triangular prismatic partitions

Over the time domain $[t_1, t_2]$, suppose there are N points $\{\pi_1, \dots, \pi_N\}$ satisfying $t_1 = \pi_0 < \pi_1 < \dots < \pi_N < \pi_{N+1} = t_2$. Let $I_b = [\pi_{b-1}, \pi_b)$, for $b = 1, \dots, N+1$. Then, $\{I_1, \dots, I_{N+1}\}$ is a partition of $[t_1, t_2]$. For the spatial dimension, we consider triangulation of a polygonal domain Ω , which is an effective

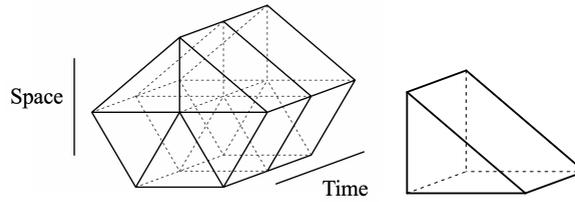


Figure 1. An example of a triangular prismatic partition.

tool to handle data distributed on irregular 2D regions with complex boundaries and/or interior holes. Lindgren and Rue (2015), Mu, Wang and Wang (2018), and Yu et al. (2020) use triangulation to partition the spatial domain into triangles. In the following, we use τ to denote a triangle, which is a convex hull of three points not located in one line. A collection $\{\tau_1, \dots, \tau_K\}$ of K triangles is called a triangulation of $\Omega = \cup_{a=1}^K \tau_a$, provided that if a pair of triangles in Δ intersect, then their intersection is either a common vertex or a common edge. Denote the size of Δ by $|\Delta|$, which is the length of the longest edge of Δ .

Note that, given triangle $\tau_a \in \Delta$ and interval $I_b \in \mathcal{T}$, $e_{a,b} = \tau_a \times I_b$ is a triangular prism element. By triangular prism, we mean a prism with two parallel triangular faces and three rectangular faces. The domain $\Omega \times \mathcal{T}$ can thus be subdivided into the union of non-overlapping shape-regular triangular prism elements, such that the nonempty intersection of any distinct pair of elements is a single common vertex, edge, or face. In the following, let $\mathcal{E} = \{e_{a,b}, 1 \leq a \leq K, 1 \leq b \leq N+1\}$ be a face-to-face partition of the polyhedron $\Omega \times \mathcal{T}$ into prisms; see Figure 1 for an example.

2.3. Tensor-product splines

We define the tensor-product splines over a triangular prismatic partition.

We first introduce the univariate splines over the time domain, which is a piecewise polynomial smoothly connected at its knots. The interior points $\{\pi_b\}_{b=1}^N$ defined above can serve as the knots. For a fixed integer ϱ , let $\boldsymbol{\pi}$ be a knots vector of $[t_1, t_2]$ with N interior knots that satisfies $\boldsymbol{\pi} = \{t_1 = \pi_{1-\varrho} = \dots = \pi_0 < \pi_1 < \dots < \pi_N < \pi_{N+1} = \dots = \pi_{N+\varrho} = t_2\}$. The polynomial splines of order ϱ are polynomial functions with $(\varrho - 1)$ -degree (or less) on subintervals $[\pi_b, \pi_{b+1}]$, for $b = 0, \dots, N-1$, and $[\pi_N, \pi_{N+1}]$, and have $\varrho - 2$ continuous derivatives globally. Let $\mathcal{U}_\varrho(\boldsymbol{\pi})$ stand for the space of such polynomial splines. A basis of $\mathcal{U}_\varrho(\boldsymbol{\pi})$ can be formed as B-splines, which are denoted as $\mathbf{U}(t) = \{U_1(t), \dots, U_{N+\varrho}(t)\}^\top$.

Next, we introduce the bivariate splines over a triangulation Δ . For a nonnegative integer r , let $\mathbb{C}^r(\Omega)$ be the collection of all r th continuously differentiable

functions over Ω . Given a triangulation Δ , let $\mathbb{S}_d^r(\Delta) = \{\zeta \in \mathbb{C}^r(\Omega) : \zeta|_\tau \in \mathbb{P}_d(\tau), \tau \in \Delta\}$ be a spline space of degree d and smoothness r over triangulation Δ , where $\zeta|_\tau$ is the polynomial piece of spline ζ restricted on triangle τ , and \mathbb{P}_d is the space of all polynomials of degree less than or equal to d . Let $\{B_m\}_{m \in \mathcal{M}}$ be the set of Bernstein basis polynomials for $\mathbb{S}_d^r(\Delta)$ constructed in Lai and Schumaker (2007), where \mathcal{M} is the index set of the Bernstein basis functions, and the cardinality of \mathcal{M} , $|\mathcal{M}|$, is $K(d+1)(d+2)/2$. Any function $\zeta(\mathbf{s}) \in \mathbb{S}_d^r(\Delta)$, $\mathbf{s} \in \Omega$, can be expressed as $\zeta(\mathbf{s}) = \sum_{m \in \mathcal{M}} B_m(\mathbf{s})\vartheta_m = \mathbf{B}(\mathbf{s})^\top \boldsymbol{\vartheta}$, subject to $\mathbf{H}\boldsymbol{\vartheta} = \mathbf{0}$, where \mathbf{H} is the matrix that collects the smoothness conditions across all shared edges of triangles, referred to as the constraint matrix. See Yu et al. (2020) for an example of \mathbf{H} . Denote the space of tensor-product splines over the triangular prismatic partition \mathcal{E} by $\mathbb{T}^{(\varrho, d, r)}(\mathcal{E}) \equiv \{\sum_{q=1}^{N+\varrho} \sum_{m \in \mathcal{M}} c_{q,m} U_q(t) B_m(\mathbf{s}); \mathbf{H}\mathbf{c}_q = \mathbf{0} \text{ for } \mathbf{c}_q = (c_{q,m}, m \in \mathcal{M})^\top\}$. Let $\{\psi_j(s_1, s_2, t)\}_{j \in \mathcal{J}} = \{U_1(t)B_1(\mathbf{s}), U_1(t)B_2(\mathbf{s}), \dots, U_{N+\varrho}(t)B_{|\mathcal{M}|}(\mathbf{s})\}$ be the tensor-product spline basis functions, where \mathcal{J} is the index set of the tensor-product spline basis and $|\mathcal{J}| = (N + \varrho)|\mathcal{M}|$. Then, any function $\phi(s_1, s_2, t) \in \mathbb{T}^{(\varrho, d, r)}(\mathcal{E})$ can be expressed as

$$\phi(s_1, s_2, t) = \sum_{j \in \mathcal{J}} \psi_j(s_1, s_2, t)\gamma_j = \boldsymbol{\psi}(s_1, s_2, t)^\top \boldsymbol{\gamma}, \quad \text{subject to } \mathcal{H}\boldsymbol{\gamma} = \mathbf{0}, \quad (2.2)$$

where $\boldsymbol{\gamma}$ is the spline coefficient vector, and $\mathcal{H} = \mathbf{I}_{N+\varrho} \otimes \mathbf{H}$ is the matrix that collects the smoothness conditions across all the shared faces of triangular prisms. The above basis can be constructed easily using via the R package TPST (Yu and Wang (2020)).

2.4. Penalized tensor-product spline estimator

To balance the goodness-of-fit and smoothness, we consider the tensor-product spline approximation with a smoothness penalty. Let $\lambda_{k,1}, \lambda_{k,2} \geq 0$ be the penalty parameter, for β_k , $k = 1, \dots, p_2$. Suppose, for now, α is known. Given $\{(\mathbf{S}_i, T_i, \mathbf{Z}_i, \mathbf{X}_i, Y_i(\alpha))\}_{i=1}^n$, we consider the following regularized minimization problem:

$$\begin{aligned} & \min_{\substack{\eta_\ell \in \mathcal{R}, \ell=0, \dots, p_1 \\ \beta_k \in \mathbb{T}^{(\varrho, d, r)}(\mathcal{E}), k=1, \dots, p_2}} \sum_{i=1}^n \left\{ Y_i(\alpha) - \sum_{\ell=0}^{p_1} Z_{i\ell} \eta_\ell - \sum_{k=1}^{p_2} X_{ik} \beta_k(\mathbf{S}_i, T_i) \right\}^2 \\ & + \sum_{m=1}^2 \sum_{k=1}^{p_2} \lambda_{k,m} f_m(\beta_k), \end{aligned} \quad (2.3)$$

where, for any trivariate function $\beta(s_1, s_2, t)$, $f_1(\beta) = \int_{\Omega \times \mathcal{T}} \{(\nabla_{s_1}^2 \beta)^2 + (\nabla_{s_2}^2 \beta)^2\} ds_1 ds_2 dt$ and $f_2(\beta) = \int_{\Omega \times \mathcal{T}} (\nabla_t^2 \beta)^2 ds_1 ds_2 dt$, and the tuning parameters $\lambda_{k,m}$ control the smoothness of the fitted coefficient functions. For a tensor-product spline function $\sum_{j \in \mathcal{J}} \psi_j(s_1, s_2, t) \gamma_j$, we have $f_1(\sum_{j \in \mathcal{J}} \psi_j \gamma_j) = \boldsymbol{\gamma}^\top \mathbf{P}_1 \boldsymbol{\gamma} = \boldsymbol{\gamma}^\top \mathbf{M}_U \otimes \mathbf{P}_B \boldsymbol{\gamma}$ and $f_2(\sum_{j \in \mathcal{J}} \psi_j \gamma_j) = \boldsymbol{\gamma}^\top \mathbf{P}_2 \boldsymbol{\gamma} = \boldsymbol{\gamma}^\top \mathbf{P}_U \otimes \mathbf{M}_B \boldsymbol{\gamma}$, where “ \otimes ” denotes the Kronecker product, \mathbf{M}_U and \mathbf{P}_U are $(N + \rho) \times (N + \rho)$ matrices with $(\mathbf{M}_U)_{q,q'} = \int_{\mathcal{T}} U_q(t) U_{q'}(t) dt$ and $(\mathbf{P}_U)_{q,q'} = \int_{\mathcal{T}} \nabla_t^2 U_q(t) \nabla_t^2 U_{q'}(t) dt$, and \mathbf{M}_B and \mathbf{P}_B are $|\mathcal{M}| \times |\mathcal{M}|$ matrices with

$$\begin{aligned}
 (\mathbf{M}_B)_{m,m'} &= \int_{\Omega} B_m(s_1, s_2) B_{m'}(s_1, s_2) ds_1 ds_2, \\
 (\mathbf{P}_B)_{m,m'} &= \int_{\Omega} \{ \nabla_{s_1}^2 B_m(s_1, s_2) \nabla_{s_1}^2 B_{m'}(s_1, s_2) \\
 &\quad + \nabla_{s_2}^2 B_m(s_1, s_2) \nabla_{s_2}^2 B_{m'}(s_1, s_2) \} ds_1 ds_2.
 \end{aligned}$$

We approximate the function $\beta_k(s_1, s_2, t)$ by $\sum_{j=1}^{|\mathcal{J}|} \psi_j(s_1, s_2, t) \gamma_{kj} = \boldsymbol{\psi}(s_1, s_2, t)^\top \boldsymbol{\gamma}_k$, where $\boldsymbol{\gamma}_k = (\gamma_{kj}, j \in \mathcal{J})^\top$ is the spline coefficient vector.

Using the tensor-product spline approximation in (2.2), solving the minimization problem in (2.3) is approximately equivalent to solving the following constrained minimization problem:

$$\begin{aligned}
 &\min_{\substack{\eta_\ell \in \mathcal{R}, \ell=0, \dots, p_1 \\ \boldsymbol{\gamma}_k \in \mathcal{R}^{|\mathcal{J}|}, k=1, \dots, p_2}} \sum_{i=1}^n \left\{ Y_i(\alpha) - \sum_{\ell=0}^{p_1} Z_{i\ell} \eta_\ell - \sum_{k=1}^{p_2} X_{ik} \boldsymbol{\psi}(\mathbf{S}_i, T_i)^\top \boldsymbol{\gamma}_k \right\}^2, \\
 &+ \sum_{k=1}^{p_2} \lambda_{k,1} \boldsymbol{\gamma}_k^\top \mathbf{P}_1 \boldsymbol{\gamma}_k + \sum_{k=1}^{p_2} \lambda_{k,2} \boldsymbol{\gamma}_k^\top \mathbf{P}_2 \boldsymbol{\gamma}_k, \text{ subject to } \mathcal{H} \boldsymbol{\gamma}_k = \mathbf{0}, \tag{2.4}
 \end{aligned}$$

where $\boldsymbol{\gamma}_k = (\gamma_{kj}, j \in \mathcal{J})^\top$ is the spline coefficient vector, for $k = 1, \dots, p_2$. We can remove the constraint using the QR decomposition $\mathcal{H}^\top = (\mathbf{Q}_1 \ \mathbf{Q}_2) \begin{pmatrix} \mathcal{R}_1 \\ \mathbf{0} \end{pmatrix}$, where $(\mathbf{Q}_1 \ \mathbf{Q}_2)$ is an orthogonal matrix, and \mathcal{R}_1 is an upper-triangle matrix. Simple algebra shows that $\mathbf{Q}_2 = \mathbf{I}_{N+\rho} \otimes \mathbf{Q}_2$. We reparametrize using $\boldsymbol{\gamma}_k = \mathbf{Q}_2 \boldsymbol{\theta}_k$, for some $\boldsymbol{\theta}_k$. Then $\mathcal{H} \boldsymbol{\gamma}_k = \mathbf{0}$ holds. Thus, the minimization problem in (2.4) is converted to

$$\begin{aligned}
 &\min_{\substack{\eta_\ell \in \mathcal{R}, \ell=0, \dots, p_1 \\ \boldsymbol{\theta}_k, k=1, \dots, p_2}} \sum_{i=1}^n \left\{ Y_i(\alpha) - \sum_{\ell=0}^{p_1} Z_{i\ell} \eta_\ell - \sum_{k=1}^{p_2} X_{ik} \boldsymbol{\psi}(\mathbf{S}_i, T_i)^\top \mathbf{Q}_2 \boldsymbol{\theta}_k \right\}^2 \\
 &+ \sum_{k=1}^{p_2} \lambda_{k,1} \boldsymbol{\theta}_k^\top \mathbf{Q}_2^\top \mathbf{P}_1 \mathbf{Q}_2 \boldsymbol{\theta}_k + \sum_{k=1}^{p_2} \lambda_{k,2} \boldsymbol{\theta}_k^\top \mathbf{Q}_2^\top \mathbf{P}_2 \mathbf{Q}_2 \boldsymbol{\theta}_k. \tag{2.5}
 \end{aligned}$$

Let $\boldsymbol{\psi}^*(s_1, s_2, t) = \mathcal{Q}_2^\top \boldsymbol{\psi}(s_1, s_2, t)$, and denote

$$\mathbb{X}_{\boldsymbol{\psi}^*}^\top = \{\mathbf{X}_i \otimes \boldsymbol{\psi}^*(\mathbf{S}_i, T_i)\}_{i=1}^n, \quad \mathbb{X}^\top = (\mathbf{X}_1, \dots, \mathbf{X}_n), \quad \mathbb{Z}^\top = (\mathbf{Z}_1, \dots, \mathbf{Z}_n), \quad \mathbb{D} = (\mathbb{Z}, \mathbb{X}_{\boldsymbol{\psi}^*}). \tag{2.6}$$

Let $\boldsymbol{\Lambda}_1 = \text{diag}(\lambda_{1,1}, \lambda_{1,1}, \dots, \lambda_{p_2,1})$, $\boldsymbol{\Lambda}_2 = \text{diag}(\lambda_{1,2}, \lambda_{1,2}, \dots, \lambda_{p_2,2})$, and

$$\mathbb{P}_\Lambda = \begin{pmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \boldsymbol{\Lambda}_1 \otimes (\mathcal{Q}_2^\top \mathbf{P}_1 \mathcal{Q}_2) + \boldsymbol{\Lambda}_2 \otimes (\mathcal{Q}_2^\top \mathbf{P}_2 \mathcal{Q}_2) \end{pmatrix}. \tag{2.7}$$

For any α , denote $\boldsymbol{\xi}(\alpha) = \{\boldsymbol{\eta}^\top(\alpha), \boldsymbol{\theta}_1^\top(\alpha), \dots, \boldsymbol{\theta}_{p_2}^\top(\alpha)\}^\top$. Solving the penalized least squares problem in (2.5) yields the following estimator of $\boldsymbol{\xi}(\alpha)$:

$$\widehat{\boldsymbol{\xi}}_\Lambda(\alpha) = \{\widehat{\boldsymbol{\eta}}_\Lambda^\top(\alpha), \widehat{\boldsymbol{\theta}}_{1,\Lambda}^\top(\alpha), \dots, \widehat{\boldsymbol{\theta}}_{p_2,\Lambda}^\top(\alpha)\}^\top = (\mathbb{D}^\top \mathbb{D} + \mathbb{P}_\Lambda)^{-1} \mathbb{D}^\top \mathbf{Y}(\alpha). \tag{2.8}$$

Therefore, the TPST estimator of $\beta_k(s_1, s_2, t)$ is $\widehat{\beta}_{k,\Lambda}(s_1, s_2, t; \alpha) = \boldsymbol{\psi}(s_1, s_2, t)^\top \widehat{\boldsymbol{\gamma}}_{k,\Lambda}(\alpha)$, where $\widehat{\boldsymbol{\gamma}}_{k,\Lambda}(\alpha) = \mathcal{Q}_2 \widehat{\boldsymbol{\theta}}_{k,\Lambda}(\alpha)$, for $k = 1, \dots, p_2$.

Plugging $\widehat{\boldsymbol{\eta}}_\Lambda^\top(\alpha)$, $\widehat{\beta}_{k,\Lambda}(s_1, s_2, t; \alpha)$, for $k = 1, \dots, p_2$, into (2.1), we now consider the maximization problem for estimating α_0 and σ_0^2 :

$$\begin{aligned} L_n(\sigma^2, \alpha) &= -\frac{n}{2} \log(2\pi) - \frac{n}{2} \log(\sigma^2) + \log(|\boldsymbol{\Xi}(\alpha)|) \\ &\quad - \frac{1}{2\sigma^2} \{\mathbf{Y}(\alpha) - \widehat{\boldsymbol{\mu}}_\Lambda(\alpha)\}^\top \{\mathbf{Y}(\alpha) - \widehat{\boldsymbol{\mu}}_\Lambda(\alpha)\}, \end{aligned}$$

where $\widehat{\boldsymbol{\mu}}_\Lambda(\alpha) = \boldsymbol{\Pi}_{\mathbb{D},\Lambda} \mathbf{Y}(\alpha)$, with $\boldsymbol{\Pi}_{\mathbb{D},\Lambda} = \mathbb{D} (\mathbb{D}^\top \mathbb{D} + \mathbb{P}_\Lambda)^{-1} \mathbb{D}^\top$. Setting the partial derivative of the objective function on σ^2 to zero and solving the equation, we have

$$\begin{aligned} \widehat{\sigma}_\Lambda^2(\alpha) &= \frac{1}{n} \{\mathbf{Y}(\alpha) - \widehat{\boldsymbol{\mu}}_\Lambda(\alpha)\}^\top \{\mathbf{Y}(\alpha) - \widehat{\boldsymbol{\mu}}_\Lambda(\alpha)\} \\ &= \frac{1}{n} \mathbf{Y}(\alpha)^\top (\mathbf{I}_n - \boldsymbol{\Pi}_{\mathbb{D},\Lambda})^\top (\mathbf{I}_n - \boldsymbol{\Pi}_{\mathbb{D},\Lambda}) \mathbf{Y}(\alpha). \end{aligned} \tag{2.9}$$

The concentrated log-likelihood function of α is

$$L_n(\alpha) = -\frac{n}{2} \{ \log(2\pi) + 1 + \log \widehat{\sigma}_\Lambda^2(\alpha) \} + \log(|\boldsymbol{\Xi}(\alpha)|). \tag{2.10}$$

Maximizing the concentrated log-likelihood in (2.10), we obtain the maximum likelihood estimator (MLE) of α_0 , that is, $\widehat{\alpha}_\Lambda = \text{argmax}_{\alpha \in \varpi} \{- (n/2) \log(\widehat{\sigma}_\Lambda^2(\alpha)) + \log(|\boldsymbol{\Xi}(\alpha)|)\}$, where ϖ is a compact parameter space. We consider $\varpi = [-1, 1]$ and, for any $\alpha \in \varpi$, $\boldsymbol{\Xi}(\alpha)^{-1}$ exists, as stated in Assumption (A9) below. Finally, we plug $\widehat{\alpha}_\Lambda$ into (2.8) and (2.9) to obtain estimators of $\boldsymbol{\eta}_0$, $\boldsymbol{\beta}_0$, and σ_0^2 , that is, $\widehat{\boldsymbol{\eta}}_\Lambda = \widehat{\boldsymbol{\eta}}_\Lambda(\widehat{\alpha}_\Lambda)$, $\widehat{\beta}_{k,\Lambda}(s_1, s_2, t) = \widehat{\beta}_{k,\Lambda}(s_1, s_2, t; \widehat{\alpha}_\Lambda)$, and $\widehat{\sigma}_\Lambda^2 = \widehat{\sigma}_\Lambda^2(\widehat{\alpha}_\Lambda)$, respec-

tively. In the rest of the paper, we suppress “ Λ ” in the estimators for notational simplicity.

3. Asymptotic Properties

3.1. Properties of tensor-product splines over a triangular prismatic partition

In this section, we first study the properties of tensor-product splines over a triangular prismatic partition, which are necessary to establish the asymptotics of the TPST estimators. We now state some regularity conditions on the partition of the domain.

For any function f on a domain \mathcal{D} , $\mathcal{D} = \mathcal{T}, \Omega, \Omega \times \mathcal{T}$, denote $\|f\|_{\infty, \mathcal{D}} = \sup_{\mathbf{x} \in \mathcal{D}} |f(\mathbf{x})|$ as the supremum norm of the function f over \mathcal{D} . For any L_2 -integrable functions $f_1(\mathbf{x})$ and $f_2(\mathbf{x})$, for $\mathbf{x} \in \mathcal{D}$, denote the L_2 inner product and the induced norm by $\langle f_1, f_2 \rangle_{L_2, \mathcal{D}} = \int_{\mathbf{x} \in \mathcal{D}} f_1(\mathbf{x}) f_2(\mathbf{x}) d\mathbf{x}$ and $\|f_1\|_{L_2, \mathcal{D}}^2 = \langle f_1, f_1 \rangle_{L_2, \mathcal{D}}$, respectively. For any function $g(\mathbf{x})$, for $\mathbf{x} \in \mathcal{D}$, for any direction x_q , let $\nabla_{x_q}^v g(\mathbf{x})$ be the v th-order derivative in the direction x_q at the point \mathbf{x} . For any nonnegative integer p , let $\mathbb{C}^p(\mathcal{T})$ be the functional space consisting of all univariate functions whose p th-order derivatives exist, and is continuous on \mathcal{T} . For any nonnegative integer ι , let $\mathbb{W}^{\iota, \infty}(\Omega) = \{f : |f|_{k, \infty, \Omega} < \infty, 0 \leq k \leq \iota\}$ be the standard Sobolev space of bivariate functions on the domain Ω , where $|f|_{v, \infty, \Omega} = \max_{i+j=v} \|\nabla_{s_1}^i \nabla_{s_2}^j f(s_1, s_2)\|_{\infty, \Omega}$. Denote by $\mathbb{W}^{\iota, \infty}(\Omega) \otimes \mathbb{C}^p(\mathcal{T})$ the functional space defined on $\Omega \times \mathcal{T}$. If function $g(s_1, s_2, t) \in \mathbb{W}^{\iota, \infty}(\Omega) \otimes \mathbb{C}^p(\mathcal{T})$, it satisfies the following: (i) for $0 \leq a_1 + a_2 \leq \iota - 1$ and any given $(s_1, s_2) \in \Omega$, $\nabla_{s_1}^{a_1} \nabla_{s_2}^{a_2} g(s_1, s_2, t) \in C^p(\mathcal{T})$; and (ii) for $0 \leq a_3 \leq p$ and any given $t \in \mathcal{T}$, $\nabla_t^{a_3} g(s_1, s_2, t) \in \mathbb{W}^{\iota, \infty}(\Omega)$.

Note that under Assumptions (A1) and (A2), h and $|\Delta|$ reflect the number of basis functions of the univariate component and the bivariate component, respectively, that is, $N \asymp h^{-1}$ and $|\mathcal{M}| \asymp |\Delta|^{-2}$. Lemma 1 illustrates how the numbers of univariate spline basis functions and bivariate spline basis functions affect the approximation power of the tensor-product splines.

Lemma 1. *Under Assumptions (A1) and (A2) in the Supplementary Material, for any function $g(s_1, s_2, t) \in \mathbb{W}^{d+1, \infty}(\Omega) \otimes \mathbb{C}^{\varrho-2}(\mathcal{T})$, there exists a spline function $g^* \in \mathbb{T}^{(\varrho, d, r)}(\mathcal{E})$ such that for any $0 \leq a_1 + a_2 \leq d$, $0 \leq a_3 \leq \varrho$, $\|\nabla_{s_1}^{a_1} \nabla_{s_2}^{a_2} \nabla_t^{a_3} (g - g^*)\|_{\infty, \Omega \times \mathcal{T}} = O(h^{\varrho - a_3} + |\Delta|^{d+1 - a_1 - a_2})$.*

To work with splines, we need to choose a basis to represent the functions. If a basis is suitable for numerical computations, functions with “small” function

values should have “small” coefficients of the basis functions, and vice versa. A basis with this property is said to be *stable*. Lemma 2 shows that the tensor-product spline basis of spline space $\mathbb{T}^{(e,d,r)}(\mathcal{E})$ exhibits this stability property; that is, small perturbations of the spline coefficients can only lead to small perturbations in the spline function.

Lemma 2. *Under Assumptions (A1)–(A2) in Section A of the Supplementary Material, for any tensor-product spline function $g(s_1, s_2, t) = \sum_{j \in \mathcal{J}} \gamma_j \psi_j(s_1, s_2, t) \in \mathbb{T}^{(e,d,r)}(\mathcal{E})$, there exist positive constants C_1 and C_2 depending on ϱ, d, r , such that, $C_1 h |\Delta|^2 \sum_{j \in \mathcal{J}} \gamma_j^2 \leq \|g\|_{L^2, \Omega \times \mathcal{T}}^2 \leq C_2 h |\Delta|^2 \sum_{j \in \mathcal{J}} \gamma_j^2$.*

3.2. Asymptotic properties of the TPST estimators

Without loss of generality, in this section, we assume $\lambda_1 = \lambda_{1,1} = \dots = \lambda_{1,p_2}$ and $\lambda_2 = \lambda_{2,1} = \dots = \lambda_{2,p_2}$. Theorem 1 illustrates that $\hat{\alpha}$, $\hat{\sigma}^2$, $\hat{\boldsymbol{\eta}}$, and $\hat{\boldsymbol{\beta}}$ are consistent estimators. See Sections B and C of the Supplementary Material for detailed proofs.

Theorem 1. *Suppose Assumptions (A1)–(A12) in Section A of the Supplementary Material hold, α_0 , σ_0^2 , $\boldsymbol{\eta}_0$, and $\boldsymbol{\beta}_0$ are globally identifiable, and $\hat{\alpha}$, $\hat{\sigma}^2$, $\hat{\boldsymbol{\eta}}$, and $\hat{\boldsymbol{\beta}}$ are consistent estimators of α_0 , σ_0^2 , $\boldsymbol{\eta}_0$, and $\boldsymbol{\beta}_0$, respectively.*

Theorem 2 establishes the asymptotic normality of the proposed estimators $\hat{\alpha}$, $\hat{\sigma}^2$, and $\hat{\boldsymbol{\eta}}$. Let $\boldsymbol{\kappa} = (\alpha, \sigma^2, \boldsymbol{\eta}^\top)^\top$ and $\boldsymbol{\kappa}_0 = (\alpha_0, \sigma_0^2, \boldsymbol{\eta}_0^\top)^\top$. Let $\mathbf{G} = \mathbf{W}\boldsymbol{\Xi}_0^{-1}$. Denote

$$\begin{aligned} \boldsymbol{\Sigma}_{11,n} &= \frac{1}{n} \mathbb{E}\{\text{tr}(\mathbf{G}^2)\} + \frac{1}{n} \mathbb{E}\{\text{tr}(\mathbf{G}^\top \mathbf{G})\} + \frac{1}{n\sigma_0^2} \mathbb{E}(\boldsymbol{\mu}_0^\top \mathbf{G}^\top \mathbf{H}_{\mathbb{X}_{\psi^*}, \Lambda}^\top \mathbf{H}_{\mathbb{X}_{\psi^*}, \Lambda} \mathbf{G} \boldsymbol{\mu}_0), \\ \boldsymbol{\Sigma}_{22,n} &= \frac{1}{2\sigma_0^4}, \quad \boldsymbol{\Sigma}_{33,n} = \frac{1}{n\sigma_0^2} \mathbb{E}(\mathbb{Z}^\top \mathbf{H}_{\mathbb{X}_{\psi^*}, \Lambda}^\top \mathbf{H}_{\mathbb{X}_{\psi^*}, \Lambda} \mathbb{Z}), \quad \boldsymbol{\Sigma}_{23,n} = \mathbf{0}, \\ \boldsymbol{\Sigma}_{12,n} &= \boldsymbol{\Sigma}_{21,n} = \frac{1}{n\sigma_0^2} \mathbb{E}\{\text{tr}(\mathbf{G})\}, \quad \boldsymbol{\Sigma}_{13,n} = \boldsymbol{\Sigma}_{31,n} = \frac{1}{n\sigma_0^4} \mathbb{E}(\mathbb{Z}^\top \mathbf{H}_{\mathbb{X}_{\psi^*}, \Lambda}^\top \mathbf{H}_{\mathbb{X}_{\psi^*}, \Lambda} \mathbf{G} \boldsymbol{\mu}_0), \end{aligned}$$

where $\mathbf{H}_{\mathbb{X}_{\psi^*}, \Lambda}$ is defined in (B.3) in the Supplementary Material. For any matrix \mathbf{A} , let $(\mathbf{A})_{ii}$ be the (i, i) th entry of matrix \mathbf{A} , and let \mathbf{A}_i represent the i th row of \mathbf{A} . Denote

$$\begin{aligned} \boldsymbol{\Omega}_{11,n} &= \frac{m_4 - \sigma_0^3}{n\sigma_0^4} \mathbb{E} \left\{ \sum_{i=1}^n (\mathbf{G})_{ii}^2 \right\} + \frac{2m_3}{n\sigma_0^4} \sum_{i=1}^n \mathbb{E}\{(\mathbf{G})_{ii} (\mathbf{H}_{\mathbb{X}_{\psi^*}, \Lambda})_i \mathbf{G} \boldsymbol{\mu}_0\}, \\ \boldsymbol{\Omega}_{22,n} &= \frac{m_4 - 3\sigma_0^4}{4n\sigma_0^8}, \quad \boldsymbol{\Omega}_{33,n} = \mathbf{0}, \quad \boldsymbol{\Omega}_{23,n} = \boldsymbol{\Omega}_{32,n} = \frac{m_3}{2n\sigma_0^6} \mathbb{E}(\mathbf{1}_n^\top \mathbf{H}_{\mathbb{X}_{\psi^*}, \Lambda} \mathbb{Z}), \end{aligned}$$

$$\begin{aligned} \Omega_{12,n} &= \Omega_{21,n} = \frac{1}{2n\sigma_0^6} \mathbb{E} \left\{ (m_4 - 3\sigma_0^4) \text{tr}(\mathbf{G}) + m_3 \mathbf{1}_n^\top \mathbf{H}_{\mathbb{X}_{\psi^*}, \Lambda} \mathbf{G} \boldsymbol{\mu}_0 \right\}, \\ \Omega_{13,n} &= \Omega_{31,n} = \frac{m_3}{2n\sigma_0^6} \mathbb{E} \left(\mathbf{1}_n^\top \mathbf{H}_{\mathbb{X}_{\psi^*}, \Lambda} \mathbb{Z} \right), \end{aligned}$$

where m_3 and m_4 are the third and fourth moments of ϵ_i , respectively. Next, let

$$\Sigma_n = \begin{pmatrix} \Sigma_{11,n} & \Sigma_{12,n} & \Sigma_{13,n} \\ \Sigma_{21,n} & \Sigma_{22,n} & \Sigma_{23,n} \\ \Sigma_{31,n} & \Sigma_{32,n} & \Sigma_{33,n} \end{pmatrix}, \quad \Omega_n = \begin{pmatrix} \Omega_{11,n} & \Omega_{12,n} & \Omega_{13,n} \\ \Omega_{21,n} & \Omega_{22,n} & \Omega_{23,n} \\ \Omega_{31,n} & \Omega_{32,n} & \Omega_{33,n} \end{pmatrix}.$$

Theorem 2. *Under Assumptions (A1)–(A12) in Section A of the Supplementary Material, $\sqrt{n}(\hat{\boldsymbol{\kappa}} - \boldsymbol{\kappa}_0) \rightarrow N(\mathbf{0}, \Sigma^{-1} + \Sigma^{-1} \Omega \Sigma^{-1})$, where $\Sigma = \lim_{n \rightarrow \infty} \Sigma_n$ and $\Omega = \lim_{n \rightarrow \infty} \Omega_n$.*

Theorem 2 illustrates the advantages of considering a model with spatiotemporally varying coefficients. If we ignore the spatiotemporal heterogeneity in the model, less variation of the data will be explained, which leads to an estimator of σ_0^2 larger than the true value, along with a larger variance of other parameters.

Theorem 3 provides the L_2 convergence rate of the spline estimators $\hat{\beta}_k$.

Theorem 3. *Under Assumptions (A1)–(A12) in Section A of the Supplementary Material, for any $k = 1, \dots, p_2$, the spline estimators $\hat{\beta}_k$ are consistent and satisfy that*

$$\begin{aligned} \|\hat{\beta}_k - \beta_{0k}\|_{L_2} &= O_P \left(|\Delta|^{d+1} + h^\varrho + \lambda_1 n^{-1} |\Delta|^{-3} h^{-1/2} \right. \\ &\quad \left. + \lambda_2 n^{-1} |\Delta|^{-1} h^{-5/2} + n^{-1/2} |\Delta|^{-1} h^{-1/2} \right). \end{aligned}$$

Remark 1. Let $a = (2\varrho + 1)(d + 2) - 1$. For $|\Delta| \asymp n^{-\varrho/a}$, $h \asymp n^{-(d+1)/a}$, $\lambda_1 = O(n^{1/2-2\varrho/a})$, and $\lambda_2 = O(n^{1/2-2(d+1)/a})$, the tensor-penalized spline estimator attains the L_2 convergence rate $n^{-\varrho(d+1)/a}$. Specifically, if $\varrho = d + 1$, and we take $|\Delta| \asymp h \asymp n^{-1/(2\varrho+3)}$ and $\lambda_1 \asymp \lambda_2 = O(n^{(2\varrho-1)/(4\varrho+6)})$, then the L_2 convergence rate is $n^{-\varrho/(2\varrho+3)}$, which is the optimal convergence rate in Stone (1982).

4. Implementation

4.1. Triangular prismatic partition selection

To construct a triangular prismatic partition, we need to determine the number and locations of the knots for the univariate splines, and use triangulation for the bivariate splines. Here, typical triangulation construction methods include the Delaunay Triangulation, MATLAB code “Distmesh” (Persson and Strang

(2004)), and R package “Triangulation” (Lai and Wang (2019)). Note that Remark 1 suggests that $|\Delta| \asymp n^{-\varrho/a}$, $h \asymp n^{-(d+1)/a}$, where $\varrho \geq 1$ is the order of the univariate spline basis functions, and $d \geq 1$ is the degree of the bivariate spline basis functions. Next, we explain how to select the knots and triangulation.

Knots selection. For the univariate splines, we consider the widely used quadratic and cubic spline basis functions. For the locations of the knots, we suggest placing knots on a grid of evenly spaced sample quantiles. For the number of the knots, we take: $N = \min\{\lfloor c_1 n^{1/(2\varrho+3)} \rfloor, \lfloor n_T/(4p_2) \rfloor\} + 1$, where n_T is the total number of observed time points, c_1 is a tuning parameter (typically, $c_1 \in [1, 5]$), and $\lfloor x \rfloor$ denotes the integer part of a real number x . The term $n^{1/(2\varrho+3)}$ ensures the property stated in Remark 1, and the term $n_T/(4p_2)$ guarantees that there are at least four observations in each subinterval between two adjacent knots to avoid getting (near) singular design matrices in smoothing. Specifically, if we use the piecewise quadratic univariate spline ($\varrho = 3$) and the piecewise quadratic bivariate spline ($d = 2$), we take $N = \min\{\lfloor c_1 n^{1/9} \rfloor, \lfloor n_T/(4p_2) \rfloor\} + 1$.

Triangulation selection. There are some core criteria that one can use when selecting a triangulation. In general, a “good” triangulation refers to those with well-shaped triangles, that is, no small angles and/or no obtuse angles. For a fixed number of triangles, Lai and Schumaker (2007) and Yu et al. (2020) recommend constructing the triangulation according to the “max-min” criterion, which maximizes the minimum angle of all the angles of the triangles in the triangulation. Monte Carlo experiments show that the triangulation should be fine enough to capture the features of the function, but once this minimum necessary number of triangles has been reached, further refining the triangulation usually has little effect on the fitting process, but increases the computational burden. In practice, if the boundary of the spatial domain is not complicated, we suggest taking the number of triangles as the following: $\min\{\lfloor c_2 n^{2/(2d+5)} \rfloor, \lfloor n_S/(4p_2) \rfloor\} + 1$, for a tuning parameter c_2 , where n_S is the number of observed spatial location points. When $\varrho = 3$ and $d = 2$, we take $K = \min\{\lfloor c_2 n^{2/9} \rfloor, \lfloor n_S/(4p_2) \rfloor\} + 1$. For simple spatial domains, we suggest taking $c_2 \in [1, 10]$. However, c_2 can be taken from 10 to 20 for complex domains, such that the triangulation well approximates the domain, and the penalty term can regularize the model complexity.

4.2. Roughness penalty selection

In the spatiotemporal problem, data are often generated with dependence. However, when performing cross-validation (CV), these dependence structures are usually ignored, which can lead to underestimation of the predictive error (Roberts et al. (2017)). To tackle this problem, we adopt the block CV strategy in

Roberts et al. (2017) and Valavi et al. (2019). The sample points are first divided into spatiotemporal blocks using nonoverlapped and equal-volume cuboids. Then, these blocks are randomly allocated to the CV folds. Choosing the optimal size of the blocks is important; see Section D.1 in the Supplementary Material for a detailed procedure.

4.3. Standard error formula for the estimates

To calculate the standard errors of the estimators of the parameters in the STAR-PLVCM, we need to estimate the matrices Σ and Ω in Theorem 2. Let $\hat{\boldsymbol{\mu}} = \hat{\boldsymbol{\mu}}(\hat{\alpha})$, $\hat{\mathbf{G}} = \mathbf{W}\Xi(\hat{\alpha})^{-1}$, $\hat{m}_3 = n^{-1} \sum_{i=1}^n \hat{\epsilon}_i^3$, $\hat{m}_4 = n^{-1} \sum_{i=1}^n \hat{\epsilon}_i^4$, and $\hat{\boldsymbol{\epsilon}} = (\hat{\epsilon}_1, \dots, \hat{\epsilon}_n)^\top = \mathbf{Y}(\hat{\alpha}) - \hat{\boldsymbol{\mu}}$. For any matrix or vector \mathbf{A} with n rows, define a linear operator P_Λ such that $P_\Lambda \mathbf{A} = \Pi_{\mathbb{X}_{\psi^*}, \Lambda} \mathbf{A}$, where $\Pi_{\mathbb{X}_{\psi^*}, \Lambda}$ is defined in (B.3). We can estimate the elements in Σ and Ω as follows:

$$\begin{aligned} \hat{\Sigma}_{11,n} &= \frac{1}{n} \text{tr}(\hat{\mathbf{G}}^2) + \frac{1}{n} \text{tr}(\hat{\mathbf{G}}^\top \hat{\mathbf{G}}) + \frac{1}{n\hat{\sigma}^2} \{ \hat{\mathbf{G}}\hat{\boldsymbol{\mu}} - P_\Lambda(\hat{\mathbf{G}}\hat{\boldsymbol{\mu}}) \}^\top \{ \hat{\mathbf{G}}\hat{\boldsymbol{\mu}} - P_\Lambda(\hat{\mathbf{G}}\hat{\boldsymbol{\mu}}) \}, \\ \hat{\Sigma}_{22,n} &= \frac{1}{2\hat{\sigma}^4}, \quad \hat{\Sigma}_{33,n} = \frac{1}{n\hat{\sigma}^2} (\mathbf{Z} - P_\Lambda \mathbf{Z})^\top (\mathbf{Z} - P_\Lambda \mathbf{Z}), \quad \hat{\Sigma}_{23,n} = \mathbf{0}, \\ \hat{\Sigma}_{12,n} &= \hat{\Sigma}_{21,n} = \frac{1}{n\hat{\sigma}^2} \text{tr}(\hat{\mathbf{G}}), \quad \hat{\Sigma}_{13,n} = \hat{\Sigma}_{31,n} = \frac{1}{n\hat{\sigma}^4} (\mathbf{Z} - P_\Lambda \mathbf{Z})^\top \{ \hat{\mathbf{G}}\hat{\boldsymbol{\mu}} - P_\Lambda(\hat{\mathbf{G}}\hat{\boldsymbol{\mu}}) \}, \\ \hat{\Omega}_{11,n} &= \frac{\hat{m}_4 - \hat{\sigma}^3}{n\hat{\sigma}^4} \sum_{i=1}^n (\hat{\mathbf{G}})_{ii}^2 + \frac{2\hat{m}_3}{n\hat{\sigma}^4} \sum_{i=1}^n (\hat{\mathbf{G}})_{ii} (\mathbf{I}_n - \Pi_{\mathbb{X}_{\psi^*}, \Lambda})_{ii} \hat{\mathbf{G}}\hat{\boldsymbol{\mu}}, \\ \hat{\Omega}_{22,n} &= \frac{\hat{m}_4 - 3\hat{\sigma}^4}{4n\hat{\sigma}^8}, \quad \hat{\Omega}_{33,n} = \mathbf{0}, \quad \hat{\Omega}_{23,n} = \hat{\Omega}_{32,n} = \frac{\hat{m}_3}{2n\hat{\sigma}^6} \mathbf{1}_n^\top (\mathbf{Z} - P_\Lambda \mathbf{Z}), \\ \hat{\Omega}_{12,n} &= \hat{\Omega}_{21,n} = \frac{1}{2n\hat{\sigma}^6} \left\{ (\hat{m}_4 - 3\hat{\sigma}^4) \text{tr}(\hat{\mathbf{G}}) + \hat{m}_3 \mathbf{1}_n^\top \{ \hat{\mathbf{G}}\hat{\boldsymbol{\mu}} - P_\Lambda(\hat{\mathbf{G}}\hat{\boldsymbol{\mu}}) \} \right\}, \\ \hat{\Omega}_{13,n} &= \hat{\Omega}_{31,n} = \frac{\hat{m}_3}{2n\hat{\sigma}^6} \mathbf{1}_n^\top (\mathbf{Z} - P_\Lambda \mathbf{Z}). \end{aligned}$$

4.4. Specification of the weight matrices

It is crucial to choose proper weights for the STAR-PLVCM. A proper weight matrix can substantially benefit the model by including both the spatial and the temporal dependence of the data. Pace et al. (1998) considered the following weight matrices: $\phi_S \mathbf{W}_S + \phi_T \mathbf{W}_T$ and $\phi_S \mathbf{W}_S + \phi_T \mathbf{W}_T + \phi_{ST} \mathbf{W}_S \mathbf{W}_T + \phi_{TS} \mathbf{W}_T \mathbf{W}_S$, where \mathbf{W}_S specifies the spatial relations among observations, \mathbf{W}_T specifies the temporal relations among observations, and ϕ_S , ϕ_T , ϕ_{ST} , and ϕ_{TS} are parameters. In Huang, Wu and Barry (2010), the authors construct the weight matrix based on the spatiotemporal distance $d = \sqrt{d_S^2 + ad_T^2}$, where d_S is the spatial distance between two points, d_T is the temporal distance, and a is some

parameter. In simulation study 2, similarly to Huang, Wu and Barry (2010), we calculate the spatiotemporal distance with $a = 1$, construct the weight matrix based on 10 nearest points as the neighborhood, and assign the same weights for the neighbor points. In the application study, the weight matrix has the form $\phi_S \mathbf{W}_S + \phi_T \mathbf{W}_T$. The parameters ϕ_S and ϕ_T are selected using CV. In practice, we can use several different weight matrices and choose the one with the best prediction performance.

5. Monte Carlo Study

In this section, we conduct two experiments to evaluate the finite-sample performance of the proposed method using the recently developed R package “STARX” (Yu, Wang and Wang (2020)). In both experiments, we randomly sample n_S points from the spatial domain Ω , which is the modified horseshoe domain in Sangalli, Ramsay and Ramsay (2013), and each point is observed at a sequence of n_T equally spaced time points over $\mathcal{T} = [0, 1]$. Both experiments are conducted on a local computer with a 3.8 GHz 8-Core Intel Core i7 processor and 32GB RAM. We conduct 100 Monte Carlo replications for each simulation setting.

5.1. Simulation study 1

In this example, we randomly sample $n_S = 200$ and 500 locations $\{\mathbf{S}_i \equiv (S_{i1}, S_{i2}), i = 1, \dots, n_S\}$ from the spatial domain Ω , and each point is observed at a sequence of $n_T = 50$ and 100 equally spaced time points over $\mathcal{T} = [0, 1]$. For the observed space-time points $\{(\mathbf{S}_1, T_1), \dots, (\mathbf{S}_n, T_n)\} \equiv \{(\mathbf{S}_1, 1/n_T), \dots, (\mathbf{S}_{n_S}, 1/n_T), (\mathbf{S}_1, 2/n_T), \dots, (\mathbf{S}_{n_S}, 1)\}$, we generate data from the following special case of the STAR-PLVCM with $\alpha_0 = 0$ and $\boldsymbol{\eta}_0 = \mathbf{0}$: for any $i = 1, \dots, n$,

$$Y_i = \beta_{00}(S_{i1}, S_{i2}, T_i) + \beta_{01}(S_{i1}, S_{i2}, T_i)X_{i1} + \beta_{02}(S_{i1}, S_{i2}, T_i)X_{i2} + \epsilon_i, \quad (5.1)$$

where $\beta_{00}(s_1, s_2, t) = 2m_0(s_1, s_2)(t - 0.5)^2$, $m_0(\cdot, \cdot)$ is a bivariate function given in Sangalli, Ramsay and Ramsay (2013), $\beta_{01}(s_1, s_2, t) = 2 \cos(0.5s_1 + s_2^2)t$, and $\beta_{02}(s_1, s_2, t) = 2 \sin\{\pi s_2(t - 0.5)\}$. Figure D.3 in the Supplementary Material shows the sequences of spatial plots of the true coefficient functions evaluated at time points $t = 0.0, 0.17, 0.50, 0.83$, and 1.0. The covariates $\{X_{i1}\}_{i=1}^n$ and $\{X_{i2}\}_{i=1}^n$ are independently generated from the normal distribution $N(0, 1)$. The error term ϵ_i is generated from $N(0, \sigma_0^2)$, with the noise level σ_0 being 1.0 or 2.0. Model (5.1) is the spatiotemporally varying-coefficient model in Huang, Wu and Barry (2010) and Fotheringham, Crespo and Yao (2015), referred to as the

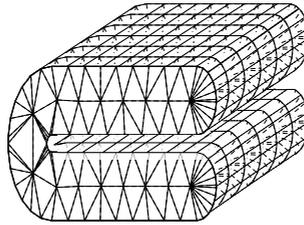


Figure 2. Triangular prime partition \mathcal{E} .

STVCM.

To implement our proposed method, we use TPST with $\varrho = 3$, $d = 2$, and $r = 1$ to estimate the coefficient functions. Figure 2 shows the triangular prismatic partition \mathcal{E} that we use in this simulation study.

We compare our method with the GTWR method, which is commonly used to fit STVCMs. This method is implemented by the R package `GWmodel`. The mean integrated squared error (MISE) for β_{00} , β_{01} , and β_{02} and the average computing time are reported in Table 1. Figure D.1 (a)–(c) in the Supplementary Material show box plots of the MISEs of the estimators of the varying coefficients. The results indicate that our method significantly outperforms the GTWR in terms of both estimation accuracy and computing speed. The MISEs of our method are much lower than those of the GTWR. In addition, our proposed TPST method is computationally efficient and easy to implement using the R package `TPST`. The computing time of the GTWR increases dramatically as the sample size grows, in contrast to our proposed method, which barely increases. Figure D.3 in Supplementary Material shows the sequences of the estimated coefficient functions for the TPST and GTWR methods based on a typical run with $n_S = 200$, $n_T = 50$, and $\sigma_0 = 1.0$. It is clear that the estimated functions of the TPST are very similar to the true functions. However, the GTWR estimates have some obvious bias because they do not take the complex boundary into any account and they smooth across the gap inappropriately.

5.2. Simulation study 2

We generate data from the following:

$$Y_i = \alpha_0 \sum_{j=1}^n w_{ij} Y_j + \eta_{00} + \eta_{01} Z_{i1} + \eta_{02} Z_{i1} + \beta_{01}(S_{i1}, S_{i2}, T_i) X_{i1} + \epsilon_i, \quad i = 1, \dots, n,$$

where $\alpha_0 = 0.5$, $\eta_{00} = 5$, $\eta_{01} = 1$, $\eta_{02} = -1$, $\beta_{01}(s_1, s_2, t) = 2m_0(s_1, s_2)(t - 0.5)^2$, and $m_0(\cdot, \cdot)$ is a bivariate function given in Sangalli, Ramsay and Ramsay (2013).

Table 1. Mean integrated squared errors (MISEs) of the varying-coefficient estimators and the average computing time in Simulation Study 1.

(n_S, n_T)	σ_0	Method	MISE $_{\beta_{00}}$	MISE $_{\beta_{01}}$	MISE $_{\beta_{02}}$	Time (seconds)
(200,50)	1.0	TPST	0.0095	0.0086	0.0103	418.5
		GTWR	0.0393	0.0428	0.0479	1919.0
	2.0	TPST	0.0169	0.0197	0.0168	408.1
		GTWR	0.0745	0.0696	0.0865	1,959.7
(200,100)	1.0	TPST	0.0082	0.0067	0.0095	534.1
		GTWR	0.0284	0.0322	0.0367	4,321.1
	2.0	TPST	0.0136	0.0143	0.0143	528.9
		GTWR	0.0543	0.0513	0.0619	4,201.0
(500,50)	1.0	TPST	0.0062	0.0060	0.0079	592.1
		GTWR	0.0248	0.0269	0.0283	5,486.3
	2.0	TPST	0.0103	0.0124	0.0112	597.9
		GTWR	0.0477	0.0450	0.0548	5,700.9
(500,100)	1.0	TPST	0.0070	0.0051	0.0082	884.1
		GTWR	–	–	–	–
	2.0	TPST	0.0098	0.0092	0.0105	896.7
		GTWR	–	–	–	–

The “–” indicates that the result is unavailable owing to out of memory crashes.

See Figure D.4 in the Supplementary Material for a sequence of spatial plots of the coefficient function β_{01} at different time points. The weight matrix $\mathbf{W} = (w_{ij})$ is a standardized row matrix; that is, $w_{ij} = \delta_{ij} / \sum_{j \neq i} \delta_{ij}$, where $\delta_{ij} = 1$ if (S_{j1}, S_{j2}, T_j) is among the 10 nearest neighbors of (S_{i1}, S_{i2}, T_i) , otherwise $\delta_{ij} = 0$. The covariates Z_{i1} , Z_{i2} , and X_{i1} are independently generated from $N(0, 1)$. Furthermore, the error term ϵ_i is generated from $N(0, \sigma_0^2)$. In our simulation below, we consider $(n_S, n_T) = (100, 30), (100, 50), (200, 30), (200, 50)$, and $\sigma_0 = 0.5$ and 1.0 .

To examine the effect of the triangular prism, we consider the following six different triangular prisms: $\mathcal{E}_q, q = 1, \dots, 6$. For $q = 1, 2, 3$, \mathcal{E}_q is constructed based on Δ_q with three equally spaced quantile interior knots; for $q = 4, 5, 6$, \mathcal{E}_q is constructed based on Δ_{q-3} with five equally spaced quantile interior knots. Figure D.2 in the Supplementary Material shows the three triangulations $\Delta_q, q = 1, 2, 3$.

We calculate the mean squared error (MSE) for the estimators of the constant parameters, $\alpha_0, \sigma_0, \eta_{0\ell}$, for $\ell = 0, 1, 2$, and the MISE for the estimator of the varying coefficient β_{01} in the STAR-PLVCM. Table D.1 in the Supplementary Material reports the MSE for the TPST estimators of the constant parameters and the MISE of the TPST estimator of the varying-coefficient function. The

results show that, at noise level $\sigma_0 = 0.5$, the MSE or MISE of the estimators is nearly constant for all six triangular prisms, indicating that the number of triangular prism elements is not very important when there is a fair amount of noise. When the triangular prism is too fine, often there is a slight penalty of statistical accuracy, especially when the sample size is small. Figure D.4 in the Supplementary Material shows sequences of spatial plots of the estimated coefficient functions using the TPST with different triangular prisms with $n_S = 100$, $n_T = 30$, and $\sigma_0 = 0.5$. The plots are very similar. Table D.1 summarizes the numerical results for these TPST estimators. The results indicate similar estimation performance across different triangular prisms.

Next, we compare our method with the classical spatiotemporal autoregressive linear model (STAR-LM), where all the coefficients are treated as constant. We also consider the GTWAR model, in which all the coefficients are treated as varying-coefficient functions. In Table 2, we use “LM,” “VCM,” and “PLVCM” to distinguish three classes of spatiotemporal autoregressive models with different types of coefficients. We use the R package `spdep` to fit the STAR-LM, and use $\hat{\beta}_{01}$ as the estimator of $\beta_{01}(s_1, s_2, t)$. The GTWAR is implemented in R, and we apply 10-fold CV to select the proper bandwidth in the kernel smoothing procedure. In Wu, Li and Huang (2014), the global parameter α_0 , η_{00} , η_{01} , and η_{02} are all considered as spatiotemporally varying functions. Therefore, in this simulation example, the GTWAR estimators of α_0 , η_{00} , η_{01} , and η_{02} are the averages of the estimated functions for $80 \times 50 \times 50$ grid points over $\Omega \times \mathcal{T}$. In addition, to illustrate the prediction capability, we conduct 10-fold CV for each Monte Carlo sample and compare the CV mean squared prediction error (MSPE). The performance of our method, the STAR-LM, and the GTWAR is reported in Table 2. From Table 2, one can observe that both the estimation error and the prediction error of our method are much lower than those of the STAR-LM and GTWAR.

Finally, we check the accuracy of the proposed standard error formula in Section 4.3 for α_0 , σ_0^2 , η_{00} , η_{01} , and η_{02} . The results, shown in Table 3, indicate that the averages or medians of the estimated standard errors are very close to the true standard deviations, which verifies the accuracy of the proposed standard error formula. In addition, the SE_{IQR} are much smaller than those of the other three SEs, which implies that the variance of the standard error calculated by our formula is very small.

Table 2. Mean squared error (MSE) and mean integrated squared error (MISE) of the estimators of the constant (functional) parameters and 10-fold cross-validation mean squared prediction error (MSPE) of Y in Simulation Study 2.

(n_S, n_T)	σ_0	Model	Method	MSE ($\times 10^3$)					MISE ($\times 10^3$)	MSPE	Time
				α_0	σ_0^2	η_{00}	η_{01}	η_{02}	β_{01}		
(100,30)	0.5	LM	STAR-LM	0.57	278.71	58.36	0.21	0.21	303.38	0.54	44.2
		VCM	GTWAR	0.76	–	75.24	0.16	0.16	45.39	0.34	1,673.7
		PLVCM	TPST(\mathcal{E}_1)	0.34	0.06	34.77	0.10	0.08	7.21	0.27	413.7
	1.0	LM	STAR-LM	1.11	277.26	113.84	0.49	0.45	303.75	1.29	42.7
		VCM	GTWAR	2.18	–	216.41	0.41	0.59	123.17	1.15	1,675.3
		PLVCM	TPST(\mathcal{E}_1)	1.04	1.08	106.18	0.39	0.33	19.09	1.03	429.4
(100,50)	0.5	LM	STAR-LM	0.38	275.58	37.65	0.11	0.09	303.11	0.53	154.4
		VCM	GTWAR	0.81	–	82.22	0.13	0.13	49.30	0.31	2,428.0
		PLVCM	TPST(\mathcal{E}_1)	0.25	0.03	25.02	0.04	0.04	5.39	0.26	570.1
	1.0	LM	STAR-LM	0.73	275.80	73.75	0.22	0.20	303.40	1.29	155.1
		VCM	GTWAR	2.11	–	212.83	0.30	0.31	88.81	1.12	2,318.6
		PLVCM	TPST(\mathcal{E}_1)	0.76	0.55	76.68	0.16	0.16	14.10	1.02	585.2
(200,30)	0.5	LM	STAR-LM	0.36	280.24	36.34	0.08	0.09	302.44	0.54	451.3
		VCM	GTWAR	0.54	–	54.13	0.14	0.10	50.05	0.32	2,554.2
		PLVCM	TPST(\mathcal{E}_1)	0.18	0.03	18.51	0.03	0.05	4.44	0.26	874.3
	1.0	LM	STAR-LM	0.63	279.26	64.63	0.18	0.22	302.62	1.29	447.3
		VCM	GTWAR	1.42	–	143.90	0.41	0.31	67.27	1.12	2,700.2
		PLVCM	TPST(\mathcal{E}_1)	0.59	0.51	59.54	0.13	0.18	11.71	1.02	874.7
(200,50)	0.5	LM	STAR-LM	0.18	275.79	17.36	0.05	0.04	302.44	0.53	1,891.3
		VCM	GTWAR	0.24	–	24.25	0.06	0.05	39.80	0.30	4,267.6
		PLVCM	TPST(\mathcal{E}_1)	0.08	0.02	8.44	0.03	0.02	3.25	0.26	2,296.8
	1.0	LM	STAR-LM	0.34	273.90	33.58	0.14	0.10	302.53	1.28	1,838.6
		VCM	GTWAR	0.76	–	75.24	0.16	0.16	45.39	1.08	4,001.9
		PLVCM	TPST(\mathcal{E}_1)	0.29	0.29	29.50	0.13	0.09	8.20	1.02	2,346.1

6. An Empirical Application to COVID-19 Infection and Death Data

6.1. COVID-19 data

As an empirical illustration, we apply the proposed methodology to a study of the spread of COVID-19 in the United States. For infectious diseases, the transmission pattern depends on many factors and varies with location and time. For example, in COVID-19 studies, the effect of control policy on the spread of SAS-CoV-2 differs from county to county. Government agencies also adjust the control measures at different stages of disease spread. The above observations motivate us to consider a spatiotemporally varying-coefficient model to

Table 3. Standard error estimates of the constant parameters in Simulation Study 2.

(n_S, n_T)	Parameter	$\sigma_0 = 0.5$				$\sigma_0 = 1.0$			
		SE _{mc}	SE _{mean}	SE _{median}	SE _{IQR}	SE _{mc}	SE _{mean}	SE _{median}	SE _{IQR}
(100,30)	α_0	0.018	0.018	0.018	0.0005	0.034	0.031	0.031	0.0008
	σ_0^2	0.006	0.006	0.006	0.0002	0.025	0.025	0.025	0.0007
	η_{00}	0.186	0.176	0.176	0.0052	0.346	0.309	0.310	0.0085
	η_{01}	0.010	0.009	0.009	0.0002	0.020	0.018	0.018	0.0004
	η_{02}	0.009	0.009	0.009	0.0002	0.018	0.018	0.018	0.0004
(100,50)	α_0	0.015	0.014	0.014	0.0003	0.027	0.024	0.024	0.0005
	σ_0^2	0.005	0.005	0.005	0.0001	0.019	0.020	0.020	0.0004
	η_{00}	0.154	0.137	0.137	0.0038	0.274	0.241	0.241	0.0054
	η_{01}	0.007	0.007	0.007	0.0001	0.013	0.014	0.014	0.0002
	η_{02}	0.006	0.007	0.007	0.0001	0.013	0.014	0.014	0.0003
(200,30)	α_0	0.014	0.013	0.013	0.0002	0.023	0.022	0.022	0.0004
	σ_0^2	0.005	0.005	0.005	0.0001	0.020	0.018	0.018	0.0004
	η_{00}	0.138	0.125	0.125	0.0024	0.231	0.220	0.220	0.0039
	η_{01}	0.006	0.006	0.006	0.0001	0.012	0.013	0.013	0.0002
	η_{02}	0.007	0.006	0.006	0.0001	0.013	0.013	0.013	0.0002
(200,50)	α_0	0.010	0.010	0.010	0.0002	0.020	0.017	0.017	0.0003
	σ_0^2	0.004	0.004	0.004	0.0001	0.015	0.014	0.014	0.0002
	η_{00}	0.100	0.098	0.098	0.0019	0.204	0.172	0.172	0.0026
	η_{01}	0.006	0.005	0.005	0.0001	0.011	0.010	0.010	0.0001
	η_{02}	0.005	0.005	0.005	0.0001	0.010	0.010	0.010	0.0001

SE_{mc}, the standard deviation of estimated parameters based on 100 Monte Carlo samples (can be viewed as the true values for SE); SE_{mean}, mean of the estimated SE from 100 simulations; SE_{median}, median of the estimated SE from 100 simulations; SE_{IQR}, interquartile range of the estimated SE from 100 Monte Carlo replications divided by 1.349.

capture the heterogeneity across space and time. In particular, we focus on the spatiotemporal dynamics of the disease, accounting for mobility and other local features. The data for the COVID-19 outbreak are collected and cleaned from a combination of public data repositories. These data sets are introduced and their sources are provided in Table D.2 in the Supplementary Material. We consider five types of local features: socioeconomic status, healthcare infrastructure, demographic characteristics, mobility, and a rural/urban factor. See Table D.3 in the Supplementary Material for a detailed explanation of the covariates.

6.2. Estimation results

We collect the daily infection count data for 3,104 counties in the 48 contiguous states and the District of Columbia from April 22 to June 21, 2020. In total, we have 186,240 observations with space-time points $\{(\mathbf{S}_1, T_1), \dots, (\mathbf{S}_n, T_n)\}$, where $n_S = 3,104$ is the number of observed spatial points, $n_T = 60$ is the length of the observed date, and $\mathbf{S}_i \equiv (S_{i1}, S_{i2})$ is the latitude and longitude of the

geographic center of a county. Denote Y_i as the new confirmed cases at county \mathbf{S}_i and date T_i . We take $\log(Y_i + 1)$ as the response variable, and consider the following model for the infection count:

$$\begin{aligned} \log(Y_i + 1) = & \sum_{j=1}^n \alpha^I w_{ij} \log(Y_j + 1) + \eta_0^I + \eta_1 \text{AA}_i + \eta_2^I \text{HL}_i + \eta_3^I \text{PD}_i^* + \eta_4^I \text{Old}_i \\ & + \eta_5^I \text{Sex}_i + \eta_6^I \text{Affluence}_i + \eta_7^I \text{Disadvantage}_i + \eta_8^I \text{Gini}_i + \eta_9^I \text{Urban}_i + \eta_{10}^I \text{NHIC}_i \\ & + \eta_{11}^I \text{EHPC}_i + \eta_{12}^I \text{TBed}_i^* + \eta_{13}^I \text{Mobility}_i + \beta^I(\mathbf{S}_i, T_i) \log(I_i + 1) + \varepsilon_i^I, \end{aligned} \quad (6.1)$$

where I_i is the number of active cases for county location \mathbf{S}_i and day $T_i - 1$. We also apply our proposed model and method to the number of fatal cases. Let D_i be the number of new deaths for county location \mathbf{S}_i and day T_i . According to CDC (2020), the median number of days from symptom onset to death is around 15 days. Therefore, in our death model, we consider the variable I_i^D , which is the number of active cases for county location \mathbf{S}_i and day $T_i - 15$. We consider the following death model:

$$\begin{aligned} \log(D_i + 1) = & \sum_{j=1}^n \alpha^D w_{ij} \log(D_j + 1) + \eta_0^D + \eta_1 \text{AA}_i + \eta_2^D \text{HL}_i + \eta_3^D \text{PD}_i^* + \eta_4^D \text{Old}_i \\ & + \eta_5^D \text{Sex}_i + \eta_6^D \text{Affluence}_i + \eta_7^D \text{Disadvantage}_i + \eta_8^D \text{Gini}_i + \eta_9^D \text{Urban}_i + \eta_{10}^D \text{NHIC}_i \\ & + \eta_{11}^D \text{EHPC}_i + \eta_{12}^D \text{TBed}_i^* + \eta_{13}^D \text{Mobility}_i + \beta^D(\mathbf{S}_i, T_i) \log(I_i^D + 1) + \varepsilon_i^D. \end{aligned} \quad (6.2)$$

In this empirical study, to create the the weight matrix \mathbf{W} , we need to generate the spatial weight matrix \mathbf{W}_S and the temporal weight matrix \mathbf{W}_T , as described in Section 4.4. We follow the idea in Nappi and Maury (2009) to construct the spatial and temporal weight matrices $\mathbf{W}_S = (w_{ij,S})$ and $\mathbf{W}_T = (w_{ij,T})$, respectively, as follows: for any $i, j = 1, \dots, n$, $w_{ij,S} = I(\mathbf{S}_i$ and \mathbf{S}_j are adjacent counties, and $T_j = T_i - 1)$, $w_{ij,T} = I(1 \leq T_i - T_j \leq r_0$ and $\mathbf{S}_i = \mathbf{S}_j)$, where $I(\cdot)$ is the indicator function and we take $r_0 = 7$. Similarly to Pace et al. (1998) and Nappi and Maury (2009), we construct the weight matrix $\mathbf{W} = \phi \widetilde{\mathbf{W}}_T + (1 - \phi) \widetilde{\mathbf{W}}_S$, where $0 \leq \phi \leq 1$, and $\widetilde{\mathbf{W}}_S$ and $\widetilde{\mathbf{W}}_T$ are the matrices obtained from \mathbf{W}_S and \mathbf{W}_T , respectively, by normalizing their row sums to one. The parameter ϕ is chosen by partitioning the data into training and testing groups and evaluating its prediction errors; $\phi = 0.8$ in this application. We use the TPST with triangulation presented in Figure D.5 in the Supplementary Material and three evenly distributed interior knots within 60 days to fit the infection model (6.1) and death model (6.2). We also consider the STAR model with the same autoregressive structure, but no additional predictors. Table 5 presents the estimated α and σ^2

Table 4. Estimated linear coefficients (Est), estimated standard errors(SE), and the corresponding p -values of county-level predictors in STAR-PLVCM.

Parameter	Infection model			Death model		
	Est ($\times 10^2$)	SE ($\times 10^2$)	p -value	Est ($\times 10^2$)	SE ($\times 10^2$)	p -value
Intercept	-7.0952	0.4345	$< 10^{-8}$	-2.9294	0.1746	$< 10^{-8}$
AA	0.0938	0.2767	0.7347	0.5880	0.1164	4.38×10^{-7}
HL	1.0838	0.2640	4.03×10^{-5}	0.0546	0.1098	0.6188
Gini	2.5419	0.1754	$< 10^{-8}$	0.9575	0.0749	$< 10^{-8}$
Affluence	0.2953	0.2482	0.2341	0.0030	0.1057	0.9775
Disadvantage	-0.9565	0.2702	4.00×10^{-4}	-0.1685	0.1143	0.1403
Urban	3.6082	0.2637	$< 10^{-8}$	0.2732	0.1123	1.50×10^{-2}
NHIC	1.5829	0.2449	$< 10^{-8}$	0.6273	0.1039	$< 10^{-8}$
EHPC	-0.0552	0.1542	0.7202	0.0302	0.0657	0.6462
Sex	-0.1764	0.1598	0.2698	-0.0694	0.0682	0.3089
PD	1.7372	0.3310	1.54×10^{-7}	-0.0501	0.1375	0.7158
TBed	1.9764	0.1625	$< 10^{-8}$	0.4414	0.0693	$< 10^{-8}$
Old	-0.9226	0.1980	3.17×10^{-6}	0.2782	0.0841	9.35×10^{-4}
Mobility	1.7222	0.1625	$< 10^{-8}$	0.4249	0.0690	$< 10^{-8}$

and the corresponding 95% confidence intervals for both the infection and the death models.

Table 4 reports the estimated coefficients and their corresponding p -values in both models using the STAR-PLVCM, which reveals how the county-level predictors influence the daily new cases. In the infection model, one can observe that “HL,” “Gini,” “Urban,” “NHIC,” “PD,” “TBed,” and “Mobility” have significantly positive effects on the number of daily new infection cases. This suggests that when the population density is high, people have more contact, and thus have more opportunities to spread the disease. The mobility data describe traffic trends for each county. When people commute more, they are more likely to get infected, and there are more daily new infection cases. In addition, “Disadvantage” and “Old” have negative effects on the number of daily new infection cases. For the death model (6.2), our analysis shows that “AA,” “Gini,” “Urban,” “NHIC,” “TBed,” “Old,” and “Mobility” have positive effects on daily new deaths. The risk of severe illness increases with age. Thus, communities with larger proportions of older adults tend to have a larger number of daily new deaths. However, at the same time, older adults have stricter precautions, which reduce the number of new infection cases.

The fitted varying-coefficient functions of β^I in infection model (6.1) are shown in Figure 3 (a)–(f) at six different days from April to June. At the end of April, higher values are captured in regions such as the northeast and west coast and the Midwestern states. After slightly slowing down in May, coronavirus

Table 5. Estimated (Est) α and σ^2 , estimated standard errors (SE), and the corresponding 95% confidence intervals (CIs) of county-level predictors in STAR-PLVCM.

Model	Parameter	Infection model			Death model		
		Est	SE	95% CI	Est	SE	95% CI
STAR-PLVCM	α	0.9107	0.0029	(0.9051, 0.9164)	0.9107	0.0033	(0.9042, 0.9172)
	σ^2	0.3626	0.0012	(0.3603, 0.3649)	0.0661	0.0002	(0.0656, 0.0665)

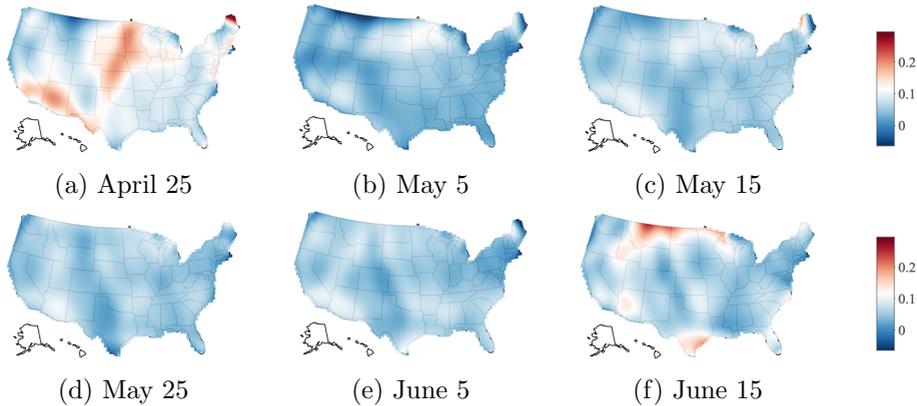


Figure 3. Spatial plots of the estimated coefficient functions in infection model.

cases started to surge in June in states such as Arizona and Texas, which is also reflected in the fitted varying-coefficient map. Figures D.6 (a)–(f) present the estimated coefficient function of β^D for the death model (6.2). The numbers of fatal cases in California and Arizona have been increasing at a faster pace than in other states since April, which corresponds to a higher value in the varying-coefficient function maps. In contrast, in the state of New York, the number of new fatal cases decreased significantly after mid-April.

6.3. Prediction performance

We compare the proposed methods with the STAR, STAR-LM, and SIR in terms of their short-term prediction performance. The STAR model does not include county-level predictors. The STAR-LM assumes the β^I and β^D in the Models (6.1) and (6.2) are constant. The SIR fits a Susceptible-Infectious-Recovered model for each county. For simplicity, we denote $\hat{I}_{i,t+h}$ and $\hat{I}_{i,t+h}^D$ as the h -day-ahead prediction of the infected cases and deaths in county i based on the data before day t , while $I_{i,t+h}$ and $I_{i,t+h}^D$ are the actual observed number of cases. We use data collected from April 2 to June 23, 2020, to evaluate the prediction performance. Each time, we fit the model with a set of $n_T = 60$ days

Table 6. Root mean squared prediction error (RMSPE) of STAR-PLVCM, STAR-LM, STAR, and SIR models for COVID-19 data.

Model	RMSPE (Infection)			RMSPE (Death)			Avg. Time(s)
	R_1^I	R_2^I	R_3^I	R_1^D	R_2^D	R_3^D	
STAR-PLVCM	29.1966	54.0458	81.2411	1.1806	1.988	2.7109	2,630.89
STAR-LM	31.9166	60.0774	91.0084	1.1585	1.9275	2.6116	7.05
STAR	33.1791	62.8897	95.4139	1.1752	1.9652	2.6798	5.77
SIR	135.4645	156.9696	180.4252	–	–	–	4,682.32

of the county-level data for the estimation, and then the h -day-ahead predictions are computed, where $h = 1, 2, 3$. The procedure is repeated 20 times. The root mean squared prediction errors (RMSPEs) are calculated and presented in Table 6: for $h = 1, 2, 3$ and $T = 20$, $R_h^I = T^{-1} \sum_{t=1}^T \{n_S^{-1} \sum_{i=1}^{n_S} (\hat{I}_{i,t+h} - I_{i,t+h})^2\}^{1/2}$ and $R_h^D = T^{-1} \sum_{t=1}^T \{n_S^{-1} \sum_{i=1}^{n_S} (\hat{I}_{i,t+h}^D - I_{i,t+h}^D)^2\}^{1/2}$.

As expected, the STAR-PLVCM outperforms the STAR, STAR-LM, and SIR in terms of the infection model's prediction accuracy. By adding county-level predictors and considering spatiotemporally varying-coefficient functions, the STAR-PLVCM is more flexible and can capture more local features. For the death model, the STAR, STAR-LM, and STAR-PLVCM have similar prediction performance. Compared with infected cases, deaths counts are more rare across the United States. Many counties have zero daily new deaths. Therefore, a flexible model with a more complex structure has a limited advantage under this scenario. Figures D.7 (a)–(d) in the Supplementary Material show example cases when the traditional SIR model does not work. Without integrating nearby information, the county-level prediction of the SIR is sensitive to the observed data of each county. For example, in Figures D.7 (b) and (d), there are jumps in the cumulative infected cases, which lead to severe over-predictions in the following seven days.

7. Conclusion

We have addressed several challenges arising from the inclusion of spatiotemporal effects in regression models. The first kind concerns the unrealistic assumption of stationary or structural stability over time and space in the regression modeling. The second addresses how to model and estimate spatiotemporal autocorrelation and heterogeneity simultaneously. We propose a flexible class of spatiotemporal autoregressive regression models that extends the ordinary spatial autoregressive models to accommodate the spatiotemporal effects of some covariates. We develop a profiled ML approach to estimate the constant param-

eters and varying-coefficient functions in the proposed semiparametric models. Our work is novel, and it also has merits in the following aspects. First, our proposed method solves the problem of “leakage” across complex domains (i.e., the inappropriate linking of parts of the domain separated by physical barriers), suffered by many conventional smoothing tools. Second, because our method does not require the data to be evenly distributed or on regular-spaced grids, it is generally applicable to many spatiotemporal data analysis problems. Finally, compared with existing approaches, such as kriging and kernel approaches, our proposed method is much more computationally efficient using the spline basis expansion technique.

The Specification of the varying covariates is crucial in the modeling of the STAR-PLVCM. Model misspecification could lead to biased regression coefficients or reduce the estimation efficiency. A test against the parametric linear coefficient will help to identify covariates with constant linear coefficients. We can use a wild bootstrap to test whether the coefficient function $\beta_{0k}(\mathbf{s}, t)$ is constant by following the idea in Ferraty, Keilegom and Vieu (2010). Such tests require simulation-based validation and a theoretical guarantee, which we leave to future work.

Owing to the existence of pre-symptomatic and asymptomatic COVID-19 cases and the potentially limited testing capacity, many infection cases may not have been reported. Our study does not consider the under-reported issue in order to simplify the illustration of the proposed methodology. Incorporating this issue is left to future work.

Supplementary Material

In the Supplemental Material, we provide the technical assumptions and proofs for the main theorems, additional results from the simulation studies, and a description of the county-level predictors used in the study of the dynamics of COVID-19.

Acknowledgments

Li Wang was partially supported by National Science Foundation award DMS-1916204. Shan Yu’s research was partially supported by the Iowa State University Plant Sciences Institute Scholars Program. The authors would like to thank the editor, associate editor, and two anonymous reviewers for their constructive comments and suggestions.

References

- Cai, Z. (2007). Trending time-varying coefficient time series models with serially correlated errors. *Journal of Econometrics* **136**, 163–188.
- CDC (2020). COVID-19 pandemic planning scenarios. Available at <https://www.cdc.gov/coronavirus/2019-ncov/hcp/planning-scenarios.html>.
- Chen, Z., Li, R. and Li, Y. (2015). Varying coefficient models for data with auto-correlated error process. *Statistica Sinica* **25**, 709–723.
- Cressie, N. and Wikle, C. K. (2011). *Statistics for Spatio-Temporal Data*. John Wiley & Sons, Hoboken, New Jersey.
- Fan, J. and Zhang, W. (2008). Statistical methods with varying coefficient models. *Statistics and its Interface* **1**, 179–195.
- Ferraty, F., Keilegom, I. V. and Vieu, P. (2010). On the validity of the bootstrap in non-parametric functional regression. *Scandinavian Journal of Statistics* **37**, 286–306.
- Fotheringham, A. S., Brunson, C. and Charlton, E. M. (2002). *Geographically Weighted Regression: The Analysis of Spatially Varying Relationships*. John Wiley & Sons, Chichester.
- Fotheringham, A. S., Crespo, R. and Yao, J. (2015). Geographical and temporal weighted regression (GTWR). *Geographical Analysis* **47**, 431–452.
- Gelfand, E. A., Kim, H.-J., Sirmans, C. F. and Banerjee, S. (2003). Spatial modeling with spatially varying coefficient processes. *Journal of the American Statistical Association* **98**, 387–396.
- Helbich, M. and Griffith, D. A. (2016). Spatially varying coefficient models in real estate: Eigenvector spatial filtering and alternative approaches. *Computers, Environment and Urban Systems* **57**, 1–11.
- Hoover, D. R., Rice, J. A., Wu, C. O. and Yang, L.-P. (1998). Nonparametric smoothing estimates of time-varying coefficient models with longitudinal data. *Biometrika* **85**, 809–822.
- Huang, B., Wu, B. and Barry, M. (2010). Geographically and temporally weighted regression for modeling spatio-temporal variation in house prices. *International Journal of Geographical Information Science* **24**, 383–401.
- Kelejian, H. H. and Prucha, I. R. (2010). Specification and estimation of spatial autoregressive models with autoregressive and heteroskedastic disturbances. *Journal of Econometrics* **157**, 53–67.
- Lai, M.-J. and Schumaker, L. L. (2007). *Spline functions on triangulations*. Cambridge University Press, New York.
- Lai, M.-J. and Wang, L. (2019). Package Triangulation. R package version 1.0. Available at <https://github.com/funstatpackages/Triangulation>.
- Lee, L.-F. (2004). Asymptotic distributions of quasi-maximum likelihood estimators for spatial autoregressive models. *Econometrica* **72**, 1899–1925.
- Lee, L.-F. and Yu, J. (2010). Estimation of spatial autoregressive panel data models with fixed effects. *Journal of Econometrics* **154**, 165–185.
- LeSage, J. P. (2008). An introduction to spatial econometrics. *Revue d'économie Industrielle*, 19–44.
- Li, R. and Liang, H. (2008). Variable selection in semiparametric regression modeling. *The Annals of Statistics* **36**, 261–286.
- Lindgren, F. and Rue, H. (2015). Bayesian spatial modelling with R-INLA. *Journal of Statistical Software* **63**, 1–25.

- Mu, J., Wang, G. and Wang, L. (2018). Estimation and inference in spatially varying coefficient models. *Environmetrics* **29**, e2485.
- Müller, H.-G., Stadtmüller, U. and Tabnak, F. (1997). Spatial smoothing of geographically aggregated data, with application to the construction of incidence maps. *Journal of the American Statistical Association* **92**, 61–71.
- Nappi, I. and Maury, T.-P. (2009). A spatiotemporal autoregressive price index for the Paris office property market. *Real Estate Economics* **37**, 305–340.
- Pace, R. K., Barry, R., Clapp, J. M. and Rodriguez, M. (1998). Spatiotemporal autoregressive models of neighborhood effects. *The Journal of Real Estate Finance and Economics* **17**, 15–33.
- Park, B. U., Mammen, E., Lee, Y. K. and Lee, E. R. (2015). Varying coefficient regression models: A review and new developments. *International Statistical Review* **83**, 36–64.
- Persson, P. O. and Strang, G. (2004). A simple mesh generator in matlab. *SIAM Review* **46**, 329–345.
- Ramsay, T. (2002). Spline smoothing over difficult regions. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* **64**, 307–319.
- Roberts, D. R., Bahn, V., Ciuti, S., Boyce, M. S., Elith, J., Guillera-Aroita, G. et al. (2017). Cross-validation strategies for data with temporal, spatial, hierarchical, or phylogenetic structure. *Ecography* **40**, 913–929.
- Sangalli, L. M., Ramsay, J. O. and Ramsay, T. O. (2013). Spatial spline regression models. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* **75**, 681–703.
- Stone, C. J. (1982). Optimal global rates of convergence for nonparametric regression. *The Annals of Statistics* **10**, 1040–1053.
- Su, L. and Jin, S. (2010). Profile quasi-maximum likelihood estimation of partially linear spatial autoregressive models. *Journal of Econometrics* **157**, 18–33.
- Valavi, R., Elith, J., Lahoz-Monfort, J. J. and Guillera-Aroita, G. (2019). blockCV: An r package for generating spatially or environmentally separated folds for k-fold cross-validation of species distribution models. *Methods in Ecology and Evolution* **10**, 225–232.
- Wang, H. and Ranalli, M. G. (2007). Low-rank smoothing splines on complicated domains. *Biometrics* **63**, 209–217.
- Wood, S. N., Bravington, M. V. and Hedley, S. L. (2008). Soap film smoothing. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* **70**, 931–955.
- Wu, B., Li, R. and Huang, B. (2014). A geographically and temporally weighted autoregressive model with application to housing prices. *International Journal of Geographical Information Science* **28**, 1186–1204.
- Xu, X., Wang, W. and Shin, Y. (2020). Dynamic spatial network quantile autoregression. IRTG 1792 Discussion Paper, No. 2020-024, Humboldt-Universität zu Berlin, International Research Training Group 1792 “High Dimensional Nonstationary Time Series”.
- Yang, L., Park, B. U., Xue, L. and Härdle, W. (2006). Estimation and testing for varying coefficients in additive models with marginal integration. *Journal of the American Statistical Association* **101**, 1212–1227.
- Yu, S., Wang, G., Wang, L., Liu, C. and Yang, L. (2020). Estimation and inference for generalized geoaddivitive models. *Journal of the American Statistical Association* **115**, 761–774.
- Yu, S. and Wang, L. (2020). Package TPST. R package version 1.0. Available at <https://github.com/funstatpackages/TPST>.

Yu, S., Wang, Y. and Wang, L. (2020). Package STARX. R package version 1.0. Available at <https://github.com/funstatpackages/STARX>.

Zhang, X. and Wang, J.-L. (2015). Varying-coefficient additive models for functional data. *Biometrika* **102**, 15–32.

Shan Yu

Department of Statistics, University of Virginia, Charlottesville, VA 22904-4135, USA.

E-mail: sy5jx@virginia.edu

Yueying Wang

Department of Biostatistics and Herbert Irving Comprehensive Cancer Center, Columbia University, New York, NY 10032, USA.

E-mail: yw3826@cumc.columbia.edu

Li Wang

Department of Statistics, George Mason University, Fairfax, VA 22030, USA.

E-mail: lwang41@gmu.edu

Lei Gao

School of Business, George Mason University, Fairfax, VA 22030, USA.

E-mail: lgao9@gmu.edu

(Received September 2020; accepted March 2021)