



A parsimonious, computationally efficient machine learning method for spatial regression

Milan Žukovič¹ · Dionissios T. Hristopoulos²

Accepted: 29 December 2023
© The Author(s) 2024

Abstract

We introduce the modified planar rotator method (MPRS), a physically inspired machine learning method for spatial/temporal regression. MPRS is a non-parametric model which incorporates spatial or temporal correlations via short-range, distance-dependent “interactions” without assuming a specific form for the underlying probability distribution. Predictions are obtained by means of a fully autonomous learning algorithm which employs equilibrium conditional Monte Carlo simulations. MPRS is able to handle scattered data and arbitrary spatial dimensions. We report tests on various synthetic and real-world data in one, two and three dimensions which demonstrate that the MPRS prediction performance (without hyperparameter tuning) is competitive with standard interpolation methods such as ordinary kriging and inverse distance weighting. MPRS is a particularly effective gap-filling method for rough and non-Gaussian data (e.g., daily precipitation time series). MPRS shows superior computational efficiency and scalability for large samples. Massive datasets involving millions of nodes can be processed in a few seconds on a standard personal computer. We also present evidence that MPRS, by avoiding the Gaussian assumption, provides more reliable prediction intervals than kriging for highly skewed distributions.

Keywords Machine learning · Interpolation · Time series · Scattered data · Non-Gaussian model · Precipitation · Autonomous algorithm

1 Introduction

The spatial prediction (interpolation) problem arises in various fields of science and engineering that study spatially distributed variables. In the case of scattered data, filling gaps facilitates understanding of the spatial features, visualization of the observed process, and it is also necessary to obtain fully populated grids of spatially dependent parameters used in partial differential equations. Spatial prediction is highly relevant to many disciplines, such as environmental

mapping, risk assessment (Christakos 2012) and environmental health studies (Christakos and Hristopoulos 2013), subsurface hydrology (Kitanidis 1997; Rubin 2003), mining (Goovaerts 1997), and oil reserves estimation (Hohn 1988; Hamzehpour and Sahimi 2006). In addition, remote sensing images often include gaps with missing data (e.g., clouds, snow, heavy precipitation, ground vegetation coverage, etc.) that need to be filled (Rossi et al. 1994). Spatial prediction is also useful in image analysis (Winkler 2003; Gui and Wei 2004) and signal processing (Unser and Blu 2005; Ramani and Unser 2006) including medical applications (Parrott et al. 1993; Cao and Worsley 2001).

Spatial interpolation methods in the literature include simple deterministic approaches, such as inverse distance weighting (Shepard 1968) and minimum curvature (Sandwell 1987), as well as the widely-used family of kriging estimators (Cressie 1990). The latter are stochastic methods, with their popularity being due to favorable statistical properties (optimality, linearity, and unbiasedness under ideal conditions). Thus, kriging usually outperforms other interpolation methods in prediction accuracy. However, the computational complexity of kriging increases

✉ Milan Žukovič
milan.zukovic@upjs.sk

Dionissios T. Hristopoulos
dchristopoulos@tuc.gr

¹ Department of Theoretical Physics and Astrophysics, Institute of Physics, Faculty of Science, Pavol Jozef Šafárik University, Park Angelinum 9, Košice 041 54, Slovak Republic

² School of Electrical and Computer Engineering, Technical University of Crete, Akrotiri Campus, Chania 73100, Crete, Greece

cubically with the sample size and thus becomes impractical or infeasible for large datasets. On the other hand, massive data are now ubiquitous due to modern sensing technologies such as radars, satellites, and lidar.

To improve computational efficiency, traditional methods can be modified leading to tolerable loss of prediction performance (Cressie and Johannesson 2018; Furrer et al. 2006; Ingram et al. 2008; Kaufman et al. 2008; Marcotte and Allard 2018; Zhong et al. 2016). With new developments in hardware architecture, another possibility is provided by parallel implementations using already rather affordable multi-core CPU and GPU hardware architectures (Cheng 2013; de Ravé et al. 2014; Hu and Shu 2015; Pesquer et al. 2011). A third option is to propose new prediction methods that are inherently computationally efficient.

One such approach employs Boltzmann-Gibbs random fields to model spatial correlations by means of short-range “interactions” instead of the empirical variogram (or covariance) function used in geostatistics (Hristopulos 2003; Hristopulos and Elogne 2007; Hristopulos 2015). This approach was later extended to non-Gaussian gridded data by using classical spin models (Žukovič and Hristopulos 2009a, b, 2013, 2018; Žukovič et al. 2020). The latter were shown to be computationally efficient and competitive in terms of prediction performance with respect to several other interpolation methods. Moreover, their ability to operate without user intervention makes them ideal candidates for automated processing of large datasets on regular spatial grids, typical in remote sensing. Furthermore, the short-range (nearest-neighbor) interactions between the variables allows parallelization and thus further increase in computational efficiency. For example, a GPU implementation of the modified planar rotator (MPR) model led to impressive speed-ups (up to almost 500 times on large grids), compared to single CPU calculations (Žukovič et al. 2020).

The MPR method is limited to 2D grids, and its extension to scattered data is not straightforward. In the present paper we propose the modified planar rotator for scattered data (MPRS). The MPRS method is non-parametric in the sense that it does not require any assumption for the underlying probability distribution. This new machine learning method can be used for scattered or gridded data in spaces with different dimensions. MPRS achieves even higher computational efficiency than MPR due to full vectorization of the algorithm. This new approach does not rely on a particular structure or dimension of the data location grid; it only needs the distances between each prediction point and a predefined number of samples in its neighborhood. This feature makes MPRS applicable to scattered data in arbitrary dimensions.

2 The MPRS Model

The MPRS model exploits an idea initially used in the modified planar rotator (MPR) model (Žukovič and Hristopulos 2018). The latter was introduced for filling data gaps of continuously-valued variables distributed on 2D rectangular grids. The key idea is to map the data to continuous “spins” (i.e., variables defined in the interval $[-1, 1]$), and then construct a model of spatial dependence by imposing interactions between spins. MPRS models such interactions even between scattered data and is thus applicable to both structured and unstructured (scattered) data over domains $D \subset \mathbb{R}^d$ where d is any integer.

2.1 Model definition

Let \mathbf{s} denote a spatial location inside the domain of interest. A random field $Z(\mathbf{s};\omega) : \mathbb{R}^d \times \Omega \rightarrow \mathbb{R}$ is defined over a complete probability space (Ω, \mathcal{F}, P) , where Ω is the sample space, \mathcal{F} is the event space (i.e., space of events comprising set of states in Ω), and P is the probability function which assigns a number between 0 and 1 to each event in \mathcal{F} . Finally, the state index ω selects a specific state from Ω . Assume that the data are sampled at points $G_s = \{\mathbf{s}_i \in \mathbb{R}^d\}_{i=1}^N$ from the field $Z(\mathbf{s};\omega)$. The dataset is denoted by $Z_s = \{z_i \in \mathbb{R}\}_{i=1}^N$, and the set of prediction points by $G_p = \{\mathbf{s}_p \in \mathbb{R}^d\}_{p=1}^P$ so that $G_s \cup G_p = G$, $G_s \cap G_p = \emptyset$ (i.e., the sampling and prediction sets are disjoint), and $P + N = N_G$. The random field values at the prediction sites will be denoted by the set Z_p .

A Boltzmann-Gibbs probability density function (PDF) can be defined for the configuration $z(\mathbf{s})$ sampled over $G_s \subset \mathbb{R}^d$. The PDF is governed by the Hamiltonian (energy functional) $\mathcal{H}(z_{G_s})$ and is given by the exponential form

$$f_Z(z_{G_s}) = \mathcal{Z}^{-1} \exp\{-\mathcal{H}(z_{G_s})/k_B T\}, \quad (1)$$

where $z_{G_s} \triangleq \{z(\mathbf{s}) : \mathbf{s} \in G_s\}$ is the set of data values at the sampling points, and \mathcal{Z} is a normalizing constant (known as the partition function). In statistical physics, T is the thermodynamic temperature and k_B is the Boltzmann constant. In the case of the MPRS model, the product $k_B T$ represents a model parameter that controls the variance of the field.

A local-interaction Hamiltonian can in general be expressed as

$$\mathcal{H} = -\langle J_{ij} \Phi(z_i, z_j) \rangle, \quad (2)$$

where J_{ij} is a location-dependent pair-coupling function and $\Phi(z_i, z_j)$ is a nonlinear function of the interacting values z_i, z_j . The notation $\langle \cdot \rangle$ implies a spatial averages defined by means of

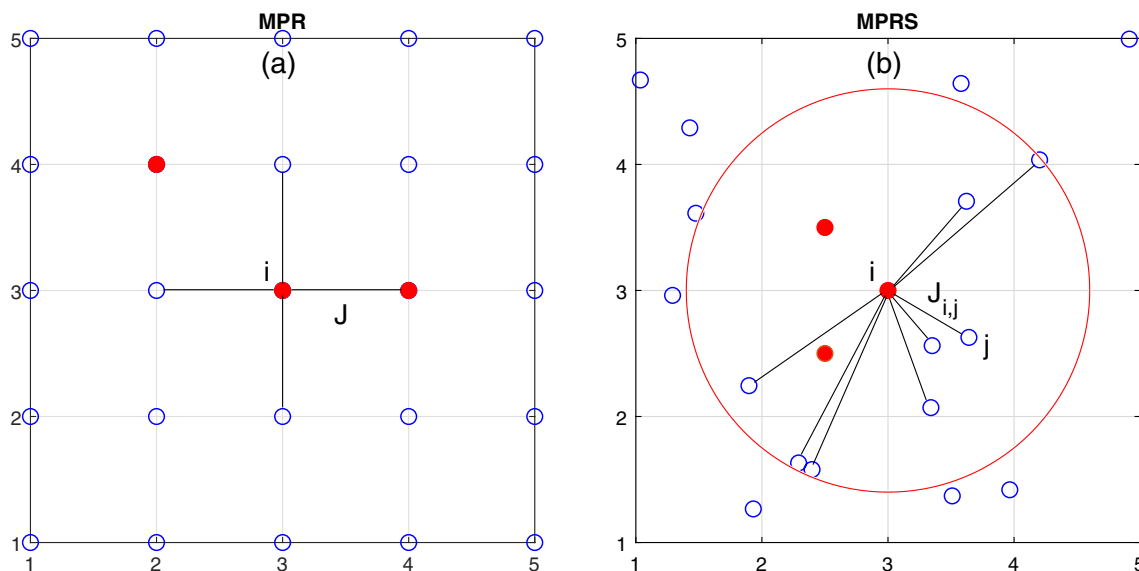


Fig. 1 Schematic illustration of the interactions of i th prediction point with **a** its four nearest neighbors (including sampling and prediction points) via the constant interaction parameter J in MPR and **b** its $n_b = 8$ nearest neighbor (only sampling) points via the mutual

distance-dependent interaction parameter $J_{i,j}$ in MPRS. Blue open and red filled circles denote sampling and prediction points, respectively, and the solid lines represent the bonds

$$\langle A_{i,j} \rangle \triangleq \sum_i \sum_{j \in \text{neighb}(i)} A_{i,j} \tag{3}$$

where $A_{i,j}$ is a two-point function, and $\text{neighb}(i)$ denotes all n_b sampling points in the interaction neighborhood of the i -th point.

To define the local interactions in the MPRS model, the original data z_i are mapped to continuously-valued “spin” variables represented by angles ϕ_i using the linear transformation

$$z_i \mapsto \phi_i = \frac{2\pi(z_i - z_{s,\min})}{z_{s,\max} - z_{s,\min}}, \text{ for } z_i \in Z_s, \tag{4}$$

where $z_{s,\min} = \min_{i \in \{1,2,\dots,N\}} Z_s$, $z_{s,\max} = \max_{i \in \{1,2,\dots,N\}} Z_s$, and $\phi_i \in [0, 2\pi]$, for $i = 1, \dots, N$. The MPRS pairwise energy is given by the equation

$$\Phi_{i,j} = \cos \phi_{i,j}, \text{ where } \phi_{i,j} = (\phi_i - \phi_j)/2. \tag{5}$$

In order to fully determine interactions between scattered data, the coupling function $J_{i,j}$ needs to be defined. It is reasonable to assume that the strength of the interactions diminishes with increasing distance. Hence, we adopt an exponential decay of the interactions between two points i and j , i.e.,

$$J_{i,j} = J_0 \exp(-r_{i,j}/b_i). \tag{6}$$

In the coupling function (6), the constant J_0 defines the maximum intensity of the interactions, $r_{i,j} = \|s_i - s_j\|$ is the

pair distance, and the locally adaptive bandwidth parameter b_i is specific to each prediction point and reflects the sampling configuration in the neighborhood of the point s_i . Note that although the coupling function decays smoothly, the energy (2) embodies interactions only between points that are inside the specified neighborhood of each point s_i .

The interactions in the MPRS model are schematically illustrated and compared with the MPR interactions in Fig. 1. The diagram clarifies how the MPRS method extends the coupling to scattered datasets.

In MPRS, regression is accomplished by means of a conditional simulation approach which is described below. To predict the value of the field at the points in G_p , the energy function (2) is extended to include the prediction points, i.e., we use $\mathcal{H}(z_G) = \mathcal{H}(z_{G_s} \cup z_{G_p})$. In $\mathcal{H}(z_G)$ we restrict interactions between each prediction point and its sample neighbors (i.e., we neglect interactions between prediction points) in order to allow vectorization of the algorithm which enhances computational performance. In practice, omitting prediction-point interactions does not impact significantly the prediction.¹ Then, the Hamiltonian comprises two parts: one that involves only sample-to-sample interactions and one that involves interactions of the prediction points with the

¹ This observation is based on the comparison with the prediction performance of the original MRR method (Žukovič and Hristopulos 2018), which is designed to operate on gridded data and which considers both prediction-sample and prediction-prediction point interactions. Our tests on gridded data indicated that the prediction performance of the two methods are comparable.

samples in their respective neighborhood. Since the sample values are fixed, the first part contributes an additive constant, while the important (for predictive purposes) contribution comes from the second part of the energy. The latter represents a summation of the contributions from all P prediction points.

The optimal values of the spin angles ϕ_p at $\mathbf{s}_p \in G_p$ can then be determined by finding the configurations which maximize the Boltzmann-Gibbs PDF (1), where the energy is now replaced with $\mathcal{H}(z_G)$. If $T = 0$, the PDF is maximized by the configuration which minimizes the total energy $\mathcal{H}(z_G)$, i.e.,

$$\hat{\phi}_p = \arg \min_{\phi_p} \mathcal{H}(z_G), \quad p = 1, \dots, P. \tag{7}$$

However, for $T \neq 0$, there can exist many configurations $\{\phi_p\}_{p=1}^P$ that lead to the same energy $\mathcal{H}(z_G) = E$. Assuming that $\Omega(E)$ is the total number of configurations with energy E , the probability $P(E)$ of observing E is $P(E) \propto \Omega(E) \exp[-\mathcal{H}(z_G)/k_B T]$. Equivalently, we can write this as follows

$$P(E) \propto e^{-\frac{1}{k_B T} [\mathcal{H}(z_G) - k_B T \log \Omega(E)]}. \tag{8}$$

Taking into account that $S(E) = k_B \log \Omega(E)$ is the entropy that corresponds to the energy E , the exponent of (8) becomes proportional to the free energy: $F(E) = \mathcal{H}(z_G) - T S(E)$. Thus, for $T \neq 0$ an ‘‘optimal configuration’’ is obtained by means of

$$\hat{\phi}_p = \arg \min_{\phi_p} [\mathcal{H}(z_G) - T S(E)], \quad p = 1, \dots, P. \tag{9}$$

The minimum free energy corresponds to the thermal equilibrium state. In practice, the latter can be achieved in the long-time limit by constructing a sequence (Markov chain) of states using one of the legitimate updating rules, such as the Metropolis algorithm (Metropolis et al. 1953), as shown in Sect. 2.3.

Finally, the MPRS prediction at the sites $\mathbf{s}_p \in G_p$ is formulated by inverting the linear transform (4), i.e.,

$$\hat{z}_p = (z_{s,\max} - z_{s,\min}) \frac{\hat{\phi}_p}{2\pi} + z_{s,\min}, \quad \text{for } p = 1, \dots, P. \tag{10}$$

The MPRS model is fully defined in terms of the equations (1)–(10).

2.2 Setting the MPRS Model Parameters and Hyperparameters

The MPRS learning process involves the model parameters and a number of hyperparameters which control the approach of the model to an equilibrium probability

distribution. The model parameters include the number of interacting neighbors per point, n_b , the decay rate vector $\mathbf{b} = (b_1, \dots, b_p)^T$ used in the exponential coupling function (6), the prefactor J_0 , and the simulation temperature T ; the ratio of the latter two sets the interaction scale via the reduced coupling parameter $J_0/k_B T$. Thus, in the following we refer to the ‘‘simulation temperature’’ (T) as shorthand for the dimensionless ratio $k_B T/J_0$. In addition, henceforward energy functions \mathcal{H} are calculated with $J_0 = 1$.

Model parameters are typically updated during the training process. However, in order to optimize computational performance, after experimentation with various datasets, we set the model parameters to reasonable default values, i.e., $n_b = 8$ (for all prediction points) and $T = 10^{-3}$; the decay rates $\{b_p\}_{p=1}^P$ are estimated as the median distance between the p -th prediction point and its four nearest sample neighbors. These choices are supported by (i) the expectation of increased spatial continuity for low T and (ii) experience with the MPR method. In particular, MPR tends to perform better at very low T (i.e., for $T \approx 10^{-3}$). In addition, using higher-order neighbor interactions ($n_b = 8$ neighbors, i.e., nearest- and second-nearest neighbors on the square grid) improves the smoothness of the regression surface. The definition of the decay rate vector \mathbf{b} enables it to adapt to potentially uneven spatial distribution of samples around prediction points.

Our exploratory tests showed that the prediction performance is not very sensitive to the default values defined above (see Sect. 5). For example, setting $n_b = 4$ or increasing (decreasing) T by one order of magnitude, we obtained similar results as for the default parameter choices. Nevertheless, we tested the default settings on various datasets and verified that even if they are not optimal, they still provide competitive performance.

The MPRS hyperparameters are used to control the learning process. The static hyperparameters are listed in Section 1.3.1 of Algorithm 1. Below, we discuss their definition, impact on prediction performance, and setting of default values. The number of equilibrium configurations, M , is arbitrarily set to 100. Smaller (larger) values would increase (decrease) computational performance and decrease (increase) prediction accuracy and precision. The frequency of equilibrium state verification is controlled by n_f which is set to 5. Lower n_f increases the frequency and thus slightly decreases the simulation speed but it can lead to earlier detection of the equilibrium state. In order to test for equilibrium conditions, we need to check the slope of the energy evolution curve: in the equilibrium regime the curve is flat, while it has an overall negative slope in the relaxation (non-equilibrium) regime. However, the fluctuations present in equilibrium at $T \neq 0$ imply that the calculated slope will always quiver around zero. To compensate

for the fluctuations, we fit the energy evolution curve with a Savitzky-Golay polynomial filter of degree equal to one using a window that contains $n_{\text{fit}} = 20$ points. This produces a smoothed curve and a more robust estimate of the slope. Larger values of n_{fit} are likely to cause undesired mixing of the relaxation and the equilibrium regimes.

Algorithm 1 MPRS learning procedure. The algorithm involves the UPDATE function which is described in Algorithm 2. Φ_s is the vector of known spin values at the

sample sites. $\hat{\Phi}$ represents the vector of estimated spin values at the prediction sites. $G(\cdot)$ is the transformation from the original field to the spin field and $G^{-1}(\cdot)$ is its inverse. $\hat{\mathbf{Z}}(j)$, $j = 1, \dots, M$ is the j -th realization of the original field. $\mathbf{U}(0, 2\pi)$ denotes a vector of random numbers from the uniform probability distribution in $[0, 2\pi]$. SG stands for Savitzky-Golay filter.

```

1: Setting of parameters and hyperparameters
1.2: Model and simulation:  $n_b$ : # of bonds between each point and neighboring
samples;  $T$ : simulation temperature;  $\mathbf{b}$ : vector of decay rates;
1.3: Setting of static hyperparameters
1.3.1: Set the following to fixed values:  $M$ : # of equilibrium configurations (length
of averaging sequence) for statistics collection;  $i_{\text{max}}$ : maximum # of MC sweeps
in non-equilibrium stage;  $A_{\text{targ}}$ : target acceptance ratio of Metropolis update;
 $k_a$ : variation rate of perturbation control factor  $a$ ;  $n_f$ : verification frequency of
equilibrium conditions;  $n_{\text{fit}}$ : # of fitting points of energy evolution function;
1.3.2: Initialize dynamic hyperparameters: slope of energy evolution function
 $k(0) \leftarrow -1$ ; spin perturbation control factor  $a(0) \leftarrow 1$ ; simulated state counter
 $i \leftarrow 0$ ;
2: Data transformation
2.1:  $\Phi_s \leftarrow G(\mathbf{Z}_s)$  using (4)                                ▷ Set data spin angles
3: Interaction neighborhood
3.1: Identify  $n_b$  neighbors                                ▷ For each prediction point find  $n_b$  neighbors from
samples and calculate distances
4: Non-equilibrium spin relaxation procedure
4.1:  $\hat{\Phi}(0) \leftarrow \mathbf{U}(0, 2\pi)$                                 ▷ Initialize spin values at prediction points
while  $[k(i) < 0] \wedge [i \leq i_{\text{max}}]$  do                                ▷ Spin updating with Metropolis sweep
    4.2: UPDATE( $\hat{\Phi}(i+1), \hat{\Phi}(i), a(i), T$ )
    if  $A < A_{\text{targ}}$  then                                ▷ Check Metropolis acceptance ratio
        4.3:  $a(i+1) \leftarrow 1 + (i+1)/k_a$                                 ▷ Update perturbation control factor
    end if
    4.4: Calculate  $E(i+1) \leftarrow \mathcal{H}(G)$                                 ▷ Obtain current energy
    if  $[i \geq n_{\text{fit}}] \wedge [i \equiv 0 \pmod{n_f}]$  then                                ▷ Check frequency of slope update of  $E$ 
        4.5:  $k(i+1) \leftarrow \text{SG}$                                 ▷ Update slope of  $E$  using last  $n_{\text{fit}}$  values
    end if
    4.6:  $i \leftarrow i + 1$                                 ▷ Update MC counter
end while
5: Equilibrium state simulation
5.1:  $\hat{\Phi}^{\text{eq}}(0) \leftarrow \hat{\Phi}(i)$                                 ▷ Initialize the equilibrium state
for  $j = 0, \dots, M - 1$  do
    5.2: UPDATE( $\hat{\Phi}^{\text{eq}}(j+1), \hat{\Phi}^{\text{eq}}(j), 1, T$ )                                ▷ Generate equilibrium realizations
    5.3:  $\hat{\mathbf{Z}}(j+1) \leftarrow G^{-1}[\hat{\Phi}^{\text{eq}}(j+1)]$                                 ▷ Back-transform spin states
end for
6: return Statistics of  $M$  realizations  $\hat{\mathbf{Z}}(j)$ ,  $j = 1, \dots, M$ 

```

Algorithm 2 Restricted Metropolis updating algorithm (non-vectorized version). $\hat{\Phi}^{\text{old}}$ is the initial spin state, and $\hat{\Phi}^{\text{new}}$ is the new spin state. $\hat{\Phi}_{-p}^{\text{old}}$ is the initial spin state

excluding the point labeled by p . $U(0, 1)$ denotes the uniform probability distribution in $[0, 1]$. The hyperparameter α is the spin perturbation control factor; T is the

Table 1 Parameters and hyperparameters of the MPRS method with default values, short descriptions and recommended setting

	Default value	Description	Setting method and range
Model parameters	$n_b = 8$	Number of interacting neighbors per point	A relatively small number to secure efficient simulation of a fairly smooth regression surface; $n_b \in [4, 16]$
	$\mathbf{b} = (b_1, \dots, b_p)^\top$	Decay rate vector	Median distance between the p -th prediction point and its n_s nearest sample neighbors; $n_s \in [4, 8]$
	$k_B T / J_0 \equiv T = 10^{-3}$	Simulation temperature that controls the variance of the field	Based on MPR tests set to a sufficiently small value to achieve some spatial continuity; $T \in [10^{-5}, 10^{-2}]$
Hyperparameters	$M = 100$	Number of equilibrium configurations used for collecting statistics	Arbitrarily set to balance precision and computational time; $M \in [50, 1000]$
	$n_f = 5$	Frequency of verifying equilibrium conditions (slope of the energy evolution curve ≈ 0)	Empirically set to detect onset of equilibrium regime with moderate checking frequency; $n_f \in [2, 10]$
	$n_{fit} = 20$	Number of points used for fitting the energy evolution function	Empirically set to a sufficiently large value that ensures accurate estimate of the energy evolution function and its slope; $n_{fit} \in [10, 50]$
	$i_{max} = 500$ (optional)	Maximum number of Monte Carlo sweeps	Used to prevent potentially very long equilibration times, if the convergence is very slow
	$A_{targ} = 0.3$	Target acceptance ratio of restricted Metropolis algorithm	Controls the computational efficiency (at low temperatures) by generating perturbations that satisfy the desired acceptance rate A_{targ} ; $A_{targ} \in [0.2, 0.7]$
	$a = 1 + t/k_a$ (adjustable)	Perturbation control factor	Adaptively reset as a linearly increasing function of the simulation time t to achieve proposal of gradually smaller (more easily acceptable) perturbations
	$k_a = 3$	Characteristic time controlling the variation of a	Arbitrarily set to obtain desirable computational efficiency; $k_a \in [2, 4]$
	Initial state = “paramagnetic” (hot start)	Initial spin angle state at $t = 0$	Other initializations give similar results; Initial state $\in \{$ “paramagnetic”, “ferromagnetic”, “nearest-neighbor” $\}$

simulation temperature; P is the number of prediction sites; $\mathcal{H}(\cdot)$ is the energy for a given spin configuration.

```

procedure UPDATE( $\hat{\Phi}^{new}, \hat{\Phi}^{old}, a, T$ )
  for  $p = 1, \dots, P$  do
    1:  $u \leftarrow U(0, 1)$ 
    2:  $\hat{\Phi}_p'' \leftarrow \hat{\Phi}_p' + 2\pi(u - 0.5)/a \pmod{2\pi}$ 
    3:  $\Delta\mathcal{H} = \mathcal{H}(\hat{\Phi}_p'', \hat{\Phi}_{-p}^{old}) - \mathcal{H}(\hat{\Phi}_p', \hat{\Phi}_{-p}^{old})$ 
    4:  $AP \leftarrow \min\{1, \exp(-\Delta\mathcal{H}/T)\}$ 
    5:  $\hat{\Phi}_{-p}^{new} \leftarrow \hat{\Phi}_{-p}^{old}$ 
    if  $AP > r \leftarrow U(0, 1)$  then
      5.1:  $\hat{\Phi}_p^{new} \leftarrow \hat{\Phi}_p''$ 
    else
      5.2:  $\hat{\Phi}_p^{new} \leftarrow \hat{\Phi}_p'$ 
    end if
  end for
  6: return  $\hat{\Phi}^{new}$ 
end procedure

```

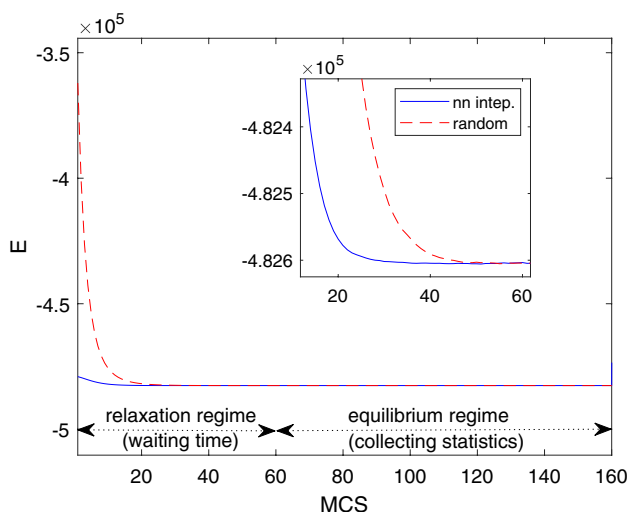



Fig. 2 Energy evolution curves starting from random (red dashed curve) and nearest-neighbor interpolation (blue solid curve) states. The simulations are performed on Gaussian synthetic data with $m = 150$, $\sigma = 25$ and Whittle-Matérn covariance model $WM(\kappa = 0.2, \nu = 0.5)$, sampled at 346, 030 and predicted at 702, 546 scattered points (non-coinciding with the sampling points) inside a square domain of length $L = 1, 024$. The inset shows a detailed view focusing on the nonequilibrium (relaxation) regime

The maximum number of Monte Carlo sweeps, i_{max} , is optional and can be set to prevent very long equilibration times, lest the convergence is very slow. Due to the efficient hybrid algorithm employed its practical impact is minimal. The target acceptance ratio of Metropolis update, A_{targ} , and the variation rate of perturbation control factor, k_a , are set to $A_{targ} = 0.3$ and $k_a = 3$. Their role is to prevent the Metropolis acceptance rate (particularly at low T) to drop to very low values, which would lead to computational inefficiency. Finally, the simulation starts from some initially selected state of the spin angle configuration. Our tests showed that different choices, such as uniform (“ferromagnetic”) or random (“paramagnetic”) initialization produced similar results. Therefore, we use as default the random state comprising spin angles drawn from the uniform distribution in $[0, 2\pi]$. While it is in principle possible to tune the hyperparameters for optimal prediction performance, using default values enables the autonomous operation of the algorithm and controls the computational efficiency. The adaptive hyperparameters, listed in Section 1.3.2 of Algorithm 1, increase the flexibility of the algorithm by automatically adapting to the current stage of the simulation process. A brief summary of all the parameters and the hyperparameters and their proposed values and recommended ranges is provided in Table 1.

2.3 Learning “Data Gaps” by Means of Restricted Metropolis Monte Carlo

MPRS predictions of the values $\{\hat{z}_p\}_{p=1}^P$ are based on conditional Monte Carlo simulation. Starting with initial guesses for the unknown values, the algorithm updates them continuously aiming to approach an equilibrium state which minimizes the free energy (see Eq. 9). The key to the computational efficiency of the MPRS algorithm is fast relaxation to equilibrium. This is achieved using the restricted Metropolis algorithm, which is particularly efficient at very low temperatures, such as the presently considered $T \approx 10^{-3}$, where the standard Metropolis updating is inefficient (Loison et al. 2004).

The classical Metropolis algorithm (Metropolis et al. 1953; Robert et al. 1999; Hristopoulos 2020) proposes random changes of the spin angles at the prediction sites (starting from an arbitrary initial state). The proposals are accepted if they lower the energy $\mathcal{H}(z_{G;curr})$, while otherwise they are accepted with probability $p = \exp[-\mathcal{H}(z_{G;prop})/T + \mathcal{H}(z_{G;curr})/T]$, where $z_{G;curr}$ is the current and $z_{G;prop}$ the proposed states. The restricted Metropolis scheme generates a proposal spin-angle state according to $\phi_{prop} = \phi_{curr} + \alpha(r - 0.5)$, where r is a uniformly distributed random number $r \in [0, 1)$ and $\alpha = 2\pi/a \in (0, 2\pi)$. The hyperparameter $a \in [1, \infty)$ controls the *spin-angle perturbations*. The value of a is dynamically tuned during the equilibration process to maintain the acceptance rate close to the target set by the *acceptance rate hyperparameter* A_{targ} . Values of $a \approx 1$ allow bigger perturbations of the current state, while $a \gg 1$ leads to proposals closer to the current state.

To achieve vectorization of the algorithm and high computational efficiency, we assume that interactions occur between prediction and sampling points in the vicinity of the former but not among prediction points. Moreover, perturbations can be performed simultaneously by means of a single sweep for all the prediction points, which increases computational efficiency (e.g., in the case of the MPR method two sweeps are required).

The learning procedure begins at an initial state ascribed to the prediction points, while the sampling points retain their values throughout the simulation.² The prediction points can be initially assigned random values drawn from the uniform distribution. It is also possible to assign values based on neighborhood relations, e.g., by means of nearest neighbor interpolation.

² If a prediction point coincides with a sample location, the MPRS algorithm allows the user to choose whether the sample value will be respected or updated. Thus, in the former (latter) case MPRS is an exact (inexact) interpolator.

Table 2 Validation measures used to assess the prediction performance of MPRS and other methods. The measures are defined as double averages: the first average is over the sites of the validation set while the second average is over all V training-validation splits. The \bar{z}_p and $\sigma_{z,p}$ are the mean and standard deviation of the validation values; $\langle \hat{z}_p^{(v)} \rangle$ and $\sigma_{\hat{z},p}^{(v)}$ are, respectively, the mean and standard deviation of the predictions for the v -th split. If $V = 1$ the initial “M” in the validation measures can be dropped

Validation measure	Definition
Mean absolute error	$MAE = \frac{1}{V} \sum_{v=1}^V \frac{1}{P} \sum_{s_p \in G_p} e^{(v)}(s_p) $
Mean absolute relative error	$MARE = \frac{1}{V} \sum_{v=1}^V \frac{1}{P} \sum_{s_p \in G_p} \frac{ e^{(v)}(s_p) }{z(s_p)}$
Mean root square error	$MRSE = \frac{1}{V} \sum_{v=1}^V \sqrt{\frac{1}{P} \sum_{s_p \in G_p} [e^{(v)}(s_p)]^2}$
Mean Pearson correlation coefficient	$MR = \frac{1}{V} \sum_{v=1}^V \frac{\sum_{p=1}^P (z(s_p) - \bar{z})(\hat{z}^{(v)}(s_p) - \langle \hat{z}_p^{(v)} \rangle)}{\sigma_{z,p} \sigma_{\hat{z},p}^{(v)}}$

Our tests showed that the initialization has marginal impact on prediction performance but opting for the latter option tends to shorten the relaxation process and thus increases computational efficiency. In Fig. 2 we illustrate the evolution of the energy (Hamiltonian) $\mathcal{H}(z_G)$ towards equilibrium using random and nearest-neighbor initial states. The curves represent interpolation on Gaussian synthetic data with Whittle-Matérn covariance (as described in Sect. 4.1). The initial energy under random initial conditions differs significantly from the equilibrium value; thus the relaxation time (measured in MC sweeps), during which the energy exhibits a decreasing trend, is somewhat longer (≈ 60 MCS) than for the nearest-neighbor initial conditions (≈ 40 MCS). Nevertheless, the curves eventually merge and level off at the same equilibrium value. In order to automatically detect the crossover to equilibrium, i.e. the flat regime of the energy curve, the energy is periodically tested every n_f MC sweeps, and the variable-degree polynomial Savitzky-Golay (SG) filter is applied (Savitzky and Golay 1964). In particular, after each n_f MC sweeps the last n_{fit} points of the energy curve are fitted to test whether the slope (decreasing trend) has disappeared.

The MPRS predictions on G_p sites are based on mean values obtained from M states that are generated via restricted Metropolis updating in the equilibrium regime. The hyperparameter M thus controls the *length of the averaging sequence*. The default value used herein is $M = 100$. Alternatively, the M values can be used to derive the *predictive distribution* at each point on G_p . The entire MPRS prediction method is summarized in Algorithms 1 and 2.

3 Study Design for Validation of MPRS Learning Method

The prediction performance of the MPRS learning algorithm is tested on various 1D, 2D, and 3D datasets. In 2D we use synthetic and real spatial data (gamma dose rates in Germany, heavy metal topsoil concentrations in the Swiss Jura mountains, Walker lake pollution, and atmospheric latent heat data over the Pacific ocean). For 1D data we use time series of temperature and precipitation. Finally, in 3D we use soil data. The MPRS performance in 1D and 2D is compared with ordinary kriging (OK) which under suitable conditions is an optimal spatial linear predictor (Kitanidis 1997; Cressie 1990; Wackernagel 2003). For better comparison (especially in terms of computational efficiency), in addition to the OK method with unrestricted neighborhood (OK-U) that uses the entire training set, we also included OK with a restricted neighborhood (OK-R) which involves the same number of neighbors as MPRS, i.e., $n_b = 8$. In 3D the MPRS method was compared with inverse distance weighting (IDW) which uses an unrestricted search neighborhood.

We compare prediction performance using different validation measures (see Table 2). The complete datasets are randomly split into disjoint training and validation subsets. In most cases, we generate $V = 100$ different training-validation splits. Let $z(s_p)$ denote the true value and $\hat{z}^{(v)}(s_p)$ its estimate at s_p for the configuration $v = 1, \dots, V$. The *prediction error* $e^{(v)}(s_p) = z(s_p) - \hat{z}^{(v)}(s_p)$ is used to define validation measures over all the training-validation splits as described in Table 2.

To assess the computational efficiency of the methods tested we record the CPU time, t_{cpu} for each split. The mean computation time $\langle t_{cpu} \rangle$ over all training-validation splits is then calculated. The MPRS interpolation method is implemented in Matlab® R2018a running on a desktop computer with 32.0 GB RAM and Intel®Core™2 i9-11900 CPU processor with a 3.50 GHz clock.

Table 3 Cross-validation measures for MPRS, OK-U and OK-R based on 100 realizations of a Gaussian random field with mean $m = 150$, standard deviation $\sigma = 25$ and covariance model $WM(\kappa = 0.2, \nu = 0.5)$, sampled at 1, 000 scattered points inside a square of length $L = 50$. 100 points are randomly selected as the training set ($tr = 0.10$) and the values at the remaining 900 points are predicted

Method	MAE	MARE (%)	RMSE	MR (%)	$\langle t_{cpu} \rangle$ (s)
MPRS	14.65	10.22	18.64	63.53	0.02
OK-U	14.28	9.94	18.19	65.63	0.01
OK-R	14.24	9.89	18.16	66.12	0.06

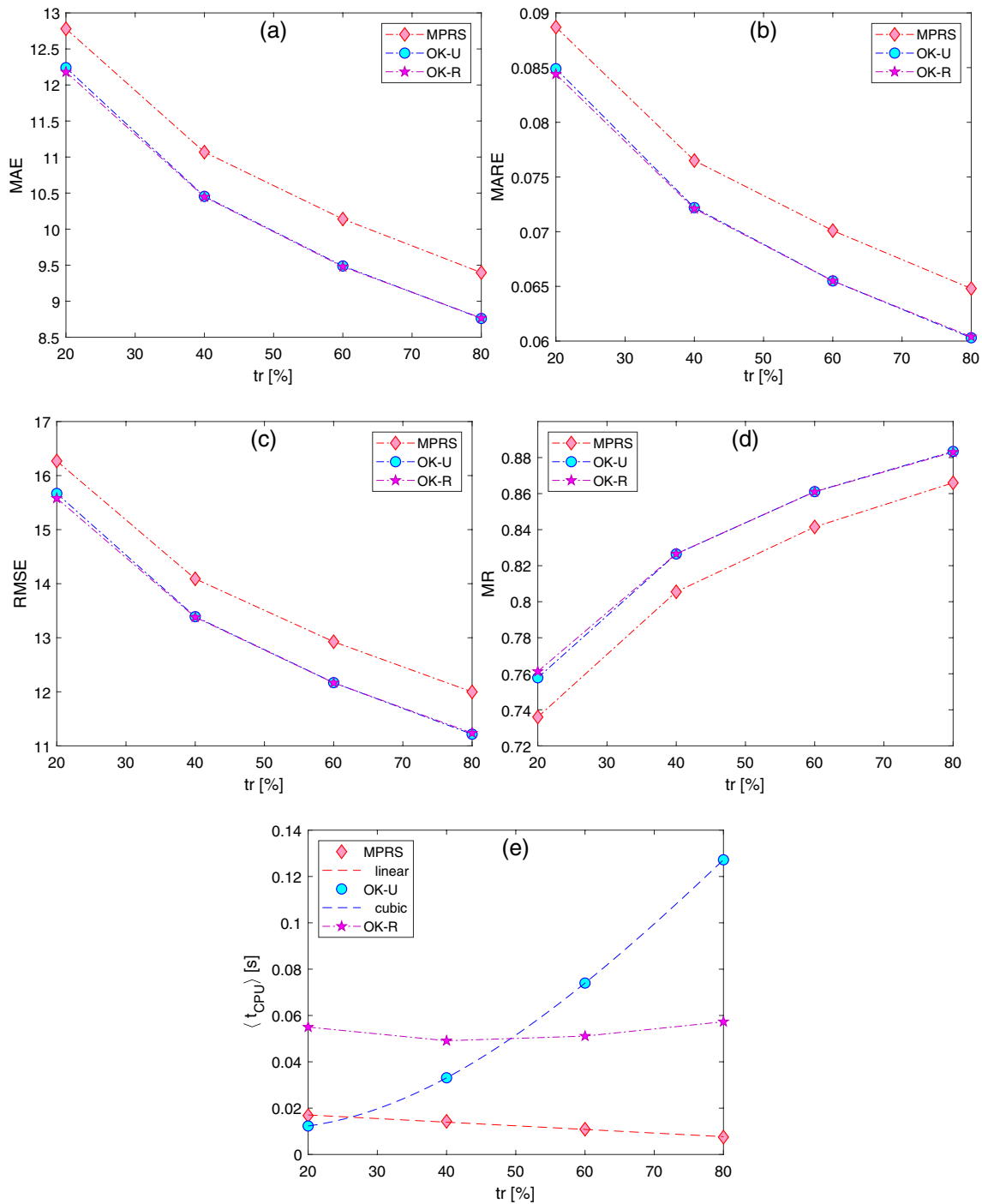


Fig. 3 Dependence of MPRS, OK-U and OK-R validation measures on the ratio of training points tr . The measures are calculated from 100 realizations of a Gaussian random field with $m = 150$, $\sigma = 25$

and covariance model $WM(\kappa = 0.2, \nu = 0.5)$; the field is sampled at 1000 scattered points inside a square domain of length $L = 50$

4 Results

4.1 Synthetic 2D data

Synthetic data are generated from Gaussian, i.e., $Z \propto N(m = 150, \sigma = 25)$, and lognormal, i.e.,

$\ln Z \propto N(m = 0, \sigma)$ spatial random fields (SRF) using the spectral method for irregular grids (Pardo-Iguzquiza and Chica-Olmo 1993). The spatial dependence is imposed by means of the Whittle-Matérn (WM) covariance given by

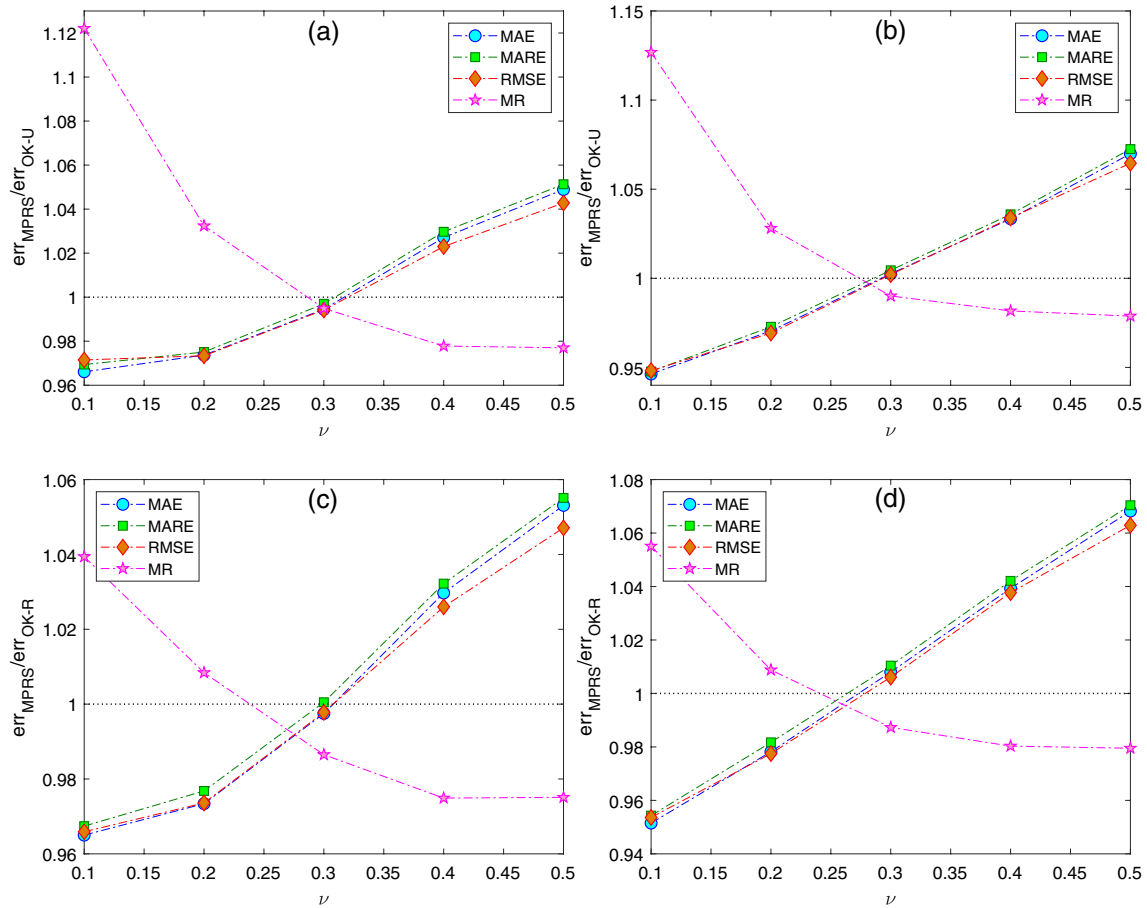


Fig. 4 Dependence of the ratios of **a, b** MPRS and OK-U and **c, d** MPRS and OK-R validation measures on the smoothness parameter. The measures are calculated based on 100 realizations of a Gaussian random field with $m = 150$, $\sigma = 25$ and the covariance model

$WM(\kappa = 0.2, \nu)$, sampled at 1,000 scattered points inside a square domain of length $L = 50$. Panels (a,c) and (b,d) show the results for $tr = 0.33$ and 0.66 , respectively

$$C_Z(\|\mathbf{h}\|) = \frac{2^{1-\nu} \sigma^2}{\Gamma(\nu)} (\kappa \|\mathbf{h}\|)^\nu K_\nu(\kappa \|\mathbf{h}\|), \tag{11}$$

where $\|\mathbf{h}\|$ is the Euclidean two-point distance, σ^2 is the variance, ν is the smoothness parameter, κ is the inverse autocorrelation length, and $K_\nu(\cdot)$ is the modified Bessel function of the second kind of order ν . Hereafter, we use the abbreviation $WM(\kappa, \nu)$ for such data. We focus on data with $\nu \leq 0.5$, which is appropriate for modeling rough spatial processes such as soil data (Minasny and McBratney 2005).

Data are generated at $N = 10^3$ random locations within a square domain of size $L \times L$, where $L = 50$. Assuming tr represents the percentage of training points, from each realization we remove $\lfloor trN \rfloor$ points to use as the training set. The predictions are cross validated with the actual values at the remaining locations. For various tr values we generate $V = 100$ different sampling configurations.

The cross-validation measures obtained by MPRS and OK for $tr = 0.10$ are summarized in Table 3. Both OK methods produce smaller errors and larger MR than MPRS. However, the relative differences are typically $\lesssim 4\%$. The CPU times of all the methods are very small, with OK-U being the fastest, followed by MPRS and OK-R being the slowest. As the analysis below will show, this ranking of methods for computational efficiency only applies for relatively small datasets. The CPU time scales with data size quite differently for MPRS and OK, favoring the former for large data sizes.

In Fig. 3 we present the evolution of all the measures with increasing tr . As expected, for higher tr values all methods give smaller errors and larger MR . Both OK methods give similar results (OK-R performing slightly better), and the differences between MPRS and OK persist and seem to slightly increase with increasing tr . While the relative prediction performance of MPRS slightly decreases its relative

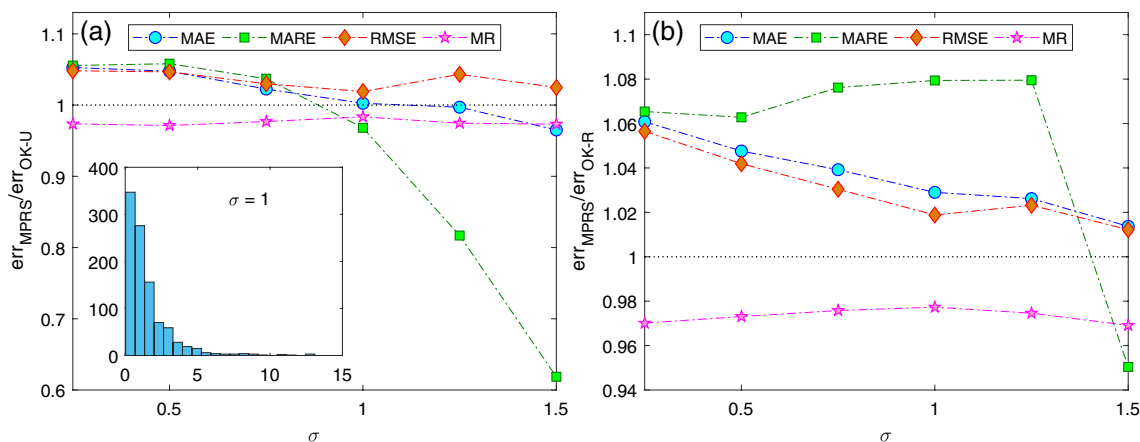


Fig. 5 Dependence of the ratios of **a** MPRS and OK-U and **b** MPRS and OK-R validation measures on the random field skewness (controlled by σ). The measures are calculated from 100 realizations of a lognormal random field with $m = 0$ and covariance model

WM($\kappa = 0.2, \nu = 0.5$), sampled at 1000 scattered points inside a square domain of length $L = 50$, for $tr = 0.33$. The inset in (a) shows, as an example, the data distribution for $\sigma = 1$

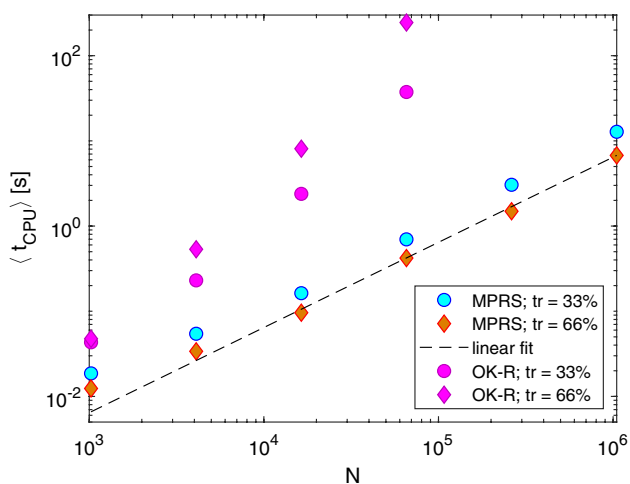


Fig. 6 MPRS and OK-R CPU times scaling vs data size N based on 100 samples of Gaussian RFs with covariance model WM($\kappa = 0.2, \nu = 0.5$). Two plots for $tr = 0.33$ (circles) and $tr = 0.66$ (diamonds) are shown. The OK-R results for the two largest N values could not be evaluated due to extremely long CPU times. The dashed line is a guide to the eye for linear dependence

computational efficiency substantially increases. As indicated above for $tr = 0.10$, for small tr OK-U is the fastest, followed by MPRS and OK-R. However, for $tr = 0.8$ MPRS is already about 16 (7) times faster than OK-U (OK-R). The computational complexity of OK-U increases cubically with sample size. The CPU time of OK-R tends to increase with increasing tr . In contrast, the MPRS computational cost only depends on P (# of prediction points), which decreases with increasing tr . The fit in Fig. 3e indicates an approximately linear decrease.

Next, we evaluate the relative MPRS prediction performance with increasing data roughness, i.e., gradually

decreasing ν . In Fig. 4 we present the ratios of different calculated measures (errors) obtained by (a,b) MPRS and OK-U and (c,d) MPRS and OK-R for (a,c) $tr = 0.33$ and (b,d) 0.66, respectively. The plots exhibit a consistent decrease (increase) of relative MPRS vs OK-U errors (correlation coefficient) with decreasing smoothness from $\nu = 0.5$ to 0.1. At $\nu \approx 0.3$ the MPRS and OK validation measures become approximately identical, and for $\nu \lesssim 0.3$ MPRS outperforms OK-U. The MPRS vs OK-R errors show similar behavior with the cross-over value of ν where MPRS outperforms OK-R shifting slightly to smaller values. Thus, MPRS seems to be more appropriate than OK for the interpolation of rougher data.

The above cases assume a Gaussian distribution, which is not universally observed in real-world data. To assess MPRS performance for non-Gaussian (skewed) distributions, we simulate synthetic data that follow the lognormal law, i.e., $\log Z \sim N(m = 0, \sigma)$ with the WM($\kappa = 0.2, \nu = 0.5$) covariance. The lognormal random field that generates the data has median $z_{0.50} = \exp(m) = 1$ and standard deviation $\sigma_Z = [\exp(\sigma^2) - 1]^{1/2} \exp(m + \sigma^2/2)$. Thus, σ controls the data skewness (see the inset of Fig. 5). Figure 5 demonstrates that MPRS can provide better interpolation performance compared to OK-U for non-Gaussian, highly skewed data as well. In particular, for $\sigma \gtrsim 1$ the MAE and MARE measures of MPRS become comparable or smaller than those obtained with OK. On the other hand, OK-R delivers better performance than OK-U, particularly for the highly skewed datasets. While one still can observe some relative improvement of MPRS vs OK-R performance in terms of decreasing MAE and MARE errors with σ , OK-R delivers superior performance for the entire range of tested σ values, except for MARE at $\sigma = 1.5$.

Table 4 Interpolation validation measures (obtained from Table 2 for $V = 1$) for the MPRS, OK-U and OK-R methods applied to the routine and emergency SIC2004 datasets

Data	Method	AE	ARE (%)	RSE	R (%)	t_{cpu} (s)
Routine	MPRS	9.22	9.29	12.58	78.07	0.03
	OK-U	9.29	9.36	12.84	77.03	0.03
	OK-R	9.39	9.45	12.96	76.56	0.05
Emergency	MPRS	17.75	11.85	76.76	40.40	0.02
	OK-U	22.05	18.74	74.19	47.05	0.02
	OK-R	21.52	16.28	78.13	48.24	0.04

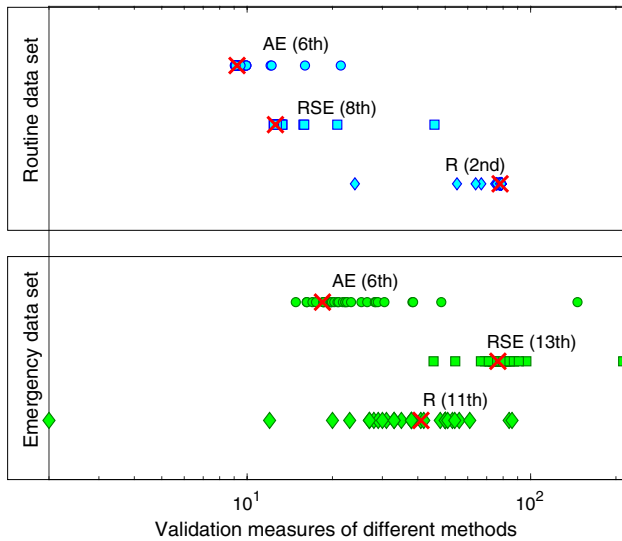


Fig. 7 Values of AE, RSE and R obtained for the routine and emergency datasets by means of 31 interpolation methods reported in the SIC2004 exercise (circles, squares and diamonds) and the MPRS approach (red crosses). The numbers in parentheses denote the ranking of MPRS performance for the particular validation measure with respect to all 32 methods

Finally, we assess the computational complexity of MPRS by measuring the CPU time necessary for interpolation of $[(1 - tr)N]$ points based on $[trN]$ samples, for increasing N between 2^{10} and 2^{20} . The results presented in Fig. 6 for $tr = 0.33$ and 0.66 confirm approximately linear (sublinear for smaller N) dependence, already suggested in Fig. 3c. The CPU times obtained for $tr = 0.66$, which involve more samples but fewer prediction points, are systematically smaller. For comparison, we also included the corresponding CPU times of the OK-R method.³ Even though it considerably alleviates the computational cost of OK-U, its scaling with data size is inferior to MPRS. This can be ascribed to the computationally demanding variogram calculation, which shows quadratic scaling with the sample size. Thus, while for $N \approx 10^3$ MPRS is faster than OK-R only several times, for $N \approx 10^5$ MPRS is faster several hundreds of times.

³ Due to computational inefficiency and high memory demands we did not include the OK-U method.

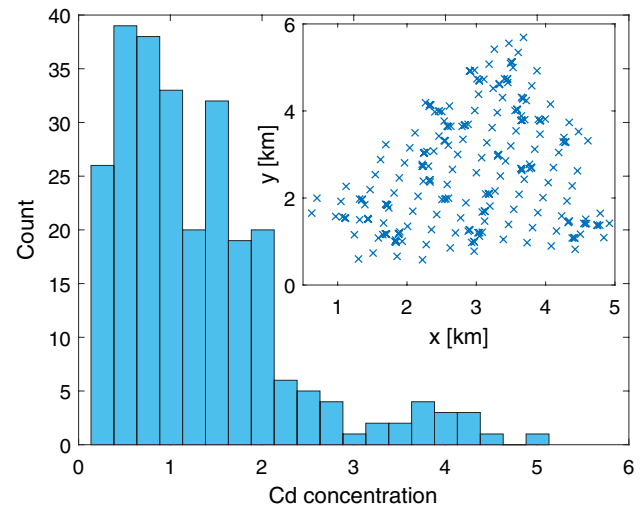


Fig. 8 Histogram of cadmium soil contamination concentrations from the Jura dataset. The inset shows the measurement locations

Furthermore, the gap between $tr = 0.33$ and $tr = 0.66$ curves for OK-R increases with N , thus making OK-R relatively inefficient for datasets with a large number of observations.

4.2 Real 2D spatial data

4.2.1 Ambient gamma dose rates

The first two datasets represent radioactivity levels in the routine and the simulated emergency scenarios (Dubois and Galmarini 2006). In particular, the routine dataset represents daily mean gamma dose rates over Germany reported by the national automatic monitoring network at 1,008 monitoring locations. In the second dataset an accidental release of radioactivity in the environment was simulated in the South-Western corner of the monitored area. These data were used in Spatial Interpolation Comparison (SIC) 2004 exercise to test the prediction performance of various methods.

The training set involves daily data from 200 randomly selected stations, while the validation set involves the remaining 808 stations. Data summary statistics and an extensive discussion of different interpolation

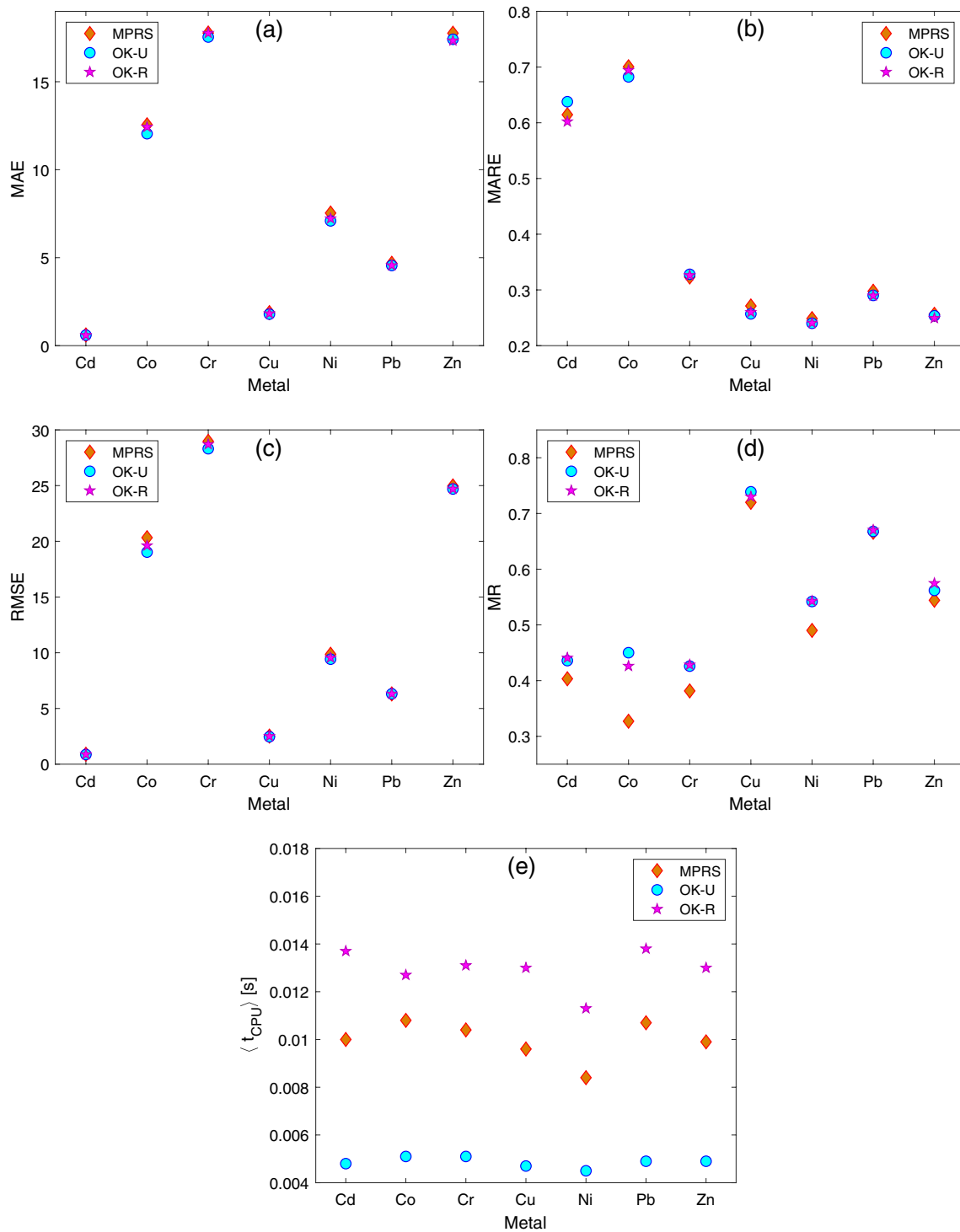


Fig. 9 MPRS, OK-U and OK-R validation measures for the Jura heavy metal contamination dataset. The measures are calculated from 100 randomly chosen training sets of 85 points leaving 174 points in the validation sets

approaches are found in Dubois and Galmarini (2006). In total, 31 algorithms were applied. Several geostatistical techniques failed in the emergency scenario due to instabilities caused by the outliers (simulated release data).

The validation measures from MPRS, OK-U and OK-R applied to both the routine and emergency datasets are presented in Table 4. Comparing the results for OK and MPRS, for the routine dataset MPRS gives slightly better

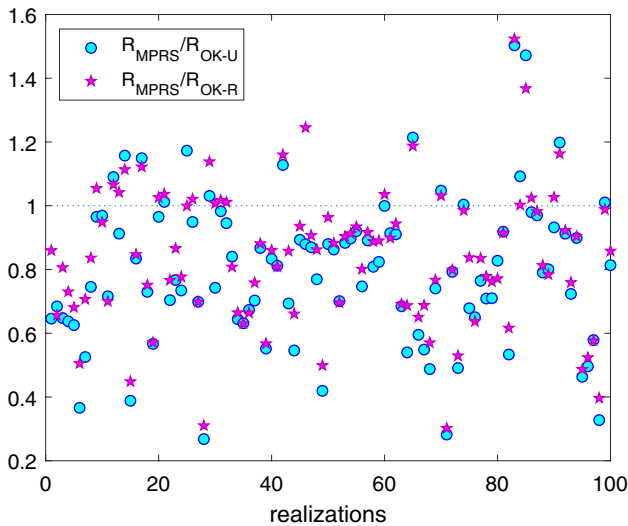


Fig. 10 Ratio of the correlation coefficients, $R_{\text{MPRS}}/R_{\text{OK-U}}$ and $R_{\text{MPRS}}/R_{\text{OK-R}}$, between validation data and predicted values for Co concentration for 100 random validation splits

results than either of the OK approaches. However, for the emergency dataset the differences are substantial. In particular, AE and ARE errors are much smaller for MPRS, while OK methods give superior values for R . The RSE error is smallest for OK-U, followed by MPRS and OK-R. The CPU time of MPRS and OK-U are almost identical, while OK-R is almost two times slower.

In Fig. 7 we compare the MPRS performance with the results obtained with the 31 different approaches reported in Dubois and Galmarini (2006). This comparison shows that MPRS is competitive with geostatistical, neural network, support vector machines and splines. In particular, for the routine dataset MPRS ranked 6th, 8th and 2nd for AE, RSE and R , respectively, and for the emergency data 6th, 13th and 11th.

Table 5 Interpolation validation measures for MPRS and OK. The measures are based on 100 randomly chosen training sets that include tr N of the $N = 40,000$ points (Walker lake dataset)

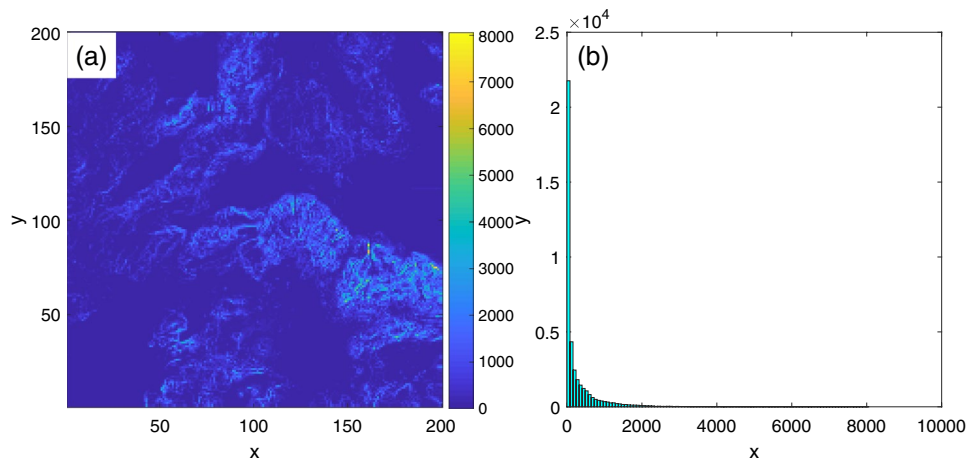
tr	Method	MAE	RMSE	MR (%)	$\langle t_{\text{cpu}} \rangle$ (s)
0.33	MPRS	166.08	334.81	74.52	0.66
	OK-U	167.74	340.42	73.97	666.32
	OK-R	163.06	340.43	73.90	13.75
0.66	MPRS	156.03	314.97	77.65	0.71
	OK-U	155.49	319.95	77.24	4999.37
	OK-R	150.25	318.51	77.36	91.33

4.2.2 Jura dataset

This dataset comprises topsoil heavy metal concentrations (in ppm) in the Jura Mountains (Switzerland) (Atteia et al. 1994; Goovaerts 1997). In particular, the dataset includes concentrations of the following metals: Cd, Co, Cr, Cu, Ni, Pb, Zn. The 259 measurement locations and the histogram of Cd concentrations, as an example, are shown in Fig. 8. The detailed statistical summary of all the datasets can be found in Atteia et al. (1994); Goovaerts (1997).

For each dataset we generate $V = 100$ different training sets consisting of 85 randomly selected points. Different panels in Fig. 9 compare MPRS, OK-U and OK-R validation measures for the 7 metal concentrations. In most cases the OK methods give slightly smaller (larger) errors (MR) than MPRS. However, MPRS gives for Cd and Cr concentrations lower MARE values than OK-U; for Cd concentration MAE and RMSE are practically the same for all methods. The differences between MPRS and OK errors are on the order of a few percent. The largest differences appear for mean R : the maximum relative difference, reaching $\sim 30\%$, was recorded for Co. Nevertheless, due to relatively large sample-to-sample fluctuations even in such cases, for certain splits MPRS shows better performance than OK. Figure 10 shows the R ratios per split for the MPRS vs OK-U and

Fig. 11 Walker lake data **a** spatial distribution and **b** histogram



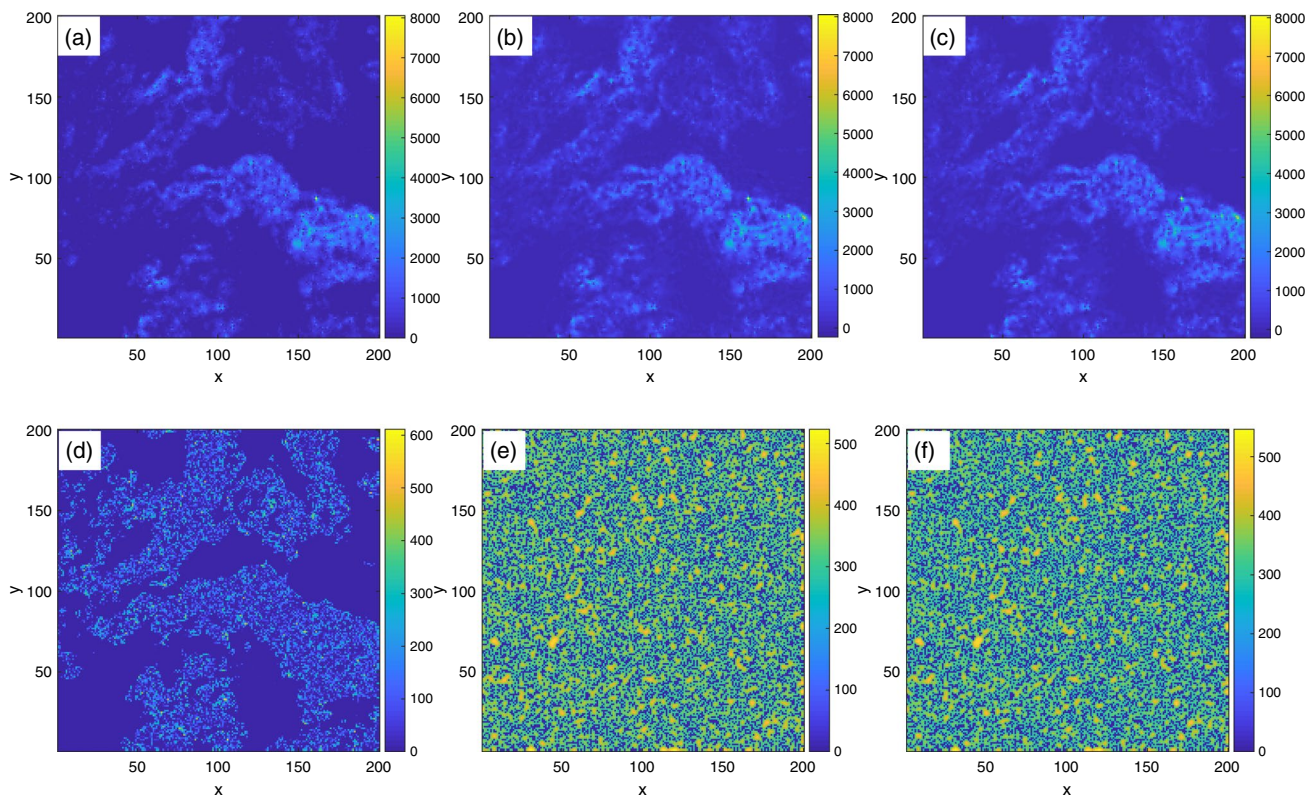


Fig. 12 Visual comparison of interpolated maps for **a** MPRS, **b** OK-U, and **c** OK-R and the corresponding prediction interval width based on **d** the difference between the 84% and the 16% percentiles of the MPRS predictive distribution, and **e**, **f** two kriging standard

deviations for (OK-U, OK-R) respectively. Results are shown for the Walker lake data with $tr = 33\%$. The points with zero width predominantly coincide with the sample locations

MPRS vs OK-R methods. In 15 (22) instances MPRS gives larger values than OK-U (OK-R). The execution times, presented in Fig. 9e, demonstrate that MPRS is slower than OK-U but faster than OK-R (in line with results above for relatively small datasets).

4.2.3 Walker lake dataset

This dataset demonstrates the ability of MPRS to fill data gaps on rectangular grids. The data represent DEM-based chemical concentrations with units in parts per million (ppm) from the Walker lake area in Nevada (Isaaks and Srivastava 1989). We use a subset of the full grid comprising a square of size $L \times L$ with $L = 200$. The summary statistics are: $N = 40,000$, $z_{\min} = 0$, $z_{\max} = 8,054.6$, $\bar{z} = 269.35$, $z_{0.50} = 59.45$, $\sigma_z = 499.43$, *skewness* (s_z) and *kurtosis* (k_z) coefficients $s_z = 3.59$ and $k_z = 22.12$, respectively. The spatial distribution and histogram of the data are shown in Fig. 11.

Training sets of size $[trN]$ are generated by randomly removing $[(1 - tr)N]$ points from the full dataset. For $tr = 0.33$ and 0.66 we generate $V = 100$ different training-validation splits. The validation measures are listed in

Table 5. Due to zero values, the relative MARE errors can not be evaluated. For this highly skewed dataset, MPRS shows slightly better prediction performance than OK-U (except MAE for $tr = 0.66$). Using a search neighborhood (OK-R) improves the kriging performance leading to superior MAE but still inferior RMSE and *MR* compared to MPRS. For this relatively large dataset the MPRS approach is for $tr = 0.33$ (0.66) 20 (128) times faster than OK-R and more than 1,000 (7,040) times faster than OK-U. While not immediately apparent (due to the skewed distribution), a visual comparison of the reconstructed gaps in Fig. 12, shows that the OK generated maps are slightly smoother than the MPRS map.

Next, we consider the performance of the methods with respect to uncertainty quantification. In the case of kriging this is based on Gaussian prediction intervals, i.e., $[\hat{m}_{OK} - z_c \sigma_{OK}, \hat{m}_{OK} + z_c \sigma_{OK}]$, where \hat{m}_{OK} is the conditional mean at the prediction site, σ_{OK} the conditional standard deviation, and z_c the critical value corresponding to a selected confidence level. The MPRS predictive distribution on the other hand strongly deviates from the Gaussian law at many validation points. Therefore, for MPRS it is more appropriate to construct prediction intervals $[\hat{x}_{\alpha/2}, \hat{x}_{1-\alpha/2}]$

Fig. 13 Monthly averaged latent heat release data measured in degrees Celsius per hour; **a** spatial distribution and **b** histogram

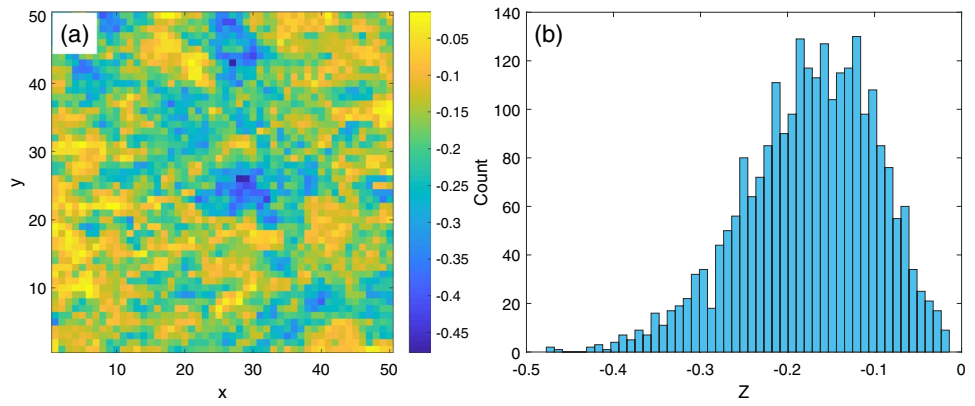


Table 6 Interpolation validation measures for MPRS, OK-U and OK-R based on 100 randomly chosen training-validation splits; the training set includes trN points where $N = 2,500$ (latent heat dataset)

tr	Method	MAE	MARE (%)	RMSE	MR (%)	$\langle t_{cpu} \rangle$ (s)
33%	MPRS	0.042	-33.18	0.053	70.83	0.04
	OK-U	0.041	-31.48	0.052	72.38	0.19
	OK-R	0.041	-31.06	0.052	72.92	0.13
66%	MPRS	0.038	-29.50	0.048	77.43	0.03
	OK-U	0.036	-27.20	0.046	79.28	0.95
	OK-R	0.036	-26.93	0.046	79.79	0.25

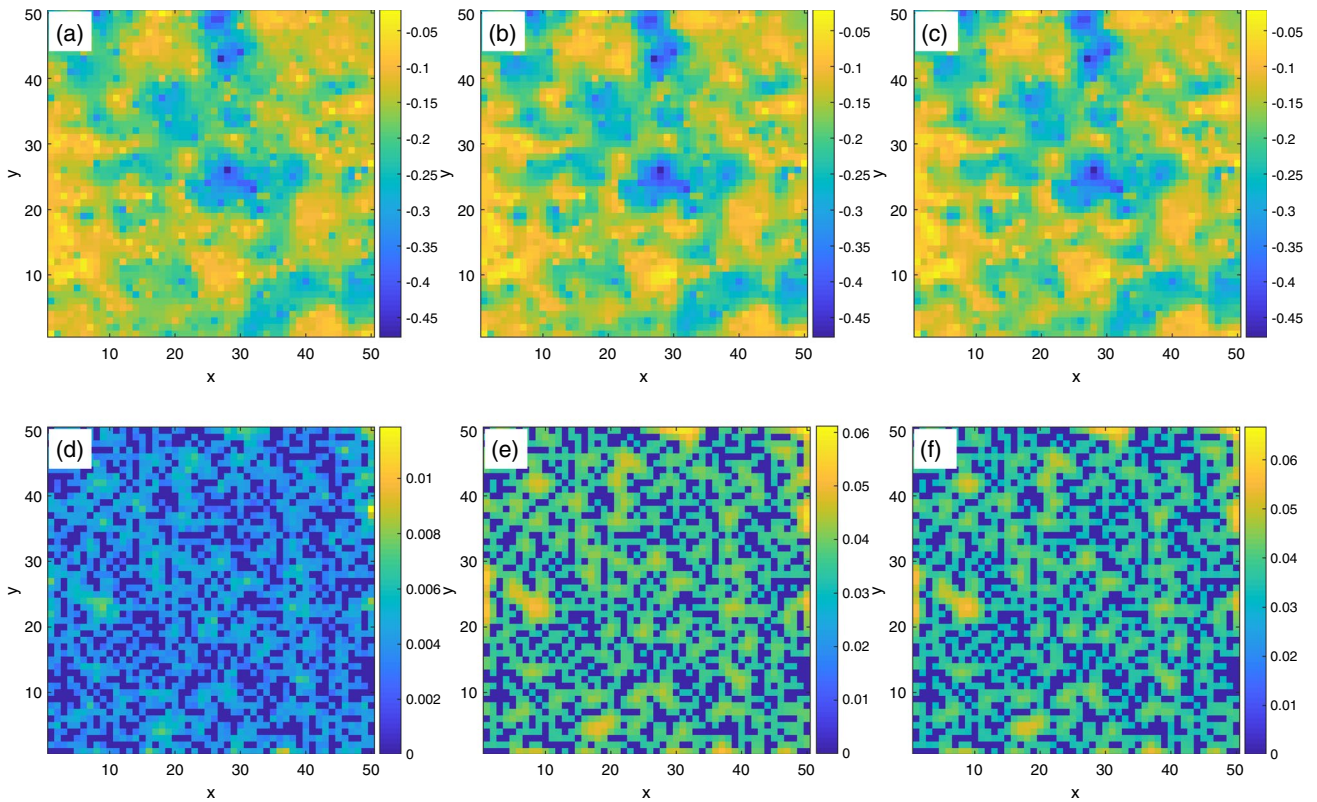


Fig. 14 Visual comparison of **a** MPRS, **b** OK-U and **c** OK-R interpolated maps (latent heat data) and corresponding prediction standard deviations **d-f** for a $tr = 0.33$ training-validation split

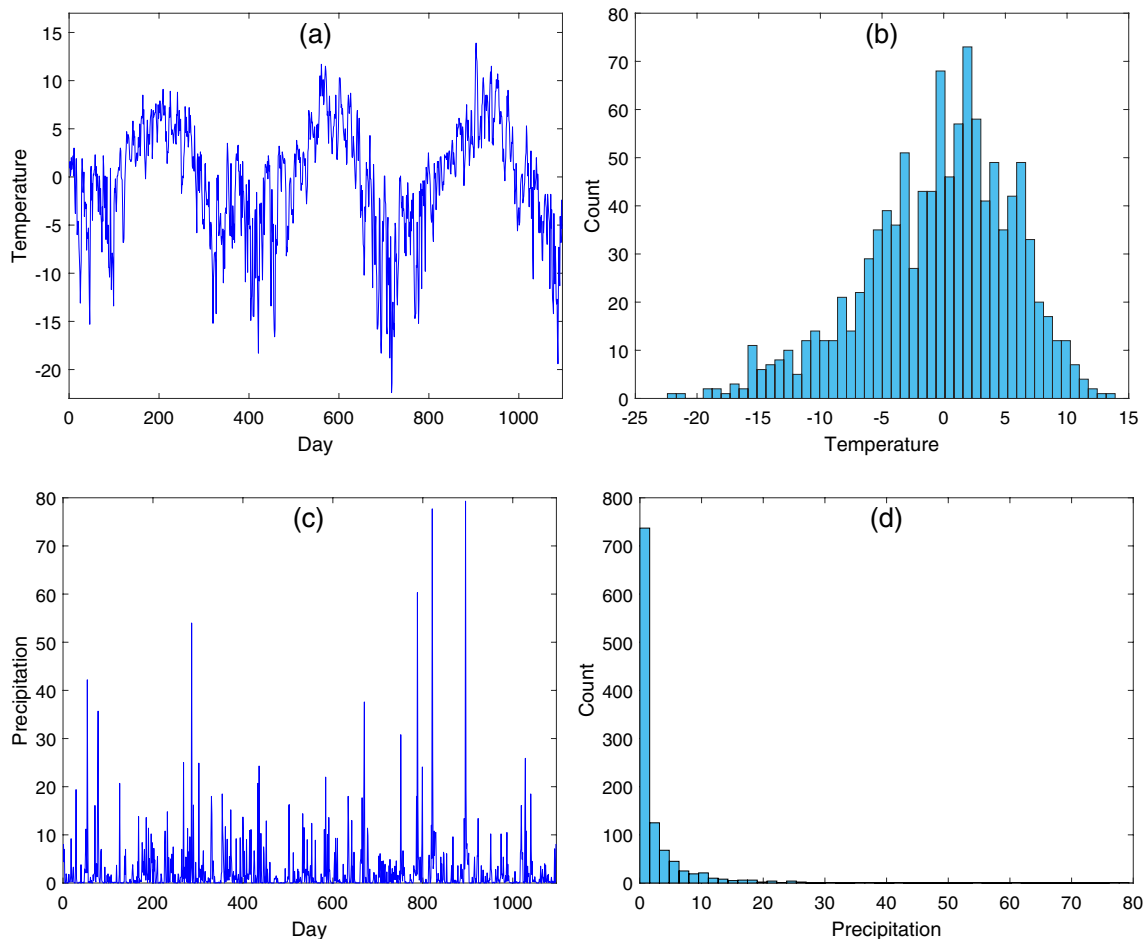


Fig. 15 Time series (a, c) and respective frequency histograms (b, d) of daily temperature (a, b) and precipitation (c, d) at Jökulsá Eystri River (Iceland) for the period between January 1, 1972 and December 31, 1974

based on the percentiles of the predictive distribution that correspond to confidence level $100(1 - \alpha)\%$. In Fig. 12d we present the width of the 68% prediction intervals ($\alpha = 0.32$), based on the 16% and 84% percentiles of the MPRS predictive distribution. The prediction interval width provides the measure of uncertainty. The respective OK uncertainty maps are shown in Figs. 12e–12f. As evidenced in these plots, the MPRS uncertainty exhibits more spatial structure than the OK uncertainty maps. Notably, under MPRS large areas (predominantly those with zero or near-zero values) are assigned much smaller uncertainty in contrast with other areas where the MPRS uncertainty is comparable to OK values.

It is reasonable to ask how successfully the prediction intervals capture the true values at the validation points. For this purpose we evaluate the percent interval coverage (PIC), i.e., the percentage of validation points for which the true value is contained inside the prediction interval. For the Walker lake dataset with $tr = 33\%$ (presented in Fig. 12) we obtain the following average PIC values based on 100

training-validation splits: 80.31% for MPRS, 71.02% for OK-U, and 71.59% for OK-R. We also evaluated the MPRS 95% prediction intervals (based on the 2.5% and 97.5% percentiles) and the $[\hat{m}_{OK} - 1.96\sigma_{OK}, \hat{m}_{OK} + 1.96\sigma_{OK}]$ prediction intervals for OK-U and OK-R. The resulting PIC values in this case are as follows: 95.97% for MPRS, 83.57% for OK-U, and 83.93% for OK-R. Hence, overall we observe that MPRS prediction intervals contain the true values more often than the OK respective intervals. We note that this behavior is not universal: for more symmetric data distributions, OK can outperform the MPRS prediction intervals. However, the MPRS performance is expected to improve by tuning the model, e.g., by increasing the number of realizations at equilibrium.

4.2.4 Atmospheric latent heat release

This section focuses on monthly (January 2006) means of vertically averaged atmospheric latent heat release (measured in degrees Celsius per hour) measurements (Tao et al.

Table 7 Interpolation validation measures for MPRS, OK-U and OK-R based on 100 randomly selected training-validation splits. The training sets include trN points ($N = 1,096$ and $tr = 0.33, 0.66$) from the daily temperature and precipitation time series

Data	tr	Method	MAE	RMSE	MR (%)	$\langle t_{cpu} \rangle$ (s)
Temperature	33%	MPRS	2.27	3.14	85.71	0.03
		OK-U	2.17	3.02	86.27	0.04
		OK-R	2.15	3.00	87.06	0.05
	66%	MPRS	1.77	2.49	91.09	0.02
		OK-U	1.66	2.37	91.99	0.11
		OK-R	1.66	2.37	91.99	0.06
Precipitation	33%	MPRS	3.09	6.85	19.25	0.03
		OK-U	3.24	6.74	20.29	0.04
		OK-R	3.23	6.84	20.11	0.05
	66%	MPRS	2.80	6.06	28.73	0.02
		OK-U	3.03	6.16	26.73	0.11
		OK-R	2.99	6.23	28.59	0.06

Table 8 Interpolation validation measures for the MPRS and IDW methods based on 100 randomly chosen training sets including $tr\%$ of the total number $N = 178$ of points in calcium and magnesium contents in soil samples at the 0–20 cm soil layer

Data	tr	Method	MAE	MARE (%)	RMSE	MR (%)	$\langle t_{cpu} \rangle$ (s)
Ca	33%	MPRS	6.93	14.67	8.95	60.51	0.006
		IDW	7.16	15.60	9.13	59.71	0.001
	66%	MPRS	6.43	13.60	8.38	65.49	0.005
		IDW	6.92	15.15	8.82	64.35	0.001
Mg	33%	MPRS	4.31	17.56	5.46	52.40	0.006
		IDW	4.48	18.54	5.59	49.68	0.001
	66%	MPRS	3.72	15.11	4.80	65.18	0.005
		IDW	4.21	17.23	5.27	60.77	0.001

2006; Anonymous 2011). The $L \times L$ data grid ($L = 50$) extends in latitude from 16 S to 8.5N and in longitude from 126.5E to 151E with cell size $0.5^\circ \times 0.5^\circ$. This area is in the Pacific region and extends over the Eastern part of the Indonesian archipelago. The data summary statistics are as follows: $N = 2,500$, $z_{\min} = -0.477$, $z_{\max} = -0.014$, $\bar{z} = -0.174$, $z_{0.50} = -0.168$, $\sigma_z = 0.076$, $s_z = -0.515$, and $k_z = 3.122$. Negative (positive) values correspond to latent heat absorption (release). The spatial distribution and histogram of the data are shown in Fig. 13.

The comparison of validation measures presented in Table 6 and a visual comparison of the reconstructed maps and prediction uncertainty, shown in Fig. 14, reveals similar patterns as the Walker lake data: the OK predictions show somewhat smoother variation and larger variance than MPRS. However, in this case MPRS displays somewhat worse prediction performance but is significantly more efficient computationally than either of the OK methods.

4.3 Time series (temperature and precipitation)

The MPRS method can be applied to data in any dimension d . We demonstrate that the MPRS method provides competitive predictive and computational performance for time series as well.

We consider two time series of daily data at Jökulsá Eystri River (Iceland), collected at the Hveravellir meteorological station, for the period between January 1, 1972 and December 31, 1974 (a total of $N = 1,096$ observations) (Tong 1990). The first set represents daily temperatures (in degrees Celsius) and the second daily precipitation (in millimeters). The time series and the respective histograms are shown in Fig. 15. The summary statistics for temperature are: $z_{\min} = -22.4$, $z_{\max} = 13.9$, $\bar{z} = -0.441$, $z_{0.50} = 0.3$, $\sigma_z = 6.021$, $s_z = -0.595$, and $k_z = 3.196$. The precipitation statistics are: $z_{\min} = 0$, $z_{\max} = 79.3$, $\bar{z} = 2.519$, $z_{0.50} = 0.3$, $\sigma_z = 6.025$, $s_z = 6.512$, and $k_z = 65.268$. The temperature follows an almost Gaussian distribution, while precipitation is strongly non-Gaussian, highly skewed, with the majority of values equal or close to zero and a small number of outliers that form an extended right tail.

The interpolation validation measures and computational times for MPRS, OK-U and OK-R are listed in Table 7. The results are based on 100 randomly selected training-validation splits which include trN points. For the temperature data, the MPRS performance relative to the OK approaches is similar as for the 2D spatial data that do not dramatically deviate from the Gaussian distribution, such as the latent heat. However, in the case of precipitation MPRS returns a lower MAE than OK for $tr = 0.33$, while for $tr = 0.66$ MPRS

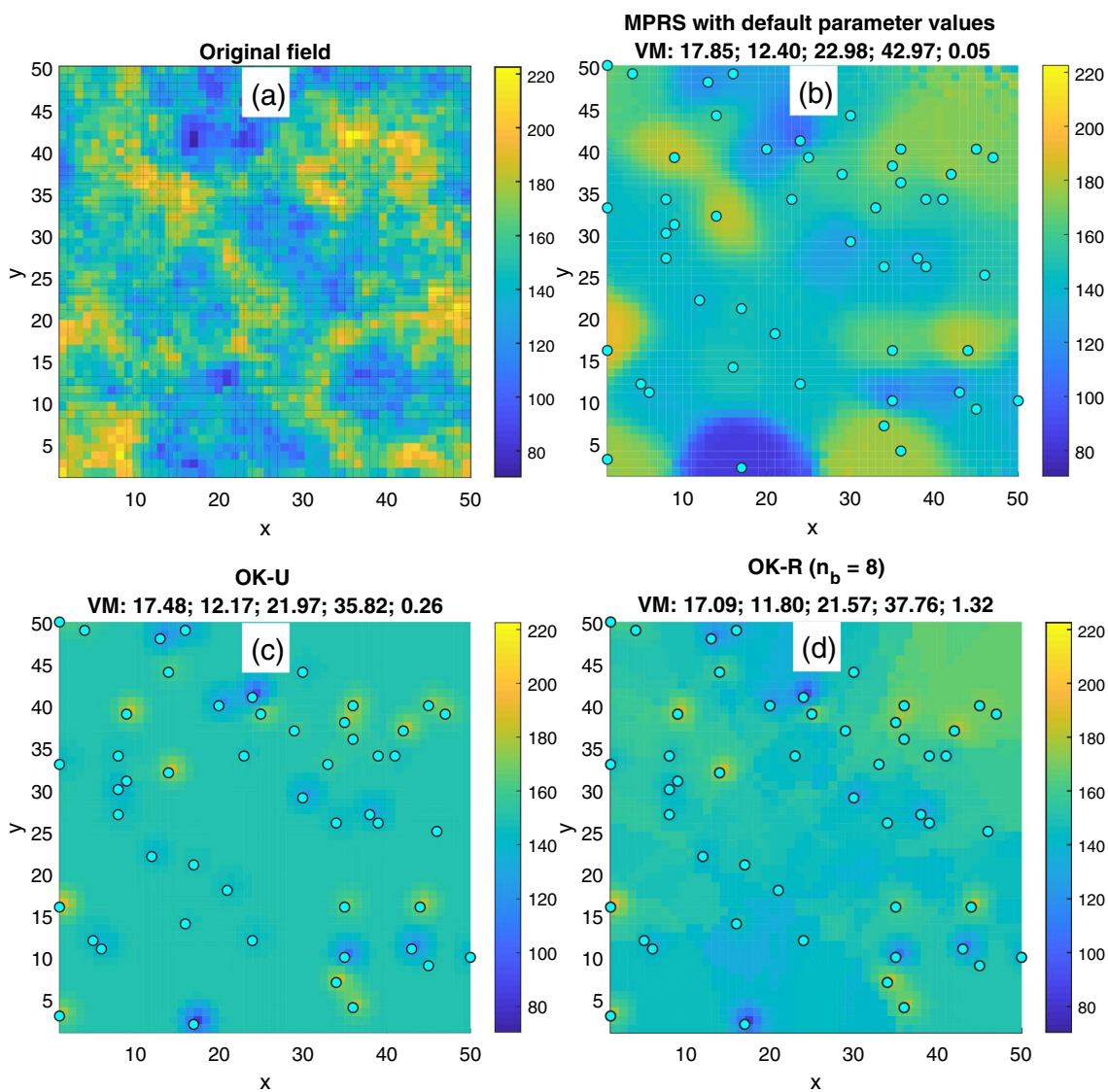


Fig. 16 **a** Synthetic Gaussian data with $m = 150$, $\sigma = 25$ and $WM(\kappa = 0.2, \nu = 0.5)$ covariance. **b–d** MPRS, OK-U, and OK-R predictions on the grid of the size 50×50 , based on 50 (2%) randomly

distributed samples (cyan circles). VM stand for the validation measures MAE, MARE, RMSE, MR , and (t_{cpu}) , respectively

is clearly better for all measures. This observation agrees with the results for the synthetic spatial data, i.e., the relative performance of MPRS improves for strongly non-Gaussian data (cf. Figure 5 which displays relative errors for lognormal data with gradually increasing s_2).

4.4 Real 3D spatial data

Finally, we study calcium and magnesium soil content sampled in the 0–20 cm soil layer (Diggle and Ribeiro Jr 2007). There are $N = 178$ observations and the data are measured in $mmol_c/dm^3$. The calcium data statistics are $z_{\min} = 21$, $z_{\max} = 78$, $\bar{z} = 50.68$, $z_{0.50} = 50.5$, $\sigma_z = 11.08$, $s_z = -0.097$, and $k_z = 2.64$, while for magnesium the respective statistics

are $z_{\min} = 11$, $z_{\max} = 46$, $\bar{z} = 27.34$, $z_{0.50} = 27$, $\sigma_z = 6.28$, $s_z = 0.031$, and $k_z = 2.744$.

In this case we compare MPRS with the IDW method (Shepard 1968) using a power exponent equal to 2 and unrestricted search radius. As evidenced in the validation measures (Table 8), MPRS outperforms IDW in terms of prediction accuracy. The relative differences change from a few percent for $tr = 0.33$ to $\sim 12\%$ for $tr = 0.66$. For this particular dataset, IDW is computationally more efficient than MPRS. However, this is due to the limited data size. With increasing N the relative computational efficiency of MPRS will improve and eventually outperform IDW, since the computational time for the former scales as $\mathcal{O}(P)$, while

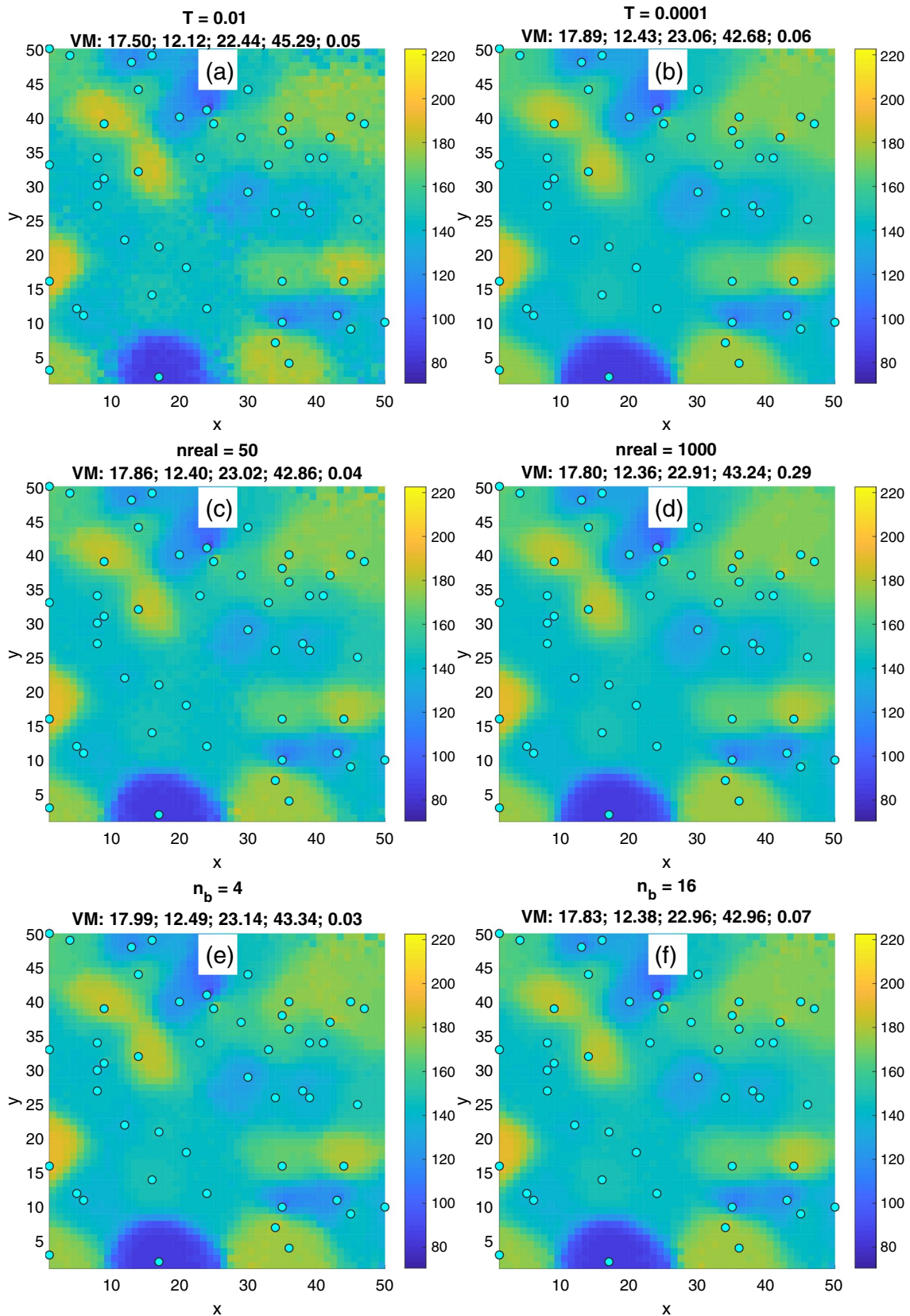


Fig. 17 Visual and numerical comparison of the MPRS predictions with the changed parameters **a**, **b** T , **c**, **d** n_{real} , and **e**, **f** n_b from the default values $T = 0.001$, $n_{real} = 100$, and $n_b = 8$. VM stand for the validation measures MAE, MARE, RMSE, MR , and t_{cpu} , respectively

for the latter as $\mathcal{O}(PN)$ [e.g., see comparison of MPR and IDW (Žukovič and Hristopoulos 2018)].

5 Discussion

The MPRS method involves a number of model parameters and hyperparameters (cf. Table 1). The model parameters are set to reasonable default values which remain constant during the training process. Some of the hyperparameters are dynamically adjusted to secure efficient and autonomous operation. In principle, optimal values for the model parameters and hyperparameters can be determined by selecting a search method and via cross-validation. Nevertheless, the default values presented above (cf. Table 1) deliver reasonable prediction performance in most cases. Below we illustrate how the MPRS prediction and computational performance are affected by changing some default settings.

We use again the synthetic Gaussian data generated from a field with $m = 150$, $\sigma = 25$ and $\text{WM}(\kappa = 0.2, \nu = 0.5)$ covariance, simulated on a 50×50 grid (see Fig. 16a). The samples are produced by randomly choosing 2% of the data, i.e., $tr = 0.02$ corresponding to 50 points. The low sampling density aims to demonstrate how MPRS copes with a lack of conditioning data around the prediction points, and how the MPRS performance is affected by tuning the model. Figure 16 illustrates the reconstructions obtained by (b) MPRS, (c) OK-U, and (d) OK-R methods, along with the calculated validation measures (VM). Compared to MPRS with the default settings, the OK methods provide considerably smoother (mainly OK-U) reconstructed fields with a pronounced averaging effect. They display smaller MAE, MARE and RMSE errors. However, the MPRS correlation coefficient and CPU time are clearly superior.

The MPRS predictions in areas with few observations (see, e.g., the upper right corner in Fig. 16b) display abrupt changes. This artifact is due to the lack of conditioning data (local constraints) close to the prediction points in the target area. Nevertheless, the artifact can be rectified by resetting certain model parameters or hyperparameters, as demonstrated in Fig. 17. Panels (a) and (b) show that the degree of data roughness (due to fluctuations) is naturally proportionate to the temperature. Thus, visually smoother (rougher) predictions can be obtained by decreasing (increasing) T . On the other hand, considering that the original Matérn field with smoothness parameter $\nu = 0.5$ is rather rough, overall better VM are obtained with the higher $T = 0.01$ value.

Similar effects can be achieved by varying the hyperparameter controlling the number of realizations, n_{real} , at thermal equilibrium. The MPRS predictions represent conditional means based on n_{real} estimates obtained from different realizations in thermal equilibrium. Consequently, higher n_{real} implies more precise estimates and smoother

reconstructions, as evidenced in panels (c,d) in Fig. 17. On the down side, increasing n_{real} also implies (linear) increase of the required CPU time.

The number of interacting neighbors per point n_b , is also expected to influence both smoothness and predictive accuracy. In particular, higher n_b implies more bonds between each prediction point and samples in its neighborhood, which should intuitively reduce fluctuations of the simulated states at prediction points leading to smoother prediction maps. At the same time, higher n_b implies interactions with more distant samples; this can be beneficial for capturing longer-range correlations resulting in more precise predictions. Panels (e) and (f) in Fig. 17 show the MPRS prediction maps for $n_b = 4$ and 16. In this case, there are no conspicuous differences in surface smoothness for the two n_b values, but there are differences between the VM, i.e., slightly smaller errors for $n_b = 16$.

We have demonstrated that at some extra computational cost the MPRS prediction performance can be improved by tuning model parameters/hyperparameters, instead of using the default values. Nevertheless, the defaults were employed in all the tests reported herein and produced competitive results with the OK and IDW approaches. We have also shown that for highly skewed distributions MPRS estimates of uncertainty can outperform OK. The sensitivity analysis presented in this section shows that, at least for the studied dataset, model tuning does not lead to dramatic changes. Nevertheless, if computational cost is not an issue, searching for optimal hyperparameters and data-driven adjustment of the MPRS parameters can be further pursued in order to optimize performance. For example, increasing the number of realizations at equilibrium may be necessary to improve the sampling of MPRS predictive distributions.

6 Conclusions

We proposed a machine learning method (MPRS) based on the modified planar rotator for spatial regression. The MPRS method is inspired from statistical physics spin models and is applicable to scattered and gridded data. Spatial correlations are captured via distance-dependent short-range spatial interactions. The method is inherently nonlinear, as evidenced in the energy equations (2) and (5). The model parameters and hyper-parameters are fixed to default values for increased computational performance. Training of the model is thus restricted to equilibrium relaxation which is achieved by means of conditional Monte Carlo simulations.

The MPRS prediction performance (using default settings) is competitive with standard spatial regression methods, such as ordinary kriging and inverse distance weighting. For data that are spatially smooth or close to the Gaussian distribution, standard methods overall show better prediction

performance. However, the relative MPRS prediction performance improves for data with rougher spatial or temporal variation, as well as for strongly non-Gaussian distributions. For example, MPRS performance is quite favorable for daily precipitation time series which involve large number of zeros.

The MPRS method is non-parametric: it does not assume a particular data probability distribution, grid structure or dimension of the data support. Moreover, it can operate fully autonomously, without user input (expertise). A significant advantage of MPRS is its superior computational efficiency and scalability with respect to data size, features that are needed for processing massive datasets. The required CPU time does not depend on the sample size and increases only linearly with the size of the prediction set. The high computational efficiency is partly due to the full vectorization of the MPRS prediction algorithm. Thus, datasets involving millions of nodes can be processed in terms of seconds on a typical personal computer.

Possible extensions include generalizations of the MPRS Hamiltonian (2). For example, spatial anisotropy can be incorporated by introducing directional dependence in the exchange interaction formula (6). Another potential extension is the use of an external polarizing field to generate spatial trends. Such a generalization involves additional parameters and respective computational cost. This approach was applied to the MPR method on 2D regular grids, and it was shown to achieve substantial benefits in terms of improved prediction performance (Žukovič and Hristopulos 2023). Finally, the training of MPRS can be extended to include the estimation of optimal values for the model parameters and hyperparameters. This tuning will improve the predictive performance at the expense of some computational cost.

Funding Open access funding provided by The Ministry of Education, Science, Research and Sport of the Slovak Republic in cooperation with Centre for Scientific and Technical Information of the Slovak Republic. This study was funded by the Scientific Grant Agency of Ministry of Education of Slovak Republic (Grant No. 1/0695/23) and the Slovak Research and Development Agency (Grant No. APVV-20-0150).

Data availability The gamma dose rate, Jura, and Walker lake datasets can be downloaded from https://wiki.52north.org/AI_GEOSTATS/AI_GEOSTATSData/. The latent heat release data can be downloaded from https://disc.gsfc.nasa.gov/datasets/TRMM_3A12_7/summary. The used time series can be downloaded from <https://pkg.yangzhuoranyang.com/tsdl/>. The 3D soil data can be downloaded from <http://www.leg.ufpr.br/doku.php/pessoais:paulojus:mbgbook:datasets>. Our Matlab code is freely downloadable from <https://www.mathworks.com/matlabcentral/fileexchange/135757-mprs-method>. For IDW and for OK interpolation we used the Matlab codes (Tovar 2014) and (Schwanghart 2010) respectively; both were downloaded from the Mathworks File Exchange site.

Declarations

Conflict of interest The authors declare no conflict of interest. All authors certify that they have no affiliations with or involvement in any organization or entity with any financial interest or non-financial interest in the subject matter or materials discussed in this manuscript.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

- Anonymous (2011) TRMM microwave imager precipitation profile L3 1 month 0.5 degree x 0.5 degree V7. https://disc.gsfc.nasa.gov/datasets/TRMM_3A12_7/summary, [NASA Tropical Rainfall Measuring Mission (TRMM)]; Accessed 30 Sept 2008
- Atteia O, Dubois JP, Webster R (1994) Geostatistical analysis of soil contamination in the Swiss Jura. *Environ Pollut* 86(3):315–327
- Cao J, Worsley K (2001) Applications of random fields in human brain mapping. In: *Spatial statistics: methodological aspects and applications*. Springer, pp 169–182
- Cheng T (2013) Accelerating universal kriging interpolation algorithm using CUDA-enabled GPU. *Comput Geosci* 54:178–183. <https://doi.org/10.1016/j.cageo.2012.11.013>
- Christakos G (2012) *Random field models in earth sciences*. Courier Corporation, New York
- Christakos G, Hristopulos D (2013) *Spatiotemporal environmental health modelling: a tractatus stochasticus*. Springer, Dordrecht
- Cressie N (1990) The origins of kriging. *Math Geol* 22:239–252
- Cressie N, Johannesson G (2018) Fixed rank kriging for very large spatial data sets. *J R Stat Soc Ser B (Stat Methodol)* 70(1):209–226. <https://doi.org/10.1111/j.1467-9868.2007.00633.x>
- de Ravé EG, Jiménez-Hornero F, Ariza-Villaverde A et al (2014) Using general-purpose computing on graphics processing units (GPGPU) to accelerate the ordinary kriging algorithm. *Comput Geosci* 64:1–6. <https://doi.org/10.1016/j.cageo.2013.11.004>
- Diggle P, Ribeiro P Jr (2007) *Model-based geostatistics*. Springer series in statistics. Springer, New York
- Dubois G, Galmarini S (2006) *Spatial interpolation comparison (SIC 2004): Introduction to the exercise and overview of results*. Tech. rep, Luxembourg. 92-894-9400-X
- Furrer R, Genton MG, Nychka D (2006) Covariance tapering for interpolation of large spatial datasets. *J Comput Graph Stat* 15(3):502–523. <https://doi.org/10.1198/106186006X132178>
- Goovaerts P (1997) *Geostatistics for natural resources evaluation*. Oxford University Press, New York
- Gui F, Wei LQ (2004) Application of variogram function in image analysis. In: *Proceedings 7th International Conference on Signal Processing, 2004, IEEE*, pp 1099–1102. <https://doi.org/10.1109/ICOSP.2004.1441515>

- Hamzehpour H, Sahimi M (2006) Generation of long-range correlations in large systems as an optimization problem. *Phys Rev E* 73(5):056121. <https://doi.org/10.1103/PhysRevE.73.056121>
- Hohn ME (1988) *Geostatistics and Petroleum Geology*. Computer Methods in the Geosciences. Springer
- Hristopulos D (2003) Spartan Gibbs random field models for geostatistical applications. *SIAM J Sci Comput* 24(6):2125–2162. <https://doi.org/10.1137/S106482750240265X>
- Hristopulos DT (2015) Stochastic local interaction (SLI) model. *Comput Geosci* 85(PB):26–37. <https://doi.org/10.1016/j.cageo.2015.05.018>
- Hristopulos DT (2020) *Random fields for spatial data modeling*. Springer, Dordrecht. <https://doi.org/10.1007/978-94-024-1918-4>
- Hristopulos DT, Elogne SN (2007) Analytic properties and covariance functions for a new class of generalized Gibbs random fields. *IEEE Trans Inf Theory* 53(12):4667–4679. <https://doi.org/10.1109/TIT.2007.909163>
- Hu H, Shu H (2015) An improved coarse-grained parallel algorithm for computational acceleration of ordinary kriging interpolation. *Comput Geosci* 78:44–52. <https://doi.org/10.1016/j.cageo.2015.02.011>
- Ingram B, Cornford D, Evans D (2008) Fast algorithms for automatic mapping with space-limited covariance functions. *Stoch Env Res Risk Assess* 22(5):661–670. <https://doi.org/10.1007/s00477-007-0163-9>
- Isaaks EH, Srivastava MR (1989) *Applied geostatistics*. 551.72 ISA, Oxford University Press
- Kaufman CG, Schervish MJ, Nychka DW (2008) Covariance tapering for likelihood-based estimation in large spatial data sets. *J Am Stat Assoc* 103(484):1545–1555. <https://doi.org/10.1198/016214508000000959>
- Kitanidis PK (1997) *Introduction to geostatistics: applications in hydrogeology*. Cambridge University Press, Cambridge
- Loison D, Qin C, Schotte K et al (2004) Canonical local algorithms for spin systems: heat bath and Hastings's methods. *Eur Phys J B-Condens Matter Complex Syst* 41(3):395–412
- Marcotte D, Allard D (2018) Half-tapering strategy for conditional simulation with large datasets. *Stoch Env Res Risk Assess* 32(1):279–294. <https://doi.org/10.1007/s00477-017-1386-z>
- Metropolis N, Rosenbluth AW, Rosenbluth MN et al (1953) Equation of state calculations by fast computing machines. *J Chem Phys* 21(6):1087–1092. <https://doi.org/10.1063/1.1699114>
- Minasny B, McBratney AB (2005) The Matérn function as a general model for soil variograms. *Geoderma* 128(3):192–207. <https://doi.org/10.1016/j.geoderma.2005.04.003>
- Pardo-Iguzquiza E, Chica-Olmo M (1993) The Fourier integral method: an efficient spectral method for simulation of random fields. *Math Geol* 25:177–217
- Parrott RW, Styzt MR, Amburn P et al (1993) Towards statistically optimal interpolation for 3d medical imaging. *IEEE Eng Med Biol Mag* 12(3):49–59
- Pesquer L, Cortés A, Pons X (2011) Parallel ordinary kriging interpolation incorporating automatic variogram fitting. *Comput Geosci* 37(4):464–473. <https://doi.org/10.1016/j.cageo.2010.10.010>
- Ramani S, Unser M (2006) Mate/spl acute/rn b-splines and the optimal reconstruction of signals. *IEEE Signal Process Lett* 13(7):437–440. <https://doi.org/10.1109/LSP.2006.872396>
- Robert CP, Casella G, Casella G (1999) *Monte Carlo statistical methods*, 2nd edn. Springer, New York. <https://doi.org/10.1007/978-1-4757-4145-2>
- Rossi RE, Dungan JL, Beck LR (1994) Kriging in the shadows: geostatistical interpolation for remote sensing. *Remote Sens Environ* 49(1):32–40
- Rubin Y (2003) *Applied stochastic hydrogeology*. Oxford University Press, Oxford
- Sandwell DT (1987) Biharmonic spline interpolation of GEOS-3 and SEASAT altimeter data. *Geophys Res Lett* 14(2):139–142
- Savitzky A, Golay MJ (1964) Smoothing and differentiation of data by simplified least squares procedures. *Anal Chem* 36(8):1627–1639
- Schwanghart W (2010) Ordinary kriging. <https://www.mathworks.com/matlabcentral/fileexchange/29025-ordinary-kriging>. Accessed 21 Oct 2022
- Shepard D (1968) A two-dimensional interpolation function for irregularly-spaced data. In: *Proceedings of the 1968 23rd ACM National Conference*, pp 517–524. <https://doi.org/10.1145/800186.810616>
- Tao WK, Smith EA, Adler RF et al (2006) Retrieval of latent heating from TRMM measurements. *Bull Am Meteor Soc* 87(11):1555–1572. <https://doi.org/10.1175/BAMS-87-11-1555>
- Tong H (1990) *Non-linear time series: a dynamical system approach*. Oxford University Press, Oxford
- Tovar A (2014) Inverse distance weight function. <https://www.mathworks.com/matlabcentral/fileexchange/46350-inverse-distance-weight-function>, [Online; accessed November 11, 2022]
- Unser M, Blu T (2005) Generalized smoothing splines and the optimal discretization of the Wiener filter. *IEEE Trans Signal Process* 53(6):2146–2159
- Wackernagel H (2003) *Multivariate geostatistics*, 3rd edn. Springer, Berlin Heidelberg
- Winkler G (2003) *Image analysis, random fields and Markov chain Monte Carlo methods: a mathematical introduction, applications of mathematics*, vol 27. Springer, Berlin
- Zhong X, Kealy A, Duckham M (2016) Stream kriging: Incremental and recursive ordinary kriging over spatiotemporal data streams. *Comput Geosci* 90:134–143. <https://doi.org/10.1016/j.cageo.2016.03.004>
- Žukovič M (2009) Hristopulos DT (2009b) Multilevel discretized random field models with ‘spin’ correlations for the simulation of environmental spatial data. *J Stat Mech: Theory Exp* 02:P02023. <https://doi.org/10.1088/1742-5468/2009/02/P02023>
- Žukovič M, Hristopulos DT (2009) Classification of missing values in spatial data using spin models. *Phys Rev E* 80(1):011116. <https://doi.org/10.1103/PhysRevE.80.011116>
- Žukovič M, Hristopulos DT (2013) Reconstruction of missing data in remote sensing images using conditional stochastic optimization with global geometric constraints. *Stoch Env Res Risk Assess* 27(4):785–806. <https://doi.org/10.1007/s00477-012-0618-5>
- Žukovič M, Hristopulos DT (2018) Gibbs Markov random fields with continuous values based on the modified planar rotator model. *Phys Rev E* 98(6):062135. <https://doi.org/10.1103/PhysRevE.98.062135>
- Žukovič M, Hristopulos DT (2023) Spatial data modeling by means of Gibbs-Markov random fields based on a generalized planar rotator model. *Physica A* 612:128509. <https://doi.org/10.1016/j.physa.2023.128509>
- Žukovič M, Borovský M, Lach M et al (2020) GPU-accelerated simulation of massive spatial data based on the modified planar rotator model. *Math Geosci* 52(1):123–143. <https://doi.org/10.1007/s11004-019-09835-3>

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.