A spatially adaptive multi-resolution generative algorithm: Application to simulating flood wave propagation

Julie Carreau, Philippe Naveau

Weather and Climate Extremes 41 (2023) 100580

Contents lists available at ScienceDirect

Weather and Climate Extremes

journal homepage: www.elsevier.com/locate/wace

A spatially adaptive multi-resolution generative algorithm: Application to simulating flood wave propagation

Julie Carreau [a,*], Philippe Naveau [b]

[a] Department of Mathematics and Applied Engineering, Polytechnique Montreal, Montreal, Canada
[b] Laboratoire des Sciences du Climat et de l'Environnement, Centre National de la Recherche Scientifique, Gif-sur-Yvette, France

ARTICLE INFO

Keywords:
Urban flood hazard
Extreme water depths and discharges
High resolution 2D meshes
Spatio-temporal lifting scheme
Spatial partitioning
Spatial pattern types
Feed-forward neural network

ABSTRACT

We propose a statistical model suitable for large spatio-temporal data sets exhibiting complex patterns such as simulated by physics-based hydraulic models over high resolution (HR) 2D meshes. Although necessary for impact studies such as urban flood hazard assessment, their long computation times limit their applicability leading to the development of statistical models that may emulate them quickly. Our model draws from the strengths of multi-resolution analysis and relies on an extension of the lifting scheme, a flexible implementation of discrete wavelet transforms, for spatio-temporal data. The extended lifting scheme exploits the idea that dominant spatial features, that may be identified with clustering, remain present through time. An easily interpretable non-parametric representation can be derived from the lifting scheme by combining a smoothed version of the data (obtained by simple averaging) with details (given by local regression residuals). A generative algorithm is built by introducing the information provided by a low resolution model, whose computation times are orders of magnitude smaller, yielding a downscaling model. This downscaling model assumes that sufficiently representative HR spatial patterns can be inferred from the training set. Our model is applied to a 2D dam break experiment using a synthetic urban configuration and to a field-scale test case simulating the propagation of a dike break flood wave into a Sacramento urban area. A comparison, carried out with spatial interpolation schemes and with a variant of our model based on principal component analysis, shows that the spatio-temporal lifting scheme based model is better at reproducing extreme events.

1. Introduction

Many natural phenomena such as meteorological or flow variables may be simulated by numerically solving physics equations. To fully capture important non-linear spatial features, meshes with high spatial resolution are required leading to very demanding computations. For instance, Mediterranean heavy precipitation is rare but impactful and may be modeled over a restricted domain with a convection-permitting regional model having a spatial resolution of 2.5 km (Caillaud et al., 2021). Another example are water level and horizontal flow velocity useful for urban flood hazard assessment that may be simulated with 2D shallow water models typically operating over meshes with tens or hundreds of thousands of discrete cells (Sanders and Schubert, 2019). Statistical models that can mimic the physics-based simulations of these natural phenomena with fast computation times are an active area of research.

The complex spatial patterns that may be exhibited by some natural phenomena invalidate standard statistical approaches and call for the development of more advanced methods. For instance, Nychka et al. (2018) proposed a convolution-based model that can capture non-stationary features in the covariance of a gaussian process to emulate local surface temperature. Surpassing the gaussian paradigm which might be restrictive for some variables, other numerous examples are provided by deep learning approaches that have attracted recently a lot of attention. As an example, Wu et al. (2021) rely on the generative adversarial network algorithm to simulate high resolution temperature in urbanized areas. However, the implementation of deep learning approaches is, in general, rather involved and requires specific domain expertise and computation technology such as GPU. In addition, the modeling mechanisms lack interpretability which is one of the reason behind the emergence of the hybrid field of physics-informed machine learning (Kashinath et al., 2021). A different less followed route focuses on semi- or non-parametric representations that rely on low dimensional embedding or feature space.

One such representation is provided by Principal Component Analysis (PCA). By assuming that the data lay on a lower dimensional hyperplan, PCA may be thought of as a multivariate gaussian model

* Corresponding author.
E-mail address: julie.carreau@polymtl.ca (J. Carreau).

https://doi.org/10.1016/j.wace.2023.100580
Received 30 March 2023; Received in revised form 29 May 2023; Accepted 2 June 2023
Available online 9 June 2023

with the covariance matrix depending on the eigenvectors that span the hyperplan (e.g., see the definition of probabilistic PCA in Bishop (2006)). PCA represents any given spatial pattern as a linear combination of the eigenvectors which can be thought of as typical spatial patterns. By combining probabilistic PCA distributions into mixtures, more flexible non-linear non-gaussian representations can be obtained (Bishop, 2006). Powerful alternatives to represent complex signals can be derived from the multi-resolution analysis framework. In particular, discrete wavelet transforms have long been used in signal analysis to model highly intricate patterns (Jensen and La Cour-Harbo, 2001). Any given signal, this may be a time series in 1D or an image in 2D, may be decomposed into a sum of scaling coefficients (a form of low pass filter that extracts the main features of the signal) and wavelet coefficients (a form of high pass filter that contains the details of the signal). Of particular interest is the lifting scheme, a simple and intuitive procedure that allows to adapt discrete wavelet transforms to irregular sampling locations, such as the irregular meshes used for hydraulic simulations, and non-dyadic sample sizes (e.g., see Park and Oh (2022)). Proposals to adapt the lifting scheme in order to handle 3D data such as video stream were developed (Zheng and Xue, 2009). Nevertheless, to our knowledge, no adaptation of the lifting scheme was proposed for spatio-temporal data such as physics-based simulations of natural phenomena. Furthermore, some additional generative mechanisms must be incorporated into these representations in order to sample temporally coherent simulations.

The so-called downscaling framework in which low resolution physics-based simulations are used to recover the high resolution ones can be seen as a way to turn a given representation into a generative algorithm. Despite that statistical downscaling has been developed for several decades in the context of climate change studies (Ayar et al., 2016), approaches that can explicitly take into account the spatial structure are relatively recent. For instance, in Cannon (2018), spatial dependence was modeled through iterative random projections of fields of climate variables onto linear one-dimensional sub-spaces. In Vrac and Friederichs (2015) and Vrac (2018), spatial patterns from the training set, defined based on ranks, were re-sampled as is, i.e., this approach boils down to using a very high dimensional empirical copula. More recent statistical downscaling models make use of deep learning algorithms (Wu et al., 2021; François et al., 2021). PCA or discrete wavelet transforms are generally applied as a way to compress the low resolution physics-based simulations, a form of pre-processing step to reduce the dimensionality of the inputs to the downscaling model (Ayar et al., 2016; Kumar et al., 2021). The use of semi- or non-parametric representations as a model for high resolution fields in the downscaling framework is not very common. Carreau and Guinot (2021) and Fraehr et al. (2022) used the PCA representation to capture the spatial structure of high resolution hydraulic fields. Although the setting is slightly different, in both cases, the PCA eigenvectors are computed on the high resolution fields in the training set and are subsequently assumed to be fixed. The goal of downscaling is then to predict the weights of the linear combination of eigenvectors based on information drawn from the low resolution hydraulic fields. Concerning the discrete wavelet transform, Nourani et al. (2020) developed an approach for the spatial disaggregation of rainfall based on the distributional properties of the 2D wavelet coefficients.

In this work, we propose an extension of the lifting scheme which is suitable for the spatio-temporal physics-based simulations of natural phenomena. This allows us to derive non-parametric representations for spatio-temporal data that can be turned into generative algorithms given appropriate mechanisms, see Section 2. The proposed spatio-temporal lifting scheme and the resulting representations rely on very simple calculations – subtractions and additions – whose implementation may run on an ordinary laptop. The powerful non-linear feature extraction capabilities arise from easily interpretable mechanisms. We illustrate the potential of the proposed spatio-temporal lifting scheme as a spatially adaptive multi-resolution generative algorithm by designing a downscaling model, see Section 3. The main idea behind the downscaling strategy is to learn a bank of high resolution spatial pattern types by applying the spatio-temporal lifting scheme on training data. The most relevant pattern is selected by exploiting information drawn from the low resolution physics-based simulations. The downscaling model that relies on the spatio-temporal lifting scheme representation is compared to direct interpolations of the low resolution simulation and to a downscaling model which is similar in spirits to the proposed model but uses PCA to learn spatial pattern types. Water depth and unit discharge norm fields simulated with 2D shallow water models serve for the evaluation. Two configurations are considered: a synthetic urban configuration made of a periodic layout of building blocks which is used to simulate a 2D oblique dam break problem and a field-scale test case consisting of a Sacramento urban area over which the propagation of a dike break flood wave is simulated. We conclude in Section 4.

2. Spatially adaptive multi-resolution representations

The water depth data illustrated in Fig. 1 is used throughout this section to illustrate the spatio-temporal lifting scheme. It is a subset of a simulation obtained by solving the two dimensional shallow water equations for a synthetic urban configuration termed the oblique dam break problem without friction over a high resolution mesh (see Carreau and Guinot (2021)). The complete geometry of this configuration is a straight street of length 1000 m made by concatenating twenty intersections. An intersection consists of a horizontal and a vertical segment of street forming the shape of a cross with identical fictitious buildings at each of the four corners of the intersection. The mesh for a single intersection contains 2304 cells organized regularly. The subset shown in Fig. 1 concerns an intersection in the middle of the street. The water is initially at rest, see Fig. 1(a), and the flow is subsequently oriented in an oblique manner. The data is sampled at 5 s time intervals for a simulation length of 75 s or 16 time steps.

2.1. 1D lifting scheme

Before explaining our version of the lifting scheme for the spatio-temporal framework, we need to recall the basic steps of the conventional implementation in 1D (Sweldens, 1998; Daubechies and Sweldens, 1998). There are various ways to describe the multi-resolution scheme, in particular from a mathematical perspective (Daubechies, 1992) or from a signal processing point of view (Mallat, 1999). In our context, we prefer to highlight the statistical elements of the lifting scheme as this will facilitate the presentation of our extension to the spatio-temporal framework. A key element in our approach is to view the wavelet transform as a simple local regression. Throughout this work, we will consider only the most naive form of regression whereby the response value is estimated directly by the value of the predictor.

For a given 1D signal, one can expect that the value observed at a given time could be well approximated by the value observed immediately before or after. The lifting scheme is a recursive three step algorithm – termed split, predict and update steps – that exploits this idea, see Fig. 2. Let $\{s_i\}_{i=1}^{n}$ be a 1D signal with $n = 2^m$, i.e., its size is assumed to be dyadic for simplicity although this restriction will be lifted in Section 2.2. The split step, the first of the three steps of the lifting scheme, consists in partitioning the initial signal into two subsets of equal size with $n/2 = 2^{m-1}$ elements. These subsets can be thought of as the *response* subset and the *predictor* subset respectively by analogy with the regression framework. Conventionally, the predictor subset contains the elements with even indices, i.e., $\{s_2, s_4, s_6, \dots\}$, while the response subset contains the elements with odd indices, i.e., $\{s_1, s_3, s_5, \dots\}$. Note that the triviality of this splitting step is often overlooked in the 1D case. However, its implementation in a multivariate or spatial context brings a few challenges, see Section 2.2, as the definition of odd and even indices only makes sense for a signal

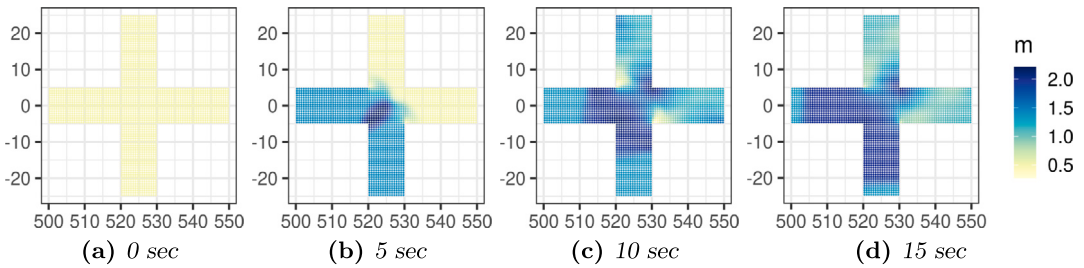


Fig. 1. Water depth in meters (Carreau and Guinot, 2021). Each panel provides a snapshot for the four first time steps (in seconds) out of 16 in total. The mesh contains 2304 cells. The x- and y-axis coordinates are in meters.

which has a single dimension. In this work, instead of focusing on the creation of two subsets, we will most often think of the split step as a way to determine pairs of predictor and response elements (x_k, y_k), with $1 \leq k \leq n/2$, from the elements of the initial signal, i.e., $x_k = s_{i_k}$ and $y_k = s_{j_k}$ for some $1 \leq i_k, j_k \leq n$, not necessarily following even and odd indices.

The second step of the lifting scheme, see Fig. 2, is the predict step that implements the aforementioned naive local regression. More precisely, for each pair of predictor and response elements, x_k is used as an approximation to y_k. The prediction errors or regression residuals yield the wavelet coefficients:

$$d_k = y_k - x_k \quad 1 \leq k \leq n/2 \tag{1}$$

which are then simply stored subsequently. Since wavelet coefficients are regression residuals, by construction, they should be centered around zero and, except near non-linearities in the signal, most residuals should be viewed as noise. In contrast, the remaining predictor elements x_k have to be updated to contain the same overall amount of information than the original signal. This is the role of the update step, the third step of the lifting scheme, which computes the scaling coefficients:

$$c_k = \frac{x_k + y_k}{2} = x_k + \frac{1}{2} d_k \quad 1 \leq k \leq n/2. \tag{2}$$

The update step ensures that the mean value of the scaling coefficients corresponds to the mean value of the initial signal. The set of scaling coefficients $\{c_k\}_k$ of length $n/2$ can be considered as a similar but smoother and shorter version of the original signal from which the highest frequency has been removed.

Computationally, the three lifting steps are local, fast and simple. They can be easily repeated which brings the idea of multi-resolution. At the second iteration of the lifting scheme, the scaling coefficients from the update step of the preceding iteration become the input, i.e., the current signal of length $n/2$, on which the three step of the lifting scheme are applied (see Fig. 2). The decomposition will yield $n/4$ wavelet coefficients (or residuals) and $n/4$ scaling coefficients (or a smoothed signal). The number of iterations, say m, is coined the resolution level, ranging from $m = 0$ where the current signal is the original signal of length n to $m = [\log n / \log 2]$ where a single value remains representing the average of the original signal. As m increases, the current signal provided by the scaling coefficients becomes coarser and coarser. In addition, large wavelet coefficients indicate local discontinuities associated with a given resolution level.

The decomposition of the lifting scheme can be reversed, i.e., the signal may be reconstructed, by reversing iteratively the steps update, predict and performing a merge step (the inverse of the split step). With the naive local regression, the update and predict steps in the inverse scheme are obtained by switching additions with subtractions and vice versa in (1) and (2) yielding:

$$x_k = c_k - \frac{1}{2} d_k \tag{3}$$

$$y_k = d_k + x_k, \tag{4}$$

with $1 \leq k \leq n/2$. In the merge step, the results of the reverse update and predict steps from (3) and (4) are combined to re-form the predictor subset of the preceding iteration.

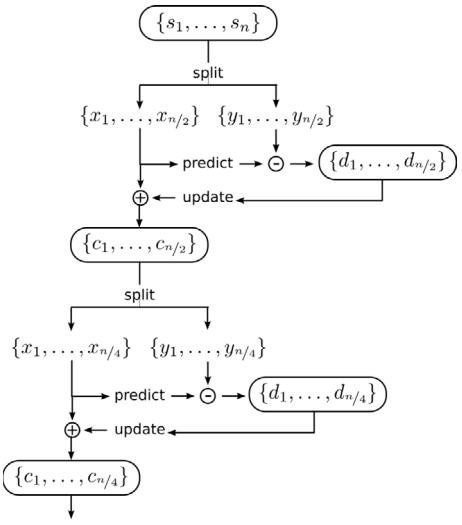
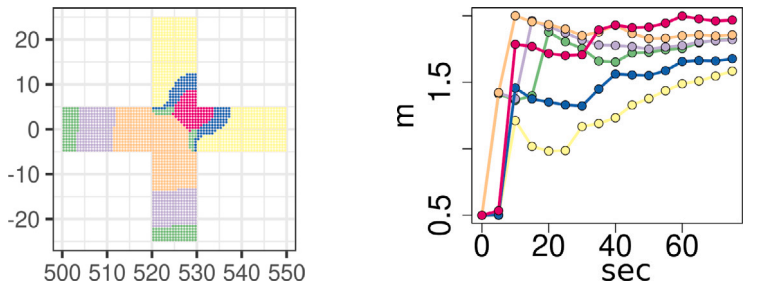


Fig. 2. Split, predict and update steps of the lifting scheme applied iteratively.

2.2. Spatio-temporal lifting scheme

The lifting scheme has been extended to 2D, in particular in the context of image analysis. Initial 2D schemes involved the application of two 1D schemes, a vertical one, along the x-axis, and a horizontal one, along the y-axis (Dong et al., 2008). However, image features are not accounted for and these separable 2D schemes may generate high prediction errors, yielding high wavelet coefficients, non necessarily related to non-linearities in the signal. Non-separable lifting schemes have been developed in which several additional directions are considered (Dong et al., 2008; Kaaniche et al., 2010). Although 3D extensions proposed to handle video streams (see Zheng and Xue (2009) for a review) could be applied to spatio-temporal data, physics-based simulations of natural phenomena such as the water depth data shown in Fig. 1 have certain spatial features that persist in time (for instance, the oblique propagation of the water) that can be exploited in the lifting scheme.

To account for the features of spatio-temporal data, we introduce two main modifications to the conventional lifting scheme. First of all, elements are defined as vectors that contain the whole time series associated to a given location. In what follows, bold font is used to indicate vector elements, e.g., $\boldsymbol{s}_i$ is a vector containing the time series at location i which corresponds to a given cell of the mesh in the water depth data example. Second, a spatial partition, obtained by clustering the elements into groups, is used to assist the split at each stage as explained below. Indeed, clustering has long been used to perform image segmentation and to identify features (Bishop, 2006). Clustering is fast to run and can help quite straightforwardly to reduce prediction errors, thereby reducing the number of significant wavelet coefficients (see (1)), by selecting pairs of predictor-response elements within the



(a) *Partition.* (b) *Cluster representatives.*

Fig. 3. Six spatial features extracted by clustering the 2304 elements based on their temporal behavior (in seconds) for the water depth data of Fig. 1.

same cluster (that are thus part of the same feature). To the best of our knowledge, clustering has not been combined with lifting schemes in such a way.

2.2.1. Clustering assisted split step

To extract the spatial features of the water depth data from Fig. 1, a clustering algorithm may be applied on the 2304 elements (one element for each cell of the mesh). The aim of clustering is to define groups (also called clusters) with the smallest intra-group distances. It will thus seek to group the elements according to their temporal behavior over the 16 time steps. As an illustration, a spatial partition of the water depth data from Fig. 1 with six clusters is shown on the left hand side of Fig. 3. The temporal water depth variation of each of the six cluster representatives (also called medoids or prototypes) is provided on the right hand side of Fig. 3. The main spatial features of the flow, as identified by the temporal behavior at each cell of the mesh, are clearly discernable. Some clusters may not be spatially contiguous yet the clustering produced is coherent. For example, the yellow cluster in Fig. 3(a) covers the upper and rightmost branches of the intersection where similar behavior occurs owing to the symmetry along the first diagonal arising from the oblique propagation of the flow (see also Fig. 1). For the water depth data, the information on the temporal variability is sufficient to infer spatially coherent clusters. This should generally be the case for environmental data (see, for example, the spatial partition of extreme precipitation from gauged stations in Carreau et al. (2017)). In any case, as mentioned previously, elements belonging to the same cluster have, by definition, smaller distances. Any pair within a cluster can thus be assigned as a pair of predictor-response elements. Even if the pair is not spatially contiguous, prediction errors are ensured to be lower than that of any other pair selected across two different clusters.

At each stage of the lifting scheme, clustering appears naturally as part of the split step. If n denotes the number of elements at the beginning of a given stage of the lifting scheme, then the split step can be thought of as defining a spatial partition into $n/2$ groups with each group containing a pair $(\boldsymbol{x}, \boldsymbol{y})$ of elements, one from the predictor set and the other from the response set. At the first stage, for the water depth data example, this would mean to determine a partition with 1152 groups with precisely two elements per group. This poses a challenge for clustering algorithms as they are designed to identify a partition with a pre-specified number of groups but are not equipped to ensure a pre-specified number of elements in each group (Bishop, 2006). To alleviate this difficulty we propose a clustering assisted split step based on the following rationale. The number of groups should be as close as possible to $n/2$ in order to capture as detailed spatial features as possible but it may be lower to ensure that each group has at least two elements so that the application of the lifting scheme is feasible. With a sufficiently detailed partition, the split step can assign pairs of predictor-response elements arbitrarily within a given cluster, e.g., based on the sequential order of their indices as in the conventional 1D lifting scheme (i.e., predictor elements have even indices and response elements have odd indices). By allowing a predictor element to be paired with two response elements, the applicability of the lifting scheme is seamlessly extended to cases where the initial number of elements is not a power of two.

The clustering assisted split step, detailed in Algorithm 1, has two main parts. In the first part, a spatial partition of the n elements composing the signal at a given stage of the lifting scheme is established. To achieve this efficiently, we rely on a hierarchical clustering algorithm that yields a dendrogram from which clusterings for any number of groups can be extracted quickly (Kaufman and Rousseeuw, 1990). The desired clustering can be obtained by trial and error starting first by extracting from the dendrogram a clustering with $\lfloor n/2 \rfloor$ groups and then reducing the number of groups, e.g., dividing the initial number by two, until no group has less then two elements (see lines 1–7 in Algorithm 1). If the spatial variability in the data is high, it might be desirable to ensure a larger number of clusters. This can be achieved by setting a lower decreasing rate for the number of clusters (for example, use $M \leftarrow \lfloor M/1.2 \rfloor$ on line 5 in Algorithm 1). This would potentially require more computation. Part two of the clustering assisted split step consists in assigning pairs of predictor and response elements following the sequential order of the indices within each group of the spatial partition established in the first part. When a group of the partition has an odd number of elements, the last element assigned as predictor is paired with the last two elements assigned as responses. As a consequence, the size of the partition obtained for the split step will in general be smaller than $\lfloor n/2 \rfloor$ (see lines 8–19 in Algorithm 1).

2.2.2. Adapted predict and update steps

Three modifications of the predict and update steps are required for the lifting scheme in the spatio-temporal framework. First of all, since the elements are vectors containing time series, the operations must be performed component-wise. Second, at a given stage, for predictor elements paired with two response elements, the predict step (see (1)) must be performed twice. Let us use the following simplified notation: $(\boldsymbol{x}, \boldsymbol{y})$ and $(\boldsymbol{x}, \boldsymbol{y}_1, \boldsymbol{y}_2)$ denote respectively a group with two and three elements. Then, for the naive local regression used throughout, the predict step consists in calculating:

$$\begin{cases} \boldsymbol{d} = \boldsymbol{y} - \boldsymbol{x} & \text{group with two elements} \\ \boldsymbol{d}_1 = \boldsymbol{y}_1 - \boldsymbol{x} \quad \boldsymbol{d}_2 = \boldsymbol{y}_2 - \boldsymbol{x} & \text{group with three elements.} \end{cases} \tag{5}$$

The third modification affects the update step in order to account for the number of elements in the aggregated partition created by the iterations of the lifting scheme. At a given stage, the split step creates a partition with groups composed of one predictor element and at most two response elements. At the next stage, these groups are merged by applying the split step on the scaling elements which were the predictor elements of the previous stage. As the number of stages increases, the partition becomes more and more aggregated. This is illustrated on a 3 by 3 toy example for two successive stages in Fig. 4. Each color represents a group with the predictor element indicated by a white x and either one or two paired response elements indicated by a white

Input : $\{s_i\}_{i=1}^n$, $s_i \in \mathbb{R}^d$, the elements to split at the beginning of the stage
Output: pairs or triplets with one predictor element and at most two response elements

$M \leftarrow \lfloor n/2 \rfloor$ the targeted number of groups in the spatial partition
Build ▲ : the dendrogram resulting from a hierarchical clustering applied to $\{s_i\}_{i=1}^n$
$C_M \leftarrow$ cut(▲, M), the clustering obtained by cutting ▲ to yield M groups
while C_M *contains at least one group with one element* **do**
  $M \leftarrow \lfloor M/2 \rfloor$ `// rate of decrease of` M `is 2`
  $C_M \leftarrow$ cut(▲, M)
end
for $j \leftarrow 1$ **to** M **do**
  `// scan each group`
  n_j: # of elements in group j
  $\{s_{i_1}, \ldots, s_{i_{n_j}}\}$: the elements in group j
  `// form the pairs of predictor and response elements`
  $\{(x_k, y_k)\}$, $k \in \mathbb{N}$ and $1 \le k \le n_j/2$ with $x_k = s_{i_{2k}}$ and $y_k = s_{i_{2k-1}}$
  if n_j *is odd* **then**
    `// there is an additional response element`
    $y_{\lfloor n_j/2 \rfloor + 1} = s_{i_{n_j}}$
    $(x_{\lfloor n_j/2 \rfloor}, y_{\lfloor n_j/2 \rfloor + 1})$ `// the last predictor element is paired with two response elements`
  end
end

Algorithm 1: Perform the split step of the lifting scheme assisted by clustering at a given resolution stage (see § 2.2.1).

(a) *Stage 1: initial partition.* (b) *Stage 2: aggregated partition.*

Fig. 4. Toy example: each group is indicated by a color with predictor (response) elements indicated with a white x (y). In stage 1, a clustering algorithm could be used to determine the groups which are then split into predictor and response elements. In stage 2, the two groups can be obtained by clustering the updated predictor elements of stage 1. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

y. At the end of the split step of the first stage shown in Fig. 4(a), there are 4 groups, one of which has 2 response elements. In the second stage shown in Fig. 4(b), the split step consists in assigning the updated predictor elements of the first stage (the scaling coefficients, see (2)) to new pairs of predictor and response elements. This can be thought as aggregating the previous partition yielding the two groups of Fig. 4(b).

At each stage, a coefficient λ such that the average over each group of the aggregated partition is preserved must be determined. For groups with two elements, $\lambda = 1/2$ and the update step from (2) is unchanged except for the operations that are performed component-wise. In contrast, for (x, y_1, y_2), a group with three elements, $\lambda = 1/3$ and the update step becomes:

$$c = \frac{x + y_1 + y_2}{3} = x + \frac{1}{3}(d_1 + d_2), \tag{6}$$

with d_1 and d_2 as in (5). In successive stages of the lifting scheme, λ depends on the number of elements in the groups in the preceding aggregated partition. In Fig. 4(b), $\lambda = 1/2$ for the purple group as there is an equal number of elements in the two groups being merged. However, for the blue group of Fig. 4(b), $\lambda = 2/5$ to account for the differences in the group sizes. The implementation of the spatio-temporal lifting scheme has to keep track of the values of λ for all groups at all stages to perform the update step adequately:

$$c = \begin{cases} x + \lambda d & \text{group with two elements,} \\ x + \lambda(d_1 + d_2) & \text{group with three elements,} \end{cases} \tag{7}$$

with d, d_1 and d_2 as in (5).

The inverse lifting scheme in the spatio-temporal framework follows the same steps as in the 1D case (see (3) and (4)) except that, as previously, the operations are performed component-wise. For the naive local regression, the reverse update and predict steps are obtained by switching additions with subtractions and vice versa in (5) and (7). In particular, for a group with two elements, the reverse update and predict steps restore the predictor and response element respectively while the merge step regroups the scaling coefficients of the previous stage:

$$x = c - \lambda d = c + \alpha_x d \tag{8}$$

$$y = d + x = c + (1 - \lambda)d = c + \alpha_y d \tag{9}$$

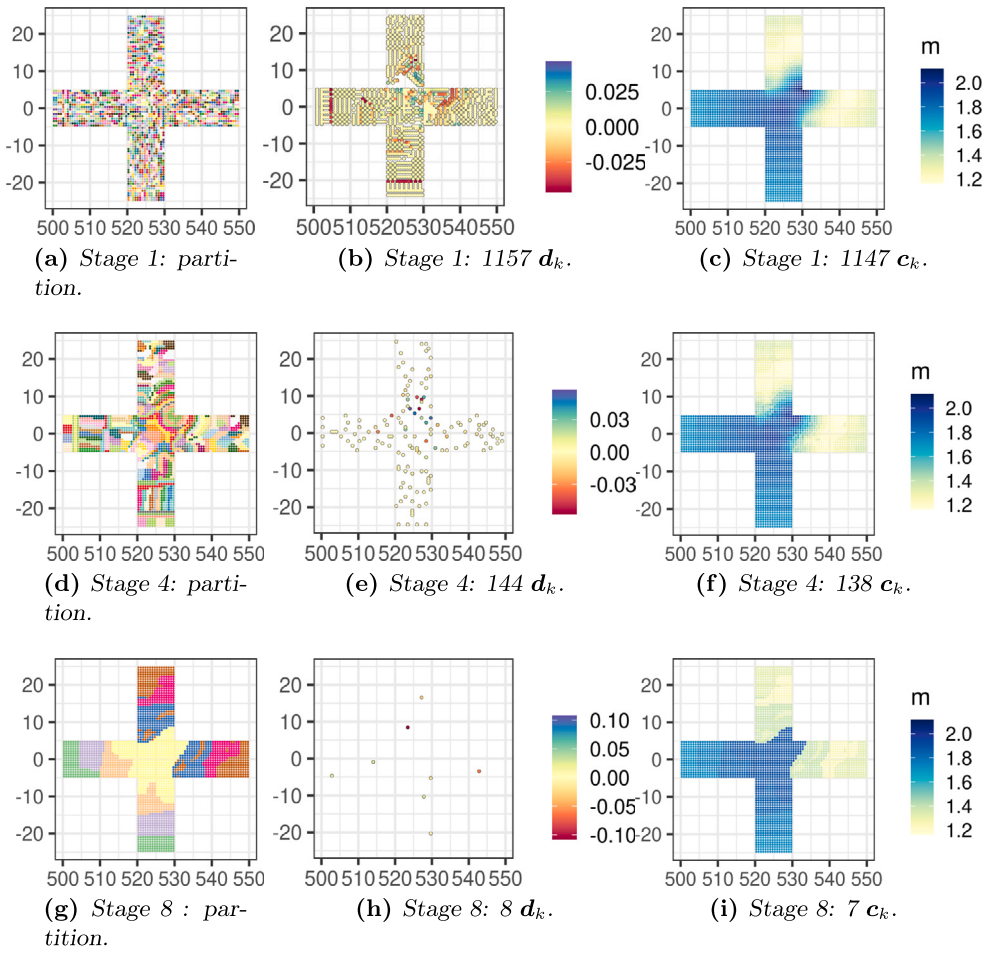
where $\alpha_x = -\lambda$ and $\alpha_y = 1 - \lambda$ are thus coefficients that depend on the aggregated partition and are linked to the particular geometry under study. Similar operations are carried out for groups with three elements.

2.2.3. *Illustration of the spatio-temporal lifting scheme*

The spatio-temporal lifting scheme is illustrated on the water depth data shown in Fig. 1. The split-predict-update steps described above are illustrated for the first stage (Fig. 5(a)-5(c)), the fourth stage (Fig. 5(d)-5(f)) and the eighth stage (Fig. 5(g)-5(i)). In Fig. 5(a), Figs. 5(d) and 5(g), the color scale corresponds to the aggregated partition, i.e., each group of a given color is made of a predictor element and the associated response elements, including the ones from the previous stages (see also Fig. 4). For the predict step, Fig. 5(b), Figs. 5(e) and 5(h), the temporal medians of the wavelet coefficients, see (5), are illustrated. In the update step, Fig. 5(c), Figs. 5(f) and 5(i), the temporal medians of the scaling coefficients computed using (7) are shown on the aggregated partition, i.e., each updated predictor element value is associated to all the elements of its group in the partition from Fig. 5(a), Figs. 5(d) and 5(g). Note that, due to the procedure described in Section 2.2.1, the split step always results in slightly more response than predictor elements, i.e., a small number of predictor elements are paired with two response elements. In some preliminary trials of the clustering assisted split step, the K-medoids algorithm with pre-defined targeted number of clusters for each stage (with values much lower than $\lfloor n/2 \rfloor$) was used instead of the hierarchical clustering based strategy with an adaptive number of clusters described in Algorithm 1. We found that, for the water depth data example, the results of the lifting scheme were not sensitive to the clustering strategy implemented to obtain the spatial partition.

2.3. *Non-parametric representations for spatio-temporal data*

From the spatio-temporal lifting scheme described in Section 2.2, we derive an exact representation of the signal, see Section 2.3.1, and a sparse representation that yields low reconstruction errors by reproducing the signal main features and leaving aside more localized details, see Section 2.3.2.



(a) *Stage 1: partition.* (b) *Stage 1: 1157* $\boldsymbol{d}_k$. (c) *Stage 1: 1147* $\boldsymbol{c}_k$.

(d) *Stage 4: partition.* (e) *Stage 4: 144* $\boldsymbol{d}_k$. (f) *Stage 4: 138* $\boldsymbol{c}_k$.

(g) *Stage 8 : partition.* (h) *Stage 8: 8* $\boldsymbol{d}_k$. (i) *Stage 8: 7* $\boldsymbol{c}_k$.

Fig. 5. Three stages of the spatio-temporal lifting scheme applied to the water depth data from Fig. 1. First column: the partition resulting from the split step, where aggregation means that the response elements of the previous stages are included. Second column: the predict step represented by the temporal medians of the wavelet coefficients. Third column: the update step illustrated by the temporal medians of the scaling coefficients over the partition of the split step. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

2.3.1. Exact representation

A forward application of the spatio-temporal lifting scheme followed by a backward application are necessary to derive this representation. Let n be the initial number of elements in the spatial dimension and T be the number of time steps. The forward application consists in iterating the lifting scheme until a single element remains. The update step ensures that the single scaling coefficient at the last stage is equal to $\boldsymbol{c} = \bar{s} = (\bar{s}_1, \dots, \bar{s}_T)'$, the spatial average over the original n elements at each time step. All the other $n-1$ elements are transformed into wavelet coefficient $\boldsymbol{d}_k = (d_{k1}, \dots, d_{kT})'$, $1 \le k \le n-1$. The backward application consists in reversing the steps of the lifting scheme iteratively, i.e., by applying (8)–(9) starting from the last stage and iterating until the original spatio-temporal field $\boldsymbol{s}_i \in \mathbb{R}^T$ at a given location i, $1 \le i \le n$, may be expressed as

$$\boldsymbol{s}_i = \bar{s} + \sum_{k=1}^{n-1} \alpha_{ki} \boldsymbol{d}_k, \tag{10}$$

where α_{ki}, $1 \le k \le n-1$ are the coefficients associated to the location i that depend on the number of elements in each group of the partitions at each stage (the indices x and y were dropped to unclutter the notation, see (8)–(9)). Note that this representation is exact in that the original signal is reconstructed with no error.

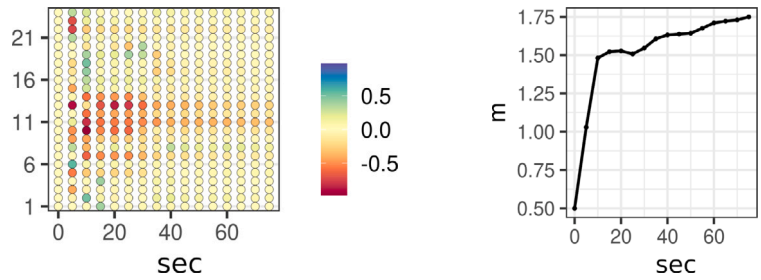
2.3.2. Sparse representation

The lifting scheme decomposition is typically used to carry out compression by eliminating wavelet coefficients that have small values in magnitude. Within the spatio-temporal framework, compression can be performed by eliminating locations for which the wavelet coefficients are small in magnitude at all the time steps thereby reducing the complexity of the expression in (10). In practice, $\boldsymbol{d}_k$, the vector of wavelet coefficients associated to location k, is eliminated (or set to zero) if $d_{kt} \in (d^{\min}_{\epsilon/2}, d^{\max}_{1-\epsilon/2})$ $\forall$ $1 \le t \le T$. The lower bound $d^{\min}_{\epsilon/2}$ of the interval is a quantile of level $\epsilon/2$ of the minimum values with respect to time at each location. The upper bound $d^{\max}_{1-\epsilon/2}$ is set similarly as a quantile of level $1-\epsilon/2$ of the maximum values with respect to time at each location. The quantile levels are thus chosen so that ϵ corresponds to the approximate proportion of locations that are eliminated. Let I_ϵ be the set of selected locations. Then the sparse representation derived by the lifting scheme is given by

$$\hat{\boldsymbol{s}}_i = \bar{s} + \sum_{k \in I_\epsilon} \alpha_{ki} \boldsymbol{d}_k, \tag{11}$$

where $\hat{\boldsymbol{s}}_i$ indicates that the representation is now an approximation to the original signal provided by the lifting scheme algorithm with compressed wavelet coefficients.

In the water depth data example of Fig. 1, there are $n = 2304$ locations. To keep about $\epsilon = 1\%$ of the $n-1 = 2303$ wavelet coefficients, 24 coefficients $\boldsymbol{d}_k$ taking the largest values over time are identified, i.e., d_{kt} is large for at least one time step t, to constitute the set I_ϵ in (11). With this compression level, the reconstructed signal $\hat{\boldsymbol{s}}_i$ based on the quantities shown in Fig. 6 that enter in (11) yields a Root Mean Squared Error (RMSE) of 0.076 m.



(a) *Wavelet coefficients* $\boldsymbol{d}_k$. (b) *Average value* $\bar{\boldsymbol{s}}$.

Fig. 6. Quantities needed for the sparse lifting scheme representation in (11) shown with respect to time (in seconds) for the water depth data from Fig. 1. The initial number of wavelet coefficients is 2303 and the reduced set of locations I_ϵ retains only 24, about 1%.

3. Spatial downscaling of hydraulic fields

In this application, the two most widely acknowledge indicators of flood hazard, namely the water depth and the unit discharge norm, are studied (Carreau and Guinot, 2021). Spatial downscaling, in this case, consists in estimating the high resolution (HR) spatio-temporal flood hazard indicator based on its low resolution (LR) counterpart. As in Carreau and Guinot (2021), in this work we consider the so-called perfect upscaling case, i.e., the LR counterpart is the exact average of the HR simulation over a LR mesh consisting of sub-domains each of which contains several HR cells.

3.1. Low and high resolution flood hazard indicator data

The data considered in this application are HR simulations obtained by solving the two-dimensional shallow water equations in a synthetic urban configuration, see Section 3.1.1, and a field-scale test case, see Section 3.1.2 (this is a subset of the data described and used in Carreau and Guinot (2021)). Let the LR mesh be defined by a partition of sub-domains Ω_i, $1 \le i \le N$ and let the HR mesh be defined by a partition of cells ω_j, $1 \le j \le n$. Let $s_{\omega_j t}$ be a HR hazard indicator at time step t for the HR cell ω_j. Then, as the perfect upscaling case is considered, the LR counterpart $S_{\Omega_i t}$ at time step t for the LR sub-domain Ω_i is computed as

$$S_{\Omega_i t} = \frac{1}{|\Omega_i|} \sum_{\omega_j \in \Omega_i} s_{\omega_j t} \tag{12}$$

where $|\Omega_i|$ is the number of HR cells ω_j in the LR sub-domain Ω_i.

3.1.1. Synthetic oblique dam break configuration

This configuration corresponds to a 2D oblique urban dam break problem without friction (Carreau and Guinot, 2021). It is the same configuration that served to create the water depth data from Fig. 1 which contains a single intersection located in the middle of a street made in total of 20 identical intersections. Each intersection corresponds to one sub-domain that contains 2304 HR cells regularly meshed. In the downscaling application, we use the eight most central sub-domains including the sub-domain from Fig. 1. The water depth and unit discharge norm data simulated over these eight sub-domains are shown in Fig. 7 over three time steps. As can be seen from Figs. 7(a) and 7(b), the water is initially at rest with the water depth piecewise constant: 3 m on the left extremity and 0.5 m on the right extremity. Ten different flow scenarios are obtained by letting these initial conditions vary, see Table 1.

3.1.2. Sacramento field-scale test case

The geometry of this test case is based on a neighborhood of the city of Sacramento flanked by a canal protected by a dike (Guinot et al., 2017). The simulation of the propagation, without friction, of a flood wave caused by a breach in the dike is performed. The LR and HR meshes are reduced to a subset of the original meshes containing the area most affected by the flood. The HR mesh has 9749 cells in total irregularly divided among the 199 subdomains of the LR mesh. The water depth and unit discharge norm data simulated over these meshes are shown in Fig. 8 for three time steps. The canal is located vertically at the left-hand side and the breach occurs about the middle of the mesh (i.e., between coordinates 596950 and 597050 on the y-axis). The initial water depth in the canal is of 6 m. Five flow scenarios are obtained by letting this initial condition vary, see Table 2.

Table 1
Flow scenarios for the synthetic dam break configuration defined in terms of the initial water depth values at the left and right extremities of the street respectively used to form the training, validation and test sets.

Set	Label	Initial water depth (m)
Training	(a)	(3, 2.5)
	(b)	(3, 0.5)
	(c)	(1, 0.5)
Validation	d	(2, 0.5)
	(e)	(3, 1.5)
Test	f	(2.3, 1.2)
	(g)	(2, 0.01)
	(h)	(5, 1.5)
	(i)	(0.1, 0.01)
	(j)	(5, 4.5)

Table 2
Flow scenarios for the Sacramento field-test case defined in terms of the initial water depth value in the canal used to form the training, validation and test sets.

Set	Label	Initial water depth (m)
Training	(a)	6
	(b)	4.5
Validation	(c)	5.25
Test	(d)	3.5
	(e)	7

3.2. Downscaling models as generative algorithms of HR fields

In the urban flood hazard application, from the operational viewpoint, it is assumed that the configuration can be known ahead but the exact flow scenario, defined by the hydraulic conditions, cannot. For instance, if we seek to assess flood hazards for the Sacramento neighborhood (see Section 3.1.2), the geometry has already been established but it might be difficult to know in advance what will be the water depth in the canal when the breach occurs. In addition, it is assumed that LR hazard indicator fields can be simulated with a hydraulic model having reasonable computation times (see for instance Guinot et al. (2017), although in our work, we assumed that the LR fields are exact upscaled versions of the HR fields). Therefore, for a given configuration, our goal is to define a fast statistical downscaling model that can

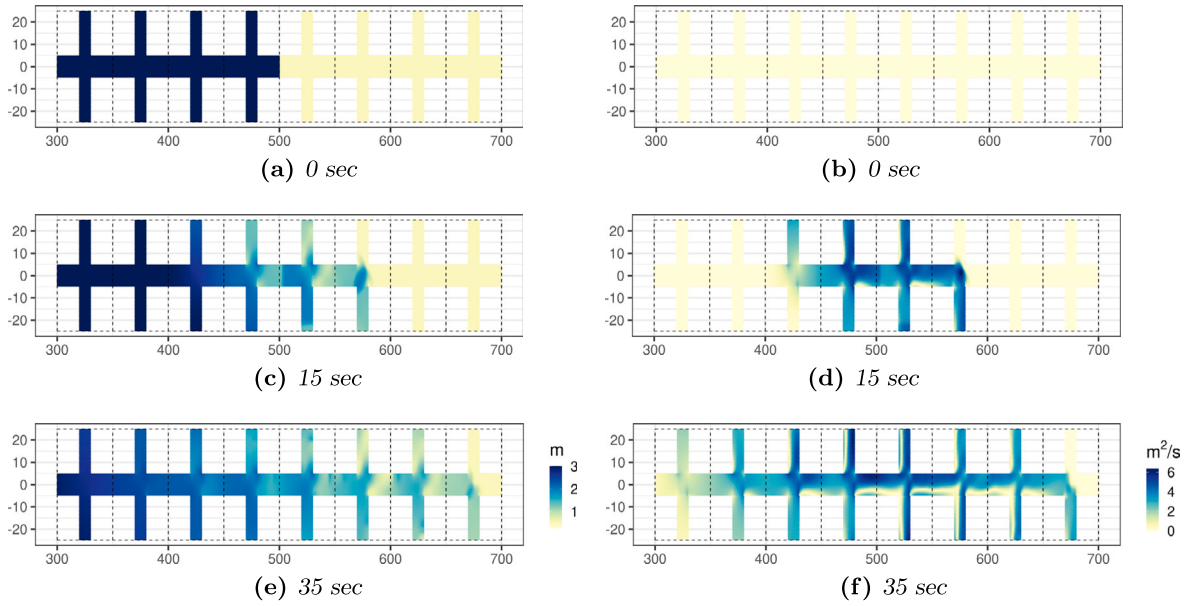


Fig. 7. Water depth (left) and unit discharge norm (right) at three time steps (in seconds) for the synthetic oblique dam break over eight sub-domains delineated with dashed black lines (each sub-domain measures 50 m by 50 m and contains 2304 HR cells). Initial conditions correspond to flow scenario (b) in Table 1. The x- and y-axis coordinates are in meters.

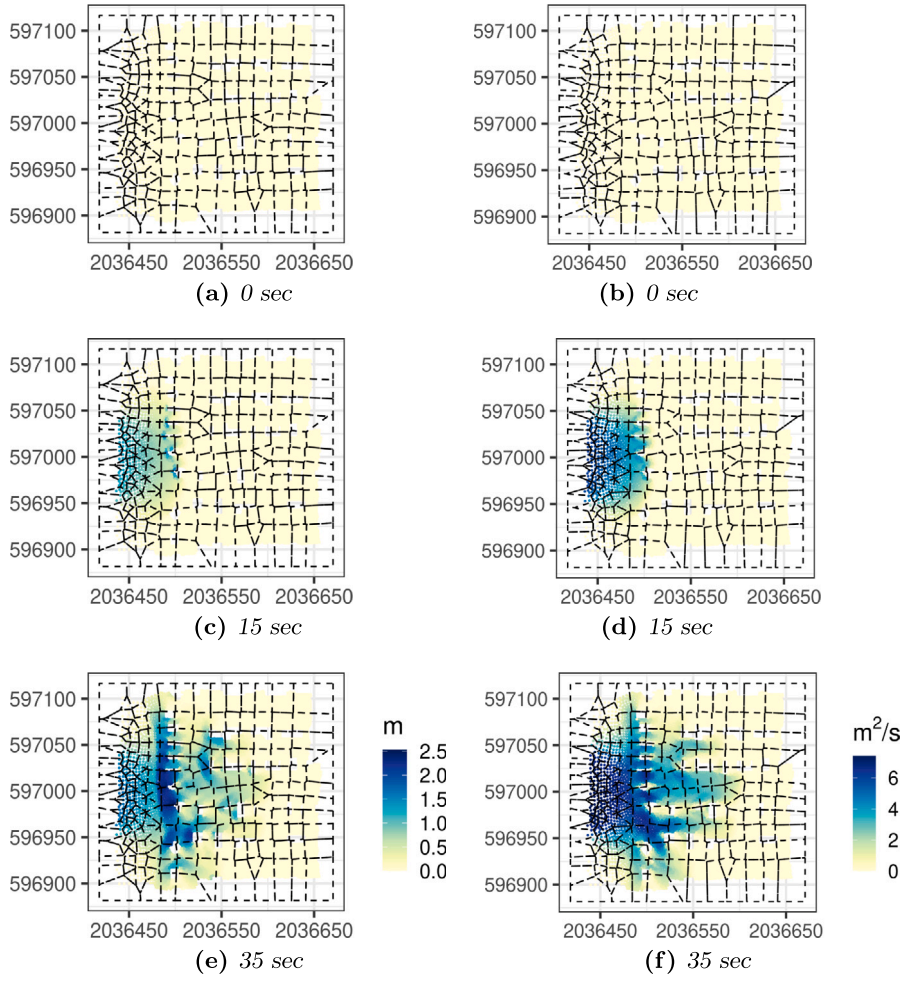


Fig. 8. Water depth (left) and unit discharge norm (right) at three time steps (in seconds) for the Sacramento field-test case (199 sub-domains delineated with dashed black lines containing in all 9749 HR cells). The initial condition corresponds to flow scenario (a) in Table 2. The x- and y-axis coordinates are in meters.

generate HR hazard indicator fields for any flow scenario based on LR counterparts. Note that the mesh of the LR field is assumed to cover the same domain as the mesh of the HR field.

By leveraging the spatio-temporal lifting scheme presented in Section 2, we first define the model for the HR hazard indicator fields in Section 3.2.1 and we explain how to derive the various coefficients and variables involved in the model from a set of flow scenarios considered for training. In Section 3.2.2, we define an alternative model that follows the same rationale but based on principal component analysis instead of the lifting scheme. Then, downscaling in itself consists in providing coefficient and variable estimates for the HR models based on the LR hazard indicator fields, a form of generative algorithm, for which a strategy is described in Section 3.2.3.

3.2.1. Standardized HR indicator model

The flow scenarios defined by the initial hydraulic conditions affect the overall magnitude of the flow. However, the spatial patterns may be shared across these flow scenarios. For this reason, the HR fields are standardized beforehand by mapping their values approximatively to the $[0, 1]$ interval as follows:

$$\sigma_{\omega_j t} = \frac{s_{\omega_j t} - a}{b - a}, \tag{13}$$

where $a = \min_{\Omega_i, \tau} S_{\Omega_i \tau}$ and $b = \max_{\Omega_i, \tau} S_{\Omega_i \tau}$ are respectively the minimum and maximum values of the LR field across all sub-domains and time steps. The above standardization may be reversed relying only on LR hazard indicator fields which is useful in the downscaling context.

The model for the standardized HR indicators relies, for each given sub-domain Ω_i, on the exact representation arising from the spatio-temporal lifting scheme, see Section 2.3.1. (10) is re-written to emphasize the temporal dynamics of $\boldsymbol{\sigma}_{\Omega_i t} = (\sigma_{\omega_1 t}, \dots, \sigma_{\omega_{n_i} t})'$, with $\omega_j \in \Omega_i \ \forall\ 1 \le j \le n_i = |\Omega_i|$, the standardized indicator at a given time step t over all the locations in sub-domain Ω_i:

$$\boldsymbol{\sigma}_{\Omega_i t} = \bar{\sigma}_{\Omega_i t} + \sum_{k=1}^{n_i - 1} \boldsymbol{\alpha}_{\Omega_i k} d_{\Omega_i k t}. \tag{14}$$

The representation for the whole HR field, i.e., across all sub-domains, can be obtained by concatenating the representations of each sub-domain yielding $\boldsymbol{\sigma}_t = (\boldsymbol{\sigma}_{\Omega_1 t}, \dots, \boldsymbol{\sigma}_{\Omega_N t})'$. Similarly, let $\bar{\boldsymbol{\sigma}}_t = (\bar{\sigma}_{\Omega_1 t}, \dots, \bar{\sigma}_{\Omega_N t})'$ contain the spatial averages at time t over all sub-domains, let $\mathcal{A}$ be the set of geometry coefficients $\boldsymbol{\alpha}$ for all sub-domains and let $\boldsymbol{d}_t = (d_{\Omega_1 1 t}, \dots, d_{\Omega_1 (n_1 - 1) t}, \dots, d_{\Omega_N 1 t}, \dots, d_{\Omega_N (n_N - 1) t})'$ contain the wavelet coefficients at time t over all sub-domains. The dimension of $\boldsymbol{d}_t$ is n, the number of cells in the HR mesh, minus N, the number of LR sub-domains.

To reduce the complexity of the exact representation, instead of letting $\boldsymbol{d}_t$ vary continuously with time, we assume that there is a finite number J of categories characterized by fixed wavelet coefficient values $\boldsymbol{d}_j$, with $1 \le j \le J$. Let Z_t be a latent variable taking discrete values $\{1, \dots, J\}$ representing the category occurring on time step t. In contrast to (14), the model for the standardized HR indicators for a given sub-domain Ω_i depends on Z_t to identify the current category associated to the wavelet coefficients:

$$\hat{\boldsymbol{\sigma}}_{\Omega_i t} = \bar{\sigma}_{\Omega_i t} + \sum_{k=1}^{n_i - 1} \boldsymbol{\alpha}_{\Omega_i k} d_{\Omega_i k Z_t}. \tag{15}$$

An estimate $\hat{\boldsymbol{\sigma}}_t$ of $\boldsymbol{\sigma}_t$ is thus obtained by combining the following quantities:

- The spatial averages in $\bar{\boldsymbol{\sigma}}_t$ at time t over each sub-domain, scalar values that contain no information on the spatial structure within the sub-domains;
- The coefficients $\boldsymbol{\alpha} \in \mathcal{A}$, the only quantities that vary spatially within the sub-domains and depend on its geometry but do not vary with time nor from one flow scenario to another;
- The wavelet coefficients $\boldsymbol{d}_j$ that describe the variation of the HR spatial pattern within the sub-domains for each category $1 \le j \le J$ and Z_t the category occurring on time step t.

To obtain estimates of the various quantities involved in the computation of $\hat{\boldsymbol{\sigma}}_t$ (see (15)) over all the sub-domains of a configuration considering several flow scenarios in the training set, we follow a strategy with two main parts (see Algorithm 2 for which three flow scenarios are considered as the training set). The first part consists in running a forward application of the spatio-temporal lifting scheme (i.e., iterate enough stages so that a single scaling coefficient remains as in Section 2.3.1) on the standardized HR indicators for all the flow scenarios retained for training for each sub-domain separately. This yields the geometry coefficients $\boldsymbol{\alpha} \in \mathcal{A}$, the spatial average values $\bar{\boldsymbol{\sigma}}_t^{(\cdot)}$ and the wavelet coefficients $\boldsymbol{d}_t^{(\cdot)}$, one for each time step t of each flow scenario $^{(\cdot)}$ (see lines 1–6 in Algorithm 2). In the second part, clustering is performed across the temporal dimension and the flow scenarios of the wavelet coefficients obtained with the forward application to identify $\boldsymbol{d}_j$. As wavelet coefficients may be very high dimensional, compression (see Section 2.3.2) is applied beforehand (see lines 7–12 in Algorithm 2). Clustering also yields C_J, the categories to which each time step of each flow scenario belongs, i.e., the values most likely taken by the latent variable $Z_t^{(\cdot)}$.

Input : $\{\boldsymbol{\sigma}_{\omega_j}^{(a)}\}_{j=1}^n$, $\{\boldsymbol{\sigma}_{\omega_j}^{(b)}\}_{j=1}^n$, $\{\boldsymbol{\sigma}_{\omega_j}^{(c)}\}_{j=1}^n$ with $\boldsymbol{\sigma}_{\omega_j}^{(\cdot)} \in \mathbb{R}^T$, the standardized HR hazard indicators at location ω_j (see (13)) for the flow scenario $^{(\cdot)}$ used for training
J, the number of categories of wavelet coefficients
ϵ, the approximate level of compression (see § 2.3.2)

Output: $\bar{\boldsymbol{\sigma}}_t^{(\cdot)}$, the spatial average over each sub-domain at each time step for each flow scenario
$\mathcal{A}$, the geometry coefficients over all sub-domains
$\{\boldsymbol{d}_j\}_{j=1}^J$, the wavelet coefficients describing each category
C_J, the categories associated to each time step of each flow scenario

1 `// concatenate flow scenarios for each location` ω_j
2 $(\sigma_{\omega_j 1}^{(a)}, \dots, \sigma_{\omega_j T}^{(a)}, \sigma_{\omega_j 1}^{(b)}, \dots, \sigma_{\omega_j T}^{(b)}, \sigma_{\omega_j 1}^{(c)}, \dots, \sigma_{\omega_j T}^{(c)}) \quad 1 \le j \le n$
3 `// make a forward application of the spatio-temporal lifting scheme on the concatenated scenarios separately for each sub-domain and obtain (§ 2.3.1):`
4 $(\bar{\boldsymbol{\sigma}}_1^{(a)}, \dots, \bar{\boldsymbol{\sigma}}_T^{(a)}, \bar{\boldsymbol{\sigma}}_1^{(b)}, \dots, \bar{\boldsymbol{\sigma}}_T^{(b)}, \bar{\boldsymbol{\sigma}}_1^{(c)}, \dots, \bar{\boldsymbol{\sigma}}_T^{(c)})$, spatial averages per sub-domain
5 $\mathcal{A}$, geometry coefficients
6 $(\boldsymbol{d}_1^{(a)}, \dots, \boldsymbol{d}_T^{(a)}, \boldsymbol{d}_1^{(b)}, \dots, \boldsymbol{d}_T^{(b)}, \boldsymbol{d}_1^{(c)}, \dots, \boldsymbol{d}_T^{(c)})$ with $\boldsymbol{d}_t^{(\cdot)} \in \mathbb{R}^{n-N}$, wavelet coefficients
7 `// perform compression (see § 2.3.2)`
8 let I_ϵ be the set of locations with significant wavelet coefficients
9 let $\boldsymbol{\delta}_t^{(\cdot)} \subset \boldsymbol{d}_t^{(\cdot)}$ such that $\boldsymbol{\delta}_t^{(\cdot)} \in \mathbb{R}^{|I_\epsilon|}$
10 `// perform clustering on` $\boldsymbol{\delta}_t^{(\cdot)}$ `yielding:`
11 $C_J = \{z_1^{(a)}, \dots, z_T^{(a)}, z_1^{(b)}, \dots, z_T^{(b)}, z_1^{(c)}, \dots, z_T^{(c)}\}$ with $z_t^{(\cdot)} \in \{1, \dots, J\}$ the type assigned to each time step of each scenario
12 $\boldsymbol{d}_j \in \mathbb{R}^{n-N}$, $1 \le j \le J$ wavelet coefficients associated with the medoids of each cluster

Algorithm 2: Estimation of the quantities involved to obtain the estimate $\hat{\boldsymbol{\sigma}}_t$ (see (15)) over all the sub-domains of a given configuration for which three flow scenarios (a), (b) and (c) are considered for training (see § 3.2.1)).

3.2.2. Alternative standardized HR indicator model

For comparison purposes, we derive a model similar to the one described in Section 3.2.1 based on PCA, see Bishop (2006) for instance

for an in-depth description. Indeed, PCA may be thought of as an alternative to obtain exact and sparse representations such as the ones provided by the spatio-temporal lifting scheme in Section 2.3 (see also the approach proposed in Carreau and Guinot (2021) that relies on PCA in a slightly different manner). As in Section 3.2.1, let $\boldsymbol{\sigma}_t$ be the vector of standardized HR indicators over all sub-domains for a given time step t. Let $\boldsymbol{\Gamma}$ be the $n \times n$ sample variance–covariance matrix of the standardized HR indicator vectors. Assuming that n is much larger than the temporal dimension T, as is the case in our application, then $\boldsymbol{\Gamma}$ has T eigenvectors $\boldsymbol{u}_k \in \mathbb{R}^n$, $1 \leq k \leq T$. The exact PCA representation for $\boldsymbol{\sigma}_t \in \mathbb{R}^n$, the spatial signal at a given time t over all the sub-domains, is given by

$$\boldsymbol{\sigma}_t = \sum_{k=1}^{T} \beta_{kt} \boldsymbol{u}_k, \tag{16}$$

where β_{kt} are scalar coefficients that represent the orthogonal projection of $\boldsymbol{\sigma}_t$ onto $\boldsymbol{u}_k$ (Bishop, 2006). To perform compression and derive a sparser approximate representation, a reduced number of eigenvectors (smaller than T) corresponding to the largest eigenvalues may be retained.

The PCA representation in (16) describes the signal in terms of a weighted sum of the eigenvectors $\boldsymbol{u}_k$ that can be thought of as spatial patterns that carry no temporal information. The temporal variation in the signal is conveyed by the weights β_{kt} of the eigenvectors. Instead of letting these weights vary continuously in time, we make an assumption similar to the one yielding (15), i.e., there is only a finite number J of categories characterized by fixed weight values $\boldsymbol{\beta}_j = (\beta_{1j}, \dots, \beta_{Tj})'$. As previously, let Z_t be a latent variable taking discrete values $\{1, \dots, J\}$ representing the category occurring on time step t. The alternative PCA model for the standardized HR indicators over all the sub-domains involves Z_t to identify the current category associated to the PCA weights:

$$\hat{\boldsymbol{\sigma}}_t = \sum_{k=1}^{T} \beta_{kZ_t} \boldsymbol{u}_k. \tag{17}$$

To obtain the quantities involved in (17), the aforementioned strategy summarized in Algorithm 2 is modified as follows. In the first part, instead of running a forward application of the spatio-temporal lifting scheme, PCA is applied on the standardized HR indicators over all the sub-domains for all the flow scenarios retained for training. This will yield the eigenvectors $\boldsymbol{u}_k^{(\cdot)}$, $1 \leq k \leq T$, for each flow scenario $^{(\cdot)}$ of the training set and the coefficients $\boldsymbol{\beta}_t^{(\cdot)}$ for each time step t and each flow scenario $^{(\cdot)}$ of the training set. In the second part, clustering is performed across the temporal dimension and the flow scenarios of the coefficients $\boldsymbol{\beta}_t^{(\cdot)}$ to identify $\boldsymbol{\beta}_j$, the coefficients associated to the medoids, and to estimate C_J, the categories most likely associated to each time step of each flow scenario. As previously, the dimension of the coefficients $\boldsymbol{\beta}_t^{(\cdot)}$ may be reduced prior to clustering by including only the coefficients associated to the principal eigenvectors, i.e., associated with the highest eigenvalues.

3.2.3. Downscaling strategy

To perform downscaling, an estimate of the standardized HR field $\hat{\boldsymbol{\sigma}}_t$ must be computed, i.e., the quantities involved in (15) or in (17) have to be estimated, based only on information provided by the LR hazard indicator fields.

We present first the downscaling model that allows to turn into a generative algorithm the HR model in (15) arising from the spatio-temporal lifting scheme. The strategy to obtain the estimates of the coefficients and latent variables of the HR model (see Algorithm 2) is ran on the HR fields of the flow scenarios available for training. Among the quantities that are estimated, the geometry coefficients in the set $\mathcal{A}$ and the wavelet coefficients $\boldsymbol{d}_j$ associated to each category define the types of spatial patterns that occur in the standardized HR fields. Assuming that the training set is sufficiently representative, $\mathcal{A}$ and $\boldsymbol{d}_j$ are henceforth held fixed given the configuration. The spatial average values $\bar{\sigma}_t^{(\cdot)}$ correspond directly to LR field values since the exact upscaling case is considered. The main challenge rests in estimating the values of $Z_t^{(\cdot)}$, the variable representing the category occurring on time step t for each flow scenario $^{(\cdot)}$.

As likely category values C_J for $Z_t^{(\cdot)}$ over the training set are computed with the estimation of the HR model (see Algorithm 2), a classifier may be set up to learn which category is associated with given LR field values. The input $\boldsymbol{X} \in \mathbb{R}^D$ that enters in the classifier must be derived from the LR hazard indicator fields. To provide a maximum of information to predict the category which is associated to the current HR spatial pattern of a given hazard indicator, we decided to use in the input both LR hazard indicator fields, i.e., the water depth and the norm of the unit discharge. For each configuration and each flow scenario, each LR hazard indicator is first standardized similarly as the HR hazard indicators (see (13)):

$$\frac{S_{\Omega_i t} - a}{b - a} \quad \forall\, \Omega_i \tag{18}$$

where, as previously, $a = \min_{\Omega_i, \tau} S_{\Omega_i \tau}$ and $b = \max_{\Omega_i, \tau} S_{\Omega_i \tau}$ are respectively the minimum and maximum values of the LR field across all sub-domains and time steps. Each standardized LR indicator field has values, for a given time step, over the whole domain and can thus be very high dimensional. To reduce the dimension, we resort to the scaling coefficients of a given stage of the spatio-temporal lifting scheme described in Section 2.2 applied to each of the standardized LR indicators. Indeed, scaling coefficients can be seen as extracting the main features of the spatial pattern, see the right column of Fig. 5. To sum up, $\boldsymbol{X}$ contains the scaling coefficients at a given stage of the lifting scheme for each of the two standardized LR indicators. For simplicity, the same stage is used for both indicators.

We consider in this implementation a flexible classifier consisting of a feed-forward one hidden layer neural network with a skip layer connection as provided in the R library described in Venables and Ripley (2002). With zero hidden units, the neural network boils down to a linear classifier. With a sufficient number of hidden units, it can in principle approximate any non-linear classification problems. This specific architecture was selected as it allows automatic tuning, by means of the selection of the number of hidden units, of the complexity level (from linear to highly non-linear) suitable to model optimally the decision surfaces of any classification problem. More precisely, the neural network consists of three stacked layers: the input layer $\boldsymbol{X} \in \mathbb{R}^D$, the hidden layer with H neurons a_k, with $1 \leq k \leq H$, and the output layer that has J neurons computing the probability p_j of belonging to a given category j, with $1 \leq j \leq J$. Each hidden neuron computes the following transformation of the inputs:

$$a_k = a(\boldsymbol{X}; \boldsymbol{w}_k^{\text{hid}}) = \tanh\left(\sum_{i=1}^{D} w_{k,i}^{\text{hid}} X_i + w_{k,0}^{\text{hid}} \right) \quad k = 1, \dots, H, \tag{19}$$

with $\boldsymbol{w}_k^{\text{hid}}$ the $(D+1)$ weight vector dedicated to the k^{th} hidden neuron. Each output neuron j then makes the following computation:

$$p_j = p_j(\boldsymbol{X}; \boldsymbol{w}) = g(o_j) \tag{20}$$

$$= g\left(\sum_{k=1}^{H} w_{jk}^{\text{out}} a_k + \sum_{i=1}^{D} w_{ji}^{\text{lin}} X_i + w_{j0}^{\text{lin}} \right), \tag{21}$$

where $\boldsymbol{w}_j^{\text{out}} = (w_{j1}^{\text{out}}, \dots, w_{jH}^{\text{out}})$ contains the H weights connecting the hidden layer to the j^{th} output neuron, $\boldsymbol{w}_j^{\text{lin}} = (w_{j0}^{\text{lin}}, w_{j1}^{\text{lin}}, \dots, w_{jD}^{\text{lin}})$ are the weights of the skip layer connection for the j^{th} output neuron and $g(o_j) = \frac{\exp(o_j)}{\sum_{k=1}^{J} \exp(o_k)}$ is the so-called *softmax* transform that allows to interpret the outputs as probabilities (i.e., $0 \leq p_j \leq 1$, $\forall j$ and $\sum_{j=1}^{J} p_j = 1$). The weights of all the layers of the neural network are gathered in $\boldsymbol{w}$ and are optimized by minimizing the cross-entropy criterion which can also be seen as the negative conditional log-likelihood of the multinomial model, see Venables and Ripley (2002). To avoid local

minima, the optimization is re-started 10 times and the neural network weights yielding the lowest training error are kept.

Several hyper-parameters control the complexity level of the above downscaling model. Concerning the standardized HR indicator model (see Section 3.2.1 and Algorithm 2), the level of compression defined by ϵ applied before clustering (in other words, the proportion of retained locations for the wavelet coefficients) and the number of categories J are considered as hyper-parameters. In the neural network classifier, the stage m at which the scaling coefficients are retained for each LR hazard indicator that determines the input dimension and the number of hidden units H are hyper-parameters directly influencing the overall number of free parameters. All the above hyper-parameter values are selected with a training-validation procedure, i.e., by selecting values that yield the best performance as measured on the validation sets.

The downscaling model that allows the HR model in (17) relying on PCA to become a generative algorithm is very similar. The variant of the strategy to estimate coefficients and latent variables for this model is ran on the HR fields of the flow scenarios used for training. This will yield the eigenvectors $\boldsymbol{u}_k^{(\cdot)}$, $1 \leq k \leq T$, for each flow scenario $^{(\cdot)}$ of the training set and $\boldsymbol{\beta}_j$ $1 \leq j \leq J$, the coefficients associated to each category. As previously, we assume that the training set is sufficiently representative so that these quantities are henceforth held fixed given the configuration. To estimate the values of $Z_t^{(\cdot)}$, the same neural network classifier approach is considered based on the likely values C_J obtained for the training set. The dimension of the LR hazard indicator fields used as inputs in the classifier is reduced with PCA instead of the lifting scheme. As previously, the same number of components is selected for both LR indicators. The hyper-parameters in this alternative downscaling model are the compression level before performing clustering controlled by K_{HR}, the number of retained eigenvectors from PCA applied to the HR fields, the number of categories J for the PCA coefficients, K_{LR}, the number of retained eigenvectors from PCA applied to the LR fields to create the inputs to the classifier and the number of hidden units H of the neural network. These hyper-parameter values are also selected with a training-validation procedure.

3.3. Evaluation and comparison

The two downscaling models described in Section 3.2.3 based on models for the HR standardized hazard indicators, derived either from the spatio-temporal lifting scheme (see Section 3.2.1) or from PCA (see Section 3.2.2), are evaluated and compared on the hazard indicator fields from the two configurations described in Section 3.1. The LR hazard indicator field, which in this work is the exact average of the HR field over each sub-domain, is included in the comparison as a benchmark (i.e., no downscaling is performed). Bilinear or bicubic interpolation approaches used to interpolate the LR values at the centers of the sub-domains to the HR cells are commonly used as benchmarks. However, these are not straightforwardly applicable in the geometries considered here as the meshes are irregular. We include as alternatives a local-linear non-parametric regression with cross-validated bandwidth selection as implemented by Hayfield and Racine (2008) for the Sacramento field-scale test and the inverse distance interpolation approach for the synthetic dam break configuration (since the centers of the sub-domains are co-linear, the former interpolation approach is not applicable).

3.3.1. Hyper-parameter selection

For both downscaling models, we followed a conventional training-validation procedure to select hyper-parameter values (see for instance Bishop (2006)). To this end, several sets of hyper-parameter values are considered. The sets are built by listing potential values for each hyper-parameter and by taking the cartesian product of these lists, i.e., all the combination of values from the lists are included. Several rounds of selection may be necessary to modify the lists in order to ensure that the selected values are not unduly limited by the choices of values available (e.g., the minimum or maximum value of the list is selected). In our R implementation running on an ordinary laptop, a typical round of selection takes between 3 to 6 min for the PCA based downscaling model and between 1 to 1.5 h for the lifting scheme based downscaling model. The implementation of the latter can certainly be optimized and brought up to the speed of the PCA based model. The training and validation sets for each configuration are made of a number of flow scenarios, see Tables 1 and 2. For each set of hyper-parameter values, training is performed on the flow scenarios forming the training set of each configuration to estimate the coefficients related to the HR spatial patterns in each model (either lifting scheme or PCA) and the weights of the neural network classifier. The set of hyper-parameter values yielding the best performance measured on the flow scenarios of the validation sets is retained. Performance is evaluated in terms of root-mean-square error of the estimated standardized HR field $\hat{\boldsymbol{\sigma}}_t$.

Table 3
Selected hyper-parameter values for each downscaling model either based on the lifting scheme or PCA (see Section 3.2), on each configuration (see Section 3.1) for each of the two considered flood hazard indicators (water depth and unit discharge norm). ϵ: proportion of retained location in the compression, J: the number of categories, m: the resolution stage for the LR fields, H: the number of hidden units, K_{HR}, K_{LR}: number of retained eigenvectors for the HR and LR fields.

Configuration	Hazard indicator	Lifting scheme				PCA			
		ϵ	J	m	H	K_{HR}	J	K_{LR}	H
dam break	Water depth	0.015	36	0	1	48	32	8	2
	Unit discharge norm	0.02	40	0	1	2	40	2	2
Sacramento	Water depth	0.1	40	5	0	4	36	8	0
	Unit discharge norm	0.015	36	4	0	98	40	8	1

The hyper-parameter values selected for each downscaling model on each configuration for each hazard indicator are provided in Table 3. The compression level for the model based on the lifting scheme before performing clustering is quite high (the list of potential values for ϵ, the proportion of retained locations, ranges from 0.015 to 0.5 while the select values vary between 0.015 and 0.1). For the model based on PCA, K_{HR} indicates the number of retained eigenvectors. The higher K_{HR}, the lesser the compression applied. The potential values ranges from 2 to the training set size (48 time steps for the dam break configuration and 98 for Sacramento), which is the maximum possible number of retained eigenvectors meaning that no compression is performed. Hyper-parameter selection in Table 3 indicates a number of retained eigenvectors that can be either very small (K_{HR} is 2 or 4) or very high (the maximum possible). The number of categories J describing the spatial patterns is rather high in all cases, with a minimum of 2 and a maximum of 40 permitted which is often reached. For the lifting scheme model on the dam break configuration, no compression is applied (the list of potential values for m, the lifting scheme stage, ranges from 0, which means no compression, to 7) to the LR fields to reduce the dimensionality of the input in the neural network classifier (the input dimension is thus 16). For the Sacramento configuration, as the number of sub-domains is much higher (199 sub-domains instead of 8 for the dam break configuration), a certain amount of compression is required (m equals to 4 or 5 that corresponds to input dimension of 18 or 8 respectively). For the PCA model, the maximum possible number of retained eigenvectors (no compression) for the LR fields corresponds to the number of sub-domains, i.e., the maximum possible K_{LR} equals to 8 for the dam break configuration or 199 for Sacramento. The minimum value for K_{LR} was set to 1. Thus, no compression is performed for the water depth in the dam break configuration ($K_{\mathrm{LR}} = 8$ which corresponds to input dimension of 16). In all other cases, a certain level of compression was selected (K_{LR} equals to 2 or 8 which corresponds to input dimension of 4 or 16 respectively). For all the cases, the number of hidden units H in the neural network classifier is low (H varies between 0, a linear model, and 2 while a maximum of 8 was permitted) meaning that the classification problem is not very complex.

3.3.2. Test set evaluation

Each downscaling model, either based on the lifting scheme or PCA, is trained anew combining the flow scenarios from the training set and the validation set with the set of hyper-parameter values selected on the validation set indicated in Table 3 above. Generation of HR hazard indicator fields is then performed on the test set with the trained downscaling models using as inputs the LR fields only, see Tables 1 and 2 for the definition of the flow scenarios forming the test set. Computation times for the test set evaluation are of about 5 s for both versions of the downscaling models. Performance is evaluated by comparing the original HR fields of the test set with the re-scaled estimates:

$$\hat{\boldsymbol{s}}_t = (b-a)\hat{\boldsymbol{\sigma}}_t + a, \tag{22}$$

where, as in (13), $a = \min_{\Omega_i,\tau} S_{\Omega_i \tau}$ and $b = \max_{\Omega_i,\tau} S_{\Omega_i \tau}$ are respectively the minimum and maximum values of the LR field across all sub-domains and time steps. In contrast, the interpolation approaches are applied directly for each time step t to $\boldsymbol{S}_t$, the LR values across all sub-domains, to obtain HR estimates $\hat{\boldsymbol{s}}_t$ over the HR mesh.

To compare $\boldsymbol{s}_t = (s_{1t}, \dots, s_{nt})'$ with $\hat{\boldsymbol{s}}_t = (\hat{s}_{1t}, \dots, \hat{s}_{nt})'$ where n is the number of cells in the HR mesh, three performance criteria are considered:

$$\text{Root-mean-square error} \ : \ RMSE(\boldsymbol{s}_t, \hat{\boldsymbol{s}}_t) = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(s_{it} - \hat{s}_{it})^2} \tag{23}$$

$$\text{Mean-absolute error} \ : \ MAE(\boldsymbol{s}_t, \hat{\boldsymbol{s}}_t) = \frac{1}{n}\sum_{i=1}^{n}|s_{it} - \hat{s}_{it}| \tag{24}$$

$$\text{Peak-signal-to-noise ratio} \ : \ PSNR(\boldsymbol{s}_t, \hat{\boldsymbol{s}}_t) = 20\log_{10}\left(\frac{\max_i s_{it}}{RMSE(\boldsymbol{s}_t, \hat{\boldsymbol{s}}_t)}\right) \tag{25}$$

PSNR is commonly used to assess the quality of reconstructed images (Korhonen and You, 2012). A higher value indicates a reconstruction of better quality, sharper contrast between signal and noise. PSNR has been used also in the context of spatial downscaling, see for instance Rezvov et al. (2021). For each configuration and each hazard indicator, the performance criteria are summarized by computing their average and standard deviation over the 10% time steps for which the LR field yielded the poorest performance across the flow scenarios in the test set. The aim is to assess the added value of the HR estimates, obtained either with downscaling or interpolation, with respect to using the LR field as is when the HR patterns deviate substantially from the spatial average and are thus more complex and more extreme.

Table 4 contains the three performance criteria averaged over the 10% time steps across all test scenarios for which the LR field yielded the worst performance along with the standard deviation in parentheses. The top (bottom) panel of

Table 4 is devoted to the water depth (unit discharge norm) indicator. The lower the RMSE and the MAE, the better the performance is. On the contrary, larger values of PSNR indicate a higher performance. Italic font highlights that a given HR estimates outperform the estimation provided by the LR field. Bold font signals which HR estimates has the best performance. The HR estimates from the downscaling model based on the lifting scheme always outperform the LR field and, most of the times, the differences are fairly large (such as in the Sacramento configuration). Moreover, the HR estimates based on the lifting scheme also outperform the HR estimates from the PCA based downscaling model in most instances, especially so in the Sacramento configuration. In contrast, the downscaling model based on PCA yields in some cases HR estimates that are worst than the LR field (this is the case for the water depth indicator in the Sacramento configuration). The inverse distance approach used in the synthetic oblique dam break configuration to interpolate the LR values to the HR mesh yielded a performance equivalent to the direct use of constant LR values over the sub-domains while the more sophisticated local-linear non-parametric regression used in the Sacramento configuration, in most cases, worsen the performance.

As a complement to these quantitative results, we illustrate the error $\hat{\boldsymbol{s}}_t - \boldsymbol{s}_t$ for a time step t that displays a complex and extreme spatial pattern. Figs. 9 and 10 show the original HR indicator field along with the spatial errors for the water depth and unit discharge norm indicators respectively made by the LR field and each downscaling model. These two figures concern the flow scenario (e) of the test set of the Sacramento configuration (see Table 2) at the time step corresponding to $t = 100$ s. In Fig. 10, the color scale for the error of the LR field is capped at the maximum error made by the downscaling models so that areas in gray (outside the color scale) indicate that the LR field was outperformed by both downscaling models. For both hazard indicators, it can be seen that the errors made by the two downscaling models are much lower than the ones made by the LR field although the gain of the downscaling models is far more important for the unit discharge norm shown in Fig. 10. In addition, the errors made by the downscaling model based on the lifting scheme are much smaller than the ones made by the downscaling model based on PCA for both hazard indicators.

4. Conclusion

We presented in this work two main methodological contributions. The first one is the adaptation of the lifting scheme to the spatio-temporal framework in order to exploit the dominant spatial features of natural phenomena such as simulated by physics-based models. To this end, the basic elements of the lifting scheme are taken to be the time series associated to each spatial location. This allows us to determine a spatial partition with a clustering algorithm so that each cluster or group may be thought of as a dominant spatial feature. These spatial features are thus defined as regions with similar temporal behavior. Once these dominant spatial features are established, the local regression implementing the wavelet transform can be applied straightforwardly within each feature. The local regression residuals contain the details of the signal, i.e., the wavelet coefficients. High wavelet coefficient values indicate the presence of non-linearities that occur along the temporal dimension. Compression can thus be achieved by eliminating spatial locations with small wavelet coefficient values at all the time steps. The more detailed the spatial partition, the more efficient the compression is (i.e., less wavelet coefficients are needed to represent the signal with a small loss). At each stage of the lifting scheme, the scaling coefficients obtained by a simple averaging operation provide a smoother version of the original signal. A non-parametric spatio-temporal representation can be derived from the adapted lifting scheme which is easy to interpret and to compute as it merely relies on additions and subtractions.

The second main methodological contribution of this work is the development of a downscaling model that allows to turn the representation derived from the spatio-temporal lifting scheme (or PCA) into a generative algorithm. In this case, the temporal coherence of the simulations is entirely driven by the LR fields. The key step in the downscaling model is to build a bank of spatial pattern types as defined by the wavelet coefficients (or by the weights in the linear combinations of eigenvectors in the case of PCA). Downscaling then consists in predicting the type of spatial patterns that occurs based on the LR fields. This discretization of the spatial patterns thus yields a classification task which may be much less difficult, especially if the number of classes is small, than attempting to predict directly the values of the wavelet coefficients (or the PCA weights), i.e., to perform a regression task. Although other choices might be necessary for other applications, in our cases study the classification was implemented with a classic one hidden layer feed-forward neural network with a skip layer connection which proved to be amply flexible (the training set was always perfectly classified with a low number of hidden units). The foremost assumption made in the downscaling strategy is that the spatial patterns contained

Table 4
Water depth hazard indicator (top panel) and unit discharge norm (bottom panel). Average and standard deviation in parentheses computed on the 10% time steps from the flow scenarios in the test set (see Table 1-2) for which the LR field performance is the poorest. Lower values of RMSE (23) and MAE (24), higher values of PSNR (25), indicate better performance. Italic font highlights that a given HR estimate outperformed the LR field while bold font signals which HR estimate has the best performance.

Water depth					
Configuration	Criterion	LR	HR estimates		
			Lifting scheme	PCA	Interp.
Dam break	RMSE	0.28 (0.063)	***0.13*** (0.046)	*0.15* (0.053)	*0.27* (0.059)
	MAE	0.21 (0.032)	***0.074*** (0.039)	*0.078* (0.041)	0.21 (0.029)
	PSNR	29 (11)	***36*** (12)	*34* (11)	*30* (11)
Sacramento	RMSE	0.26 (0.0013)	***0.061*** (0.025)	0.36 (0.39)	0.53 (0.017)
	MAE	0.18 (0.0025)	***0.035*** (0.015)	0.44 (0.44)	0.45 (0.0028)
	PSNR	25 (1.7)	***33*** (6.6)	23 (9)	19 (1.5)
Unit discharge norm					
Configuration	Criterion	LR	HR estimates		
			Lifting scheme	PCA	Interp.
Dam break	RMSE	3 (0.038)	*0.5* (0.15)	*0.5* (0.12)	3 (0.037)
	MAE	2.6 (0.084)	***0.37*** (0.11)	*0.38* (0.1)	2.6 (0.084)
	PSNR	15 (2)	***25*** (5.9)	*24* (5.2)	15 (2)
Sacramento	RMSE	1.8 (0.016)	***0.12*** (0.015)	*0.23* (0.057)	2.7 (0.025)
	MAE	1.3 (0.0063)	***0.079*** (0.0051)	*0.16* (0.028)	2.3 (0.021)
	PSNR	20 (3)	***33*** (6.1)	*26* (9)	17 (3.4)

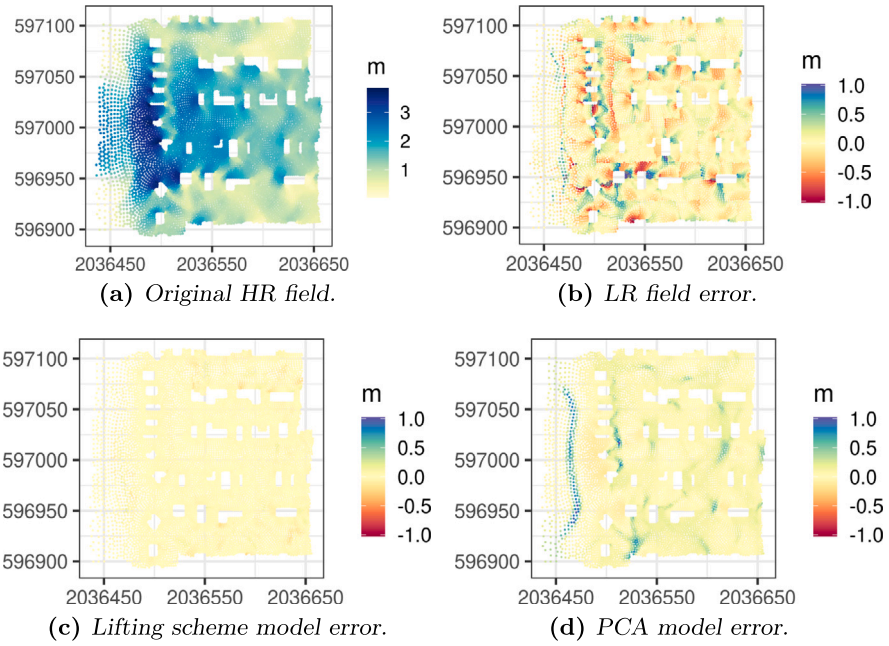


Fig. 9. Sacramento configuration, water depth for flow scenario (e) in Table 2. Original HR indicator and estimation errors $\hat{s}_t - s_t$ on time step $t = 100$ s.

in the training set are sufficiently representative to constitute the bank of pattern types. Indeed, the classifier is only able to predict a spatial pattern type that has been seen in the training set. To generate new spatial patterns, a possibility would be to use a mixture of existing spatial patterns (e.g., by using a probabilistic classifier). Nevertheless, this downscaling model provides a rather innovative and intuitive approach in order to take into account the spatial structure.

Both methodological contributions may be used with large spatio-temporal data sets such as the simulations of physics-based models. They could also be applied to observational data sets which would be, in general, much less dense spatially and often not very deep temporally. The core underlying rationale behind the approaches is to exploit recurrent spatial patterns in the data. Both data sets considered in this work are available upon request. There are two rather large HR hydraulic data sets containing either 9749 or 18432 spatial locations (i.e., cells in the HR meshes of the Sacramento or dam break configuration respectively) and either 49 or 16 time steps. The spatial dimension of the LR data sets, i.e., the number of sub-domains, is either 199 or 8. In simple words, the downscaling model accomplishes a regression with inputs that are of very low dimension (either 199 or 8) with respect to its outputs (either 9749 or 18432). To face this challenging task, a modeling assumption such as the representativity of the spatial patterns contained in the training set is necessary. This is, in our opinion, a rather mild assumption since the spatial structure is modeled with great flexibility with the non-parametric representations derived from the lifting scheme or PCA. In the hydraulic data sets considered, the spatial patterns are highly complex (non isotropic and non-stationary) so that conventional statistical approaches such as kriging would be challenged. However, the HR meshes are very dense and the spatial patterns in the data are rather smooth. In applications with less densely distributed data and more irregular or jagged spatial patterns, the

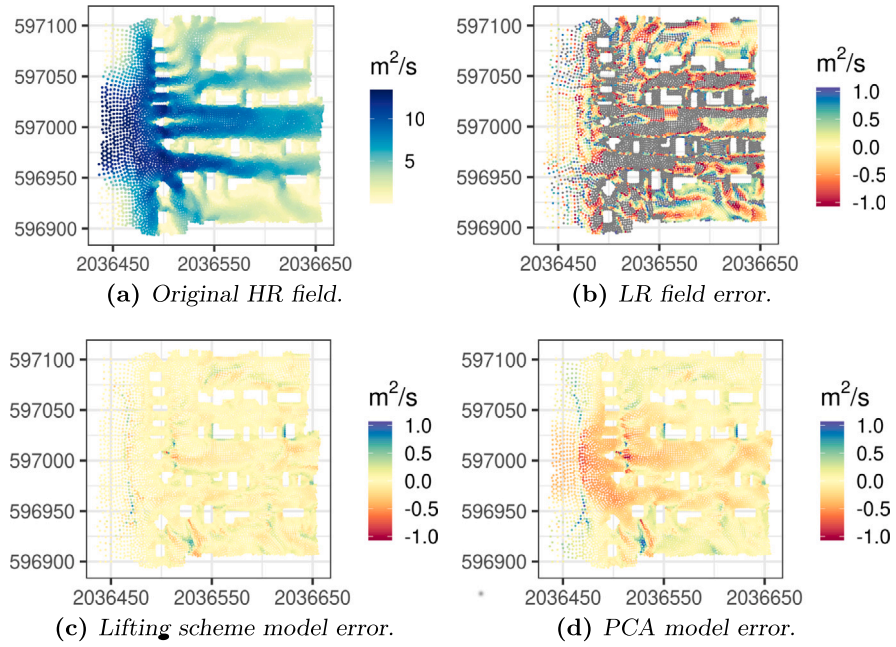


Fig. 10. Sacramento configuration, unit discharge norm for flow scenario (e) in Table 2. Original HR indicator and estimation errors $\hat{s}_t - s_t$ on time step $t = 100$ s. Gray areas in the LR field error indicate errors larger than the ones made by both downscaling models. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

proposed approaches may not performed as well and may require some adaptations. All the algorithms are implemented in the R language and were applied to the aforementioned data sets on an ordinary laptop. Although the lifting scheme implementation could certainly be optimized to run faster, results (including the selection of hyper-parameter values) were obtained within a few hours. The code will be made available as an official R library in the near future.

There are many possible perspectives to this work. Concerning the lifting scheme, instead of operating on vectors of time series, we could consider to operate truly in the spatio-temporal space, such as in spatio-temporal geostatistics, e.g., Montero et al. (2015). In other words, the local regression from (5) and the update step from (7) could be applied to neighbors in that space. Another issue to look at would be to model several variables, e.g., the water depth and the unit discharge norm, jointly. Besides, other possible invertible simple local regression models instead of the naive local regression could be considered. Finally, the clustering assisted split step might need to be fine-tuned (in terms of which clustering algorithm to apply and what rate of decrease to select for the number of clusters, see Algorithm 1) to account for the complexity level of the spatial structure in the data. Concerning the downscaling strategy, we could seek to explicitly model the temporal dependence with, for instance, an auto-regressive scheme, and to introduce stochasticity in order to provide uncertainty assessment. Furthermore, it only addresses spatial disaggregation as we considered the perfect upscaling case. Thus, it would be interesting to evaluate the consequences of more realistic imperfect upscaling cases, in particular, whether a form of bias correction would be necessary (e.g., François et al. (2021)).

Funding

The first author was supported by the starting grants from Polytechnique Montreal, Canada and IVADO, Canada. In addition, she acknowledges the support of the Natural Sciences and Engineering Research Council of Canada (NSERC), [funding reference number RGPIN-2022-0405]. The second author was supported in part by the program 80 PRIME CNRS-INSU. He also acknowledges the support of the French Agence Nationale de la Recherche (ANR) under the T-REX (ANR-20-CE40-0025-01) and Melody (ANR-19-CE46-0011) projects.

CRediT authorship contribution statement

Julie Carreau: Conceptualization, Methodology, Software, Validation, Formal analysis, Writing – original draft, Writing – review & editing, Visualization. **Philippe Naveau:** Conceptualization, Methodology, Validation, Formal analysis, Writing – original draft, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

References

Ayar, P.V., Vrac, M., Bastin, S., Carreau, J., Déqué, M., Gallardo, C., 2016. Inter-comparison of statistical and dynamical downscaling models under the EURO-and MED-CORDEX initiative framework: Present climate evaluations. Clim. Dynam. 46 (3–4), 1301–1329.

Bishop, C.M., 2006. Pattern Recognition and Machine Learning. In: Information Science and Statistics, Springer.

Caillaud, C., Somot, S., Alias, A., et al., 2021. Modelling Mediterranean heavy precipitation events at climate scale: An object-oriented evaluation of the CNRM-AROME convection-permitting regional climate model. Clim. Dynam. 56, 1717–1752.

Cannon, A.J., 2018. Multivariate quantile mapping bias correction: An N-dimensional probability density function transform for climate model simulations of multiple variables. Clim. Dynam. 50 (1–2), 31–49.

Carreau, J., Guinot, V., 2021. A PCA spatial pattern based artificial neural network downscaling model for urban flood hazard assessment. Adv. Water Resour. 147, 103821.

Carreau, J., Naveau, P., Neppel, L., 2017. Partitioning into hazard subregions for regional peaks-over-threshold modeling of heavy precipitation. Water Resour. Res. 53 (5), 4407–4426.

Daubechies, I., 1992. Ten Lectures on Wavelets. SIAM.

Daubechies, I., Sweldens, W., 1998. Factoring wavelet transforms into lifting steps. J. Fourier Anal. Appl. 4 (3), 247–269.

Dong, W., Shi, G., Xu, J., 2008. Signal-adapted directional lifting scheme for image compression. In: 2008 IEEE International Symposium on Circuits and Systems. ISCAS, IEEE, pp. 1392–1395.

Fraehr, N., Wang, Q.J., Wu, W., Nathan, R., 2022. Upskilling low-fidelity hydrodynamic models of flood inundation through spatial analysis and Gaussian process learning. Water Resour. Res. e2022WR032248, e2022WR032248 2022WR032248.

François, B., Thao, S., Vrac, M., 2021. Adjusting spatial dependence of climate model outputs with cycle-consistent adversarial networks. Clim. Dynam. 57, 3323–3353.

Guinot, V., Sanders, B.F., Schubert, J.E., 2017. Dual integral porosity shallow water model for urban flood modelling. Adv. Water Resour. 103, 16–31.

Hayfield, T., Racine, J., 2008. Nonparametric econometrics: The np package. J. Stat. Softw. 27 (5), 1–32.

Jensen, A., La Cour-Harbo, A., 2001. Ripples in Mathematics : The Discrete Wavelet Transform. Springer.

Kaaniche, M., Pesquet, J.-C., Benazza-Benyahia, A., Pesquet-Popescu, B., 2010. Two-dimensional non separable adaptive lifting scheme for still and stereo image coding. In: 2010 IEEE International Conference on Acoustics, Speech and Signal Processing. IEEE, pp. 1298–1301.

Kashinath, K., Mustafa, M., Albert, A., Wu, J.-L., Jiang, C., Esmaeilzadeh, S., Azizzadenesheli, K., Wang, R., Chattopadhyay, A., Singh, A., Manepalli, A., Chirila, D., Yu, R., Walters, R., White, B., Xiao, H., Tchelepi, H.A., Marcus, P., Anandkumar, A., Hassanzadeh, P., Prabhat, n., 2021. Physics-informed machine learning: Case studies for weather and climate modelling. Phil. Trans. R. Soc. A 379 (2194), 20200093.

Kaufman, L., Rousseeuw, P.J. (Eds.), 1990. Finding Groups in Data: An Introduction to Cluster Analysis. John Wiley & Sons, Ltd.

Korhonen, J., You, J., 2012. Peak signal-to-noise ratio revisited: Is simple beautiful? In: 2012 Fourth International Workshop on Quality of Multimedia Experience. pp. 37–38.

Kumar, Y.P., Maheswaran, R., Agarwal, A., Sivakumar, B., 2021. Intercomparison of downscaling methods for daily precipitation with emphasis on wavelet-based hybrid models. J. Hydrol. 599, 126373.

Mallat, S., 1999. A Wavelet Tour of Signal Processing, second ed. Academic Press, pp. I–XXIV, 1–637.

Montero, J.M., Fernàndez-Avilés, G., Mateu, J., 2015. Spatial and Spatio-Temporal Geostatistical Modeling and Kriging. Wiley.

Nourani, V., Farshbaf, A., Adarsh, S., 2020. Spatial downscaling of radar-derived rainfall field by two-dimensional wavelet transform. Hydrol. Res. 51 (3), 456–469.

Nychka, D., Hammerling, D., Krock, M., Wiens, A., 2018. Modeling and emulation of nonstationary Gaussian fields. Spat. Stat. 28, 21–38, One world, one health.

Park, S., Oh, H.-S., 2022. Lifting scheme for streamflow data in river networks. J. R. Stat. Soc. Ser. C. Appl. Stat. 71 (2), 467–490.

Rezvov, V., Krinitskiy, M., Gavrikov, A., Gulev, S., 2021. Comparison of AI-based approaches for statistical downscaling of surface wind fields in the north atlantic. In: VI International Conference Information Technologies and High-Performance Computing. ITHPC-2021, p. September.

Sanders, B., Schubert, J., 2019. PRIMo: Parallel raster inundation model. Adv. Water Resour. 126, 79–95.

Sweldens, W., 1998. The lifting scheme: A construction of second generation wavelets. SIAM J. Math. Anal. 29 (2), 511–546.

Venables, W.N., Ripley, B.D., 2002. Modern Applied Statistics with S, fourth ed. Springer, New York, ISBN: 0-387-95457-0.

Vrac, M., 2018. Multivariate bias adjustment of high-dimensional climate simulations: The rank resampling for distributions and dependences (R^2D^2) bias correction. Hydrol. Earth Syst. Sci. 22 (6), 3175–3196.

Vrac, M., Friederichs, P., 2015. Multivariable - intervariable, spatial and temporal - bias correction. J. Clim. 28 (1), 218–237.

Wu, Y., Teufel, B., Sushama, L., Belair, S., Sun, L., 2021. Deep learning-based super-resolution climate simulator-emulator framework for urban heat studies. Geophys. Res. Lett. 48 (19), e2021GL094737, e2021GL094737 2021GL094737.

Zheng, N., Xue, J., 2009. Statistical Learning and Pattern Analysis for Image and Video Processing, first ed. In: Advances in Computer Vision and Pattern Recognition, Springer London.