Physically interpretable machine learning algorithm on multidimensional non-linear fields

Rem-Sophia Mouradi, Cédric Goeury, Olivier Thual, Fabrice Zaoui, Pablo Tassi

Journal of Computational Physics 428 (2021) 110074

Contents lists available at ScienceDirect

Journal of Computational Physics

www.elsevier.com/locate/jcp

Physically interpretable machine learning algorithm on multidimensional non-linear fields

Rem-Sophia Mouradi [a,b,*], Cédric Goeury [a], Olivier Thual [b,c], Fabrice Zaoui [a], Pablo Tassi [a,d]

[a] EDF R&D, National Laboratory for Hydraulics and Environment (LNHE), 6 Quai Watier, 78400 Chatou, France
[b] Climate, Environment, Coupling and Uncertainties research unit (CECI) at the European Center for Research and Advanced Training in Scientific Computation (CERFACS), French National Research Center (CNRS), 42 Avenue Gaspard Coriolis, 31820 Toulouse, France
[c] Institut de Mécanique des Fluides de Toulouse (IMFT), Université de Toulouse, CNRS, Toulouse, France
[d] Saint-Venant Laboratory for Hydraulics (LHSV), Chatou, France

ARTICLE INFO

Article history:
Received 25 May 2020
Received in revised form 6 November 2020
Accepted 10 December 2020
Available online 7 January 2021

Keywords:
Data-Driven Model (DDM)
Proper Orthogonal Decomposition (POD)
Dimensionality Reduction (DM)
Polynomial Chaos Expansion (PCE)
Machine Learning (ML)
Geosciences

ABSTRACT

In an ever-increasing interest for Machine Learning (ML) and a favorable data development context, we here propose an original methodology for data-based prediction of two-dimensional physical fields. Polynomial Chaos Expansion (PCE), widely used in the Uncertainty Quantification community (UQ), has long been employed as a robust representation for probabilistic input-to-output mapping. It has been recently tested in a pure ML context, and shown to be as powerful as classical ML techniques for point-wise prediction. Some advantages are inherent to the method, such as its explicitness and adaptability to small training sets, in addition to the associated probabilistic framework. Simultaneously, Dimensionality Reduction (DR) techniques are increasingly used for pattern recognition and data compression and have gained interest due to improved data quality. In this study, the interest of Proper Orthogonal Decomposition (POD) for the construction of a statistical predictive model is demonstrated. Both POD and PCE have amply proved their worth in their respective frameworks. The goal of the present paper was to combine them for a field-measurement-based forecasting. The described steps are also useful to analyze the data. Some challenging issues encountered when using multidimensional field measurements are addressed, for example when dealing with few data. The POD-PCE coupling methodology is presented, with particular focus on input data characteristics and training-set choice. A simple methodology for evaluating the importance of each physical parameter is proposed for the PCE model and extended to the POD-PCE coupling.

1. Introduction

Deep Learning techniques (DL [1]) and more generally Machine Learning (ML [2]), and their applications to physical problems (fluid mechanics [3]; plasma physics [4]; quantum mechanics [5], etc.) have made a promising take-off in the last few years. This has been particularly the case for fields where the measurement potential has dramatically increased (e.g. Geoscience Data [6]). In this context, learning techniques are of interest to establish non-linear physical relationships from

* Corresponding author at: EDF R&D, National Laboratory for Hydraulics and Environment (LNHE), 6 Quai Watier, 78400 Chatou, France.
E-mail address: remsophia.mouradi@gmail.com (R.-S. Mouradi).

https://doi.org/10.1016/j.jcp.2020.110074

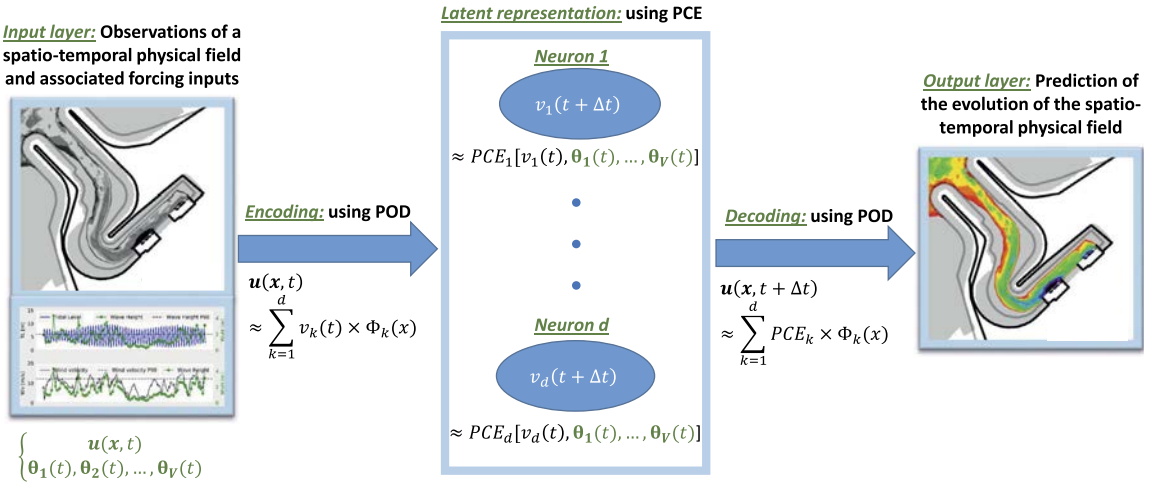


Fig. 1. Representation of the POD-PCE ML approach.

the data by a combination of steps, in particular using transformation functions, to capture the complexity of the system [2].

In particular, multi-layer Neural Networks (NN) [7] are widely used for physical applications. Their popularity comes from this complex structure, which makes them adaptable for various applications [8,9]. However, some limitations prevent the use of NN for physical applications: (i) it is difficult to provide an explicit input-to-output formulation, due to the combinations of steps involved in the learning (*Activation Functions*, *Hidden Layers* [1]). Physical interpretation of the constructed model is therefore tedious [10]; (ii) too many hyper-parameters and choices are involved, depending on the number of neurons and layers (*curse of dimensionality*) [11]; (iii) no general proof for the theoretical ability of approximating arbitrary functions is available, except the *Universal Approximation Theorem* and its extensions [12,13] for particular cases.

To overcome these limitations, we propose an alternative ML method, based on a coupling between Proper Orthogonal Decomposition (POD) [14] and Polynomial Chaos Expansion (PCE) [15,16]. This approach is proposed for the prediction of spatially-distributed physical fields that vary in time. The idea is to use POD to separate the spatial patterns from the temporal variations, that are related to the conditioning parameters using PCE. To correspond to common NN paradigms, an adequate representation of this idea is given in Fig. 1. In particular, POD is used for both *Encoding* and *Decoding* whereas PCE is used as an *Activation Function* in the *Latent Representation* [1].

The proposed POD-PCE addresses these drawbacks of ML.

(*i*) It is explicit and simple to implement, as it consists of the association of two linear decompositions. POD is a linear separation of the spatiotemporal patterns [17], shown to be accurate for both linear and non-linear problems [18], combining simplicity and relevance. PCE is a well-established method in Uncertainty Quantification (UQ) [19,20], widely used for the study of stochastic behavior in physics [21,22]. It is a linear polynomial expansion that allows non-linearities to be gradually added to the model by increasing the polynomial degree. The linearity and orthonormality of the POD and PCE components and the probabilistic framework of PCE make the output's statistical moments easier to study [23], enabling straightforward physical interpretation of the model [24].

(*ii*) It only has two hyper-parameters: a number of POD components, and a PCE polynomial degree. Both can be chosen according to quantitative criteria [14,25]. All other forms of parameterization (choice of the polynomial basis) can be achieved with robust physical and/or statistical arguments [26], as assessed in the present paper. Furthermore, the orthonormality of the POD and PCE bases minimizes the number of components necessary to capture essential variations in data. Additionally, the POD modes capture more energy than any other decomposition [27], PCE is known to exponentially converge with polynomial degree [16], and the cardinality of the latter can be reduced by sparse basis selection [28].

(*iii*) It can be considered as a universal expansion for physical field approximation: a physical field has a finite variance, which implies that it belongs to the Hilbert space of random variables with finite second order moments. There therefore exists a numerable set of orthogonal random variables, that form the basis of this Hilbert space, on which the field of interest can be expanded (strict equality, not approximation) [20]. A mathematical setting for basis construction based on input was established by Soize and Ghanem [26] for the general case of dependent variables with arbitrary density, provided that the set of inputs is finite.

In the literature, associating regression techniques to Reduced Order Models (ROM), that include POD, is not novel [29, 30]. The cited studies, however, focused on dimensionality reduction, whereas the explicit formulation and applicability to complex physical processes are emphasized in the present study. Secondly, coupling PCE to ROM was recently addressed

[31,32] and the use of PCE as ML is consistent with the work of Torre et al. [33], where the authors showed that PCE is as powerful as classical ML techniques. However, neither spatiotemporal fields nor physical interpretability were addressed. The data in these studies were either obtained from numerical experiments, emulated from analytical benchmark functions such as Sobol or Ishigami, or based on one-dimensional data sets [33]. In contrast, the proposed POD-PCE methodology is herein assessed on two-dimensional physical fields. In particular, a toy example where synthetic data are emulated using an analytical function (groundwater perturbations due to tidal loadings [34]), and a real data set (high-resolution field measurements of underwater topography) are used. Although similar from a learning point of view, these two applications are characterized with differences. In particular, the toy problem is purely parametric and controllable, whereas the real data concern temporal dynamics and are of limited size. The cases are therefore complementary, in the sense that they allow demonstrating different properties of the proposed methodology. Hence, using the particularities of each case, the study consists in: i) the evaluation of the combined use of POD and PCE as ML for point-wise prediction; ii) the robustness of the methodology to noise; iii) the application to field data with the inherent challenges not encountered with numerical data (e.g. paucity); iv) a focus on model explicitness as a key condition for physical understanding and v) the influence of forcing variables study, based on a classical measure of importance (Garson weights [35]) directly computed with the POD-PCE expansion coefficients.

The paper is organized as follows. Section 2 gives a detailed explanation of the methodology, with a proposal for physical importance measures in Subsection 2.2.2. Section 3 deals with the assessment of the methodology on synthetic data, for both prediction and physical interpretation. In particular, the robustness of the approach to noise is evaluated in Subsection 3.3. The model is then deployed on field measurements in Section 4. The study case and data are described in 4.1. POD and PCE performances are then demonstrated independently in 4.2 with a deep physical analysis. The performance of the POD-PCE predictor is discussed in 4.3. A summary of the study and perspectives of the proposed methodology are presented in Section 5.

2. Theoretical framework

In this section, the objective is to define the framework of the proposed POD-PCE Machine Learning methodology, along with physical influence indicators for the inputs. This is the object of Subsection 2.3, but first, a reminder of the existing POD and PCE theoretical bases is presented in 2.1 and 2.2 respectively.

2.1. Proper Orthogonal Decomposition

POD is a dimensionality reduction technique [17] that is well documented in literature [14,18]. Theoretical details and demonstrations can be found in [27,36]. For clarity's sake, the essential elements of POD are summarized below.

The goal of POD is to extract the main patterns of continuous bi-variate functions. These patterns, when added and multiplied by appropriate coefficients, explain the dynamics of the variable of interest: a real-valued physical field.

Let $\mathbf{u} : \Omega \times \mathbb{T} \to \mathbb{D}$ be a continuous function of two variables $(\mathbf{x}, t) \in \Omega \times \mathbb{T}$. The following relationships and properties hold for any $\Omega \times \mathbb{T}$ and Hilbert space $\mathbb{D}$ characterized by its scalar product $(.\ ,.)_{\mathbb{D}}$ and induced norm $||.||_{\mathbb{D}}$. However, as is the case for a majority of physical fields, we shall consider Ω as a set of spatial coordinates (e.g. $\mathbb{R}^2$ or $\mathbb{R}^3$), $\mathbb{T}$ an event space (e.g. parameters space $\mathbb{R}^V$ with $V \in \mathbb{N}^*$, or a temporal subset $[0, T] \subseteq \mathbb{R}^+$), and $\mathbb{D}$ as a set of scalar real values or vector real values (e.g. $\mathbb{R}$ or $\mathbb{R}^2$). POD consists in an approximation of $\mathbf{u}(\mathbf{x}, t)$ at a given order $d \in \mathbb{N}$ (Lumley [17]) as in Equation (1),

$$\mathbf{u}(\mathbf{x}, t) \approx \sum_{k=1}^{d} v_k(t)\boldsymbol{\phi}_k(\mathbf{x}) \ , \tag{1}$$

where $\{v_k(.)\}_{k=1}^d \subset \mathcal{C}(\mathbb{T}, \mathbb{R})$ and $\{\boldsymbol{\phi}_k(.)\}_{k=1}^d \subset \mathcal{C}(\Omega, \mathbb{D})$, with $\mathcal{C}(\mathbb{A}, \mathbb{B})$ denoting the space of continuous functions defined over $\mathbb{A}$ and arriving at $\mathbb{B}$. The objective of POD is to identify $\{\boldsymbol{\phi}_k(.)\}_{k=1}^d$ that minimizes the distance of the approximation from the true value $\mathbf{u}(.,.)$, over the whole $\Omega \times \mathbb{T}$ domain, with an orthogonality constraint for $\{\boldsymbol{\phi}_k(.)\}_{k=1}^d$ using the scalar product $(.\ ,.)_{\mathbb{D}}$. This can be defined, in the least-squares sense, as a minimization problem.

The minimization problem is defined for all orders $d \in \mathbb{N}$, so that the members $\boldsymbol{\phi}_k$ are ordered according to their importance. In particular, for order 1, $\boldsymbol{\phi}_1$ is the linear generator of the sub-vector space most representative of $\mathbf{u}(\mathbf{x}, t)$ in $\mathbb{D}$. For $\mathbb{D} = Im(\mathbf{u})$, the family $\{\boldsymbol{\phi}_k(.)\}_{k=1}^d$ is called the POD basis of $\mathbb{D}$ of rank d. The solution to this problem has already been established in literature [17,37]. The theoretical aspects of POD and demonstrations of mathematical properties can, for example, be found in [27]: the POD basis of $\mathbb{D}$ of order d is the orthonormal set of eigenvectors of an operator $\mathcal{R} : \mathbb{D} \to \mathbb{D}$ defined as $\mathcal{R}\boldsymbol{\phi} = \langle(\mathbf{u}, \boldsymbol{\phi})_{\mathbb{D}} \times \mathbf{u}\rangle_{\mathbb{T}}$, if the eigenvectors are taken in decreasing order of the corresponding eigenvalues $\{\lambda_k\}_{k=1}^d$.

For this expansion, an accuracy rate, also called the Explained Variance Rate (EVR), denoted e_d at rank d, can be calculated as in Equation (2). EVR tends to 1 (perfect approximation) when $d \to +\infty$.

$$e_d = \frac{\sum_{k \leq d} \lambda_k}{\sum_{k=1}^{+\infty} \lambda_k} \ . \tag{2}$$

In practice, for $\mathbb{D}=\mathbb{R}$, when $\mathbf{u}(.,.)$ is a discrete sample on a set of $m\in\mathbb{N}$ space coordinates $\boldsymbol{\mathcal{X}}=\{\mathbf{x}_1,\dots,\mathbf{x}_m\}$ and for $n\in\mathbb{N}$ measurement events $\mathcal{T}=\{t_1,\dots,t_n\}$ (e.g. realizations of the parameters, time coordinates, etc.), the available data set is arranged in a matrix $\mathbf{U}(\mathcal{X},\mathcal{T})=[\mathbf{u}(\mathbf{x}_i,t_j)]_{i,j}\in\mathbb{R}^{m\times n}$, called the snapshot matrix, so as to be able to work in a discrete space. The POD problem formulated in Equation (1) can be written in its discrete form as $\mathbf{U}(\boldsymbol{\mathcal{X}},\mathcal{T})=\boldsymbol{\Phi}^{(d)}(\boldsymbol{\mathcal{X}})\mathbf{V}^{(d)}(\mathcal{T})$, where $\boldsymbol{\Phi}^{(d)}(\boldsymbol{\mathcal{X}}):=[\boldsymbol{\phi}_j(\mathbf{x}_i)]_{i,j}\in\mathbb{R}^{m\times d}$ and $\mathbf{V}^{(d)}(\mathcal{T}):=[v_i(t_j)]_{i,j}\in\mathbb{R}^{d\times n}$. The problem can therefore be viewed as if working with a new function $\mathbf{U}(\boldsymbol{\mathcal{X}},.)=[\mathbf{u}(\mathbf{x}_i,.)]_{i\in\{1,\dots,m\}}:\mathcal{T}\to\mathbb{D}=\mathbb{R}^M$. Then, the average over $\mathbb{T}$ can be defined as the statistical mean over the subset $\mathcal{T}$, and the scalar product $(.\ ,.)_{\mathbb{D}}$ as the canonical product over $\mathbb{R}^m$. The POD operator $\mathcal{R}$ can be written as in Equation (3),

$$\mathcal{R}\boldsymbol{\phi}(\boldsymbol{\mathcal{X}})=\frac{1}{n}\sum_{j=1}^{n}\mathbf{U}(\boldsymbol{\mathcal{X}},t_j)^T\boldsymbol{\Phi}(\boldsymbol{\mathcal{X}})\mathbf{U}(\boldsymbol{\mathcal{X}},t_j)=\frac{1}{n}\mathbf{U}(\boldsymbol{\mathcal{X}},\mathcal{T})\mathbf{U}(\boldsymbol{\mathcal{X}},\mathcal{T})^T\boldsymbol{\Phi}(\boldsymbol{\mathcal{X}})\ , \tag{3}$$

where $\mathbf{U}(\mathcal{X},t_j)=[\mathbf{u}(\mathbf{x}_i,.)]_{i\in\{1,\dots,m\}}$ is the column number j of the matrix $\mathbf{U}(\boldsymbol{\mathcal{X}},\mathcal{T})$ (i.e realization t_j of the measurement over $\boldsymbol{\mathcal{X}}$), and $\boldsymbol{\Phi}(\mathcal{X})=[\boldsymbol{\phi}(\mathbf{x}_i)]_{i\in\{1,\dots,m\}}$. As finding the POD basis is equivalent to identifying the orthonormal set of eigenvectors of the operator $\mathcal{R}$, then for this discrete representation the problem becomes equivalent to solving the eigen problem of the matrix $\mathbf{R}:=\frac{1}{n}\mathbf{U}(\boldsymbol{\mathcal{X}},\mathcal{T})\mathbf{U}(\boldsymbol{\mathcal{X}},\mathcal{T})^T$, called the covariance matrix. A number $d\in\mathbb{N}$ of eigen vectors $\boldsymbol{\Phi}(\boldsymbol{\mathcal{X}})$ are identified and stored in the columns of the matrix $\boldsymbol{\Phi}^{(d)}(\boldsymbol{\mathcal{X}})$. For the eigenvalues of the covariance matrix $\mathbf{R}$ denoted $\{\lambda_k\}_{k=1}^d$, the expansion in Equation (1) can also be written as in Equation (4), where $\{\boldsymbol{\phi}_k(.)\}_{k=1}^d$ together with $\{a_k(.)\}_{k=1}^d$ are bi-orthonormal, and $v_k(.)=a_k(.)\sqrt{n\times\lambda_k}$.

$$\mathbf{u}(\mathbf{x},t)\approx\sum_{k=1}^{d}a_k(t)\sqrt{n\times\lambda_k}\boldsymbol{\phi}_k(\mathbf{x})\ . \tag{4}$$

By defining the matrix $\mathbf{A}^{(d)}(\mathcal{T}):=[a_i(t_j)]_{i,j}\in\mathbb{R}^{d\times n}$ and the operator $\mathbf{D}^{(d)}(\lambda_1,...,\lambda_d)$ corresponding to the diagonal matrix of elements λ_i, we have $\mathbf{U}(\boldsymbol{\mathcal{X}},\mathcal{T})=\boldsymbol{\Phi}^{(d)}(\boldsymbol{\mathcal{X}})\mathbf{D}^{(d)}(\sqrt{n\times\lambda_1},...,\sqrt{n\times\lambda_d})\mathbf{A}^{(d)}(\mathcal{T})$. Therefore the transposed form is $\mathbf{U}(\boldsymbol{\mathcal{X}},\mathcal{T})^T=\mathbf{A}^{(d)}(\mathcal{T})^T\mathbf{D}^{(d)}(\sqrt{n\times\lambda_1},...,\sqrt{n\times\lambda_d})\boldsymbol{\Phi}^{(d)}(\boldsymbol{\mathcal{X}})^T$. Thanks to the orthonormality of $\{a_k(.)\}_{k=1}^d$, the covariance matrix reads $\mathbf{R}=\frac{1}{n}\boldsymbol{\Phi}^{(d)}(\boldsymbol{\mathcal{X}})\mathbf{D}^{(d)}(n\times\lambda_1,...,n\times\lambda_d)\boldsymbol{\Phi}^{(d)}(\boldsymbol{\mathcal{X}})^T=\boldsymbol{\Phi}^{(d)}(\boldsymbol{\mathcal{X}})\mathbf{D}^{(d)}(\lambda_1,...,\lambda_d)\boldsymbol{\Phi}^{(d)}(\boldsymbol{\mathcal{X}})^T$.

When $n<<m$, it is more computationally efficient to solve the eigenproblem of $\mathbf{R}^T$ instead of the eigenproblem of $\mathbf{R}$ as highlighted by Sirovich [37]. This is often the case when a limited number of occurrences is measured for a two-dimensional physical field, as is the case encountered for the application described in Section 4.

When an order $d<<min(m,n)$ corresponds to a high EVR as defined in Equation (2), we speak of dimensionality reduction, because the data are projected in a sub-space that is of much smaller dimension than $\mathbb{R}^{m\times n}$. When diverse enough records are available for the variable under study, we may consider that $\{\boldsymbol{\phi}_k(\boldsymbol{\mathcal{X}})\}_{k=1}^d=\{[\boldsymbol{\phi}_k(\mathbf{x}_i)]_{i\in\{1,\dots,m\}}\}_{k=1}^d$, i.e. the resulting POD basis, is a generator of all possible states. Predicting the associated expansion coefficients $\{a_k(t)\}_{k=1}^d$ for a given event t would therefore be enough to predict the whole state. Hence, we propose to use the POD as a basis extractor. This would first enable study of the dynamics of the variable of interest and eventually extraction of physical information, as shown in the applications Sections 4 and 3. Then, the basis can be used as a generator for the prediction of diverse states. This implies predicting $\{a_k(t)\}_{k=1}^d$, for which we propose to use Polynomial Chaos Expansion (PCE), as described in the following Section 2.2.

2.2. Polynomial Chaos Expansion

A reminder of the theoretical base of PCE is presented in Subsection 2.2.1. Theoretical details, demonstrations and interesting references can be found in [23,19]. Then, a simple indicator is proposed in Subsection 2.2.2 for the analysis of the variables influence on the output value. The latter is later generalized for POD-PCE in Section 2.3.

2.2.1. Learning

The idea behind Polynomial Chaos Expansion (PCE) is to formulate an explicit model that links a variable of interest (output) to conditioning parameters (inputs), both in a probability space. This enables the propagation path of probabilistic information (uncertainties, occurrence frequencies) to be mapped from the input to the output space. The variable of interest, $\mathbf{Y}$, and the input parameters $\boldsymbol{\Theta}=(\theta_1,\theta_2,...,\theta_V)$ are therefore considered random variables, characterized by a given Probability Density Function (PDF) denoted $f_{\boldsymbol{\Theta}}$. It should be kept in mind that the outputs of our problem are the POD expansion coefficients $\mathbf{Y}=[a_k(t)]_{k\in\{1,\dots,d\}}$, and that the inputs correspond to physical forcings, as described later in Section 2.3. The objective is to derive the variations of the POD coefficients as the outcome of the forcings. Let us now recall some fundamentals of the mathematical probabilistic framework, taking the example of a one dimensional real-valued variable. The definitions can be easily extended to $\mathbb{R}^M$.

Let $(\Omega,F,\mathbb{P})$ be a probability space, where Ω is the event space (space of all the possible events ω) equipped with σ-algebra F (some events of Ω) and its probability measure $\mathbb{P}$ (likelihood of a given event occurrence). A random variable defines an application $Y(\omega):\Omega\to D_Y\subseteq\mathbb{R}$, with realizations denoted by $y\in D_Y$. The PDF of Y is a function $f_Y:D_Y\to\mathbb{R}$

that verifies $\mathbb{P}(Y \in E \subseteq D_Y) = \int_E f_Y(y)dy$. The *kth moments* of Y are defined as $\mathbb{E}[Y^k] := \int_{D_Y} y^k f_Y(y)dy$, the first being the expectation denoted $\mathbb{E}[Y]$. In the same manner, we define the *kth central moments* of Y as $\mathbb{E}[(Y - \mathbb{E}[Y])^k]$, the first being 0 and the second the variance of Y denoted by $\mathbb{V}[Y]$. The covariance of two random variables is defined as $cov(X, Y) = \mathbb{E}[(X - \mathbb{E}[X])(Y - \mathbb{E}[Y])]$ and a resulting property is $\mathbb{V}[Y] = cov(Y, Y)$.

Returning to the PCE construction, inputs $\mathbf{\Theta} = (\theta_1, \theta_2, ..., \theta_V)$ are considered to live in the space of real random variables with finite second moments (and finite variances). This space is denoted by $\mathcal{L}^2_{\mathbb{R}}(\Omega, F, \mathbb{P}; \mathbb{R})$ and is a Hilbert space equipped with the inner product $(\theta_1, \theta_2)_{\mathcal{L}^2_{\mathbb{R}}} := \mathbb{E}[\theta_1\theta_2] = \int_\Omega \theta_1(\omega)\theta_2(\omega)d\mathbb{P}(\omega)$ and its induced norm $||\theta_1||_{\mathcal{L}^2_{\mathbb{R}}} := \sqrt{\mathbb{E}[\theta_1^2]}$. The PCE objective is to map the output space from the input space with a model $\mathcal{M}$ as in Equation (5):

$$\begin{aligned} Y &= \mathcal{M}(\mathbf{\Theta}) = \textstyle\sum_{\mathcal{I}\subseteq\{1,...,V\}} \mathcal{M}_{\mathcal{I}}(\theta_{\mathcal{I}}) \\ &= \mathcal{M}_0 + \textstyle\sum_{i=1}^{V} \mathcal{M}_i(\theta_i) + \sum_{1\le i<j\le V} \mathcal{M}_{i,j}(\theta_i, \theta_j) + ... + \mathcal{M}_{1,...,V}(\theta_1, \theta_2, ..., \theta_V)\,, \end{aligned} \tag{5}$$

where $\mathcal{M}_0$ is the expectation of Y and $\mathcal{M}_{\mathcal{I}\subseteq\{1,...,V\}}$ represents the common contribution of the variables $\mathcal{I} \subseteq \{1, ..., V\}$ to the variation in Y. For the PCE model, these contributions have a polynomial form. We shall define, for each input variable θ_i, an orthonormal univariate polynomial basis $\left\{\xi^{(i)}_\beta(.), \beta \in [|0, p|]\right\}$ where $p \in \mathbb{N}$ is a chosen polynomial degree and $\xi^{(i)}_\beta(.)$ is of degree β. The orthonormality is defined with respect to the inner product $(.,.)_{\mathcal{L}^2_{\mathbb{R}}}$. If we introduce the multi-index notation $\boldsymbol{\alpha} = (\alpha_1, ..., \alpha_V) \in \mathbb{N}^V$ so that $|\boldsymbol{\alpha}| = \sum_{i=1}^V \alpha_i$, we can define a multivariate basis $\left\{\zeta^{\mathbf{\Theta}}_{\boldsymbol{\alpha}}(.), |\boldsymbol{\alpha}| \in [|0, p|]\right\}$ as $\zeta^{\mathbf{\Theta}}_{\boldsymbol{\alpha}}(\theta_1, \theta_2, ..., \theta_V) := \prod_{i=1}^V \xi^{(i)}_{\alpha_i}(\theta_i)$. Therefore, the model in Equation (5) can be written as:

$$Y = \mathcal{M}(\mathbf{\Theta}) = \sum_{|\boldsymbol{\alpha}|\le P} c_{\boldsymbol{\alpha}} \zeta^{\mathbf{\Theta}}_{\boldsymbol{\alpha}}(\theta_1, \theta_2, ..., \theta_V)\,, \tag{6}$$

where $c_{\boldsymbol{\alpha}} \in \mathbb{R}$ are deterministic coefficients that can be estimated thanks to different methods. It can be formulated as a minimization problem, and regularization methods can be used when dealing with small data sets. In the present study, we used the Least Angle Regression Stagewise method (LARS) in order to construct an adaptive sparse PCE. It is an iterative procedure, consisting on an improved version of forward selection. The algorithm begins by finding the polynomial pattern, here denoted ζ_i for simplicity, that is the most correlated to the output. The latter is linearly approximated by $\epsilon_i\zeta_i$, where $\epsilon_i \in \mathbb{R}$. Coefficient ϵ_i is not set to its maximal value, but increased starting from 0, until another pattern ζ_j is found to be as correlated to $Y - \epsilon_i\zeta_i$, and so on. In this approach, a collection of possible PCE, ordered by sparsity, is provided and an optimum can be chosen with an accuracy estimate. It was performed in this study using corrected leave-one-out error. The reader can refer to the work of Blatman and Sudret [25] for further details on LARS and more generally on sparse constructions.

The choice of the basis is crucial and is directly related to the choice of input variable marginals, via the inner product $(.,.)_{\mathcal{L}^2_{\mathbb{R}}}$. Chaos polynomials were first introduced in [38] for input variables characterized by Gaussian distributions. The orthonormal basis with respect to this marginal is the Hermite polynomials family. Later, other Askey scheme hypergeometric polynomial families were associated to some well-known parametric distributions [39]. For example, the Legendre family is orthonormal with respect to the Uniform marginals. This is called *gPC* (generalized Polynomial Chaos) when variables of different PDFs are used as inputs. In practice however, the input distributions of physical variables can be different from usual parametric marginals. In such cases, the marginals can be inferred by empirical methods such as the Kernel Smoother (see [40] for theoretical elements). In this case, an orthonormal polynomial basis with respect to arbitrary marginals can be built with a Gram-Schmidt orthonormalization process as in [41] or via the Stieltjes three-term recurrence procedure as in [42].

To highlight the importance of the marginals and choice of polynomial basis for the learning process, several configurations are attempted in Section 4. Different input sets and distributions (Gaussian, Uniform, inferred by Kernel Smoothing) were tested. The influence of the polynomial basis on the ML is investigated in Section 4.2.2.

2.2.2. Physical importance measures

Once the PCE construction is achieved, a physical interpretation can be performed. It is notable that classical NN indicators can be used [35]. The PCE can be represented in the Feedforward NN paradigm as in Fig. 2. Such networks are classically composed, in addition to the input and output layers, of successive *hidden layers*. Each hidden layer is composed of *neurons* that transform the variables of the previous layer (outputs of the previous *neurons*) into a new set of variables. This is done by combining a linear transformation, giving different *weights* to the previous *neurons*, and a transformation function, called *Activation Function* (AF). This succession of layers is called the *latent representation*. For a number of hidden layers $L \ge 1$, the NN can be formally written as $\mathbf{Y} \approx f_{out}(\mathbf{A}_L\ f_L(...\mathbf{A}_2\ f_2(\mathbf{A}_1\ f_1(\mathbf{A}_{in}\ \mathbf{\Theta}))))$ where $\{\mathbf{A}_k\}_{k\in[|1,L|]}$ and $\{f_k\}_{k\in[|1,L|]}$ are the hidden layer weight matrices and AFs, $\mathbf{A}_{in}$ is the input-to-hidden connection matrix and f_{out} is the final hidden-output transformation [2].

The PCE-based NN represented in Fig. 2 is a single layer feedforward, composed of $l \in \mathbb{N}$ neurons, that can be written as $\mathbf{Y} \approx f_{out}(\mathbf{A}_1\ f_1(\mathbf{A}_{in}\mathbf{\Theta}))$. The first matrix $\mathbf{A}_{in}$ is the input-to-hidden connection matrix of dimension $V \times V$, that links the input layer to the PCE hidden layer containing the multivariate polynomials $\left\{\zeta^{\mathbf{\Theta}}_{\boldsymbol{\alpha}}, \boldsymbol{\alpha} \in \{\boldsymbol{\alpha}_1, ..., \boldsymbol{\alpha}_l\}\right\}$, where V is the number of

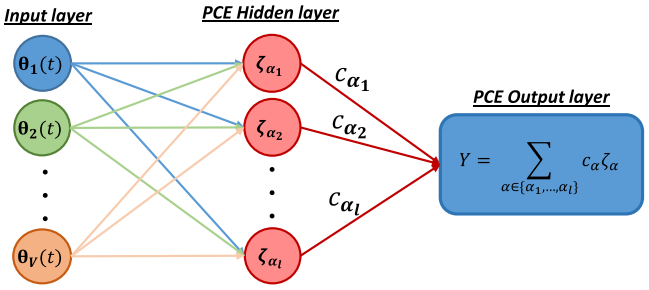


Fig. 2. Representation of the PCE learning in the NN paradigm.

inputs and the multivariate indexes $\{\boldsymbol{\alpha}_1, ..., \boldsymbol{\alpha}_l\}$ are conditioned by the chosen polynomial degree p such as $\forall i \in [|1, l|]\ 0 \leq |\boldsymbol{\alpha}_i| \leq p$, and by the number of selected features if a sparse polynomial is constructed, as in the present case using LARS [28]. Matrix $\mathbf{A}_{in}$ represents the contributions of the V variables to the multivariate polynomials $\left\{\zeta^{\boldsymbol{\Theta}}_{\boldsymbol{\alpha}}, \boldsymbol{\alpha} \in \{\boldsymbol{\alpha}_1, ..., \boldsymbol{\alpha}_l\}\right\}$. It is a diagonal matrix such that $[\boldsymbol{A}_{in}]_{j,j\in[|1,V|]^2}$ is 0 if $\forall i \in [|1, l|]\ (\alpha_i)_j = 0$ and 1 if not. The first multi-dimensional AF f_1 is a vector of multivariate functions that transforms the set of selected inputs corresponding to $[\boldsymbol{A}_{in}]_{i,i\in[|1,V|]^2} = 1$ to the multivariate polynomials of the chosen basis (Hermite, Legendre, etc.) by tensor product over the univariate basis. The hidden layer weight matrix $\mathbf{A}_1$ gives different weights to the constructed polynomial features. It is a diagonal matrix composed of the PCE expansion coefficients such as $[\mathbf{A}_1]_{i,j\in[|1,l|]^2} = [c_{\boldsymbol{\alpha}_i}]_{i\in[|1,l|]}$.

The final hidden-output transformation f_{out} is a summation. Fig. 2 can also be presented differently: another hidden layer can be added to the PCE latent representation as $\mathbf{Y} \approx f_{out}(\mathbf{A}_2\ f_2(\mathbf{A}_1\ f_1(\mathbf{A}_{in}\ \boldsymbol{\Theta}))$. The first AF f_1 would represent a transformation of each input variable to a list of monomials of degrees 1 to p (here, $\mathbf{A}_{in}$ is identity). The second AF f_2 therefore represents the tensor product that transforms the different monomials to multivariate features, with $\mathbf{A}_1$ appropriately filled with zeros and ones, and $[\mathbf{A}_2]_{i,j\in[|1,l|]^2} = [c_{\boldsymbol{\alpha}_i}]_{i\in[|1,l|]}$.

To capture the importance of each feature, the Garson relative Weights (GW) defined in Equation (7) are a classical measure to quantify the relative importance of each neuron of the last hidden layer, and therefore of each polynomial pattern, for the output value [35,43].

$$w_{\zeta^{\boldsymbol{\Theta}}_{\boldsymbol{\alpha}}} = \frac{|c_{\boldsymbol{\alpha}}|}{\sum_{0\leq\boldsymbol{\beta}\leq 1} |c_{\boldsymbol{\beta}}|}\ . \tag{7}$$

This measure can be used to understand the importance given by the NN algorithm to the variables and their possible interactions, especially when using feature selection algorithms as LARS: "feature interactions [...] are created at hidden units with nonlinear activation functions, and the influences of the interactions are propagated layer-by-layer to the final output" [43]. In the particular case of a polynomial expansion, the interpretation is straightforward, the importance of each variable alone corresponds to its monomials, and the importance of its interactions with other variables corresponds to the multivariate polynomials in which it is involved.

For the particular case of the orthonormal basis provided by PCE, the GW defined in (7) can be interpreted in terms of Pearson's correlations between output Y and the polynomial basis elements $\zeta^{\boldsymbol{\Theta}}_{\boldsymbol{\alpha}}$ denoted $\rho_{Y,\ \zeta^{\boldsymbol{\Theta}}_{\boldsymbol{\alpha}}}$, with $\alpha \neq (0, ..., 0)$. Indeed, Pearson's correlations $\rho_{Y,\ \zeta^{\boldsymbol{\Theta}}_{\boldsymbol{\alpha}}}$ are defined as in Equation (8),

$$\rho_{Y,\ \zeta^{\boldsymbol{\Theta}}_{\boldsymbol{\alpha}}} = \frac{\mathbb{E}\left[(Y - \mathbb{E}(Y))(\zeta^{\boldsymbol{\Theta}}_{\boldsymbol{\alpha}} - \mathbb{E}(\zeta^{\boldsymbol{\Theta}}_{\boldsymbol{\alpha}}))\right]}{\sqrt{\mathbb{V}(Y)\mathbb{V}(\zeta^{\boldsymbol{\Theta}}_{\boldsymbol{\alpha}})}} = \frac{c_{\boldsymbol{\alpha}}}{\sqrt{\sum_{1\leq|\boldsymbol{\beta}|\leq p} c^2_{\boldsymbol{\beta}}}}\ , \tag{8}$$

thanks to the orthonormality of the basis with respect to the scalar product $(.\ ,.)_{\mathcal{L}^2_{\mathbb{R}}}$ that guarantees:

- $\mathbb{E}\left[\zeta^{\boldsymbol{\Theta}}_{\boldsymbol{\alpha}}\right] = \left(\zeta^{\boldsymbol{\Theta}}_{\boldsymbol{\alpha}}, \zeta^{\boldsymbol{\Theta}}_{\boldsymbol{\beta}=(\mathbf{0},...,\mathbf{0})} = 1\right)_{\mathcal{L}^2_{\mathbb{R}}} = 0$;
- $\mathbb{E}\left[Y\right] = \left(\sum_{0\leq|\boldsymbol{\beta}|\leq p} c_{\boldsymbol{\beta}}\zeta^{\boldsymbol{\Theta}}_{\boldsymbol{\beta}}\ , \zeta^{\boldsymbol{\Theta}}_{\boldsymbol{\beta}=(\mathbf{0},...,\mathbf{0})}\right)_{\mathcal{L}^2_{\mathbb{R}}} = c_{\beta=(0,...,0)}$;
- $\mathbb{E}\left[Y, \zeta^{\boldsymbol{\Theta}}_{\boldsymbol{\alpha}}\right] = \left(\sum_{0\leq|\boldsymbol{\beta}|\leq p} c_{\boldsymbol{\beta}}\zeta^{\boldsymbol{\Theta}}_{\boldsymbol{\beta}}\ , \zeta^{\boldsymbol{\Theta}}_{\boldsymbol{\alpha}}\right)_{\mathcal{L}^2_{\mathbb{R}}} = c_{\boldsymbol{\alpha}}$;
- $\mathbb{V}\left[\zeta^{\boldsymbol{\Theta}}_{\boldsymbol{\alpha}}\right] = \mathbb{E}\left[\left(\zeta^{\boldsymbol{\Theta}}_{\boldsymbol{\alpha}} - \mathbb{E}\left[\zeta^{\boldsymbol{\Theta}}_{\boldsymbol{\alpha}}\right]\right)^2\right] = \mathbb{E}\left[\left(\zeta^{\boldsymbol{\Theta}}_{\boldsymbol{\alpha}}\right)^2\right] = ||\zeta^{\boldsymbol{\Theta}}_{\boldsymbol{\alpha}}||^2_{\mathcal{L}^2_{\mathbb{R}}} = 1$;
- $\mathbb{V}\left[Y\right] = \mathbb{E}\left[\left(Y - \mathbb{E}\left[Y\right]\right)^2\right] = \left(\sum_{1\leq|\boldsymbol{\beta}|\leq p} c_{\boldsymbol{\beta}}\zeta^{\boldsymbol{\Theta}}_{\boldsymbol{\beta}}\ , \sum_{1\leq|\boldsymbol{\beta}|\leq p} c_{\boldsymbol{\beta}}\zeta^{\boldsymbol{\Theta}}_{\boldsymbol{\beta}}\right)_{\mathcal{L}^2_{\mathbb{R}}} = \sum_{1\leq|\boldsymbol{\beta}|\leq p} c^2_{\boldsymbol{\beta}}$.

Therefore, the weights $w_{\zeta^{\boldsymbol{\Theta}}_{\boldsymbol{\alpha}}}$ can also be computed as $|\rho_{Y,\ \zeta^{\boldsymbol{\Theta}}_{\boldsymbol{\alpha}}}| / \sum_{1\leq|\boldsymbol{\beta}|\leq p} |\rho_{Y,\ \zeta^{\boldsymbol{\Theta}}_{\boldsymbol{\beta}}}|$. This means that they measure the relative importance of the basis element in the expansion of the output, in terms of linear correlation, regardless of the sign

of the latter. These "relative Pearson's correlations" can be seen as a physical contribution since the PCE model is strictly linear. The sum of the GW $w_{\zeta_{\boldsymbol{\alpha}}\boldsymbol{\Theta}}$ for all the polynomial features equals 1. This means that they allow $\{\zeta_{\boldsymbol{\alpha}}\}_{|\boldsymbol{\alpha}|\leq p}$ to be ranked in terms of relative contribution to the output Y. The contributions can be analyzed either for each polynomial pattern separately, or for a single variable θ_i by adding all the polynomial shares related to this variable alone, or by adding all the polynomial shares related to this variable and its interactions (Sobol' indices analogy [23]).

2.3. POD-PCE based predictor

POD and PCE were introduced separately in Subsections 2.1 and 2.2 respectively. We are now fully equipped with the adequate theoretical basis and mathematical notations, to present the POD-PCE ML methodology for a data-based model learning of a multidimensional physical field. In this Subsection, we will first summarize the proposed approach, then the formal details of the coupling will be given with the definition of adequate accuracy measures. Finally the previously discussed importance measures will be generalized for the POD-PCE physical study.

The proposed POD-PCE ML consists of steps, in a learning and a prediction phase, summed up as follows:

- Learning phase:
  - ∗ POD basis construction: given a set of measurements $\mathbf{U}(\mathcal{X},\mathcal{T}) = [\mathbf{u}(\mathbf{x}_i,t_j)]_{i,j} \in \mathbb{R}^{m\times n}$ (snapshot matrix), construct a spatial POD basis accordingly. Variable t_j can represent time in case of temporal dynamics, or more generally an occurrence of $\mathbf{U}(\mathcal{X},.)$. Then, in general, $\mathcal{T}$ would be an event space;
  - ∗ PCE learning: construct PCE models that map each POD coefficient, obtained in the previous step along with the spatial basis, to a set of inputs. In the particular case of temporal dynamics, previous values of the physical field, represented by previous POD coefficients, can be part of the learning inputs. For example, one could use an initial field value $\mathbf{U}(\boldsymbol{\mathcal{X}},t_j)$ to learn a future field $\mathbf{U}(\boldsymbol{\mathcal{X}},t_{j+1})$ from a set of physical parameters that condition the evolution over $[t_j,t_{j+1}]$. The latter can consist in time series of physical variables, representative statistics of the latter, physical constants, etc. and can be denoted $\boldsymbol{\Theta}(t_j\rightarrow t_{j+1})$;
- Prediction phase:
  - ∗ Given a new realization of the inputs, predict the new POD coefficients using the learned PCE models, then reconstruct the new estimate $\mathbf{U}(\boldsymbol{\mathcal{X}},t_k)$ on the POD basis. As previously explained, an initial value to the physical field, denoted $\mathbf{U}(\boldsymbol{\mathcal{X}},t_{k-1})$, may be part of the inputs for temporal dynamics. In particular, its reduced form, consisting in temporal POD coefficients, is used. In this case, an additional step is needed: $\mathbf{U}(\boldsymbol{\mathcal{X}},t_{k-1})$ is projected on the constructed POD basis in order to retrieve the values of associated POD coefficients which are then used as PCE inputs.

The learning and prediction set-ups are more complex to establish for temporal evolution problems, because the field information at previous times are required. Therefore, for the sake of clarity, the steps are explicitly developed in the following Subsection 2.3.1. The accuracy of the methodology is later demonstrated on both a parametric toy problem in Section 3, and a field measurements-based temporal problem in Section 4. These two can be considered as complementary applications, and demonstrate that the POD-PCE ML can be applied in different learning set-ups of multi-dimensional physical fields. Similarities in the treatment of both problems can be noticed, but their particularities are also used to demonstrate different properties of the POD-PCE learning, that are shortly described at the beginning of each section.

2.3.1. Machine learning methodology

Here, the formal hypothesis behind the POD-PCE ML reasoning and its mathematical formulation are discussed. Let $\mathbf{U}(\boldsymbol{\mathcal{X}},.) = [\mathbf{u}(\mathbf{x}_i,.)]_{i\in\{1,\dots,m\}}$ be a field of interest defined on a set of $m\in\mathbb{N}$ space coordinates $\boldsymbol{\mathcal{X}} = \{\mathbf{x}_1,\dots,\mathbf{x}_m\}$. Let $\boldsymbol{\Theta}(.) = (\theta_1(.),\theta_2(.),\dots,\theta_V(.))$ be a vector of the inputs supposed to condition the evolution of $\mathbf{U}(\boldsymbol{\mathcal{X}},.)$ over time. The dynamic model, denoted $\mathcal{H}$, that gives an estimation of a future state $\mathbf{U}(\boldsymbol{\mathcal{X}},t_{j+1})$ from a past state $\mathbf{U}(\boldsymbol{\mathcal{X}},t_j)$ and an estimation of $\boldsymbol{\Theta}(t_j\rightarrow t_{j+1})$ over the time interval $[t_j,t_{j+1}]$, where $t_j<t_{j+1}\in\mathbb{R}^+$, is formulated as in Equation (9).

$$\mathbf{U}(\boldsymbol{\mathcal{X}},t_{j+1}) \approx \mathcal{H}\left[\mathbf{U}(\boldsymbol{\mathcal{X}},t_j), t_{j+1}-t_j, \boldsymbol{\Theta}(t_j\rightarrow t_{j+1})\right] . \tag{9}$$

If the field of interest has been recorded over a set of past times $\mathcal{T} = \{t_1,\dots,t_n\}\subset\mathbb{R}^+$, where $t_j<t_{j+1}$, a POD basis can be constructed as in Section 2.1, consisting of $d\in\mathbb{N}$ vectors of dimension m stored in a matrix as $\boldsymbol{\Phi}^{(d)}(\boldsymbol{\mathcal{X}}) = (\boldsymbol{\Phi}_1^{(d)}(\boldsymbol{\mathcal{X}}),\dots,\boldsymbol{\Phi}_d^{(d)}(\boldsymbol{\mathcal{X}}))\in\mathbb{R}^{m\times n}$, and can be seen as a generator of all possible states if enough records are available. If so, any future state $\mathbf{U}(\boldsymbol{\mathcal{X}},t_j)$ can be expanded on this POD basis and the associated temporal coefficients are simply the weights of $\mathbf{U}(\boldsymbol{\mathcal{X}},t_j)$ on the POD basis. They are therefore obtained using the canonical scalar product over $\mathbb{R}^m$, as in Equation (10).

$$\begin{aligned}\mathbf{U}(\boldsymbol{\mathcal{X}},t_j) &\approx \sum_{k=1}^{d} a_k(t_j)\sqrt{n\times\lambda_k}\boldsymbol{\Phi}_k^{(d)}(\boldsymbol{\mathcal{X}})\\ &\approx \sum_{k=1}^{d}(\mathbf{U}(\boldsymbol{\mathcal{X}},t_j),\boldsymbol{\Phi}_k^{(d)}(\boldsymbol{\mathcal{X}}))_{\mathbb{R}^m}\boldsymbol{\Phi}_k^{(d)}(\boldsymbol{\mathcal{X}})\\ &\approx \sum_{k=1}^{d}\mathbf{U}(\boldsymbol{\mathcal{X}},t_j)^T\boldsymbol{\Phi}_k^{(d)}(\boldsymbol{\mathcal{X}})\boldsymbol{\Phi}_k^{(d)}(\boldsymbol{\mathcal{X}}) .\end{aligned} \tag{10}$$

Hence, the variable part of $\mathbf{U}(\boldsymbol{\mathcal{X}},t_j)$ is fully expressed in the temporal coefficients $a_k(t_j)$. The field of interest $\mathbf{U}(\boldsymbol{\mathcal{X}},t_j)$ can be either a field measurement, a laboratory or a numerical experiment. In any-case, it can be considered as being generated

by a random process "in the sense that nature happens without consideration of what could be the best realizations for the learning algorithm" [2]. Therefore, the coefficients $a_k(t_j)$ can also be seen as the jth realization of a random variable A_k. We can therefore construct a PCE approximation $\mathcal{H}_k$ that maps A_k from its input space. The latter is taken as a collection of random variables, composed from the set $(A_1, ..., A_d)$ at a previous time, the duration of the dynamic, and the input variables $\mathbf{\Theta}(t_j \to t_{j+1})$. This is formulated as a classical dynamic model in Equation (11).

$$a_k(t_{j+1}) \approx \mathcal{H}_k\left[a_1(t_j), \dots, a_d(t_j), t_{j+1} - t_j, \mathbf{\Theta}(t_j \to t_{j+1})\right] . \tag{11}$$

The model $\mathcal{H}$ in Equation (9) is approximated as in Equation (12).

$$\mathcal{H}\left[\mathbf{U}(\boldsymbol{\mathcal{X}}, t_j), t_{j+1} - t_j, \mathbf{\Theta}(t_j \to t_{j+1})\right] \approx \sum_{k=1}^{d} \mathcal{H}_k\left[a_1(t_j), \dots, a_d(t_j), t_{j+1} - t_j, \mathbf{\Theta}(t_j \to t_{j+1})\right] \sqrt{n \times \lambda_k} \mathbf{\Phi}_k^{(d)}(\boldsymbol{\mathcal{X}}) . \tag{12}$$

Some limitations to the introduced formulations in Equations (11) and (12) can be highlighted. A first limitation concerns discontinuities that can be met in physical fields. This can occur either in the complete spatial field $\mathbf{U}(.,.)$, in its reduced version represented by the POD coefficients $a_k(.)$, or in the inputs $\mathbf{\Theta}$. In the first case, the classical linear approximations as POD may be inefficient [44]. One solution developed by Taddei [44], called RePOD (Registration POD), consists in a parametric transformation of the interest discontinuous field into a smoother one for linear transformations. In the second case, where discontinuity happens in the POD temporal coefficients, this would impact the learning with PCE. Innovative solutions were identified to apply PCE when the output's space is characterized with rapid variations or discontinuities, for instance near a critical point in the inputs space. As an example, a method called adaptive Multi-Element PCE was developed for Legendre polynomials in [45] and extended to arbitrary measures in [42]. The inputs space is decomposed to a union of subsets, and the output variable is locally expanded on each subset. The final solution is then a combination of PCE sub-problems. In the last discontinuity case that concerns the inputs $\mathbf{\Theta}$, the previous splitting techniques can also be used. For example, the sub-intervals in the inputs space can be constructed in such way to avoid the discontinuity. PCE sub-problems would therefore be treated as usual.

A second limitation concerns the choice of input variables for regression models, and is an ongoing research question in statistics [46]. As a practical illustration, the dynamical problem written in Equation (11) can incorporate additional inputs, for example the information at previous times t_{j-1}, t_{j-2}, etc. However, when a large set of inputs can be used and only a small set of realizations is available for learning, a well-posedness problem occurs. One solution consists in transforming the large set of inputs to a reduced version, for example with the help of PCA [47] for DR. This approach was not studied here and will be the topic of a future study. However, different input configurations will be evaluated, to investigate the influence of variable selection on the proposed learning. For example, the hypothesis of dependence between the random variables $(A_1, \dots, A_d)$ could be relaxed. This would imply writing the approximation in Equation (11) in a relaxed form as $\mathcal{H}_k\left[a_k(t_j), t_{j+1} - t_j, \mathbf{\Theta}(t_j \to t_{j+1})\right]$. In that case a simpler model $\mathcal{H}$, under the strong independence assumption, can be formulated as in Equation (13).

$$\mathcal{H}\left[\mathbf{U}(\boldsymbol{\mathcal{X}}, t_j), t_{j+1} - t_j, \mathbf{\Theta}(t_j \to t_{j+1})\right] \approx \sum_{k=1}^{d} \mathcal{H}_k\left[a_k(t_j), t_{j+1} - t_j, \mathbf{\Theta}(t_j \to t_{j+1})\right] \sqrt{n \times \lambda_k} \mathbf{\Phi}_k^{(d)}(\boldsymbol{\mathcal{X}}) . \tag{13}$$

Both alternatives are tested in Section 4. To investigate the influence of input selection on learning accuracy, a quantitative evaluation of the hypothesis is needed. More generally, whether for the above-mentioned simplifications or for the approximated form of the model in general, accuracy estimators are needed. These are presented below.

2.3.2. Accuracy tests for the approximation

There are two determining parts in the POD-PCE learning process. Firstly, the PCE learning $\mathcal{H}_k(.)$ of each mode A_k should be as accurate as possible. Secondly, the reconstructed field $\sum_{k=1}^{d} \mathcal{H}_k(.)\sqrt{n \times \lambda_k}\mathbf{\Phi}_k^{(d)}(\boldsymbol{\mathcal{X}})$ for a given rank d should be as close to the real field $\mathbf{U}(\boldsymbol{\mathcal{X}})$ as possible.

The distance between each mode and its PCE approximate can be evaluated using the *generalization error*, denoted $\delta(A_k, \mathcal{H}_k)$ and defined as in Equation (14).

$$\delta(A_k, \mathcal{H}_k) = \mathbb{E}\left[\left(A_k - \mathcal{H}_k(.)\right)^2\right] . \tag{14}$$

For the model defined in Equation (13), this error can be estimated, on a set of paired realizations $(a_k(t_1), \dots, a_j(t_n))$ and $(\mathbf{\Theta}(t_0 \to t_1), \dots, \mathbf{\Theta}(t_{n-1} \to t_n))$, as in Equation (15) as explained by Blatman [28]. This approximated version of the *generalization error* is called the *empirical error*.

$$\delta(A_k, \mathcal{H}_k) \approx \delta_{emp}(A_k, \mathcal{H}_k) := \frac{1}{n}\sum_{j=1}^{n}\left(a_k(t_j) - \mathcal{H}_k\left[a_k(t_{j-1}), t_{j+1} - t_j, \mathbf{\Theta}(t_{j-1} \to t_j)\right]\right)^2 . \tag{15}$$

Its relative estimate denoted $\epsilon_{emp}(A_k, \mathcal{H}_k)$ can be defined as in Equation (16).

$$\epsilon_{emp}(A_k, \mathcal{H}_k) := \frac{\delta_{emp}(A_k, \mathcal{H}_k)}{\mathbb{V}[A_k]} \,. \tag{16}$$

Once the PCE learnings can be trusted, the distance at time t_j between the true state $\mathbf{U}(\boldsymbol{\mathcal{X}}, t_j)$ and the POD-PCE approximation $\mathcal{H}\left[\mathbf{U}(\boldsymbol{\mathcal{X}}, t_j), t_{j+1} - t_j, \boldsymbol{\Theta}(t_j \to t_{j+1})\right]$ can be defined. It might be estimated using the relative Root Mean Squared Error (relative RMSE), denoted $r[\mathbf{U}, \mathcal{H}](t_j)$ and calculated as in Equation (17), where $\mathbf{h}(\mathbf{x}_i, t_j)$ refers to the value of the POD-PCE approximation at coordinate $\mathbf{x}_i$ and time t_j.

$$r[\mathbf{U}, \mathcal{H}](t_j) := \sqrt{\frac{\sum_{i=1}^{m} \left[\mathbf{u}(\mathbf{x}_i, t_j) - \mathbf{h}(\mathbf{x}_i, t_j)\right]^2}{\sum_{i=1}^{m} \left[\mathbf{u}(\mathbf{x}_i, t_j)\right]^2}} \,. \tag{17}$$

A mean value of the relative RMSE is calculated over a set of realizations corresponding to a set of times $\mathcal{T} = \{t_1, \ldots, t_n\}$. It is denoted $r[\mathbf{U}, \mathcal{H}]^{(\mathcal{T})}$ and estimated as in Equation (18).

$$r[\mathbf{U}, \mathcal{H}]^{(\mathcal{T})} := \frac{1}{n} \sum_{j=1}^{n} r[\mathbf{U}, \mathcal{H}](t_j) \,. \tag{18}$$

Once the accuracies of the PCE learnings and the POD-PCE coupling have been evaluated, a final model, which will be the most accurate one, can be chosen. This model would, for our ML set-up, be the best representation of the dependence structure between inputs and outputs. It is used to shed light on the underlying physical relationships. Therefore the inputs are ranked in terms of physical influence, using an appropriate ranking indicator, presented in the following Subsection.

2.3.3. Physical influence of inputs based on the POD-PCE model

The GW influence measures presented for the PCE models in Subsection 2.2 are here extended for the POD-PCE coupling. These indicators are adequate for the analysis of each PCE model $\mathcal{H}_k$, i.e. for interpreting the contribution of the inputs to each random variable A_k separately. However, calculating the contributions to each A_k independently precludes putting them in perspective according to the importance of A_k in the final reconstructed model $\mathcal{H}$ that approximates $\mathbf{U}(\boldsymbol{\mathcal{X}}, .)$. Hence, adapted indicators should be calculated.

Let $\mathbf{U}(\boldsymbol{\mathcal{X}}, .)$ be the random spatiotemporal field approximated by the POD-PCE ML, for prediction from time t_j to time t_{j+1} and let $\mathcal{H}_k$ be the PCE approximation at degree $p^{(k)}$ that maps the random POD temporal coefficient A_k from a set of input variables, using the expansion on the multivariate polynomial basis $\left\{\zeta_{\boldsymbol{\alpha}}^{(k)}(.)\right\}_{|\boldsymbol{\alpha}| \leq p^{(k)}}$. The POD-PCE model formulated in Equation (12) is written as in Equation (19):

$$\mathbf{U}(\boldsymbol{\mathcal{X}}, .) \approx \sum_{k=1}^{d} A_k \sqrt{n \times \lambda_k} \boldsymbol{\Phi}_k^{(d)}(\boldsymbol{\mathcal{X}}) \approx \sum_{k=1}^{d} \left(\sum_{|\boldsymbol{\alpha}| \leq p^{(k)}} c_{\boldsymbol{\alpha}}^{(k)} \zeta_{\boldsymbol{\alpha}}^{(k)}(.) \right) \sqrt{n \times \lambda_k} \boldsymbol{\Phi}_k^{(d)}(\boldsymbol{\mathcal{X}}) \,. \tag{19}$$

Thanks to its linearity, the POD-PCE ML can be represented as a single-layered NN, as shown in Fig. 3.

Therefore, a new indicator, *Generalized Garson Weights* (GGW), denoted $W_{\zeta_{\boldsymbol{\alpha}}^{(k)}}$, is computed and simply re-evaluated from the PCE Garson weights (GW), here denoted $w_{\zeta_{\boldsymbol{\alpha}}^{(k)}}$, as in Equation (20).

$$\begin{aligned} W_{\zeta_{\boldsymbol{\alpha}}^{(k)}} &:= \frac{|c_{\boldsymbol{\alpha}}^{(k)}| \sqrt{n \times \lambda_k}}{\sum_{e=1}^{d} \sum_{|\boldsymbol{\beta}| \leq p^{(e)}} \left(|c_{\boldsymbol{\beta}}^{(e)}| \sqrt{n \times \lambda_e} \right)} \\ &= \frac{\left(\sum_{|\boldsymbol{\beta}| \leq p^{(k)}} |c_{\boldsymbol{\beta}}^{(k)}| \right) w_{\zeta_{\boldsymbol{\alpha}}^{(k)}} \sqrt{\lambda_k}}{\sum_{e=1}^{d} \sum_{|\boldsymbol{\beta}| \leq p^{(e)}} \left(|c_{\boldsymbol{\beta}}^{(e)}| \sqrt{\lambda_e} \right)} = \left(\frac{\sum_{|\boldsymbol{\beta}| \leq p^{(k)}} \left(|c_{\boldsymbol{\beta}}^{(k)}| \sqrt{\lambda_k} \right)}{\sum_{e=1}^{d} \sum_{|\boldsymbol{\beta}| \leq p^{(e)}} \left(|c_{\boldsymbol{\beta}}^{(e)}| \sqrt{\lambda_e} \right)} \right) w_{\zeta_{\boldsymbol{\alpha}}^{(k)}} \,. \end{aligned} \tag{20}$$

These GGW indicators show that the contribution of the polynomials $\{\zeta_{\boldsymbol{\alpha}}^{(k)}\}_{|\boldsymbol{\alpha}| \leq p^{(k)}}$ of A_k are enhanced with the eigenvalue λ_k, which is directly linked to the importance of the POD mode $\boldsymbol{\Phi}_k^{(d)}(\boldsymbol{\mathcal{X}})$ (EVR in Equation (2)). An analogy can be drawn with the generalized sensitivity indices for a reduced order model [48]. The $\sum_{k=1}^{d} \sum_{|\boldsymbol{\alpha}| \leq p^{(k)}} W_{\zeta_{\boldsymbol{\alpha}}^{(k)}} = 1$ property holds. This means that the indices allow $\{\{\zeta_{\boldsymbol{\alpha}}^{(k)}\}_{|\boldsymbol{\alpha}| \leq p^{(k)}}\}_{k \in \{1,\ldots,d\}}$ to be ranked altogether in terms of contribution to output $\mathbf{U}$. The influences can be analyzed following the indications of Section 2.2.2.

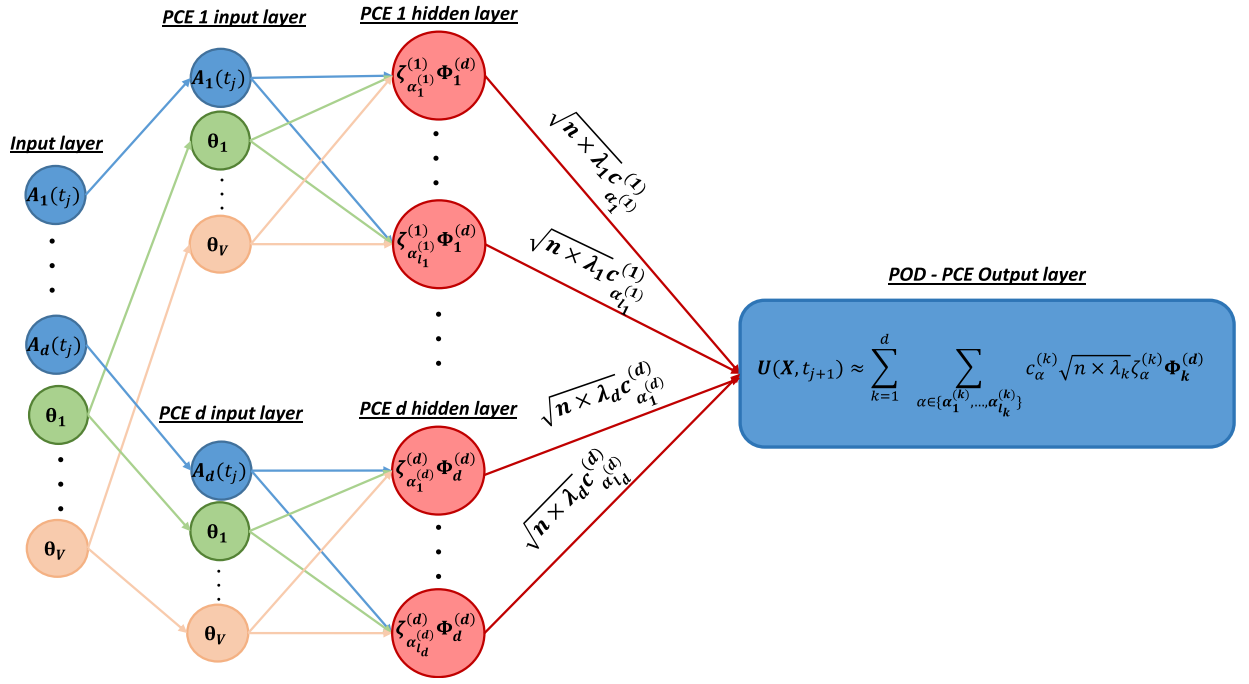


Fig. 3. Representation of the POD-PCE ML approach in the NN paradigm.

3. Application on a parametric toy problem

The theoretical framework of the proposed POD-PCE learning was presented in the previous Section 2, including the detailed coupling formulation, accuracy estimators and physical influence measures in Subsection 2.3. In the latter, it was highlighted that there is a slight difference in the learning and prediction steps between temporal problems and parametric problems. In this section, the POD-PCE ML is applied to a parametric toy problem, for which the analytical solution is introduced in Subsection 3.1. The problem is simple and controllable, and allows demonstrating the learning performance, the consistency of physical interpretations in comparison with the analytical information, and the robustness of the learning to noise in the data. Subsection 3.2 therefore deals with the application of the POD-PCE methodology for physical analysis and prediction, while in Subsection 3.3, robustness to different noise levels is investigated.

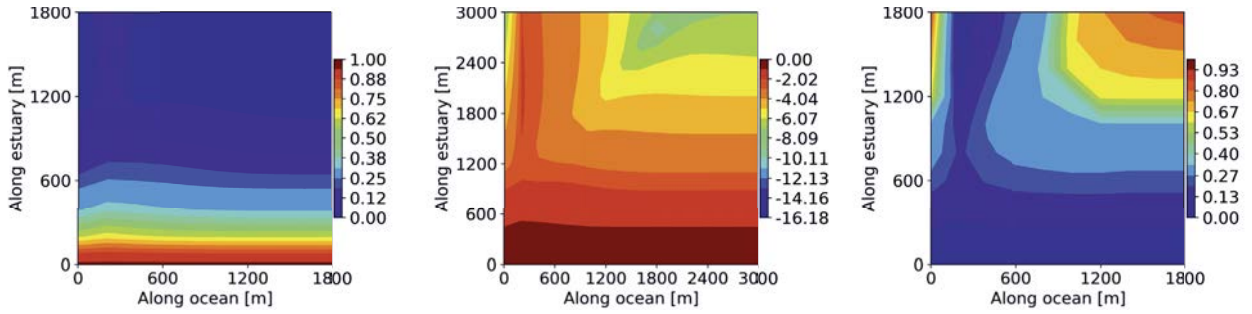
3.1. Problem description

The chosen toy problem deals with the representation of groundwater flow in a confined aquifer. Such a flow can be complex to describe and is generally represented using the depth-averaged groundwater flow equations [34]. Analytical solutions for these equations can be found for particular configurations. For example, a solution was identified by Li et al. [34] in case of a semi-infinite coastal aquifer subject to oscillating boundary conditions, resulting from oceanic and estuarine tidal loadings. The solution is given for the particular case where the estuary and coastline are perpendicular. The oceanic BC (along coastline) is taken as a single and spatially uniform tidal harmonic constituent $Acos(\omega t)$, where A and ω are the tidal amplitude and pulsation respectively. The corresponding BC along the estuary is a non-uniform tidal loading $Aexp(-\kappa_{er}x)cos(\omega t - \kappa_{ei}x)$, where κ_{er} and κ_{ei} are the estuary's tidal damping coefficient and tidal wave number respectively, that represent changes in the amplitude and phase along the estuary.

This forcing results with fluctuations in the *water table*, that is defined as the level separating the water and saturated ground from the remaining upper unsaturated ground. The fluctuations, denoted f, can be calculated using the analytical solution defined in [34] as in Equation (21).

$$\begin{cases} f(x,y,t) &= f_0(x,t) + f_1(x,y,t) \\ f_0(x,t) &= Aexp\left(-\sqrt{\frac{\omega}{2D}}x\right) cos\left(\omega t - \sqrt{\frac{\omega}{2D}}x\right) \\ f_1(x,y,t) &= A \times Re\left\{\int_0^t [g(k_1,x) - g(k_1,-x) - g(k_2,x) + g(k_2,-x)]\, dt_0\right\} \end{cases} \tag{21}$$

where constant D is the diffusivity of the aquifer [34], t is the time variable, and (x, y) are the cartesian cross- and long-shore coordinates, corresponding to the distance from ocean and estuary respectively. The operator $Re\{z\}$ denotes the real



(a) Amplitude (m) (b) log-Amplitude (c) Phase

Fig. 4. Amplitude and phase of the water table fluctuation, using the parameters proposed by [34] in Table 1. (For interpretation of the colors in the figure(s), the reader is referred to the web version of this article.)

part of complex z. Coefficients k_1 and k_2 are defined as $k_1 := -(\kappa_{er} + \kappa_{ei} i)$ and $k_2 := -(\sqrt{\frac{\omega}{2D}} + \sqrt{\frac{\omega}{2D}} i)$, where $i = \sqrt{-1}$. Function f is defined in equation (22), where $erf(z) := \frac{2}{\sqrt{\pi}} \int_0^z e^{-t^2} dt$ is the Gauss error function.

$$g(\psi, \xi) = \frac{y}{4\sqrt{\pi}[D(t-t_0)]^{3/2}} \times exp\left(\psi^2 D(t-t_0) + i\omega t_0 + \psi\xi - \frac{y^2}{4D(t-t_0)}\right) \times \left[1 + erf\left(\frac{2\psi D(t-t_0) + \xi}{2\sqrt{D(t-t_0)}}\right)\right] \tag{22}$$

This solution is complex and non-linear due to the presence of an interaction zone where the effects of the ocean and estuary are coupled. This results with complex fluctuation patterns that can extend to several square kilometers [34], depending on the aquifer configuration. For example, the diurnal tide configuration proposed by [34] in Table 1 is used for illustration. The amplitude of the fluctuation calculated at each (x, y) location as $[\max_t(f(x, y, t)) - \min_t(f(x, y, t))]/2$ over $t \in [0, T]$, and the phase calculated at each (x, y) location as the time lag, relative to T, between the time series $f(x, y, t)$ and $f(0, 0, t)$ over $t \in [0, T]$, are shown in Fig. 4.

The amplitude is decreasing through the aquifer (Figs. 4-a and 4-b), and a time lag is noticed in the tidal propagation (Fig. 4-c). Both the amplitude damping and time lag are increasing through the aquifer and along the estuary. It can therefore be interesting to see if the POD-PCE methodology succeeds in recovering and explaining such patterns, in particular by learning their dependency to the tidal, estuary and aquifer parameters, from a statistical sample of the solution.

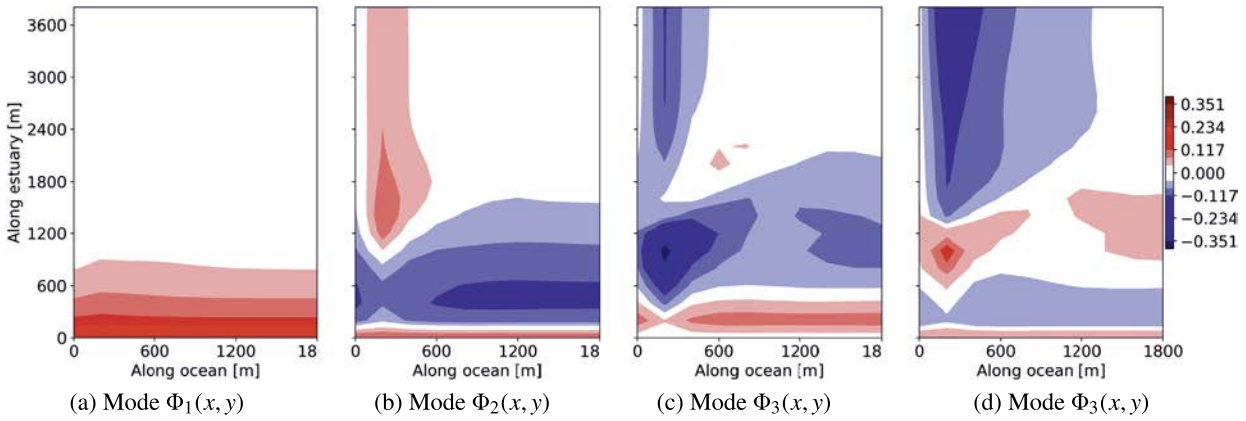
In order to apply the POD-PCE methodology on the aquifer case, a statistical sample of the solution and corresponding input sample of parameters are needed. In the presented study, the tidal period T (and therefore pulsation ω) is fixed to the diurnal configuration of Li et al. [34], whereas an ensemble of realizations is generated for the remaining control parameters (A, k_{er}, k_{ei}, D). For this, Gaussian PDFs are used with mean values corresponding to the setting used by Li et al. [34], and a variation coefficient (standard deviation divided by mean) of 20%. This value corresponds to the average variation coefficient associated to optimal fitting of groundwater flow parameters performed by [49] on several cases. Indeed, the maximum variation coefficient was between 12% and 28% depending on the case. A random sample of size $n = 200$ is produced using the Gaussian PDFs (Monte Carlo), and each realization of the inputs denoted $\boldsymbol{\Theta}_{j\in\{1,\dots,n\}}$ is associated to a realization of the output by calculating $f(x, y, t)_{j\in\{1,\dots,n\}}$.

3.2. POD-PCE learning

The methodology is applied on the perturbation amplitude in the aquifer. The objective is to understand how the perturbation propagates from the boundaries, for different tidal, aquifer and estuary characteristics. The perturbations are calculated over a tidal period on a cartesian spatial grid composed of $m \in \mathbb{N}$ points, denoted $(x, y)_{i\in\{1,\dots,m\}}$. The spatial discretization step is 200 m in both directions, and the temporal step is 1 hour. The amplitude of the perturbation, denoted a', is then locally computed on each point of the grid. It depends on both the spatial location in the aquifer and the simulation parameters $\boldsymbol{\Theta}$. The solutions can be stored in a snapshot matrix as $\boldsymbol{A}'(\boldsymbol{\mathcal{X}}, \mathcal{T}) = \left[a'\left((x, y)_i, \boldsymbol{\Theta}_j\right)\right]_{i,j} \in \mathbb{R}^{m\times n}$, where $\boldsymbol{\mathcal{X}}$ designates the spatial coordinates space and $\mathcal{T}$ designates the parameters space. The snapshot matrix is then POD-processed as explained in Section 2.1. Therefore, at each spatial coordinate (x, y), each realization of the amplitude associated to a given parameterization $\boldsymbol{\Theta}$ can be approximated as $a'(x, y, \boldsymbol{\Theta}) \approx \sum_{k=1}^{d} a_k(\boldsymbol{\Theta})\phi_k(x, y)$, where $d \in \mathbb{N}$ is a chosen POD approximation rank.

The EVR defined in Equation (2) is calculated for each POD approximation rank. More than 99% of the variance is already captured by the first mode, and the problem is therefore highly reducible. The spatial components of the first four POD modes are plotted in Fig. 5. The first mode shows a gradual damping of the amplitude in the cross-shore direction. Its spatial values are all positive and the corresponding POD coefficient is strictly positive as well.

As the coefficient directly multiplies the spatial mode, it plays, at the same time, the role of a magnitude enhancer and a gradient intensification. Indeed, the higher the coefficient, the higher the amplitude at the ocean boundary, and the higher



(a) Mode $\Phi_1(x, y)$ (b) Mode $\Phi_2(x, y)$ (c) Mode $\Phi_3(x, y)$ (d) Mode $\Phi_3(x, y)$

Fig. 5. The first four spatial patterns of the POD applied to aquifer toy problem.

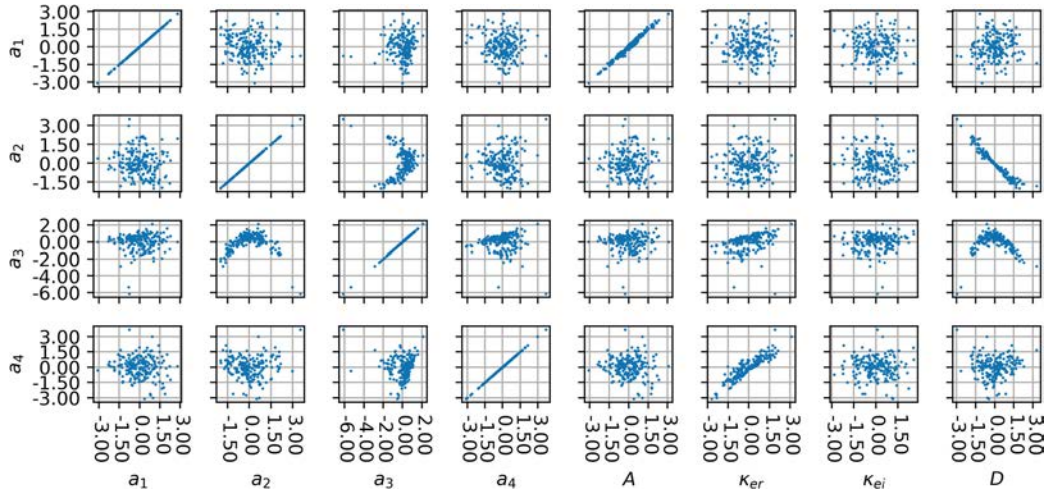


Fig. 6. Scatter plot of the first four POD coefficients and control parameters of the aquifer. The variables are centered and reduced.

the difference between the latter and the aquifer amplitudes. The second spatial mode plays a regulation role, through a succession of positive and negative spatial values in the cross-shore direction. The corresponding POD coefficients are also either positive or negative. When positive, they enhance the amplitude gradient in the cross-shore direction, and the opposite occurs when they are negative. Added to that is a variation in the longitudinal direction, from the estuary onward. The third modes looks similar to the second with added spatial details, whereas the fourth mode puts more emphasis on the damping in the longitudinal direction, from the estuary onward.

A scatter plot can be used to understand the dependencies between the modes and parameters, as in Fig. 6, and confirms the previous interpretations. Namely, a clear linear dependency between Mode 1 and the amplitude A is noticed. The relation of Mode 1 to damping, that is rather related to diffusivity D and estuary coefficient κ_{er}, is however not visible, although a dispersion of the mode around the linear tendency is noticed. This dispersion may be related to D or κ_{er}, even in smaller proportions, or to possible interactions, later clarified using PCE. The dependency of Modes 2 and 3 to the diffusivity D is also obvious, and the shapes indicate existing non-linearities. Mode 4 is highly dependent on the estuary amplitude damping coefficient κ_{er}, and no obvious dependency to the wave number in the estuary κ_{ei}, whatsoever, is noticed.

The dependencies that may explain the dispersion of the clouds around their main shapes need to be investigated. Hence, PCE models (theory in Section 2.2) can be used to detect additional physical relationships. They are learned from the data for each POD coefficient as $a_i = \mathcal{H}_i(\boldsymbol{\Theta})$, using Hermite polynomials (orthonormal basis with regards to the used Gaussian marginals). The statistical set is separated to a learning set of size 150 and a prediction set of size 50. The PCE polynomial degree is optimized for each mode separately. Degrees from 1 to 7 were tested, and the associated relative empirical errors on the training and prediction sets, respectively denoted ϵ_T and ϵ_P, were calculated as in Equation (16). The PCE degree that minimized the training and prediction errors for each model was chosen.

The optimal PCE fitting for the first four modes shows good point-wise evaluations on the learning and prediction datasets. PCE performs better for the modes of higher variance percentages. The smaller the variance rate, the higher the errors.

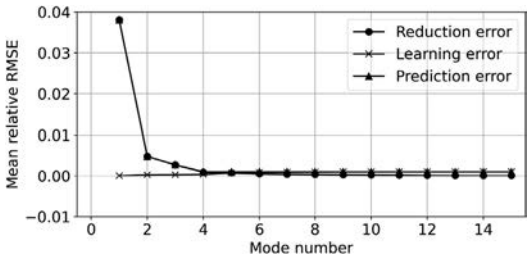


Fig. 7. Mean relative RMSE generated at different steps of the POD-PCE ML applied to the aquifer case, with different approximation ranks.

Consequently, in this particular case, PCE succeeds in constructing causal models for the first four modes, but stops at an average evaluation (constant) for modes of higher ranks (smaller variance). For illustration, the prediction relative empirical errors ϵ_P of the first four modes are 6×10^{-5}%, 4×10^{-3}%, 9×10^{-2}% and 3×10^{-1}% respectively. At least one order of magnitude of precision is lost at each rank. The relative prediction residuals between the POD coefficient and their PCE estimation are also calculated for each sample member. For the first four modes, 90% of the absolute relative residuals are lower than 1×10^{-2}%, 6.9%, 5.65% and 37.9% respectively.

The good performance of PCE encourages its use for POD-PCE prediction, as well as for physical interpretation. Firstly, in order to choose the adequate number of POD modes for the full model, the errors generated by the various steps of the algorithm (POD, PCE and coupling) are analyzed. To do so, the mean relative RMSE (averaged over the prediction set, as in Equation (18)) was calculated for each step and for each approximation rank d, as follows:

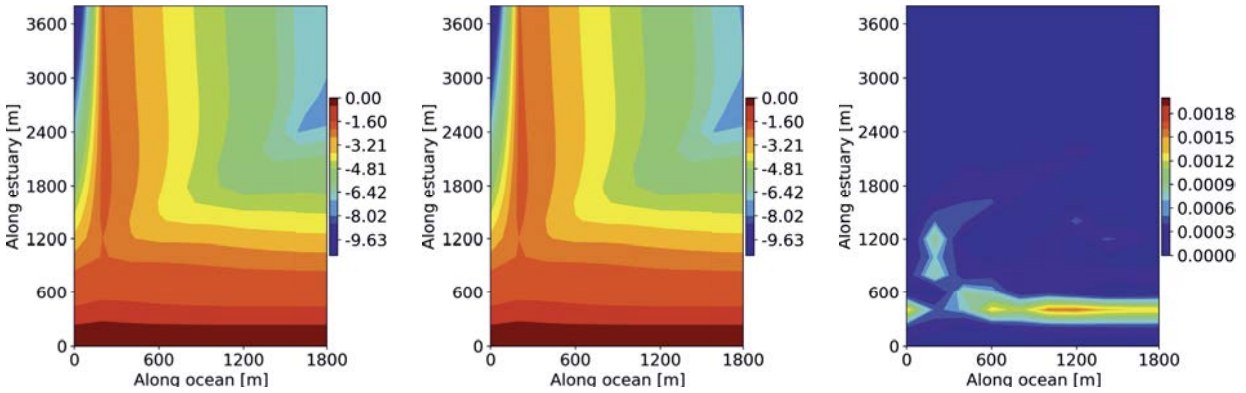
- *Reduction error:* distance between the POD approximation $\sum_{k=1}^{d} a_k \mathbf{\Phi}_k(x, y)$ and the corresponding amplitudes two-dimensional field $\mathbf{a}'(x, y, \mathbf{\Theta})$;
- *Learning error*: distance between the POD approximation $\sum_{k=1}^{d} a_k \mathbf{\Phi}_k(x, y)$ and the prediction using the POD-PCE coupling formulated as $\sum_{k=1}^{d} \mathcal{H}_k(\mathbf{\Theta}) \mathbf{\Phi}_k(x, y)$;
- *Prediction error:* the resulting final error between the prediction using POD-PCE coupling and the corresponding amplitudes two-dimensional field $\mathbf{a}'(x, y, \mathbf{\Theta})$.

The results are shown in Fig. 7. Reduction error decreased from 3.8% at rank 1, to 9.5×10^{-2}% at rank 4. The decrease is exponential, with a stabilization starting from rank 4. The learning error increased from 8.8×10^{-3}% at rank 1 to 4.1×10^{-2}% at rank 4, which is related to the increase in the PCE prediction error of the POD modes coefficients. In fact, a prediction of rank $d+1$ has an additional temporal coefficient that is predicted as compared to rank d. It is therefore natural that the distance between the approximation $\sum_{k=1}^{d} a_k(\mathbf{\Theta}) \phi_k(x, y)$ and its prediction $\sum_{k=1}^{d} \mathcal{H}_k(\mathbf{\Theta}) \phi_k(x, y)$ increased with increasing rank. The learning error order of magnitude keeps however low. Hence, the prediction error trend is almost identical to reduction error, decreasing from 3.8% at rank 1 (identical to reduction error) to 1×10^{-1}% at rank 4, where it stabilizes. It is the balance of, on the one hand, the increase in accuracy by adding POD modes and, on the other hand, the increase in forecasting error with increasing number of POD coefficients to be predicted. Hence, a 4-Modes POD-PCE model was selected for prediction.

An example of prediction is shown in Fig. 8. The model gives good qualitative estimation of the two-dimensional amplitude distribution along the estuary and through the aquifer. Slight differences may be noticed however between the analytical solution and POD-PCE prediction. Namely, the absolute residuals can go up to 0.002 m, but this occurs, for example in Fig. 8-c, in a zone where the amplitude is 0.3 m, which represents a local error of 0.7%.

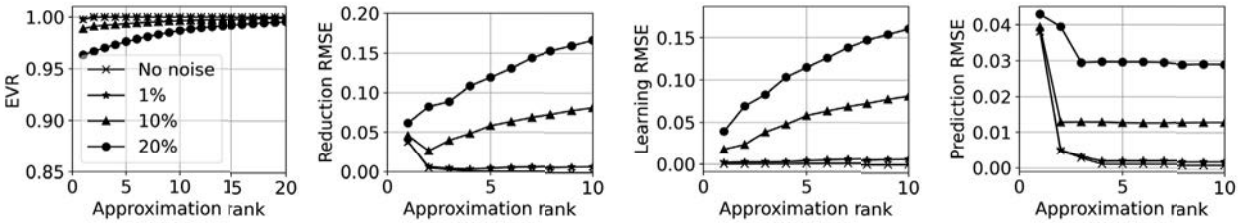
The fitted PCE models for the first four POD modes were used to rank and analyze the physical contributions. To do this, the *Garson Weights* (GW) and *Generalized Garson Weights* (GGW), respectively presented in Sections 2.2 and 2.3, were calculated for each polynomial term. The indicators values are shown in Appendix A, Table 1. The GW results confirm the great dependency of Mode 1 on the tidal amplitude A (85%), and a dispersion mainly caused by the diffusivity D (11%). Mode 2 and Mode 3 are principally dependent on the diffusivity D, and the noticed dispersion is related to the interaction of the diffusivity D with the tidal amplitude A in Mode 2 (15%), whereas it is explained by the tidal damping in the estuary κ_{er} for Mode 3 (19%). The main variation of Mode 4 is captured linearly around κ_{er}. The GGW show that the most important parameter is the tidal amplitude, with a total of 80% of influence (without interactions), followed by the diffusivity (more than 15% without interactions). The main dynamics (a total of 93%) are explained by first degree monomials. This means that the non-linearities are represented in the spatial POD patterns. Simple interactions expressed by second degree polynomials are then added to complete the dynamics, as well as other non-linear contributions, for example second to third degree monomials of the diffusivity D.

The same strategy is adopted to learn the phase, or time lag between $p(x, y, t)$ and $p(0, 0, t)$, in Appendix A. The model is optimal when 3 POD modes are used, stabilizing around a 6% prediction error. It gives a good mapping of the two-dimensional time lag distribution in the aquifer, even though more important differences between the analytical solution and the prediction are noticed compared to the amplitude prediction. The error can locally go up to 25%, but the global performance of the model remains satisfying. Lastly, calculation of GW and GGW indicators shows that the phase problem



(a) Analytical (b) POD-PCE (c) Absolute residual

Fig. 8. Analytical solution vs. POD-PCE prediction of the aquifer's log-amplitude in meters, and resulting absolute residual of the amplitude.



(a) EVR (b) Reduction RMSE (c) Learning RMSE (d) Prediction RMSE

Fig. 9. EVR and POD-PCE steps RMSE with different noise levels added to the aquifer toy problem.

involves higher polynomial degrees, and higher orders of interaction. Additionally, the wave number in the estuary κ_{ei}, which did not appear as an influencing parameter for the amplitude distribution, is necessary for the phase representation.

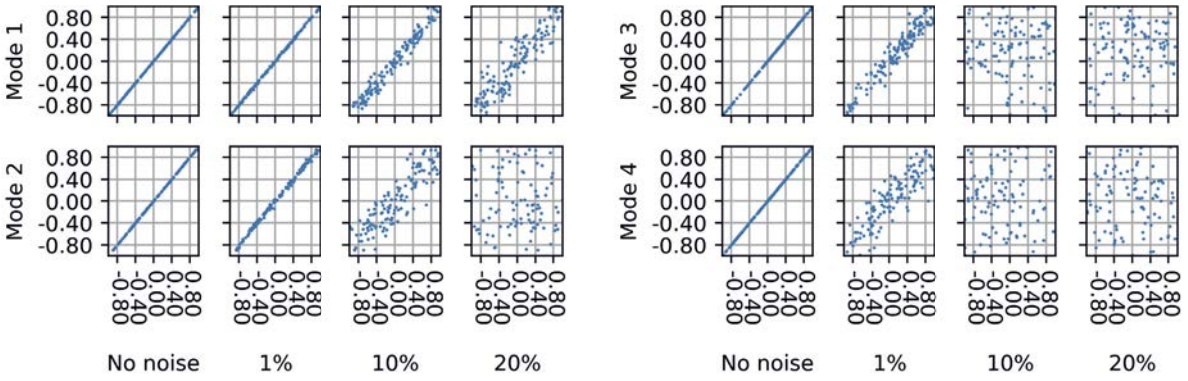
3.3. Robustness to noise

In this section, three noise levels (1%, 10% and 20%) are added to the data in order to evaluate the POD-PCE methodology. Perturbations are directly added to the $2D$ amplitude distributions in the aquifer. For each realization of the latter, the local value at a given location is perturbed using a zero-mean Gaussian PDF (white noise), with a standard deviation calculated as the average local value over the ensemble, multiplied by the chosen noise level percentage.

The EVR are compared for the different noises in Fig. 9-a. The represented variance is smaller with the same approximation rank for the noisiest data. This is natural because the variance of the random Gaussian noise is added, and the highest the noise level, the more it is statistically important. Hence, the POD modes are either impacted with random dispersion, or are totally random. A scatter plot, where the modes at different noise levels are plotted against the original modes without noise, is shown in Fig. 10, where dispersion is clearly visible.

The perturbed data are then used to evaluate the POD-PCE methodology. The reduction, learning and prediction average RMSE are calculated for each approximation rank as previously described in Section 3.2. The results are shown in Figs. 9-b, 9-c and 9-d respectively. A difference in the calculation should be noted however: for the reduction and prediction steps, the RMSE are evaluated between the original field (without noise), and the approximation resulting from the noisy field. For reduction for example, if the original field is denoted $\mathbf{a}'(x, y, \boldsymbol{\Theta})$ and the noisy field is denoted $\mathbf{b}'(x, y, \boldsymbol{\Theta})$, the POD approximation $\sum_{k=1}^{d} a_k \boldsymbol{\Phi}_k(x, y)$ is deduced from realizations of $\mathbf{b}'(x, y, \boldsymbol{\Theta})$, but the RMSE is calculated between $\sum_{k=1}^{d} a_k \boldsymbol{\Phi}_k(x, y)$ and the original field $\mathbf{a}'(x, y, \boldsymbol{\Theta})$ that represents the "truth".

Firstly, it can be noticed in Fig. 9-b that the higher the noise, the more difficult the reduction. This is coherent with the previous EVR analysis. Additionally, while reduction error decreases with the mode number up to 1% of noise, it may increase with the mode number for higher noise levels. Indeed, when noise perturbs the modes, adding them to the approximation may move the resulting field away from the "truth" (original field without noise). Secondly, Fig. 9-c shows that learning is more difficult with noisy data. In fact, if the higher rank modes are purely random, then it is impossible for PCE to provide a causal model from the inputs. If a given mode contains a physical information and a random perturbation at the same time (dispersion in Fig. 10), PCE may succeed in capturing the physical dependencies, and is shown to be robust up to 30% of noise in [33]. However, in both cases, the PCE expansion does not represent pure randomness and the learning error naturally increases with the noise level.



(a) Modes 1 and 2

(b) Modes 3 and 4

Fig. 10. Original vs. noisy POD modes resulting from 2D perturbations. The plotted data are centered and reduced.

Lastly, for the prediction RMSE (Fig. 9-d), the 1% and 10% noisy data perform similar to unperturbed ones, where prediction error follows the same decreasing trend as reduction error, then stabilizes to a minimal value where learning is not interesting anymore. Conversely, for the 20% noise level, prediction error decreases from Mode 1 to 3 although reduction error increases. This can be explained by the fact that PCE succeeds in detecting the physical patterns (explainable with inputs), while eliminating the noise from the approximation. Adding the PCE models to the POD-PCE prediction, contribute in the constitution of realistic physical fields (prediction error decrease), while directly adding the noisy POD modes moves the approximation away from the "truth" (reduction error increase).

As a conclusion, even with a maximum of 20% of added noise, a POD-PCE model of rank 1 that could be considered as the simplest approximation, does not exceed an average of 5% RMSE compared to the "truth" for the amplitude prediction. The most optimal model in this case decreases to 3% of RMSE.

3.4. Summary of the POD-PCE ML performances on a toy problem

The proposed POD-PCE ML approach was applied to this toy problem for different motivations. Firstly, the coherence of physical interpretation needed to be confronted to reality for validation, which is here possible due to the availability of analytical solution. Secondly, demonstrating the proposed ML capacity on a parametric problem is complementary with the application to a temporal problem, as in Section 4, and allows at the same time clarifying the learning steps on a simpler case. Lastly, assessing the robustness of the methodology to noise was a capital question to investigate, and here made possible by adding artificial perturbations of different levels to the data.

Firstly, the POD spatial patterns were interpretable, and the associated coefficients show dependencies to the control parameters. The physical analysis was completed using PCE and inputs ranking indicators (GW and GGW). For example, it is noticed that while the wave number in the estuary, denoted κ_{ei}, has no influence on the amplitude distribution, it controls however the time-lag in the aquifer (see Appendix A). This completely makes sense regarding the analytical formula. Next, the errors at the different POD-PCE algorithm steps were analyzed. While a gain in accuracy is established by increasing the POD modes number in the reduction phase, the learning error using PCE is inversely increased with model complexity. However, only a small number of modes is necessary for an accurate prediction, as the average RMSE for the optimal 4-Modes model is around 0.1%, reaching a local maximum of 0.7% for the amplitude. Lastly, robustness to noise was tested using different perturbation levels. The prediction's average RMSE settles around 3% even for a noise level of 20%. PCE with LARS is here of particular interest, as it allows incorporating physical dependencies in the model, while ignoring random perturbations. This assures that the method is trustworthy for an application to a purely measurement-based set-up, as in the following Section 4.

4. Application to a measurement based temporal problem

The POD-PCE ML properties are now investigated on a temporal problem. In particular, the approach is tested on field measurements, introduced with an industrial study case and inherent challenges. As is the case in many measurement based problems, noise can occur due to device errors, and the problem is characterized with data paucity. The noise problematic was treated in Section 3 on the parametric problem. However, the data paucity was not an issue, and supplementary tests are here necessary. They consist in the evaluation of learning choices (selected inputs, marginals, polynomial basis) and thorough analysis of the statistical convergence at different learning stages. The latter is of capital importance to demonstrate the trustworthiness of the physical analysis, as no analytical model is available to confront the conclusions. In particular, convergence as a function of the training set size, with associated confidence intervals, is shown in Subsection 4.2.1 for POD and in Subsection 4.2.2 for PCE. Additionally, different learning set-ups, with different marginals choices and input configurations are compared in Subsection 4.2.2. Lastly, random selection of the training members allows presenting probability

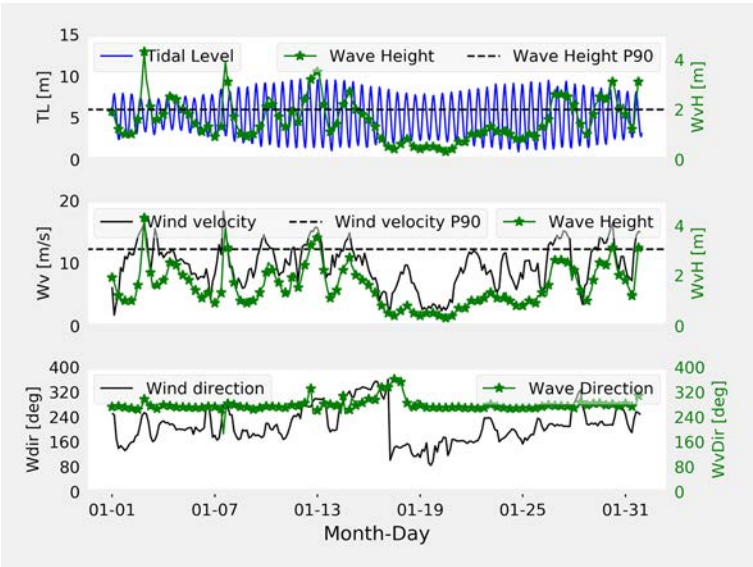


Fig. 11. Measurements of Tidal Level (TL), Wind velocity (Wv), Wind direction ($Wdir$), Wave Height (WvH) and Wave Direction (WvD) on January 2016. (P: Percentile.)

distributions of the GW indicators in Subsection 4.2.2. This helps demonstrating the robustness of physical interpretation to the learning set selection.

The physics, data and industrial context are described in Subsection 4.1. Subsection 4.2 deals with application of the POD-PCE learning phase to the data and assessment of accuracy and robustness with respect to the numerical choices (data set, inputs, marginals and polynomial basis). Finally, the prediction phase using POD-PCE is dealt with in Subsection 4.3, and the ability of the proposed ML to predict mean quantities and spatial details is demonstrated.

4.1. Study case

Sedimentation processes in nearshore areas can be responsible for the excessive sediment deposition commonly observed in cooling water intakes in power plants. As a result, the carrying capacity of the water intake can be drastically reduced, by decreasing its effective area of transport [50]. Cooling water intakes usually incorporate jetties, of which the angle with the shoreline and position relative to the direction of the net longshore sediment transport influence the amount of sediments diverted into the channel inlet by waves and tidal currents. Jetties also reduces littoral drift, resulting in localized sediment accretion against the shore-normal structure due to the longshore sediment transport being trapped by the jetty [51]. In addition, a return current is prone to develop, in the form of a swirling vortex at the end of the structure, and can induce sediment deposition in the vicinity of the channel entrance, consequently affecting the amount of sediments delivered into the cooling water intake [52]. Consequently, effective water intake management involves frequent dredging, with high operational costs and usually hindered by a tight schedule. It is then necessary to assess intake sedimentation under different natural forcing and plant operation scenarios in order to optimize dredging operations to help mitigate the potentially adverse impact of the waves, tidal currents, and meteorological forcing combined with plant functioning.

Site characteristics

The study site is located on the eastern English Channel coast in northern France. Tide in the study zone is classified as mega-tidal and is dominated by semi-diurnal circulation, with low-tide water depth of $10-15$ m, and a mean tidal range of approximately 8.5 m, reaching 10 m during the spring tide [53]. Hydrodynamics are influenced by asymmetrical current velocities, with flood and ebb currents in the E-NE and W-SW directions, respectively. Current velocity at 2.2 m above seabed vary from 0.70 to 0.98 m/s, depending on flood/ebb phase, respectively [54]. Wave activity in this open exposed environment is moderate, with significant annual and decennial wave height of 3.8 m and 4.7 m, respectively, with maximum values of $4.2-5.8$ m, averaged period of $7-9$ s and a predominant W direction. Orbital velocities measured during the spring-tide period ranges between 0.5-1.3 m/s. An example of tidal levels, wind direction and velocity and wave height and direction in January 2016 is shown in Fig. 11. In the study area, bed sediment varies from medium to fine silted sands, with a morphology characterized by the presence of mega-rides parallel to the coast. In this zone, rock occupies less than 4% of bed surface [52].

Data

Hydrodynamic and meteorological information comprise wave and wind variables, provided by the VAG prediction model of the sea state [55], using retrospective 3-hourly simulations between 2009 and 2018. Tidal water levels were obtained from the SHOM-REFMAR tide gauge station located in the vicinity of the study zone, with hourly survey frequency [56]. Bathymetric measurements were available from Single-Beam Echo Sounding on 39 cross-sectional profiles of intake measured at 25 m intervals, collected fortnightly between 2005 and 2018. Mean profiles were 100 m long with 0.5 m spatial resolution of bathymetric data. Additional information such as the daily coolant flow rates, and channel dredging volumes and frequency, were provided by the plant operator.

The available measurements of the forcings did not have the same frequencies. One solution to homogenize frequencies consists in reducing the measured data to representative statistics over the sedimentation interval $\Delta t \approx 15$ *days* separating two bed elevations measurements. Hence, the following statistics were used:

- *Tidal level indicators:* average low tide ($TLmean$), minimum low tide ($TLmin$), maximum tidal range ($TLrange$) and standard deviation ($TLstd$);
- *Wind indicators:* average wind velocity ($Wmean$) and average direction weighted by velocity ($Wdir$);
- *Wave indicators:* average wave height (WvH), standard deviation ($WvstdH$), average wave period and average wave direction weighted by height (*resp.* $WvperH$ and $WvdirH$), average wave height exceeding the 90th percentile (arbitrary storm indicator, $Wv2m$) and percentage occurrence ($Wv2m\%$);
- *Operational indicators:* average pumping flowrate (Qmean); time lapse since last dredging (Dp), and last dredged volume (Dv).

These statistical indicators were calculated for each sedimentation interval, and may be characterized with correlations. For example, a positive correlation was noted between mean low tide $TLmean$ and wave parameters $WvperH$ and WvH. Mean wave periods $WvperH$ and mean wave heights WvH were also positively correlated.

For the learning part, the data overlapped only over a limited period. A maximum of 60 measurements could therefore be used, with up to 15 forcing variables. Obviously, this "small data" configuration is a considerable handicap for the dimension of the problem, especially given that the variable of interest is a two-dimensional bathymetric field. However, permanent intake monitoring ensures that the data set will always grow and can be used to update the learning. This limitation shall not prevent testing the accuracy of the methodology on small sets such as are often encountered in physical applications, as attempted below, where learning and prediction using POD-PCE is applied to the described data. For the learning algorithm, input variables are needed, corresponding to the reduced statistical indicators described above, and denoted $(\theta_1, ..., \theta_V)$, where V is the supposed dimension of the problem.

4.2. Measurement-based learning of a physical field using POD and PCE

This section concerns learning the spatio-temporal bathymetric field using POD and PCE independently. The POD modes are extracted in Subsection 4.2.1 and the temporal patterns are learned from the forcing parameters using PCE in Subsection 4.2.2. Throughout this investigation, particular attention is given to the convergence of the learning and to its robustness with respect to the numerical choices. Trusted POD-PCE learning is immediately used for physical interpretation and the most important physical insights are summarized in Subsection 4.2.3.

4.2.1. Physical analysis and data reduction using POD

First, POD was applied on the bathymetry measurements. The aim was to identify morphodynamic patterns so as to better understand the sediment deposition inside the channel, and to characterize variations in depositions with the external forcing variables. After setting aside poor-quality measurements (e.g. incomplete bathymetries), a total $n = 156$ realizations were used. The bathymetry points were sonar boat measurements on $m_p = 39$ cross-sections inside the intake. Linear interpolation was performed on $m_i = 100$ fixed points for each profile, in order to express all measurements on the same grid, giving a total $m = m_i \times m_p = 3,900$ spatial points. The interpolated realizations were then stored in a snapshot matrix $\boldsymbol{Z}(\mathcal{X}, \mathcal{T}) = [z(x_i, t_j)]_{i,j} \in \mathbb{R}^{m \times n}$ and POD-processed as explained in Section 2.1. The EVR defined in Equation (2) and the mean relative RMSE between the POD approximation and the complete measurement (averaged over the realization set as in Equation (18)) were calculated for each POD approximation rank and are plotted in Fig. 12.

The first pattern represents over 94% of the variance, and explains most of the variation in dynamics. The variance percentage reached 99% at rank 14, where the mean error was slightly over 10%, decreasing to 8% at rank 20. Dimensionality reduction is therefore a realistic option for this specific dynamic problem.

This encouraged the learning and prediction attempts undertaken in Subsections 4.2.2 and 4.3 respectively. The spatial and temporal components of the first four POD modes corresponding to an EVR higher than 97% are respectively plotted in Figs. 13 and 14. The first spatial pattern (Fig. 13-a) represents the channel's slope. Its temporal coefficient (Fig. 14-a) shows regularity in time that is almost periodicity. When it increased, overall sediment deposition in the channel increased, because the difference between the upstream and the downstream bed elevations, and therefore the slope, diminished. The sediment deposition in the channel might be related to the increasing sediment supply caused by the external forcing influence. Decrease always corresponded to a dredging episode. The apparent periodicity is therefore not natural or seasonal

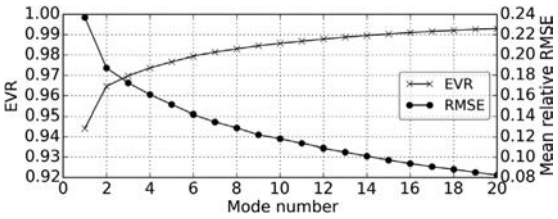
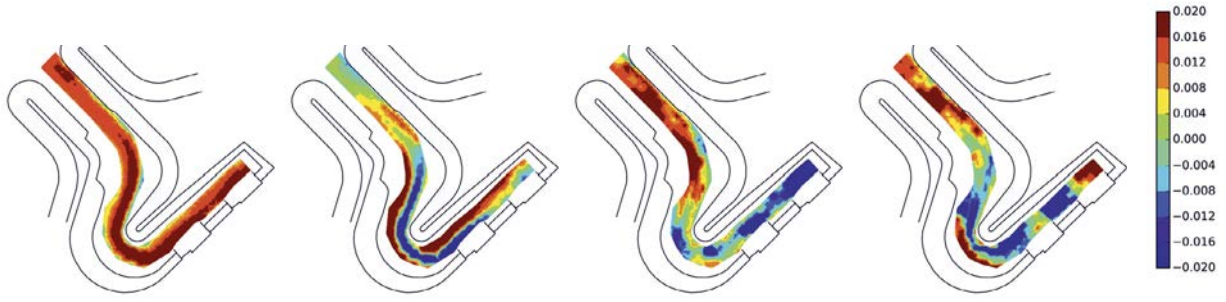
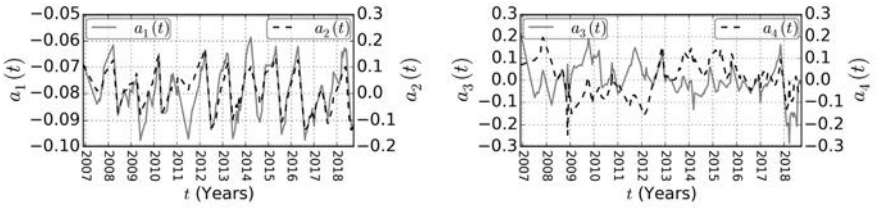


Fig. 12. Evolution of the EVR and mean relative RMSE with mode number for the POD applied to the intake bathymetries.



(a) Mode $\Phi_1(x)$ (b) Mode $\Phi_2(x)$ (c) Mode $\Phi_3(x)$ (d) Mode $\Phi_4(x)$ (e)

Fig. 13. The first four spatial patterns of the POD applied to intake bathymetries.



(a) Coefficients $a_1(t)$ and $a_2(t)$ (b) Coefficients $a_3(t)$ and $a_3(t)$

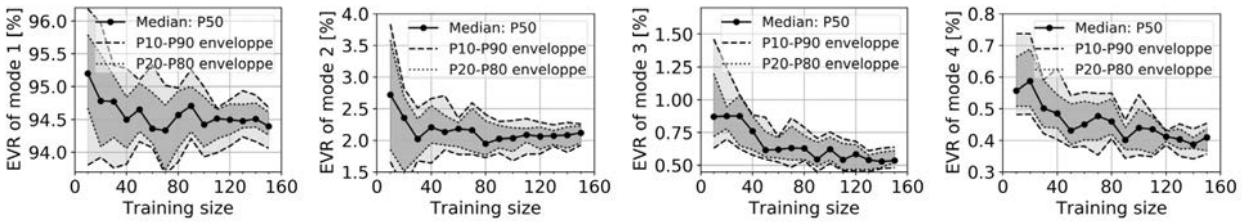
Fig. 14. The first four temporal coefficients of the POD applied to intake bathymetries.

but due to periodicity of operational intervention: sediment deposition in the channel is tolerated up to a certain level and then dredging is always undertaken at a certain point, which corresponds to the maximum of the temporal coefficient.

The second pattern (Fig. 13-b) acts as a geometric distribution function of the sediment deposition. In general, when the first temporal coefficient was maximal, the second coefficient was positive, meaning that the sedimentation mainly occurred in the middle of the first portion of the channel (upstream), on the right bank of the bend and on the left bank in front of the pumps. This spatial distribution can be associated to the internal flow characterized by a velocity distribution inside the channel. In fact, the sediments settle where velocity is the lowest, which is probably the case where the banks appear. The computed sediment deposition and erosional patterns are analogous to those commonly observed in meandering rivers [57]. The third pattern (Fig. 13-c) shows sediment deposition concentrated in the first portion of the intake, and the fourth pattern (Fig. 13-c) emphasizes sediment dynamics, particularly in the downstream part of the channel. This behavior is statistical proof and quantification of finer sediment supply. The finer sediment fraction was transported in suspension and deposited at the end of the intake channel. The temporal coefficients associated with the third and fourth mode (Fig. 14-b) were less regular than those of the first and second mode, and seemed to follow a more stochastic dynamic. The peaks may represent unusual sediment supply, probably linked to extreme events (e.g. storms).

To check the robustness of the statistical conclusions deduced from POD, convergence analysis is necessary. This was performed on the EVR values associated with the first four patterns, using bootstrap analysis [58]. The results are shown in Fig. 15. The convergence of the mean values and the tightening of the confidence intervals around the mean with increasing matrix size are clear for these first four modes. However, whereas the confidence intervals represent at most an error of $\pm 0.6\%$ around the mean for the first mode, they reached respectively $\pm 12.5\%$, $\pm 25\%$ and $\pm 12.5\%$ for the second, third and fourth modes.

The analysis proved that the POD results could be used to pursue the learning. Firstly, a high EVR and low RMSE were associated with a small number of modes, guaranteeing optimal data reduction ($d << min(m, n)$ as explained in Section 2.1). The number of POD modes to accurately represent the bathymetry can be chosen accordingly. In the present study, the configuration was $d = 11$ modes (discussed in *Step 3*), guaranteeing EVR $\geq 98\%$ and information loss $\leq 12\%$ (mean relative



(a) Mode 1 (b) Mode 2 (c) Mode 3 (d) Mode 4

Fig. 15. EVR convergence of the first four bathymetry POD modes, using a bootstrap of size 20. Plots show median values and confidence intervals (P: Percentile).

RMSE). Secondly, the EVRs were guaranteed to converge statistically at least for the first four modes, with error of $\pm 0.6\%$ around the mean for the most important mode, representing over 94% of the variance. Thirdly, the deduced patterns were physically coherent. Lastly, more than a decade of evolution was used to extract the POD basis, under variable operational and environmental conditions. As long as the operating conditions of the intake remained unchanged, it can be assumed that a wide range of evolutions has been covered, except for extreme events that rarely occur and that are not specifically treated in this study [59]. Hence, the POD basis can be considered as a physically trustworthy and mathematically complete basis to understand past evolutions and to predict future ones. The learning of temporal coefficients is therefore attempted in Subsection 4.2.2.

4.2.2. Learning of the POD patterns using PCE

The temporal coefficients calculated with the POD in Section 4.2.1 (Fig. 14) were learned using PCE (theory in Section 2.2). The aim was to learn the way these coefficients evolve over time, as a function of the forcing parameters presented in Section 4.1 , with the ultimate objective of field prediction as explained in Section 2.3 and applied in Section 4.3. The present section focuses strictly on the learning phase and the physical analysis of the learned model, highlighting quality of learning (robustness, convergence, etc.).

The investigation of learning is organized in four steps.

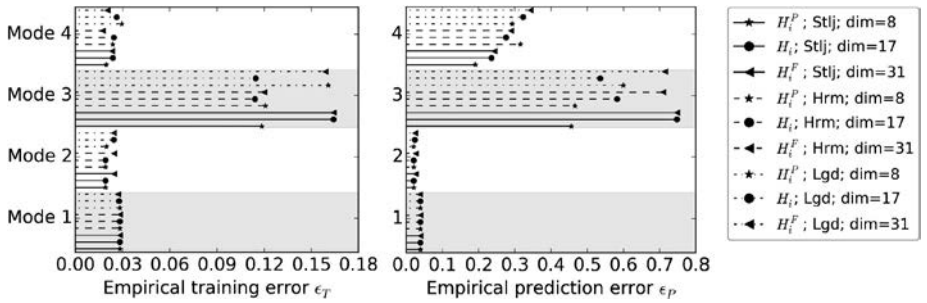
- *Step 1 - Sensitivity of learning to inputs and marginals:* different configurations were tested to practically demonstrate the implications of these choices on the accuracy of fit.
- *Step 2 - Convergence and Robustness of fit:* The best model resulting from Step 1 was studied more deeply. Its convergence and robustness with respect to the choice of training members are were analyzed.
- *Step 3 - Physical interpretation of the best learned model:* the best model was chosen, and the most influential forcings were ranked using the *Garson Weights* (GW) and *Generalized Garson Weights* (GGW) presented in Sections 2.2 and 2.3 respectively.
- *Step 4 - Robustness of the physical interpretation with respect to the learning-set members:* the physical conclusions of the model were shown to be statistically meaningful.

These steps, in the above order, follow the logic of statistical model construction to build a trustworthy prediction algorithm, used in Section 4.3.

Step 1 - Sensitivity of learning to inputs and marginals.

Input variable selection is capital, and marginals must be chosen wisely. Below, we demonstrate the influence of these choices on the performance of the learning. Different configurations were tested.

Before introducing the tested configurations, the training steps that are common to all configuration need to be defined. A learning data-set is classically separated into different sub-sets, corresponding to different steps of the learning algorithm. This is commonly referred to as the "Train-Validation-Test Split" [2]: a training set is used for the learning, a validation set is used to check the learning and for further calibration, and a test set is used to assess the prediction capability of the statistical model. However, the data-set used in this study was small: the bathymetry and forcings measurements shown in Subsection 4.1 overlap for the 2012-2017 period only, leaving 64 sedimentation periods to study. Therefore, only a "Train-Predict" split was performed, where the prediction set played the role of both the test set and validation set. Hence, the numerical choices were calibrated on the training set, and validated on the prediction set for both statistical accuracy and physical prediction. The learning was then performed with an arbitrary choice of training-set size at 50, which left a prediction set of 14. The training data were chosen in chronological order (first 50 records), to mimic the learning process in an industrial context. This arbitrary training-set choice had consequences for learning; the sensitivity of learning to training set choice was investigated (*Step 2*). All the model choices presented below (choice of inputs and marginals) were assessed on this training configuration. To assure that comparison is made between models at their best performances, the PCE polynomial degree was optimized for each separately. Degrees from 1 to 7 were tested, and the associated relative empirical errors on the training and prediction sets, respectively ϵ_T and ϵ_P, were calculated as in Equation (16). The PCE



(a) Training error (b) Prediction error

Fig. 16. The empirical training error ϵ_T and prediction error ϵ_P corresponding to the optimal fitting of models with different dimensions and marginals. The figure is organized as follows: errors are plotted for each mode vertically, separated by a gray band. Each marginal type corresponds to the same line style, and each dimension to the same marker style. The legend is shown in the order of the plots, down to top for each mode.

degree that minimized the training and prediction errors for each model was chosen; the corresponding result is referred to as "optimal" learning.

Three different input configurations were used for the learning of each temporal coefficient a_i as generally formulated in Equation (11).

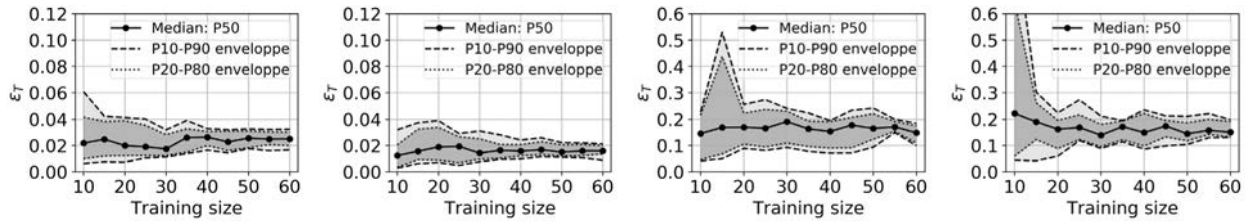
- $\mathcal{H}_i$-model: a first simple configuration where all the statistical indicators described in Subsection 4.1 were used and an independence hypothesis between the POD temporal coefficients is considered. The model is written as in Equation (13): $a_i(t_{j+1}) \approx \mathcal{H}_i\left[a_i(t_j), t_{j+1}-t_j, \mathbf{\Theta}(t_j \to t_{j+1})\right]$. This is a model of dimension 17.
- $\mathcal{H}_i^F$-model: a more complex configuration where a "Full" 15-mode POD approximation is considered with possible dependencies between the temporal coefficients. Of course, the choice of the basis size and the dependency structure can be optimized, but the objective here was to make a first step toward a more optimal configuration. The model can be written as: $a_i(t_{j+1}) \approx \mathcal{H}_i^F\left[a_1(t_j), \ldots, a_{15}(t_j), t_{j+1}-t_j, \mathbf{\Theta}(t_j \to t_{j+1})\right]$. This is a model of dimension 31.
- $\mathcal{H}_i^P$-model: a smaller set of inputs, used by the operators to qualitatively evaluate sediment deposition risk, was used. It corresponds to the six variables $TLmean$, Wvh, $Wvper$, $Wvdir$, $Wv2m$ and $Wv2m\%$. This mimics the physical expertise that may be engaged when building a statistical model. It is written as $a_i(t_{j+1}) \approx \mathcal{H}_i^P\left[a_i(t_j), t_{j+1}-t_j, \mathbf{\Theta^P}(t_j \to t_{j+1})\right]$, where $\mathbf{\Theta^P}$ stands for the "physical". This is a model of dimension 8.

To these variable choices were associated three choices of marginals, conditioning the choice of the PCE orthonormal polynomial (Section 2.2).

- *Lgd:* all the variables follow a Uniform PDF. The bounds of the marginal were set to the minimum and maximum chronological values ±1% as in [33]. The associated orthonormal polynomial basis is the Legendre family.
- *Hrm:* all the variables have Gaussian marginals characterized by the empirical mean and variance deduced from the data. The associated orthonormal polynomial basis is the Hermite family.
- *Stlj:* the marginals were inferred from the data using Gaussian Kernel density estimates. The orthonormal polynomial basis was constructed from the knowledge of the marginal using a Stieltjes orthogonalization.

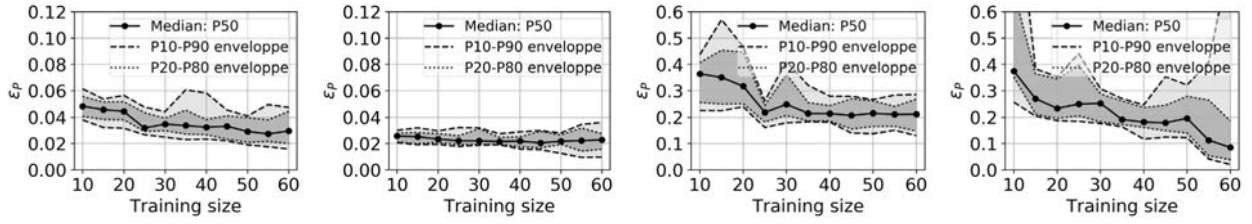
The three marginal choices *(Lgd, Hrm, Stlj)* were trained with the three dimension choices ($\mathcal{H}_i^P$ dim = 8, $\mathcal{H}_i$ dim = 17, $\mathcal{H}_i^F$ dim = 31). The empirical errors of the "optimal" learnings are compared in Fig. 16. For Modes 1 and 2, training and prediction errors were almost identical for all configurations, although with a slight advantage with the smallest dimensions for all the marginal types in the learning of Mode 2. Starting from Mode 3, bigger differences emerged. At the learning step of Mode 3, models of dimension 17 and 31 were poorly fitted for the *Stlj* and *Lgd* configurations compared to others. At the prediction step of Mode 3, the errors of models with dimension 31 were much greater than smaller dimensions for all marginal choices. There seemed to be an overfitting of the model by selecting a larger number of inputs. The best models for Mode 3 were those of the smallest dimension, 8, with either the *Stlj* or the *Hrm* model. Lastly, for Mode 4, two orderings were observed for the prediction error. Firstly, for each marginal choice, prediction error increased with dimension, which confirmed the overfitting hypothesis. Secondly, error was the smallest with the *Stlj* model (Kernel density), followed by the *Hrm* model (Gaussian) and lastly by the *Lgd* model (Uniform). Here, Uniform marginals performed worst; they were probably too different from the real data marginals and did not account for particularities in the inputs. In the parametric family, Gaussian marginals probably fitted real density better.

To conclude this comparison, the best marginal choice was the Kernel density estimate. The smallest dimensions performed the best, with the *Stlj* choice for the polynomial basis. The $\mathcal{H}_i^P$; *Stlj* model of dimension 8 was therefore selected. However, the training was performed with an arbitrary split of the available statistical set. The sensitivity of the model to the learning set size and members is performed in the following step.



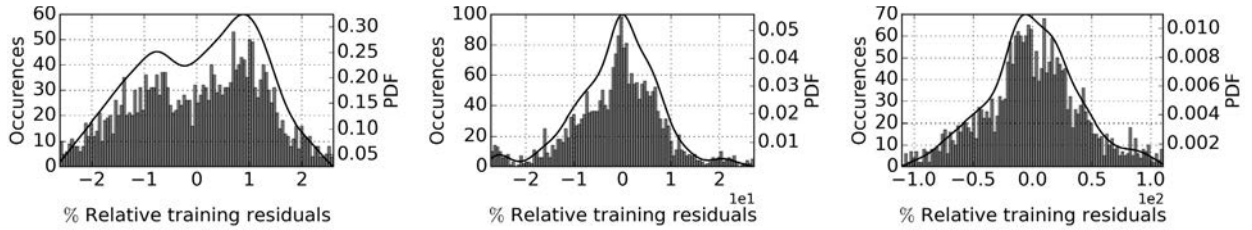
(a) Mode 1 - Stlj (b) Mode 2 - Stlj (c) Mode 3 - Stlj (d) Mode 3 - Hrm

Fig. 17. Training empirical errors ϵ_T calculated for diverse training sizes with a Bootstrap of size 20. Plots show median value and confidence intervals (P: Percentile).



(a) Mode 1 - Stlj (b) Mode 2 - Stlj (c) Mode 3 - Stlj (d) Mode 3 - Hrm

Fig. 18. The prediction empirical errors ϵ_P calculated for diverse training sizes with a Bootstrap of size 20. Plots show median value and confidence intervals (P: Percentile).



(a) First mode $a_1(t)$ (b) Second mode $a_2(t)$ (c) Third mode $a_3(t)$

Fig. 19. The training residuals distributions of the $\mathcal{H}_i^P$; *Stlj* model calculated for diverse training sizes with a Bootstrap of size 20.

Step 2 - Convergence and robustness of the fit.

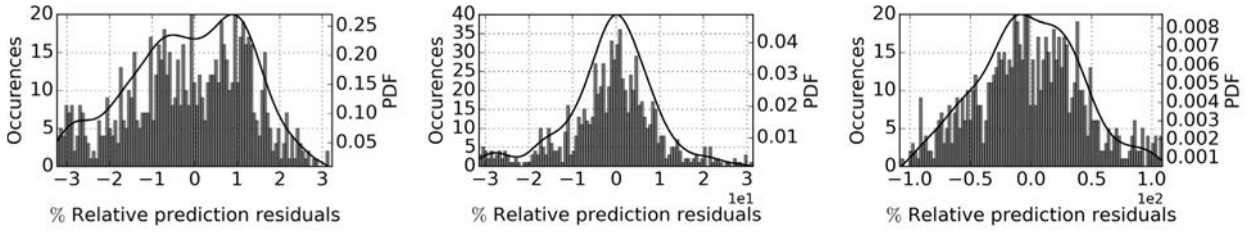
Up to this point, an arbitrary number of 50 measurements was used for training, leaving 14 prediction points for testing purposes. In the following, the influence of training set size on the learning and prediction error is assessed. The objective is to check the robustness of the previous best model $\mathcal{H}_i^P$; *Stlj* with respect to the data-set size and members. The evolution of the training and prediction empirical errors according to training set size is shown in Fig. 17 and 18 respectively. For comparison, the convergence of the $\mathcal{H}_i^P$; *Hrm* model for Mode 3 is also shown for both errors. For each training set size, members were chosen randomly among the full data-set, and the remaining members were used for the prediction phase. For the estimation of the confidence intervals, bootstrap analysis was performed [58].

For the first two modes, the training errors in Fig. 17 show a convergence of the median value and a tightening of the confidence intervals. The trainings can be considered as converging from around training size 40.

The associated median prediction errors in Fig. 18 globally decreased with increasing training set size. However, although the final median values were lower for the *Hrm* model, the confidence intervals were much larger than for the *Stlj* model. The latter seems much more robust with respect to changes in training scenario.

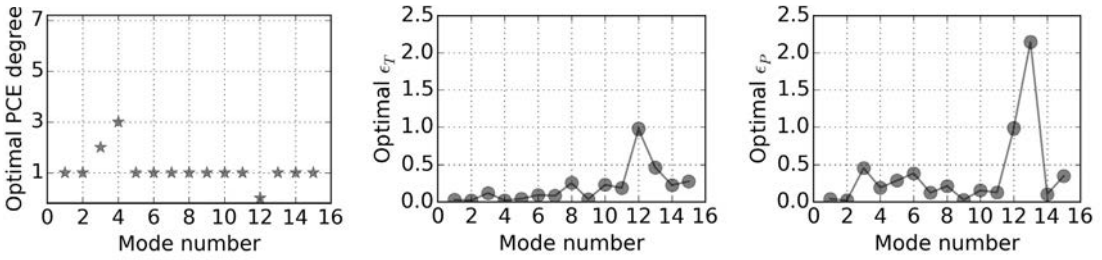
The residuals distributions of the $\mathcal{H}_i^P$; *Stlj* model, calculated as $a_i(.) - \mathcal{H}_i^P[.]$ on all the training sizes and Bootstraps, are shown in Fig. 19 and Fig. 20 for training and prediction, respectively.

Only the middle 80% portion of the residuals range is plotted, in order to analyze the center of the distribution; the full residuals distribution was long tailed, because the confidence intervals associated with small training set sizes were too large and produced extreme behaviors of the model. The training residuals were generally centered around zero: i.e., the models are unbiased. A slight asymmetry was, however, observed for Mode 1, which means that $a_1(.)$ was more often over-estimated by $\mathcal{H}_i^P$; *Stlj*. Consequently, the mean elevation in the channel and the mean global sedimentation may be slightly exaggerated. These exaggerations, however, remained within a reasonable range, as most of the residuals fell within the $\pm 2\%$ interval. The training residuals of Modes 2 and 3 were perfectly centered, but percentage error dramatically increased. Most of the residuals fell within the $\pm 10\%$ interval for Mode 2, whereas they reached $\pm 50\%$ for Mode 3. However, this error



(a) First mode $a_1(t)$ (b) Second mode $a_2(t)$ (c) Third mode $a_3(t)$

Fig. 20. The prediction residuals distributions using *Stlj* model of dimension 8 calculated for diverse training sizes with a Bootstrap of size 20.



(a) PCE degrees (b) Training errors (c) Prediction errors

Fig. 21. Optimal PCE degrees for the $\mathcal{H}_i^P$; *Stlj* model and associated empirical errors of the training (ϵ_T) and the prediction (ϵ_P) sets.

concerns modes that represent at most 4% of the total bathymetry variance, as more than 96% of the total variance was already captured by the addition of the first two modes.

The residuals shapes (i.e. slight overestimation for Mode 1 and perfect centering for Modes 2 and 3) were maintained through the prediction phase. Furthermore, the residuals mostly fell within the ranges identified in the training phase. $\mathcal{H}_i^P$; *Stlj* model behavior was stable. The prediction uncertainty could therefore be measured and trusted and the physical interpretation was consequently robust, as discussed below in *Step 3*.

Step 3 - Physical interpretation of the best learned model.

The calibrated $\mathcal{H}_i^P$; *Stlj* PCE models were considered optimal, as they showed good fit, convergence and robustness with respect to the training choices. Here, they are analyzed to deduce physical information. Firstly, the optimal polynomial degrees selected for each mode and the associated training and prediction empirical errors are shown in Fig. 21. Linear models were optimal for Modes 1 and 2 (degree 1), and the associated errors were low for both the training and the prediction sets. For Modes 3 and 4, the optimal polynomial degrees increased, which implies higher-order contributions and/or higher-order interactions for the input variables. For modes of higher ranks, the models were either linear (degree 1) or approximated by a simple average value (degree 0). This means that LARS rejects polynomial terms of higher degrees because they do not significantly improve the learning [28].

Prediction relative empirical errors in Fig. 21-c (calculated as in Equation (16)) increased from Mode 3, but remained under 50% up to mode 11. This must be interpreted according to the meaning of this indicator: it is a measure of the missing variations (distance between the model and reality) relative to the variance of the data. It therefore represents the amount of variance that was not captured by the PCE model (also called "the fraction of unexplained variance" [60]). For example, for mode 12, estimated with a degree 0 PCE model, 100% of the variance is not captured, which is natural because only the average value is accounted for with degree 0. For Modes 3 to 11 with error up 50%, this means that either the training set or the used inputs made it impossible to predict more than 50% of the variance. However, as presented in *Step 2*, this 50% error concerned at most 4% of the total bathymetry variance. Hence, the errors starting from Mode 3 represented at most 2% of missing variance. Beyond Mode 11, prediction with PCE would not be optimal, as the prediction error dramatically increases.

Secondly, PCE models were used to analyze the contribution of each forcing variable to the dynamics. For this, the *Garson Weights* (GW) defined in Equation (7) were used to estimate the influence of the forcings on each temporal coefficient. The global influence on the whole bathymetry field was quantified using the *Generalized Garson Weights* (GGW), as in Equation (20). The ranking of the modes and the impact of the inputs is represented in Fig. 22.

Mode 1 corresponds to a major contribution and the following modes are ranked according to their POD importance. The share of each polynomial term corresponds to the GGW in relation to the global contribution (full circle). When this share is compared to the importance of the corresponding mode, it corresponds to the GW. Lastly, the polynomial terms corresponding to more than 0.5% GGW are indicated. $\zeta_{\boldsymbol{\alpha}=(.)}(.)$ corresponds to the notation introduced in Subsection 2.1, with the multi-index notation for α that represents the polynomial degree of each monomial. For example, $\zeta_{\boldsymbol{\alpha}=(\alpha_1,\alpha_2)}(\theta_1,\theta_2)$

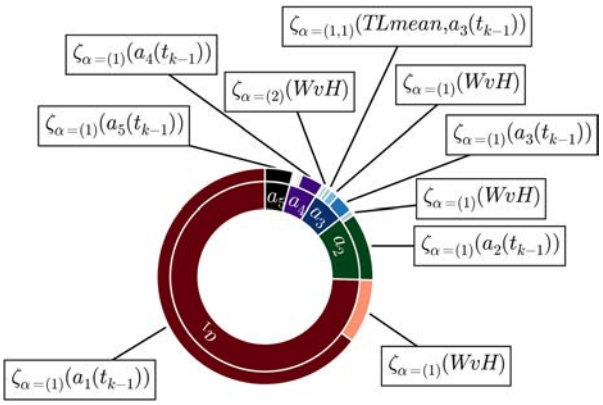


Fig. 22. Piechart of the most influential parameters, using GW and GGW on the POD-PCE. The inner circle represents the share of each mode. The outer circle represents the share of each polynomial term. The polynomial terms corresponding to GGW higher than 0.5% are shown.

corresponds to a polynomial of degree $\alpha_1 + \alpha_2$, where θ_1 contributes as a monomial of degree α_1 and θ_2 as a monomial of degree α_2. The meaning of the variables that appear in Fig. 22 can be found in Subsection 4.1.

For all the temporal coefficients $a_i(.)$, the most influential contributor by far was the value of the previous state $a_i(t_{j-1})$, in the form of a monomial of degree 1. It is followed by contributions involving the mean wave height during the sedimentation period period WvH for all the modes, which makes WvH the most important external forcing, figuring in the third position among all the forcings, with a contribution of 9.6% through the first mode, a total of 12.6% if only WvH monomials are considered, and 13.3% if interactions with other variables are taken into account.

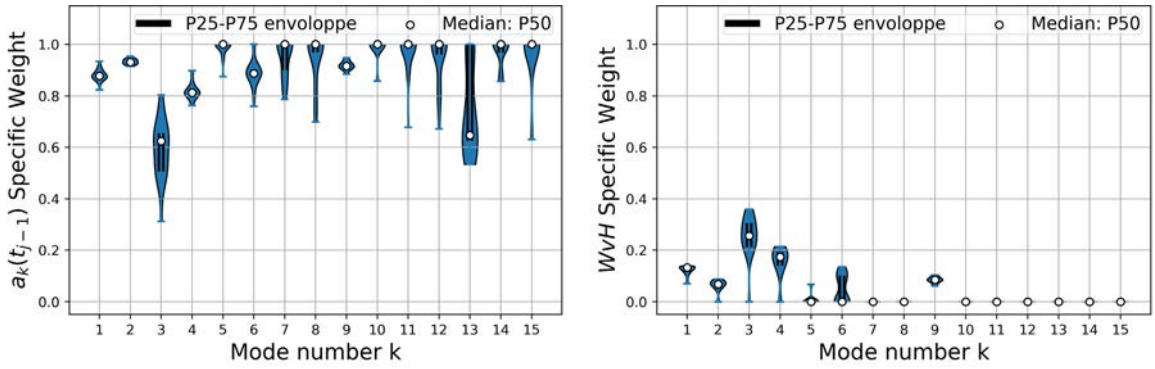
The other forcing contributions also appeared, but with much less importance: e.g., the influence of mean low tide level $TLmean$, which took an interaction form with the previous bathymetry shape for Mode 3. Firstly, this interaction makes sense in terms of physics, as sediment deposition is conditioned by the value of bed shear stress [61], which depends on velocity and water depth. The water depth value is exactly the tidal level minus the bed elevation value, which here appears as a multiplicative interaction between $TLmean$ and Mode 3. Second, the value of this contribution was only 0.7% GGW, which is negligible when compared to the first contribution of WvH. The learned model gave much more importance to waves than to tides. This does not necessarily mean that tides have no influence on sediment deposition, but may simply suggest that, in the present configuration, sediment mobilization by the tide is always more or less the same, and that the forcing that makes a considerable difference is the variation in wave heights. Waves are a determining factor for sediment mobilization in coastal configurations, through the influence they have on bed shear stress [61]. Further more, a noticeable correlation between WvH and $TLmean$ was noticed in the used data-set, which means that the information of low-tide levels is to a certain extent contained in the mean wave height. There is therefore a probable dependency between these variables. In case of dependencies, the iterative process used by LARS may drop a variable that is physically important because the important information is already contained in another variable, due to their dependency.

Lastly, it is important to note that the contribution of less frequent wave events was also present but to a much smaller extent. It is represented by a polynomial term in the form $\zeta_{\alpha=(1,1)(Wv2m, Wv2m\%)}$, where $Wv2m$ and $Wv2m\%$ are respectively mean wave height exceeding $2m$ and the associated frequency of occurrence (arbitrary storm indicator chosen in Subsection 4.1). This term appears in Modes 3 and 4 for a maximum total influence of 0.3%. Higher-order interactions and less frequent events are therefore represented by modes of higher rank, associated with smaller variance percentages.

Step 4 - Robustness of the physical interpretation with respect to the learning-set members.
As a last proof of the robustness of the proposed learning algorithm, specifically concerning physical interpretation, a sensitivity analysis with respect to the training set members was performed. The robustness of the calculated *Garson Weights* (GW) with respect to the choice the training members was studied. This is equivalent to studying the robustness of the polynomial basis term selection as produced by LARS, and their associated multiplicative coefficients.

For this, a Bootstrap analysis was again used to construct different learning sets of size 50, instead of choosing the first 50 measurements. This produces a distribution of the GWs rather than a single value, for each polynomial term. The result is shown in Fig. 23 for the weights of the $a_k(t_{j-1})$ and WvH monomials.

For modes 1 to 4, the median weights P50 (Percentile 50) of the $a_k(t_{j-1})$ monomials, represented in Fig. 23-a, were always over 0.6, but the variation range was strictly less than 1, with density functions centered around the median and a small standard deviation for modes 1, 2 and 4. This means that whatever the training set, the previous state $a_k(t_{j-1})$ value was always predominant but never enough to estimate the evolution of the first four modes. A tendency (in particular linear) using the last state was not sufficient, and additional information was always needed (forcing). In parallel, Fig. 23-b shows that this information is certainly the waves, as the median values of the GW for the first four modes were between 10 and 25%, corresponding to the information gap left by the previous value variable $a_k(t_{j-1})$ in the fitted PCE model. Starting from mode 5, the median values of $a_k(t_{j-1})$'s GW had greater chance of falling around 1, which means that the associated



(a) Polynomial term $a_k(t_{j-1})$ (b) Polynomial term WvH

Fig. 23. Probability density functions of the GWs associated with the degree 1 monomials of variables $a_k(t_{j-1})$ and WvH. The training size is 50 using 20 different random picks for each size.

polynomial models only rely on the last recorded value of the mode for the future guess. In other terms, the constructed model consists of a linearization around the previous value (tendency capturing) and does not incorporate the correlations between the future-state and the forcing variables (causality model). This can be explained by the small variances of the higher-rank modes and the difficulty of learning the PCE models from statistics averaged over the sedimentation periods. Additionally, the P25-P75 confidence interval moved to the upper bound of the density functions.

4.2.3. Summary of the physical insights from the learning

The spatial patterns as deduced by POD express the spatial correlation in the sediment deposition from the upstream to the downstream part of the channel. The EVR reached 99% with $d = 20$ modes only, where the mean relative RMSE between the approximation and reality was slightly over 10%. This is a statistical proof that the spatial correlations expressed in the POD patterns are explanatory of the physical dynamics over their whole range of variation (at least that observed from 2010 to 2018), with a low approximation rank. In conclusion, the dynamic problem exhibits fairly strong spatial correlations, and the solution to the problem can be expressed on a finite orthonormal basis.

The temporal patterns express the evolution of the sedimentation, as they multiply the spatial patterns. They were learned using PCE as a function of the previously cited inputs (previous states and forcings). The statistical model configuration (dimension and marginals) was chosen after an investigation of different options. The associated training and prediction error converged for the first three modes, and are characterized by tight confidence intervals. The residuals of the selected model were either negligible or centered around zero, demonstrating the unbiased character of the learning and prediction. The fitted models are of lower degree for the low-rank modes 1 and 2 and of higher degrees for modes 3 and 4, which are higher-rank, due to the emergence of interactions between the forcings, namely variables related to extreme behavior (storm events). The model mainly relies on the last state information, showing a strong correlation/continuity in time of the studied physics. Using GW, which measures the forcing influence for the first five modes, the action of waves was highlighted by the PCE model as a determining phenomenon. The first mode influenced the dynamic with a rate of 64.9%, the previous value of the second mode with a rate of 10.2% and, in third position, the mean wave height with a rate of 9.6%. The remaining 15.3% is essentially associated with previous values of higher order modes (10.3%), interactions with tides and contributions of other wave indicators. The GWs show robustness with respect to the choice of the training set members, which makes them trustworthy, at least for temporal correlation and analysis of wave influence. The main physical conclusions are that the dynamic problem is characterized by strong temporal correlations, representing more than 85% of the evolution, with an external sediment source, mainly represented by the waves, representing not more than 15%.

4.3. Prediction of a physical field using POD-PCE coupling

After performing both POD and PCE independently, the accuracy of a Machine Learning process using a POD-PCE coupling was assessed as in Section 2.3. In the continuity with Section 4.2, the first 50 historical bathymetries were used for training and the other 14 for forecasting. First, the impact of the size of the POD basis on the prediction process is assessed in Subsection 4.3.1. Then, the best size was determined and the average prediction behavior is analyzed in Subsection 4.3.2. The accuracy of the POD-PCE ML in predicting spatial details is assessed on cross-section examples in Subsection 4.3.3 and a summary is given in Subsection 4.3.4.

4.3.1. Influence of POD basis size

In order to track the errors generated by the various steps of the algorithm (POD, PCE and coupling), the mean relative RMSE (averaged over the prediction set, as in Equation (18)) was calculated for each step (reduction, learning and prediction) and for each approximation rank d, as described in Section 3.2. The results are shown in Fig. 24.

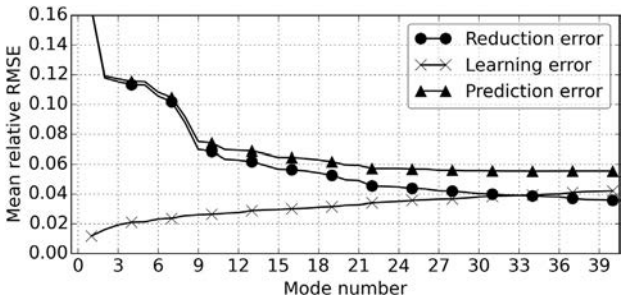


Fig. 24. Mean relative RMSE generated by the reduction and the learning, and the resulting prediction errors for different approximation ranks.

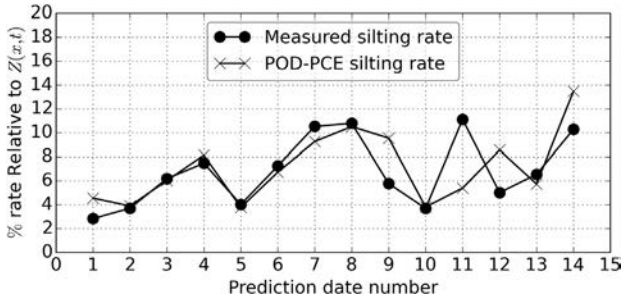


Fig. 25. A comparison between the real sedimentation rates and the POD-PCE prediction of the sedimentation.

Reduction error decreased from 16% to 3%, with increasing approximation rank. The error followed a logarithmic trend, with a significant slowdown from rank 9. These errors are coherent with the errors averaged over the full set (rather than the prediction set only) in the POD results Section 4.2.1 (around 8%). The learning error increased from 1% to 5% with increasing approximation rank, which is natural because the complexity of the model is increased. The learning error order of magnitude was consistent with the empirical prediction error of 4% for mode 1 (as calculated in Section 4.2.2), associated with an EVR of over 94%. Lastly, the prediction error decrease is the balance of, on the one hand, the increase in accuracy by adding POD modes and, on the other hand, the increase in forecasting error with increasing number of temporal coefficients to be predicted. Consequently, the prediction error decreased from 16 to 6.9% up to rank 11, following almost the same decreasing trend as the reduction error. However, the decrease rate became slower and increasingly subdued, being overtaken by the learning errors, which dramatically increased starting from mode 12, as seen in Fig. 21. Hence, a POD-PCE model of size 11 was selected for prediction.

4.3.2. Average performance of the chosen model

Average sediment deposition was predicted using the POD-PCE model of rank 11, for each of the 14 prediction dates. The average sedimentation rate, denoted S_r, was calculated for time t_j representing the sedimentation over $[t_{j-1}, t_j]$, as in Equation (23). For operational estimation of sediment deposition, only the positive evolutions are of interest; therefore, the erosion points were discarded in calculating rate S_r by canceling negative evolutions. Indeed, $z(\mathbf{x}_i, t_j) < z(\mathbf{x}_i, t_{j-1})$ implies $\big(z(\mathbf{x}_i, t_j) - z(\mathbf{x}_i, t_{j-1})\big) = -|z(\mathbf{x}_i, t_j) - z(\mathbf{x}_i, t_{j-1})|$, and therefore a null contribution to the sedimentation rate S_r. Furthermore, only regions of considerable depth are of interest. Therefore only n_p bathymetry points under -1 m ($\mathbf{x}_i, i \in \mathcal{N}_p$) were taken into account. The results are shown in Fig. 25

$$S_r = \frac{1}{2n_p} \sum_{i \in \mathcal{N}_p} \frac{\big(z(\mathbf{x}_i, t_j) - z(\mathbf{x}_i, t_{j-1})\big) + |z(\mathbf{x}_i, t_j) - z(\mathbf{x}_i, t_{j-1})|}{|z(\mathbf{x}_i, t_j)|}. \tag{23}$$

The POD-PCE prediction globally followed the real sedimentation trend, for example from Dates 1 to 8. When it was not equal to the real sedimentation rate, it was generally an overestimation, which is coherent with the asymmetry observed in the distribution of Mode 1 training and prediction residuals (Fig. 19 in Subsection 4.2.2).

For the particular Date 11 however, half of the sedimentation was missing. Investigation of the data for this particular measurement showed that the previous record, taken as input, had been made 29 days previously, which is far from the average $\Delta t \approx 15$ days; it is twice the mean interval, thus underestimating sedimentation by half. As measurement intervals were in general around the average, sedimentation time interval Δt was not selected as a key parameter by LARS for the dynamic model, although it was given as an input and is physically significant. For the particular case of the time variable, multiplicative enhancement can be intended as a correction. However, this shows one of the limitations of statistical modeling: statistical significance can be confused with physical importance. Indeed, for the statistical conclusions to be physically

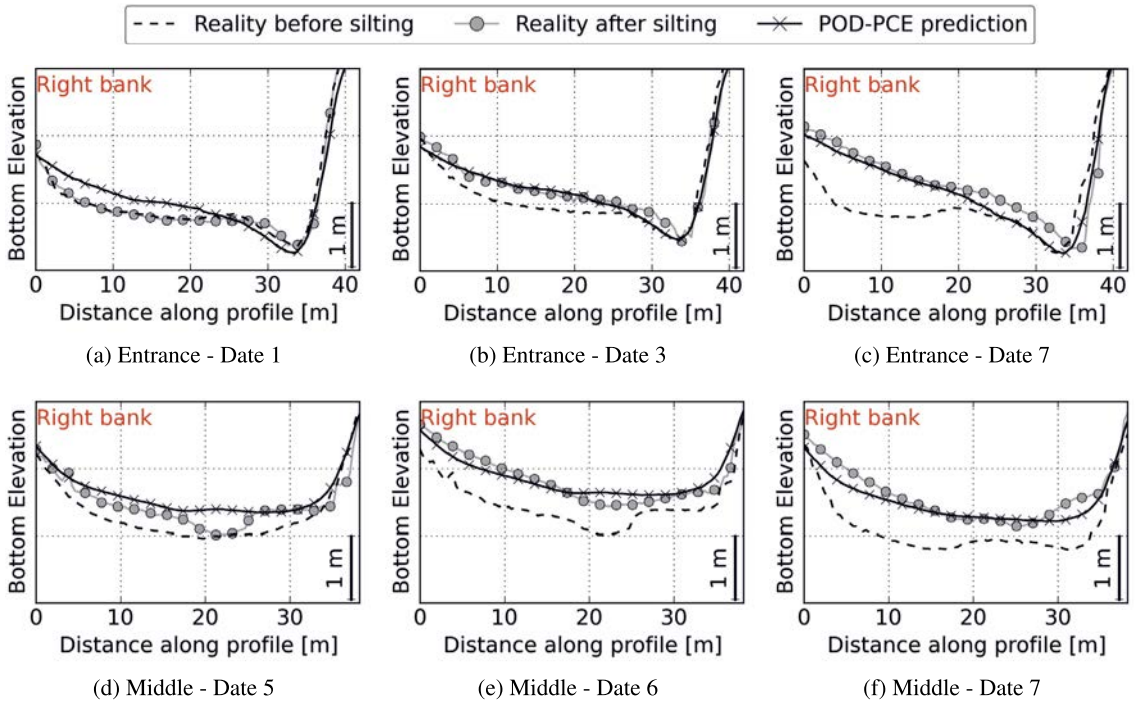


Fig. 26. POD-PCE prediction vs. reality on cross-sections at the entrance (a, b and c) and middle of the first portion (d, e and f) of the intake.

significant, the measurements should be diverse enough to account for the variations in the inputs and the impact of these variations on the output. This was unfortunately not guaranteed for sedimentation measurement intervals, as they were often equal to 2 weeks. Additionally, for Dates 12 and 13, a large part of the wave measurements were missing in the sedimentation time interval. Consequently, mean wave height WvH was estimated over only a small portion of the time interval. This may lead to a good prediction (Date 13) if the interval used is representative enough of the full interval, and to bad prediction (Date 12) when not, and highlights the limitations of statistical averaging.

4.3.3. Spatial details of prediction by the chosen model

The spatial details of the prediction were analyzed on cross-sections for specific prediction dates. First, sediment deposition was observed on a cross-section at the entrance of the intake (Figs. 26-a, b and c).

In accordance with the previous conclusions from Fig. 25, the POD-PCE prediction captured various sedimentation ranges, as shown with Dates 2 and 7. However, a slight artificial sedimentation was predicted whereas there was no dynamics in reality, for example for Date 1 (Fig. 26-a), due to the fact that the model is continuous, whereas threshold phenomena can occur in reality. Next, sediment deposition was observed on a cross-section at the middle of the first portion of the intake (Figs. 26-d, e and f). Mean sedimentation was well captured, but some details of the bathymetry were missing, such as formation of a new feature for Date 5 (distance 20 to 25 m) and Date 7 (distance 30 to 35 m). For Date 6, sediment deposition was slightly underestimated in the right bank and overestimated in the left bank. However, although the details of sediment deposition were not perfectly captured, the value of the sedimentation area corresponds well enough to reality. It can also be concluded that the way the RMSE and relative errors are averaged in space, for example in Fig. 24, actually penalizes the accuracy of the algorithm because it does not take account of the oscillation of the prediction around an accurate mean.

Then sediment deposition was observed on a cross-section at the bending part of the intake (Figs. 27-a, b and c). It shows that the prediction algorithm understood that the sediment deposition mainly occurred in the right bank of the channel, for example for Date 7, even though it was overestimated. Furthermore, considering modes of higher rank from the previous measurement as an input, the algorithm captured the swing in the profile throughout its history, which is here observed from Date 4 to 14. Lastly, a cross-section of the last portion of the channel, in front of the downstream pumping station, is analyzed (Figs. 27-d, e and f). Once again, the prediction algorithm understood where the sediment deposition occurs, this time in the left bank of the channel, coherent with the pattern represented by Mode 2 (Fig. 13-b). However, it can be seen that unusual sediment deposition occurred for Date 11, which was not captured by the model, and may correspond to the arrival of less frequent fine sediment that appears in Mode 4 (Fig. 13-d). This also explains the sediment deposition error observed for Date 11 in Fig. 25.

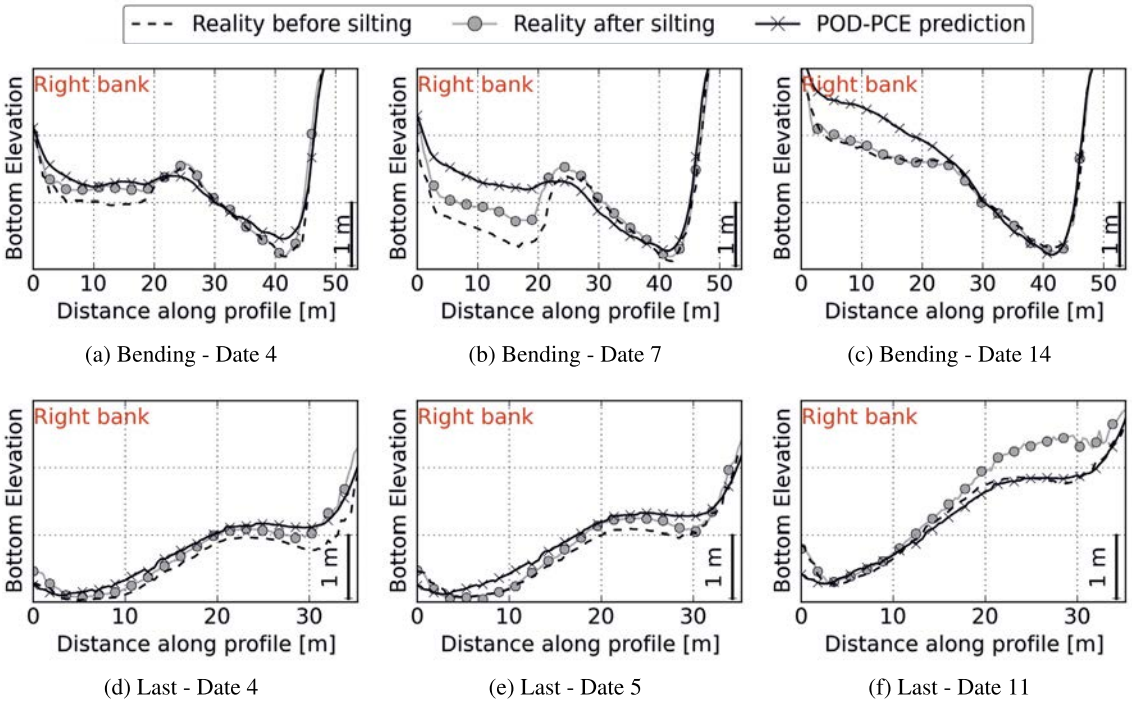


Fig. 27. POD-PCE prediction vs. reality on cross-sections at the bending (a, b and c) and last portion (d, e and f) of the intake.

4.3.4. Summary of POD-PCE algorithm performance

Overall, the proposed learning algorithm showed interesting prediction characteristics. The model complexity can be increased gradually, by increasing the number of POD modes when accurate. Plotting error against the number of modes shows a convergence that helps in selecting the optimal number of modes. The average RMSE of the predicted field remains reasonably low. It was around 6.9% with the 11 modes selected in the present case.

As a comparison, additional learnings were performed using different NN set-ups in Appendix A. It is shown that POD-PCE gives the best equilibrium between accuracy and fit-time. Among the tested algorithms, only one allowed a RMSE reduction of 0.2%. However, this is of little importance in view of the very large increase in fit-time. Indeed, the latter nearly equals two-hours whereas POD-PCE is fitted in seconds (Table 3 in Appendix A). In addition to fit-time, choosing an algorithmic set-up for NN may be time consuming, as many network architectures are possible, not to mention the numerous choices for the involved hyper-parameters (e.g. Activation Functions).

With the proposed POD-PCE algorithm, the trends are well captured for spatially averaged quantities (here sedimentation rate) and for detailed spatial representations of the field. Good spatial distribution of evolution is guaranteed by the POD modes, even when evolution amplitude is over- or under-estimated.

Nevertheless, some disadvantages should be noted. For example, less frequent events that are represented by modes of higher ranks can be overlooked. Furthermore, sudden changes in features were not sufficiently captured, due to the high temporal correlation between last state and future state that was incorporated in the learning.

5. Summary and discussion

In this study, POD-PCE coupling for field-measurement based Machine Learning was proposed and assessed on a toy problem and a real case. The first one concerns the learning of an analytical solution to groundwater perturbations in an aquifer subject to tidal solicitations, and the second, in an industrial context in the field of geosciences, concerns bathymetry prediction. Both are complex physical phenomena involving non-linear dynamics and various forcings.

POD showed excellent performance on both applications, for dimensionality reduction and physical analysis. This is an important property of POD [62], where the mathematical basis ends up to be physically interpretable, because it efficiently expresses the dynamics. However, adding random perturbations to the toy problem data showed that noise may contaminate the POD patterns (temporal and/or spatial), although modes of high variance are robust. If the noise is significant enough, it may also take more important positions in the decomposition than physical patterns of smaller statistical occurrence. It can then be interesting, for real data, to eliminate modes showing completely stochastic behavior in favor of explainable modes of lower variances. Next, the investigation of POD coefficients is also physically informative: dependency to inputs can sometimes be directly noticed with appropriate plots, and the regularity of the modes can be related either to the representation of different space and time scale physics, or to less frequent events. The potential of POD for

detecting biased and missing data was also assessed in the real case. POD was first applied to the whole set of measurements, but discontinuities emerged in the temporal signals of the decomposition. Such a procedure is important because, in most of cases, the data need to be filtered, which is a time-consuming task. The POD enabled fast recognition of elements that react differently from the average. However, many points of improvement are worth mentioning. Firstly, the choice of POD as a decomposition technique was here motivated by its simplicity and the possibility of interpretation when coupled to a linear learning formulation such as PCE. Other decomposition techniques exist, and many authors attempted comparisons, for example with Fourier [63], extensions of POD [64] or other classes of decomposition [65]. For the real case application, other decomposition techniques such as Kernel Principal Component Analysis (KPCA) [66] and Sparse PCA [67] were analyzed, without significant improvements. Secondly, data filtering using POD consisted only in deleting the poor-quality measurements and extracting the spatial zones where data were always measured. POD can however be used to reconstruct missing data, by inverse projection on POD basis elements deduced from qualitative data [68]. This could help to extend the statistical set for the learning. Lastly, a linear interpolation of the bathymetry was used to project all the measurements onto the same grid for POD application. The uncertainty that emerged from this interpolation process was not treated. This, added to the measurement errors, can shed light on model behavior. For example, comparison of mono-beam cross-sections with multi-beam measurements and uncertainty propagation of bathymetry errors through the learning could be attempted, especially because uncertainties in the bathymetric information may impact the flow field computation [69].

PCE was used to learn the POD modes coefficients as 1D data. We showed the importance of the polynomial basis, and therefore of marginals choice, for the learning phase of the real case problem. Indeed, choosing for example uniform distributions, associated with Legendre family, might not be appropriate even though it is widely used when no input information is available [33]. Moreover, the number of inputs can alter the learning. When using LARS, the presence of numerous variables can mislead the algorithm to overfitting. Hence, a good combination between polynomial basis and dimension choice could significantly improve convergence speed, centering of residuals and mean training and prediction errors. The proposed contribution analysis using the PCE coefficients has been successfully tested on the toy problem, resulting with physically coherent conclusions. On the measurements set, it showed that the last-state information is often the most influential input. A robustness test was conducted on the latter by varying the training set, and the observation was stable. Additionally, the noise tests performed on the toy problem showed that LARS selects physically significant polynomial patterns even when the noise contaminates the POD coefficients. PCE and in particular LARS are therefore robust to noise, that propagates from a two-dimensional field to its POD coefficients. This is coherent with the conclusions in [33] about PCE robustness to noise in $1D$ data. In the bathymetry case, for the modes of small ranks associated with the largest variances, wave height was the most influential forcing, whatever the chosen learning set. This is consistent with physical knowledge of sediment mobilization in coastal configurations, where waves are known to be determining through the influence they have on bed shear stress [61]. For modes of higher rank, however, the only selected variable by LARS is the last-state information. Firstly, the forcings that were used as PCE inputs were simple statistical estimators deduced from the data (means, percentiles, etc.). This reduction was used instead of giving all the time series as an input, because the problem would become ultrahigh-dimensional. This unfortunately wastes the richness of the available data as tidal information that are measured on an hourly basis. A more accurate statistical reduction of the inputs could be used. For example, Lataniotis et al. [32] used PCA and KPCA for surrogate modeling with PCE and Gaussian Processes on ultrahigh-dimensional problems. Secondly, dependencies were not specifically modeled. These can be incorporated using the mathematical setting for the construction of the polynomial basis established by Soize and Ghanem [26]. The dependencies, however, indirectly influenced the construction of the model via selection of basis elements by LARS, which avoids redundancy. Thirdly, the choice of tested input configurations for PCE was arbitrary. A more objective variable selection technique is necessary [70]. For example, the information from previous times could also be used as inputs for temporal evolution problems. This may raise other technical questions, such as the number of previous times that should be accounted for (time lag estimation) [71]. Lastly, PCE was chosen for the interpretation possibilities that it allows when combined to POD, thanks to the direct computation of importance measures from the expansion coefficients. Other interesting properties can encourage the use of PCE, for example the developed theoretical frameworks for the treatment of discontinuity [45]. However, some limitations are noted. For example, PCE worked better for modes associated with high than low variances. Although it may be tempting to conclude that modeling of high rank modes is not necessary, it should be noted that their accurate prediction can make the difference between average forecasting and capturing of less frequent events and/or smaller scale features. Therefore, the present ML could be enhanced by improving the learning of high-rank modes. For example, the construction of marginals and the use of random draw with confidence intervals, or extreme statistics models [59], instead of causal models like PCE, could be attempted.

Finally, the robustness and convergence properties added to the physical interpretability supported the choice of POD-PCE coupling as a ML prediction algorithm. It respects the PDR (Predictive, Descriptive, Relevant) framework defined in [72]. It is characterized by both predictive and descriptive accuracy (simplicity) and is stable with respect to data disturbance. It offers the best equilibrium between accuracy and fit-time, compared to other NN configurations tested on the real-case problem. This is consistent with the conclusions by Torre et al. [33], where PCE errors are comparable to the best NN from literature, on classical ML cases, while being much faster. Additionally, POD-PCE is interpretable, as the sparsity, simulatability and modularity defined in [72] are respected by construction. Finally, it is both interpretable at features level (POD components and their PCE) and at multidimensional output level (GW compared to the proposed GGW indicators).

The POD-PCE ML was therefore implemented using maximum the first 4 modes for the toy problem, and using the first 11 modes for the real case problem, after sensitivity test to number of modes. Mean information (e.g. sediment deposition rate) was in general well reproduced. Profile-by-profile investigation and $2D$ maps comparisons also showed that POD-PCE coupling was promising, as the spatial distribution of the groundwater perturbation on one hand, and the sediment deposition patch locations and amplitudes on the other hand, were well represented. Some general limitations should be highlighted and could be good perspectives for improving the process. The small data-set was a clear handicap in the measurement based problem. Some events, such as sediment downstream the intake or variation in measurement intervals, were poorly represented. It would be interesting to test the methodology on an enriched data set in order to assess the real potential of POD-PCE Machine Learning. Due to the lack of such data, input distributions were certainly not well approximated. One way of improving POD-PCE coupling would be the development of hybrid measurement-based/process-based data learning [73,74]. This could be used to enrich the data set, not only by increasing its size (emulated realistic scenarios) but also by adding new input parameters that are not measured but obtained from process-based modeling. Last, the used POD and PCE basis may not always be sufficient for fields whose dependence to conditioning parameters considerably varies over time. Namely, the PDFs of the inputs may evolve, and the number of the basis elements needed for a good representation of the output may increase in time (stochastic drift [75]). Solutions as the Time-Dependent generalized Polynomial Chaos (TD-gPC) [75] could be interesting to explore for long-time learning problems. In particular, an adaptive strategy is used to update the basis elements when needed. Alternatively, the Dynamically Orthogonal (DO) decomposition [76], where both the basis elements and expansion coefficients vary over time in a Karhunen-Loève form, offers a good perspective.

CRediT authorship contribution statement

Rem-Sophia Mouradi: Conceptualization, Formal analysis, Investigation, Methodology, Validation, Visualization, Writing – original draft. **Cédric Goeury:** Conceptualization, Formal analysis, Supervision, Writing – review & editing. **Olivier Thual:** Conceptualization, Project administration, Supervision, Writing – review & editing. **Fabrice Zaoui:** Conceptualization, Visualization, Writing – review & editing. **Pablo Tassi:** Conceptualization, 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.

Acknowledgements

This work was funded by the French National Association of Research and Technology (ANRT) and EDF R&D with the Industrial Conventions for Training through REsearch (CIFRE grant agreement 2017/1452). The authors acknowledge their support, and are grateful for data collection and feedback from EDF operators. In particular, we would like to thank D. Rougé for providing the data-set used in this study and for his continuous availability. We also would like to thank Pr. L. Terray (CERFACS) and Dr. M. Rochoux (CERFACS) for constructive discussions on POD and PCE respectively, and Pr. B. Sudret (ETH Zurich) for providing key literature elements on the treatment of ultrahigh dimensional problems and functional inputs using PCE. The authors also gratefully acknowledge the OpenTURNS open source community (An Open source initiative for the Treatment of Uncertainties, Risks'N Statistics). Finally, we would like to thank the anonymous reviewers, whose comments and suggestions helped improve the manuscript.

Appendix A. Complementary results on the parametric toy problem

In addition to the content of Section 3, supplementary materials are herein given for the parametric toy problem. Firstly, *Garson Weights* (GW) and *Generalized Garson Weights* (GGW), respectively presented in Sections 2.2 and 2.3, were calculated for fitted PCE models of the first four POD modes, in the amplitude learning case, as in Table 1.

Secondly, the POD-PCE strategy has also been deployed to learn the time lag between $f(x, y, t)$ and $f(0, 0, t)$ relative to the period T, at each location (x, y). POD was applied to the corresponding snapshot matrix. While 98% of the variance is already represented by Mode 1, a total of 5 modes is needed to approach the 100%. This increase is slow compared to the EVR of the amplitude. As a result, the POD-PCE performances, evaluated at each step of the algorithm, are different, as can be seen in Fig. 28.

The reduction error equals 10% at rank 1 (compared to 3.8% for the amplitude). It decreases following three slopes, the first one being from 10% at rank 1 to 5% at rank 3. The learning error is here much higher, almost equal to 1% for a 1-Mode approximation, and goes up to 4% for 3-Mode approximation, where it keeps on increasing. The modes coefficients seem more difficult to learn for the phase. Consequently, the prediction error decreases from 10% at rank 1 to 6% rank 3, where it stabilizes. Indeed, even though adding more POD patterns is interesting, learning them with PCE becomes more and more difficult as the represented variance decreases. The gain in accuracy with POD modes is therefore compensated with the loss of precision in PCE learning. Hence, a 3-Modes POD-PCE model was selected for prediction. Examples of phase

Table 1
Polynomial terms of PCE models calibrated on the aquifer case, for the 4 first modes of the amplitude ordered by their influence, using the GGWs in Equation (20). Also shown are the GWs calculated as in Equation (7). The contributions are shown up to a total of 99%.

Polynomial term	GGW	Total	Mode	GW
$\zeta_{\boldsymbol{\alpha}=(1)}(A)$	0.8016	0.80	1	0.84675
$\zeta_{\boldsymbol{\alpha}=(1)}(D)$	0.10138	0.90	1	0.10709
$\zeta_{\boldsymbol{\alpha}=(1)}(D)$	0.03199	0.93	2	0.71521
$\zeta_{\boldsymbol{\alpha}=(1,1)}(A, D)$	0.02115	0.96	1	0.02234
$\zeta_{\boldsymbol{\alpha}=(2)}(D)$	0.0145	0.97	1	0.01532
$\zeta_{\boldsymbol{\alpha}=(1,1)}(A, D)$	0.00667	0.98	2	0.14921
$\zeta_{\boldsymbol{\alpha}=(2)}(D)$	0.00375	0.98	2	0.08373
$\zeta_{\boldsymbol{\alpha}=(1,2)}(A, D)$	0.00316	0.98	1	0.00334
$\zeta_{\boldsymbol{\alpha}=(2)}(D)$	0.00238	0.99	3	0.47515
$\zeta_{\boldsymbol{\alpha}=(1)}(\kappa_{er})$	0.002	0.99	1	0.00211
$\zeta_{\boldsymbol{\alpha}=(1)}(\kappa_{er})$	0.00165	0.99	4	0.46206
$\zeta_{\boldsymbol{\alpha}=(3)}(D)$	0.00118	0.99	1	0.00125
$\zeta_{\boldsymbol{\alpha}=(1)}(\kappa_{er})$	0.00094	0.99	3	0.18655

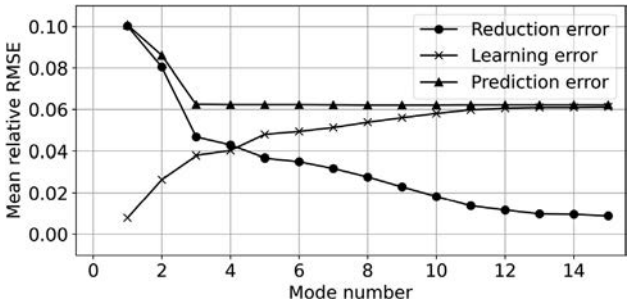


Fig. 28. Mean relative RMSE generated at different steps of the POD-PCE ML applied to the time-lag case, with different approximation ranks.

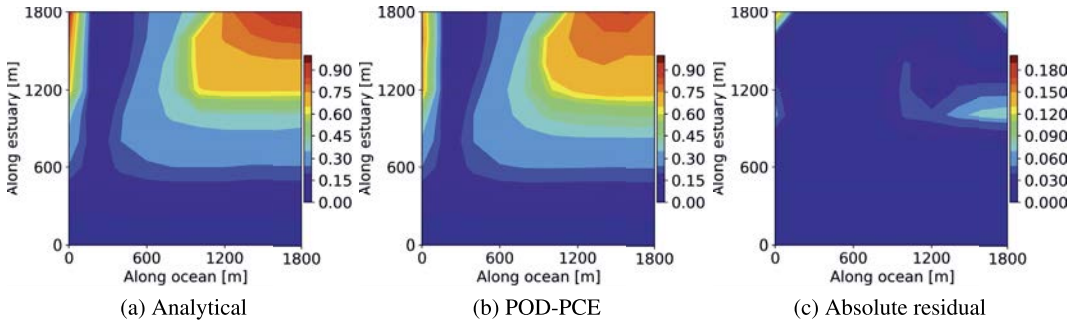


Fig. 29. Analytical solution vs. POD-PCE prediction of the time-lag, relative to the tidal period T in the aquifer, and resulting absolute residual.

prediction are shown in Fig. 29. The model gives a good mapping of the two-dimensional time lag distribution along the estuary and through the aquifer. However, the residuals are more important compared to the amplitude prediction. For example, an absolute residual of 0.1 T time-lag is noticed in the middle of the aquifer in Fig. 29-c, where the analytical time lag (Fig. 29-a) is around 0.4 T, representing a local error of 25%. The global performance of the model remains however satisfying.

The physical analysis of the latter are therefore performed using the GW and GGW indicators, reported in Table 2. First, the most important polynomial pattern for the coupled POD-PCE model is the diffusivity D at degree 1 (GGW 47%), whereas it was the tidal amplitude A at degree 1 for the amplitude distribution (GGW 80%). In particular, it barely represents half of the dynamics. It is completed by higher degree monomials of the same parameter D up to 79%. The phase representation exhibits more non-linearities than the amplitude. The contribution of D is followed by an interaction between A, D and the wave number in the estuary κ_{ei}. As a reminder, the latter did not appear as an influencing parameter for the amplitude distribution. Globally, the phase problem involves higher polynomial degrees, and higher orders of interaction.

Table 2
Polynomial terms of PCE models calibrated on the aquifer case, for the 3 first modes of the time-lag ordered by their influence, using the GGWs in Equation (20). Also shown are the GWs calculated as in Equation (7).

Polynomial term	GGW	Total	Mode	GW
$\zeta_{\boldsymbol{\alpha}=(1)}(D)$	0.4684	0.47	1	0.54334
$\zeta_{\boldsymbol{\alpha}=(3)}(D)$	0.14637	0.61	1	0.16979
$\zeta_{\boldsymbol{\alpha}=(4)}(D)$	0.0912	0.71	1	0.10579
$\zeta_{\boldsymbol{\alpha}=(2)}(D)$	0.08382	0.79	1	0.09723
$\zeta_{\boldsymbol{\alpha}=(2,1,1)}(A,\kappa_{ei},D)$	0.03942	0.83	1	0.04572
$\zeta_{\boldsymbol{\alpha}=(1)}(D)$	0.02889	0.86	2	0.35127
$\zeta_{\boldsymbol{\alpha}=(2)}(D)$	0.02066	0.88	3	0.37113
$\zeta_{\boldsymbol{\alpha}=(1)}(D)$	0.01986	0.90	3	0.35668
$\zeta_{\boldsymbol{\alpha}=(1,1)}(\kappa_{er},D)$	0.01594	0.91	1	0.01849
$\zeta_{\boldsymbol{\alpha}=(3)}(D)$	0.01515	0.93	3	0.27218
$\zeta_{\boldsymbol{\alpha}=(2)}(D)$	0.01389	0.94	2	0.16892
$\zeta_{\boldsymbol{\alpha}=(1,3)}(\kappa_{er},D)$	0.01288	0.96	1	0.01494
$\zeta_{\boldsymbol{\alpha}=(5)}(D)$	0.00907	0.97	2	0.11024
$\zeta_{\boldsymbol{\alpha}=(2,2)}(A,D)$	0.00658	0.97	2	0.07994
$\zeta_{\boldsymbol{\alpha}=(1,4)}(\kappa_{ei},D)$	0.0059	0.98	2	0.07178
$\zeta_{\boldsymbol{\alpha}=(2,1,1)}(A,\kappa_{ei},D)$	0.00531	0.98	2	0.06453
$\zeta_{\boldsymbol{\alpha}=(1,2)}(\kappa_{er},D)$	0.00405	0.99	1	0.0047

Appendix B. Confronting POD-PCE to NN

As an additional proof for the POD-PCE Machine Learning capacity, multiple NN configurations are tested on the measurement-based problem for confrontation. The latter is a small-data problem, considered as the most challenging case in the presented work. The python library Scikit-learn [77] (https://scikit-learn.org) was used for fitting.

A first NN set-up, aiming at learning the bathymetry fields $[z(x_i,t_j)]_{i,j} \in \mathbb{R}^{m\times n}$ directly from their previous values $[z(x_i,t_{j-1})]_{i,j} \in \mathbb{R}^{m\times n}$ and a set of parameters $\boldsymbol{\Theta}$, was attempted. A simple configuration was tested, where $z(x_i,t_j)$ is learned for each x_i independently, from its own previous value $z(x_i,t_{j-1})$ and the seven physical parameters Δt, $TLmean$, WvH, $Wvper$, $Wvdir$, $Wv2m$ and $Wv2m\%$, as done in the most optimal POD-PCE configuration ($\mathcal{H}_i^P$ described in Subsection 4.2.2). Therefore, m independent learnings are performed (points number), each characterized with an input dimension of $V=8$, and an output dimension of $o=1$.

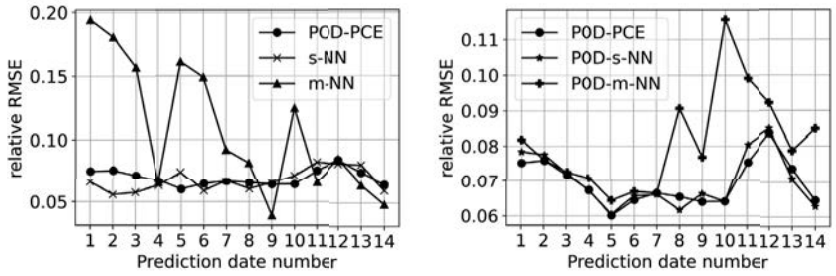
Two learning strategies are adopted. The first one consists in a single-layer NN, where only the *Activation Function* (AF) and the number of *neurons*, denoted l, are varied. The considered AFs are the ones available in Scikit-learn (identity, tanh, logistic and ReLu) [77], and allow to be in the theoretical conditions for the "shallow and wide" *Universal Approximation Theorem* [12]. The second alternative consists in a multi-layer NN using the ReLu AF, where the number of neurons l is fix, and the number of layers, denoted L varies. This allows to be in the framework of the "deep and narrow" version of the theorem [13].

For the single-layer NN, the maximal number of neurons is constrained to $l=5$. Indeed, the input-to-hidden connection matrix is of size $V\times l$, and the hidden-to-output matrix is of size $l\times o$. In this case, with $l=5$, a maximum number of 45 matrix coefficients should be estimated from the training sample of size 50. An additional neuron would result with an ill-posed problem. For the multi-layer ReLu NN, the number of neurons is set to $l=2$ and the maximum number of layers to $L=9$ (number of coefficients to estimate is $V\times l+(L-1)\times l^2+l\times o=50$). Using both configurations, the optimal choices (AF, neurons, layers) are selected for each coordinate x_i, based on the relative empirical error calculated on the test set, as in Equation (16). The RMSE for each prediction date are then calculated with the whole field $z(.,t_j)$ (NN prediction vs. reality), and confronted to POD-PCE in Fig. 30-a. The single-layer NN is denoted s-NN and the multi-layer ReLu NN is denoted m-NN.

To account for spatial correlations, a supplementary set-up was tested, where POD is performed before NN. Similarly to the POD-PCE learning set-up, NN is used to learn the first 11 POD coefficients, corresponding to the optimal POD-PCE learning in Section 4.2.2, and a POD-NN coupling is performed. The learning configurations mentioned above (single-layer, and multi-layer ReLu) are tested, and the algorithmic choices corresponding to the minimal relative empirical error are selected for each POD mode independently. The RMSE results are shown in Fig. 30-b, where POD-s-NN and POD-m-NN denote the coupling of POD with s-NN and m-NN respectively.

A last test is conducted, where an L2-penalty is used to fit sparse POD-NN. This is performed in scikit-learn [77] by adding a constraint to the learning minimization problem, consisting in a regularization term, controlled with an additional hyper-parameter. The values of hyper-parameters that were previously constrained can here be increased: the maximal number of neurons is set to $l=50$ for POD-s-NN, while the maximal number of layers is set to $L=10$ for POD-m-NN with a number of neurons fixed to $l=5$. The L2-penalty coefficient is varied from 10^{-4} (low sparsity) to 10^4 (high sparsity). A comparison of all algorithms in terms of average RMSE and fit-time can be found in Table 3.

Firstly, it can be noticed in Fig. 3 that the worst learning is performed with m-NN (average RMSE of 10.8% in Table 3). It might be due to the fact that available data are not sufficient for a deep network fitting. This is followed in terms of worst



(a) Comparison to NN (b) Comparison to POD-NN

Fig. 30. Comparison of relative prediction RMSE between the POD-PCE algorithm and different NN set-ups.

Table 3
Summary the performances for all tested learning algorithms.

Algorithm	Sparsity	Average RMSE	Fit-time
POD-PCE	LARS [25]	6.9%	11 s
s-NN	None	6.7%	1 h 47 m 14 s
m-NN	None	10.8%	20 m 16 s
POD-s-NN	None	7%	25 s
POD-m-NN	None	8.1%	10 s
POD-s-NN	L2 penalty [77]	6.9%	11 m 16 s
POD-m-NN	L2 penalty [77]	6.9%	2 m 48 s

performance by POD-m-NN (average RMSE of 8.1%), with the same interpretation. The other learning choices (POD-PCE, s-NN and POD-s-NN) have global similar behaviors. Among the last three, it is noted that s-NN performs the best for the three first dates, while it performs the worst for dates 5, 10, and 11 (Fig. 3). It scores the lowest average RMSE of 6.7%, but also by far the worst fit-time (Table 3). The performances of POD-PCE and POD-s-NN are very close, their average RMSE are 6.9% and 7%, but POD-PCE is twice faster. Sparsity added to POD-s-NN and POD-m-NN helps reducing the errors by 0.1 and 1.2% respectively. The resulting RMSE are equivalent to POD-PCE using LARS, which takes much less fit-time.

The POD-PCE coupling methodology offers an interesting alternative to NN in terms of accuracy and fit-time balance. It competes with POD-s-NN which is slightly less accurate, but POD-PCE is here twice-faster. Additionally, the most optimal POD-s-NN is composed of different AFs for the different modes (combinations of logistic and ReLu), which makes the interpretation difficult compared to polynomial patterns, and results with a superiority of POD-PCE for physical analysis. However, these conclusions should be interpreted in light of the learning choices, which can be improved. For example, a combination of the best single-layer networks and best multi-layer ReLu networks can be attempted to optimize the previous set-ups. This can even be further improved by choosing PCE or NN when appropriate. Lastly, as was the case with PCE, limitations to the previous learnings can be noted, among which the physical parameters selection and time-lag choice for the previous field value.

References

[1] Y. LeCun, Y. Bengio, G. Hinton, Deep learning, Nature 521 (2015) 436–444.
[2] S. Shalev-Shwartz, S. Ben-David, Understanding Machine Learning: From Theory to Algorithms, Cambridge University Press, 2014.
[3] S.L. Brunton, B.R. Noack, P. Koumoutsakos, Machine learning for fluid mechanics, Annu. Rev. Fluid Mech. 52 (2020) 477–508.
[4] M.S. Parsons, Interpretation of machine-learning-based disruption models for plasma control, Plasma Phys. Control. Fusion 59 (2017) 085001.
[5] K. Mills, M. Spanner, I. Tamblyn, Deep learning and the Schrödinger equation, Phys. Rev. A 96 (2017) 042113.
[6] A. Karpatne, I. Ebert-Uphoff, S. Ravela, H.A. Babaie, V. Kumar, Machine learning for the geosciences: challenges and opportunities, IEEE Trans. Knowl. Data Eng. 31 (2019) 1544–1554.
[7] J. Schmidhuber, Deep learning in neural networks: an overview, Neural Netw. 61 (2015) 85–117.
[8] S. Sengupta, S. Basak, P. Saikia, S. Paul, V. Tsalavoutis, F. Atiah, V. Ravi, A. Peters, A review of deep learning with special emphasis on architectures, applications and recent trends, arXiv:1905.13294, 2019.
[9] O.I. Abiodun, A. Jantan, A.E. Omolara, K.V. Dada, N.A. Mohamed, H. Arshad, State-of-the-art in artificial neural network applications: a survey, Heliyon 4 (2018).
[10] R. Iten, T. Metger, H. Wilming, L. del Rio, R. Renner, Discovering physical concepts with neural networks, Phys. Rev. Lett. 124 (2020).
[11] A. Laudani, G.M. Lozito, F.R. Fulginei, A. Salvini, On training efficiency and computational costs of a feed forward neural network: a review, Comput. Intell. Neurosci. (2015).
[12] K. Hornik, Approximation capabilities of multilayer feedforward networks, Neural Netw. 4 (1991) 251–257.
[13] B. Hanin, Universal function approximation by deep neural nets with bounded width and ReLU activations, Mathematics 7 (2019).
[14] L. Cordier, M. Bergmann, Proper orthogonal decomposition: an overview, in: Lecture Series 2002-04, 2003-03 and 2008-01 on Post-Processing of Experimental and Numerical Data, Von Karman Institute for Fluid Dynamics VKI, 2008, 46 pages.
[15] O.P. Le Maitre, O.M. Knio, H.N. Najm, R.G. Ghanem, A stochastic projection method for fluid flow: I. Basic formulation, J. Comput. Phys. 173 (2001) 481–511.
[16] O.P. Le Maitre, M.T. Reagan, H.N. Najm, R.G. Ghanem, O.M. Knio, A stochastic projection method for fluid flow: II. Random process, J. Comput. Phys. 181 (2002) 9–44.

[17] J.L. Lumley, The structure of inhomogeneous turbulent flows, in: Atmospheric Turbulence and Radio Wave Propagation, 1967.
[18] K. Taira, S.L. Brunton, S.T.M. Dawson, C.W. Rowley, T. Colonius, B.J. McKeon, O.T. Schmidt, S. Gordeyev, V. Theofilis, L.S. Ukeiley, Modal analysis of fluid flows: an overview, AIAA J. 55 (2017) 4013–4041.
[19] D. Xiu, G.E. Karniadakis, Modeling uncertainty in flow simulations via generalized polynomial chaos, J. Comput. Phys. 187 (2003) 137–167.
[20] B. Sudret, Polynomial chaos expansions and stochastic finite element methods, in: Kok-Kwang Phoon, Jianye Ching (Eds.), Risk and Reliability in Geotechnical Engineering, CRC Press, 2014, pp. 265–300.
[21] A. Tarakanov, A.H. Elsheikh, Regression-based sparse polynomial chaos for uncertainty quantification of subsurface flow models, J. Comput. Phys. 399 (2019) 108909.
[22] B.A. Jones, A. Doostan, Satellite collision probability estimation using polynomial chaos expansions, Adv. Space Res. 52 (2013) 1860–1875.
[23] B. Sudret, Global sensitivity analysis using polynomial chaos expansions, Reliab. Eng. Syst. Saf. 93 (2008) 964–979.
[24] O. Garcia-Cabrejo, A. Valocchi, Global sensitivity analysis for multivariate output using polynomial chaos expansion, Reliab. Eng. Syst. Saf. 126 (2014) 25–36.
[25] G. Blatman, B. Sudret, Adaptive sparse polynomial chaos expansion based on least angle regression, J. Comput. Phys. 230 (2011) 2345–2367.
[26] C. Soize, R. Ghanem, Physical systems with random uncertainties: chaos representations with arbitrary probability measure, SIAM J. Sci. Comput. 26 (2) (2004) 395–410.
[27] M. Muller, On the POD method: an abstract investigation with applications to reduced-order modeling and suboptimal control, Ph.D. thesis, 2008.
[28] G. Blatman, Adaptive sparse polynomial chaos expansions for uncertainty propagation and sensitivity analysis, Ph.D. thesis, 2009.
[29] M. Larson, M. Capobianco, M. Jansen, G. Różyński, H. Southgate, M. Stive, K. Wijnberg, S. Hulscher, Analysis and modeling of field data on coastal morphological evolution over yearly and decadal time scales. part 1: background and linear techniques, J. Coast. Res. 19 (2003).
[30] Q. Wang, J.S. Hesthaven, D. Ray, Non-intrusive reduced order modeling of unsteady flows using artificial neural networks with application to a combustion problem, J. Comput. Phys. 384 (2019) 289–307.
[31] J.B. Nagel, J. Rieckermann, B. Sudret, Principal component analysis and sparse polynomial chaos expansions for global sensitivity analysis and model calibration: application to urban drainage simulation, Reliab. Eng. Syst. Saf. 195 (2020) 106737.
[32] C. Lataniotis, S. Marelli, B. Sudret, Extending classical surrogate modelling to ultrahigh dimensional problems through supervised dimensionality reduction: a data-driven approach, arXiv:1812.06309, 2018.
[33] E. Torre, S. Marelli, P. Embrechts, B. Sudret, Data-driven polynomial chaos expansion for machine learning regression, J. Comput. Phys. 388 (2019) 601–623.
[34] L. Li, D. Barry, C. Cunningham, F. Stagnitti, J.-Y. Parlange, A two-dimensional analytical solution of groundwater responses to tidal loading in an estuary and ocean, Adv. Water Resour. 23 (2000) 825–833.
[35] M. Gevrey, I. Dimopoulos, S. Lek, Review and comparison of methods to study the contribution of variables in artificial neural network models, Ecol. Model. 160 (2003) 249–264.
[36] M. Couplet, Reduced-order POD-Galerkin modelling for the control of unsteady flows, Ph.d. thesis, Université Paris-Nord - Paris XIII, 2005.
[37] L. Sirovich, Turbulence and the dynamics of coherent structures: I, II and III, Q. Appl. Math. 45 (1987) 561.
[38] N. Wiener, The homogeneous chaos, Am. J. Math. 60 (1938) 897–936.
[39] D. Xiu, G.E. Karniadakis, The Wiener–Askey polynomial chaos for stochastic differential equations, SIAM J. Sci. Comput. 24 (2002) 619–644.
[40] T. Hastie, R. Tibshirani, J. Friedman, The Elements of Statistical Learning: Data Mining, Inference, and Prediction, second edition, Springer Series in Statistics, 2009.
[41] J.A. Witteveen, H. Bijl, Modeling arbitrary uncertainties using Gram-Schmidt polynomial chaos, in: 44th AIAA Aerospace Sciences Meeting and Exhibit, 2006.
[42] X. Wan, G.E. Karniadakis, Multi-element generalized polynomial chaos for arbitrary probability measures, SIAM J. Sci. Comput. 28 (2006) 901–928.
[43] M. Tsang, D. Cheng, Y. Liu, Detecting statistical interactions from neural network weights, arXiv:1705.04977, 2017.
[44] T. Taddei, A registration method for model order reduction: data compression and geometry reduction, SIAM J. Sci. Comput. 42 (2020) A997–A1027.
[45] X. Wan, G. Karniadakis, An adaptive multi-element generalized polynomial chaos method for stochastic differential equations, J. Comput. Phys. (2006).
[46] I. Guyon, A. Elisseeff, An introduction to variable and feature selection, J. Mach. Learn. Res. 3 (2003) 1157–1182.
[47] I. Jolliffe, Principal Component Analysis, Springer, Berlin, Heidelberg, 2011, pp. 1094–1096.
[48] M. Lamboni, H. Monod, D. Makowski, Multivariate sensitivity analysis to measure global contribution of input factors in dynamic models, Reliab. Eng. Syst. Saf. 96 (2011) 450–459.
[49] B.J. Wagner, Simultaneous parameter estimation and contaminant source characterization for coupled groundwater flow and contaminant transport modelling, J. Hydrol. 135 (1992) 275–303.
[50] T. Sruthi, K. Ranjith, V. Chandra, Control of Sediment Entry into an Intake Canal by Using Submerged Vanes, AIP Conference Proceedings, vol. 1875, AIP Publishing LLC, 2017, p. 030007.
[51] R.G. Dean, R.A. Dalrymple, Coastal Processes with Engineering Applications, Cambridge University Press, 2004.
[52] S. Costa, P. Letortu, B. Laignel, The hydro-sedimentary system of the Upper-Normandy coast, in: Synthesis, Sediment Fluxes in Coastal Areas, 2015, pp. 121–147.
[53] S. Le Bot, R. Lafite, M. Fournier, A. Baltzer, M. Desprez, Morphological and sedimentary impacts and recovery on a mixed sandy to pebbly seabed exposed to marine aggregate extraction (eastern English channel, France), Estuar. Coast. Shelf Sci. 89 (2010) 221–233.
[54] C. Michel, S. Le Bot, F. Druine, S. Costa, F. Levoy, C. Dubrulle-Brunaud, R. Lafite, Stages of sedimentary infilling in a hypertidal bay using a combination of sedimentological, morphological and dynamic criteria (Bay of Somme, France), J. Maps 13 (2017) 858–865.
[55] A. Guillaume, VAG-Modele de prevision de l'etat de la mer en eau profonde, Dir. de la Meteorologie Nationale, 1987.
[56] REFMAR, Réseaux de référence des observations MARégraphiques, https://doi.org/10.17183/REFMAR, 2020.
[57] M. Janocko, M. Cartigny, W. Nemec, E. Hansen, Turbidity current hydraulics and sediment deposition in erodible sinuous channels: laboratory experiments and numerical simulations, Mar. Pet. Geol. 41 (2013) 222–249.
[58] B. Efron, R. Tibshirani, Bootstrap methods for standard errors, confidence intervals, and other measures of statistical accuracy, Stat. Sci. (1986) 54–75.
[59] M. Ghil, P. Yiou, S. Hallegatte, B.D. Malamud, P. Naveau, A. Soloviev, P. Friederichs, V. Keilis-Borok, D. Kondrashov, V. Kossobokov, O. Mestre, C. Nicolis, H.W. Rust, P. Shebalin, M. Vrac, A. Witt, I. Zaliapin, Extreme events: dynamics, statistics and prediction, Nonlinear Process. Geophys. 18 (2011) 295.
[60] G. Cruciani, M. Baroni, S. Clementi, G. Costantino, D. Riganelli, B. Skagerberg, Predictive ability of regression models. Part I: Standard deviation of prediction errors (sdep), J. Chemom. 6 (1992) 335–346.
[61] L.C. van Rijn, Unified view of sediment transport by currents and waves. I: Initiation of motion, bed roughness, and bed-load transport, J. Hydraul. Eng. 133 (2007) 649–667.
[62] G. Kerschen, J. Golinval, Physical interpretation of the proper orthogonal modes using the singular value decomposition, J. Sound Vib. 249 (2002) 849–865.
[63] S. Paul, M.K. Verma, Proper orthogonal decomposition vs. Fourier analysis for extraction of large-scale structures of thermal convection, in: Advances in Computation, Modeling and Control of Transitional and Turbulent Flows, 2017, pp. 433–441.
[64] A. Hekmati, D. Ricot, P. Druault, About the convergence of pod and epod modes computed from cfd simulation, Comput. Fluids 50 (2011) 60–71.

[65] P.J. Schmid, Dynamic mode decomposition of numerical and experimental data, J. Fluid Mech. 656 (2010) 5–28.
[66] S. Mika, B. Schölkopf, A.J. Smola, K.-R. Müller, M. Scholz, G. Rätsch, Kernel pca and de-noising in feature spaces, Adv. Neural Inf. Process. Syst. (1999) 536–542.
[67] I.M. Johnstone, A.Y. Lu, On consistency and sparsity for principal components analysis in high dimensions, J. Am. Stat. Assoc. 104 (2009) 682–693.
[68] P. Saini, C.M. Arndt, A.M. Steinberg, Development and evaluation of gappy-pod as a data reconstruction technique for noisy piv measurements in gas turbine combustors, Exp. Fluids 57 (2016) 122.
[69] C.J. Legleiter, P.C. Kyriakidis, R.R. McDonald, J.M. Nelson, Effects of uncertain topographic input data on two-dimensional flow modeling in a gravel-bed river, Water Resour. Res. 47 (2011).
[70] R. Noori, A. Karbassi, A. Moghaddamnia, D. Han, M. Zokaei-Ashtiani, A. Farokhnia, M.G. Gousheh, Assessment of input variables determination on the svm model performance using pca, gamma test, and forward selection techniques for monthly stream flow prediction, J. Hydrol. 401 (2011) 177–189.
[71] S. Du, G. Song, H. Hong, Collective causal inference with lag estimation, Neurocomputing 323 (2019) 299–310.
[72] W.J. Murdoch, C. Singh, K. Kumbier, R. Abbasi-Asl, B. Yu, Interpretable machine learning: definitions, methods, and applications, arXiv:1901.04592, 2019.
[73] J. Senent-Aparicio, P. Jimeno-Sáez, A. Bueno-Crespo, J. Pérez-Sánchez, D. Pulido-Velázquez, Coupling machine-learning techniques with swat model for instantaneous peak flow prediction, Biosyst. Eng. 177 (2019) 67–77.
[74] A. Mosavi, S. Shamshirband, E. Salwana, K.-w. Chau, J.H. Tah, Prediction of multi-inputs bubble column reactor using a novel hybrid model of computational fluid dynamics and machine learning, Eng. Appl. Comput. Fluid Mech. 13 (2019) 482–492.
[75] M. Gerritsma, J.-B. Van der Steen, P. Vos, G. Karniadakis, Time-dependent generalized polynomial chaos, J. Comput. Phys. 229 (2010) 8333–8363.
[76] E. Musharbash, F. Nobile, T. Zhou, Error analysis of the dynamically orthogonal approximation of time dependent random pdes, SIAM J. Sci. Comput. 37 (2015) A776–A810.
[77] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, E. Duchesnay, Scikit-learn: machine learning in Python, J. Mach. Learn. Res. 12 (2011) 2825–2830.