Some applications of compressed sensing in computational mechanics: model order reduction, manifold learning, data-driven applications and nonlinear dimensionality reduction

R. Ibañez[1] · E. Abisset-Chavanne[2] · E. Cueto[3] · A. Ammar[4] · J. -L. Duval[5] · F. Chinesta[6]

Abstract

Compressed sensing is a signal compression technique with very remarkable properties. Among them, maybe the most salient one is its ability of overcoming the Shannon–Nyquist sampling theorem. In other words, it is able to reconstruct a signal at less than $2Q$ samplings per second, where Q stands for the highest frequency content of the signal. This property has, however, important applications in the field of computational mechanics, as we analyze in this paper. We consider a wide variety of applications, such as model order reduction, manifold learning, data-driven applications and nonlinear dimensionality reduction. Examples are provided for all of them that show the potentialities of compressed sensing in terms of CPU savings in the field of computational mechanics.

Keywords Compressed sensing · Model order reduction · Manifold learning · Nonlinear dimensionality reduction

1 Introduction

Model Order Reduction (MOR) is acquiring an utmost importance for simulation-based engineering. These techniques allow solving efficiently complex mathematical models, thanks to the use of adapted approximation bases to describe their solutions. Among the numerous existing MOR techniques, Proper Orthogonal Decomposition (POD), Proper Generalized Decomposition (PGD) and Reduced Basis (RB) are largely considered in a variety of applications [13].

Proper Orthogonal Decomposition is a general technique to extract the most significant characteristics of a system's behavior and to represent them in a set of optimal "POD basis vectors". These basis vectors provide an efficient (typically, low-dimensional) representation of the essential features of the system behavior, which has proven useful in a variety of ways. The most common use is to project the solution of the governing equations onto the reduced-order subspace defined by these POD basis vectors. This yields an explicit POD reduced model that can be solved instead of the original system. The POD basis can also provide a low-dimensional description on which to perform parametric interpolation, infill missing or "gappy" data, and perform model adaptation. There is an extensive literature and POD has seen broad application across fields. Some review of POD and its applications can be found in [4,43].

✉ E. Cueto
ecueto@unizar.es

R. Ibañez
ruben.ibanez-pinillo@eleves.ec-nantes.fr

E. Abisset-Chavanne
Emmanuelle.Abisset-Chavanne@ensam.eu

A. Ammar
Amine.AMMAR@ensam.eu

J. -L. Duval
jean-louis.duval@esi-group.com

F. Chinesta
francisco.chinesta@ensam.eu

[1] ESI Group Chair, Ecole Centrale de Nantes, Nantes, France

[2] ESI Chair and I2M Lab, Arts et Métiers ParisTech Centre de Bordeaux-Talence, Esplanade des Arts et Métiers, 33400 Talence, France

[3] Aragon Institute of Engineering Research, Universidad de Zaragoza, Zaragoza, Spain

[4] LAMPA, ENSAM Angers, 2 Boulevard du Ronceray, BP 93525, 49035 Angers Cedex 01, France

[5] ESI Group, 3bis rue Saarinen, 94528 Rungis CEDEX, France

[6] ESI Chair at PIMM Lab, ENSAM ParisTech, Paris, France

Reduced Basis techniques employ an approximation basis constructed by combining a greedy algorithm and *a posteriori* error indicators. As for the POD, the Reduced Basis method requires some amount offline work, but then the reduced basis model can be used online for solving different models with control of the solution accuracy, because the availability of error bounds. When the error is unacceptably high, the reduced basis can be enriched by invoking a greedy adaption strategy. Useful review works on the subject are [37,38,40].

Finally, there exist techniques based on the use of separated representations, at the heart of the so-called Proper Generalized Decomposition methods. Such separated representations are considered when solving at-hand partial differential equations by employing procedures based on the separation of variables. They were already considered in quantum chemistry to approximate multidimensional quantum wave-functions, e.g., Hartree–Fock and post-Hartree–Fock methods. In the 80s, Pierre Ladeveze proposed the use of space-time separated representations of transient solutions involved in strongly nonlinear models, defining a non-incremental integration procedure [27,28]. Later, separated representations were employed for solving multidimensional models suffering the so-called curse of dimensionality [1,2] as well as in the context of stochastic modeling [36]. Then, they were extended for separating space coordinates making possible the solution of models defined in degenerated domains, e.g. plate and shells [5] as well as for addressing parametric models where model parameters were considered as model extra-coordinates, making possible the offline calculation of the parametric solution that can be viewed as a metamodel or a computational vademecum, to be used online for real time simulation, optimization, inverse analysis and simulation-based control (see [15] for a recent review). Some recent reviews concerning the PGD can be found in [11,14].

These techniques improved traditional strategies based on DoE (Design of Experiments), that allowed defining metamodels, surrogate models or response surfaces. In these cases, experiments or expensive computational solutions are performed for a sampling of possible states of the system, from which a simplified model linking the inputs to the outputs of interest is elaborated. The main difficulties associated to this procedure concerns the best sampling strategy and the most adequate interpolation scheme for making prediction everywhere in the design space from the only knowledge of the few analyzed scenarios. Latin hypercube and Kriging are two usual responses to these questions. However, other questions remain, such as model verification (error estimation and bounds) as well as the definition of adaptive strategies able to reduce such error locally or globally. Even if there is a panoply of proposals and applied strategies, most of them are problem-dependent and fail to be robust and reliable. As just indicated, model order reduction established routes to achieve similar goals while circumventing the main issues just indicated, to finally define a "numerical or graphical handbook", constructed offline and efficiently used online for robust design purposes.

Even if, as just indicated, MOR technologies facilitate better approaches, their main difficulty is that they remain often too intrusive. Nowadays, the most recents works concerning MOR techniques focus on non-intrusive algorithms. However, from a pragmatic point of view, all these proposals remain less direct than usual DoE methodologies, the last simply consisting of evaluation the model at different points if the design space by using standard commercial solvers (adapted to the problem at hand) and then simply interpolating these solution to any other point. Despite the conceptual difficulties just referred, the procedure is very simple and attracted the favor of engineers, designers and practicians.

Thus, the big picture could be formulated as follows: could direct sampling lead to a robust and reliable parametric solution?

The answer to this question has been traditionally addressed in a variety of ways. First, response surface based methodologies (e.g. [8] and references therein) proposes a sort of adaptive procedure when the model is refined by zooming-in when solution approaches to the optimal solution with respect to a given couple model / optimization criterion. However, such a procure requires an amount of online computation because the fine representation is not *a priori* available and it must be constructed online during the optimization process.

Other possibility consists in reconstructing the unknown solution everywhere from the only knowledge of the calculated scenarios by making use of adequate interpolations: polynomials or POD-based modes (inspired from the gappy-POD formulations [17]). Reduced Basis perform this job offline: it extract a basis in which the model solution is projected. This projection is then solved online. However, one could imagine using directly the offline computed solution for interpolating it everywhere. The main advantage of this procedure lies in the fact that in the RB framework the sampling points (as indicated above) are determined from an adequate "a priori"—or more generally "a posteriori"—error indicator, defining a sort of greedy strategy that samples the space at (almost) optimal points. The main drawback is that very often the definition of those error indicators requires some deep knowledge of the considered model and it is not evident for many complex engineering problems. It is important to note that the fact that extracting the basis for projecting the problem solution and solving online the reduced problem offers higher precision that the option of directly interpolating the sampled solutions.

Close to the methodology just described, Borzacchiello et al. [6,7] proposed the use of hierarchical approximation bases, making possible that at each level of representation

only the contribution from the previous level to the present one must be calculated. This strategy allows defining simple error indicators and refine adaptively the parametric domain. Similar strategies, all them inspired of sparse-grids [10] methodologies can be combined with the use of wavelet representations in order to profit their inherent multi-resolution properties, that provide natural error indicators associated with the weights of the wavelet coefficients at each representation level [30]. This techniques allows even addressing multi-parametric models in a moderate number of dimensions.

More recently, the authors proposed a novel technique based on the use of sparse identification and Proper Generalized Decompositions [22]. In it, models established in up to ten dimensions are identified with a minimum of data thanks to the sparse structure of PGD and its ability to overcome the curse of dimensionality.

It is at this point important to note that interpolation is a tricky issue when the solution defines a slow manifold embedded in the whole space. In that case to define safe interpolation one must proceed by interpolating on the manifold. Within the MOR framework, Farhat was one of the first to claim this necessity [3]. In [3,19,21,31,34,35] manifold learning was used (using nonlinear dimensionality reduction strategies [29,32,39]) to extract latent parameters and the structure of the solution manifold in order to define accurate interpolations. It has also been employed for the construction of reduced models operating on the manifold, to define parametric solutions on it, or simply to define successfully data-driven computational mechanics applications. In these last cases, traditional constitutive equations were replaced by a manifold consisting of collected data.

In the present paper we address the problem of the reconstruction of a parametric solution from a coarse sampling, but from a different perspective. Compressive sensing provides a solid framework for performing random samplings. There is a vast literature on compressive sensing, extensively used in data and image analysis (see [25,26,33] and the references therein). It has recently attracted the interest of the modeling and simulation scientific communities to works like [9]. In what follows we first revisit the main concepts related to compressive sensing. Then in Sect. 3 we apply such ideas to random sampling of parametric models. Section 4 presents and discusses some other applications related to data-driven simulations and hyper-reduction techniques.

2 Overview of compressed sensing

Most of nonlinear dimensionality reduction techniques consider least-squares fitting of the data. However, compressed sensing is based in the use of the L^1 norm instead. As described in [26], there is a subtle link between sparsity and the use of the L^1 norm. When considering curve fitting, the use of standard L^2 norms magnifies the importance of outlying points because of the squared norm. The impact of these outlying points in the fitted curve can be significant.

In the same spirit, the solution of underdetermined algebraic systems is a tricky issue because they represent an infinite number of solutions. As illustrated in [26], the use of the pseudo-inverse produces a fully populated solution vector whereas when considering the "Matlab" backslash, the obtained solution contains many zero entries, so that it results to be sparse. When solving the problem with L^2 and L^1 optimizations (trying to obtain the minimum norm solution), the former becomes much less sparse than the last. In the case of overdetermined systems the same tendencies can be observed.

Thus, from a purely engineering viewpoint, L^1-norm can be associated to sparsity. For this reason the L^1 norm was considered as an appealing candidate for addressing signal reconstruction. It is able to overcome the Nyquist–Shannon sampling theory that states that for recovering a signal, one must sample at twice the rate of the highest frequency involved in the signal.

Let us consider a vector $\mathbf{f}$, in the usual space or time domains, and its counterpart in a domain in which it should accept a sparse representation, i.e., its vector counterpart $\mathbf{c}$ contains many zeros. These spaces are in general the ones related to frequency (Fourier or discrete cosines transforms) or the ones related to multi-resolution wavelets, among many other possible choices. We denote by $\mathbf{T}$ the matrix making possible this discrete transformation, i.e.,

$$\mathbf{Tc} = \mathbf{f}. \tag{1}$$

Since vector $\mathbf{c}$ is expected to have many zero entries (as soon as it corresponds, by assumption, to a space in which the signal becomes sparse), one could expect that its expression could be determined by employing only some rows of matrix $\mathbf{T}$ and vector $\mathbf{f}$. This implies solving the resulting underdetermined system making use of a L^1-norm based optimization.

The choice of such rows can be made in different ways. However, the most usual one consists of a random selection. From a matrix perspective, such extraction simply consists of defining a diagonal matrix, with unit entries at the rows we want to extract. If the set of rows to be extracted is denoted by $\mathcal{S}$, the extraction matrix $\mathbf{E}$ is defined from

$$\begin{cases} E_{ii} = 1 & \text{if } i \in \mathcal{S}, \\ E_{ij} = 0 & \text{otherwise.} \end{cases}$$

Rows containing only zeros are then eliminated from the matrix, thus generating a rectangular one, here denoted as $\mathbf{E}$.

The solution of problem (1) can thus be approximated by that of the underdetermined system

$$\mathbf{ETc} = \mathbf{Ef}, \tag{2}$$

by using a L^1-norm based optimization.

In sum, the two main ingredients sparse sampling are: (i) the use of an adequate space in which the solution of the problem at hand is expected to exhibit sparsity, and (ii) the solution of the underdetermined problem by using a L^1 norm.

Compressed sensing is at the origin os the so-called "single pixel camera". In it, instead of acquiring the global image information, i.e., a pixel vector $\mathbf{f}$, to be then compressed, only a few of its entries are acquired, namely $\mathbf{Ef}$. As soon as vector $\mathbf{c}$ is calculated by solving Eq. (2), the whole solution can be reconstructed from Eq. (1).

In the sequel, to solve the system of Eq. (2) we have preferred to employ the least absolute shrinkage and selection operator, LASSO, method [42]. Just as a recall, LASSO solves a minimization problem that involves an L2- minimization of the system of equations plus a penalty term involving a L1-norm of the unknown field,

$$\min_{\mathbf{c}}(||\mathbf{ETc} - \mathbf{Ef}||_{L2}^2 + \lambda ||\mathbf{c}||_{L1}).$$

This simple modification of the original compressed sensing procedure has revealed to avoid some of the numerical problems associated with the solution of an under-constrained system, providing a new coordinate system such that the solution is sparse. Its performance will be analyzed in Sect. 6 below.

3 Obtention of parametric models by compressed sensing

As discussed in the introduction, parametric solutions of the type $u(\mathbf{x}, t, \boldsymbol{\mu})$—where $\boldsymbol{\mu}$ represents the set of parameters in the problem—obtained by the (off-line) application of PGD techniques are extremely valuable for conducting (on-line) real-time simulations as well as optimization, inverse analysis, simulation-based control and uncertainty propagation under real-time constraints [12,16]. However, the PGD constructor is strongly invasive with respect to the use of commercial simulation codes. To circumvent or, at least, alleviate such a constraint, sampling of the parametric space and a subsequent interpolation of these samples could be an alternative route. However, by making it in the original space the sampling becomes sometimes too dense to capture the richness or as consequence of the Nyquist theorem. As just discussed in the previous section, compressed sensing bypasses such difficulties in many cases.

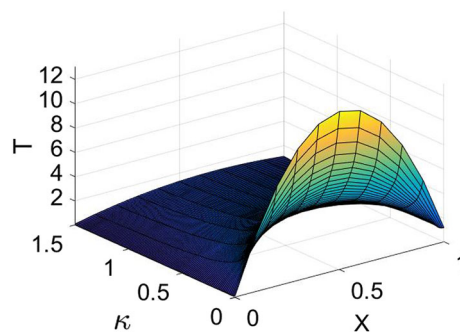In order to illustrate the proposed procedure, we consider the parametric heat equation

Fig. 1 Parametric solution $u(x, \kappa)$ of Eq. (3)

$$\kappa \frac{\partial^2 u}{\partial x^2} = s, \quad \text{in } \Omega_x = (0, L = 1), \tag{3}$$

with κ the thermal conductivity, $s = 1$ the source term (assumed constant in the space domain), and with boundary conditions $u(x = 0) = 0$ and $u(x = L) = 1$.

We are interested in solving this thermal model for any thermal conductivity $\kappa \in \Omega_\kappa = [0.1, 1.5]$. Solving it using the standard PGD approach (for an in-deep discussion of this problem the interested reader should consult [12,16]) we obtain the solution depicted in Fig. 1. It reveals that, by increasing the conductivity, the solution becomes flatter, since the generated heat can easily reach the domain boundaries $x = 0$ and $x = L$ leaving the domain. Lower conductivities imply higher temperatures because of the difficulty of evacuating the produced heat.

In order to show the potential of compressed sensing, we consider $K = 100$ coordinates along Ω_κ. From them, we randomly select $K_r = 10$ samples, defining the sampling set $\mathcal{S} = \{\kappa_1, \ldots, \kappa_{K_r}\}$. At these particular *locations*, Eq. (3) is solved by using standard finite differences or finite elements. These discrete solutions, consisting of vectors containing nodal temperatures for each choice of the thermal conductivity are denoted by $\mathbf{u}^k$, $k = 1, \ldots, K_r$.

Considering now different nodes in the spatial mesh associated to Ω_x, x_i, we define vectors $\mathbf{f}_i$ whose j-th entry reads

$$f_{ij} = u^{\kappa_j}(x_i).$$

Vector $\mathbf{f}_i$ contains $K - K_r$ unknown entries, those related to thermal conductivities not in the sampling set $\mathcal{S}$. However this does not constitute a problem, since the corresponding rows are not affected by the extraction matrix $\mathbf{E}$. Only the rows with known entries will be extracted.

At this point, by using a wavelet representation to define the transformation matrix $\mathbf{T}$—in particular, a biorthogonal 3.1 one—, the parametric model at node x_i will read
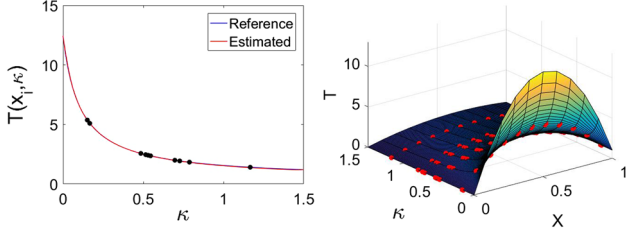
$$\mathbf{Tc}_i = \mathbf{f}_i, \tag{4}$$

Fig. 2 Reference versus compressed sensing based reconstruction of the nodal solution evolution with the model parameter. Left: Solution at a given nodal position. Black nodes represent the sampling points. These are also indicated on the right

or, by extracting the selected entries,

$$\mathbf{ETc}_i = \mathbf{Ef}_i. \tag{5}$$

Solving the underdetermined problem using a L^1-norm, we obtain $\mathbf{c}_i$, from which the nodal parametric equation can be reconstructed everywhere from

$$\mathbf{f}_i = \mathbf{Tc}_i. \tag{6}$$

Figure 2 compares the reference and reconstructed solutions at a particular node. We deliberately chose one located in the center of the domain in order to involve large gradients. In this figure the sampling nodes are also depicted.

We thus see how by sampling only the ten percent of all parametric nodal positions we obtain a remarkable accuracy in the reconstruction of the thermal field. Notably, this strategy allows the use of commercial software to obtain response surface-like solutions to parametric models and constitutes an alternative and valuable constructor of meta-models. It avoids the typical oscillations that polynomial approximations provoke when the sampling points are chosen randomly and do not correspond with the Gauss-Lobatto points. In that sense, compressed sensing produces solutions closer to the ones associated with the use of Kriging, that avoids large oscillations thanks to its statistical nature.

4 Manifold learning

Compressed sensing could also play a fundamental role in data-driven simulations. For the sake of simplicity, but without loss of generality, we consider mechanical tests conducted on a perfectly linear elastic material. Thus, for M randomly applied external loads, we assume ourselves able to collect M couples $(\boldsymbol{\sigma}_m, \boldsymbol{\varepsilon}_m)$, $m = 1, \ldots, M$. Every stress-strain couple could be thought of as a single point $\mathbf{P}_m$ in a phase space of dimension $D = 12$ (the six distinct components of the stress and strain tensors, respectively). In the sequel, Voigt notation will be considered, i.e., stress and strain tensors will be represented as vectors and consequently the fourth-order elastic tensor reduces to a 6×6 square matrix.

In the spirit of Locally Linear Embeddings [39], we assume that every point on the constitutive manifold can be linearly approximated, within a small enough neighborhood, as a function of the M available data. In other words,

$$\mathbf{P}_m = \sum_{i=1}^{M} W_{mi} \mathbf{P}_i,$$

with $W_{mi} = 0$ if $i \notin \mathcal{S}_m$ (the set containing the K-nearest neighbors of $\mathbf{P}_m$). By minimizing the functional

$$\mathcal{H}(\mathbf{C}) = \sum_{i \in \mathcal{S}_m} (\boldsymbol{\sigma}_i - \mathbf{C}\boldsymbol{\varepsilon}_i)^2.$$

we obtain the secant elastic behavior $\mathbf{C}(\mathbf{P}_m) \equiv \mathbf{C}_m$. The standard weak form of the equilibrium equation will be therefore

$$\int_{\Omega} \boldsymbol{\varepsilon}^*(\mathbf{x}) : \boldsymbol{\sigma}(\mathbf{x}) \, d\Omega = \int_{\Gamma_N} \mathbf{u}^*(\mathbf{x}) \cdot \mathbf{t}(\mathbf{x}) \, d\Gamma, \tag{7}$$

that can be rewritten as [24]

$$\int_{\Omega} \boldsymbol{\varepsilon}^*(\mathbf{x}) \cdot (\mathbf{C}(\mathbf{x})\boldsymbol{\varepsilon}(\mathbf{x})) \, d\Omega = \int_{\Gamma_N} \mathbf{u}^*(\mathbf{x}) \cdot \mathbf{t}(\mathbf{x}) \, d\Gamma.$$

This weak form allows us to solve the mechanical problem at one of the iterations of the nonlinear solver. Other discretization alternative strategies were discussed in [24].

However, as previously argued, prior to proceed with the calculations summarized above, one must accomplish the construction of the so-called constitutive manifold. In [24] the authors considered an inverse approach that assumed a tentative constitutive manifold. From it, a complex mechanical test was simulated and the strain prediction compared to data coming from an experimental field measurement. From the measured gap between predictions and experimental results, the constitutive manifold is updated providing a new strain prediction. The process continues until convergence is reached. In other words, until the fixed point of the data-driven inverse strategy is found. We proved in [24] the capability of such a method to identify the constitutive manifold associated to nonlinear elasticity. However, its generalization to more complex behaviors—like those involving internal variables—seems technically complex [20,23].

One possible route to explore consists in making use of well-experienced experimental methodologies. These were often developed for calibrating constitutive equations by testing coupons subjected to simple stress states. These tests are very well understood and offer valuable information for calibrating complex constitutive equations. However, by
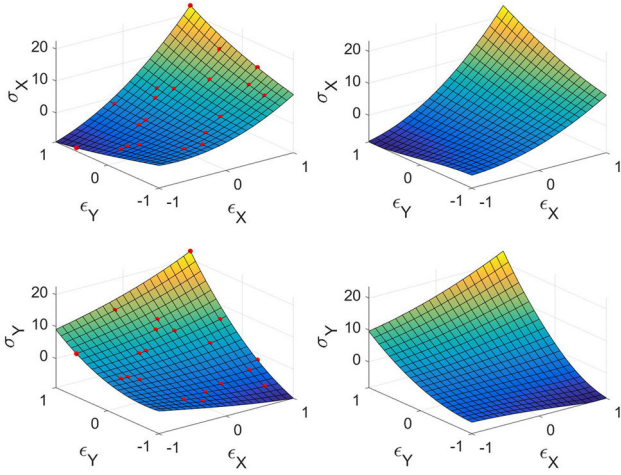
Fig. 3 Reference (left column) versus compressed sensing based reconstructed behavior manifold (right column). Red points indicate sampling locations. (Color figure online)

restricting to them, the constitutive manifold remains mostly unexplored, since too sparse information is accessible.

It is at this point that compressed sensing seems to offer a valuable opportunity. Indeed, if the constitutive manifold is viewed as a sort of image of the phase space of the material, a small quantity of data points could be enough to determine the whole manifold. A parallelism could be established with the so-called single pixel camera, following the rationale described in Sect. 2 [26].

To evaluate the performance of such a procedure, we consider a hypothetical nonlinear plane-stress elastic behavior, that in Voigt notation reads

$$\mathbf{C} = \frac{E}{1 - \nu^2} \begin{bmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & \frac{1-\nu}{2} \end{bmatrix}, \tag{8}$$

with the elastic coefficients given by

$$\begin{cases} E = E_0 + E_1 \text{Tr}(\boldsymbol{\varepsilon}) \\ \nu = \nu_0 + \nu_1 \text{Tr}(\boldsymbol{\varepsilon}) \end{cases},$$

with E_0, E_1, ν_0 and ν_1 positive constants, and where $\text{Tr}(\bullet)$ refers to the trace operator acting on tensor $\bullet$. In the numerical example discussed below the material coefficients were selected as $E_0 = 10$, $\nu_0 = 0.1$, $E_1 = 10$ and $\nu_1 = 0.1$.

We considered different strain couples $(\varepsilon_{xx}, \varepsilon_{yy})$ and determine from the constitutive Eq. (8) the associated stress couples $(\sigma_{xx}, \sigma_{yy})$. Figure 3 depicts the reference solutions and the considered points (in red) that served to reconstruct the approximated manifold from the compressed sensing rationale. We do not considered the off-diagonal components because they define a one-dimensional manifold that is quite simple to approximate, as proved in the previous section.

It can be noticed that, despite the small number of sampling points, the reconstructed constitutive manifold reproduces accurately the reference solution, thus constituting an excellent candidate to perform few iterations of the data-driven inverse strategy described in [24] to improve it. Obviously, an advantage of the methodology here described is the possibility to update the reconstructed solution as soon as new data-points are available from testing facilities able to explore new regions of the constitutive manifold. In any case, sparse sampling within the compressed sensing framework appears as a valuable option in data-driven computational mechanics applications.

5 Model order reduction

We consider a last possible application of compressed sensing. It concerns the application to model order reduction, particularly in its hyper-reduction variant, revisited below.

Standard discretization of a given model in the form of a PDE equipped with suitable initial and boundary conditions leads to a linearized system

$$\mathbf{KU} = \mathbf{G}, \tag{9}$$

where, as usual, $\mathbf{K}$ represents the tangent stiffness matrix, $\mathbf{U}$ a vector containing the nodal degrees of freedom, and $\mathbf{G}$ the nodal force vector.

When considering a reduced basis—based on the application of POD or RB methodologies, for instance—, the unknown vector $\mathbf{U}$ can be projected onto the reduced basis according to $\mathbf{U} = \mathbf{Bu}$. The size of vector $\mathbf{u}$ is in general much smaller than the size of the original unknown vector $\mathbf{U}$. Here, $\mathbf{B}$ represents the basis transformation matrix, whose columns are the nodal description of the approximation functions involved in the reduced basis. Thus, the original algebraic system can be rewritten as

$$\mathbf{KBu} = \mathbf{G},$$

that premultiplying by the transpose of $\mathbf{B}$ leads to the reduced system

$$\mathbf{B}^T \mathbf{KBu} = \mathbf{B}^T \mathbf{G} = \mathbf{g},$$

that can be viewed as a Galerkin discretization operating with the reduced basis instead of the one related to the usual finite element approximation.

However, Ryckelynck noted that since the size of vector $\mathbf{u}$ is reduced, one could consider only a few equations for computing it, and called the technique hyper-reduction [41].

Thus, he suggested to perform integration only in some particular elements of the mesh, thus leading to a matrix $\mathbf{K}^{hr}$ (where the superscript refers to its hyper-reduced nature), whose majority of rows are actually not evaluated and thus populated by zeros. However, the resulting reduced system $\mathbf{B}^T\mathbf{K}^{hr}\mathbf{B}$ is invertible. Ryckelynck studied is his works different strategies to choose the best elements in which to perform integration, and many other authors proposed different alternatives with a similar objective. It is also important to note that $\mathbf{B}^T\mathbf{K}^{hr}$ could be expressed as $\mathbf{B}^{PG,T}\mathbf{K}$, with $\mathbf{B}^{PG}$ ensuring the equivalence $\mathbf{B}^{PG,T}\mathbf{K} = \mathbf{B}^T\mathbf{K}^{hr}$, that allows us to interpret the hyper-reduction as a Petrov-Galerkin formulation. Advanced hyper-reduction methodologies were considered by the same authors as well as by Farhat and coauthors [18]. In what follows, we consider a similar procedure, but now inspired from the compressed sensing rationale.

The starting point is again the original algebraic system (9) that, by assuming sparsity in a target space (e.g. discrete cosines, Fourier, wavelet, …) for both the source and the unknown vectors, $\mathbf{G}$ and $\mathbf{U}$ respectively, allows writing, after assuming the same target space for both vectors, expressed from the transformation matrix $\mathbf{T}$,

$$\begin{cases} \mathbf{G} = \mathbf{T}\hat{\mathbf{G}} \\ \mathbf{U} = \mathbf{T}\hat{\mathbf{U}} \end{cases},$$

and consequently

$$\mathbf{K}\mathbf{T}\hat{\mathbf{U}} = \mathbf{T}\hat{\mathbf{G}},$$

or

$$\mathbf{T}^{-1}\mathbf{K}\mathbf{T}\hat{\mathbf{U}} = \hat{\mathbf{G}},$$

on which the extraction applies.

To show the potential of this proposal we consider again the discrete system that results from the discretization of a plane-stress elastic problem. Figure 4 depicts the reference solution and the one resulting from the reduced formulation just described when extracting randomly 25% of the involved equations, proving the potential of the proposed methodology.

6 Time-dependent problems

In this section we will develop a space-time approach for the solution of transient problems with the help of compressed sensing techniques. We will study three different partial differential equations. Namely, steady, transient Poisson and wave equations will be considered, Eqs. (10), (11) and (12), respectively:
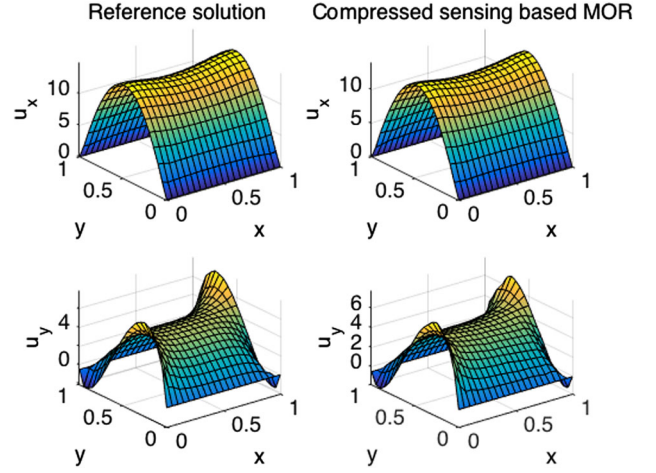

Fig. 4 Reference versus compressed sensing-based model order reduction

$$\alpha \Delta u(\mathbf{x}, t) = b(\mathbf{x}, t) \quad \forall \mathbf{x} \in \Omega, \tag{10}$$

$$\frac{\partial u}{\partial t} - \alpha \Delta u(\mathbf{x}, t) = b(\mathbf{x}, t) \quad \forall \mathbf{x} \in \Omega, \tag{11}$$

and

$$\frac{\partial^2 u}{\partial^2 t} - \alpha \Delta u(\mathbf{x}, t) = b(\mathbf{x}, t) \quad \forall \mathbf{x} \in \Omega. \tag{12}$$

These equations must be equipped with suitable Dirichlet boundary conditions at some part of the boundary $\partial\Omega_D$ in order to make the solution unique. For the sake of simplicity, but without loosing generality, we will impose homogeneous Dirichlet boundary conditions at the Dirichlet portion of the Domain $\Omega = [0, 1] \times [0, 1]$, Ω_D,

$$u(x = 0, y) = 0.$$

When dealing with time derivatives, initial conditions have to be imposed,

$$u(\mathbf{x}, 0) = 0,$$

and, possibly, in the case of the wave equation also,

$$\dot{u}(\mathbf{x}, 0) = 0.$$

We will consider a source term that varies in time as,

$$b(\mathbf{x}, t) = A \cos \omega t.$$

To approximately solve Eqs. (10), (11) and (12), finite elements in space and finite differences in time have been used. Therefore, equilibrium for steady/transient Poisson and wave equations at the i-th time step read

$$\mathbf{Ku}^i = \mathbf{f}^i,$$

$$\frac{1}{\Delta t}\mathbf{M}(\mathbf{u}^i - \mathbf{u}^{i-1}) + \mathbf{Ku}^i = \mathbf{f}^i,$$

and

$$\frac{1}{\Delta t^2}\mathbf{M}(\mathbf{u}^{i+1} - 2\mathbf{u}^i + \mathbf{u}^{i-1}) + \mathbf{Ku}^i = \mathbf{f}^i,$$

respectively.

Assume now that we prefer to solve every time step together. We should write a single system of equations of the form

$$\mathbf{A}\hat{\mathbf{u}} = \hat{\mathbf{f}}. \tag{13}$$

Here, matrix $\mathbf{A}$ is a block-diagonal matrix containing a matrix $\mathbf{K}$ at each block for the steady Poisson case. The first and second time derivatives appearing in transient Poisson and wave equations will generate coupling terms between consecutive time steps by means of the $\mathbf{M}$ matrices. $\hat{\mathbf{u}}$ and $\hat{\mathbf{f}}$ are the concatenation of spatial nodal unknowns and forces for every time step.

Solving directly the system (13) is a legit approach. Indeed, the usual time-marching approach is recovered. However, we would like to explore the advantages of compressed sensing. If the unknown field $\hat{\mathbf{u}}$ is projected onto a new basis enabling a sparse representation, a hyper-reduction results, as seen in the previous section. Therefore, an underdetermined system needs to be solved by performing a L1-norm minimization. Thus, we will seek to solve the following system:

$$\mathbf{EAT}\hat{\mathbf{c}} = \mathbf{E}\hat{\mathbf{f}},$$

where $\mathbf{T}$ is again the projection matrix, and $\mathbf{E}$ is the extraction operator that defines randomly which rows are going to be selected to perform the L1 minimization. Finally, $\hat{\mathbf{c}}$ are the unknown coefficients in the new basis. When a time step is chosen, every spatial node related to this time step is automatically selected to keep spatial global equilibrium.

Since we know that the dependance in time of the source term was caused by a cosine function, a smart choice for the projection base in our case is the discrete cosines transform (DCT). It is worth to say that the choice of the projection basis is problem dependent. For instance, if our excitation force evolves as a Heaviside step function, it may be convenient to use a Haar-wavelet-based projection.

6.1 Numerical performance of the LASSO scheme

The functional that the LASSO algorithm seeks to minimize does not have a closed-form solution. Hence, an iterative algorithm has to be used in order to find the minimum of the
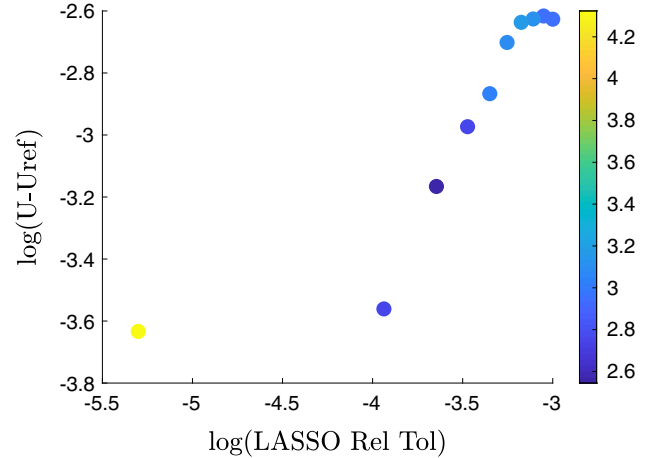

Fig. 5 Relative Tolerance of LASSO method versus solution error in logarithmic scale. The legend represent the time, in seconds, required to solve the LASSO minimization. 10 out 75 time steps are sampled

functional, such as a steepest descent method. When using LASSO, there is a numerical parameter called the relative tolerance, which controls the maximum admissible difference between two iterations of the steepest descent method. If the difference between two iterations is smaller than the relative tolerance, the iterative algorithm stops. Therefore, the smaller the relative tolerance, the higher number of iterations are required, but the solution will be more accurate. Figure 5 shows the relative tolerance of LASSO method versus solution error in logarithmic scale. The colors of the points represent the time, in seconds, required to solve the underdetermined system by means of LASSO. As it can be noticed, the lower the relative tolerance, the lower error in the solution is obtained. Yet, the time to solve the system increases. The number of spatial snapshots $\mathbf{u}^i$ to build the LASSO system was 10 out of 75 total time steps. The solution error is measured by

$$\epsilon_U = \frac{1}{N}||\hat{\mathbf{u}} - \hat{\mathbf{u}}_R||_{L2},$$

where $\hat{\mathbf{u}}_R$ is the reference solution, which is assumed to be the one of the usual time marching approach. N represents the number of total unknowns.

At the end of the day, a compromise between the solution error and computational cost should be accomplished. The quality of the reconstructed solution depends also on the number of snapshots. Figure 6 shows the number of snapshots versus solution error in logarithmic scale. The legend represents the time required to solve the LASSO minimization. The relative tolerance is set to $0.8e-5$. Obviously, the more snapshots $\mathbf{u}^i$ we consider, the more accurate is the solution and less time is required to solve the system of equations.

In our cases, a relative tolerance of $1e-5$ provides good results. If the relative tolerance is set to $1e-6$, it will provide
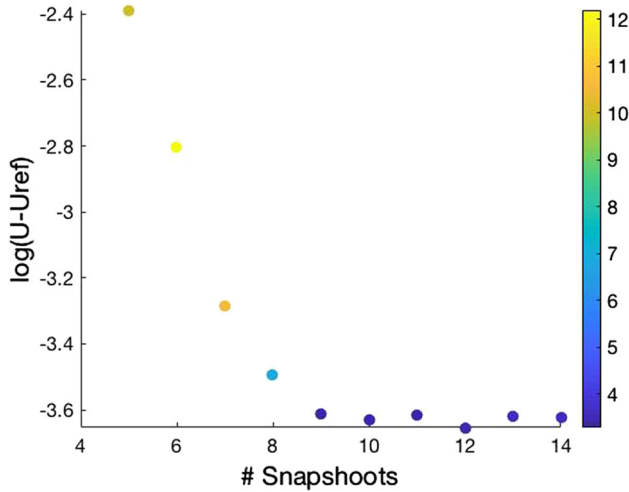
Fig. 6 Number of snapshots versus solution error in logarithmic scale. The legend represents time, in seconds, required to solve the LASSO minimization. Relative tolerance set to $1e-5$
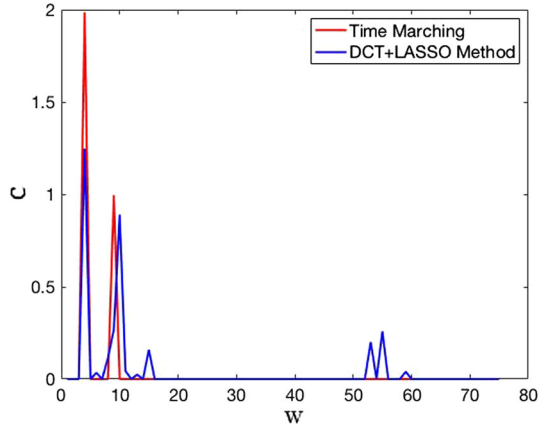

Fig. 7 DCT coefficients at a given spatial node with a bad tolerance in LASSO (Relative tolerance $1e-4$)

even better results in terms of error, but it will take more time to minimize the functional. Figure 7 shows the DCT coefficients solving the entire system of equations (red) and the ones obtained after minimizing LASSO (blue) with a relative tolerance of $1e-4$. As it can be noticed, there are some differences between the red and blue curves, meaning that the iterative algorithm has finished before reaching the proper minimum of the functional. Figure 8 shows how imposing a smaller tolerance will provide better results. Of course, the price to pay is that the LASSO minimization takes more time.

6.2 Numerical results

In this section, several numerical examples involving the three equations considered in the previous section will be analyzed. Sparsity of the new basis will play an important role to
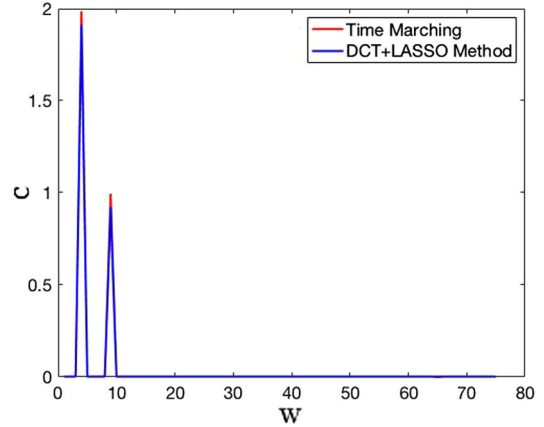

Fig. 8 DCT coefficients at a given spatial node equation with a good tolerance in LASSO (Relative tolerance $0.8e-5$)

Table 1 Parameters employed in the analysis of Eqs. (10)–(12)

Parameter	Value
T_f, sim. time	0.1
ω	125.6
Number of time steps	200
α	1
A	-100
Ω	$[0, 1] \times [0, 1]$
Spatial dofs	16×16

make the LASSO algorithm efficient. Parameters employed in the solution of Eqs. (10)–(12) are compiled in Table 1.

The first equation to be tested is Eq. (10). The global system of equations is uncoupled from time step to time step due to the fact that it does not involve a time derivative. Figure 9 depicts the value of $u(\mathbf{x} = (0.5, 0.5), t)$ for both an usual time-marching approach and for a compressed sensing-based procedure. In this case only 5 time instants out of 200 are considered in the LASSO minimization. As it can be noticed, the blue curve tends to capture the overall behavior of the red curve (reference solution). However, there are still some noisy peaks appearing in the blue curve due to the small number of time instants involved in the computation. Figure 10 shows the coefficients of the discrete cosines transform, which is indeed what the LASSO algorithm computes. It can be noticed how some high frequency peaks are appearing causing the small oscillations in the u field.

The results are better if we increase the number of sampled time instants up to ten. Figures 11 and 12 show u and c fields, respectively. It should be highlighted that the high-frequency peaks are no longer appearing. Furthermore, the leakage pollution close to the main peak is filtered thanks to the LASSO algorithm.
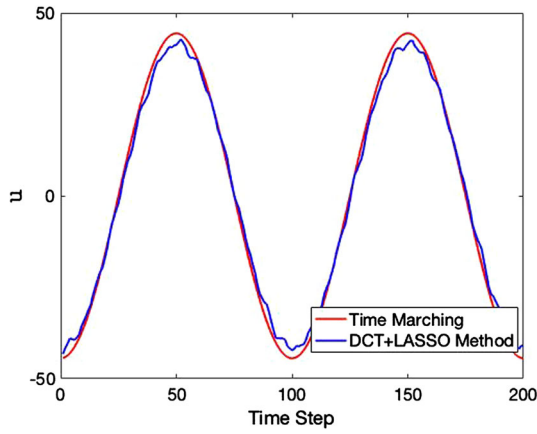
Fig. 9 $u(\mathbf{x} = (0.5, 0.50), t)$ for steady Poisson's case. Five time instants are sampled out of two hundred possible time steps
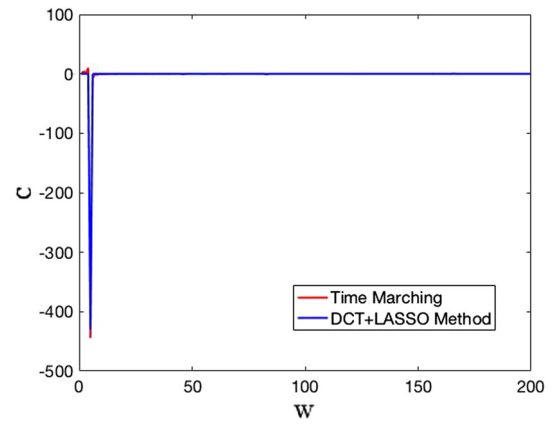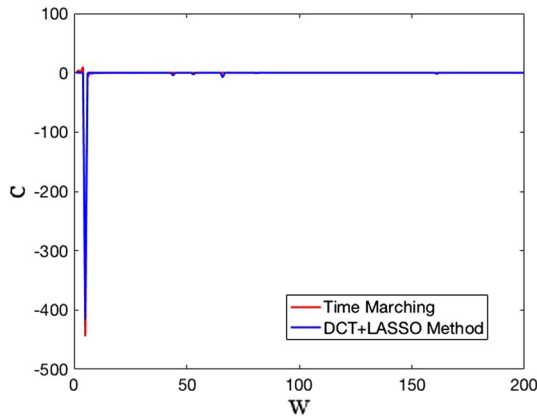

Fig. 10 $c(\mathbf{x} = (0.5, 0.50), \omega)$ for steady Poisson's case. Five time instants are sampled out of two hundred possible time steps
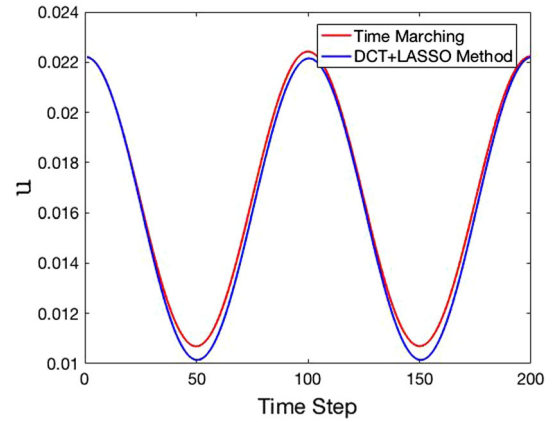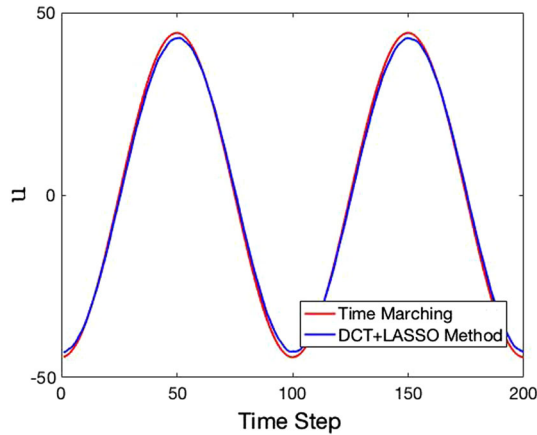

Fig. 11 $u(\mathbf{x} = (0.5, 0.50), t)$ for steady Poisson's case. Five time instants are sampled out of two hundred possible time steps

The case of the wave Eq. (12), with a second derivative in time, is less problematic than the one involving first derivatives only, see Eq. (11), since it forces the response to follow the loading, whereas the transient Poisson equation involves

Fig. 12 $c(\mathbf{x} = (0.5, 0.50), \omega)$ for steady Poisson's case. Five time instants are sampled out of two hundred possible time steps

Fig. 13 $u(\mathbf{x} = (0.5, 0.50), t)$ for the wave equation. 30 time instants are sampled out of two hundred possible time steps

diffusion. Therefore, we will expect a sparser solution than in problem (11).

Figure 13 shows the u field for the wave equation case. Since the source term is a cosine function, the response of the system is also a cosine. Furthermore, the system must be initialized with a non-homogeneous initial condition. Otherwise, compatibility conditions of the time marching approach will no longer be satisfied. Figure 14 shows the DCT coefficients $\mathbf{c}$ for the wave equation case. As it can be noticed, the first coefficient accounts for a solid rigid-like translation (i.e., the cosine is not centered in 0) and the other peak coincides with the frequency of the source term.

Figures 15, 16, 17 and 18 show the expected behavior: imposing a cosine in the source term with a first time derivative will cause a sinusoidal response, which is not sparse in the projected base, due to the diffusion effects. Hence, more time instants need to be taken into account in order to achieve an accurate result.
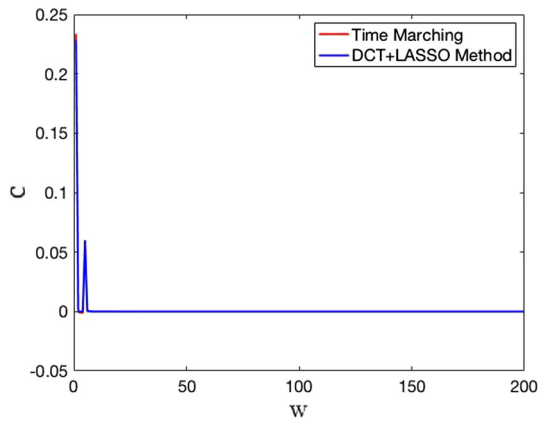
Fig. 14 $c(\mathbf{x} = (0.5, 0.50), t)$ for the wave equation. 30 time instants are sampled out of two hundred possible time steps
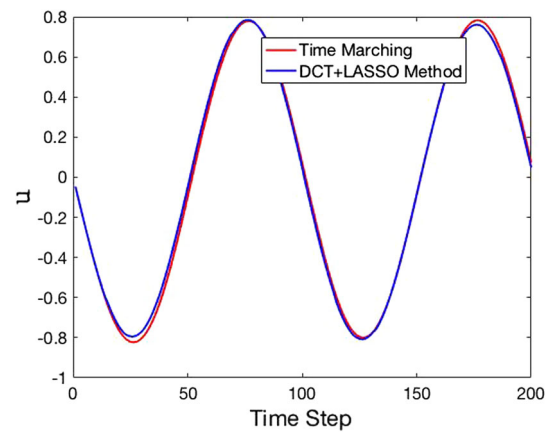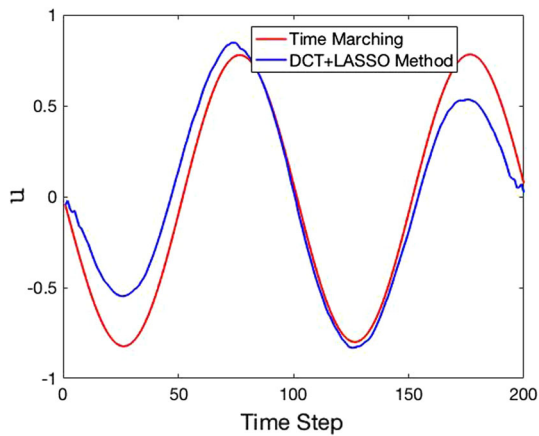

Fig. 15 $u(\mathbf{x} = (0.5, 0.50), t)$ for transient Poisson's case. 100 time instants are sampled out of two hundred possible time steps

Fig. 16 $c(\mathbf{x} = (0.5, 0.50), t)$ for transient Poisson's case. 100 time instants are sampled out of two hundred possible time steps

7 Conclusions

In this work we explored different applications of compressed sensing in computational mechanics. First, we proved that it
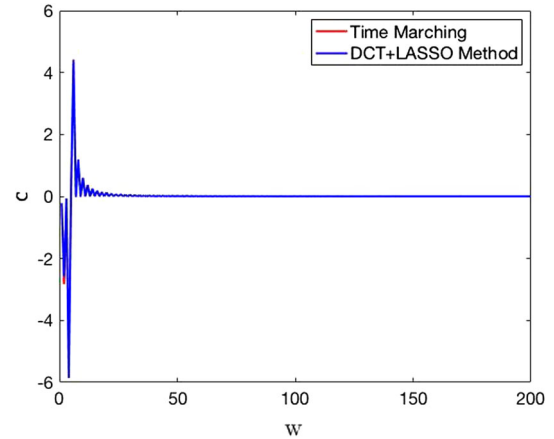

Fig. 17 $u(\mathbf{x} = (0.5, 0.50), t)$ for transient Poisson's case. 150 time instants are sampled out of two hundred possible time steps

Fig. 18 $c(\mathbf{x} = (0.5, 0.50), t)$ for transient Poisson's case. 150 time instants are sampled out of two hundred possible time steps
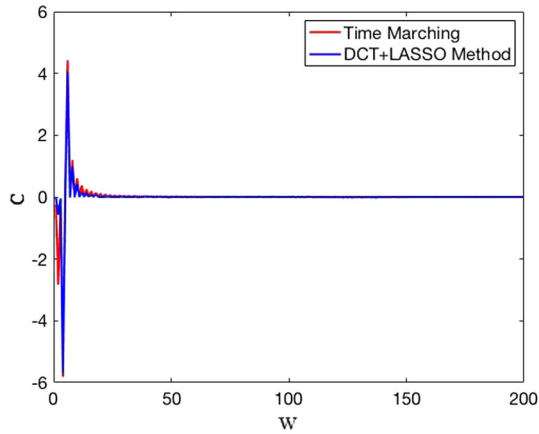
could be a valuable strategy for performing random samplings to evaluate solutions of parametric models. Then, inspired from the so-called "single pixel camera", we analyzed the use of the compressed sensing methodology to reconstruct constitutive manifolds from the only knowledge of a quire reduced number of data-points. Finally, in the preceding section, we proved that the same methodology can be employed for reducing significantly the size of the discrete system of equations that results from the application of standard discretization techniques defining a new kind of model order reduction techniques.

All these different application rely on a single technique and we strongly believe that it will play a very relevant role in computational mechanics for the years to come.

References

1. Ammar A, Mokdad B, Chinesta F, Keunings R (2006) A new family of solvers for some classes of multidimensional partial differential equations encountered in kinetic theory modeling of complex fluids. J Non-Newtonian Fluid Mech 139:153–176

2. Ammar A, Mokdad B, Chinesta F, Keunings R (2007) A new family of solvers for some classes of multidimensional partial differential equations encountered in kinetic theory modeling of complex fluids. Part II: transient simulation using space-time separated representations. J Non-Newton Fluid Mech 144:98–121

3. Amsallem D, Farhat C (2008) An interpolation method for adapting reduced-order models and application to aeroelasticity. AIAA J 46:1803–1813

4. Benner P, Gugercin S, Willcox K (2015) A survey of projection-based model reduction methods for parametric dynamical systems. SIAM Rev 57(4):483–531

5. Bognet B, Bordeu F, Chinesta F, Leygue A, Poitou A (2012) Advanced simulation of models defined in plate geometries: 3D solutions with 2D computational complexity. Comput Methods Appl Mech Eng 201–204:1–12. https://doi.org/10.1016/j.cma.2011.08.025

6. Borzacchiello D, Aguado JV, Chinesta F (2017) Non-intrusive sparse subspace learning for parametrized problems. Arch Comput Methods Eng. https://doi.org/10.1007/s11831-017-9241-4

7. Borzacchiello D, Aguado JV, Chinesta F (2017) Reduced order modelling for efficient numerical optimisation of a hot-wall chemical vapour deposition reactor. Int J Numer Methods Heat Fluid Flow 27(7):1602–1622. https://doi.org/10.1108/HFF-04-2016-0153

8. Breitkopf P, Naceur H, Rassineux A, Villon P (2005) Moving least squares response surface approximation: formulation and metal forming applications. Comput Struct 83(17–18):1411–1428

9. Brunton SL, Proctor JL, Kutz JN (2016) Discovering governing equations from data by sparse identification of nonlinear dynamical systems. Proc Natl Acad Sci. https://doi.org/10.1073/pnas.1517384113

10. Bungartz HJ, Griebel M (2004) Sparse grids. Acta Numer 13:147–269

11. Chinesta F, Ammar A, Cueto E (2010) Recent advances in the use of the proper generalized decomposition for solving multidimensional models. Arch Comput Methods Eng 17(4):327–350

12. Chinesta F, Cueto E (2014) PGD-based modeling of materials, structures and processes. Springer, Berlin

13. Chinesta F, Huerta A, Rozza G, Willcox K (2017) Model reduction methods. Encyclopedia of computational mechanics, 2nd edn. Wiley, Hoboken

14. Chinesta F, Ladeveze P, Cueto E (2011) A short review on model order reduction based on proper generalized decomposition. Arch Comput Methods Eng 18:395–404

15. Chinesta F, Leygue A, Bordeu F, Aguado J, Cueto E, Gonzalez D, Alfaro I, Ammar A, Huerta A (2013) PGD-based computational vademecum for efficient design, optimization and control. Arch Comput Methods Eng 20(1):31–59. https://doi.org/10.1007/s11831-013-9080-x

16. Cueto E, González D, Alfaro I (2016) Proper generalized decompositions: an introduction to computer implementation with Matlab. SpringerBriefs in applied sciences and technology. Springer, Berlin

17. Everson R, Sirovich L (1995) Karhunen-loève procedure for gappy data. J Opt Soc Am A 12(8):1657–1664. https://doi.org/10.1364/JOSAA.12.001657

18. Farhat C, Chapman T, Avery P (2015) Structure-preserving, stability, and accuracy properties of the energy-conserving sampling and weighting method for the hyper reduction of nonlinear finite element dynamic models. Int J Numer Methods Eng 102(5):1077–1110. https://doi.org/10.1002/nme.4820

19. González D, Aguado JV, Cueto E, Abisset-Chavanne E, Chinesta F (2016) kPCA-based parametric solutions within the PGD framework. Arch Comput Methods Eng. https://doi.org/10.1007/s11831-016-9173-4

20. González D, Chinesta F, Cueto E (2019) Learning corrections for hyperelastic models from data. Front Mater. https://doi.org/10.3389/fmats.2019.00014

21. Ibanez R, Abisset-Chavanne E, Aguado JV, Gonzalez D, Cueto E, Chinesta F (2018) A manifold learning approach to data-driven computational elasticity and inelasticity. Arch Comput Methods Eng 25(1):47–57

22. Ibañez R, Abisset-Chavanne E, Ammar A, González D, Cueto E, Huerta A, Duval JL, Chinesta F (2018) A multi-dimensional data-driven sparse identification technique: the sparse proper generalized decomposition. Complexity. https://doi.org/10.1155/2018/5608286

23. Ibañez R, Abisset-Chavanne E, Gonzalez D, Duval J, Cueto E, Chinesta F (2018) Hybrid constitutive modeling: Data-driven learning of corrections to plasticity models. Int J Mater Form. https://doi.org/10.1007/s12289-018-1448-x

24. Ibañez R, Borzacchiello D, Aguado JV, Abisset-Chavanne E, Cueto E, Ladeveze P, Chinesta F (2017) Data-driven non-linear elasticity: constitutive manifold construction and problem discretization. Comput Mech 60(5):813–826. https://doi.org/10.1007/s00466-017-1440-1

25. Kaiser E, Kutz JN, Brunton SL (2018) Sparse identification of nonlinear dynamics for model predictive control in the low-data limit. Proc R Soc Lond A Math Phys Eng Sci 474(2219):20180335. https://doi.org/10.1098/rspa.2018.0335

26. Kutz JN (2013) Data-driven modeling and scientific computation. Methods for complex systems and big-data. Oxford University Press, Oxford

27. Ladeveze P (1989) The large time increment method for the analyze of structures with nonlinear constitutive relation described by internal variables. Comptes Rendus Académie des Sciences Paris 309:1095–1099

28. Ladeveze P (1999) Nonlinear computational structural mechanics. Springer, New York

29. Lee J, Verleysen M (2007) Nonlinear dimensionality reduction. Springer, New York

30. Leon A, Barasinski A, Abisset-Chavanne E, Cueto E, Chinesta F (2018) Wavelet-based multiscale proper generalized decomposition. Comptes Rendus Academie de Sciences - Mécanique 346(7):485–500

31. Lopez E, Gonzalez D, Aguado JV, Abisset-Chavanne E, Cueto E, Binetruy C, Chinesta F (2016) A manifold learning approach for integrated computational materials engineering. Arch Comput Methods Eng. https://doi.org/10.1007/s11831-016-9172-5

32. Maaten Lvd, Hinton G (2008) Visualizing data using t-SNE. J Mach Learn Res 9(1):2579–2605

33. Mangan NM, Brunton SL, Proctor JL, Kutz JN (2016) Inferring biological networks by sparse identification of nonlinear dynamics. IEEE Trans Mol Biol Multi-Scale Commun 2(1):52–63. https://doi.org/10.1109/TMBMC.2016.2633265

34. Meng L, Breitkopf P, Le Quilliec G, Raghavan B, Villon P (2018) Nonlinear shape-manifold learning approach: concepts, tools and applications. Arch Comput Methods Eng 25(1):1–21

35. Millán D, Arroyo M (2013) Nonlinear manifold learning for model reduction in finite elastodynamics. Comput Methods Appl Mech Eng 261–262:118–131. https://doi.org/10.1016/j.cma.2013.04.007

36. Nouy A (2010) A priori model reduction through proper generalized decomposition for solving time-dependent partial differential equations. Comput Methods Appl Mech Eng 199(23–24):1603–1626. https://doi.org/10.1016/j.cma.2010.01.009
37. Patera A, Rozza G (2007) Reduced basis approximation and a posteriori error estimation for parametrized partial differential equations. Tech. rep., MIT Pappalardo Monographs in Mechanical Engineering
38. Quarteroni A, Rozza G, Manzoni A (2011) Certified reduced basis approximation for parametrized PDE and applications. J Math Ind 1:3
39. Roweis ST, Saul LK (2000) Nonlinear dimensionality reduction by locally linear embedding. Science 290(5500):2323–2326. https://doi.org/10.1126/science.290.5500.2323
40. Rozza G, Huynh D, Patera A (2008) Reduced basis approximation and a posteriori error estimation for affinely parametrized elliptic coercive partial differential equations—application to transport and continuum mechanics. Arch Comput Methods Eng 15(3):229–275
41. Ryckelynck D (2005) A priori hyperreduction method: an adaptive approach. J Comput Phys 202(1):346–366
42. Tibshirani R (1996) Regression shrinkage and selection via the lasso. J R Stat Soc Ser B (Methodol) 58(1):267–288
43. Volkwein S (2001) Model reduction using proper orthogonal decomposition. Tech. rep., Lecture Notes, Institute of Mathematics and Scientific Computing, University of Graz