Robust online updating of a digital twin with imprecise probability

Marco de Angelis\textsuperscript{a,b,*}, Ander Gray\textsuperscript{a}, Scott Ferson\textsuperscript{a}, Edoardo Patelli\textsuperscript{b}

\textsuperscript{a} Institute for Risk and Uncertainty, University of Liverpool, United Kingdom
\textsuperscript{b} Civil and Environmental Engineering, University of Strathclyde, United Kingdom

**A R T I C L E   I N F O**

Communicated by J.E. Mottershead

**Keywords:**
- Imprecise probability
- Evidence theory
- Possibility theory
- Nonparametric inference
- Online updating
- Digital twins
- Engineering dynamics
- Model calibration

**A B S T R A C T**

We present a method for the robust online updating of the parameters of a digital twin for engineering dynamics. The method is robust because it makes neither prior nor likelihood assumptions while rigorously quantifying the inferential uncertainty, which is useful in the context of scarce data. With the proposed updating strategy the digital twin can update effectively and nearly instantaneously when presented with additional data. An algorithm for nonparametric inference with consonant random sets recently developed by the authors is applied. The method constructs a consonant joint structure of the updating parameters. Such structure is composed of a sequence of nested sets, each of which is assigned a belief measure, i.e. a lower probability, with the desired level of confidence. This joint structure represents an outer approximation of the credal set where the target joint probability distribution resides, thus can be regarded as a rigorous inferential model. The obtained joint structure can be given a possibilistic interpretation thus seen as a joint fuzzy set, and can be converted to a joint probability box using established transformation. The method is arguably more efficient than traditional competitive Bayesian updating methods because intrinsically non-sequential thus parallelizable. The inference method can also be extended, with nearly no additional computational cost, to the case of interval measurement uncertainty. Because of its efficiency the method can update the parameters of a digital twin model practically in real time, provided that: (1) the data generating mechanism is stationary, (2) the physical model has not changed, e.g. due to damage, and (3) input–output simulation data within the range of interest is available.

We present an application to the online updating of a digital twin of an aluminium three-storey structure subject to impact-hammer excitation.

**1. Introduction**

The concept of imprecise probability has gained traction within the statistical inference community in recent years [1,2]. The idea that inferential uncertainty could be quantified by relaxing the axioms of classical probability seemed odd initially, but proved largely beneficial. Imprecise probability is based on the abstract concept of a credal set, i.e. a set of coherent probability distributions, whereby each individual element is a precise probability distribution [3–5]. This concept can be conveniently visualised in the context of discrete univariate distributions, but tend to be very abstract, thus difficult to practically deal with in the context of multivariate (correlated) continuous distribution functions. To this date, the level of abstraction required by a credal set of multivariate distributions is still too high for practical computations. Theoretical paradigms like evidence theory [6], possibility theory [7], and probability bounds analysis [8,9], just to mention a few, have emerged to address the problem of computing with imprecise probabilities, by restraining abstraction, while preserving conservatism in quantifying epistemic uncertainty. The
literature on structural model updating, beyond the outdated deterministic approach, has mainly been Bayesian [10–13]. The Bayesian paradigm provides an attractive and flexible framework to formulate the problem while taking explicit account of the uncertainties. With the method of approximate Bayesian computations, the analysis can be conducted nonparametrically without making distributional nor dependence assumptions, and the computation can take place efficiently using popular and widespread Markov chain Monte Carlo algorithms. Nevertheless, the Bayesian updating is tied to the choice of a prior joint probability distribution, which can contaminate the results if data is scarce. Moreover, results from a Bayesian updating come in the form of samples from a posterior joint probability distribution, which can be difficult to handle for further uncertainty propagation. For these reasons, the need for alternative, competitive methodologies have emerged in the literature for interval and fuzzy model updating [14,15]. These have enabled the analyst to make fewer assumptions in the presence of scarce data, while keeping a grounded connection with the physical problem under study.

Imprecise probability relaxes the fundamental axiom of additivity to allow for more flexibility in the inference process. This unlocks a whole new spectrum of possibilities for the inference task, effectively removing the need for specifying a prior. The goal of the inference task is no longer determining a posterior probability distribution, but approximating the set of probability distributions that are conformant with the data. If this approximation is done in such a way that no conformant probability distributions are left out, then results are said to be rigorous, in the sense that the inferential uncertainty is outer approximated.

The paper is organised as follows: in Section 2 the nonparametric inference method used to construct the consonant joint structure is presented, in Section 3 the method is applied to the updating of a digital twin. In Section 4 we close the paper with a concluding statement followed by directions to reproduce the results presented in the paper in Section 5.

1.1. Novelty of the contribution

This paper has two main elements of novelty: (i) a full-blown imprecise probability nonparametric methodology for structural model updating, and (ii) the application of such methodology to a digital twin.

2. Robust inference

In this section, the prior- and likelihood-free nonparametric inference method at the core of this paper is introduced. The method is designed to characterise the inferential uncertainty of a vector of parameters, given set of observations, and given an input–output relationship \( f \). The inferential uncertainty of the updated parameters is expressed in the form of a consonant joint belief structure.

2.1. Problem statement

Let \( X \in \mathbb{R}^d \) be a real-valued discrete random variable on a measurable space \( \mathcal{X} \) with unknown probability distribution \( P_X \). We study the belief (measure) held by an agent about a future realisation of \( X \) from past observations \( X_1, \ldots, X_n \) drawn randomly from the same distribution (iid). In evidence theory, this belief function is demanded to be less committed than an exact probability distribution; in particular, for any set in \( \mathcal{X} \), its measure ought to be a lower bound on the true unknown probability measure.

2.2. Inference with belief functions

Let us denote a belief function in \( \mathcal{X} \), given some observations \( X_1, \ldots, X_n \), by \( \text{Bel}_X := \text{Bel}(\cdot|X_1, \ldots, X_n) \). What properties should such belief function satisfy? As more data is gathered, it seems reasonable to postulate that \( \text{Bel}_X = P_X \) for \( n \to \infty \), from a frequentist perspective [16]. A sample set of infinite size should be equivalent to knowing the distribution of \( X \), hence the belief function should asymptotically become identical to \( P_X \). For finite \( n \), it seems natural to impose that \( \text{Bel}_X \) is less committed than \( P_X \), which leads to \( \text{Bel}_X \leq P_X \). However, because of limited data, the inequality is often too stringent to hold in all cases. In other terms, assuming the experiment is repeated indefinitely, we demand \( \text{Bel}_X \) to be less committed than \( P_X \) most of the time, with at least some prescribed long run frequency \( 1 - \beta \), where \( \beta \in (0, 1) \) is a small number. In summary, the following two requirements are typically imposed to the belief function:

1. \( \forall A \subseteq \mathcal{X}, \text{Bel}_X(A) = P_X(A), \text{ for } n \to \infty, \text{ and} \)
2. \( P^n(\text{Bel}_X \leq P_X) \geq 1 - \beta \).

A belief function satisfying these two requirements is called a predictive belief function. \( 1 - \beta \) is often interpreted as a level of confidence attached to the belief measure.

Several methods have been proposed to construct these belief functions, like in [16], using multinomial confidence intervals, and then further extended to the continuous case [17]. In this work, we restrict our construction to sets that are mutually nested, while computing the belief measure separately. A consonant belief function will only meet requirement (2). Because of consonance, requirement (1) is never fully realised, so as we gather more and more data the belief never converges to the true probability measure. For this reason the method is particularly suited for small-sized data sets. When the size of the data set is large parametric or Bayesian inferential models should be used to harness the full potential of the data set and achieve better resolution towards the true underlying probability, see e.g. [1,2].
2.3. A belief function is a coherent lower probability

Let us recall that a belief function is a coherent lower probability. Given a power set \( 2^X \), the function \( m : 2^X \rightarrow [0, 1] \) is called a basic belief assignment or bba, if the following two properties hold: (i) \( m(\emptyset) = 0 \), (ii) \( \sum_{A \subseteq X} m(A) = 1 \). Note that (i) implies \( m(A) > 0 \).

The basic belief assignment \( m(A) \) can be interpreted as the part of the agent’s belief allocated exactly to the hypothesis that \( X \) takes some value in \( A \) [6]. The subsets \( A \subseteq X \) such that \( m(A) > 0 \) are called the focal sets of \( m \). For any focal set \( A \subseteq X \), its belief can be obtained summing up the masses of the focal elements \( B \in 2^X \) contained in it,

\[
    \text{Bel}_X(A) = \sum_{B : B \subseteq A} m(B).
\] (1)

When the focal sets are nested, \( \text{Bel}_X \) is said to be consonant. Nested focal sets are also called level sets in this paper. Consonance has recently been used to build confidence preserving structures [18]. Eq. (1) constitutes a lower bound on the true yet unknown probability \( P_X(A) \) to the hypothesis that \( X \) takes some value in \( A \). Thus the belief function \( \text{Bel}_X \) falls in a subclass of coherent lower probabilities [5,19].

2.4. Possibility theory

Possibility theory is a special branch of evidence theory that deals with bodies of evidence whose focal elements are nested.

A possibility distribution is defined by its membership function \( r_X \) which is a function that maps from the sample space \( X \) to the unit interval: \( r_X : X \rightarrow [0, 1] \), requiring that the largest value is unitary: \( \sup_{x \in X} r_X(x) = 1 \). A possibility distribution induces two measure, the possibility measure \( \text{Pos}_X \), given by

\[
    \text{Pos}_X(U) = \sup_{x \in U} r_X(x),
\] (2)

and the necessity measure \( \text{Nec}_X \),

\[
    \text{Nec}_X(U) = 1 - \sup_{x \in U} r_X(x) = 1 - \text{Pos}_X(U),
\] (3)

where \( U \) denotes the complement of \( U \). These dual measures can be interpreted as being bounds on the probability measure \( P_X \) of an imprecisely known random variable [20] \( \text{Nec}_X(U) \leq P_X(U) \leq \text{Pos}_X(U) \). Stated another way, a random variable \( X \) is a member of the set of probability distributions (the credal set) defined by the possibility distribution \( r_X \) if its probability measure is bounded by \( \text{Nec}_X \) and \( \text{Pos}_X \) for all sets \( U \). Under this interpretation, a possibility distribution can be considered as a special case of a belief function with nested focal elements, i.e. a consonant belief function. The possibility and necessity measures are the special counterpart of evidence theory’s plausibility and belief measures.

2.4.1. The extension principle

One of the advantages of possibility theory as an imprecise probability theory is its allowing us to compute with uncertainty. For a function \( f : \mathbb{R}^d \rightarrow \mathbb{R} \) with \( d \) input marginal possibility distributions \( r_{X_1}, \ldots, r_{X_d} \), the possibility distribution of the output variable \( Y = f(X_1, \ldots, X_d) \) can be found by the classical extension principle

\[
    r_Y(y) = \sup_{(x_1, \ldots, x_d) \in \mathbb{R}^d} \min_{y = f(x_1, \ldots, x_d)} \left( r_{X_1}(x_1) \ldots r_{X_d}(x_d) \right).
\] (4)

Some authors (e.g. see [21,22]) found that such extension principle is not consistent with probability theory in the majority of situations. That is, the multivariate possibility distribution defined by \( J_X = \min \left( r_{X_1}(x_1) \ldots r_{X_d}(x_d) \right) \) need not bound a multivariate probability distribution, even if the marginal possibilities bounds marginal probability distributions. The original extension principle neglects stochastic dependence if used for probability bounding [23]. For this reason, herein the updating produces a joint possibility distribution altogether, effectively removing the need to apply the extension principle as appears in (4). This ensures that the stochastic dependency is captured and bounded by the resulting joint possibility distribution.

2.5. Inversion

In this work, we circumvent the use of the extension principle as in (4), thus we construct by inversion a multivariate possibility distribution \( J_X^{\text{inv}} \) directly from data, while capturing the dependence. Since the credal set defined by \( J_X^{\text{inv}} \) bounds the unknown random variable \( X \), the computed joint possibility distribution \( J_Y \) is consistent with the unknown random variable \( Y = f(X_1, \ldots, X_d) \). The extension principle has also been used for solving inverse problems in [24], where \( Y = f(X) \) and where we have data for \( Y \) and require \( X \). Here, \( J_Y \) is constructed from the data, and a consistent possibility distribution,

\[
    J_X^{\text{inv}}(x) = J_Y \left( f(x_1, \ldots, x_d) \right).
\] (5)
is obtained for the unknown input $X$. With the multivariate possibility distribution $J^{inv}_X(x)$, forward uncertainty computations can take place using the machinery induced by the extension principle, also known as possibilistic calculus, and the following possibility distribution can be obtained for any other given output, $z := g(x)$:

$$r_Z(z) = \sup_{(x_1, \ldots, x_d) \in \mathbb{R}^d, \, z(g(x_1, \ldots, x_d))} J^{inv}_X(x_1, \ldots, x_d),$$

without having to worry about the aggregation operator that combines the individual marginal distributions.

2.6. Comparing evidence, possibility and p-box theory in one simple example

An example of a belief function with three focal elements is $K = \{[1, 3], [2.25, 2.75], [2, 4]\}$ and $m = \{1/3, 1/3, 1/3\}$. Two measures can be compute from the belief function, the belief given by

$$\text{Bel}(U) = \sum_{x : K \subseteq U} m(x),$$

and the plausibility given by

$$\text{Pl}(U) = \sum_{x : \text{K\cap U \not= \emptyset}} m(x).$$

The belief and plausibility can be used as bounds on an imprecisely known probability measure $P_X$

$$\text{Bel}(U) \leq P_X(U) \leq \text{Pl}(U).$$

As an example, for a set $U = [0, 1.5]$, Bel([0, 1.5]) = 0, i.e. no focal elements are subsets of $U$, and Pl([0, 1.5]) = 1/3, i.e. the focal element [1, 3] intersects with $U$. The probability measure is thus bounded by $P_X \in [0, 1/3]$. For a particular random variable, if the above bounds hold for all sets $U$, then the random variable is in the set of distributions bounded by the belief function.

Both probability boxes and possibility distributions can be regarded as special cases of belief functions. Fig. 1 shows a visualisation of three belief functions. The leftmost plot shows the belief function from the previous example, with the vertical width of each focal element corresponding to the amount of basic mass assignment of each interval. The centre plot shows a belief function which is also a p-box. A p-box can be considered a belief function if the focal elements can be ordered by $\leq$. A belief function can be converted to a p-box by computing the cumulative lower and upper probability, that is, by a cumulative sum of the belief and plausibility

$$F_X(x) = \frac{P_X([-\infty, x]) = \text{Bel}([-\infty, x])},$$

$$F_X(x) = \frac{P_X([-\infty, x]) = \text{Pl}([-\infty, x])}.$$
Consider the previous example of computing the measure on \( U = [0, 1.5] \), where the belief function gave \( P_X \in [0, 1/3] \). Performing the same computation on the p-box gives the same interval probability of \([0, 1/3]\), by computing the belief and plausibility (with the focal element \([1, 2.75]\) intersecting \( U \), and there are no subsets). Note the same interval probability can be obtained using bounds on the cdf. Performing the same calculation on the possibility distribution gives a wider interval probability of \([0, 2/3]\), as in this instance two focal elements intersect and there are no subsets. Note the same probability interval can be obtained by computing the possibility (2) and the necessity (3). I.e. \( \text{Pos}_X([0, 1.5]) = \sup_{x \in [0,1.5]} r_X(x) = 2/3 \), and \( \text{Nec}_X([0, 1.5]) = 1 - \sup_{x \in [0,1.5]} r_X(x) = 0 \). In both cases the computed interval probability is the same or wider. If we try with a different \( U \) interval, the possibility distribution performs better than the p-box. For example, choosing \( U = [2.2, 2.8] \), the belief function gives the interval probability \([1/3, 1]\), where all focal elements intersect with \( U \) and one is a subset. The p-box yields a vacuous \([0, 1]\) in this case, because no p-box focal element is contained in \( U \) and all p-box focal element intersect \( U \), while the possibility distribution returns the exact same bound as the original belief function of \([1/3, 1]\) (see Table 1).

### 2.7. Data-depth algorithm

We present a simple procedure to construct a sequence of mutually nested sets from multivariate observations. The resulting sequence of sets is the basis for constructing the desired joint belief structure. The number of overlapping sets at any given point is interpreted as the depth, which increases as the “centre” of the data is neared. Based on the shape of the constraint-type dependency that we intend to model, a set can be a (hyper-) box, a (hyper-) sphere, or a more complex shape. The easiest choice for the sake of the analysis is a box. Once the shape is chosen, the algorithm solves a so called scenario optimisation problem, to determine the set with the smallest enclosing area or volume. We refer to such smallest enclosing set as the minimal set. The optimisation is performed in such a way that the observation vectors lying on the boundary of the optimal set, are identified and indexed. These vectors restrain the set to get any smaller, thus are called support vectors. The number of support vectors depends on the shape and dimension of the set; for example, a rectangle – i.e. a box in two dimensions – can have from two to four support vectors. The number of support vectors can also increase if some relaxation is induced in the constraint satisfaction. The sequence is constructed as follows: (i) the minimal enclosing set, along with its support vectors is found; (ii) the support vectors are removed from the observations. These steps are repeated until there are no more observations left. The procedure is depicted in Fig. 2 on a two-dimensional dataset. The observations \( X_i \in \mathbb{R}^2 \) with \( i = 1, \ldots, n \) are shown in the top left scatter. The data-depth algorithm outputs two lists of the same length: (1) a list of boxes, where each box is a 2-interval for this particular example; (2) a list of sub-indices identifying the support vectors in the data set corresponding to each box.

Let us consider the observations \( X_i \in \mathbb{R}^2 \) with \( i = 1, \ldots, n \) shown in the scatter plot of Fig. 2. The rectangle with the smallest area enclosing the whole dataset \( X \) is uniquely determined by 3 support vectors. These vectors are depicted in red, see the top-right scatter plot in Fig. 2. The second rectangle, \( k = 2 \) is obtained determining the smallest box containing all data except the support vectors found at the previous step. When all the data points have been turned into support vectors the algorithm terminates. In the example of Fig. 2, the algorithm outputs 54 level sets. In this case, we say that the data set has depth 54.

### 2.8. Computing the belief measure

The data-depth algorithm outputs a list of level sets with their corresponding support vectors. This is sufficient to compute the belief measure at any given level set. The lower probability is computed counting the cumulative number of support vectors peeled away, i.e. the cumulative number of all the previously identified support scenarios plus the support scenarios at the current iteration. The (one-sided) lower bound \( p_j \) is a random variable of the probability space of the data generating mechanism that can be computed using the formula and code provided in [26]. A code for the two-sided case for both lower and upper bounds is also available, see [27]. The computation consists in finding the largest root of the explicit polynomial:

\[
\frac{\beta}{n+1} \sum_{m=0}^{n} \binom{n}{m} a^{m-j} - \binom{n}{j} a^{n-j} = 0.
\]

The solution to the polynomial (7) is unique in the interval \((0, 1)\). Under the two assumptions of uniqueness and non-degeneracy, it holds that \( p_j = a_j \), see Theorem 2 in [26]. To compute \( a_j \) a simple bisection algorithm can be used. More about the two assumptions can also be found in [26]. The positive integer \( j := \pi_k = \sum_{l=1}^{k} n_l[i] \) is the cumulative number of support vectors, and it can be interpreted in multiple ways: (i) as a parameter defining how complex is the optimisation problem; (ii) the minimum number of scenarios that are necessary to reconstruct the optimised set; (iii) an over-fitting parameter for constructing the mutually nested sets. For more on the interpretation of the lower probability \( a_j \) within scenario theory, the reader is referred to [28]. Alternative ways to compute such measures exist. For example in [29], the Clopper-Pearson confidence formula is used. This belief measure
Fig. 2. Summary of the data-depth algorithm. A set of 200 observation is generated using a stationary data generating mechanism. The observations display a v-shaped scattergram. The data-depth algorithm solves for the optimal enclosing set and outputs the index of the observations that determines the constrained optimum. These observations are also called support vectors. At $k = 1$ there are $n_s[k] = 3$ support vectors. The second step $k = 2$ consists in repeating this same procedure but on the set of observations exclusive of the previous support vectors. The procedure continues until there are no more observations to process, i.e. at step $k = 54$ where only one observation is left. The number of steps determines the number of level sets and the depth of the data.
Mechanical Systems and Signal Processing 186 (2023) 109877

M. de Angelis et al.

Fig. 3. Summary of the back propagation data-depth algorithm for the illustrative example in Section 2.9.1. The figure must be read following the arrows, from the top right where output data is displayed to bottom left where the input joint belief structure is shown. The expression of \( f: \mathbb{R}^2 \rightarrow \mathbb{R}^2 \) is 
\[
(f_1(x), f_2(x)) = (4x_1, x_1^2 + x_2),
\]
which produces a v-shaped data set when evaluated on data generated from a normal distribution. 200 iid samples of \( X_1 \) and \( X_2 \) are drawn from an independent bivariate normal distribution with zero mean vector. The results of the back propagation algorithm meet the expectation of the analyst: the marginal structure of \( X_2 \) tightly bounds the normal distribution with zero mean, while the marginal structure of \( X_1 \) displays more conservatism because of the loss of information due to the square operation in \( f \).

can then be attached to each level set to determine the height of the level sets within the consonant structure, see e.g. marginal structures in Figs. 2 and 3. In particular the height of each level (y coordinate on a Cartesian plane) is given by 
\[
1 - \alpha_j.
\]
Assigning \( \alpha_j \) as a belief measure to the nested sets produced by the data-depth algorithm, is equivalent to constructing coherent belief structures, as shown in [30].

2.9. Back propagation algorithm

The data-depth algorithm described in the previous section can be extended to solve the back propagation problem of characterising the inferential uncertainty of the inputs, given some observations of the output. In engineering, this task is also known as model calibration or improperly model updating. The aim is constructing a joint belief structure of the inputs, when only data of the output is available. The process is depicted in Fig. 3 for the simple case a two-dimensional output obtained from a two-dimensional input, \( f: \mathbb{R}^2 \rightarrow \mathbb{R}^2 \). The back propagation algorithm can be summarised in five simple steps.

1. Select a function \( f \), thus the target output response, \( Y \).
2. Obtain data depth of target response \( Y \) using the data-depth algorithm described in Section 2.7.
3. Assign a belief measure to the level sets at each depth using the direct inference algorithm as in Section 3.7.1.
4. Evaluate \( f \) on some generated coverage samples in the range of interest.
5. Index the coverage samples by their depth back in the input space, see left of Fig. 3.

In step (1), the function \( f \) must be inclusion monotonic in order for the back propagation method to work. Inclusion monotonicity ensures that the consonant structure in the output is preserved when mapped back through \( f \) to the input space. Inclusion monotonicity can be automatically checked with interval arithmetic. To conclude that a function is not inclusion monotonic, it will be sufficient to show that for any two intervals \( X \) and \( Y \), the following does not hold: \( X \subset Y \iff f(X) \subset f(Y) \). Even though examples of interval-valued functions that are not inclusion monotonic are sporadic in the applications, it is best practice to check that this
condition is not violated before the updating process commences. This property may have wider implications to the problem of identification.

Step (2) often called the direct inference step in this paper, is tasked with constructing the joint possibility distribution of the output, \( Y \). More details on this step are given in Section 2.7.

Step (3) uses scenario optimisation to assign a lower probability measure to the belief that a new observation is inside the given level set. This lower probability measure is denoted by \( a \), see Section 2.8.

Step (4) has two main functions: (i) enable non-intrusive interval arithmetic; (ii) outsource the heavy function evaluations to an independent step which relies only on the knowledge of the range of interest. With this step, the function can be evaluated “offline” without compromising the performance of the updating algorithm. In other works [31,32], this step is often taken further to use the samples to train a global interpolator like a neural network. This additional step is necessary when the function evaluations are very costly, like for finite element models. In these cases, this step should be done resorting to parallel computing. Of course, this step relies on the number of coverage samples, and on the quality of coverage, which can be improved by means of latin-hypercube algorithms. The dimensionality of the input space can dramatically affect coverage given the same number of coverage samples. The choice of the range of interest is also critical, because can affect the efficiency of the updating strategy. A possible way forward is to select a large range of interest to start with, comprising only a few samples, and then home in on a refined range until satisfactory. Eventually, the values from these multiple repeated evaluations are stored into an input–output tabular data set. Storing these values can be strategic for the efficiency of the updating algorithm, as it will be explained in the online updating section Section 3.1.

Step (5) is the direct implementation of the inversion method described in Section 2.5. This step ensures that the belief measures belonging to the output level sets are mapped back to the input level sets through \( f \). This step can be taken even further to construct soft boundaries between level sets. Nevertheless, constructing these boundaries may be expensive in high dimensional spaces, so this should only be done if needed. While non-rigorous, a characterisation of the input level sets by samples is still the most flexible option when it comes to visualisation. If a more rigorous characterisation of the shape of these sets is needed, additional computational and theoretical effort will be needed that goes beyond the scope of this paper.

A parameter updating strategy based on the back propagation algorithm described in this section is summarised in Fig. 6. There are only two decisions that the agent needs to make in order to run the algorithm: (1) the target response, used to characterise the inputs of interest, and the range of the inputs of interest. If the target is chosen incorrectly, the algorithm will return a vacuous answer, while if the range of interest is chosen too small, the resulting characterisation will be incomplete.

2.9.1. Illustrative example

The simplest example to demonstrate the back propagation algorithm is a two-input two-output function, \( f : \mathbb{R}^2 \rightarrow \mathbb{R}^2 \). The expression of \( f \) is \( f(x) = (y_1, y_2) = (4x_1^2 + x_2, x_1) \), which produces the same v-shaped data set shown in Fig. 2. First, \( n = 200 \) iid samples are drawn from a normal distribution \( x_1, x_2 \sim \mathcal{N}(0, \sigma^2_{1,2}) \), then these samples are pushed through \( f \), resulting in the scatter shown in Fig. 2. The results of the back propagation algorithm are shown in Fig. 3. All the details about this example, including the data used in Figs. 2 and 3 are available online, see pointer in Section 5.

3. Application to the digital twin of a three-storey structure

The digital twin under consideration is described in [33]. The aim of this study is to update the parameters of a three-degree-of-freedom (DOF) mass–spring–damper model, shown in Fig. 4, while characterising the inferential uncertainty. The digital twin stores a batch of \( n = 200 \) repetitions of the same hammer impact experiment. Each run of the experiment generates a time series of displacement for each mass, which are gathered using triaxial sensors placed on each of the three masses. The exact location of the sensor on the mass is irrelevant for the purpose of the study, as each metal slab can be considered as a rigid body. The time histories and the respective Fourier amplitudes are shown in Fig. 5.

3.1. Online updating

Upon excitation, the tri-axial sensors glued to the metal slabs, as shown in Fig. 4 transmit the acceleration signal to the acquisition system. The signal is digitised and integrated to produce the corresponding velocity and displacement. The resulting time histories are then stored into a database labelled by impact run number. The digital twin, described in [33], fetches the data from the database with the associated metadata reporting the run number and the conditions of the experiment, such as excitation type, excitation intensity, timestamp of data acquisition etc.

The digital twin of the three storey structure is connected to the server running the database. When a new run of the experiment is recorded, this is visible and accessible. The digital twin operational platform (DTOP) is tasked with fetching the newly recorded time histories and, in turn, deploy the updating. Because the DTOP is connected to the database server, a new run of the impact experiment is automatically converted into a new sample for the inference algorithm. Therefore, given that the inference back propagation algorithm only takes a few seconds to run, the updating procedure effectively takes place “online”. Clearly, this is true insofar as the data produced by the physical simulator are stored and available either in memory or for the DTOP to fetch from a dedicated database. Even though it takes significant computational time to produce sufficient simulated data, this can be done offline or one-off, thus, without affecting the online updating. After each updating, a snapshot of the updated parameters is timestamped and stored.
Fig. 4. Three DOF mass–spring–damper model of the three-storey aluminium structure. The motion of the three masses is horizontal. In this model, the stiffness and viscous damping are used to model the connection between the three masses.

Fig. 5. Fifty simulated scenarios: displacement time–history and corresponding Fourier amplitude for hammer-test simulated data. The data was generated with zero initial conditions, and an impact excitation of peak intensity $\hat{f} = 100$ N applied at time $t_0 = 0.01$ s and with an overall duration of $\tau = 0.005$ s. The impact force is modelled with the law: $f(t) = \hat{f} e^{\frac{(t-t_0)^2}{2\tau^2}}$, where $\hat{f}$ is the peak intensity, $t_0$ is the peak time, and $\tau$ is the duration of the impact.

3.2. Hammer-test data

Each run of the hammer test generates a displacement time history for each mass. Each time series has length 500,000 and is converted to the Fourier domain. The right side of Fig. 5 shows the Fourier amplitude for the first 251 frequencies up to 50 Hz, which will constitute the target of the inference task.

The experimental data set thus consists of a vector of Fourier amplitudes of length 251 for each mass for each run. An array of shape $(200 \times 3 \times 251)$ of floats stores all the experimental data needed to conduct this study.

The batch of 200 time histories considered in this study has been acquired with an impact hammer test simulator, which is detailed in the following two sections.

3.3. Simulated natural frequencies

The natural frequencies of the mass–spring–damper are simulated using a linear modal analysis, which consists in solving for the characteristic polynomial of the generalised eigenvalue problem:

$$K \phi_i = \lambda_i M \phi_i, \quad i = 1, 2, 3,$$

(8)

where, $M$ and $K$ are the mass and stiffness matrices, $\phi_i \in \mathbb{R}^3$ is the $i$th column eigenvector and $\lambda_i \in \mathbb{R}$ the $i$th eigenvalue. The scalar $\lambda$ and a vector $\phi$ that satisfy (8), in symbols

$$(K - \lambda M) \phi = 0.$$
which leads to the three-degree polynomial with coefficient $a_i$, $i = 0, \ldots, 3$ that is shown in Table 2:

$$a_0 + a_1 \lambda + a_2 \lambda^2 + a_3 \lambda^3 = 0.$$ (12)

This polynomial is solved numerically using matrix analysis. The first (smallest) eigenvalue corresponds to the first natural frequency and so on. The $i$th natural frequency $f_i$, is related to the $i$th eigenvalue $\lambda_i$ as follows: $f_i = \frac{\sqrt{\lambda_i}}{2\pi}$.

### 3.4. Deterministic impact simulator

The impact force is applied to the first mass of the mass–spring–damper systems. The coupled second-order ordinary differential equations yielding the velocity and displacement of the masses are,

$m_1 \ddot{x}_1 + c_1 \dot{x}_1 + c_2 (\dot{x}_1 - \dot{x}_2) + k_1 x_1 + k_2 (x_1 - x_2) = f(t)$

$m_2 \ddot{x}_2 + c_2 (\dot{x}_2 - \dot{x}_1) + c_3 (\dot{x}_2 - \dot{x}_3) + k_2 (x_2 - x_1) + k_3 (x_2 - x_3) = 0$

$m_3 \ddot{x}_3 + c_3 (\dot{x}_3 - \dot{x}_2) + k_3 (x_3 - x_2) = 0.$

Keeping only the second order terms to the left-hand side of the equations, and dividing by the mass, the following holds,

$$\ddot{x}_i = \frac{1}{m_i} (-c_i \dot{x}_i - c_i (\dot{x}_1 - \dot{x}_2) - k_i x_1 - k_i (x_1 - x_2) + f(t))$$

<table>
<thead>
<tr>
<th>Table 2</th>
<th>Coefficients of the characteristic polynomial.</th>
</tr>
</thead>
<tbody>
<tr>
<td>$a_0$</td>
<td>$k_1 k_2 k_3$</td>
</tr>
<tr>
<td>$a_1$</td>
<td>$-m_1 k_2 k_3 - m_2 (k_1 k_3 + k_2 k_3)$</td>
</tr>
<tr>
<td>$a_2$</td>
<td>$m_1 m_2 k_3 + m_2 m_3 (k_1 + k_2) + m_1 m_3 (k_1 + k_3)$</td>
</tr>
<tr>
<td>$a_3$</td>
<td>$-m_1 m_2 m_3$</td>
</tr>
</tbody>
</table>

Fig. 6. Overview of the updating strategy. Choosing the range of interest for each of the updating parameters is the first step. The analyst should choose a box that is large enough to include the updated parameters. After the largest box is fixed, a large number of coverage samples is generated; if the dimension is low, these would ideally be samples arranged on a regular grid. The physics solver is then called on each of these samples and the results stored in a table of simulated data. The simulated data is then compared against the experimental data using the back propagation data-depth algorithm. The algorithm outputs the depth of the experimental data, and the corresponding structure of level sets of the updated parameters with the desired confidence.
\[
\begin{align*}
\ddot{x}_2 &= \frac{1}{m_2} \left( -c_2 (\ddot{x}_2 - \dot{x}_1) - c_3 (\ddot{x}_2 - \dot{x}_3) - k_2 (x_2 - x_1) - k_3 (x_2 - x_3) \right) \\
\ddot{x}_3 &= \frac{1}{m_3} \left( -c_3 (\ddot{x}_3 - \dot{x}_2) - k_3 (x_3 - x_2) \right). 
\end{align*}
\]

With the following change of variables \( v = \dot{x} , \ u = x \), it is immediate to see the equivalent first order system of differential equations
\[
\begin{align*}
\dot{u}_i &= v_i , \quad i = 1, 2, 3 \\
\dot{v}_1 &= \frac{1}{m_1} \left( -c_1 v_1 - c_2 (v_1 - v_2) - k_1 u_1 - k_2 (u_1 - u_2) + f(t) \right) \\
\dot{v}_2 &= \frac{1}{m_2} \left( -c_2 v_2 - c_3 (v_2 - v_3) - k_2 (u_2 - u_1) - k_3 (u_2 - u_3) \right) \\
\dot{v}_3 &= \frac{1}{m_3} \left( -c_3 v_3 - k_3 (u_3 - u_2) \right).
\end{align*}
\]

The above system of ODEs can be solved in \( u, v \) by any step-wise numerical integrator. Collecting the six unknowns in \( u = (u_1, u_2, u_3, v_1, v_2, v_3) \), the system of ODEs can be written in compact form as \( \dot{u} = Au \), where \( A \) is a \( 6 \times 6 \) matrix with the mass, stiffness, and damping coefficients.

The impact force is modelled by a smooth exponential form: \( f(t) = f(\hat{\tau}) \ e^{\frac{u - \hat{\tau}}{\tau}} \), where \( f \) is the peak intensity, \( \hat{\tau} \) is the peak time, and \( \tau \) is the duration of the impact.

The code implementing the impact simulator is available in the digital twin operational platform [33].

3.5. Goal of the updating task

The algorithm is tasked with updating the stiffness vector \( k = (k_1, k_2, k_3) \in \mathbb{R}_+^3 \) and the damping vector \( c = (c_1, c_2, c_3) \in \mathbb{R}_+^3 \) of the mass–spring–damper system. Note that solving this inference task is equivalent to taking indirect measurements of the stiffness and damping. The mass vector \( m = (m_1, m_2, m_3) \in \mathbb{R}_+^3 \) has been directly measured by means of multiple scales before assembly, thus it will not be targeted by the updating algorithm.

3.6. Updating strategy

It is reasonable to assume linearity of the structural response, given the simple structure, and the relatively low magnitude of the excitation, which rules out geometrical nonlinearities. Note that linearity here need not mean that the response is linear with respect to the updating parameters. It rather means that the governing ordinary differential equations are linear, see Section 3.4. Because the mass has been measured directly, the updating strategy will: (1) update the stiffness vectors by targeting the natural frequencies of the system; (2) update the damping vector by targeting the Fourier amplitude spectrum. It is reasonable to assume that, due to degradation or damage, only the stiffness and the damping of the system will need to be updated. Note, however, that the proposed strategy quantifies the inferential uncertainty, not the model uncertainty. If the damage is such that the three DOF model is no longer suited, a new model of the physical system will be needed. Even though the updating strategy does not give a quantitative indication of the model uncertainty, it can still be used to detect a change in the model. For example, if the three DOF model is no longer suited, the updating will output an interval, i.e. a vacuous belief structure.

An overview of the strategy comprising each major step is given in Fig. 6. The strategy relies on the analyst to choose a range of interest that is large enough to include the final updated parameter, and to choose the number of coverage samples. The latter must be chosen so to strike a balance between coverage and simulation cost. The simulation samples are then passed through the physics solver to output harmonic amplitudes and natural frequencies. These constitute the simulated data. The empirical data also in the form of harmonic amplitudes and natural frequencies is then compared against the simulated data using the data-depth algorithm. Eventually, the back propagation algorithm makes use of the inference information to construct the consonant structures for the updated parameters.

The updating strategy specialised to this particular case study is shown in Fig. 7. Only the stiffness and damping vectors will be updated. In Fig. 7, the boxes with smooth corners (orange) denote an action, while all the other boxes denote data. The integer \( n \) denote the number of coverage samples used in the updating, which is typically large, \( n = 10^6 \). In this case study, the experimental data is composed of 200 time histories of displacement with 500000 time points for each of the three floors. The dynamic response displays three main resonance frequencies, while the frequency spectrum of Fourier amplitudes shows a compression up to the first 251 frequencies.

3.7. Updating the stiffness vector

The stiffness vector is updated targeting the natural frequencies, see the left side of the diagram in Fig. 7. So the stiffness vector is mapped onto the natural frequencies by \( f : \mathbb{R}_+^3 \rightarrow \mathbb{R}_+^3 \), where \( f \) is computed numerically via matrix analysis to extract the roots of the polynomial in Eq. (12). The updating of the stiffness vector comprises two stages: (1) the deployment of the direct inference algorithm on the three natural frequencies, (2) the deployment of the back propagation algorithm to yield the updated parameters.
Fig. 7. Updating strategy specialised to the case study. The mass is known, while the stiffness and damping vectors will be updated. The boxes with smooth corners (grey) denote an action, while all the other boxes denote data. The integer \( n \) denote the number of coverage samples used in the updating, which is typically large, \( n = 10^6 \). In this case study, the experimental data is composed of 200 time histories of displacement with 500,000 time points for each of the 3 floors. The dynamic response consists of three main resonance frequencies used in the updating of the stiffness, and three peak Fourier amplitudes used in the updating of the damping.

3.7.1. Deployment of the direct inference algorithm on natural frequencies

The direct inference algorithm is tasked with constructing the sequence of nested sets and assign to these sets a belief measure, while targeting the natural frequencies of the 3 DOF system. The results of running the algorithm with boxes on the 200 experimental natural frequencies are shown in Fig. 8. 35 levels were found. The depth of the data set can be better appreciated in the marginal structures on the diagonal of Fig. 8. The belief measure is computed as per Section 2.8 with a confidence parameter \( \beta = 0.01 \). The absolute height \( 1 - \alpha \) of each level set, appreciable on the marginal structures, is the upper probability of observing a new datum outside that particular level set. On the other hand, the belief measure \( \alpha \) is the lower probability of observing a new datum in that particular level set. This absolute height is determined by the cumulative number of support vectors. The number of support vectors for a box in two dimension can range from two to four; this explains the irregular increments shown on the marginal structures. For this particular example, the probability of a new observation to fall in the largest set is at least \( \alpha_1 = 0.9123 \). Equivalently, for the other sets: the probability of a new observation to fall in the \( i \)th set, with \( i = 1, \ldots, 35 \) is at least \( \alpha_i \). Note that, however, nothing can be said about the upper bound of such probability. Nevertheless, if an upper bound is needed, this can computed on an individual basis using a two-sided confidence distribution.

3.7.2. Deployment of the back propagation algorithm on the stiffness vector

The back propagation algorithm is deployed using \( 10^6 \) coverage samples, which were obtained offline running \( 10^6 \) times the eigenvalue solver of Section 3.3. The coverage samples are subsequently indexed following the procedure described in Section 2.9. The results of the back propagation are shown in Fig. 9. The joint possibility structure is composed of 35 nested level sets, whose shape encodes the functional dependence induced by the eigenvalue solver. In Fig. 9 this dependence can be visually appreciated on the bivariate plots shown off the diagonal. The sharp edges of the level sets in the bivariate projection may be an artefact due to the choice of boxes for the direct inference in Section 3.7.1. The plots on the diagonal show the marginal possibility distributions, where the belief measure can be appreciated.

The probability to see a new observation outside these structures, i.e. in the grey region of the bivariate plots, is bounded by the interval \([0, 1 - \alpha_1]\) with a desired level of confidence. For this particular case study, the confidence is set to \( 1 - \beta = 0.99 \). The first level set has a belief (necessity) measure of \( \alpha_1 = 0.9123 \), which means that the probability of observing a new datum inside such level set is at least 0.9123. On the other hand, the probability of observing a new datum outside the first level set is at most \( 1 - \alpha_1 = 1 - 0.9123 = 0.0877 \). At the second level set the belief measure decreases to \( \alpha_2 = 0.8879 \), and it keeps decreasing until
Fig. 8. The target response in terms of natural frequencies for the updating of the stiffness vector. The plots off the diagonal show the nested boxes produced by the data-depth algorithm projected onto each pair of coordinate axis, as well as the experimental data as a scatter plot. The plots on the diagonal show the boxes projected onto each individual axis. The fuzzy-looking structures on the diagonal are obtained assigning to each level a belief measure \( \alpha \), as explained in Section 2.8. The absolute height of each step, given by \( 1 - \alpha \), is the upper probability of observing a new datum outside that particular box, which is also known as membership. Contrarily, the belief measure \( \alpha \) is the lower probability of observing a new datum inside that particular box. The belief structures were obtained setting the following confidence \( \beta = 0.01 \).

The last level set whose belief measure is \( \alpha_{35} = 0 \). The number on the y-axis of the marginal distribution, \( 1 - \alpha \), is the plausibility (possibility) distribution of (2). This can be interpreted as an upper bound on the probability of obtaining a new observation outside any particular level set. For example, the probability of a new observation falling outside the last level set is at most 1, which need not mean it is exactly 1.

3.8. Updating the damping vector

The damping vector is updated targeting the Fourier amplitude of the displacement of the three masses in correspondence of the three natural frequencies. The results of running the data-depth inference algorithm on the whole target, comprising natural frequencies and Fourier amplitudes, are shown in Fig. 10.
While the damping has different intensities at each connection, with the largest being the damping $c_1$ connecting the first mass to the ground, it is reasonable to assume that all three parameters $c_1, c_2, c_3$ share the same uncertainty space. Stated another way, it will be assumed that the damping coefficients are fully and positively dependent. This drastically reduces the dimensionality of the inference problem, from six to four uncertain variables; one variable for the damping and three for the stiffness vector. With this modelling choice, the dimensionality of the target can also be reduced. Thus, for the updating of the damping, additionally to the three natural frequencies, only the Fourier amplitude of the first natural frequency $a_1$ will be considered. The stiffness vector must be updated jointly with the damping vector.

The results of the updating are shown in Fig. 11. The Figure does not show the joint possibility distribution because of the modelling choice to consider of positive perfect dependence between $c_1$. The asymmetric shape of the structures in Fig. 11 may be artefactual due to the sparsity of the coverage samples in the deepest level sets. The updated parameters, unlike the stiffness, do not

Fig. 9. Updated stiffness vector. The range of interest is $[[31000, 43000], [28000, 50000], [35000, 57000]]$. The plots off the diagonal show the coverage samples colour-coded according to each level. The colours display a recurrent pattern that reflects the structure of the marginal numbers, which are shown on the plots on the diagonal. The grey coverage samples (also called background samples) corresponds to those input combinations whose output response does not fall in the largest box, so they are assigned belief measure equal to zero. The probability to observe a new datum in the grey region is bounded by the interval $[0, 1 - \alpha_1]$ with confidence $1 - \beta = 0.99$, where $1 - \alpha_1 = 0.0877$ is the belief measure assigned to the largest set at the bottom of the pyramidal structure. The probability of a new datum falling in the first (largest) set is at least $\alpha_1 = 0.9123$, but nothing can be said about its upper bound. Equivalently for the other sets: the probability of a new datum falling in the $i$th set, with $i = 1, \ldots, 35$, is at least $\alpha_i$, but nothing can be said about its upper bound.
Fig. 10. Full target response in terms of natural frequencies and Fourier amplitudes for the updating of the damping vector. The plots on the diagonal show the marginal consonant structures for the three natural frequencies $f_i$ and for the three Fourier amplitudes of the displacement in correspondence of these natural frequencies $a_i$, with $i = 1, 2, 3$. The plots off the diagonal show the boxes that the data-depth algorithm outputs, projected onto pairs of coordinate axes. The empirical data are depicted in the scattergram of squared dots. As expected, natural frequencies and amplitudes with the same index show some pronounced functional dependence.

display an evident gradation of the size of the level sets. This is to be attributed to the low sensitivity that the damping parameters have with respect to the Fourier amplitudes in the given range of variability.

4. Conclusions

The pressing need to update “online” the physical models within a digital twin is addressed in this paper. In order to be considered online, the updating algorithm must run in seconds if not milliseconds. The digital twin of a three-storey structure has been used to illustrate the strategy. Such digital twin can be accessed via browser on the Digital Twin Operational Platform. The platform stores simulated and empirical data collected by means of tri axial accelerometers. While a digital twin can house models of multiple fidelity, a three DOF system was used to model the dynamics of the structure. Nevertheless, the presented updating strategy is not tied to any particular modelling choice, and can apply to other types of models, including finite element models.
Fig. 11. Updated damping vector. The range of interest whereby the updating took place is $[3, 12], [1, 10], [1, 10]$. This range of interest is larger than the base level found by the algorithm, which confirms that the updating was successful. Only the marginal consonant structures are shown because of the modelling choice of considering the damping parameters perfectly positively dependent. The updated parameters, unlike the stiffness, do not display an evident gradation of the size of the level sets. This is due to the low sensitivity that the damping parameters have with respect to the Fourier amplitudes in the given range of variability.

The presented data-depth algorithm updates the parameters of the physical model solving an inverse problem. Thanks to the back propagation strategy the algorithm can update in a few seconds, without the need to re-run the model. All the model runs necessary for the updating are done in advance, offline, and in parallel if needed. The strategy works well in the context of stationary response and negligible model uncertainty, i.e. during the ordinary operation of the digital twin. The updating strategy should also be able to detect changes in the model physics, if the parameters get increasingly more uncertain as more data is gathered. This feature can be very useful during the operation of the digital twin, because it can prompt the deployment of algorithms for model selection.

Lastly, the whole updating strategy has been cast within the general framework of imprecise probability. In contrast with the more popular Bayesian counterpart, imprecise probability can be deployed to unlock the potential to operate without prior nor likelihood assumptions, as it has been demonstrated throughout the paper. The absence of prior and likelihood assumptions has meant that the updating strategy is robust against the inferential uncertainty, so it can be used on small sample sizes. The inferential method is designed to produce an uncertainty structure that is conservative in the sense that it provides an outer approximation of the credal set whereby the stationary unknown probability distribution resides. Moreover, the choice of a possibility distribution for the uncertainty structure has meant that additional forward uncertainty propagation can take place non-intrusively after the updating. All the computations that follow the updating can be conducted with a desired level of confidence.

5. Reproducibility

The code running the data-depth algorithm and the numerical examples lives at this repository: https://github.com/FnPIG/data-depth-inference (last accessed July 2022). The code in relation to the three-storey structure, including: the hammer-test simulator, the code to calculate the natural frequencies, the code to compute the power spectral density, instead lives in the Digital Twin Operational Platform at: https://github.com/Digital-Twin-Operational-Platform/Cristallo (last accessed July 2022).

Declaration of competing interest

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

Data availability

Data will be made available on request.
Acknowledgements

We are very grateful to those who have attended the virtual Fuzzy and Possibility interest group (FnPig), hosted by the Institute for Risk and Uncertainty of the University of Liverpool, for the many fruitful discussions which led to this paper. Thanks to Roberto Rocchetta for sharing his knowledge of scenario optimisation, and David Wagg for the code running the hammer-test simulator. Thanks to the anonymous reviewers for their constructive comments.

The authors would like to acknowledge the support of the EPSRC, United Kingdom programme grant “Digital twins for improved dynamic design”, under grant number EP/R006768/1.

References