Towards the Development of an Operational Digital Twin

Paul Gardner [1,*] , **Mattia Dal Borgo** [2] , **Valentina Ruffini** [3] , **Aidan J. Hughes** [1] , **Yichen Zhu** [1] and **David J. Wagg** [1]

[1] Dynamics Research Group, Department of Mechanical Engineering, University of Sheffield, Sheffield S1 3JD, UK; ajhughes2@sheffield.ac.uk (A.J.H.); yichen.zhu@sheffield.ac.uk (Y.Z.); david.wagg@sheffield.ac.uk (D.J.W.)

[2] Institute of Sound and Vibration Research, University of Southampton, Southampton SO17 1BJ, UK; m.dal-borgo@soton.ac.uk

[3] Department of Mechanical Engineering, University of Bristol, Bristol BS8 1TR, UK; v.ruffini@bristol.ac.uk

[*] Correspondence: p.gardner@sheffield.ac.uk

Abstract: A digital twin is a powerful new concept in computational modelling that aims to produce a one-to-one mapping of a physical structure, operating in a specific context, into the digital domain. The development of a digital twin provides clear benefits in improved predictive performance and in aiding robust decision making for operators and asset managers. One key feature of a digital twin is the ability to improve the predictive performance over time, via improvements of the digital twin. An important secondary function is the ability to inform the user when predictive performance will be poor. If regions of poor performance are identified, the digital twin must offer a course of action for improving its predictive capabilities. In this paper three sources of improvement are investigated; (i) better estimates of the model parameters, (ii) adding/updating a data-based component to model unknown physics, and (iii) the addition of more physics-based modelling into the digital twin. These three courses of actions (along with taking no further action) are investigated through a probabilistic modelling approach, where the confidence of the current digital twin is used to inform when an action is required. In addition to addressing how a digital twin targets improvement in predictive performance, this paper also considers the implications of utilising a digital twin in a control context, particularly when the digital twin identifies poor performance of the underlying modelling assumptions. The framework is applied to a three-storey shear structure, where the objective is to construct a digital twin that predicts the acceleration response at each of the three floors given an unknown (and hence, unmodelled) structural state, caused by a contact nonlinearity between the upper two floors. This is intended to represent a realistic challenge for a digital twin, the case where the physical twin will degrade with age and the digital twin will have to make predictions in the presence of unforeseen physics at the time of the original model development phase.

Keywords: digital twin; data-driven modelling; machine learning; validation; active learning; hybrid testing; active control

1. Introduction

A digital twin is a virtual duplicate of a physical system (called the physical twin). This powerful new concept can enable computational models to capture the dynamic behaviour of the physical twin with a greater degree of accuracy than previously possible. This is because they combine the best aspects

of both data-based and physics-based models. As will be described below, this type of combination can allow a digital twin to provide improved predictive performance and support robust decision making for operators and asset managers.

The idea of a digital twin arose from work relating to product lifecycle management—the interested reader can find full details in the review articles [1–3] and references therein. Since its inception, the meaning of the term digital twin has been widely interpreted, and as a result the details of the precise meaning depend on the specific context involved. In terms of asset management (the context of this paper) digital twins have been considered for tasks such as damage detection and structural-health/condition monitoring [4–7], predictive maintenance [8–11], and uncertainty quantification [12–14]. In this context the key application areas are offshore oil & gas [15,16], offshore wind turbines [17–20], aerospace [21–23] and nuclear fusion [24–26].

The conceptual framework used here for a digital twin is shown in Figure 1. In this framework there are four fundamental components:

- Models: computational models based on physical reasoning;
- Data: both quantitive and qualitative sets of information from the physical twin;
- Knowledge: both in-depth expert understanding and context specific detail;
- Connectivity: time evolving digital and organisational interactions that are free from significant interruptions or other barriers.
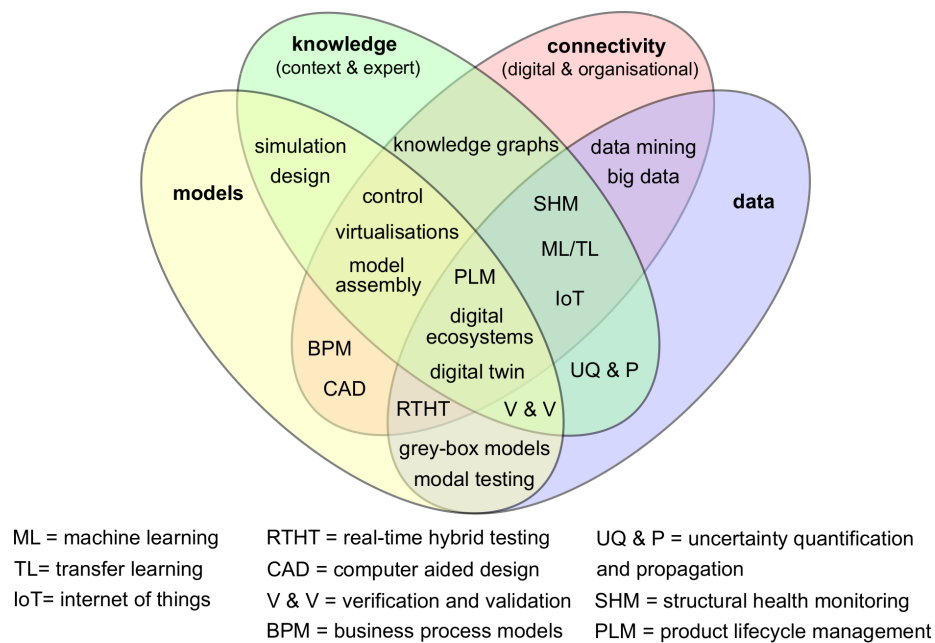

ML = machine learning	RTHT = real-time hybrid testing	UQ & P = uncertainty quantification
TL= transfer learning	CAD = computer aided design	and propagation
IoT= internet of things	V & V = verification and validation	SHM = structural health monitoring
	BPM = business process models	PLM = product lifecycle management

Figure 1. Diagram of the interactions between subject disciplines and technologies that form a digital twin.

As shown in Figure 1, if all of these components are present then it is possible to create a digital twin—shown at the intersection of all four fundamental components. Shared also in this intersection space are the concepts of product lifecycle management (the precursor to digital twin) and digital ecosystems, which is the idea of sets of inter-connected digital twins. The other elements shown in Figure 1 at the intersection of two and three of the fundamental components are the required building blocks for a digital twin.

A key point for this current study is that digital twins are more than just validated models. This is clearly denoted in Figure 1 where validated models are the intersection of 'models', 'data' and 'knowledge'. For a validated model to be 'evolved' into a digital twin it requires some form of connectivity and knowledge.

In this paper a validated model for asset management will be used as a building block for demonstrating the progression towards a digital twin. The objective will be to improve and optimise the predictive performance of the digital twin as an asset management tool. This will be achieved by selecting between options at each decision time. Specifically the choices will be: do nothing, recalibrate, add/improve data-based model, or add physics. These actions form mechanisms that allow the digital twin to 'learn' over time; where automating these actions will be part of a digital twin framework. The paper proposes a digital twin model structure, combining physics- and data-based modelling, that is utilised in generating an active learning procedure, whereby the digital twin can be updated based on its predictive uncertainty. The active learning technique can detect when structural changes have occurred, and coupled with hybrid testing, can be used to isolate and identify missing physics in the digital twin model.

The case study considered in this paper involves a three storey building structure with an unknown (and hence, unmodelled) structural state, caused by a contact nonlinearity between the second and third floors. A digital twin is constructed that seeks to predict the acceleration response of each floor, compensating for missing physics, and adapting throughout the physical twins lifespan. The impact of a 'learning' digital twin on model-based control is subsequently discussed, particularly focusing on the digital twin's ability to inform when assumptions in the control strategy are broken, as this could lead to a degraded vibration attenuation performance.

The structure of the paper is as follows. Section 2 introduces the case study, discussing the initial validated model and stating the digital twin model structure. Section 3 discusses problems associated with only using model updating as a 'learning' strategy. Following from this discussion the physics-based model is augmented with a data-based model in Section 4. The data-based model is incorporated into an active learning strategy in Section 4.3, demonstrating that the digital twin can adapt to unseen structural states. The use of real-time hybrid testing, as a method for isolating and identifying missing physics unmodelled in previous sections, is outlined in Section 5. In Section 6 a discussion is provided on the impact of the digital twin on model-based control, particularly considering a Linear Quadratic Regular (LQR) control scheme. Following this work, a discussion is provided in Section 7, outlining the progress made towards realising a digital twin and the remaining challenges before conclusions are stated in Section 8.

2. Overview of the Digital Twin

The aim of a digital twin is to provide a predictive tool that accurately estimates specific quantities of interest throughout the operational life of a structure. In order to remain accurate for the complete deployment period, the digital twin must make decisions that overcome any deviations in prediction from the actual response of the structure. These deviations in performance will occur for several reasons, e.g., numerical approximations required to solve differential equations, lack of knowledge on deployment of the digital twin of *all physics* affecting the physical twin, unknown future loading histories etc. A digital twin must therefore be able to 'learn', adapting over time such that it remains informative and up-to-date from an operational context. This section outlines a case study in order to motivate the challenges in constructing a digital twin, highlighting potential solutions and technologies that can be used to identify, and overcome, predictive difficulties. The implication of these technologies is subsequently discussed for an active control context.

The case study considers an experimental, aluminium, three storey shear structure (the physical twin), where the digital twin's aim is to predict the acceleration response of the three floors throughout the operational phase. A schematic of the structure is depicted in Figure 2, detailing the contact mechanism, formed from an aluminium block and aluminium column located between the second and third floors. This mechanism forms a contact nonlinearity when specific initial conditions and excitation are applied.

The dimensions and masses of the components are as follows: each floor was $350 \times 255 \times 5$ mm (length $\times$ width $\times$ height) with a mass of 5.2 kg, the vertical columns were $555 \times 25 \times 1.5$ mm with a mass of 55 g, and the attachment blocks were $25 \times 25 \times 1.5$ mm with a mass of 18 g.
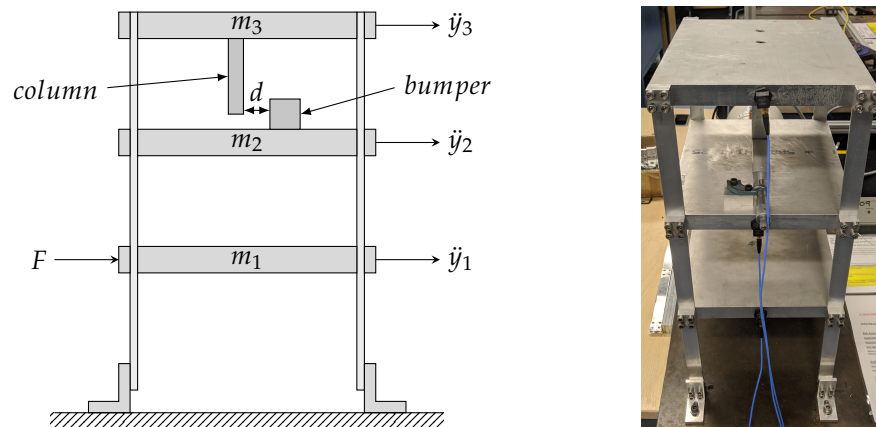

Figure 2. Three storey shear structure (physical twin). Left panel depicts a schematic detailing the contact mechanism, input excitation location and accelerometer locations. Right panel shows the experimental setup.

In order to demonstrate the challenges of an operational digital twin, an imaginary scenario is considered where the physical twin is designed to operate under an arbitrary excitation, close to a band-limited white noise at a constant forcing. However, in the design and construction phase of the physical twin it is assumed that the column and bumper do not come into contact, and therefore the structure can be assumed and modelled as a linear system. Initial data from the structure confirmed this modelling decision, with the contact mechanism remaining inactive. This represents common practice within industrial settings, where an engineering modelling team assume, based on the available evidence and performance of the model on a validation dataset, that a simpler linear computer model is satisfactory for this scenario (especially given the extra cost in developing and validating a nonlinear model, as well as the computational expense involved in solving the model). As the linear model has passed validation procedures it is deployed in operation. At some point during operation the excitation causes a large enough difference in displacement between the second and third floors that the column and bumper to come into contact, creating a harsh nonlinearity. This reflects real world scenarios whereby unforeseen behaviour, not initially captured in the computer model, reduces predictive performance. A digital twin must therefore adapt when these challenges arise, both informing the engineer of poor predictive performance, and modifying the underlying model structures such that accurate predictions are maintained. This is especially crucial when the digital twin is used to inform other processes, such as active control.

It is noted that the physics-based models investigated in this paper do not represent the most optimal physics-based modelling strategies for modelling this particular structure, or the contact problem, as their purpose is to illustrate ideas about learning for a digital twin. In fact, it is useful, for illustrating the power of augmenting the physics-based model with a data-based model, that the physics-based model contains

some modelling errors (as typically all physics about a structure is imprecise in real world applications). In industrial scenarios it will always be advisable to use all the available engineering knowledge in selecting the appropriate physics-based model. For example, in this case study a more appropriate model may be a non-smooth state-space model, such as a piecewise-linear model [27], or changing from a shear-type modelling assumption to something that better accounts for flexible joints in lightweight structures [28,29] etc.

2.1. Experimental Data

Experimental data was collected in order to demonstrate the scenario described in Section 2. The physical twin (Figure 2) was subject to a 25.6 Hz band-limited arbitrary forcing *F* applied to the first floor by an electrodynamic shaker. The acceleration responses $\{\ddot{y}_i\}_{i=1}^3$, (the set of three accelerations where the *i*th index relates to each floor) were measured at each floor using accelerometers; data was recorded at a sampling frequency of 51.2 Hz. Three 20 s datasets were obtained; two where the column and bumper *did not* (Italic is used to clarify the difference in the data and used for emphasis.)come into contact (datasets one and two), and one dataset where contact *did* occur (dataset three). In the imagined scenario, the datasets arrive in this order.

It is noted that in dataset three the bumper and column are not in contact throughout the 20 s duration. This means that dataset three is a mixture of the engaged and disengaged contact states, with the contact mechanism, on the whole, being disengaged for a longer time than instances where it is engaged. An investigation into the type of contact physics that occurred in dataset three was also not explored, given the aims of the paper, which were to illustrate how a digital twin model could learn during the operational phase of a structure.

2.2. Initial Validated Model

The initial modelling was performed by analysing dataset one, where the column and bumper *did not* come into contact, and therefore the nonlinearity was not observed. Given the frequency response functions for this dataset, presented in Figure 3, a three degree-of-freedom model structure was selected,

$$
\begin{aligned}
\ddot{y}_1 &= (F - k_1 y_1 - k_2(y_1 - y_2) - c_1\dot{y}_1 - c_2(\dot{y}_1 - \dot{y}_2))/m_1 \\
\ddot{y}_2 &= (k_2(y_1 - y_2) - k_3(y_2 - y_3) + c_2(\dot{y}_1 - \dot{y}_2) - c_3(\dot{y}_2 - \dot{y}_3))/m_2 \\
\ddot{y}_3 &= (k_3(y_2 - y_3) + c_3(\dot{y}_2 - \dot{y}_3))/m_3
\end{aligned}
\tag{1}
$$

where the mass $\{m_i\}_{i=1}^3$, damping $\{c_i\}_{i=1}^3$ and stiffness coefficients $\{k_i\}_{i=1}^3$ are specified for each of the three floors (indexed by *i*). The force, displacement, velocity and acceleration terms are denoted as, F, $\{y_i\}_{i=1}^3$, $\{\dot{y}_i\}_{i=1}^3$ and $\{\ddot{y}_i\}_{i=1}^3$, respectively. The ordinary differential equations were numerically approximated using a fourth order Runge-Kutta scheme. It is noted that although in this case study the physics-based model is analytical, the principles and techniques discussed are applicable to more complex model forms, such as finite element (FE) or multi-physics models.
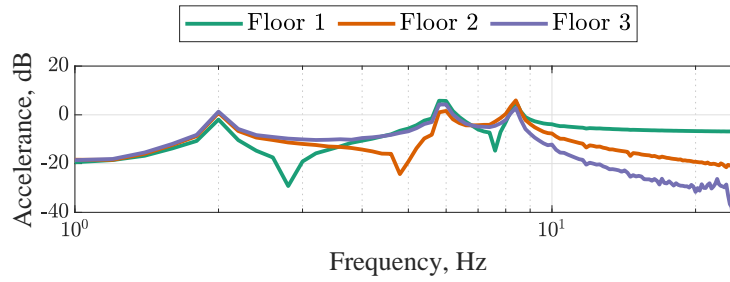
Figure 3. Frequency response functions between the first floor and acceleration at each floor for dataset one, where the column and bumper did not come into contact.

The parameters $\boldsymbol{\theta} = \{m_i, c_i, k_i\}_{i=1}^3$ of the linear model $\mathcal{M}^l(F, \boldsymbol{\theta})$ (Equation (1)) were estimated along with independent measurement noise $\Sigma_n = \{\sigma_{n,i}^2\}_{i=1}^3$ for each floor, via a maximum a posteriori (MAP) approach, i.e., $\boldsymbol{\theta}^{MAP} = \arg\max p(\mathcal{D} \mid \{\boldsymbol{\theta}, \Sigma_n\}) p(\{\boldsymbol{\theta}, \Sigma_n\})$, where $\mathcal{D} = \{\ddot{\boldsymbol{y}}_i\}_{i=1}^3$ are a set of N training observations, $p(\mathcal{D} \mid \{\boldsymbol{\theta}, \Sigma_n\})$ is the likelihood and $p(\{\boldsymbol{\theta}, \Sigma_n\})$ the prior. A joint Gaussian likelihood and prior distribution for the parameters $\{\boldsymbol{\theta}, \Sigma_n\}$ were defined as,

$$\log(p(\mathcal{D} \mid \{\boldsymbol{\theta}, \Sigma_n\})) = \sum_{i=1}^3 \frac{N}{2} \log \sigma_{n,i}^2 + \frac{N}{2} \log 2\pi + \frac{1}{2} \frac{\sum_{j=N_t}^N (\ddot{y}_{i,j} - \mathcal{M}_i^l(\boldsymbol{\theta}, F_j))}{\sigma_{n,i}^2} \tag{2}$$

$$p(\{\boldsymbol{\theta}, \Sigma_n\}) = \prod_i^3 \mathcal{N}(m_i \mid \mu_i^m, \sigma_i^m) \mathcal{N}(\log c_i \mid \mu_i^c, \sigma_i^c) \mathcal{N}(k_i \mid \mu_i^k, \sigma_i^k) \mathcal{N}(\log \sigma_{n,i}^2 \mid \mu_i^n, \sigma_i^n) \tag{3}$$

where N_t defines a cut-off, such that the transient is removed when comparing the model and measured accelerations (given unknown initial conditions). The priors are parametrised by a mean $\{\mu_i^m, \mu_i^c, \mu_i^k, \mu_i^n\}_{i=1}^3$ and standard deviation $\{\sigma_i^m, \sigma_i^c, \sigma_i^k, \sigma_i^n\}_{i=1}^3$ where superscripts m, c and k denote mass, damping and stiffness coefficients, and n denotes the noise variance. It is noted that the priors for damping coefficients c and noise variance σ_n^2 are defined in the log space such that these values remain positive (and therefore physical).

The parameters for the prior distributions were defined based on engineering judgement, shown in Table 1. The prior mass parameters were selected based on the measured mass of the floors. Due to the thin vertical column thickness, and known difficulty in identifying damping for real structures, the prior for the damping coefficients were selected to reflect a lightly damped structure with large uncertainty. The prior parameters for the stiffness coefficients were estimated based on the vertical column acting as a combination of cantilever and pinned-pinned beams between the floors. Finally, the prior parameters for the noise variances were chosen based on the variance of the measured accelerations.

Table 1. Parameters of prior distributions, and linear model parameter MAP estimates from dataset one $\mathcal{D}_1$ and dataset three $\mathcal{D}_3$. * estimated $c_3 = 2 \times 10^{-9}$ Ns/m; † estimated as $c_2 = c_3 \approx 0$ Ns/m.

	Masses, m kg	Damping Coefficients, c Ns/m	Stiffness Coefficients, k N/m	Signal Noises, σ_n^2 (g/N)^2
μ	$\{5.200, 5.200, 5.200\}$	$\{-4.605, -4.605, -4.605\}$	$\{4021, 4021, 4021\}$	$\{1.022, -0.359, -1.004\}$
σ^2	$\{0.5, 0.5, 0.5\}$	$\{1, 1, 1\}$	$\{10{,}000, 10{,}000, 10{,}000\}$	$\{0.1, 0.1, 0.1\}$
$\boldsymbol{\theta}_{\mathcal{D}1}^{MAP}$	$\{4.908, 5.577, 5.187\}$	$\{3.307, 0.002, 0.000^*\}$	$\{4051, 4320, 4964\}$	$\{0.006, 0.106, 0.031\}$
$\boldsymbol{\theta}_{\mathcal{D}3}^{MAP}$	$\{5.187, 5.281, 5.292\}$	$\{0.198, 0.000^\dagger, 0.000^\dagger\}$	$\{4263, 4332, 5263\}$	$\{0.223, 0.281, 0.165\}$

The *MAP* estimates for dataset one, $\boldsymbol{\theta}_{\mathcal{D}1}^{MAP}$, are outlined in Table 1 (where $N_t = 512$), and where the normalised mean squared errors (NMSEs) were $\{0.204, 1.655, 1.599\}$ for each floor, respectively. The linear

model was subsequently applied to dataset two (as an independent validation dataset), where the NMSEs were $\{0.260, 2.438, 2.938\}$, respectively, indicating that the model has generalised well and can be assumed valid. Figure 4 shows a visualisation of the predictions, which appear in very good agreement, with a low residual between the linear model and measured data. Given this analysis the linear model would be deemed fit for purpose and deployed. However, the last column of Figure 4 displays the model predictions on dataset three, where the model fails to accurately predict the behaviour of the system (due to the nonlinearity). In fact the NMSEs, when the linear model is applied to dataset three, increase to $\{34.595, 65.310, 42.214\}$, which, as expected, shows a failure of the linear model to generalise, as it is incapable of expressing the behaviour of a harsh nonlinearity.
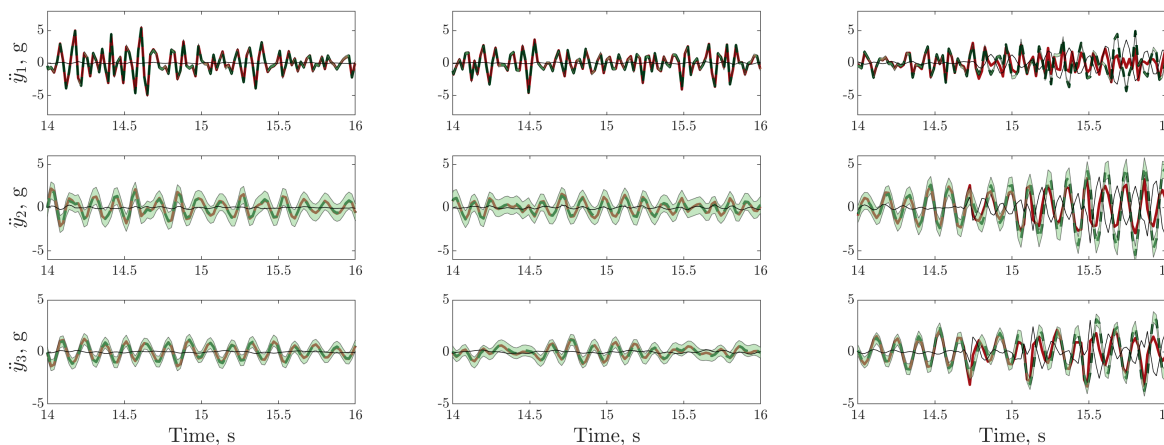

Figure 4. Linear model $\mathcal{M}^l(F, \boldsymbol{\theta}_{\mathcal{D}1}^{MAP})$ predictions, calibrated using dataset one $\mathcal{D}_1$, between 14 and 16s on datasets one to three (left to right); measured acceleration $\{\ddot{y}_i\}_{i=1}^3$ (—), model prediction $\{\mathcal{M}_i^l(F, \boldsymbol{\theta}_{\mathcal{D}1}^{MAP})\}_{i=1}^3$ (- -), $\{\mathcal{M}_i^l(F, \boldsymbol{\theta}_{\mathcal{D}1}^{MAP}) \pm 3\sigma_{n,i}\}_{i=1}^3$ (green shaded region), residual error $\{\ddot{y}_i - \mathcal{M}_i^l(F, \boldsymbol{\theta}_{\mathcal{D}1}^{MAP})\}_{i=1}^3$ (—).

These results motivate the following questions for developing a digital twin (explored in this paper):

1. What does a digital twin do when predictive performance is poor?
2. How does a digital twin account for missing physics?
3. How does a digital twin learn new physics?
4. What is the impact on the control of the structure?

2.3. Proposed Digital Twin Model Structure

Developing a digital twin that allows predictions to be in one-to-one correspondence with the physical twin for the complete operational phase of the structure is the overall objective of this study. This is a particularly challenging aim, and along with the results in Section 2.2, raises the question, *what must a validated model do to progress towards becoming a digital twin?* Here it is suggested that to become a digital twin some form of decision making and 'learning' must be performed, such that the model is always kept in good agreement with the real structure, with many of these decisions being made autonomously (i.e., combining knowledge and connectively from Figure 1). With this aim in mind, it is clear that a digital twin will struggle to always provide accurate predictions if constructed from physics-based models alone, as *all physics* governing a structure are never fully known (even solving physics that is known can often require some form of numerical approximation). This paper therefore proposes that digital twins are a blend of physics-based and data-based modelling techniques, where the data-based component aims

to model (or at least compensate for) the unknown physics—given the assumption that data from the physical twin is reliable and not misleading due to faulty sensors etc. A key benefit of constructing a data-based component will be that it can be utilised in highlighting when improvements are required (Section 4). Once identified several main actions can be considered:

- Recalibrate the physics-based model: improve estimates of the model parameters.
- Update the data-based component: improve modelling of unknown physics.
- Addition of more physics: add new identified physics into the physics-based model.
- Do nothing.

These potential actions structure the remainder of the paper. Section 3 considers issues associated with only ever recalibrating the physics-based model. The data-based component is introduced in Section 4, where its role in both improving predictive accuracy and identifying regions of poor performance are outlined. Section 5 explores a method for identifying additional physics through the use of hybrid testing.

The modelling structure of the digital twin utilised in this paper is,

$$\ddot{Y} = \ddot{Y}^{dt} + e = \mathcal{M}^d(\mathcal{M}^p(X, \boldsymbol{\theta}), \boldsymbol{\phi}) + e \tag{4}$$

where $\ddot{Y} \in \mathbb{R}^{N \times d_y}$ are the noisy real world observations, based on the assumption of independent Gaussian noise $e \in \mathbb{R}^N$. The digital twin outputs are denoted $\ddot{Y}^{dt} \in \mathbb{R}^{N \times d_y}$, constructed from a machine learner $\mathcal{M}^d(\cdot, \cdot)$, that has a set of (hyper)parameters $\boldsymbol{\phi} \in \mathbb{R}^q$ and whose inputs are the outputs (and potentially also the inputs) of a physics-based model $\mathcal{M}^p(\cdot, \cdot)$. The set of parameters $\boldsymbol{\theta} \in \mathbb{R}^m$ (which typically have some physical meaning) and inputs $X \in \mathbb{R}^{N \times d_x}$ complete the definition of the physics-based model $\mathcal{M}^p(\cdot, \cdot)$.

This modelling structure, which uses a machine learner as a bias corrector, is useful when little is known about the discrepancy between the physics-based model and the response of the physical twin, as is common in many real world applications. However, in scenarios where more knowledge is known about the potential sources of missing physics, this information should be used in constructing a model structure that utilises the machine learner in such a way that it reflects these assumptions; for example, in this case study two machine learners could be constructed, one for each contact state (engaged or disengaged), where the physics-based model could be a piece-wise linear model that informs the digital twin when the contact state transitions from being disengaged to engaged. The proposed digital twin model structure (Equation (4)) will be used to construct a framework that seeks to move beyond a validated model and towards a digital twin for an operational context.

3. The Problem of Model Updating

Recalibration is one tool for improving a digital twin, more commonly known as model updating within the dynamics literature [30–33]. The idea behind model updating is that predictive performance can be improved by finding better estimates of the physics-based model parameters $\boldsymbol{\theta}$. This strategy can be applied in a continuous manner, where improved parameter estimates are obtained given more training data $\mathcal{D}$ [34]; however, the underlying assumption is that the physics-based model (almost) perfectly replicates the real world, such that bias is not introduced in the calibration stage. Any significant deviations between the model structure and the real world, arising due to missing physics in the model, will lead to biased parameter estimates during calibration and a failure of the model to generalise, given these parameter estimates. This is especially problematic if the model parameters have some physical meaning, e.g., relating to the size of a crack, as biased parameter estimates will lead to suboptimal or even negative asset management decisions. As such, model updating cannot be the only tool utilised in improving the predictive performance of a digital twin.

To demonstrate the problem of model updating, the linear model $\mathcal{M}_i^l(F, \boldsymbol{\theta}_{\mathcal{D}3}^{MAP})$ has been recalibrated (using the approach in Section 2) using dataset three $\mathcal{D}_3$; parameter estimates are shown in Table 1. This reflects the scenario where, given the physics-based model performed poorly on dataset three, the model should be recalibrated. Predictions from the linear model given the new parameter set $\boldsymbol{\theta}_{\mathcal{D}3}^{MAP}$ on datasets one to three are shown in Figure 5; NMSEs are $\{3.169, 20.135, 14.249\}$, $\{3.070, 15.259, 15.113\}$ and $\{10.768, 19.794, 19.745\}$ for datasets one to three, respectively. There are several observations to be made based on these results. Firstly, the predictive performance on dataset three is still far from perfect, even though this dataset was observed during the calibration process. This result is expected, given the linear model is being fitted to the data with a harsh nonlinearity. This observation may appear trivial, but it relates to some industrial practice, where given poor model performance, the first potential solution is to try recalibrating the model. The second observation, is that by calibrating the model to dataset three, bias has been introduced to the parameters, leading to poor generalisation on datasets one and two (when compared to Figure 4); indicated by the higher NMSEs and larger, more structured residual errors.
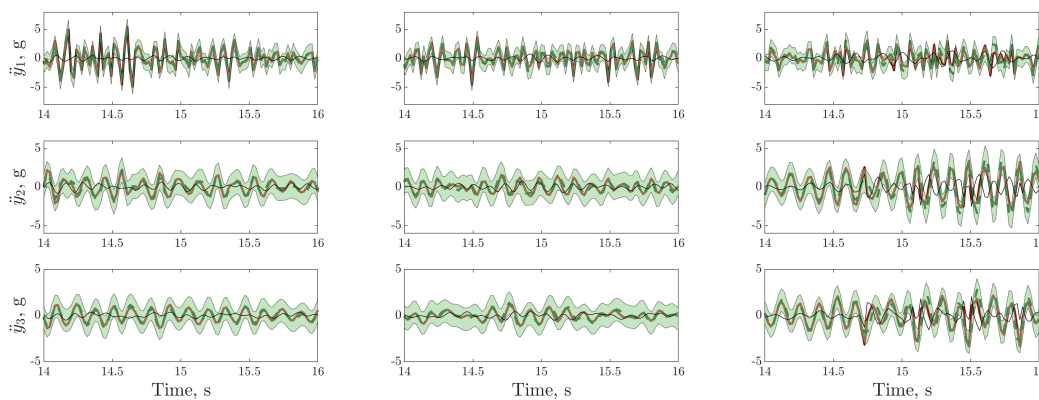

Figure 5. Linear model $\mathcal{M}^l(F, \boldsymbol{\theta}_{\mathcal{D}3}^{MAP})$ predictions, calibrated using dataset three $\mathcal{D}_3$, between 14 and 16s on datasets one to three (left to right); measured acceleration $\{\ddot{y}_i\}_{i=1}^3$ (—), model prediction $\{\mathcal{M}_i^l(F, \boldsymbol{\theta}_{\mathcal{D}3}^{MAP})\}_{i=1}^3$ (- -), $\{\mathcal{M}_i^l(F, \boldsymbol{\theta}_{\mathcal{D}3}^{MAP}) \pm 3\sigma_{n,i}\}_{i=1}^3$ (green shaded region), residual error $\{\ddot{y}_i - \mathcal{M}_i^l(F, \boldsymbol{\theta}_{\mathcal{D}3}^{MAP})\}_{i=1}^3$ (—).

If a digital twin is to consistently overcome poor predictive performance, and remain an accurate tool, then it must be able to do more than just recalibrate model parameters; there must be some mechanism for estimating and approximating the missing physics. It could be, in this more obvious case study, that the engineer sees the linear models' predictions on dataset three and decides to choose a more appropriate physics-based model, one that may better describe the contact behaviour, for example a piecewise-linear model. However, in many applications the choice of improved model may not be so obvious. In this scenario, the engineer may want the digital twin to provide evidence that a different model is appropriate, and if so, help the engineer decide which model to choose. This reflects the fact that developing a nonlinear model, for example, is a complex and expensive task. Before exploring this route, the engineer may wish to see if a data-based model could be used as an alternative to overcome modelling limitations. The next section therefore explores using data-based models, based on machine learning techniques, to predict missing physics.

4. Data-Augmented Modelling

It has been argued above that a digital twin must have some mechanism to account for missing physics during deployment. A natural choice for this mechanism is to add some corrective factor to the

physics-based model that accounts for any missing phenomena [35–37]. However, the parametric form of the corrective factor will often be unknown—otherwise it would have been included in the physics-based model. Instead a non-parametric approach is required, allowing any arbitrary functional form to be inferred. Machine learning offers several types of non-parametric regressors that could be used to map from the output of the physics-based model to the measured data (as shown in Equation (4)), e.g., support vector machines [38], neural networks [39,40] etc. It will be argued that a probabilistic machine learner will be most appropriate, given the uncertainty about the missing functional form. For this reason Gaussian process (GP) regression [41] is selected to augment the physical model; this approach is one possible answer to *how does a digital twin account for missing physics?*

4.1. Gaussian Process Regression

Gaussian process regression seeks to infer the unknown latent function $f(X)$ (where $X \in \mathbb{R}^{N \times d}$ are inputs) of the noisy function $y = f(X) + e$ (where $y \in \mathbb{R}^N$ are the target outputs). First a Gaussian process prior is defined over the latent function,

$$f \sim \mathcal{GP}(m(X)\,k(X, X')), \tag{5}$$

where $\mathcal{GP}(\cdot, \cdot)$ is a Gaussian process, with a mean function $m(\cdot)$ and covariance function $k(\cdot, \cdot)$, defining prior belief about the types of possible functions that could create $f \in \mathbb{R}^N$. The mean function defines the prior mean of the process, and the covariance function states the correlation between any two points in the inputs space (hence a function of X and X') and controls the functions 'smoothness'. Here zero mean and Matérn 3/2 covariance functions are utilised, given that no prior mean information is known about the missing functional form, and that a Matérn 3/2 is (3/2-1) times mean square differentiable [42], making it well suited to modelling relatively 'smooth' real world functions (for other mean and covariance function the reader is referred to [41]). The covariance function is defined as,

$$K_{f,f} = k(X, X') = \sigma_f^2 \left(1 + \sqrt{3}r \exp\left(\sqrt{3}r\right)\right), \tag{6}$$

where,

$$r = \sqrt{(X - X')^{\mathsf{T}} L^{-1} (X - X')}, \tag{7}$$

where $K_{f,f} \in \mathbb{R}^{N \times N}$ is the covariance matrix, $\boldsymbol{\phi} = \{\sigma_f^2, L\}$ are a set of hyperparmeters called the signal variance and lengthscale, respectively, where $L = \mathrm{diag}(l_1, \ldots, l_d)$ (making the covariance function an automatic relevance determination prior, i.e., it reduces the effect of redundant inputs). The shorthand notation $K_{f,*} = k(X, X'_*)$ is used, where f indicates training and $*$ test data. Predictions can be made by forming the joint Gaussian distribution between a set of training $\{X, y\}$ and testing data $\{X_*, y_*\}$, assuming a Gaussian likelihood,

$$\begin{bmatrix} y \\ y_* \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix} \begin{bmatrix} K_{f,f} + \mathbb{I}_f \sigma_n^2 & K_{f,*} \\ K_{*,f} & K_{*,*} + \mathbb{I}_* \sigma_n^2 \end{bmatrix} \right), \tag{8}$$

where $\mathbb{I}$ denotes an identity matrix and σ_n^2 is a Gaussian noise variance (i.e., $e \sim \mathcal{N}(0, \sigma_n^2)$). Standard Gaussian conditionals can be applied in order to obtain the predictive posterior distribution,

$$p(\boldsymbol{y}_* \mid X_*, \boldsymbol{y}, X, \boldsymbol{\phi}) = \mathcal{N}(\mathbb{E}(\boldsymbol{y}_*), \mathbb{V}(\boldsymbol{y}_*)) \tag{9}$$

$$\mathbb{E}(\boldsymbol{y}_*) = K_{*,f}(K_{f,f} + \mathbb{I}_f \sigma_n^2)^{-1}\boldsymbol{y} \tag{10}$$

$$\mathbb{V}(\boldsymbol{y}_*) = K_{*,*} + \mathbb{I}_* \sigma_n^2 - K_{*,f}(K_{f,f} + \mathbb{I}_f \sigma_n^2)^{-1}K_{f,*}. \tag{11}$$

The hyperparameters (including the noise variance) are typically inferred through a type-II maximum likelihood approach [41], found by minimising the negative log marginal likelihood $\hat{\boldsymbol{\phi}} = \arg\min - \log p(\boldsymbol{y} \mid X, \boldsymbol{\phi})$, where,

$$\log p(\boldsymbol{y} \mid X, \boldsymbol{\phi}) = \frac{1}{2}\boldsymbol{y}^{\mathsf{T}}(K_{f,f} + \mathbb{I}\sigma_n^2)^{-1}\boldsymbol{y} + \frac{1}{2}\log|K_{f,f} + \mathbb{I}\sigma_n^2| + \frac{N}{2}\log 2\pi. \tag{12}$$

The probabilistic nature of a GP model means that the variance associated with posterior process (Equation (11)) reflects the uncertainty in the identified latent function. Given the tool is designed for interpolation, the variance provides a measure of extrapolation, which can be used to identify regions where the input-output pairs were 'far' from the training data (where 'far' is defined by the lengthscales in the covariance function). Figure 6 demonstrates the effect, whereby the variance increases away from the training observations, indicating that the model is less certain about the mean prediction; a useful property in addressing the question of *how to access when predictive performance of a digital twin will be poor*.
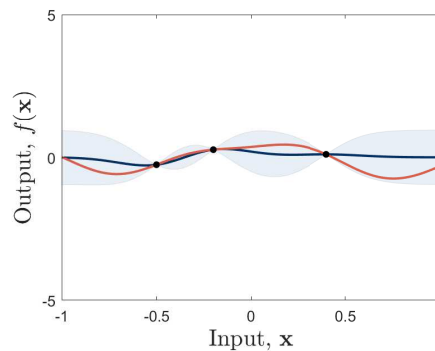

Figure 6. Example of a Gaussian process model trained on three data points (●), where the true function (—), predictive mean (—) and standard deviation, $\pm 3\sqrt{\mathbb{V}(\boldsymbol{y}_*)}$ (blue shaded region), are shown.

4.2. Data-Based Model Component

For the digital twin of the shear structure, the data-based component is expected to capture some dynamic content (as the missing latent function will be dynamic). In order to capture this behaviour, the GP input matrix X will contain lagged information (in a similar form to an autoregressive model structure). However, the input vector will only contain lagged information from the forcing F and the physics-based model outputs $\ddot{Y}^p = \mathcal{M}^p(F, \boldsymbol{\theta})$; meaning $X = \{\{\ddot{\boldsymbol{y}}_i^p(t_n - q), \ldots, \ddot{\boldsymbol{y}}_i^p(t_n - 1), \ddot{\boldsymbol{y}}_i^p(t_n)\}_{i=1}^3, F(t_n - q), \ldots, F(t_n - 1), F(t_n)\}$, where q is the number of lags. The outputs for each of the three Gaussian process models are the accelerations at each floor; for example, the output of the Gaussian process modelling the acceleration at the first floor is $y = \ddot{y}_1(t_n)$. This input-output structure means that the GP model can be used to make m-step ahead predictions online, as long as the forcing input is known for the m-steps (where the physics-based model only requires the forcing to produce output predictions).

It is important to note that this structure differs from a GP-NARX (nonlinear autoregressive exogenous) model structure [43,44], where the measured response (rather than physics-based model outputs) become inputs to the GP regression model in an autoregressive manner. Additionally, this modelling structure is similar to a nonlinear finite impulse response (FIR) model [45], augmented with the physics-based model outputs. As the input matrix in the data-based component is independent of any measured response variable, the combined physics and data-based models in Equation (4) can be used to simulate the predicted output response for any forcing input.

The initial digital twin model, structured as Equation (4), was formed from the linear physics-based model trained on dataset one, i.e., $\ddot{Y}^p = \mathcal{M}^p(X, \boldsymbol{\theta}) = \mathcal{M}^l(F, \boldsymbol{\theta}_{\mathcal{D}1}^{MAP})$ and three independent Gaussian process models (one for the acceleration at each floor) where nine lags of the forcing and linear model outputs were used (based on the number of lags with the lowest negative log marginal likelihood in training). Figure 7 shows predictions when the GP models are trained using dataset one, i.e., $\ddot{Y}^{dt} = \mathcal{M}^d(\mathcal{M}^p(X, \boldsymbol{\theta}), \boldsymbol{\phi}) = \mathcal{M}^{GP}(\{\ddot{Y}^p, F\}, \hat{\boldsymbol{\phi}}_{\mathcal{D}1})$, and applied on datasets one to three, where the NMSEs are $\{0.000, 0.000, 0.000\}$, $\{0.309, 2.479, 2.089\}$ and $\{29.554, 44.518, 32.434\}$, respectively. It is noted that the addition of the data-based model has not completely reduced the error on dataset three (where the harsh nonlinearity was active), which is expected given that dataset one contains only information from the structure in a non-contact state (although the performance is better than the linear model alone that produced NMSEs $\{34.595, 65.310, 42.214\}$ on dataset three). The reason the data-augmented model performs better on dataset three than the linear model alone, is that the GP model has learnt to compensate for some of the model-form misspecification. For example, the physics of the joints are simplified in the linear shear-model structure, and the residual from this behaviour may have been identified by the GP from the linear behaviour in dataset one. However, the data-based component can be updated, given new information about the nonlinearity, subsequently improving predictive performance in dataset three, without biasing any physical parameters.
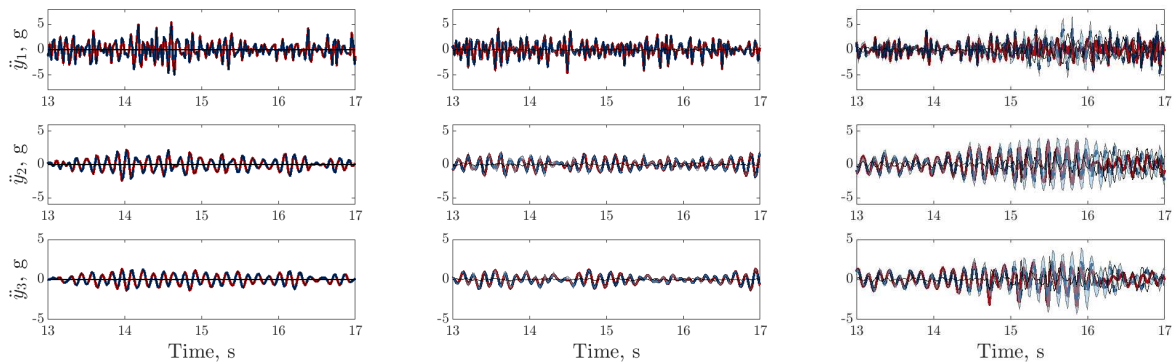

Figure 7. Digital twin $\mathcal{M}^{GP}(\{\mathcal{M}^l(F, \boldsymbol{\theta}_{\mathcal{D}1}^{MAP}), F\}, \hat{\boldsymbol{\phi}}_{\mathcal{D}1})$ predictions, trained using dataset one $\mathcal{D}_1$, between 13 and 17s on datasets one to three (left to right); measured acceleration $\{\ddot{y}_i\}_{i=1}^3$ (—), digital twin mean prediction (--), digital twin $\pm3\sigma$ confidence intervals (blue shaded region), residual error (—).

In addition to improved predictive accuracy, the digital twin now includes a measure of predictive uncertainty, expressed through the GP predictive latent function variance (i.e., $\mathbb{V}(\boldsymbol{f}_*) = \mathbb{V}(\boldsymbol{y}_*) - \mathbb{I}_* \sigma_n^2$), displayed in Figure 8. Intuitively, a threshold can be been defined in order to determine when predictive performance is uncertain and a decision should be made. There are several methods for setting this threshold, discussed in Section 4.3, where, for the example in Figure 8, a threshold has been obtained by taking the maximum point-wise predictive variance from the digital twins initial predictions on an

independent test set (in this case the first 100 data points in dataset two). It is noted that as the digital twin, once trained, only requires the forcing input, simulation can be made for any forcing schedule in order to evaluate loading conditions where the digital twin is uncertain, prior to deployment. Future data points that would be valuable to improving the digital twin can therefore be highlighted before they occur; a valuable property particularly if the digital twin is being used for control or health monitoring. The following section extends this observation, describing an active learning procedure that can be used to highlight data points of interest, and continuously update the data-based component of the digital twin.
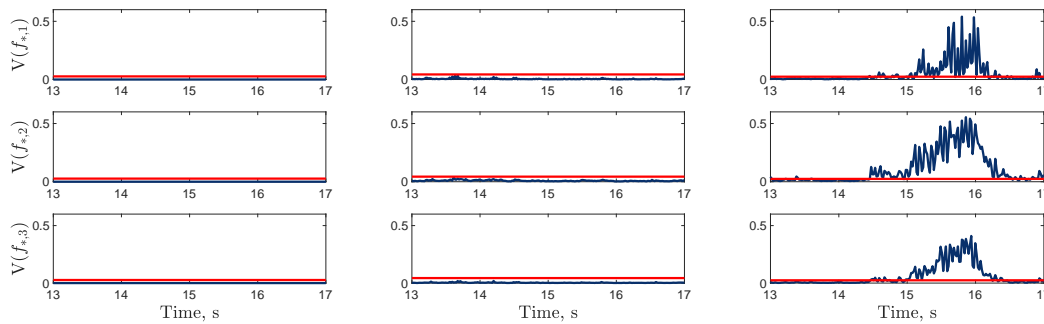
Figure 8. Digital twin $\mathcal{M}^{GP}(\{\mathcal{M}^l(F, \boldsymbol{\theta}_{\mathcal{D}1}^{MAP}), F\}, \hat{\boldsymbol{\phi}}_{\mathcal{D}1})$ predictive variance (**—**), trained using dataset one $\mathcal{D}_1$, between 13 and 17s on datasets one to three (left to right); threshold (**—**) determined by the maximum variance from the first 100 data points on dataset two.

4.3. Active Learning Approach

Active learning, a branch of machine learning, forms algorithms that actively query data points, with the aim of improving the learner [46,47]. For a digital twin predicting dynamic outputs, the most suitable type of active learning is stream-based active learning [48,49] (a form of selective sampling), in which the learner observes data points and makes a decision whether to query an observation or discard it. By querying a data point, the learner has the ability to use the data point in updating the model. Key to active learning is the aim that by using a model-informed approach, the training dataset selected by the model will be more informative (leading to better generalisation) than selecting a training dataset using a non-model informed approach, such as random or uniform sampling etc. This type of approach can be used as a way of answering *what does a digital twin do when predictive performance is poor?*

The proposed active learning approach seeks to use the predictive variance of the latent function (i.e., $\{\mathbb{V}(\boldsymbol{f}_{*,i}) = \mathbb{V}(\boldsymbol{y}_{*,i}) - \mathbb{I}_* \sigma_{n,i}^2\}_{i=1}^3$) and a threshold T to determine when a decision is made to query a particular observation (in this case only considering future observations). Any *j*th instance where the latent predictive variance is greater than the threshold, i.e., $\mathbb{V}(\boldsymbol{f}_{j,*}) > T$, for any output, are data points that should be queried. In a digital twin context, a query can result in three main actions: recalibrating the physics-based model, retraining the data-based model, or updating the physics-based model with new physics. The latter action is the most challenging to perform in an automated manner, and as such is not considered in the proposed active learning scheme. Instead, when a data point is queried, the instance $\{\boldsymbol{x}_j, y_j\}$ becomes part of a new training dataset, i.e., $\mathcal{D}_{k+1} = \{\mathcal{D}_k, \{\boldsymbol{x}_j, y_j\}\}$. The new training dataset can be used to either recalibrate the physics-based model parameters $\boldsymbol{\theta}$ or update the GP model hyperparameters $\boldsymbol{\phi}$. Unfortunately, recalibrating the physics-based model changes the input-output map of the GP model (as the physics-based model outputs are inputs to the Gaussian process). As a result, deciding to recalibrate the physics-based model is computationally more expensive than updating the data-based component alone. For this reason, the active learning approach will initially consider two actions: do nothing or update the GP hyperparameters $\boldsymbol{\phi}$, outlined in Algorithm 1.

Algorithm 1 Active learning for data-based component of a digital twin

---

1: Set initial dataset $\mathcal{D}_1 = \{x_j, y_j\}_{j=1}^{N_{initial}}$

2: Train GP on $\mathcal{D}_1$ and obtain $\hat{\phi}_1$

3: Set initial threshold $T_{N_{initial}-1}$

4: $k = 1$

5: **for** $t = N_{initial} : N$ **do**

6:      Predict GP at $\{x_{*,t}\}$ using $\hat{\phi}_k$

7:      Apply 'forgetting factor', $T_t = f_f T_{t-1}$

8:      **if** $\mathbb{V}(f(x_{*,t})) > T$ **then**

9:          Update dataset $\mathcal{D}_{k+1} = \{\mathcal{D}_k, \{x_t, y_t\}\}$

10:          Retrain GP on $\mathcal{D}_{k+1}$ and obtain $\hat{\phi}_{k+1}$

11:          Update predictions at $\{x_{*,1:t}\}$ using $\hat{\phi}_{k+1}$

12:          Update threshold, $T_k = \max\left[\mathbb{V}(f(x_{*,1:t}))\right]$

13:          $k = k + 1$

14:      **end if**

15: **end for**

The main consideration in Algorithm 1 is how to set (and whether to update) the threshold. One approach is to fix the threshold (i.e., lines 7 and 11 in Algorithm 1 are not performed), using some expected performance of the Gaussian process. This may be challenging to set *a priori* to deployment, and if set at a low value, will lead to very frequent querying, and if too high, will result in all data points being discarded. Another approach is to set an adaptive threshold; two approaches are considered. The first scheme sets the threshold T_k at query k to be the maximum variance of the latent function at the previous t observations, i.e., $T_k = \max\left[\mathbb{V}(f(x_{*,1:t}))\right]$ (ignoring line 7 in Algorithm 1). This criteria states that the next queried observation must be more informative than past observations. However, this criteria may lead to scenarios where the threshold rises to a value where no future points can be queried. To overcome this problem the criteria can be amended with a 'forgetting factor' f_f, allowing the threshold T_k to decrease at each sample point t from the previous update k (line 7 of Algorithm 1). The 'forgetting factor' should be set $f_f < 1$ (where $f_f = 1$ is the same as taking T_k), meaning that the threshold will decrease until a new point is queried. A value of f_f close to zero will very frequently query observations regardless of their informativeness, whereas a value close to one states only data points with a large latent variance are sampled.

Furthermore, in Algorithm 1, the latest GP model (from update k) is used to (re-)predict the output at all test instances. This assumes that the latest GP model is the most optimal so far. However, it is trivial to amend the algorithm such that the latest GP trained on dataset $\mathcal{D}_k$, is only used to make predictions on time points from k until the next query at $k + 1$. This change in implementation would be useful if it is assumed that the GP for each training dataset k is optimal until the next queried data point.

Comparisons of the performance between uniform sampling (left panels) and active learning with both fixed (middle panels) and adaptive thresholds, where $f_f = 1$ (right panels), are presented in Figure 9. In addition, Figure 10 provides a comparison of active learning with different 'forgetting factors': $f_f = 0.999$ (left panels), $f_f = 0.99$ (middle panels) and $f_f = 0.9$ (right panels)—the NMSEs for both figures are calculated for the complete dataset at every update step, therefore assessing both the digital twins ability to generalise, as well as predictive performance. The physics-based model for each approach was linear (Equation (1)), where the parameters were $\theta_{\mathcal{D}1}^{MAP}$. Each method was initialised with 75 observations from dataset one, i.e., $N_{initial} = 75$. The uniform sampling approach queried every 25th data point (where the threshold criteria is ignored) and for active learning with a fixed threshold, $T = 0.001$ for

each floor (based on the performance of GP models in Figure 8). For datasets two and three, $\mathcal{D}_1$ (line 1 in Algorithm 1) was set as the final $\mathcal{D}_k$ from the previous dataset (meaning information from the previous datasets is carried forward). The number of additional queries for each dataset is provided in Table 2.

The performance of uniform sampling is poor compared to the other approaches, with the algorithm taking a long time to converge for dataset one, and producing higher NMSEs for all three datasets when compared to the active learning approaches. For dataset three, the approach is actually detrimental to performance, with the predictions for floors two and three becoming worse with additional queries. In contrast, both the fixed threshold and adaptive thresholds (where $f_f = 1$) converge quickly for dataset one, and maintain low NMSEs for both datasets one and two, with methods both querying a large number of observations around 14 s on dataset three, leading to an increase in performance. The main difference between these two approaches is the number of queries made, with the fixed threshold querying significantly more observations, 365 in total, compared to the adaptive threshold ($f_f = 1$) with 130 queries. This difference is particularly clear for dataset three just after 16 s, where the adaptive threshold is no longer sampling every data point (due to a high threshold), leading to the approach plateauing earlier, with higher final NMSEs. This shows that there is a trade-off between how the threshold is set, such that the number of queries are not too high, slowing the GP retrain step, whilst balancing potential performance.
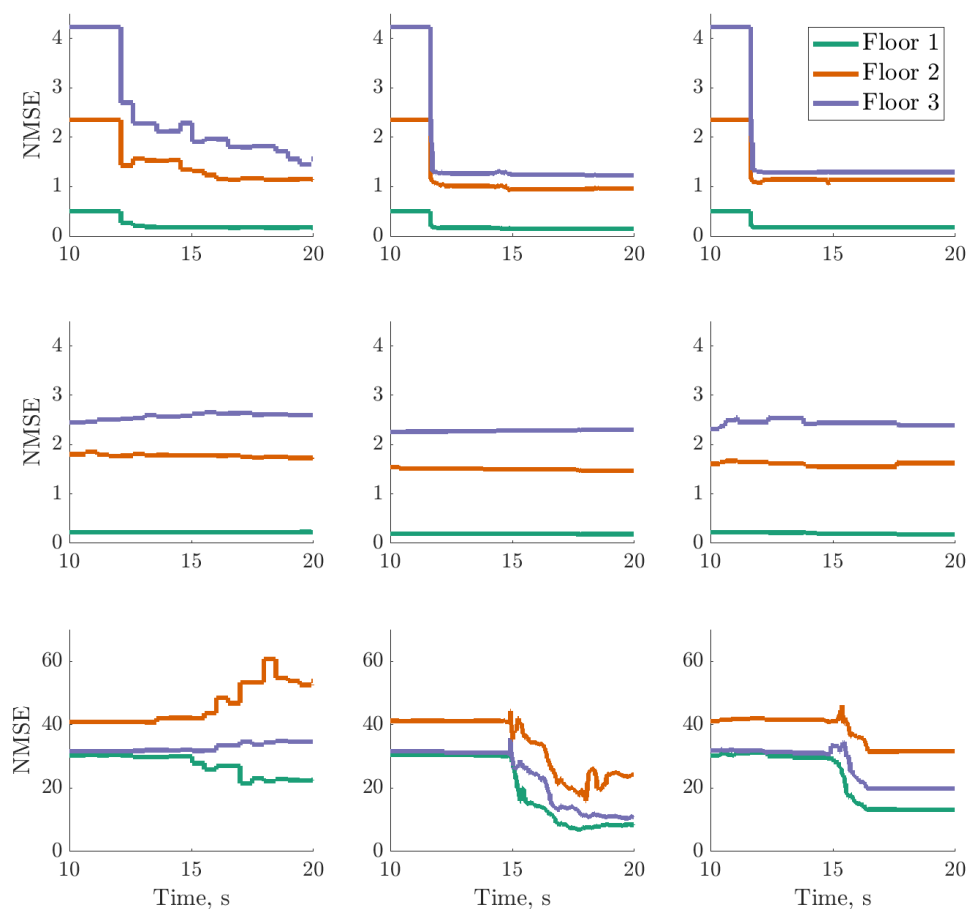

Figure 9. Normalised mean squared errors (NMSEs) for digital twin predictions on datasets one to three (top to bottom panels). Left panels, uniform sampling; middle panels, active learning with a fixed threshold; right panels, active learning with adaptive threshold ($f_f = 1$).
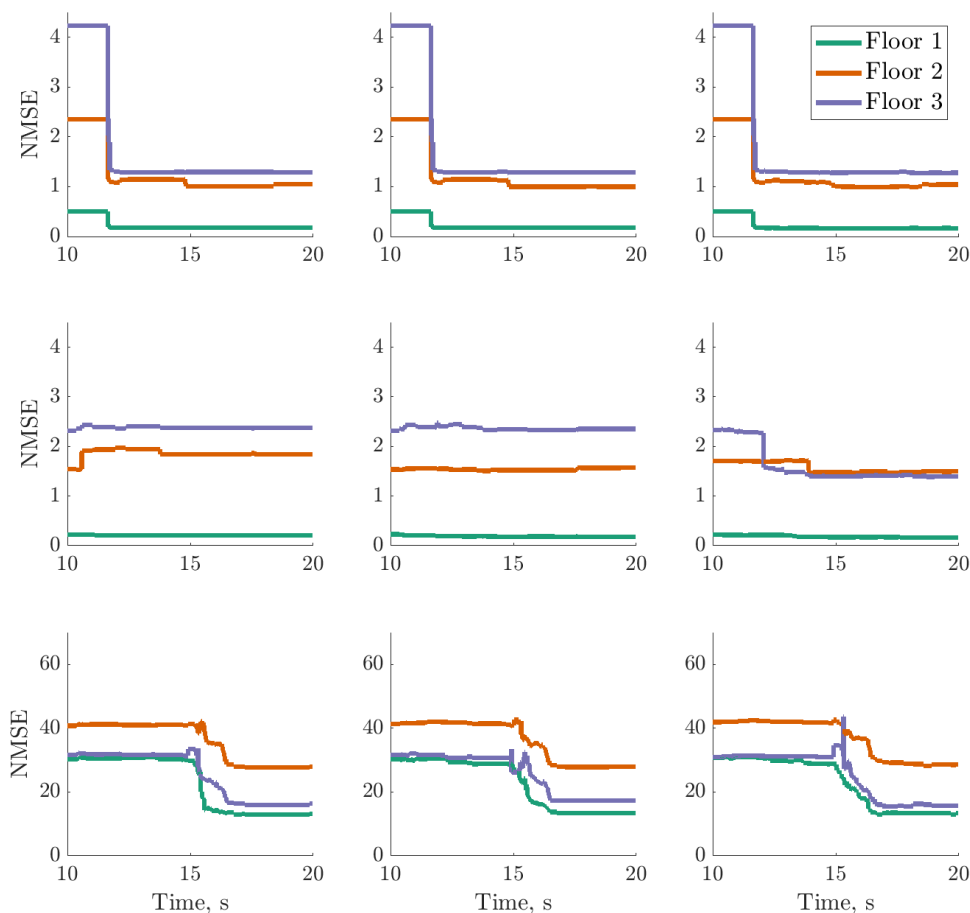
Figure 10. Normalised mean squared errors (NMSEs) for digital twin predictions on datasets one to three (top to bottom panels). Active learning with adaptive threshold where $f_f = 0.999$ (left panels), $f_f = 0.99$ (middle panels) and $f_f = 0.9$ (right panels).

Table 2. Comparison of additional queries for each dataset.

Dataset	Uniform	Fixed	$f_f = 1$	$f_f = 0.999$	$f_f = 0.99$	$f_f = 0.9$
One	18	69	11	14	17	40
Two	21	16	21	18	37	71
Three	21	280	98	115	106	135

A key decision in performing the active learning approach is setting a reasonable threshold, given some engineering judgement. Three other 'forgetting factors' were also compared, $f_f = 0.999$, $f_f = 0.99$ and $f_f = 0.9$, where a lower value leads to more queries at locations where the predictive latent variance may not be maximum. This expectation is confirmed by Table 2, where the number of queries increases as f_f decreases (with all three approaches querying less observations than the fixed threshold). It is interesting to note that the three approaches provide similar NMSE performance on dataset one, even though $f_f = 0.9$ queries over double the amount of observations. Both $f_f = 0.999$ and $f_f = 0.99$ perform relatively similarly for dataset two, with $f_f = 0.99$ obtaining a slightly better NMSE for floor two. However, $f_f = 0.9$ shows the best performance on dataset two, even compared to the fixed threshold. All three 'forgetting factor' values produce similar initial performance on dataset three,

with each querying observations from 14 s, and unlike $f_f = 1$, keep querying data points after the high amplitude response from the initial contact; with relatively similar final NMSEs around $\{13, 28, 16\}$ for floors one to three, respectively—better than $f_f = 1$, $\{13.278, 30.787, 19.761\}$, but worse than the fixed threshold, $\{8.414, 24.263, 10.921\}$.

Figure 11 shows a comparison of the updated predictions at the end of dataset two, i.e., the predictions from the GP model trained on the final training dataset. The figure also shows query locations (vertical lines) which are data points that have been selected to form the training dataset. Figure 11 compares the fixed (left panels) and adaptive (right panels) threshold when $f_f = 0.99$. Queries are sparse for both methods, expected given the system behaves linearly. In contrast, Figure 12 presents the final updated predictions and locations of queries for the fixed threshold (left panels) and $f_f = 0.99$ (right panels). It is clear from the fixed threshold results that the system response has changed in dataset three, leading to all the observations being queried from around 14 s. In comparison, $f_f = 0.99$ queries continuously around the large amplitude response where the harsh nonlinearity is active. A heuristic could be introduced to the algorithm, that given a large number of continuous queries, the physical system is expected to have changed and therefore other actions should be taken, changing the underlying physics-based model. These actions could either be recalibration, or the addition of new physics to the model. This modification is left as future research, but could be aided by considering the associated cost of these actions (both in terms of the consequences of poor performance and in terms of computational resource requires to perform the action), making a utility-based approach appropriate [50,51]. However, the following section, on real-time hybrid testing, considers the scenario where active learning results on dataset three have indicated additional physics are required. Hybrid testing is then considered as one approach for isolating and identifying these physics.
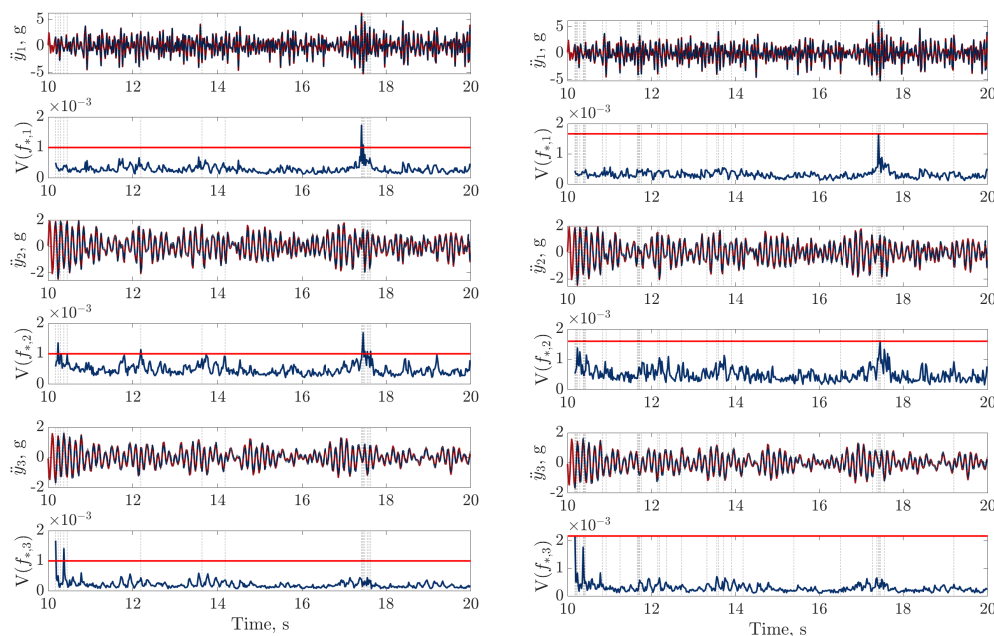

Figure 11. A comparison of the final active learning predictions with fixed (left) and adaptive (right) thresholds ($f_f = 0.99$) on dataset two. The vertical lines (:) indicate a queried observation, alongside the measured acceleration $\{\ddot{y}_i\}_{i=1}^3$ (—), digital twin mean prediction (- -) and digital twin $\pm 3\sigma$ confidence intervals (blue shaded region) (top sub-panels), with the predictive variance (—) and final threshold (—) in the bottom sub-panels.
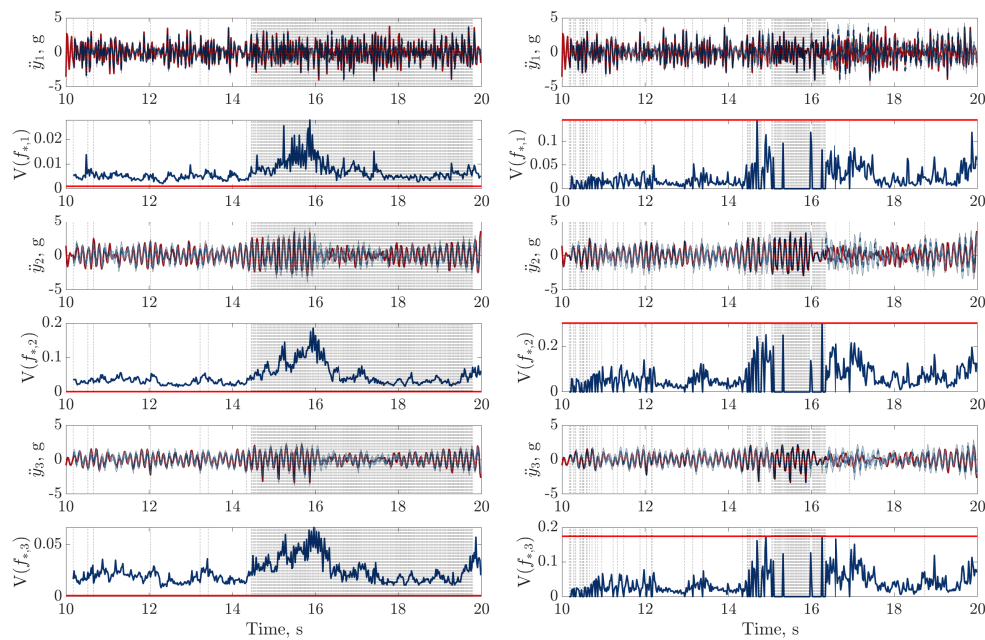
Figure 12. A comparison of the final active learning predictions with fixed (left) and adaptive (right) thresholds ($f_f = 0.99$) on dataset three. The vertical lines (:) indicate a queried observation, alongside the measured acceleration $\{\ddot{y}_i\}_{i=1}^3$ (——), digital twin mean prediction (- -) and digital twin $\pm 3\sigma$ confidence intervals (blue shaded region) (top sub-panels), with the predictive variance (——) and final threshold (——) in the bottom sub-panels.

4.4. Autonomous Decision Making: Challenges and Limitations

A challenge with automating decisions that improve predictive performance is that future accuracy of a model is difficult to determine without knowledge of future observations. The proposed method in Algorithm 1 does not contain any mechanism of assessing future predictive performance, and instead relies on the interpolation and extrapolation uncertainty in the Gaussian process model, quantified in the predictive latent variance. This measure will only inform when the predictive inputs are 'far' away from previously observed inputs in the training data (where 'far' is stated by the choice of covariance matrix and current hyperparameters). The assumption is that the physical twin will still behave as observed at previous training observations, if this is broken then the patterns learnt by the data-based model will not generalise. It may therefore be useful it remove observations from the training dataset overtime if the physical twin changes significantly (i.e., old observations are forgotten in favour of learning new behaviour). Moreover, the current active learning algorithm requires some input to be known for the data-based model, in this case the input force. This may be unrealistic in some real world scenarios, however, in an offline setting the digital twin can be interrogated (in a design of experiments manner) such that the modellers can be prepared for input scenarios where the digital twin outputs are uncertain.

Lastly, the technique is currently formed from GP regression, which has a computational cost of $\mathcal{O}(N^3)$ to train; where N are the number of training observations. Predictions from the GP have a computational cost of $\mathcal{O}(N)$ and $\mathcal{O}(N^2)$ for the mean and variance, respectively. Clearly this is a limitation to the approach, with the algorithm becoming slower to train at each queried observation. Approaches have been proposed for reducing this computational load, such as sparse Gaussian process regression [52–54], reducing the computational cost of training and testing to $\mathcal{O}(NM^2)$ and $\mathcal{O}(M^2)$, respectively; where M

is the size of a set of pseudo-inputs that is much smaller than N. Online sparse GP regression updates has also been proposed [55,56], which can be substituted into the proposed active learning technique, make the approach practical in a wider range of industrial applications.

5. Identifying Physics through Hybrid Testing

It is difficult, if not impossible, to know all physics that affect a physical twin prior to the deployment of a digital twin. There will therefore be instances where new physics must be identified and incorporated into the digital twin. In the context of this paper, the logic is that the component(s) tested experimentally are those where the lack of knowledge regarding the physics is greatest; particularly, strong or harsh nonlinear effects, structural failure, extreme loading etc. By devising a scheme for performing isolated and targeted testing of these problematic components, a cost-effective process of offline model development can be carried out. This type of targetted testing, though performing *real-time hybrid testing*, is one proposed solution to *how does a digital twin learn new physics?*

Real-time hybrid testing is an experiential framework used to assess the dynamics of an assembly by isolating a component or sub-assembly of interest (the *physical substructure*). This technique allows the physical substructure to be tested under realistic, dynamics boundary conditions, imposed through a set of reactions estimated from some numerical model (the *numerical substructure*) and applied by a set of actuators, see for example [57] and references therein. The numerical substructure can naturally be considered as some part of the digital twin, with hybrid testing used to explore modelling modifications, particularly the possibility of nonlinear elements, in a controlled environment without requiring the construction of a full prototype or intervening on the physical twin.

Figure 13 shows a schematic of the hybrid test setup for the three storey shear structure. In this scenario the hybrid testing setup is designed such that the dynamics of the contact mechanism are characterised, hence only the top two storeys are isolated in the physical substructure. The remaining part of the physical system, the first floor and lower columns, are reduced into a numerical substructure, which is characterised by the validated linear components in the digital twin model. In this simulated case study it is assumed that the physical system is either floating, or that the second floor (mass m_2) is attached to a frictionless rail, with the actuator fastened at one side of the structure, with the piston head connected to the second floor. The numerical substructure, characterised through the validated linear components of the digital twin, can be implemented in a real-time digital controller, where the numerical displacement of the second floor y_{2n}, corresponding to the force imparted by the actuator F_m (applied to the second floor, m_2), can effectively be exchanged and measured at the interface between the substructures, and sent to a servo-drive and actuator to be imposed as target boundary conditions (i.e., displacement control is used). The idea is that, in the case of perfect control, the actual physical displacement y_{2p} (at the physical side of the interface) would be equal to the numerical displacement y_{2n}, meaning the hybrid assembly can be considered dynamically equivalent to the complete physical system (within the uncertainty bounds of the numerical substructure).

A challenge for any hybrid testing scenario is that the intrinsic dynamics of the actuator, and limited bandwidth of the PID controllers, typically implemented in off-the-shelf servo-drives, may lead to the physical displacement y_{2p} having a different amplitude and, as a result, be shifted in time with respect to y_{2n}. This error in the applied boundary condition can lead to, at best, a non-representative hybrid assembly, and at worse, dangerous instabilities—as first pointed out by [58].
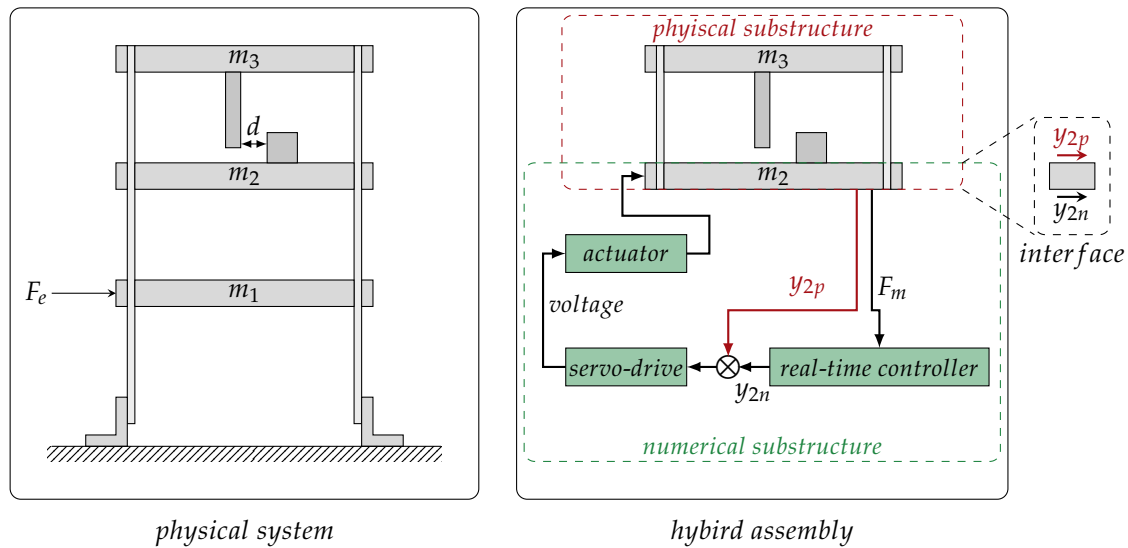
physical system *hybird assembly*

Figure 13. Schematic overview of hybrid testing applied to the three storey shear structure.

In order to highlight the potential and challenges associated with utilising hybrid testing for model development in a digital twin, a simplified simulation of Figure 13 was conducted, for example the actuator is assumed to have no delay. The analysis aimed to detect the threshold at which the contact nonlinearity was excited, with the aim of utilising this information in deciding whether a nonlinear physics-based model is required. Adapting Equation (1), and using hybrid testing nomenclature, the equations can be written as,

$$
\begin{aligned}
F_e &= m_1\ddot{y}_{1n} + c_1\dot{y}_{1n} + c_2(\dot{y}_{1n} - \dot{y}_{2n}) + k_1 y_{1n} + k_2(y_{1n} - y_{2n}) \\
F_m &= m_2\ddot{y}_{2p} + c_3(\dot{y}_{2p} - \dot{y}_{3p}) + k_3(y_{2p} - y_{3p}) - F_{nl}(y_r) \\
0 &= m_3\ddot{y}_{3p} + c_3(\dot{y}_{3p} - \dot{y}_{2p}) + k_3(y_{3p} - y_{2p}) + F_{nl}(y_r)
\end{aligned}
\tag{13}
$$

where $y_r = y_{2p} - y_{3p}$ is the relative displacement between floors two and three, $F_{nl}(y_r)$ is the nonlinear force introduced by the contact mechanism, and F_e the external excitation force. It is noted that the equations related to the second and third floors are used here to simulate the response of the physical substructure, while, in a real hybrid test, y_{2p}, y_{3p} and F_m would be measured experimentally, and the equations with respect to the first floor are of the numerical substructure. The force F_m between the physical substructure and the actuator coincides with the interaction between the second and first floors, expressed as,

$$
- F_m = c_2(\dot{y}_{2n} - \dot{y}_{1n}) + k_2(y_{2n} - y_{1n}).
\tag{14}
$$

In an actual hybrid test, the equations for the numerical substructure (first equation in Equations (13) and (14)) would be integrated within a real-time controller in order to calculate the numerical displacement of the second floor y_{2n}, which would be used as a target signal in the servo-drive.

The contact nonlinearity was modelled in this case study using a bilinear spring,

$$
F_{nl}(y_r) = \begin{cases} 0 & \text{if } y_r \leq d \\ k_{nl}y_r & \text{if } y_r > d, \end{cases}
\tag{15}
$$

where $k_{nl} = 9$ N/mm represents the stiffness of the column, and $d = 0.5$ mm the gap width. The excitation F_e was chosen to reproduce the spectral content of the El Centro earthquake. Its time-domain amplitude was scaled in the simulations to identify the force level that brings the bumper into contact and activates the nonlinear response of the building. For a qualitative assessment of the effect of an imperfect control system on the results, an amplitude error $a_{err} = \pm 0.1$ was deliberately introduced at the interface with the actuator so that the physical displacement would differ from the numerical one ($y_{2p} = (1 + a_{err}) y_{2n}$).

The simulation results are shown in Figure 14. With perfect control, the bumper nonlinearity would be activated at a peak excitation amplitude of 0.7 N. However, the threshold of nonlinear behaviour exhibits a noticeable sensitivity to control error, which could lead to an overestimation of its value of up to 35% against an amplitude error of just 8%.
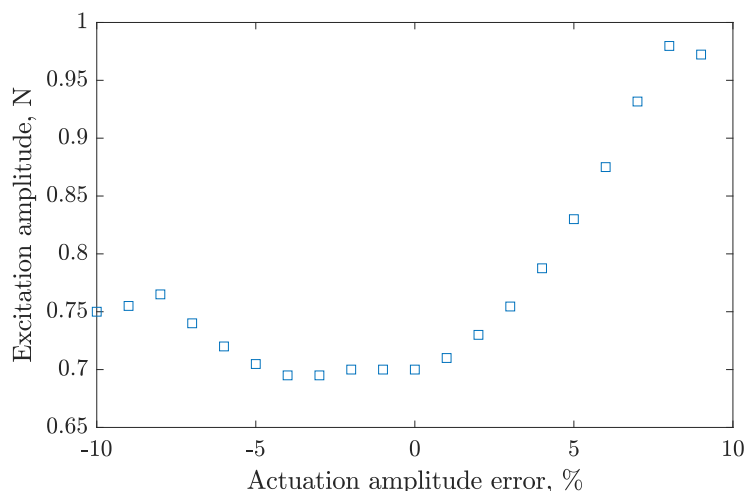
Figure 14. Identification of the threshold of nonlinear behaviour and dependence on the accuracy of the control and actuation system in a hybrid test setup.

To further highlight the importance of adequate control in capturing the real physics of the system, the El Centro earthquake was scaled to yield a peak force of 5 N and applied as the external excitation F_e to Equation (13). This was integrated under two different control conditions at the interface between numerical and physical substructures: (i) perfect control, $y_{2p} \equiv y_{2n}$, and (ii) +5% control amplitude error, $y_{2p} \equiv 1.05 y_{2n}$. In this simplified analysis, the perfect control case also represents the outcome of a traditional test, i.e., a test of the whole structure.

Figure 15 shows the response of the second storey x_2 in the time and frequency domain for both boundary conditions (ideal control case in the left panels, imperfect control in the right panels). In both cases the nonlinear mechanism was activated. It can be seen that the time domain responses (top panels of Figure 15) exhibit very similar envelopes, showing that even an imperfect control can, in this case, lead to a good qualitative assessment of the dynamics of the assembly.

However, in the ideal control case, the bumper and column came into contact nearly three times as often as with the imperfect control, which led to noticeably different spectral characteristics (bottom panels of Figure 15). In the ideal control case, the response exhibits a clear multiharmonic behaviour, with distinct peaks corresponding to multiples of the fundamental frequency 9.6 Hz. It is also interesting to note that, of the peaks identified in Figure 3, only the one corresponding to the first mode of the underlying linear structure remains, while the others reflect the presence of the bilinear stiffness in the contact mechanism. In the imperfect control case, on the other hand, all three peaks corresponding to the modes of the linear

structure are visible, and no harmonic multiples are present, showing that accurate control is needed in order to capture a representative response of the nonlinear dynamics of the structure.

This simple analysis illustrates how hybrid testing can provide experimental data representative of the whole physical twin without having to modify the physical structure and in controlled conditions, thanks to the real-time interaction between the component of interest and the digital twin of the rest of assembly. It can also be used to selectively target the physics of complex components in the context of the larger assembly, as the effects of the harsh nonlinearity introduced by the bumper mechanism can be measured and then propagated to the numerical substructure without modelling approximations. At the same time, this analysis highlights the need for active research in order to address the mutual influence between the uncertainties introduced by the hybrid test control and those associated with the digital twin itself.
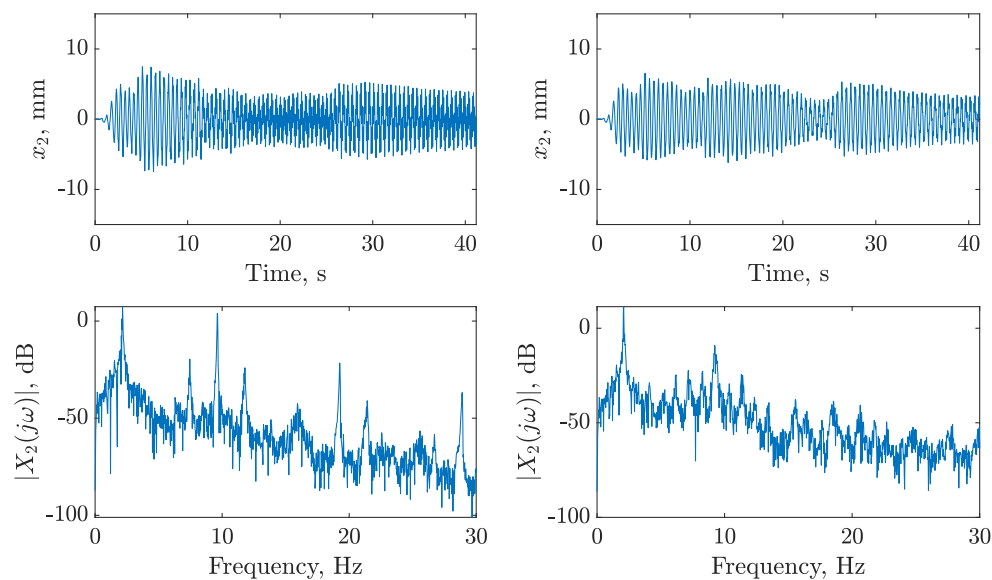
Figure 15. Displacement response of the second storey to an excitation reproducing the El Centro earthquake in the ideal control case (left panels) and with imperfect control (right panels) at the interface between numerical and physical substructures. Time-domain responses are displayed in the top panels, spectra in the bottom panels.

6. Impact of a Digital Twin on Active Control

One of the key advantages of a digital twin, besides improved predictive performance, is the ability to inform the engineer about expected poor predictive performance. This information can be used to aid decision making, helping the engineer understand the limitations of their modelling assumptions, compared to the behaviour the physical twin. This section investigates the impact of a digital twin on the control of the structure, via an active controller (where this scenario is different to that investigated in the hybrid test scenario in Section 5).

At the initial modelling phase, an active control system was designed for the purpose of vibration attenuation, with the aim of removing the potential for the column and bumper to come into contact. The control force was designed to act on the first floor of the structure, in the opposite direction of the external shaker excitation. A model-based feedback controller known as linear-quadratic regulator (LQR) [59] was designed using the model parameters inferred for dataset one $\theta_{\mathcal{D}1}^{MAP}$—given the parameter

set appeared valid based on their performance on dataset two. The equation of motion for the linear model, with a control force, can be written as,

$$M_s \ddot{y} + C_s \dot{y} + K_s y = \phi_p f_p - \phi_c f_c \tag{16}$$

where M_s, C_s, K_s are the mass, damping and stiffness matrices of the structure, respectively; the vector is ordered $y = \{y_1, y_2, y_3\}^\mathsf{T}$. $\phi_p = \{1\,0\,0\}^\mathsf{T}$ is the location of the shaker excitation f_p and $\phi_c = \{1\,0\,0\}^\mathsf{T}$ is the location of the control force f_c. The controller location has been chosen to be the same as the external force, as this delivers optimal performance by reducing the vibration at its source avoiding the transmission through the structure. From Equation 16, the state-space model of the structure results in,

$$\begin{aligned} \dot{w} &= Aw + B_p f_p - B_c f_c \\ z &= Cw + D_p f_p - D_c f_c \end{aligned}, \tag{17}$$

where $w = \{y, \dot{y}\}^\mathsf{T}$ represents the states of the system, z the measured output (in this case acceleration) and the state-space matrices are defined as,

$$A = \begin{bmatrix} 0 & I \\ -M_s^{-1} K_s & -M_s^{-1} C_s \end{bmatrix}, \ B_c = \begin{bmatrix} 0 \\ M_s^{-1} \end{bmatrix} \phi_c, \ B_p = \begin{bmatrix} 0 \\ M_s^{-1} \end{bmatrix} \phi_p, \tag{18}$$

$$C = \begin{bmatrix} -M_s^{-1} K_s & -M_s^{-1} C_s \end{bmatrix}, \ D_p = M_s^{-1} \phi_p, \ D_c = M_s^{-1} \phi_c. \tag{19}$$

The control force generated by the LQR is,

$$f_c = K_{lqr} w \tag{20}$$

where the state vector w is calculated from the measured acceleration via integration, and K_{lqr} is the optimal full-state feedback gain matrix that minimises the following quadratic cost function [59],

$$J = \int_0^\infty \left(w^T Q w + f_c^T R f_c \right) dt, \tag{21}$$

where the state weight matrix is defined as,

$$Q = q \begin{bmatrix} 0 & 0 \\ 0 & I \end{bmatrix}. \tag{22}$$

Initially the state weight and cost weight were set to $q = 10^3$ and $R = 10^{-1}$, respectively.

Figure 16 shows the time history and the frequency response of the accelerations when the LQR is implemented on dataset two (as an independent test dataset). Given the structure behaves linearly, the LQR controller is optimally able to reduce the amplitude of vibration, especially around the resonances of the structure.
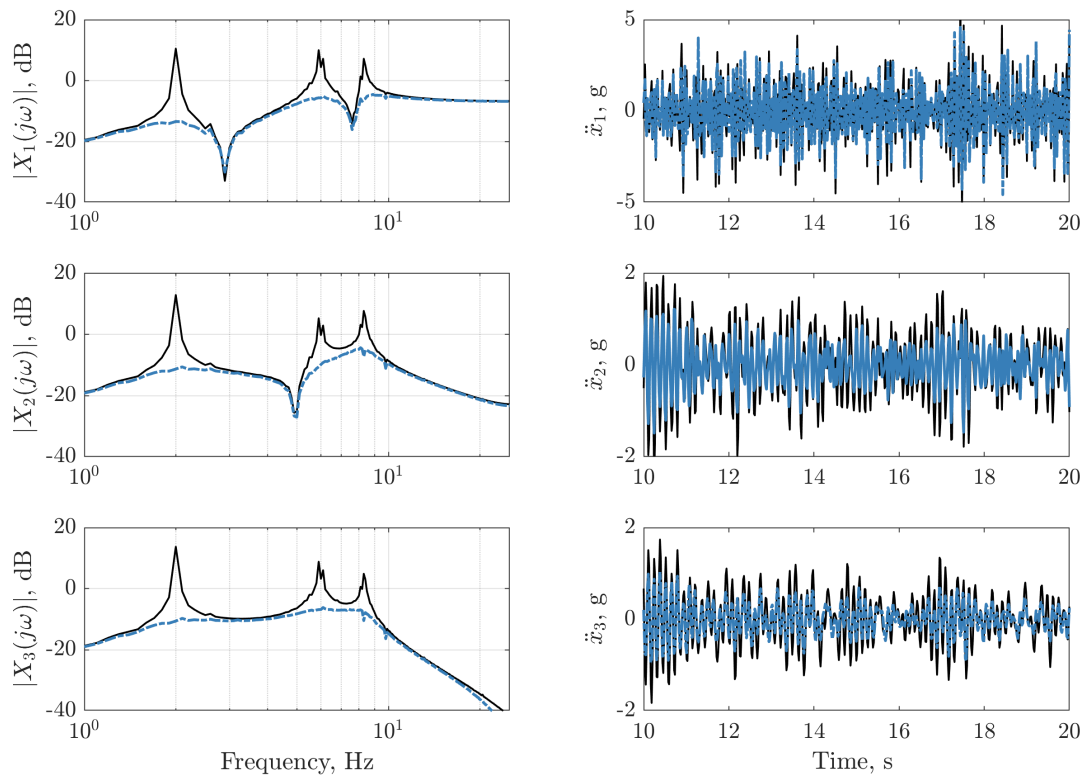
Figure 16. Frequency response function (left panels) and acceleration time history (right panels) for dataset two without control (—), and with LQR controller (- -).

However, as with the initial modelling assumptions, the controller is developed based on a linear model structure, with the parameters fixed at the estimates from dataset one. It is therefore expected that performance of the controller will not be optimal when either the contact nonlinearity is excited or if the model parameters change. Figure 17 demonstrates the suboptimal control performance when the same LQR controller is applied to dataset three, where the forcing leads to the contact nonlinearity being excited. The vertical lines in the bottom right panel of Figure 17 indicate instances where the nonlinearity is triggered, i.e., instances where the difference in displacement between floors two and three is less than the gap between the column and bumper, showing a similar number of instances as the passive system, which leads to the suboptimal vibration attenuation performance as the underlying linear modelling assumptions are invalid. It can be noticed that the performance of the controller has worsened particularly in the region between 10 Hz and 25 Hz, due to the higher frequency content of the contact nonlinearity. This represents a challenge for a digital twin, where the initial modelling assumptions may have been used to devise a control strategy that turns out to be suboptimal in the operational phase due to unforeseen structural and loading conditions.
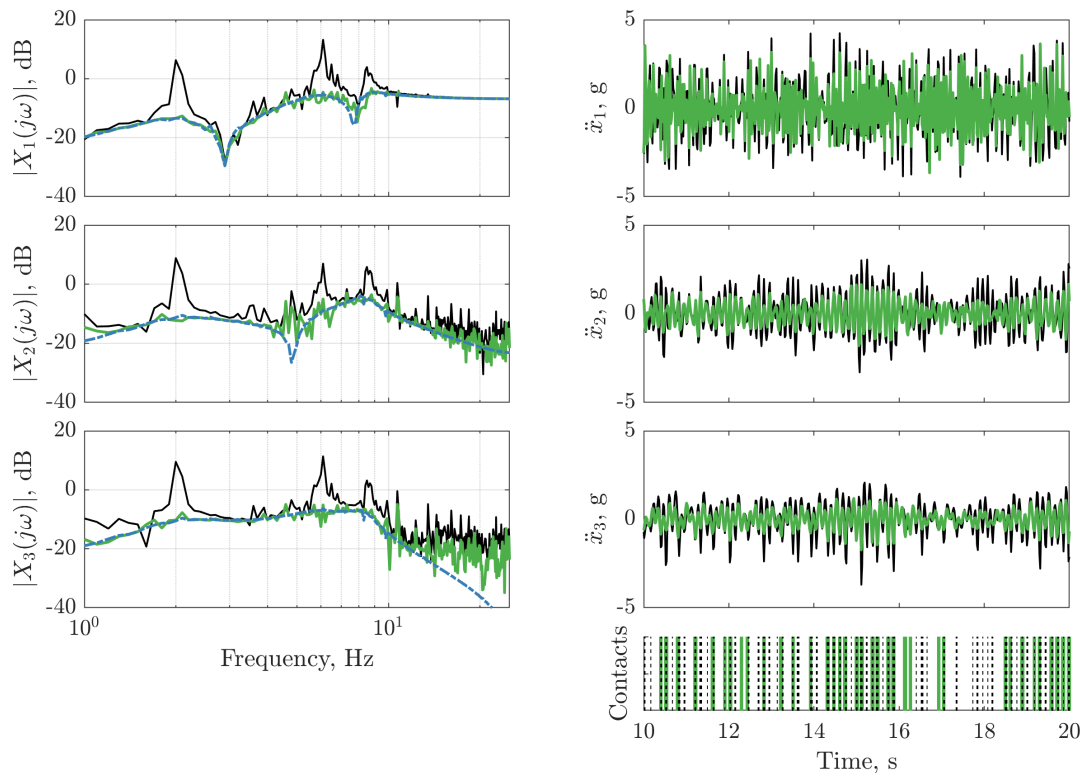
Figure 17. Frequency response function (left panels) and acceleration time history (right panels) for dataset three without (—), and with LQR controller (- -). The frequency response function of the LQR controller on dataset two (- -) is depicted (left panels) for reference. The vertical lines on the bottom right panel indicate instances where the nonlinearity is triggered for the passive structure (:) and the structure with the LQR controller (|).

At this stage the digital twin can inform the decision maker to take action to either switch off the controller to preserve its integrity, or to recalibrate the controller with the newly identified model parameters. The former case can be informed by the data-augmented model, given time instances where the digital twin expects performance to be poor. Given the active learning algorithm queries a large number of observations in dataset three, the forward predictions would have variances that regularly exceed the threshold. The output from the digital twin would therefore inform the engineer that contact between the bumper and column is likely to occur given the digital twins uncertainty, and the engineering can decide to increase the LQR gain, with the aim of preventing the contract nonlinearity from being excited, meaning the linear LQR assumptions could be kept valid. Figure 18 presents the case where the LQR feedback gain has been updated by increasing the state weight, $q = 5 \times 10^3$. This increase, based on the observations of the digital twin, has reduced the number of contact initiations to three, and as a result the vibration attenuation performance of the LQR controller has improved, being comparable for dataset three as to dataset two (where the nonlinearity was not excited). The change of the state weight q was done offline. An adaptive LQR can be obtained using a gain scheduling strategy, where the feedback gain matrix switches between different values depending on the variable that defines if the bumper is in contact or not. An adaptive control strategy based solely on the digital twin predictions remains a topic of future research.
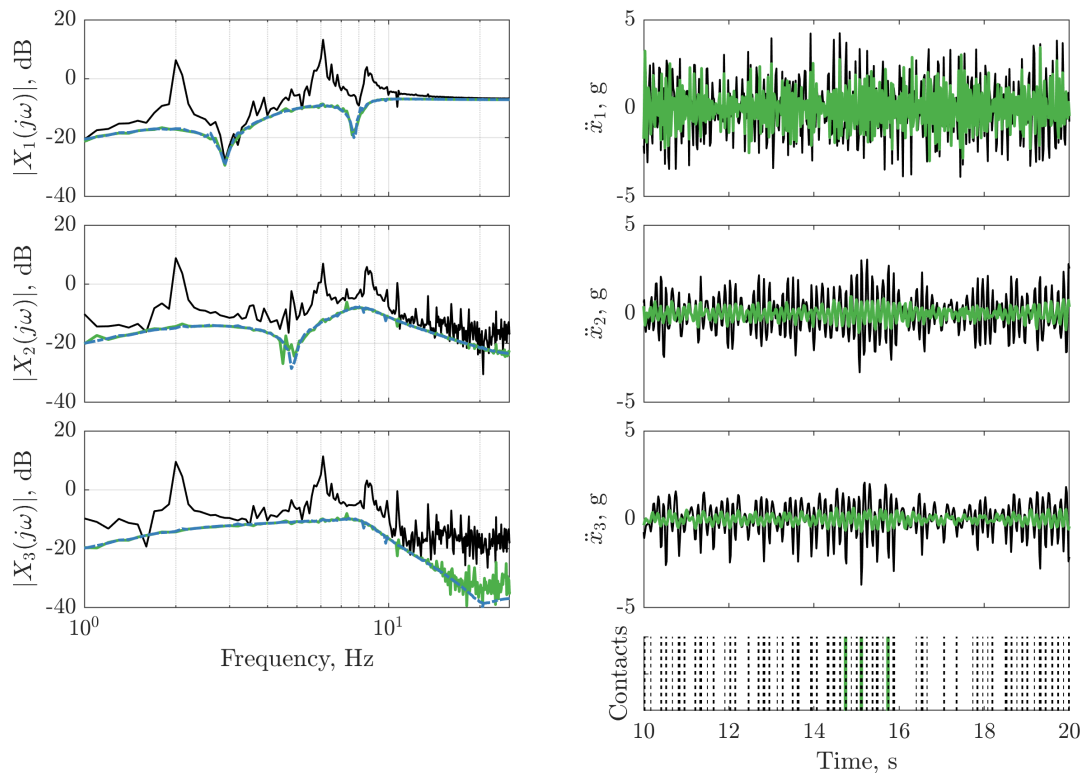
Figure 18. Frequency response function (left panels) and acceleration time history (right panels) for dataset three without (—), and with the updated LQR controller (- -). The frequency response function of the LQR controller on dataset two (- -) is depicted (left panels) for reference. The vertical lines on the bottom right panels indicate instances where the nonlinearity is triggered for the passive structure (:) and the structure with the updated LQR controller (|).

Clearly, by moving towards a digital twin, more information is available to inform decision making, particularly with respect to control. In this case study the data-augmented model has been used to inform that the control parameters should be updated. However, an area of further research is incorporating the digital twin model proposed in this paper as part of a model predictive control (MPC) strategy, where the digital twin directly affects control performance, forming an adaptive controller. Although this would be challenging to implement, it would have the benefit that any action the digital twin makes to improve predictive performance could be directly linked to an improvement in control performance.

7. Discussion

A digital twin will require some combination of models, data, knowledge and connectivity, as shown in Figure 1. This paper has explored aspects related particularly to data and models, with a discussion about how knowledge, such as that from hybrid testing, or from data-augmented models, can be connected to decisions such as model development or updated control strategies. A key concept introduced in this paper is that a digital twin will need to 'learn' over time, adapting to knew unforeseen events and structural states. This will require automated decision making, performing actions in an optimal and autonomous manner, based on available information. The path from a validated model

to a digital twin, therefore, involves numerous technical challenges, some of which have been addressed in the work outlined in this paper.

This paper has discussed that a digital twin must be more than performing model updating on a physics-based model. A digital twin will be required to overcome discrepancies between the known physics in the digital twin and the behaviour of the real physical system. In the long term it will be desirable for the underlying physics-based knowledge in the digital twin to expand, enabling the digital twin to confidently predict new scenarios. As the digital twin evolves, it will be necessary for data-based components to compensate for unmodelled phenomena. Machine learning provides such tools, enabling algorithms to augment the physics-based modelling, meaning that patterns from data can be incorporated into predictions, such as performed in Section 4. At this stage it is important to comment that quantifying the uncertainty in the machine learning component will be critical, as this informs the digital twin about the level of confidence it should place in the data-based component; as data-based models will only be able to mimic behaviour seen from data, and will generally be poor in extrapolation. Further research will be required in monitoring the range of behaviour currently explained by the physics-based and data-based components, as this may provide more information about which course of action, improving the physics- or data-based parts, provides the biggest gains in performance.

A continually adapting digital twin will benefit from a clear decision framework; where the active learning method, proposed in Section 4.3, would be part of a wider decision process. Digital twins will have to balance a number of desired tasks, in addition to predictive performance, whilst considering costs associated with each action, and the potential gain for each action. Costs for some actions may vary over time, and in relation to the level of detail required; for example a complex finite element model may provide accurate predictions, but be too costly to run in real-time. Understanding these costs and the utility associated with an action will allow a digital twin to progress from decisions based on heuristics, to actions that are guaranteed to minimise risk and maximum performance.

Coupling the digital twin with real-time hybrid testing is one potential avenue for isolating and targeting new physics, outlined in Section 5. As the digital twin improves, the numerical substructure will also improve, gaining further insight into the physical twin. A system where an online digital twin can submit a design of experiments request to a hybrid test rig in a laboratory environment, could further improve the automation of identifying new physics. However, restrictions in actuators and real-time controllers will limit the level to which the hybrid simulation can be solely used to develop new physics-based understanding. Consequently, other techniques will also need to be considered, as part of the wider knowledge map. Methods such as equation discovery [60], which select the most appropriate physics from data, given a range candidate equations, could provide useful additional methods available to a digital twin. Similar techniques, such as model selection [61], will also provide a way of determining whether new physics should be introduced, and are areas of further research.

It is worth noting that a digital twin will be used for a wider range of purposes than just making accurate predictions alone. However, all the concepts outlined in this paper will have an impact on other tasks. For example, the digital twin in this paper has been shown as having a positive impact on active control; discussed in Section 6. Other control strategies, such as model predictive control, may also benefit considerably from being linked to a digital twin. However, significant challenges remain in developing a control strategy based on modelling assumptions that evolve and update with time.

8. Conclusions

Digital twins offer a powerful new computational tool for aiding assets management decisions throughout the operational phase of a structure. This paper has outlined a path for progressing from a validated model towards a digital twin. A key requirement is that the digital twin is able to perform

actions, ideally autonomously. Several actions have been considered: obtaining better parameter estimates, updating a data-based component, the additions of new physics, or doing nothing. It has been argued that these actions form part of a learning strategy for a digital twin.

A data-based component is a required part of a digital twin, allowing unknown physics to be compensated for in predictions; one solution to how a digital twin accounts for missing physics. A Gaussian process model has been coupled with a physics-based model, allowing predictions to be augmented, overcoming poor predictive performance. In addition, a Gaussian process provides a measure of uncertainty in its functional form which has been used to help identify when the digital twins predictive performance is poor. Furthermore, by utilising this uncertainty, an active learning strategy has been proposed, allowing the data-based component to be updated based on informative observations in a stream-based manner. A significant challenge for a digital twin is identifying new physics. Hybrid testing has been proposed as one strategy for isolating substructures, and identifying problematic physics. The technology allows the digital twin to be coupled with a physical substructure, meaning tests can be conducted in a cost-effective manner, whilst remaining as realistic as possible. The ability for a digital twin to identify poor performance has been demonstrated to be useful in aiding control decisions. This property is important in understanding whether the assumptions in a control strategy are invalid or suboptimal.

Finally, this paper has highlighted several key stages in progressing towards a digital twin. Incorporating the multiple tasks and actions a digital twin can perform within a decision framework is seen as a crucial avenue of further research. By implementing such a framework, a digital twin will be able to 'learn', leading to optimal actions that not only keep the digital twin as a one-to-one mapping with the physical twin, but also reduce the risk of failure and increase the lifespan of the structure.

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, or in the decision to publish the results.

References

1. Wagg, D.; Worden, K.; Barthorpe, R.; Gardner, P. Digital Twins: State-of-the-Art and Future Directions for Modeling and Simulation in Engineering Dynamics Applications. *ASME J. Risk Uncertain. Part B* **2020**, *6*, 030901. [CrossRef]

2. Fuller, A.; Fan, Z.; Day, C.; Barlow, C. Digital Twin: Enabling Technologies, Challenges and Open Research. *IEEE Access* **2020**.

3. Jones, D.; Snider, C.; Nassehi, A.; Yon, J.; Hicks, B. Characterising the Digital Twin: A systematic literature review. *CIRP J. Manuf. Sci. Technol.* **2020**, *29*, 36–52. [CrossRef]

4. Lee, J.; Ni, J.; Djurdjanovic, D.; Qiu, H.; Liao, H. Intelligent prognostics tools and e-maintenance. *Comput. Ind.* **2006**, *57*, 476–489. [CrossRef]

5. Cerrone, A.; Hochhalter, J.; Heber, G.; Ingraffea, A. On the effects of modeling as-manufactured geometry: Toward digital twin. *Int. J. Aerosp. Eng.* **2014**, *2014*, 439278. [CrossRef]

6. Seshadri, B.R.; Krishnamurthy, T. Structural health management of damaged aircraft structures using digital twin concept. In Proceedings of the 25th AIAA/AHS Adaptive Structures Conference, Grapevine, TX, USA, 9–13 January 2017; p. 1675.

7.   Karve, P.M.; Guo, Y.; Kapusuzoglu, B.; Mahadevan, S.; Haile, M.A. Digital twin approach for damage-tolerant mission planning under uncertainty. *Eng. Fract. Mech.* **2020**, *225*, 106766. [CrossRef]

8.   Liu, Z.; Meyendorf, N.; Mrad, N. The role of data fusion in predictive maintenance using Digital Twin. *AIP Conf. Proc.* **2018**, *1949*, 020023.

9.   Aivaliotis, P.; Georgoulias, K.; Arkouli, Z.; Makris, S. Methodology for enabling digital twin using advanced physics-based modelling in predictive maintenance. *Procedia CIRP* **2019**, *81*, 417–422. [CrossRef]

10.   Werner, A.; Zimmermann, N.; Lentes, J. Approach for a Holistic Predictive Maintenance Strategy by Incorporating a Digital Twin. *Procedia Manuf.* **2019**, *39*, 1743–1751. [CrossRef]

11.   Kapteyn, M.G.; Knezevic, D.J.; Willcox, K. Toward predictive digital twins via component-based reduced-order models and interpretable machine learning. In Proceedings of the AIAA Scitech 2020 Forum, Orlando, FL, USA, 6–10 January 2020; p. 0418.

12.   Li, C.; Mahadevan, S.; Ling, Y.; Choze, S.; Wang, L. Dynamic Bayesian Network for Aircraft Wing Health Monitoring Digital Twin. *AIAA J.* **2017**, *55*, 930–941. [CrossRef]

13.   Grieves, M.; Vickers, J. Digital twin: Mitigating unpredictable, undesirable emergent behavior in complex systems. In *Transdisciplinary Perspectives on Complex Systems*; Springer: Cham, Switzerland 2017; pp. 85–113.

14.   Worden, K.; Cross, E.; Barthorpe, R.; Wagg, D.; Gardner, P. On Digital Twins, Mirrors, and Virtualizations: Frameworks for Model Verification and Validation. *ASME J. Risk Uncertain. Part B* **2020**, *6*, 030902. [CrossRef]

15.   Sharma, P.; Hamedifar, H.; Brown, A.; Green, R. The Dawn of the New Age of the Industrial Internet and How it can Radically Transform the Offshore Oil and Gas Industry. In Proceedings of the Offshore Technology Conference, Houston, TX, USA, 1–4 May 2017.

16.   Wanasinghe, T.R.; Wroblewski, L.; Petersen, B.; Gosine, R.G.; James, L.A.; De Silva, O.; Mann, G.K.; Warrian, P.J. Digital Twin for the Oil and Gas Industry: Overview, Research Trends, Opportunities, and Challenges. *IEEE Access* **2020**.

17.   Sivalingam, K.; Sepulveda, M.; Spring, M.; Davies, P. A Review and Methodology Development for Remaining Useful Life Prediction of Offshore Fixed and Floating Wind turbine Power Converter with Digital Twin Technology Perspective. In Proceedings of the 2018 2nd International Conference on Green Energy and Applications (ICGEA), Singapore, 24–26 March 2018; IEEE: New York, NY, USA, 2018; pp. 197–204.

18.   Tygesen, U.T.; Jepsen, M.S.; Vestermark, J.; Dollerup, N.; Pedersen, A. The True Digital Twin Concept for Fatigue Re-Assessment of Marine Structures. In Proceedings of the ASME 2018 37th International Conference on Ocean, Offshore and Arctic Engineering, Madrid, Spain, 17–22 June 2018; American Society of Mechanical Engineers: New York, NY, USA 2018; pp. V001T01A021.

19.   Pargmann, H.; Euhausen, D.; Faber, R. Intelligent big data processing for wind farm monitoring and analysis based on cloud-technologies and digital twins: A quantitative approach. In Proceedings of the 2018 IEEE 3rd International Conference on Cloud Computing and Big Data Analysis (ICCCBDA), Chengdu, China, 20–22 April 2018; IEEE: New York, NY, USA, 2018; pp. 233–237.

20.   Oñederra, O.; Asensio, F.; Eguia, P.; Perea, E.; Pujana, A.; Martinez, L. MV Cable Modeling for Application in the Digital Twin of a Windfarm. In Proceedings of the 2019 International Conference on Clean Electrical Power (ICCEP), Otranto, Italy, 2–4 July 2019; IEEE: New York, NY, USA, 2019; pp. 617–622.

21.   Glaessgen, E.H.; Stargel, D. The Digital Twin paradigm for future NASA and US Air Force vehicles. In Proceedings of the 53rd Structures, Structural Dynamics and Materials Conference, Honolulu, HI, USA, 23–26 April 2012; pp. 1–14.

22.   Zaccaria, V.; Stenfelt, M.; Aslanidou, I.; Kyprianidis, K.G. Fleet monitoring and diagnostics framework based on digital twin of aero-engines. In Proceedings of the Turbo Expo: Power for Land, Sea, and Air, Oslo, Norway, 11–15 June 2018; American Society of Mechanical Engineers: New York, NY, USA, 2018; Volume 51128, p. V006T05A021.

23.   Dawes, W.N.; Meah, N.; Kudryavtsev, A.; Evans, R.; Hunt, M.; Tiller, P. Digital Geometry to Support a Gas Turbine Digital Twin. In Proceedings of the AIAA Scitech 2019 Forum, San Diego, CA, USA, 7–11 January 2019; p. 1715.

24.   Patterson, E.A.; Taylor, R.J.; Bankhead, M. A framework for an integrated nuclear digital environment. *Prog. Nucl. Energy* **2016**, *87*, 97–103. [CrossRef]

25. Iglesias, D.; Bunting, P.; Esquembri, S.; Hollocombe, J.; Silburn, S.; Vitton-Mea, L.; Balboa, I.; Huber, A.; Matthews, G.; Riccardo, V.; et al. Digital twin applications for the JET divertor. *Fusion Eng. Des.* **2017**, *125*, 71–76. [CrossRef]

26. Volodin, V.; Tolokonskii, A. Concept of instrumentation of digital twins of nuclear power plants units as observers for digital NPP I&C system. *J. Phys. Conf. Ser.* **2019**, *1391*, 012083.

27. Chatzis, M.; Chatzi, E.; Triantafyllou, S. A Discontinuous Extended Kalman Filter for non-smooth dynamic problems. *Mech. Syst. Signal Process.* **2017**, *92*, 13–29. [CrossRef]

28. Kim, J.; Lynch, J.P. Subspace system identification of support excited structures—part II: Gray-box interpretations and damage detection. *Earthq. Eng. Struct. Dyn.* **2012**, *41*, 2253–2271. [CrossRef]

29. Chatzis, M.; Chatzi, E.; Smyth, A. An experimental validation of time domain system identification methods with fusion of heterogeneous data. *Earthq. Eng. Struct. Dyn.* **2015**, *44*, 523–547. [CrossRef]

30. Beck, J.L.; Katafygiotis, L.S. Updating models and their uncertainties. I: Bayesian statistical framework. *J. Eng. Mech.* **1998**, *124*, 455–461. [CrossRef]

31. Katafygiotis, L.S.; Beck, J.L. Updating models and their uncertainties. II: Model identifiability. *J. Eng. Mech.* **1998**, *124*, 463–467. [CrossRef]

32. Mottershead, J.E.; Link, M.; Friswell, M.I. The sensitivity method in finite element model updating: A tutorial. *Mech. Syst. Signal Process.* **2011**, *25*, 2275–2296. [CrossRef]

33. Simoen, E.; De Roeck, G.; Lombaert, G. Dealing with uncertainty in model updating for damage assessment: A review. *Mech. Syst. Signal Process.* **2015**, *56*, 123–149. [CrossRef]

34. Chatzi, E.N.; Smyth, A.W. Particle filter scheme with mutation for the estimation of time-invariant parameters in structural health monitoring applications. *Struct. Control Health Monit.* **2013**, *20*, 1081–1095. [CrossRef]

35. Kennedy, M.C.; O'Hagan, A. Bayesian calibration of computer models. *J. R. Stat. Soc. Ser. B (Stat. Methodol.* **2001**, *63*, 425–464. [CrossRef]

36. Berger, J.O.; Cavendish, J.; Paulo, R.; Lin, C.H.; Cafeo, J.A.; Sacks, J.; Bayarri, M.J.; Tu, J. A framework for validation of computer models. *Technometrics* **2007**, *49*, 138–154.

37. Brynjarsdóttir, J.; O'Hagan, A. Learning about physical parameters: The importance of model discrepancy. *Inverse Probl.* **2014**, *30*, 114007. [CrossRef]

38. Vapnik, V.N. *The Nature of Statistical Learning Theory*; Springer: New York, NY, USA, 1999.

39. Bishop, C.M. *Neural Networks for Pattern Recognition*; Oxford University Press, Inc.: New York, NY, USA, 1995.

40. LeCun, Y.; Bengio, Y.; Hinton, G. Deep learning. *Nature* **2015**, *521*, 436–444. [CrossRef]

41. Rasmussen, C.E.; Williams, C.K.I. *Gaussian Processes for Machine Learning*; MIT Press: Cambridge, MA, USA, 2006.

42. Stein, M.L. *Interpolation of Spatial Data: Some Theory for Kriging*; Springer Series in Statistics; Springer: New York, NY, USA, 1999.

43. Worden, K.; Surace, C.; Becker, W. Uncertainty bounds on higher-order FRFs from Gaussian process NARX models. *Procedia Eng.* **2017**, *199*, 1994–2000. [CrossRef]

44. Worden, K.; Becker, W.; Rogers, T.; Cross, E. On the confidence bounds of Gaussian process NARX models and their higher-order frequency response functions. *Mech. Syst. Signal Process.* **2018**, *104*, 188–223. [CrossRef]

45. Enqvist, M.; Ljung, L. Linear approximations of nonlinear FIR systems for separable input processes. *Automatica* **2005**, *41*, 459–473. [CrossRef]

46. Settles, B. Active learning literature survey. In *Computer Sciences Technical Report*; University of Wisconsin–Madison: Madison, WI, USA, 2009.

47. Bull, L.; Worden, K.; Manson, G.; Dervilis, N. Active learning for semi-supervised structural health monitoring. *J. Sound Vib.* **2018**, *437*, 373–388. [CrossRef]

48. Ho, S.; Wechsler, H. Query by transduction. *IEEE Trans. Pattern Anal. Mach. Intell.* **2008**, *30*, 1557–1571. [PubMed]

49. Loy, C.C.; Hospedales, T.M.; Xiang, T.; Gong, S. Stream-based joint exploration-exploitation active learning. In Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition, Providence, RI, USA, 16–21 June 2012; pp. 1560–1567.

50. Bedford, T.; Cooke, R. *Probabilistic Risk Analysis: Foundations and Methods*; Cambridge University Press: Cambridge, UK, 2001.

51. Berger, J. *Statistical Decision Theory and Bayesian Analysis*; Springer: New York, NY, USA, 1985.

52. Snelson, E.; Ghahramani, Z. Sparse Gaussian processes using pseudo-inputs. In Proceedings of the Advances in Neural Information Processing Systems, Vancouver, BC, Canada, 5–8 December 2005.

53. Titsias, M. Variational learning of inducing variables in sparse Gaussian processes. In Proceedings of the International Conference on Artificial Intelligence and Statistics, Clearwater, FL, USA , 16–19 April 2009.

54. Bui, T.D.; Yan, J.; Turner, R.E. A unifying framework for Gaussian process pseudo-point approximations using power expectation propagation. *J. Mach. Learn. Res.* **2017**, *18*, 3649–3720.

55. Bijl, H.; van Wingerden, J.W.; Schön, T.B.; Verhaegen, M. Online sparse Gaussian process regression using FITC and PITC approximations. *IFAC* **2015**, *48*, 703–708.

56. Bui, T.D.; Nguyen, C.; Turner, R.E. Streaming Sparse Gaussian Process Approximations. In Proceedings of the Advances in Neural Information Processing Systems, Long Beach, CA, USA, 4–9 December,2017; pp. 3299–3307.

57. Wagg, D.; Neild, S.; Gawthrop, P. Real-time testing with dynamic substructuring. In *Modern Testing Techniques for Structural Systems*; Springer: Wien, Austria, 2008; Volume 502.

58. Horiuchi, T.; Inoue, M.; Konno, T.; Namita, Y. Real-time hybrid experimental system with actuator delay compensation and its application to a piping system with energy absorber. *Earthq. Eng. Struct. Dyn.* **1999**, *28*, 1121–1141. [CrossRef]

59. Preumont, A. *Vibration Control of Active Structures*; Springer: Berlin, Germany 1997; Volume 2.

60. Fuentes, R.; Dervilis, N.; Worden, K.; Cross, E. Efficient parameter identification and model selection in nonlinear dynamical systems via sparse Bayesian learning. *J. Phys. Conf. Ser.* **2019**, *1264*, 012050. [CrossRef]

61. Beck, J.L.; Yuen, K.V. Model Selection Using Response Measurements: Bayesian Probabilistic Approach. *J. Eng. Mech.* **2004**, *130*, 192–203. [CrossRef]