Detection of Model Uncertainty in the Dynamic Linear-Elastic Model of Vibrations in a Truss

Alexander Matei^(✉) and Stefan Ulbrich

Department of Mathematics, Research Group Optimization,
Technical University of Darmstadt, Dolivostraße 15, 64293 Darmstadt, Germany
`matei@mathematik.tu-darmstadt.de`

Abstract. Dynamic processes have always been of profound interest for scientists and engineers alike. Often, the mathematical models used to describe and predict time-variant phenomena are uncertain in the sense that governing relations between model parameters, state variables and the time domain are incomplete. In this paper we adopt a recently proposed algorithm for the detection of model uncertainty and apply it to dynamic models. This algorithm combines parameter estimation, optimum experimental design and classical hypothesis testing within a probabilistic frequentist framework. The best setup of an experiment is defined by optimal sensor positions and optimal input configurations which both are the solution of a PDE-constrained optimization problem. The data collected by this optimized experiment then leads to variance-minimal parameter estimates. We develop efficient adjoint-based methods to solve this optimization problem with SQP-type solvers. The crucial test which a model has to pass is conducted over the claimed true values of the model parameters which are estimated from pairwise distinct data sets. For this hypothesis test, we divide the data into k equally-sized parts and follow a k -fold cross-validation procedure. We demonstrate the usefulness of our approach in simulated experiments with a vibrating linear-elastic truss.

Keywords: Model uncertainty · Optimum experimental design · Sensor placement · Optimal input configuration · k -fold cross-validation

1 Introduction

In science and technology, dynamic processes are often described by time-variant mathematical models. However, the accurate prediction of the motion and behavior of technical systems is still challenging. Due to our incomplete knowledge of the internal relations between model parameters, state variables and the time domain, the user frequently encounters model uncertainty [20]. In [12] we developed an algorithm to identify this model uncertainty using parameter estimation, optimal experimental design and classical hypothesis testing. It is the aim of this paper to extend this approach to dynamic models. In the following, we adopt the

framework described in [12] and extend it to meet mathematical models which are comprised of time-variant partial differential equations (PDE).

There is abundant literature on the assessment of descriptive and predictive qualities of dynamic models. Most common are techniques like residual analysis [27, 29] and interval simulation [25], maximum likelihood methods [29] and Bayesian model updating [31]. Our approach comes from a frequentist perspective and offers an alternative: we minimize the extent of data uncertainty by optimizing the experimental design and employ a k -fold cross-validation to test the model's fitness and consistency. The validation is hereby performed via a classical hypothesis test in the parameter space.

A subproblem that needs to be solved in our approach to detect model uncertainty is the PDE-constrained optimal experimental design (OED) problem where the PDE is time-dependent. We specifically focus on experiments where sensors need to be positioned and inputs must be chosen in order to achieve a maximum information gain for the estimated values of the model parameters. Optimal sensor placement has been addressed within the PDE-context in [1, 2, 23] and optimal input configuration has been extensively analyzed for both linear and nonlinear ordinary differential equations in various engineering applications [6, 17, 21, 22, 28]. However, in these cases the problem dimension is small compared to a (discretized) time-variant PDE and thus, gradient-based optimization with a sensitivity approach, as suggested by [4] and [19], works fine. In our case, this approach is no longer computationally tractable. Our framework meets high-dimensionality by employing efficient adjoint techniques in a sequential quadratic programming (SQP) solver scheme.

This paper is organized as follows. In Sect. 2 and Sect. 3, we introduce our model equations and briefly present the concepts of parameter estimation and OED followed by efficient solution techniques for the OED problem. Then, in Sect. 4 we show how our algorithm to detect model uncertainty is adapted to the dynamic setting. Section 5 contains numerical results for the OED problem applied to vibrations of a truss and the application of our algorithm to detect model uncertainty. We end the paper with concluding remarks.

2 Model Equations of Transient Linear Elasticity and Their Discretization

Let $G \subset \mathbb{R}^2$ be a bounded Lipschitz domain with sufficiently smooth boundary $\partial G = \Gamma_D \cup \Gamma_F \cup \Gamma_N$ where $\Gamma_D, \Gamma_F, \Gamma_N$ are pairwise disjoint and non-empty. Furthermore, let $(0, T)$, with $T > 0$, be an open and bounded time interval. We consider the parameter-dependent equations of motion for the linear-elastic body G of mass density $\varrho > 0$ and weak damping constant $a > 0$, see [15, Sec. 7.2]:

$$\begin{aligned}
 \varrho \partial_{tt}^2 y + a \varrho \partial_t y - \operatorname{div} \sigma(y, \partial_t y, p) &= 0, & \text{in } (0, T) \times G, \\
 y &= 0, & \text{on } (0, T) \times \Gamma_D, \\
 \sigma(y, \partial_t y, p) \cdot n &= 0, & \text{on } (0, T) \times \Gamma_F, \\
 \sigma(y, \partial_t y, p) \cdot n &= u, & \text{on } (0, T) \times \Gamma_N, \\
 y(0, \cdot) &= 0, & \text{in } G, \\
 \partial_t y(0, \cdot) &= 0, & \text{in } G.
 \end{aligned} \tag{1}$$

We include Rayleigh damping in our modeling by the generalized law of Hooke:

$$\sigma(y, \partial_t y, p) = \mathcal{C}(p)(\varepsilon(y) + b\varepsilon(\partial_t y)),$$

where $b > 0$ is the strong damping constant, $\varepsilon(y) = \frac{1}{2}(\nabla y^\top + \nabla y)$ is the linearized strain and $\mathcal{C} : \varepsilon \mapsto p_1 \cdot \operatorname{trace}(\varepsilon) I + 2p_2 \cdot \varepsilon$ is the fourth order elastic tensor, see also [7]. The parameters in this PDE are the well known Lamé constants $p = (\lambda_L, \mu_L) \in \mathbb{R}_+^2$. It is evident from (1) that the displacement $y : (0, T) \times G \mapsto \mathbb{R}^2$ is caused by the traction $u : (0, T) \times \Gamma_N \mapsto \mathbb{R}^2$ alone.

After adopting the weak formulation of (1) according to [15, Sec. 7.2] and [9] we perform a finite-dimensional approximation of this weak formulation known as the Galerkin ansatz. We employ standard quadratic finite elements for the elastic body G for the space discretization. Then the finite element approximation leads to the (high-dimensional) second-order ordinary differential equation

$$M \partial_{tt}^2 y(t) + C(p) \partial_t y(t) + A(p)y(t) - Nu(t) = 0, \tag{2}$$

with the stiffness matrix $A(p)$, the mass matrix M and the boundary mass matrix N . For the Rayleigh damping term, we introduce the damping matrix

$$C(p) := aM + bA(p), \tag{3}$$

where $a, b > 0$ are the damping constants as before.

We want to use a numerical time-update scheme with a predefined step size Δt to solve (2). Therefore, we rewrite (2) in the form

$$Ma_n + C(p)v_n + A(p)d_n - Nu_n = 0,$$

with the acceleration vector $a_n = \partial_{tt}^2 y(t_n)$, the velocities $v_n = \partial_t y(t_n)$ and the displacements $d_n = y(t_n)$ for time steps t_n , where $n = 1, \dots, n_t$, respectively. The implicit Newmark method is suitable to solve this equation. It can be implemented in the following way. First, we choose constants $\beta_N = \frac{1}{4}$, $\gamma_N = \frac{1}{2}$ for stability reasons, see [15, 30], and define with them other constants $\alpha_1, \dots, \alpha_6$ as depicted in [30, Sec. 6.1.2]. Then the iteration scheme reads as follows:

$$\begin{aligned}
 a_{n+1} &= \alpha_1(d_{n+1} - d_n) - \alpha_2 v_n - \alpha_3 a_n, \\
 v_{n+1} &= \alpha_4(d_{n+1} - d_n) + \alpha_5 v_n + \alpha_6 a_n, \\
 [\alpha_1 M + \alpha_4 C(p) + A(p)] d_{n+1} &= Nu_{n+1} + M(\alpha_1 d_n + \alpha_2 v_n + \alpha_3 a_n) \\
 &\quad + C(p)(\alpha_4 d_n - \alpha_5 v_n - \alpha_6 a_n).
 \end{aligned} \tag{4}$$

This scheme can be written in matrix form:

$$L(p)y - Fu = 0,$$

where $y = (y_1, \dots, y_{n_t})^\top$ are the states, $u = (u_1, \dots, u_{n_t})^\top$ are the boundary forces at all time points, $y_n = (a_n, v_n, d_n)^\top$ and $u_n = (u_{n,x}, u_{n,y})^\top$. The matrices L and F have the block form

$$L(p) = \begin{bmatrix} Q(p) & & & \\ P(p), X(p) & & & \\ & \ddots & \ddots & \\ & & P(p), X(p) & \end{bmatrix}, \quad F = \begin{bmatrix} E_0 & & & \\ & E_1 & & \\ & & \ddots & \\ & & & E_1 \end{bmatrix},$$

where

$$Q(p) = \begin{bmatrix} M, C(p), A(p) \\ I \\ I \end{bmatrix}, \quad X(p) = \begin{bmatrix} I, 0, -\alpha_1 I \\ 0, I, -\alpha_4 I \\ 0, 0, D(p) \end{bmatrix}, \quad E_0 = \begin{bmatrix} N \\ 0 \\ 0 \end{bmatrix}, \quad E_1 = \begin{bmatrix} 0 \\ 0 \\ N \end{bmatrix},$$

with $D(p) := \alpha_1 M + \alpha_4 C(p) + A(p)$ and

$$P(p) = \begin{bmatrix} \alpha_3 I, & \alpha_2 I, & \alpha_1 I \\ -\alpha_6 I, & -\alpha_5 I, & \alpha_4 I \\ -\alpha_3 M + \alpha_6 C(p), & -\alpha_2 M + \alpha_5 C(p), & -\alpha_1 M - \alpha_4 C(p) \end{bmatrix}.$$

For the following optimization problems let $Y = Z = \mathbb{R}^{n_y}$ be the state space and let

$$U_{\text{ad}} := \{u \in H^1(0, T; \Gamma_N) : [u(t)](x) = c(t) \text{ and } u_{\min} \leq u \leq u_{\max}\},$$

be the space of admissible inputs with $n_y = n_d n_t$ being the product of the space dimension after discretization n_d and the number of time steps n_t . Furthermore, let $e : Y \times \mathbb{R}_+^2 \times U_{\text{ad}} \rightarrow Z$ be an operator defining the *state equation* as

$$e(y, p, u) := L(p)y - Fu = 0 \quad (5)$$

and denote its unique solution by $y(p, u)$. We assume the operator $\partial_y e(y, p, u)$ to be continuously invertible such that we can use the Implicit Function Theorem to define a mapping $p \mapsto y(p, u)$. Its derivatives $s_i := \partial_{p_i} y(p, u)$ for $i = 1, 2$ are computed by solving

$$\partial_y e(y(p, u), p, u) s_i + \partial_{p_i} e(y(p, u), p, u) = 0,$$

which in our setting is equivalent to

$$L(p) s_i + \partial_{p_i} L(p) y(p, u) = 0, \quad i = 1, 2. \quad (6)$$

Thus, the sensitivity variable $s := [s_1, s_2] \in Y \times Y$ depends on the solution of the state equation and on the parameters, i.e., $s_i = s_i(y(p, u), p_i)$. Equations (6) are solved by rewriting them using the iteration scheme (4).

For the input space U_{ad} we employ a time discretization with linear finite elements and denote by M_T the mass matrix and by A_T the stiffness matrix in the time domain.

3 Lamé-Parameter Estimation and the Optimal Experimental Design Problem

Given a set of experimental data, we are concerned with an accurate estimation of the Lamé-parameters $p = (\lambda, \mu) \in \mathbb{R}_+^2$ which are part of the model equations. The measurements are taken at selected points on the discretized free boundary part Γ_F of the elastic body G with specified sensor types. We denote by n_s the number of available sensors. In order to compare the output of the model equations, i.e., the state, with experimental data we introduce a nonlinear *observation operator* $h: Y \rightarrow \mathbb{R}^{n_s n_t}$ that maps components of the state to quantities that are actually measured during the experiment at all n_t time steps.

Within the framework of optimal experimental design, we introduce binary weights $\omega \in \{0, 1\}^{n_s}$ for all sensor locations and types. These weights operate as a selection tool, i.e., $\omega_k = 1$ if, and only if, sensor k is used at its specified location. Since the position of these sensors and their usage throughout the experiments stay the same, the values of ω are copied n_t times and summarized in the diagonal matrix $\Omega \in \mathbb{R}^{n_z \times n_z}$, where $n_z = n_s n_t$. In addition, each sensor has a fixed operating precision, i.e., standard deviation, which we associate by the variable $\sigma_{pr} \in \mathbb{R}^{n_s}$. We again summarize n_t copies of σ_{pr} in a diagonal matrix $\Sigma \in \mathbb{R}^{n_z \times n_z}$.

The data $z \in \mathbb{R}^{n_z}$ is used to estimate the parameters $p \in \mathbb{R}_+^2$ by solving a least-squares problem:

$$\min_{p \in \mathbb{R}_+^2} \frac{1}{2} r(z, y(p, u))^\top \Omega \Sigma^{-2} r(z, y(p, u)), \quad (7)$$

where $r(z, y(p, u)) := h(y(p, u)) - z$ are the residuals and $y(p, u)$ is the unique solution of (5) for given p and u . Since the measurements are random variables $z = z^* + \varepsilon$ with unknown true values z^* and noise ε , so are the parameters. We model the noise to be Gaussian, i.e., $\varepsilon \in \mathcal{N}(0, \Omega^{-1} \Sigma^2)$. In a first order approximation, like in a Gauss-Newton solver scheme, the parameters are also Gaussian with unknown mean p^* and covariance matrix C , see [8, 19]. Then the confidence region of the parameters with a fixed confidence level $1 - \alpha$, where $\alpha \in (0, 1)$, is given by

$$K(p^*, C, \alpha) := \{p \in \mathbb{R}_+^2 : (p - p^*)^\top C^{-1} (p - p^*) \leq \chi_2^2(1 - \alpha)\}.$$

We assume that the solution $\bar{p}$ of (7) for given z and ω , emerging from the Gauss-Newton algorithm, is sufficiently close to p^* . Thus, we make the assumption that for a given data set, $\bar{p}$ is a fairly good approximation of p^* . Then the covariance matrix C can be approximated by employing the Gauss-Newton scheme as well and it has the following form [8]:

$$C_{GN} = [s(y(p, u), p)^\top \partial_y h(y(p, u))^\top \Omega \Sigma^{-2} \partial_y h(y(p, u)) s(y(p, u), p)]^{-1}.$$

We aim at minimizing the confidence region where the estimated parameters $\bar{p}$ lie:

$$K(\bar{p}, C_{GN}, \alpha) = \{p \in \mathbb{R}_+^2 : (p - \bar{p})^\top C_{GN}^{-1} (p - \bar{p}) \leq \chi_2^2(1 - \alpha)\} \rightarrow \min.$$

The reduction of the size of the confidence ellipsoid K is equivalent to reducing the “size” of the covariance matrix. This is realized by choosing best sensor locations, determined by the weights ω , and by finding optimal inputs u . In practice, there are various design criteria Ψ that measure the “size” of a matrix C , see [11]. In this paper we decide to use the E -criterion which is related to the maximal expansion of K :

$$\Psi(C) = \Psi_E(C) = \lambda_{\max}(C) \sim \text{diameter}(K)^2.$$

We add a cost term $P_\varepsilon(\omega)$ to penalize the number of used sensors and a regularizer $R(u) := u^\top (M_T + A_T)u$ to the objective function. Moreover, we relax the binary restriction on ω to employ gradient-based solution techniques for the following optimal experimental design problem.

Definition 1. Let $p \in \mathbb{R}_+^2$ be an estimate of p^* and let $\kappa, \beta > 0$ be fixed. Furthermore, choose $\varepsilon \in (0, 1]$. Then we call $(\bar{\omega}, \bar{u})$ an optimal design of an experiment with the linear-elastic body G if it is the solution of

$$\min_{\omega, u, y, s} \Psi(C_{\text{GN}}(\omega, y, s)) + \kappa \cdot P_\varepsilon(\omega) + \beta \cdot R(u), \quad (8)$$

where (u, y, s) are subject to the equality constraints

$$\begin{aligned} L(p)y - Fu &= 0, \\ L(p)s_i + \partial_{p_i} L(p)y &= 0, \end{aligned} \quad (9)$$

for $i = 1, 2$ and (ω, u) satisfy the inequality constraints

$$\omega \in [0, 1]^{n_s}, \quad u \in U_{\text{ad}}. \quad (10)$$

The penalty term $P_\varepsilon(\omega)$ is a smooth approximation of the l_0 -“norm”. It ensures sparse solutions in ω for suitable choices of κ but does not lead to $\{0, 1\}$ -valued weights yet. To achieve the latter, we adopt a continuation strategy as described in [1, 2].

Note, that the penalty parameter κ must not be chosen too large since the matrix C_{GN} becomes singular if too many weights ω are switched to zero. We refer to [19] for more details on lower bounds for the sum of the weight variables.

In practice, problem (8)–(10) is solved using its reduced formulation, i.e., by eliminating the equality constraints (9) and inserting $y(p, u)$ and $s(y(p, u), p)$ into the objective function.

3.1 Derivative and Adjoint Computation

Let $J(\omega, u, y, s_1, s_2)$ be the objective function in (8). We show how the derivative of the reduced objective function $\hat{J}(\omega, u)$, where the solutions $y(p, u)$ and $s(y(p, u), p)$ of (9) have been inserted into J , with respect to the inputs u is efficiently computed. To do so, we follow a standard Lagrangian view of the optimization problem (8)–(10). For simplicity, we ignore the inequality constraints (10) and still denote by $\partial_y \Psi$ the derivative of Ψ with respect to y even

though we used the Clarke directional derivatives in the case of $\Psi = \Psi_E$, cf. [13]. Let $\mu, \lambda_1, \lambda_2 \in Y^*$ be Lagrange multipliers and let the Lagrangian be defined as

$$\begin{aligned} \mathcal{L}(\omega, u, y, s_1, s_2, \mu, \lambda_1, \lambda_2) := & J(\omega, u, y, s_1, s_2) + \langle \mu, L(p)y - Fu \rangle_{Y^*, Y} \\ & + \sum_{i=1}^2 \langle \lambda_i, L(p)s_i + \partial_{p_i} L(p)y \rangle_{Y^*, Y}. \end{aligned}$$

The adjoint equations follow from $\partial_y \mathcal{L} = \partial_{s_i} \mathcal{L} = 0$ for $i = 1, 2$:

$$\begin{aligned} L(p)^\top \mu + [\partial_{p_1} L(p)]^\top \lambda_1 + [\partial_{p_2} L(p)]^\top \lambda_2 + \partial_y \Psi &= 0, \\ L(p)^\top \lambda_1 + \partial_{s_1} \Psi &= 0, \\ L(p)^\top \lambda_2 + \partial_{s_2} \Psi &= 0. \end{aligned} \tag{11}$$

The fact that the matrix $L(p)$ is transposed on the left hand side of (11) leads to an iteration scheme backwards in time. We demonstrate this for the second and third adjoint equations in order to obtain λ_i whereby adopting ideas from [18, Sec. 5.4]. Let $\lambda = \lambda_i$ and $r := \partial_{s_i} \Psi$ for $i \in \{1, 2\}$. Note that $r = (r_1, \dots, r_{n_t})^\top$ and $r_n = (r_n^d, 0, 0)$ since the velocities and accelerations do not enter Ψ . In the terminal point t_{n_t} we have to solve

$$X(p)^\top \lambda_{n_t} = r_{n_t},$$

or equivalently $\lambda_{n_t}^a = 0, \lambda_{n_t}^v = 0$ and

$$-\alpha_1 \lambda_{n_t}^a - \alpha_4 \lambda_{n_t}^v + D(p) \lambda_{n_t}^d = r_{n_t}^d.$$

For other time points t_n , $n \neq 1$ the current iterate is obtained from the one which is a step forward in time:

$$[X(p)^\top, P(p)^\top] \begin{pmatrix} \lambda_n \\ \lambda_{n+1} \end{pmatrix} = r_n,$$

or equivalently

$$\begin{aligned} \lambda_n^a &= \alpha_3 \lambda_{n+1}^a + \alpha_6 \lambda_{n+1}^v + [\alpha_3 M - \alpha_6 C(p)] \lambda_{n+1}^d, \\ \lambda_n^v &= \alpha_2 \lambda_{n+1}^a + \alpha_5 \lambda_{n+1}^v + [\alpha_2 M - \alpha_5 C(p)] \lambda_{n+1}^d, \\ -\alpha_1 \lambda_n^a - \alpha_4 \lambda_n^v + D(p) \lambda_n^d &= r_n^d - \alpha_1 \lambda_{n+1}^a - \alpha_4 \lambda_{n+1}^v + [\alpha_1 M + \alpha_4 C(p)] \lambda_{n+1}^d. \end{aligned}$$

In order to obtain the adjoint variable at the initial time point t_1 we solve

$$[Q(p)^\top, P(p)^\top] \begin{pmatrix} \lambda_1 \\ \lambda_2 \end{pmatrix} = r_1,$$

or equivalently

$$\begin{aligned} M \lambda_1^a &= \alpha_3 \lambda_2^a + \alpha_6 \lambda_2^v + [\alpha_3 M - \alpha_6 C(p)] \lambda_2^d, \\ C(p) \lambda_1^a + \lambda_1^v &= \alpha_2 \lambda_2^a + \alpha_5 \lambda_2^v + [\alpha_2 M - \alpha_5 C(p)] \lambda_2^d, \\ A(p) \lambda_1^a + \lambda_1^d &= r_1^d - \alpha_1 \lambda_2^a - \alpha_4 \lambda_2^v + [\alpha_1 M + \alpha_4 C(p)] \lambda_2^d. \end{aligned}$$

The matrix vector product $q := [\partial_{p_i} L(p)]^\top \lambda_i$ is computed likewise using the iteration scheme.

Finally, the full derivative of the reduced objective function $\hat{J}(\omega, u)$ with respect to the inputs u is given by

$$\frac{d\hat{J}}{du} = -F^\top \mu + 2\beta(M_T + A_T)u,$$

where $\mu \in Y^*$ is the adjoint variable obtained from (11).

3.2 Computational Remarks

In order to solve (8)–(10) we employ an SQP algorithm with BFGS updates [10] for the Hessian H_k of the Lagrangian. We modify the update formula in the following way:

$$H_0 = \begin{bmatrix} I \\ 2\beta(M_T + A_T) \end{bmatrix},$$

$$H_{k+1} = H_k - \frac{H_k d^k (H_k d^k)^\top}{(d^k)^\top H_k d^k} + \frac{r^k (r^k)^\top}{(r^k)^\top d^k},$$

where d^k is the current step, y^k is the difference between gradients of the Lagrangian at the new and old iterate and

$$r^k := \begin{cases} y^k & \text{if } (y^k)^\top d^k \geq 0.2(d^k)^\top H_k d^k, \\ \theta y^k + (1 - \theta)H_k d^k & \text{otherwise,} \end{cases}$$

with $\theta = \frac{0.8(d^k)^\top H_k d^k}{(d^k)^\top H_k d^k - (y^k)^\top d^k}$. After every tenth iteration we reset the Hessian to H_0 to avoid matrix filling and to ensure a gradient descent with respect to ω from time to time.

4 Detection of Uncertainty in Dynamic Models

We adopt the algorithm presented in [12] and describe the main differences when applied to a time-variant model $\mathcal{M}$ of a dynamic process. In general, we presuppose that a valid model should reproduce *all* measurements obtained with *all* admissible inputs at *all* sensor locations with the *same* set of parameters. Our approach is summarized in Algorithm 1.

First, initial (or artificial) data z_{ini} is needed for an appropriate guess p_{ini} of the parameter values. Having fixed these parameters, one can solve the OED problem (8)–(10) to obtain best sensor positions $\bar{\omega}$ and optimal input configurations $\bar{u}$, see lines 02 and 03.

Algorithm 1. (Detection of Model Uncertainty, adapted from [12])**Input:** Model $\mathcal{M}$, test level TOL (e.g. 5%), number of test scenarios n_{tests} .**Output:** Does $\mathcal{M}$ need to be rejected? YES (1) or NO (0).

```

01: Initialize  $i := 1$ .
02: Generate initial data  $z_{\text{ini}}$  in all feasible sensor locations.
03: Calculate  $p_{\text{ini}}$  from (7) using  $z_{\text{ini}}$ . Fix  $p = p_{\text{ini}}$  and solve (8)–(10). Obtain  $(\bar{\omega}, \bar{u})$ .
04: Acquire data  $z$  in optimal sensor locations  $\bar{\omega}$  for inputs close to the optimum  $\bar{u}$ .
05: Check whether measurement errors are Gaussian. Otherwise go to line 04 or exit.
06: Divide  $z$  into one calibration set  $z_{\text{cal}}$  and one validation set  $z_{\text{val}}$ .
07: Calculate  $(p_{\text{cal}}, C_{\text{cal}})$  using  $z_{\text{cal}}$ . Likewise, obtain  $p_{\text{val}}$  using  $z_{\text{val}}$ .
08: Determine  $\alpha_{\text{min}} \in (0, 1)$ , such that  $p_{\text{val}}$  lies on boundary of  $K(p_{\text{cal}}, C_{\text{cal}}, \alpha_{\text{min}})$ .
09: if  $\alpha_{\text{min}} \geq \text{TOL}$  then
10:   if  $i < n_{\text{tests}}$  then
11:      $i := i + 1$ . Go to line 06.
12:   else
13:     return 0.
14:   end if
15: else if  $\alpha_{\text{min}} < \text{TOL}$  then
16:   return 1.
17: end if

```

The acquisition of experimental data z in line 04 is done at the optimal sensor locations and for inputs close to the optimum. Since we assume that the true values of the model parameters remain the *same* for all inputs $u \in U_{\text{ad}}$, we can ensure that our data are truly divers by performing measurements for different input values within a small neighborhood of the optimum $\bar{u}$. Evidently, the size of the confidence ellipsoid K stays small because of continuity of the objective function (8) with respect to the inputs.

Recall, that for time-variant systems each measurement at a given time depends on the past. Since the order of the data is important, the splitting of z into one calibration and one validation set must not happen over the time axis. Since our methodology is different from forecasting [5] we do not allow such splittings over the time domain.

We perform the division regarding the *different inputs* in a k -fold cross-validation manner [16]. Divide the data into k groups where each group is distinguished by the input for which it was collected in the whole time domain. We then use $k - 1$ groups for calibration and the remaining group for validation. When repeating this procedure we run through all k possible combinations.

For the validation itself, we perform a classical hypothesis test from line 08 onward as documented in [12]. The threshold TOL is identical to the error of the first kind. It is common to set a 5% limit to this error. The α_{min} which is computed in line 08 is the p -value of the statistical test. This is the smallest test level for which the null hypothesis can only just be rejected.

There is no need to account for the problem of multiple testing here, since we are using a k -fold cross-validation manner to divide the data which ensures pairwise disjoint validation sets.

5 Numerical Results for Simulated Vibrations of a Truss

We employ a 2D-truss consisting of nine beams and six connectors with about 5 000 spatial degrees of freedom in order to exemplify the application of Algorithm 1. The Dirichlet boundary Γ_D is located at the two outer top connectors and the Neumann boundary Γ_N on the bottom left connector, see Fig. 1. We use pairs of strain gauges as sensors that can measure either the axial deflection or the displacement caused by bending of the beams, see [14] and [26]. The strain gauges are located on the upper and lower boundaries of the beams, indicated as black bullets and connecting lines in the figure, which are part of the free boundary Γ_F of the body G . Each strain gauge measures the relative displacement of two adjacent nodes: $\varepsilon_u = y_{N1} - y_{N2}$ and $\varepsilon_\ell = y_{N3} - y_{N4}$, see Fig. 1a. For simplicity, we compute the square of the axial deflection $h_a(y)$ and the square of the displacement caused by bending $h_b(y)$:

$$h_a(y) = \frac{1}{4} \|\varepsilon_u + \varepsilon_\ell\|^2, \quad h_b(y) = \frac{1}{4} \|\varepsilon_u - \varepsilon_\ell\|^2.$$

Thus, the overall observation operator h consists of h_a and h_b at all time points and we create for each such sensor five weight variables. These additional weights shall give the experimenter information about which pairs of strain gauges are more important than others. The discretization of the truss allows for 117 sensors in total. Hence, we have $n_s = 117 \times 2 \times 5 = 1\,170$ weight variables.

Throughout our numerical simulations we use pure stiffness damping, i.e., $a = 0$ in (3). This is promising to provide better resemblance with actual experimental data, see [3] and [24]. The accuracy of our sensors is fixed to $\sigma_{pr,k} = 10\,\mu\text{m}$ for $k = 1, \dots, n_s$.

We simulated vibrations of the truss for $n_t = 600$ time steps with a step size of $\Delta t = 5$ ms. Thus, three seconds were simulated in total and the solution of the PDE (1) involves about 3 000 000 degrees of freedom. Initially, there were all 117 pairs of strain gauges used measuring both the axial deflection and the displacement caused by bending with maximum weight, respectively. We also use a constant maximally feasible force as a starting point for the inputs u . The excitation forces u act solely on the Neumann boundary Γ_N .

Since we were not able to conduct real experiments, all the data was simulated, i.e., generated on the computer with random numbers. Thus, line 05 in Algorithm 1 became obsolete. We assume the beams of the *real truss* $\mathcal{R}$ to have an equal cross-sectional area in the displacement-free state except for two beams having a 5% and a 7% smaller diameter, respectively. For the detection of model uncertainty it is not important to know which beams differ from the standard diameter. However, *our model* $\mathcal{M}$ operates on the assumption that all

Fig. 1. (a) snapshot of the truss with all possible locations for the strain gauges marked as bullets with connecting lines and the excitation force displayed by a red arrow, (b) snapshot after problem (8)–(10) has been solved with displayed optimal positions for strain gauges and optimal excitation

beams have the same cross-sectional area. This directly impacts the mathematical terms in the mass, damping and stiffness matrices, see (2), since a model with different cross-sectional beam areas would induce other finite element terms. It is our aim in this section to show that Algorithm 1 successfully detects model uncertainty in $\mathcal{M}$ when compared to $\mathcal{R}$.

Since we only simulate experiments, we skipped line 02 in Algorithm 1 and adopted textbook values for p_{ini} , namely, the well-known Lamé-constants for steel $\lambda_L = 121\,154 \text{ N/mm}^2$ and $\mu_L = 80\,769 \text{ N/mm}^2$. These are the values which we use to generate all measurements from the real truss $\mathcal{R}$. Problem (8)–(10) is solved after about 80 iterations with an overall computation time of about 8 h on an AMD EPYC $48 \times 2.8 \text{ GHz}$ machine. The design criterion decreased by $\approx 99\%$ which means that the maximal expansion of the confidence ellipsoid decreased by $\approx 98\%$ compared to the initial design, see Fig. 2. The final design employs only two pairs of strain gauges that measure the axial deflection, the upper with weight two the lower with weight five, cf. Fig. 1b.

Let $\bar{u}$ be the optimal input force obtained from solving (8)–(10). For the application of the hypothesis test in Algorithm 1, consider the following perturbed inputs:

$$\begin{aligned} u_1(t) &= \bar{u}(t) + \delta_1, & u_2(t) &= \bar{u}(t) + 4 \sin(t/(2\pi)), \\ u_3(t) &= \bar{u}(t) + 4 \cos(t/(2\pi)), & u_4(t) &= \bar{u}(t) + \delta_2(t), \\ u_5(t) &= \bar{u}(t) + \delta_3(t), & u_6(t) &= \bar{u}(t) \cdot (1 + 0.06 \sin(t/(2\pi))), \\ u_7(t) &= \bar{u}(t) \cdot (1 + 0.06 \cos(t/(2\pi))), & u_8(t) &= \bar{u}(t) + \delta_4, \end{aligned}$$

where $\delta_1, \delta_4 \sim \mathcal{N}(0, 4 \cdot I)$ and $\delta_2(t), \delta_3(t) \sim \mathcal{N}(0, 4t/n_t \cdot I)$ for all $t \in \{t_1, \dots, t_{n_t}\}$ with equal time step size Δt as introduced before. With these inputs we generate

Fig. 2. Optimization results for problem (8)–(10): (a) first order optimality (—) and norm of the step (.....), (b) objective function (— $\times$ —), design criterion (— $\blacktriangle$ —), penalty (— $\blacksquare$ —) and regularization value (— $\circ$ —)

eight different data sets and perform an 8-fold cross-validation. We use seven sets for calibration and one set for validation in line 06 of Algorithm 1. Thus, we conducted eight different hypothesis tests, four of which are shown in Table 1. It is clearly seen, that the model $\mathcal{M}$ does not pass any test when a threshold of 5% is applied to $\alpha_{\min}$. According to our assumption that a valid model should reproduce *all* measurements conducted with *all* admissible inputs with the *same* set of parameters, this is a significant indication of model uncertainty.

Table 1. Excerpt of the results for the hypothesis tests from Algorithm 1

#	p_{cal}	p_{val}	$\ p_{\text{cal}} - p_{\text{val}}\ $	$\lambda_{\min}(C)^{-1}$	$\alpha_{\min}$ in %
1	$4.588 \cdot 10^{-5}$	$1.206 \cdot 10^5$	$1.253 \cdot 10^5$	1.073	$\ll 0.001$
	$1.130 \cdot 10^5$	$7.909 \cdot 10^4$			
2	$4.562 \cdot 10^{-5}$	$1.206 \cdot 10^5$	$1.253 \cdot 10^5$	1.073	$\ll 0.001$
	$1.130 \cdot 10^5$	$7.909 \cdot 10^4$			
3	$4.340 \cdot 10^{-5}$	$1.206 \cdot 10^5$	$1.253 \cdot 10^5$	1.073	$\ll 0.001$
	$1.130 \cdot 10^5$	$7.909 \cdot 10^4$			
4	$6.167 \cdot 10^{-5}$	$1.206 \cdot 10^5$	$1.253 \cdot 10^5$	1.073	$\ll 0.001$
	$1.130 \cdot 10^5$	$7.909 \cdot 10^4$			

6 Conclusion

In this paper we showed that our algorithm to detect model uncertainty, which was first presented in [12], is applicable to dynamic models. We efficiently solved the OED problem with time-dependent PDE-constraints using modified BFGS-updates and adjoint methods within an SQP solver scheme. Thus, in finding optimal sensor positions and optimal inputs we were able to significantly reduce the size of the confidence region of the estimated model parameters. By an 8-fold cross-validation using hypothesis tests in the parameter space, we demonstrated on simulations of vibrations in a truss that our algorithm is able to detect inaccuracies of the linear-elastic model which is deficient in the geometrical description of the truss. It is the object of further investigation to show that our algorithm detects other forms or kinds of model uncertainty as well.

Acknowledgements. This research was funded by the German Research Foundation (DFG) – project number 57157498 – CRC 805 within the subproject A3. We would like to thank the DFG for funding.

The authors would also like to thank Philip Kolvenbach for providing efficient finite element code.

References

1. Alexanderian, A., Petra, N., Stadler, G., Ghattas, O.: A-optimal design of experiments for infinite-dimensional Bayesian linear inverse problems with regularized ℓ_0 -sparsification. *SIAM J. Sci. Comput.* **36**(5), A2122–A2148 (2014)
2. Alexanderian, A., Petra, N., Stadler, G., Ghattas, O.: A fast and scalable method for A-optimal design of experiments for infinite-dimensional Bayesian nonlinear inverse problems. *SIAM J. Sci. Comput.* **38**(1), A243–A272 (2016)
3. Alipour, A., Zareian, F.: Study rayleigh damping in structures; uncertainties and treatments. In: *Proceedings of the 14th World Conference on Earthquake Engineering*, Beijing, China, pp. 1–8 (2008)
4. Bauer, I., Bock, H.G., Körkel, S., Schlöder, J.P.: Numerical methods for optimum experimental design in DAE systems. *J. Comput. Appl. Math.* **120**(1–2), 1–25 (2000)
5. Bergmeir, C., Benítez, J.M.: On the use of cross-validation for time series predictor evaluation. *Inf. Sci.* **191**, 192–213 (2012)
6. Chianeh, H.A., Stigter, J., Keesman, K.J.: Optimal input design for parameter estimation in a single and double tank system through direct control of parametric output sensitivities. *J. Process Control* **21**(1), 111–118 (2011)
7. Ciarlet, P.G.: *Mathematical Elasticity. Studies in Mathematics and its Applications*, vol. 20. North-Holland Publishing Co., Amsterdam (1988)
8. Donaldson, J.R., Schnabel, R.B.: Computational experience with confidence regions and confidence intervals for nonlinear least squares. *Technometrics* **29**(1), 67–82 (1987)
9. Evans, L.C.: *Partial Differential Equations. Graduate Studies in Mathematics*, vol. 19, 2nd edn. American Mathematical Society, Providence (2010)
10. Fletcher, R.: *Practical Methods of Optimization*, 2nd edn. A Wiley-Interscience Publication. John Wiley & Sons Ltd, Chichester (1987)

11. Franceschini, G., Macchietto, S.: Model-based design of experiments for parameter precision: state of the art. *Chem. Eng. Sci.* **63**(19), 4846–4872 (2008)
12. Gally, T., Groche, P., Hoppe, F., Kuttich, A., Matei, A., Pfetsch, M.E., Rakowitzsch, M., Ulbrich, S.: Identification of model uncertainty via optimal design of experiments applied to a mechanical press. *Optim. Eng.* (2021). <https://doi.org/10.1007/s11081-021-09600-8>
13. Hiriart-Urruty, J.B., Lewis, A.S.: The Clarke and Michel-Penot subdifferentials of the eigenvalues of a symmetric matrix. *Comput. Optim. Appl.* **13**(1), 13–23 (1999)
14. Hoffmann, K.: An Introduction to Stress Analysis Using Strain Gauges (1987)
15. Hughes, T.J.R.: The Finite Element Method. Prentice Hall Inc., Hoboken (1987)
16. James, G., Witten, D., Hastie, T., Tibshirani, R.: An Introduction to Statistical Learning, vol. 112. Springer, Heidelberg (2013)
17. Jauberthie, C., Bournonville, F., Coton, P., Rendell, F.: Optimal input design for aircraft parameter estimation. *Aerosp. Sci. Technol.* **10**(4), 331–337 (2006)
18. Kolvenbach, P.: Robust optimization of PDE-constrained problems using second-order models and nonsmooth approaches. Ph.D. thesis, TU Darmstadt (2018)
19. Körkel, S., Kostina, E., Bock, H.G., Schlöder, J.P.: Numerical methods for optimal control problems in design of robust optimal experiments for nonlinear dynamic processes. *Optim. Methods Softw.* **19**(3–4), 327–338 (2004)
20. Mallapur, S., Platz, R.: Uncertainty quantification in the mathematical modelling of a suspension strut using Bayesian inference. *Mech. Syst. Signal Process.* **118**, 158–170 (2019). <https://doi.org/10.1016/j.ymssp.2018.08.046>
21. Mehra, R.: Optimal input signals for parameter estimation in dynamic systems—survey and new results. *IEEE Trans. Autom. Control* **19**(6), 753–768 (1974)
22. Morelli, E.A., Klein, V.: Optimal input design for aircraft parameter estimation using dynamic programming principles. In: Proceedings of the 17th Atmospheric Flight Mechanics Conference (1990). <https://doi.org/10.2514/6.1990-2801>
23. Neitzel, I., Pieper, K., Vexler, B., Walter, D.: A sparse control approach to optimal sensor placement in PDE-constrained parameter estimation problems. *Numer. Math.* **143**(4), 943–984 (2019)
24. Otani, S.: Nonlinear dynamic analysis of reinforced concrete building structures. *Can. J. Civ. Eng.* **7**(2), 333–344 (1980)
25. Puig, V., Quevedo, J., Escobet, T., Nejjari, F., de las Heras, S.: Passive robust fault detection of dynamic processes using interval models. *IEEE Trans. Control Syst. Technol.* **16**(5), 1083–1089 (2008)
26. Rohrbach, C.: Handbuch für elektrisches Messen mechanischer Größen. VDI-Verlag, Düsseldorf (1967)
27. Simani, S., Fantuzzi, C., Patton, R.J.: Model-based fault diagnosis techniques. In: *Model-Based Fault Diagnosis in Dynamic Systems Using Identification Techniques*, pp. 19–60. Springer (2003)
28. Stigter, J.D., Keesman, K.J.: Optimal parametric sensitivity control of a fed-batch reactor. *Automatica* **40**(8), 1459–1464 (2004)
29. Willsky, A.S.: Detection of abrupt changes in dynamic systems. In: *Detection of Abrupt Changes in Signals and Dynamical Systems*, pp. 27–49. Springer (1985)
30. Wriggers, P.: *Nonlinear Finite Element Methods*. Springer Science & Business Media, Hoboken (2008)
31. Yuen, K.V., Kuok, S.C.: Bayesian methods for updating dynamic models. *Appl. Mech. Rev.* **64**(1) (2011). <https://doi.org/10.1115/1.4004479>

Open Access This chapter is licensed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.