

Technical Paper

# Decomposing deviations of scanned surfaces of sheet metal assemblies

Felix Claus<sup>a,\*</sup>, Bernd Hamann<sup>b</sup>, Heike Leitte<sup>a</sup>, Hans Hagen<sup>a</sup>

<sup>a</sup> Department of Computer Science, TU-Kaiserslautern, Erwin-Schrödinger-Straße 52, 67663 Kaiserslautern, Germany

<sup>b</sup> Department of Computer Science, University of California, Davis, CA 95616-8562, USA

## ARTICLE INFO

## Keywords:

3D-metrology  
FE-simulation  
Parameter-estimation

## ABSTRACT

We present a finite-element-based approach to support quality assurance for assembly processes of non-rigid sheet metal parts in the automotive and aerospace industries. These parts have adjustable mechanical boundaries, making it possible to compensate geometrical deviations caused by the production process of components. Every part produced must be adjusted individually. Determining what boundaries to adjust to and in what direction is challenging and requires experience. Our goal was to identify sources of measured surface deviations after part assembly and provide valuable information to the quality assurance engineer. We solve the corresponding parameter estimation problem via a least-squares approach. From the estimated parameter states that led to measured deviations, additional visualizations are calculated to support the inspection engineer when deriving countermeasures.

## 1. Introduction

Measuring the dimensions of produced parts is one of the most important aspects of quality assurance. We focus on the assembly of exterior chassis components of a car body. In sheet metal assembly processes in today's automotive industry, the dimension assessment often is condensed by measuring single points on an assembled part. These arbitrary points, referred to as KPIs (key performance indicators) have to be chosen carefully to obtain a satisfactory assessment of the assembly quality. KPIs are chosen such that gap and flushness between parts are controlled as these measures are highly important for the car's aesthetics. Unfortunately, gaps and flushness need tight tolerances and are difficult to tune in. These tolerances are below the dimensional variability of the parts being assembled. Adjustable boundaries are necessary to fine-tune gap and flushness for every individual assembly. To support this kind of adaptive assembly, for instance, expensive tools with mounted measurement devices are used to control KPIs during the process. However, additional inspection steps are needed to evaluate such automated assembly processes. During the detailed analysis, more inspection points are evaluated, in addition to the KPIs that are actively controlled during the assembly. To increase the amount of information obtained by a single measurement, optical measurement methods become important for assessing sheet metal assemblies. Optical measurement technology enables capturing the whole visible surface of an assembly with a high-resolution point cloud. A point cloud provides us

with detailed information about the geometry of a measured part but, at the same time, the measurement data is hard to interpret. Often, 3D-scans are more difficult to interpret as a comparison between desired geometry and measured geometry would result in a 3D vector field. For assessment one commonly uses a condensed scalar field, the signed distance field, plotted in the 3D domain. The amount of information is reduced making it easier for the inspection engineer to analyze acquired point clouds to identify effects like dents, buckling, or elevations of large areas. However, identifying the reason for observed effects is not straightforward. Naming misplaced boundaries and the necessary amount of correction requires a quality engineer with a lot of experience and a well-understood assembly component. To improve this decision-making process, our approach supports the inspection engineer by making the results of optical measurements easier to understand. This is achieved by a decomposition of the measured offset vector field into components that relate to the adjustable mechanical boundaries of the assembly. With the provided output of our method, an engineer can more easily identify potential sources of deviations to derive and apply countermeasures. Besides that, the method is also capable of making decisions and could be used as a kernel for setting up a self-compensating assembly line for exterior chassis assembly.

To get an understanding of how the method complements and is used as part of a production environment, Fig. 1 shows a high-level overview. The shown pipeline starts with a real part with a geometry that differs from the desired ideal CAD geometry. This part is scanned before the assembly process and a finite element (FE) simulation considering the

\* Corresponding author.

E-mail addresses: [claus@cs.uni-kl.de](mailto:claus@cs.uni-kl.de) (F. Claus), [hamann@cs.ucdavis.edu](mailto:hamann@cs.ucdavis.edu) (B. Hamann), [leitte@cs.uni-kl.de](mailto:leitte@cs.uni-kl.de) (H. Leitte), [hagen@informatik.uni-kl.de](mailto:hagen@informatik.uni-kl.de) (H. Hagen).

<https://doi.org/10.1016/j.jmsy.2021.08.011>

Received 4 March 2021; Received in revised form 25 August 2021; Accepted 26 August 2021

Available online 8 September 2021

0278-6125/© 2021 The Author(s). Published by Elsevier Ltd on behalf of The Society of Manufacturing Engineers. This is an open access article under the CC BY

license (<http://creativecommons.org/licenses/by/4.0/>).

Notation	
$a$	simulation parameter
$\vec{B}$	basis vector field
$\vec{u}$	displacement vector field
$c$	dimensionless factor

measured geometry is performed to determine optimal assembly parameters for the measured geometry. The part will be attached to the chassis, and a post-assembly measurement is performed to assess assembly quality. The measured post-assembly geometrical shape is compared to the FE-simulation results. The resulting differences are evaluated with our method. Fig. 1 shows two potential use cases. The first case involves a manual process control loop: The inspection engineer performs adjustments to the process controller, informed by the plots that result from the method. The second use case is based on an automatic process control loop: The parameter estimation capabilities of our method are used to apply adjustments to the process controller directly.

## 2. Background

This section summarizes the topics that are related and relevant to the work presented in this paper. In particular, the topics discussed in the following are:

- Free-form surface inspection
- Causal analysis of sheet metal assembly errors
- Parameter estimation
- Proper orthogonal decomposition (POD) and modal decomposition
- Assembly simulation and smart assembly

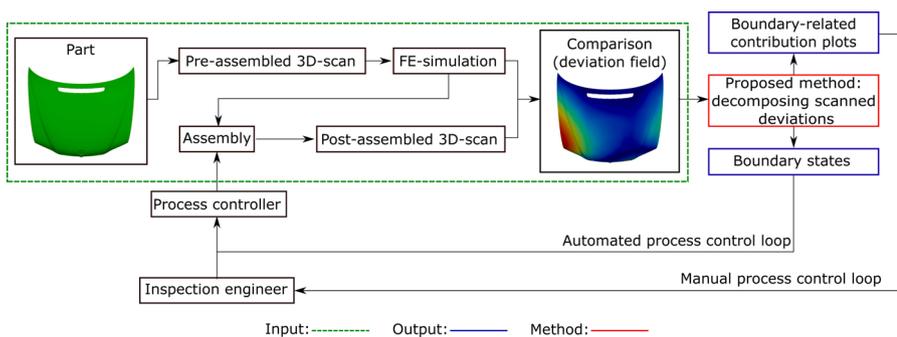
In the past, free-form surface inspections of sheet metal parts were a challenging task in terms of acquiring measurement data. Traditionally, the inspection of free-form surfaces has been carried out by measuring multiple points with tactile coordinate measurement machines (CMMs) for instance see [1]. To overcome the limitations of tactile measurement systems, recently developed approaches focus on the application of 3D-metrology for assessing sheet metal surfaces. These measurement systems are capable of acquiring several million vertices within seconds, and they support a high-resolution, detailed analysis of a surface. Nevertheless, using automated 3D-scanning technologies for assessing sheet metal parts leads to computational challenges (time and data size) to be managed. To reduce information loss due to effects like reflections and shadows, many approaches attempt to optimize the amount and location of viewpoints, i.e., see [2,3]. Due to the number of vertices that can be acquired in a short time, recent work focuses also on the integration of optical measurement systems for in-line inspection. Kiraci

et al. [4] discuss a concept for using a LIDAR Sensor mounted on a robotic arm for measuring a chassis in-line. Bergström et al. [5] use photogrammetry for recognizing smaller sheet metal part components when they are on a moving belt. The method presented by Denkena et al. [6] is a surface inspection system based on a structured light scan that is used for a complete inspection for stamped parts. Concerning state-of-the-art robot-mounted measurement systems, Babu et al. [7] presents a sampling method that reduces drastically the number of necessary measurements for recognizing a part's surface and the time needed for measurement. Our paper focuses on the free-from inspection performed with optical measurement machines. This method of acquiring geometrical information of an assembly provides the input for the method we propose.

Based on the acquired measurements, analyses are performed by the inspection engineer to identify potential causes for measured deviations. This kind of analysis is called Root Cause Analysis (RCA). Once the reason(s) is(are) identified, countermeasures are derived and applied. The concept of RCA is highly related to the topic of this paper, as we also aim to identify and visualize reasons for measured deviations. Carlson and Söderberg [8] uses RCA for reducing the dimensional variation in multi-stage assemblies. For visualization, [9] are using spectrograms to show interdependencies between manufacturing parameters. In sheet metal assembly, typical causal factors are, for example, assembly tool variation, part variation, the order of fastening screws or spot welds. Published approaches also deal with optimizing these identified weak spots, i.e., see [10–14].

RCA is mainly used to find unknown sources of variance in the system, and parameter estimation approaches deal with the determination of specific error values for known influence factors. In contrast, parameter estimation is widely used in different applications for quantify known variables in simulation models, see [15–17]. For FE-simulations one estimates parameter values for fine-tuning material parameters of the simulation model. The parameter estimator is applied to experimental data, see [18,19]. Reina et al. [20] use parameter estimation as a kernel of a closed-loop control system for automotive purposes to predict critical motion states. The most relevant branch of parameter estimation methods used for our work is based on least-squares methods. An overview of these methods is given in [21]. More recently the parameter estimation is also solved by using state-of-the-art machine learning and neural network approaches, e.g., the ones presented by Sinha et al. [22] and Leonhardt et al. [23]. The authors also discuss their method in the context of quality inspection in sheet metal assembly processes.

To set up a parameter estimator, a large data set containing the necessary information is decomposed into main contributing components. These components are obtained with different methods. The most commonly used method is proper orthogonal decomposition (POD). POD is a branch of dimension reduction methods for finding a lower-dimensional basis for high-dimensional data. A review of POD methods is given in [24]. PODs are used in a wide variety of applications. For example, Bui-Thanh et al. [25] use decomposition for



**Fig. 1.** Overview of example assembly process and two possible control loops. As input serves a comparison between simulated and measured assembly geometry (green). The proposed method (red) predicts from the deviation field, corresponding boundary states (blue) as well as boundary-related plots (blue). It can support automated process control as well as human-in-the-loop process control by providing additional insight into the measured deviation field. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

reconstructing pressure fields on aerofoils from measured point data. Babu et al. [7] use POD for reducing the amount of necessary single optical measures for recognizing a sheet metal part by reconstructing non-measured areas. The principle of POD provides a foundation for the method proposed in this paper. The method from the branch POD used in this paper is the principal component analysis (PCA). For example, Lindau et al. [26] use PCA to predict statistical shape models.

A physical-based way of decomposing the solution space of a part's behavior is to perform a modal decomposition. Modal decomposition is the most relevant approach when large ensembles of training data are not applicable. The assumption is based on this principle: Every deflection of a part can be expressed as a linear combination of eigenmodes. Modal decomposition is used for a wide variety of applications see [27–30]. Also, this principle is used for improving sheet metal assembly processes as presented by Adragna et al. [31]. In this work, the authors introduce the phrase “Modal Tolerancing”. Here the modal decomposition is used to obtain the contribution of single eigenmodes to explain present variance. The goal is to generate better insights into complex part behavior for the inspection engineer. This goal and approach are very similar to the work this paper presents. However, eigenmodes cannot be used to generate plots that are related to specific boundary states.

Another relevant topic discussed in the literature is assembly simulation and smart assembly. Söderberg et al. [32] simulated geometrical variation in non-rigid assemblies to improve decision-making processes. For our approach predicting the post-assembly shape is crucial, as it has been designed to point out differences between ideal (simulated) and real (measured) assemblies. As the geometry of every individual part is subject to variation due to manufacturing processes, recent work focuses on the simulation of the post-assembly shape by considering assembly process steps and part geometry measured before assembly. For instance, Gentilini and Shimada [33] used optical measurements and FE-simulations to predict the post-assembly shape of a measured part. [34] presented an approach for an individualized sheet metal assembly based on measurements of the assembly components. Here, the goal was to individualize the process parameters of the assembly which is also called “smart assembly” or “self-compensating” assembly process. A similar approach that specializes in the assembly of exterior sheet metal parts was presented in [35]. Also, a digital twin reference model explicitly for smart assembly processes was developed by Yi et al. [36].

In summary, high-resolution digital scans are becoming increasingly important for quality assurance processes for sheet metal assembly. Predicting post-assembly shapes based on optical measurements and adjusting process parameters is a current research topic. Our approach focuses on the quality assurance loop, especially the interpretation of post-assembly 3D-scans compared to a predicted simulation result. Least-squares parameter estimation and POD are utilized to identify and quantify the sources of measured deviations. Additionally, our approach visualizes contributions of single boundaries to a measured displacement field based on a systematic physical-based decomposition related to modal decomposition.

We briefly compare the proposed method with previously mentioned methods. The approaches described by Adragna et al. [31] and Babu et al. [7] are especially similar, in terms of the method itself and the goal. Babu et al. [7] use the same PCA decomposition of a training data set to reconstruct shapes of partially scanned sheet metal parts. The difference to our method is the fact that their goal is the reconstruction of the geometry of an individual part, whereas we are interested in assessing post-assembly shapes. We need to link the PCA basis to adjustable process parameters that are reconstructed as well. Adragna et al. [31] use a modal decomposition to find a physically meaningful basis to provide an understandable decomposition of measured deviations. Their approach is similar as far as method, goal, and application are concerned; but our approach is different as we generate the basis for the approximation differently. This has the benefit that we can relate observed deviations directly to adjustable boundaries instead of

eigenmodes that would require further interpretation to devise countermeasures. A minor drawback of this approach is a costly generation of the basis.

The novelty of our method is the decomposition of measured displacement into contributions of adjustable boundaries that are plotted on the parts surface mesh. The key contributions are:

- Method for identifying and visualizing errors in sheet metal assemblies caused by wrong positioning of attachment points.
- Decomposition of measured displacement fields into portions contributed by each mechanical boundary con-straining the part.
- Improvement of decision-making processes for quality assurance of exterior assemblies.

This paper is organized as follows: Section 3 provides a detailed description of the method and the calculation steps. Section 4 describes a use case as well as the application of the method to simulation data, 4.2, and experimental data, 4.3. Results are provided and discussed in Section 5, followed by conclusions in Section 6.

### 3. Method

Our goal is to decompose a displacement vector field generated by comparing a measurement of a post-assembly geometry with the predicted ideal geometry. As an example, one can consider an FE-simulation performed with the pre-assembly shape to predict the optimal assembly. The described decomposition of the vector field is not a vector field decomposition in the mathematical sense. It should be viewed as a reconstruction based on multiple FE-simulations using varying parameter states.

We provide an overview of our approach with a flowchart and a detailed description of the mathematical aspects. Fig. 2 shows the workflow that considers “One-time preparation,” “Input” and “Output.” The path marked in orange is the “critical path.” Steps 1–7 along this path decompose the input field. The workflow is explained at a high level, but we also cover the mathematical details. As one-time preparatory steps, we set up an FE-simulation model of the part we must inspect. This model needs to be validated to ensure that it reflects the behavior of the real part. The geometry of the simulation model is generated via a CAD system. Although the geometric stiffness slightly differs between the simulation model and the real part, we assume that this difference in stiffness is negligible for our purposes. This aspect, with examples, is discussed in [37]. Nevertheless, the geometrical deviations of a real part relative to CAD model must be taken into account during the pre-processing step of the input to isolate measured displacements caused by misplaced boundaries, see Fig. 1. Next, we need to identify the parameters that are considered adjustable during the assembly process. When these parameters and their ranges are known, the simulation is performed with a systematic variation of parameter values. The resulting ensemble of simulation results is used to generate two different bases for least-squares-fitting. Both are capable of approximating the whole solution space within a defined tolerance. One basis is set up by performing a PCA for the results of the whole simulation ensemble. The basis generated by PCA is an ensemble of computationally generated synthetic displacement fields with corresponding parameter states. This basis is ideal for estimating the parameter states of the input displacement field. The second basis is generated by a hierarchical selection of important simulation runs from the ensemble. This basis is not ideal for approximating the parameter states, but it has the significant advantage that each single basis field can be assigned to one parameter. The desired contribution plots can be generated from this basis.

We now consider the input to our method. The input is a displacement field calculated by comparing a measurement with the desired ideal. This comparison results in a scalar distance field. To obtain a displacement vector field all scalar values at the simulation's mesh

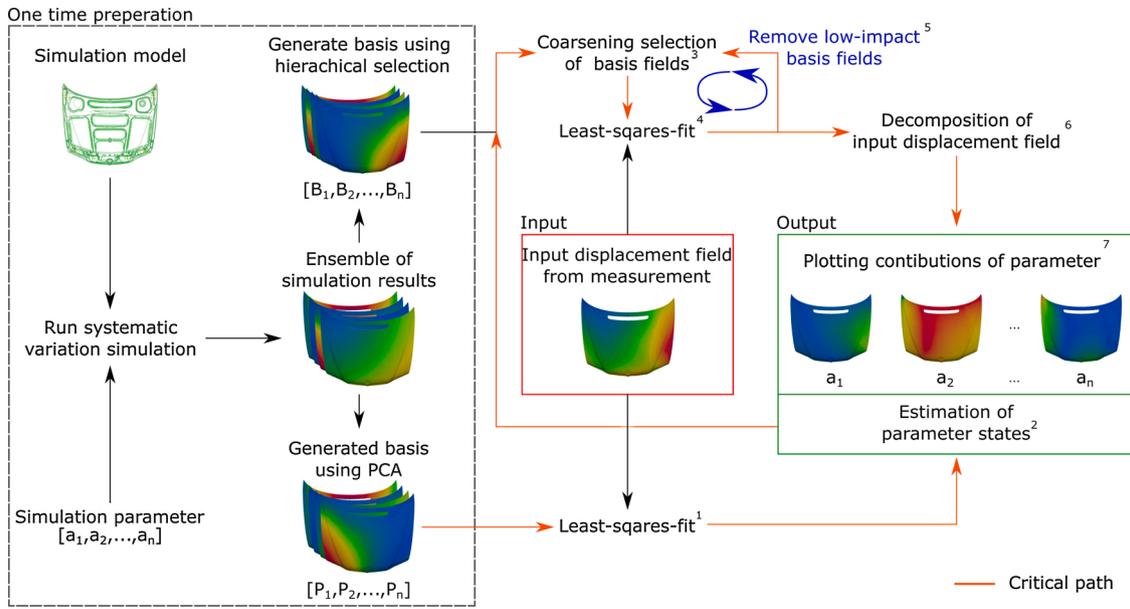


Fig. 2. Workflow. One-time preparation, input and output of the method. The one-time preparation step concerns simulation set-up and performing simulations to create basis fields. The input displacement field is approximated via a least-squares step.

nodes are multiplied with an associated normal vector. The calculation of the displacement magnitude is not unique, as distance values can vary slightly when changing reference and target geometries. This problem is depicted in Fig. 3. For our purposes, we require results for every node of the simulation mesh. The distance metric used is the “Cloud to Mesh Distance” implemented in the software package “CloudCompare” [38]. It calculates the distance from each vertex of the source mesh to the nearest triangle of the target mesh. The position of the vertex and the triangle’s outward normal are used to (i) determine whether the vertex is inside or outside the triangle and (ii) compute a signed distance to the plane containing the triangle. One could consider using more accurate and complex distance measures like the Hausdorff distance [39]. Point set registration could also be performed [40]. We use the simple CloudCompare metric as it provides sufficiently precise results for our application. For more complex geometries a different metric might be necessary.

We now explain the steps to process one input field. The 1–7 along the critical path are discussed. In (1), the input displacement field is used as a target for a least-squares-fit using the basis generated by PCA. This fit produces “pre-factors” for each basis from which we can estimate the parameter states (2) by multiplying the pre-factors with the corresponding basis parameters. These estimated parameters are one output that is communicated to the user and, at the same time, used for further calculations. Based on the estimated parameters a subset of basis fields (3) is selected from the hierarchically generated basis by excluding the basis fields that cannot be present for the current input. For instance, if the parameter estimation of (2) reveals that the parameter  $a_1$  is zero, all basis fields of the hierarchical basis with a non-zero value for  $a_1$  can be excluded. With this subset selected from the hierarchical basis, another least-squares-fit (4) is performed, also having the input displacement

field as a target. The resulting pre-factors are analyzed and further basis fields with minor contributions are excluded iteratively (5). This step is necessary to reduce the effects needed to explain the input displacement field to a minimum. The actual decomposition is calculated per node (6), the contribution of each basis field to the total measured displacement. Finally, the contribution plots (7) are generated.

### 3.1. Preparation steps

We use a simulation model that has  $n$  adjustable mechanical boundaries  $a_i$ . The values of these boundaries are between specific minimal and maximal values; they are parameters. Formally, we use the following notation:

$$\vec{a} = (a_1, a_2, \dots, a_n), a_i \in [\min_i, \max_i]. \quad (1)$$

Combinations of parameters  $\vec{a}$  are called parameter states. Given parameter states, an FE-simulation is performed, calculating a displacement vector field  $\vec{u}$ :

$$\vec{u} = \vec{u}(a_1, a_2, \dots, a_n). \quad (2)$$

We express the resulting displacement field  $\vec{u}$  as a function of the input parameters  $\vec{a}$  as  $\vec{u}(\vec{a})$ . By varying the parameters  $\vec{a}$  within the defined intervals, displacement fields  $\vec{u}_k(\vec{a}_k)$  are generated. These vector fields define the solution space for all simulation results, where the values of the parameters  $a_i$  are in the interval  $[\min_i, \max_i]$ . The solution space  $\vec{u}_k(\vec{a}_k)$  is decomposed via PCA, for example, to express the solution space as a linear combination of a small number of vector fields, i.e.,

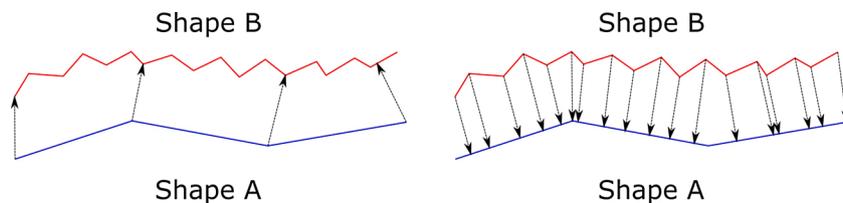


Fig. 3. Calculating distances between two meshed shapes can lead to different results depending on the choice of distance metric.

$$\vec{u} \approx (c_1, c_2, \dots, c_m) \begin{pmatrix} \vec{B}_1 \\ \vec{B}_2 \\ \vdots \\ \vec{B}_m \end{pmatrix} = c_1 \vec{B}_1 + c_2 \vec{B}_2 + \dots + c_m \vec{B}_m = \sum_{i=1}^m c_i \vec{B}_i, \tag{3}$$

where every  $B_i$  is a displacement basis field generated by PCA and  $c_i$  is a pre-factor of the linear combination. To define the vector  $\vec{c}$  a least-squares step is performed with a target displacement field created from a measurement as input. To determine a suitable basis  $\vec{B}$  different approaches are applicable. Common approaches often use PCA for this purpose.

### 3.1.1. Sampling distribution for simulation ensemble

To generate basis fields using PCA we first sample the solution space  $\vec{u}_k$ . This step is done by selecting discrete points in the parameter space  $\vec{a}_k$  and performing simulations for every selected parameter state. In our approach, we sample the whole parameter space by varying the parameters  $a_1$  to  $a_n$  using the values of min,0,max. This sampling strategy is illustrated in Fig. 4 for one n-dimensional hypercube of the parameter space for  $n = 3$ . As shown, only the corners of every cube are sampled. This sampling scheme is called full-factorial sampling. The sampling resolution of three values per axis might seem to be coarse, but the number of sampling points grows exponentially with the number of parameters. Six parameters already result in  $3^6 = 729$  possible combinations and simulation runs to be performed. Depending on the output behavior of the simulation, the sampling resolution must be adjusted or chosen adaptively. For our purpose uniform sampling is sufficient.

For the example shown in Fig. 4, the number of samples is  $3^3 = 27$ . As each sample point can be written as a vector  $\vec{a}$ , the information is presented compactly in Table 1. In this table, different “Ranks” of the vectors  $\vec{a}$  are introduced by counting the number of non-zero values in a vector  $\vec{a}$ . For each sample point, a simulation is performed to create the required ensemble of simulation results.

### 3.1.2. Generating the bases for least-square-fitting

As shown in Fig. 2, two different bases are necessary for the proposed method. One basis is created via PCA applied to the solution space of the created simulation ensemble. In this case, the whole ensemble is decomposed into its principal components. These principal components are represented as displacement fields with associated parameter states. These basis fields are synthetically generated by the PCA method and do not represent an actual simulation. A critical parameter that must be chosen depending on the simulation ensemble is the number of components the PCA should use. This number is obtained by plotting the variance explained by the generated basis.

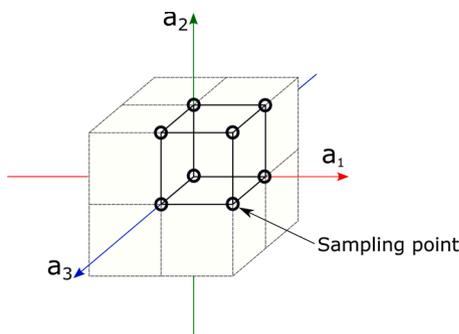


Fig. 4. Locations of chosen sample points for basis fields. Points in parameter space are placed at positions defined by combining maximum, minimum and zero parameter values.(This figure was published in [41] licensed under Attribution 4.0 International (CC BY 4.0), see <https://creativecommons.org/licenses/by/4.0/>, changes: variable names).

We now discuss the particular steps performed to generate such a basis using PCA applied to an ensemble of 3D vector fields. First, all vector fields of the simulation ensemble must be “condensed” to scalar fields. This is achieved by calculating, displacement magnitudes per node. To reduce information loss, we use the signed magnitude along the surface normal. This value is computed per node by calculating the scalar product of the displacement vector and the surface normal. After condensing the whole ensemble of displacement fields to scalar fields, PCA is performed. The result of the PCA computation is a set of weights for each principal component. Every principal component is a linear combination of the ensemble of scalar fields. Instead of applying the calculated weights to the scalar fields, we define the linear combination with the non-condensed 3D vector fields from the simulations. The linear combination defines one displacement field per principal component. Again, the same weights are used to calculate an analogous linear combination in parameter space. Every principal component is a pair of synthetically created 3D vector field data and associated parameter states.

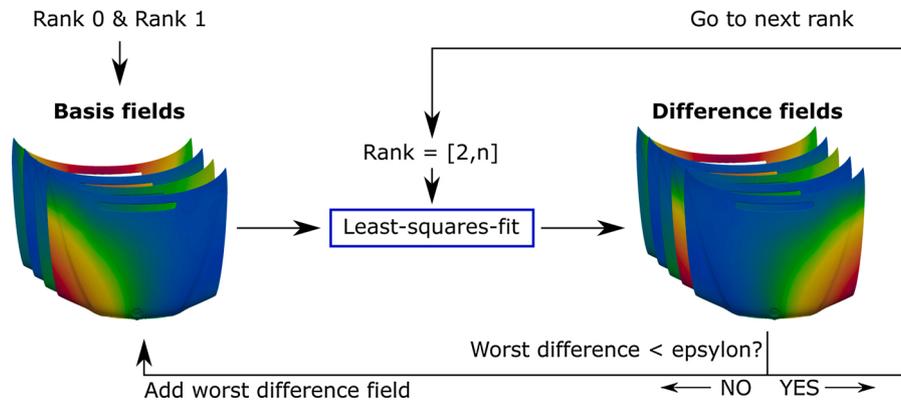
The second basis needed for the proposed method is generated by a hierarchical selection of important simulation results from the ensemble. The workflow of this selection is depicted in Fig. 5. The motivation for a hierarchical selection is the goal to find simulations that can be used as bases for a linear least-squares-fit and are related to single effects at the same time. The objective of the hierarchical selection is to automatically discover the given ensemble of simulation results from simple to complex parameter states. The steps performed during this automatic discovery are explained next. First, an initial basis is chosen that consists only of simulation results that have parameter states, located on the axis of the parameter space (Rank 0 and Rank 1 – see Table 1). This initial basis is assumed to be almost orthogonal as only one parameter is varied per simulation. Second, the initial basis is used to approximate all simulation results of rank two via a least-squares-fit. The result of each approximation is compared to each actual simulation. Each comparison produces a “Difference Field.” We select the difference field with the largest (absolute) difference. When the magnitude is larger than a tolerance  $\epsilon$  a new effect must be considered for approximating the current rank within the defined tolerance. The initial basis is expanded by adding the difference field with the largest difference. This new effect can also be interpreted as a new axis in parameter space. As the sampling of the parameter space of the original data set is uniform (see Fig. 4), a second difference field with inverse parameter states exists, located on the same axis. This difference field with inverse parameter states is added to the basis. Two corresponding fields per effect are present, covering both axis directions. The added difference fields are considered essential for capturing the new effect that cannot be explained by the previous basis. This second step is repeated until all simulation results from Rank two are approximated within the tolerance  $\epsilon$ . Third, all displacement fields from rank three are approximated with the new basis, and the iterative process is repeated until all ranks are approximated within the defined tolerance. Choosing the value for  $\epsilon$  is highly dependent on the required precision of an application.

### 3.2. Parameter estimation and contribution plots

The generated bases are used for approximating a measured displacement field via a least-squares-fit. This fitting step determines the best pre-factors  $\vec{c}$  to approximate a target displacement field  $\vec{u}_T$  by a linear combination of the basis fields  $\vec{B}_i$ . The index “T” refers to the target of the least-squares-fit. For the presented use case the target displacement field is obtained by comparing a post-assembly measurement with the desired geometry. All three displacement directions X, Y, and Z are considered for the least-squares-fit. We consider every displacement field  $\vec{u}$  or basis field  $\vec{B}_i$  as a field caused by the parameter states  $\vec{a}$ , i.e.,

**Table 1**  
Sampling scheme for parameter space. The classification of ranks is done by grouping samples based on the numbers of non-zero parameter values.

Rank	$\vec{a}$	$a_1$	$a_2$	$a_3$	$a_4$	$a_5$	$a_6$	$a_7$	...
0	$\begin{pmatrix} a_1 \\ a_2 \\ a_3 \end{pmatrix}$	$\begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}$							
1	$\begin{pmatrix} a_1 \\ a_2 \\ a_3 \end{pmatrix}$	$\begin{pmatrix} \max \\ 0 \\ 0 \end{pmatrix}$	$\begin{pmatrix} 0 \\ \max \\ 0 \end{pmatrix}$	$\begin{pmatrix} 0 \\ 0 \\ \max \end{pmatrix}$	$\begin{pmatrix} \min \\ 0 \\ 0 \end{pmatrix}$	$\begin{pmatrix} 0 \\ \min \\ 0 \end{pmatrix}$	$\begin{pmatrix} 0 \\ 0 \\ \min \end{pmatrix}$		
2	$\begin{pmatrix} a_1 \\ a_2 \\ a_3 \end{pmatrix}$	$\begin{pmatrix} \max \\ \max \\ 0 \end{pmatrix}$	$\begin{pmatrix} 0 \\ \max \\ 0 \end{pmatrix}$	$\begin{pmatrix} \max \\ \max \\ \max \end{pmatrix}$	$\begin{pmatrix} \min \\ \max \\ 0 \end{pmatrix}$	$\begin{pmatrix} 0 \\ \min \\ 0 \end{pmatrix}$	$\begin{pmatrix} \min \\ \max \\ 0 \end{pmatrix}$	$\begin{pmatrix} \max \\ 0 \\ \max \end{pmatrix}$	...
3	$\begin{pmatrix} a_1 \\ a_2 \\ a_3 \end{pmatrix}$	$\begin{pmatrix} \max \\ \max \\ \max \end{pmatrix}$	$\begin{pmatrix} \max \\ \max \\ \min \end{pmatrix}$	$\begin{pmatrix} \max \\ \min \\ \min \end{pmatrix}$	$\begin{pmatrix} \min \\ \min \\ \max \end{pmatrix}$	$\begin{pmatrix} \min \\ \max \\ \min \end{pmatrix}$	$\begin{pmatrix} \min \\ \min \\ \min \end{pmatrix}$	$\begin{pmatrix} \max \\ 0 \\ \min \end{pmatrix}$	



**Fig. 5.** Hierarchical basis generation for least-squares-fit. The algorithm approximates every simulation rank-by-rank from the generated ensemble and adds an “effect” via a displacement field when approximation error is too large.

$$\vec{u}_T(\vec{a}_T) \approx \sum c_i \vec{B}_i(\vec{a}_{B_i}) \tag{4}$$

Eq. (4) is solved with a linear least-squares algorithm that calculates the optimal pre-factors  $c_i$  for the over-determined equation system. Eq. (4) is also used for more detailed explanations.

3.2.1. Parameter estimation using PCA-generated basis fields

We explore Eq. (4) in more depth. The values for  $\vec{u}_T(\vec{a}_T)$ ,  $\vec{B}_i(\vec{a}_{B_i})$  and  $\vec{a}_{B_i}$  are known values, and  $c_i$  results from the least-squares-fit. The only unknown vector is  $\vec{a}_T$ , representing the parameter states that led to the measured displacement field  $\vec{u}_T$ . This unknown vector is calculated by defining a linear combination using the same pre-factors  $c_i$  that have been calculated via Eq. (4):

$$\vec{a}_T \approx \sum c_i \vec{a}_{B_i}. \tag{5}$$

As  $c_i$  and  $\vec{a}_{B_i}$  are known, the parameter states  $\vec{a}_T$  can be calculated. By that, the basis, generated using PCA, is used to estimate parameter states for any displacement field within the solution space of the simulation ensemble.

3.2.2. Contribution plots

The basis fields generated via hierarchical selection, see Fig. 5, are not ideal for approximating the solution space as none of the selected basis fields has the same properties as a PCA basis element. The hierarchical basis must contain more basis fields to represent the solution space at least within the specified tolerance. Although the selected displacement fields might not be an optimal basis for the approximation problem, they have a significant advantage: The chosen basis vector fields can be interpreted as contributors to a single effect (one parameter or a combination of parameters). The contribution per effect is calculated as follows: We use the hierarchical basis for a least-squares-fit with the measured displacement field as input. The obtained linear

combination can be interpreted as a combination of single effects. To plot the contribution of one single effect to the total measured displacement additional calculations are necessary.

Calculating the single contribution per effect is illustrated in Fig. 6. The measured displacement for a node  $N_i$  is the distance between the desired geometry and the actual measurement. This distance is reconstructed by a combination of vectors from the basis field using the least-squares-fit. The used basis has two basis fields per effect, see Section 3.1.2. This property makes it necessary to add the corresponding terms of the linear combination to obtain the contribution of the underlying effect. The partial displacement field  $d_i$  is calculated by adding the terms  $B_i c_i$  and  $B_i^{-1} c_i^{-1}$ . The partial displacement field is condensed to a scalar field by calculating the magnitude of the displacement vector for each node:

Partial displacement  $d_i$  of the effect  $e_i$  :

$$d_i = |c_i \vec{B}_i + c_i^{-1} \vec{B}_i^{-1}|.$$

Here,  $B_i$  and  $B_i^{-1}$  are the basis fields that relate to a single effect, and  $c_i$ ,  $c_i^{-1}$  are the pre-factors resulting from the least-squares-fit. Next, the total magnitudes of all partial displacements are computed for all nodes, by adding the partial contributions:

Total displacement  $D$  :

$$D = \sum d_i \quad (\text{sum of all partial displacements of } e_i) \tag{7}$$

Eqs. (6) and (7) are illustrated on the right-hand side of Fig. 6. The calculated total displacement length is used for contribution normalization to the interval [0, 1]:

Contribution of  $e_i$  :

$$d_i/D \quad (\text{ratio of partial displacement of } e_i \text{ and total displacement}) \tag{8}$$

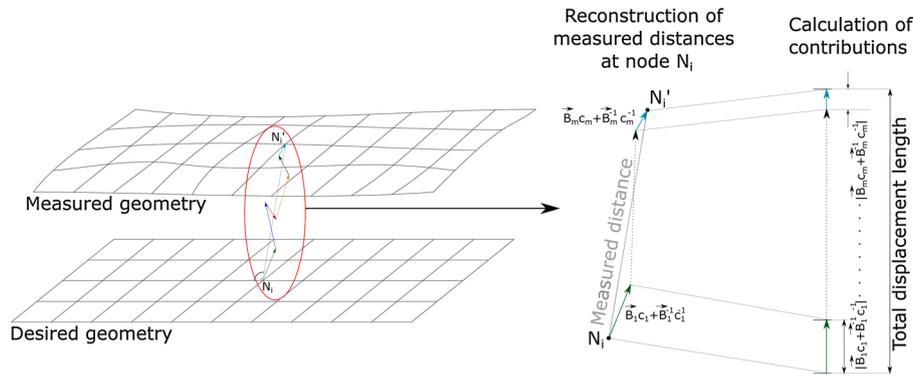


Fig. 6. Left: Desired geometry and measured geometry represented by same mesh. Right: Reconstruction of measured distance for single node  $N_i$  and contribution for every effect calculated from corresponding basis fields.

Once the contribution  $d_i/D$  has been calculated for every effect  $e_i$ , we can plot scalar fields in the 3D domain. These plots show the contributions for every effect to the measured displacement field on the scale of [0%, 100%], providing detailed insight to an inspection engineer.

#### 4. Use case – engine hood

To obtain a better understanding of the method, we describe a real-world use case. First, the used simulation model is explained, then our method is applied to it. As input displacement field we use a simulated data set with a randomly generated simulation parameter, together with a scanned measurement data set, chosen for demonstration.

##### 4.1. Simulation set-up

The chosen part is the engine hood of an automotive vehicle. This car body part is an assembly consisting of seven individual parts. The assembly structure is shown in Fig. 7. Besides the visible outer skin, two inner parts are visible when lifting the hood. Four thicker sheet metal parts are inside the hood to reinforce its structure. For this hood, an FE-simulation model based on the CAD geometry is created. Every component of the hood has meshed with 365,154 first-order 3D shell elements. All components are joined together by modeling all spot welds as well as structural and acoustic adhesives. Finally, mechanical boundaries are added, shown in Fig. 7. This figure introduces the chosen coordinate system (X-, Y-, and Z-directions). The hood is attached with two hinges to the chassis. The right (direction-of-travel) hinge is held in position while the left hinge is adjustable in X- and Y-directions. In the closed state, one gas spring per side is pushing with 580 N in X–Z-direction against the hood near the hinges. In the front, two locks and two buffers are modeled as boundaries in Z-direction. While the buffers can only push against the hood the locks can push and pull in both Z directions. All boundaries are applied to the inner parts or the

reinforced structure. Fig. 7 provides an overview of the locations of all boundaries.

As material properties, a linear elastic material model is chosen. As all parts are made of steel, an E-modulus of 210 GPa and a Poisson ratio of 0.3 are chosen. The material thickness is set to 0.7 mm for the outer skin and the inner parts. The reinforcing structure has a thickness of 2 mm.

Once the simulation model is set up, we need to define the simulation parameters that represent adjustable boundaries. These adjustable parameters are depicted in Fig. 7 as blue boundaries. For this example, not all boundaries are considered adjustable. The reason why the right hinge is fixed is to have only relative motions between the two hinges when varying the boundaries. We exclude pure translations of the hood, as they are not of interest and could easily be recognized and pre-filtered in real-world cases. All adjustable simulation parameters are displacement boundaries, listed in Table 2.

In the next step, after identifying the degrees of freedom and setting up the simulation model, the simulation ensemble is generated by sampling the parameter space, see description in Section 3.1.1. For this

Table 2  
Simulation parameters and corresponding value ranges.

Simulation parameter	Label	Description	Range of values in mm
$a_1$	HINGE X	Hinge left, X-direction	[– 1; 1]
$a_2$	HINGE Y	Hinge left, Y-direction	[– 1; 1]
$a_3$	LOCK L	Lock left, Z-direction	[– 1; 1]
$a_4$	LOCK R	Lock right, Z-direction	[– 1; 1]
$a_5$	BUFFER R	Buffer right, Z-direction	[– 1; 1]
$a_6$	BUFFER L	Buffer left, Z-direction	[– 1; 1]

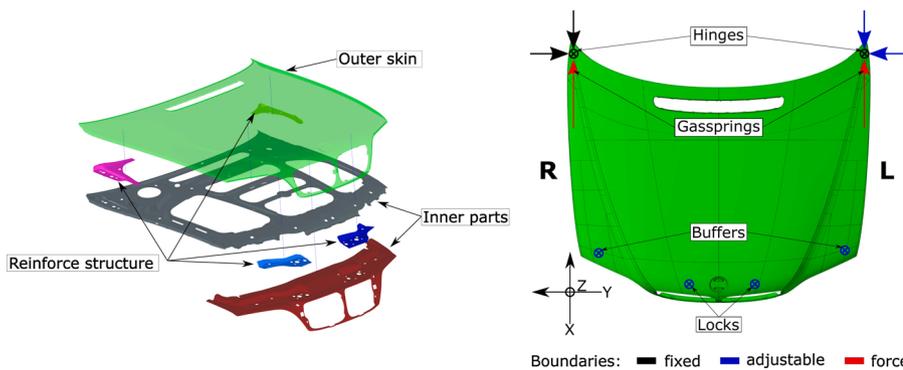


Fig. 7. Left: Assembly of the used hood. Right: Boundaries of simulation together with the coordinate system. The boundaries are fixed (cannot move), adjustable (used to manipulate the hood’s shape), or forced (pre-defined forces modeling gas springs attached to the hood). Left and Right are defined respectively to the driving direction and indicated with “L” and “R”. (This figure was published in [41] licensed under Attribution 4.0 International (CC BY 4.0), see <https://creativecommons.org/licenses/by/4.0/>, changes: Added L & R indicators, font size.)

particular example, 729 simulations were performed to sample all six parameters with the values  $-1, 0,$  and  $1$ . Additionally, 1400 simulations, evenly distributed in the parameter space, were performed for validation purposes. All the simulation outputs were post-processed by calculating the scalar product between the surface normal and displacement vectors. Thus, only the displacement perpendicular to the surface is obtained. This step is necessary as the displacement field extracted from the 3D scan does not recognize in-plane displacements. The generated ensembles are used to generate the two bases (PCA basis, hierarchical selection basis).

4.1.1. Preparing the PCA basis

The basis obtained with PCA is calculated as described in Section 3.1.2. For this PCA decomposition, only the simulation results of the visible outer skin are used. Only the portion of displacement that is perpendicular to the geometry surface is considered. The information used for subsequent calculations is restricted to the same information we can acquire by scanning the part in the assembled situation.

When performing PCA, the selection of the number of principal components (PCs) must be chosen carefully. This number affects the quality of the reconstruction achieved with the generated basis. To ensure that this number is not chosen too low, we plot the explained variance of the first ten principal components in Fig. 8. This figure shows that the whole solution space can be decomposed into seven components. Adding more components does not change the amount of information represented by the basis in a relevant way. However, to prevent errors potentially caused by choosing an insufficient number of PCs the first ten components are selected. The only disadvantage of selecting a larger number of PCs is increased computational cost of the approximation step using least-squares-fitting. The accumulated explained variance of the first ten components is 99.99985% of the total variance of the solution space. The whole solution space can be approximated with the basis found via PCA. The number of PCs highly depends on part behavior and the number of attachment points; it must be chosen application-dependent. The number of PCs could be determined automatically, e.g., by setting a minimum value for the accumulated explained variance. To ensure that the least-squares method is capable of correctly approximating parameters from arbitrary input-displacement fields, all 1400 displacement fields from the validation ensemble are approximated. The estimations for the parameter states are compared to the hidden input parameter state for each simulation. The estimation error is shown as a boxplot in Fig. 9. All 8400 values (six parameters  $\times$  1400 simulations) were estimated within the tolerance of  $\pm 0.004$  mm. One approximation with 10 basis fields requires about 5 s. None of the approximated parameter values can be obtained merely by evaluating a single node in the simulation mesh that is provided to the PCA. The simulation results provided to the PCA only contain information about the visible outer skin. However, all mechanical boundaries

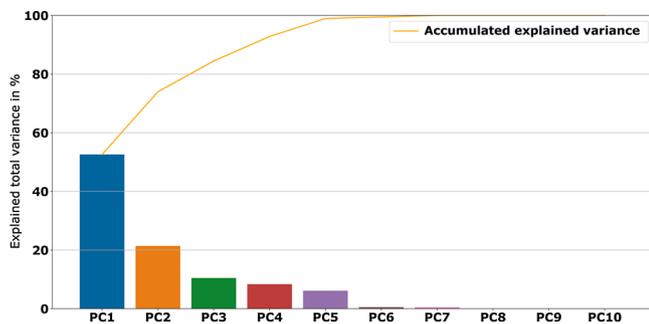


Fig. 8. Distribution of variance that are explained by each principal component. The cumulative explained variance is plotted in orange. Most of the information is contained in PC1–PC7. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

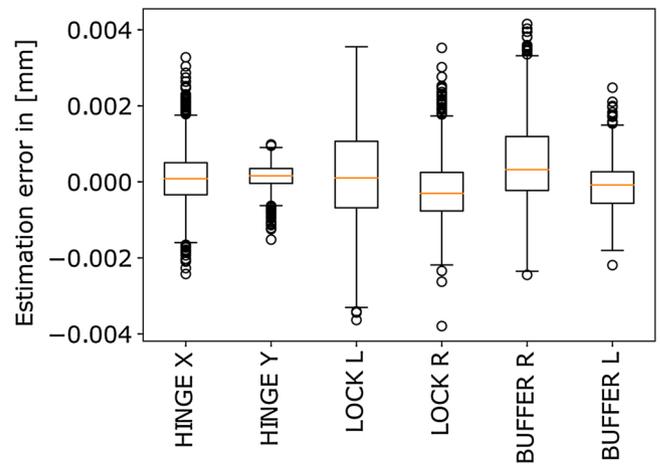


Fig. 9. Boxplot of estimation error per parameter for estimation of 1400 simulations from validation ensemble. The maximum error is below 0.004 mm for all estimates.

are interacting with the inner parts of the hood.

4.1.2. Preparing the hierarchical basis

The hierarchical basis needed for generating the contribution plots is calculated with the scheme described in Section 3.1.2. The user-defined value  $\epsilon$ , see Fig. 5, is 0.1 mm. This value is the maximally allowed error for the approximated displacement fields before the basis is extended by an additional effect. The generation of the hierarchical basis is done with the same ensemble used as input for PCA (729 simulations). The resulting basis consists of 38 effects, which one can assemble from two corresponding basis fields. To determine whether the generated basis is flexible enough to approximate the whole solution space, all 1400 displacement fields from the validation ensemble are approximated using all 38 effects. Here, the approximation output is a displacement field, and not an estimate of parameter states, see Section 3.2. As a benchmark for the quality of the approximation result, the maximum difference between each approximation and the input displacement field is used. A histogram generated from all 1400 maximum error values is shown in Fig. 10. Most approximations have a maximum error of below 0.01 mm, and single simulations can only be approximated within a tolerance of  $\pm 0.025$  mm, which is still sufficient for the presented use case. The method never uses the whole hierarchical basis, as it is coarsened during the workflow to isolate the driving effects, see Fig. 2.

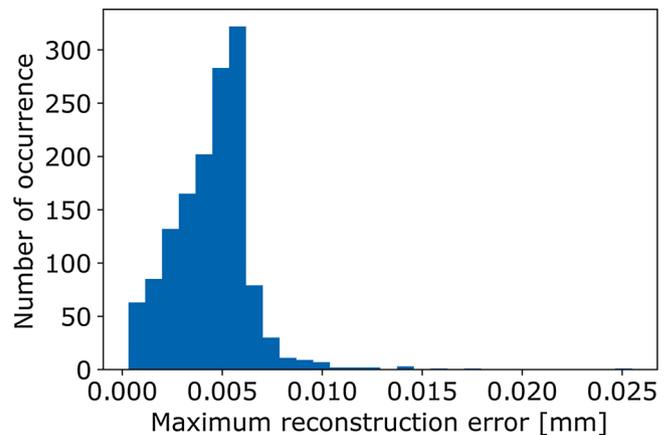


Fig. 10. Histogram for maximum reconstruction error (worst mesh node) when reconstructing 1400 simulation results from simulation ensemble using hierarchical basis for least-squares-fitting.

## 4.2. Application to simulated data

To demonstrate the output of the method a random simulation result from the validation ensemble serves as an input displacement field. The reasons for using a simulation as a pseudo-measurement are demonstration and validation. As the parameter states are well-known, we can later compare these with predictions of the method. Parameter states are considered unknown or hidden for validation. The proposed method is used to estimate the parameter states and the contribution plots for the input displacement field. To clarify this process, every step is explained with Fig. 2. First, a least-squares-fitting step (1) is performed using the basis generated by PCA. The results are estimates of parameter states (2) and a reconstruction of the displacement field. The displacement field is used to assess the quality of the reconstruction. The hierarchical basis is coarsened (3) by excluding irrelevant basis fields based on the estimated parameter state. This coarsened basis is used for an additional least-squares-fitting step (4). The resulting fit is analyzed to iteratively remove (5) further basis fields that have negligible contributions to the calculated linear combination. When the basis cannot be coarsened any longer, the remaining basis fields are used to calculate the decomposition of the input displacement field (6). This decomposition is plotted per effect (7).

### 4.2.1. Results

We discuss the results shown in Fig. 11. As mentioned in Section 3, the proposed method provides two kinds of outputs: parameter estimation and visualizations of contributions for every adjustable boundary.

The input displacement field is shown in the first picture of the top row. The corresponding parameter values are listed in the table (“Input parameter”). These parameter states are considered as hidden. The reconstruction using the PCA basis (1) is displayed next to the input displacement field. The estimated parameter (2) states are listed in the table (“Estimated Parameter”). As the simulation used for validation is contained in the validation ensemble, we know that the error of the parameter estimate is below  $\pm 0.004$  mm, see Fig. 9. The difference between the input and reconstructed displacement fields is used to validate the desired quality of the reconstruction. A direct comparison between these two fields is plotted in the right picture of the first row on a scale of 0–0.25 mm. This approximation does not show any recognizable differences. To compare different approximation results, the RMS<sup>1</sup> error is calculated for all node displacement differences. The reconstruction performed with the coarsened hierarchical basis (4) is shown next to the reconstruction based on the PCA basis. Again, the reconstruction cannot visibly be distinguished from the input field. Considering the RMS error, the reconstruction is slightly worse than the reconstruction based on the PCA basis. The reconstruction using the hierarchical basis, shown in this figure, only uses five effects (HINGE X, HINGE Y, LOCK L, LOCK R, LOCK L & LOCK R). All other basis fields are excluded already by the coarsening loop (5). The effect “LOCK L & LOCK R” is a combination of the parameter LOCK L and LOCK R that cannot be explained by a combination of the single effects. The contributions per effect are plotted in the lower half of Fig. 11 on the scale of 0–100%.

The output of our method is understandable and simple to interpret. The user does not have to identify critical parameters as the method can provide information about the parameter contributions for all measured points. Our proposed method could be used, for example, as a kernel for a process control loop to achieve a self-compensating assembly process using optical measurement for assessing process quality.

## 4.3. Application to measurement data

We have performed a validation of the proposed method by

performing an experiment with a real car body part. The car body part used for the validation with experimental data is the one presented in Section 4 – an engine hood from a BMW E46.

### 4.3.1. Hardware Set-up

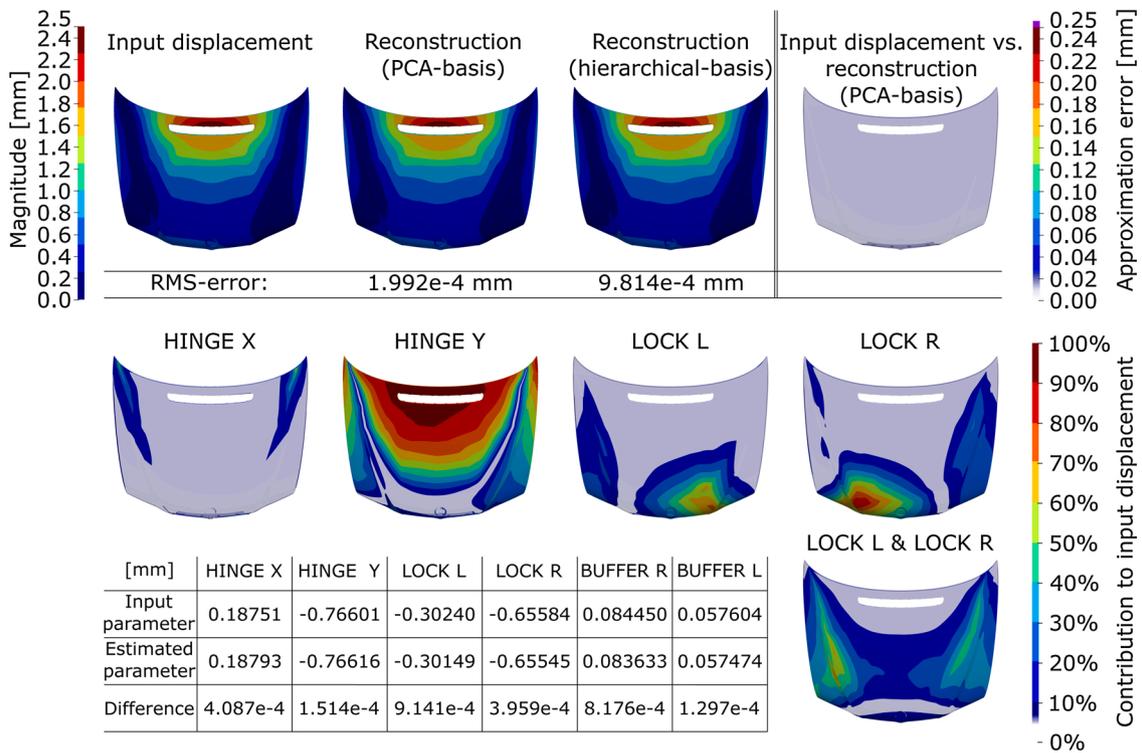
The hood is prepared for measurement by applying a light gray paint-job to it. This is necessary to scan the part with a structured light scanner. Besides the part itself, a fixture is designed and built up from aluminum profiles. Locks, buffers, hinges, and gas springs are mounted onto this frame. The whole set-up is depicted in Fig. 12. To make the relative position of the hinges adjustable, the left hinge is mounted onto a compound slide. The locks and buffers have proprietary adjustment options. The frame is equipped with markers to align different measurements. A dial gauge is used to change boundaries precisely. The hood is not mounted in the orientation as assembled in a car, but this only affects the direction of gravity used in the simulation. For validation, the orientation of the hood does not matter; orientation matters when considering the scanning set-up, see the right side of Fig. 12. The cameras and projector of the structured light scan system must have a certain distance to the scanned object to capture the entire object. Usually, complex parts are captured in many pictures that are stitched together. However, to reduce the uncertainties caused by data registration, we condense every measured state of the hood into a single picture. This approach leads to a much coarser resolution and noisier measurement. Nevertheless, the quality of the results is still sufficient for validation. A significant advantage of this setup is the fact that the hood and the hardware components of the scanner are not moved between scans. All measurements are made in the same coordinate system. This eliminates the need for alignment. The markers that are mounted to the fixture are primarily used to determine whether something has moved between scans.

### 4.3.2. Verifying the scan setup

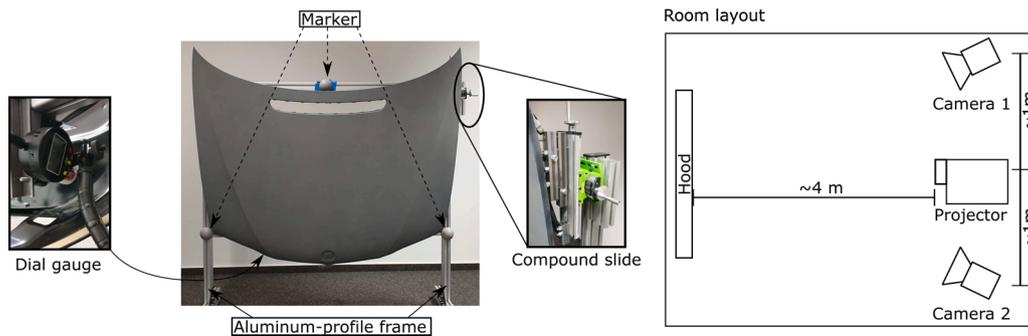
The used measurement system is a modified HP pro S3 structured light scanner, calibrated with custom calibration panels for the large size of the scan window. To verify the precision and reproducibility of the scanning system, we must perform experiments with different scan setups. First, measurement noise is investigated by performing scans with different settings. In particular, exposure time of the cameras is varied in the range  $\frac{1}{75}$  to  $\frac{1}{20}$  s. Fig. 13 shows a comparison between the generated point cloud and a smoothed surface for a smaller region of the geometry. The table in the figure lists statistical properties for each region. The average deviation shows that the measurement results are approximated well by a smooth representation of the geometry. As local profile defects are not relevant for validation, a smooth surface representation is desired. This geometry is obtained by smoothing the acquired point cloud. All scans generated with different exposure times produce almost the same shape after smoothing. For validation  $\frac{1}{15}$  s is used, although the measurement noise is higher than that obtained with  $\frac{1}{75}$  or  $\frac{1}{10}$  s. The reason for choosing this value is: By scanning with this exposure time one obtains the best overall percentage of points classified as “high-quality points” by the scanning software (HP 3D Scan 5). A single scan produces about 1 million high-quality points. One measurement step samples the geometry of the shown part approximately five times more densely compared to the simulation mesh (outer shell visible by scanner: 188,559 mesh nodes).

A second test was performed to assess the repeatability of the whole setup. We perform an initial scan and a complete full experiment, where the positions of the Locks and Buffer are changed. Afterward, all boundaries are returned to their initial positions and a second scan is performed. A comparison between the initial and second scan is shown in Fig. 14. The signed distance between the scans is plotted on a scale of  $\pm 0.1$  mm. Besides measurement noise, no deviations of relevant magnitudes can be identified.

<sup>1</sup> Root-mean-square.



**Fig. 11.** Validation results with artificial input data. Top row: Input displacement field and reconstructions with PCA basis and hierarchical basis. Right: Reconstruction error when using PCA basis. Middle row: Contribution plots generated for the reconstruction obtained with the hierarchical basis. Table: Parameter values estimates and estimation errors obtained from the reconstruction with the PCA basis.



**Fig. 12.** Left: Experimental setup used for the validations with measurement Right: High-level depiction of set-up for scanning the hood. (This figure was published in [41] licensed under Attribution 4.0 International (CC BY 4.0), see <https://creativecommons.org/licenses/by/4.0/>, changes: spelling.)

4.3.3. Experimental validation

To validate the proposed method based on experimental data, an experiment similar to the validation with simulation data as presented in Section 4.2 is performed. To generate the necessary input displacement field, two scans are compared. The first scan is acquired for the initial situation. This situation is considered as desired geometry or “target geometry” for validation. In a second step, different mechanical boundaries are adjusted with the dial gauge to change the state of boundaries. It turns out that the fixture cannot be assumed to be entirely rigid. Unfortunately, this leads to minor movement of some boundaries while adjusting others. Setting all boundaries to a certain state is not straightforward with one dial gauge. To solve this problem, multiple gauges are used to capture and compensate for unwanted run-away. When boundary states have been changed precisely, a second scan of the new configuration is acquired. The second scan is considered an undesirable situation with misplaced boundaries. The two scans are aligned with the simulation mesh (CAD geometry) and the distance between each scan and the simulation mesh is calculated. The

displacement magnitude (distance between two scans) is obtained for the simulation mesh. The direction of the displacement is assumed to be perpendicular to the surface of the CAD geometry. A displacement vector per node is calculated by multiplying the surface normal with the calculated displacement magnitude. The resulting vector field is the desired input for validation.

4.3.4. Results

The results of the validation with experimental data are summarized in Fig. 15. The input displacement field is obtained as described before. The upper picture on the left side of the figure shows the input displacement field. The changed boundaries – changed relative to the initial measured state – are the left lock in negative Z-direction, the right lock in positive Z-direction, and the left buffer in positive Z-direction, all of the dimension 1 mm. The values are summarized in the table contained in the figure. In this configuration BUFFER R is not actively controlled as the hood lifts off the buffer loses contact. The position of the contact point for BUFFER R results from the change of part shape.

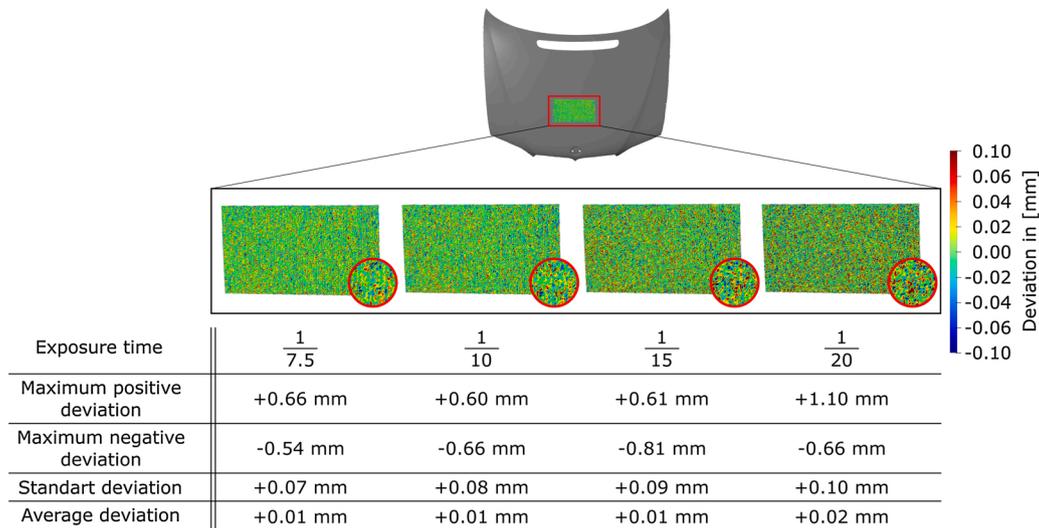


Fig. 13. Investigation of measurement noise when using different exposure times (scan settings). Top: Point clouds compared to surface generated by smoothing of measured point clouds. Bottom: statistical properties for each scan enabling comparison of results.

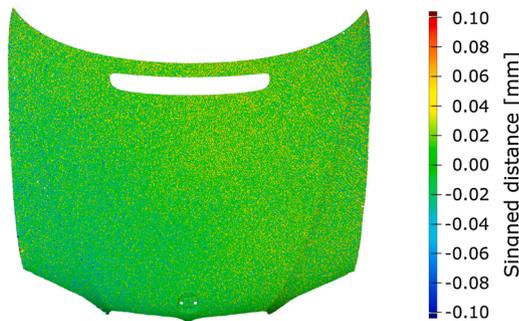


Fig. 14. Investigation of the reproducibility scan setup for two situations. First: Initial measurement. Second: Changing boundary position and measuring after returning all boundaries to initial position.

However, this position is estimated by the method. The picture next to the input field shows the reconstruction performed with the same PCA basis also used for the validation, see Section 4.2. Under the picture, the RMS-error value is provided. The difference between input field and reconstruction using the PCA basis is plotted in the top-right of the figure, on a scale of 0–0.25 mm. The error of reconstruction is mostly below 0.1 mm, and some local areas show error values of up to 0.25 mm. The estimates of the parameter states obtained from the reconstruction with the PCA basis are shown in the table shown in the figure. Compared with the set parameter states, the maximum estimation error is 0.14 mm for the left buffer, and all other parameter values are estimated within the tolerance of  $\pm 0.06$  mm.

Next to the picture showing the reconstruction with the PCA basis, the reconstruction with the hierarchical basis is depicted. The quality of this reconstruction is also assessed with the RMS error. As expected, the error value is slightly higher when compared to the optimal PCA basis approach. The reconstruction with the hierarchical basis is again used for creating the contribution plots shown in the lower half of Fig. 15 on a scale of 0–100%. It is striking that the hierarchical basis captures the effect of moving the hinge in Y-direction although it was not moved during the experiment – but the estimator barely recognizes a movement of this boundary.

#### 4.4. Computation times

Table 3 summarizes computational run times. All times were

obtained with the same system configuration:

- Processor: Intel i7-5820k@3.6GHz
- Memory: 64Gb DRR4@2100MHz
- Storage: M.2 SSD read; write 3200 MB/s; 1400 MB/s
- GPU: not used for calculations/simulation
- OS: Microsoft Windows 10 Pro
- FE-solver: ABAQUS – Dassault Systèmes

The computation of the simulation ensemble takes up most of the overall time. However, this preparation step could be performed in a parallel fashion; thus, this step scales with the amount of available resources. All simulation results are converted to binary vtk file format – [42]. To perform PCA, our system uses the python implementation of the “scikit-learn” library – [43]. All displacement fields are pre-loaded to the ram as a list of “NumPy” matrices – [44]. The generation of the bases does take about 41 s for performing PCA for the ensemble of displacement fields and calculating the resulting basis sets with corresponding parameter states. The generation of the hierarchical basis is more costly, requiring about 3:21 h of computing time. This is explained by the iterative algorithm, depicted in Fig. 5. However, as this step must be performed only once, this computing time does not impact the applicability of the method. The critical run times are the execution times. The estimation of the parameter states, performing a least-squares-fitting step with the PCA-basis, takes about 1.84 s. To compute a least-squares-fit, we use the python library “SciPy” – [45]. Coarsening the hierarchical basis iteratively and calculating the contribution plots takes about 19.77 s. Besides the used libraries, all custom implementations of the presented algorithms were done in python.

## 5. Discussion

The discussion of this paper addresses various aspects: First, the method is discussed in detail, followed by the achieved results obtained with artificially generated simulation data. Next, the experimental setup and the achieved results of the pre-studies are addressed. The validation results performed with real measurement data obtained from the experiment are discussed. Last, the run times are assessed in terms of usability for a real-world setting, followed by a comparison with state-of-the-art methods.

The described method has two outputs. The first output is an estimate of the parameter states based on an acquired input displacement

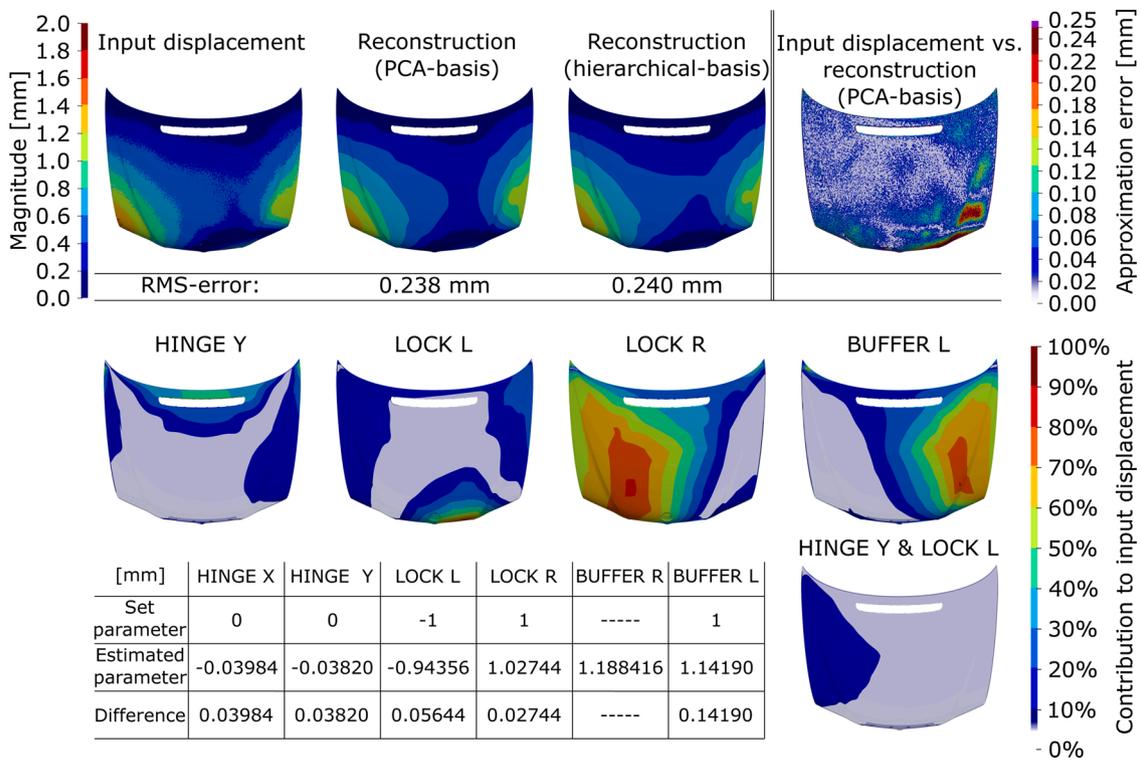


Fig. 15. Validation results with experimental measurement input data. Top row: Input displacement field and reconstructions with PCA basis and hierarchical basis. Right: Reconstruction error when using PCA basis. Middle row: Contribution plots generated from reconstruction with hierarchical basis. Table: Parameter value estimates and estimation errors obtained from reconstruction with PCA basis.

Table 3 Computational run times. “Preparation times” and “Execution times” are listed.

Task	Time per simulation	# of simulations	Total time
<i>Preparation times</i>			
Generating simulation ensemble	187 s	729	37.87 h
Generating validation ensemble	187 s	1400	72.72 h
PCA-basis generation	–	–	41 s
Hierarchical-basis generation	–	–	3:21 h
<i>Execution times</i>			
Performing parameter estimation	–	–	1.84 s
Coarsening hierarchical basis and generate contribution plots	–	–	19.77 s

field. This output is only an interim result that arises during calculations. The methods employed for this calculation step (PCA and least-squares-fitting), are commonly used for solving this type of problem, and they are not a contribution of this paper. However, the validations with experimental data showing the usability for parameter estimation in sheet metal assembly processes is considered as a contribution.

The second output of the method are the contribution plots, decomposing the input displacement field into single plots per effect. The presented approach is unique and the main contribution of this paper. The contribution plots provided by the method, see Fig. 15, support a quality inspection engineer in a “human-in-the-loop decision-making process”. The plots provide a visually intuitive way to understand the superposition of boundary effects. However, one disadvantage of this visualization is that these plots do not have an underlying physical basis that could be used for determining correctness.

The validation with simulation shows the potential of the proposed method under ideal conditions, without any disturbance caused by real-world circumstances. The results shown in Fig. 11 demonstrate two facts. First, the parameter estimation of the six boundary states is working extremely well, with estimation errors below 0.004 mm. This

value does not only hold for the shown input displacement field but also for the whole validation ensemble of 1400 simulations evenly distributed in parameter space, see Fig. 9. None of the parameter states can be obtained by evaluating single nodes of the input field, as the attack points of the boundaries are all attached to the underlying structure of the part. Second, the reconstruction of the input field using the coarsened hierarchical basis performs almost as well as the reconstruction method using the PCA basis. The similarity between the two different reconstruction bases and the input displacement field is the foundation of the validation. The contribution plots themselves cannot be validated by performing measurements on the input data or set-up. However, the quality of reconstruction and the effects can be evaluated. The presented example documents satisfying reconstruction results, and all identified effects and plotted areas of impact are plausible. Thus, our method works as intended and provides correct information to the user.

The validation performed with experimental data of a measured real physical part shows that the proposed method has great potential for real-world applications. This step is very important as most research discussed in the literature, e.g., [22], does not use real measurement data but adds artificial noise to simulation data instead. The problems that arise when applying a method to real-world set-ups are not addressed. In our work, two different pre-studies are performed. The first study addresses the measurement noise of the scanning system. Fig. 13 summarizes the effect of different exposure times on measurement noise. One of our important findings for evaluating raw measurement data is the fact that measurement noise does not significantly affect the resulting shape when smoothing point clouds. As we are not interested in surface defects, like buckles or dents, our scan system set-up is sufficient for validation. The second pre-study addresses the repeatability of the experimental set-up, see results in Fig. 14. Our comparison shows differences below 0.1 mm between an initial scan and a second scan after changing boundary conditions and returning them to the initial position. With proper care, repeatability of ±0.1 mm was achieved.

The actual results of the validation with experimental data are summarized in Fig. 15. The input displacement field contains noise from the scans, being the foundation for the displacement field. The estimated parameter shown in the table has a higher estimation error when compared with the previous validation. However, all parameter values can still be estimated within the tolerance of  $\pm 0.06$  mm, except BUFFER L having an estimation error of 0.14 mm. To explain this value we must explore the approximation quality of the reconstruction method with the PCA Basis. A direct comparison between this reconstruction and the input displacement field is also shown in Fig. 15. This comparison shows error values up to 0.25 mm in some areas that appear to be mostly near the location of BUFFER L. We explain this deviation between measurement and approximation by a behavior of the real part that cannot be expressed by the simulation model. When performing an in-depth parameter study directed at fine-tuning the simulation model to replicated the measured state we are unable to explain or “correct” the observed behavior. We assume that the used physical part does not exactly behave as expected. This leads to approximation errors when performing the least-squares-fitting step, and to errors when estimating parameter values. However, our validation results demonstrate the robustness of the method. When working with noisy measurement data and slightly different part behavior, one could still evaluate the parameter states at the proper scale. Furthermore, indicators for problematic areas that cannot be explained by the solution space are obtained (differences between measured field and reconstruction).

We now focus on the contribution plots. We evaluate the quality and value of these plots by comparing the reconstruction performed with the PCA basis to the reconstruction based on the hierarchical basis. Slight differences can be spotted, but the underlying behavior is nearly identical. Our discrete color map is highly sensible, e.g., slight differences affect the path of iso-lines. Unfortunately, when using a continuous rainbow color map, for example, completely hides these differences. The contribution plots themselves show the correct contributing effects to create the input displacement field. The effect HINGE Y was not changed during the experiment, but it is used to explain the measured displacement. This might not seem intuitive, but this is plausible when examining the reason for this behavior. The effect HINGE Y that corresponds to the relative distance between the two hinges in Y-direction is highly sensitive for two reasons: (i) changing this boundary state of magnitude 1 mm does cause displacements of up to 3.1 mm affecting large areas of the part. Even small movements of this boundary cause recognizable effects for the contribution plots. (ii) Small movements of the hinges in Y-direction cannot be excluded due to the physical setup that is not entirely rigid at this scale.

The computational run times are listed in Table 3. As expected, most time is needed for generating the simulation ensemble. Nevertheless, these times are obtained via sequential execution of individual simulation runs, and run times would scale with available computing resources. The generation of the different bases requires a total of about 3.5 h. All pre-computation times are not excessive, as these steps are performed only once. More relevant for the application in real-world assembly processes are the execution times for the steps necessary for every individual measurement. The parameter estimation performed with the PCA basis requires less than 2 s. The result of the parameter estimation could already be used as a kernel for a self-compensating assembly system, see Fig. 1. This approximation uses all 188,559 displacement vectors to perform the least-squares-fitting step. This step could be accelerated significantly by coarsening the input displacement field. The second step, performing the approximation again with the hierarchical basis, requires about 19.77 s, explained by the iterative coarsening of the full basis. The ability to generate contribution plots is an extremely helpful functionality of a human-in-the-loop system, but it is not a needed component of an in-line system, where real-time performance is expected. Our computation times are within an acceptable range for the applications we consider.

## 6. Conclusions

We have introduced a method to gain insight into a measured high-resolution displacement field describing the deviation of a desired, ideal CAD geometry part from its physically manufactured counterpart. As an application, we have considered sheet metal assembly and used our method to visualize contributions of individual part boundaries to a measured displacement field. Our method can greatly assist a quality inspection engineer by decomposing the measured displacement field into its constituent components associated with specific boundaries. The engineer can more easily understand the cause(s) of observed deviations and perform appropriate countermeasures for correction. Our implementation uses commercially available FE-simulation and visualization software. We have described the use of our method in a real-world scenario and performed validations for simulated and experimental data. We have examined the usability of our approach for parameter value estimation in depth. Our results document that it is possible to reconstruct an input displacement field with a difference less than 0.25 mm, compared to the input when using experimental data. Our validations demonstrate that it is viable to use our method as a stand-alone solution or as a component of a decision-making system in a self-compensating assembly setting.

To improve the presented method one could investigate the mapping between measured displacement fields and CAD geometry in more depth, where pre-simulated basis fields would be defined. In our current approach, we use simple distance calculation available in the software CloudCompare. If one were to use the proposed method for self-compensating assembly processes, the generation of the input displacement field, see Fig. 1, would have to be accelerated significantly. The FE-simulation that predicts an ideal assembly based on a pre-assembly 3D-scan, is too slow for in-line usage. However, the prediction of the boundary states, achieved by the proposed method in approximately 1.85 s, already suffices for most applications and could be accelerated more by reducing the number of points used for prediction.

### Authors' contribution

Felix Claus: methodology, implementation, validations, writing.  
Bernd Hamann: writing.

### Declaration of Competing Interest

The authors report no declarations of interest.

### Acknowledgements

This research was funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) – 252408385 – IRTG 2057. We thank Q-DAS GmbH for providing CAD models.

**Conflict of interest:** None declared.

### References

- [1] Yu M, Zhang Y, Li Y, Zhang D. Adaptive sampling method for inspection planning on cmm for free-form surfaces. *Int J Adv Manuf Technol* 2013;67:1967–75. <https://doi.org/10.1007/s00170-012-4623-0>.
- [2] Shi Quan, Xi Ning, Sheng Weihua, Chen Yifan. Development of dynamic inspection methods for dimensional measurement of automotive body parts. *Proceedings 2006 IEEE international conference on robotics and automation ICRA 2006* 2006: 315–20. <https://doi.org/10.1109/ROBOT.2006.1641730>.
- [3] Lee KH, Pung Park H. Automated inspection planning of free-form shape parts by laser scanning. *Robot Comput-Integr Manuf* 2000;16:201–10. [https://doi.org/10.1016/S0736-5845\(99\)00060-5](https://doi.org/10.1016/S0736-5845(99)00060-5).
- [4] Kiraci E, Franciosa P, Turley GA, Olifent A, Attridge A, Williams MA. Moving towards in-line metrology: evaluation of a laser radar system for in-line dimensional inspection for automotive assembly systems. *Int J Adv Manuf Technol* 2017;91:69–78. <https://doi.org/10.1007/s00170-016-9696-8>.
- [5] Bergström P, Fergusson M, Folkesson P, Runnemalm A, Ottosson M, Andersson A, et al. Automatic in-line inspection of shape based on photogrammetry. In: *The 7th*

- international Swedish production symposium, SPS16; 2016. p. 1–9. <http://www.sps16.se>.
- [6] Denkena B, Berg F, Acker W. Surface inspection system for large sheet metal parts. *Sheet metal* 2005;55:59–64. <https://doi.org/10.4028/www.scientific.net/AMR.6-8.559>.
- [7] Babu M, Franciosa P, Ceglarek D. Spatio-temporal adaptive sampling for effective coverage measurement planning during quality inspection of free form surfaces using robotic 3d optical scanner. *J Manuf Syst* 2019;53:93–108. <https://doi.org/10.1016/j.jmsy.2019.08.003>.
- [8] Carlson JS, Söderberg R. Assembly root cause analysis: a way to reduce dimensional variation in assembled products. *Int J Flex Manuf Syst* 2003;15:113–50. <https://doi.org/10.1023/A:1024453207632>.
- [9] Baier L, Frommherz J, Nöth E, Donhauser T, Schuderer P, Franke J. Identifying failure root causes by visualizing parameter interdependencies with spectrograms. *J Manuf Syst* 2019;53:11–7. <https://doi.org/10.1016/j.jmsy.2019.08.002>.
- [10] Lu C, Zhao HW. Fixture layout optimization for deformable sheet metal workpiece. *Int J Adv Manuf Technol* 2015;78:85–98. <https://doi.org/10.1007/s00170-014-6647-0>.
- [11] Camelio JA, Hu SJ, Ceglarek D. Impact of fixture design on sheet metal assembly variation. *J Manuf Syst* 2004;23:182–93. [https://doi.org/10.1016/S0278-6125\(05\)00006-3](https://doi.org/10.1016/S0278-6125(05)00006-3).
- [12] Liu SC, Hu SJ, Woo TC. Tolerance analysis for sheet metal assemblies. *J Mech Des* 1996;118:62–7. <https://doi.org/10.1115/1.2826857>.
- [13] Camelio J, Hu SJ, Ceglarek D. Modeling variation propagation of multi-station assembly systems with compliant parts. *J Mech Des* 2004;125:673–81. <https://doi.org/10.1115/1.1631574>.
- [14] Liao X, Wang GG. Simultaneous optimization of fixture and joint positions for non-rigid sheet metal assembly. *Int J Adv Manuf Technol* 2008;36:386–94. <https://doi.org/10.1007/s00170-006-0827-5>.
- [15] Zerwer A, Cascante G, Hutchinson J. Parameter estimation in finite element simulations of Rayleigh waves. *J Geotechn Geoenviron Eng* 2002;128:250–61. [https://doi.org/10.1061/\(ASCE\)1090-0241\(2002\)128:3\(250\)](https://doi.org/10.1061/(ASCE)1090-0241(2002)128:3(250)).
- [16] Bell ES, Sanayei M, Javdekar CN, Slavsky E. Multiresponse parameter estimation for finite-element model updating using nondestructive test data. *J Struct Eng* 2007;133:1067–79. [https://doi.org/10.1061/\(ASCE\)0733-9445\(2007\)133:8\(1067\)](https://doi.org/10.1061/(ASCE)0733-9445(2007)133:8(1067)).
- [17] Teixeira Parente M, Mattis S, Gupta S, Deusner C, Wohlmuth B. Efficient parameter estimation for a methane hydrate model with active subspaces. *Comput Geosci* 2019;23:355–72. <https://doi.org/10.1007/s10596-018-9769-x>.
- [18] Gendy AS, Saleeb AF. Nonlinear material parameter estimation for characterizing hyper elastic large strain models. *Comput Mech* 2000;25:66–77. <https://doi.org/10.1007/s004660050016>.
- [19] Yoshida F, Urabe M, Toropov V. Identification of material parameters in constitutive model for sheet metals from cyclic bending tests. *Int J Mech Sci* 1998;40:237–49. [https://doi.org/10.1016/S0020-7403\(97\)00052-0](https://doi.org/10.1016/S0020-7403(97)00052-0).
- [20] Reina G, Paiano M, Blanco-Claraco JL. Vehicle parameter estimation using a model-based estimator. *Mech Syst Signal Process* 2017;87:227–41. <https://doi.org/10.1016/j.ymsp.2016.06.038>. *Signal Processing and Control Challenges for Smart Vehicles*.
- [21] Johnson ML, Faunt LM. parameter estimation by least-squares methods. *Numerical computer methods*. Vol. 210 of *Methods in enzymology* 1992:1–37. [https://doi.org/10.1016/0076-6879\(92\)10003-V](https://doi.org/10.1016/0076-6879(92)10003-V).
- [22] Sinha S, Glorieux E, Franciosa P, Ceglarek D. 3d convolutional neural networks to estimate assembly process parameters using 3d point-clouds. In: Stella E, editor. *Multimodal sensing: technologies and applications, international society for optics and photonics*. SPIE; 2019. p. 89–101. <https://doi.org/10.1117/12.2526062>.
- [23] Leonhardt V, Claus F, Garth C. Pen: process estimator neural network for root cause analysis using graph convolution. *J Manuf Syst* 2021 (in press).
- [24] Lu K, Jin Y, Chen Y, Yang Y, Hou L, Zhang Z, et al. Review for order reduction based on proper orthogonal decomposition and outlooks of applications in mechanical systems. *Mech Syst Signal Process* 2019;123:264–97. <https://doi.org/10.1016/j.ymsp.2019.01.018>.
- [25] Bui-Thanh T, Damodaran M, Willcox K. Aerodynamic data reconstruction and inverse design using proper orthogonal decomposition. *AIAA J* 2004;42:1505–16. <https://doi.org/10.2514/1.2159>.
- [26] Lindau B, Lindkvist L, Andersson A, Söderberg R. Statistical shape modeling in virtual assembly using pca-technique. *J Manuf Syst* 2013;32:456–63. <https://doi.org/10.1016/j.jmsy.2013.02.002>. *Assembly Technologies and Systems*.
- [27] Homri L, Goka E, Levasseur G, Dantan JY. Tolerance analysis – form defects modeling and simulation by modal decomposition and optimization. *Comput-Aided Des*. 2017;91:46–59. <https://doi.org/10.1016/j.cad.2017.04.007>.
- [28] Yoo H, Cho J, Chung J. Modal analysis and shape optimization of rotating cantilever beams. *J Sound Vibr* 2006;290:223–41. <https://doi.org/10.1016/j.jsv.2005.03.014>.
- [29] Das A, Franciosa P, Pesce A, Gerbino S. Parametric effect analysis of free-form shape error during sheet metal forming. *Int J Eng Sci Technol* 2017;9:117–24. <http://wrap.warwick.ac.uk/93407/>.
- [30] Goka E, Homri L, Beaurepaire P, Dantan JY. Geometrical variation simulation for assembly with form defects. 2017. <https://doi.org/10.1115/IMECE2017-70118>.
- [31] Adragna PA, Samper S, Formosa F, Pillet M. Modal tolerancing – application to gap and flush analyses. Dordrecht: Springer Netherlands; 2007. p. 417–30. [https://doi.org/10.1007/978-1-4020-6761-7\\_28](https://doi.org/10.1007/978-1-4020-6761-7_28).
- [32] Söderberg R, Wickman C, Lindkvist L. Improving decision making by simulating and visualizing geometrical variation in non-rigid assemblies. *CIRP Ann* 2008;57:175–8. <https://doi.org/10.1016/j.cirp.2008.03.040>.
- [33] Gentilini I, Shimada K. Predicting and evaluating the post-assembly shape of thin-walled components via 3d laser digitization and fea simulation of the assembly process. *Comput-Aided Des* 2011;43:316–28. <https://doi.org/10.1016/j.cad.2010.11.004>.
- [34] Söderberg R, Wärmefjord K, Carlson JS, Lindkvist L. Toward a digital twin for real-time geometry assurance in individualized production. *CIRP Ann* 2017;66:137–40. <https://doi.org/10.1016/j.cirp.2017.04.038>.
- [35] Claus F, Rupprecht FA, Hagen H. Online simulation considering production uncertainties to improve assembly quality. *NAFEMS*; 2019. [https://www.nafems.org/publications/resource\\_center/nwc\\_19\\_356/](https://www.nafems.org/publications/resource_center/nwc_19_356/).
- [36] Yi Y, Yan Y, Liu X, Ni Z, Feng J, Liu J. Digital twin-based smart assembly process design and application framework for complex products and its case study. *J Manuf Syst* 2021;58:94–107. <https://doi.org/10.1016/j.jmsy.2020.04.013>. *Digital Twin towards Smart Manufacturing and Industry 4.0*.
- [37] Claus F, Hagen H, Hamann B. Calculating the gravity-free shape of sheet metal parts. *Int J Adv Manuf Technol* 2021;113:3401–17. <https://doi.org/10.1007/s00170-021-06702-6>.
- [38] Cloudcompare, 2020. <http://www.cloudcompare.org/>.
- [39] Aspert N, Santa-Cruz D, Ebrahimi T. Mesh: measuring errors between surfaces using the hausdorff distance. *Proceedings. IEEE international conference on multimedia and expo*, vol. 1 2002:705–8. <https://doi.org/10.1109/ICME.2002.1035879>.
- [40] Kenta-Tanaka, probreg. <https://probreg.readthedocs.io/en/latest/>.
- [41] Claus F, Hagen H, Leonhardt V, Leitte H, Hamann B, et al. Interactive quality inspection of measured deviations in sheet metal assemblies. In: Garth C, Aurich JC, Linke B, Müller R, Ravani B, Weber GH, editors. *2nd international conference of the DFG international research training group 2057 – physical modeling for virtual manufacturing (IPMVM 2020)*; 2021. p. 1–6. <https://doi.org/10.4230/OASiCS.iPMVM.2020.6>.
- [42] Schroeder W, Martin K, Lorensen B. *The visualization toolkit: an object-oriented approach to 3D graphics; [visualize data in 3D – medical, engineering or scientific; build your own applications with C++, Tcl, Java or Python; includes source code for VTK (supports Unix, Windows and Mac)]*. 4th ed. Clifton Park, NY: Kitware Inc; 2006.
- [43] Pedregosa F, Varoquaux G, Gramfort A, Michel V, Thirion B, Grisel O, et al. *Scikit-learn: machine learning in Python*. *J Mach Learn Res* 2011;12:2825–30.
- [44] Harris CR, Millman KJ, van der Walt SJ, Gommers R, Virtanen P, Cournapeau D, et al. *Array programming with NumPy*. *Nature* 2020;585:357–62. <https://doi.org/10.1038/s41586-020-2649-2>.
- [45] Virtanen P, Gommers R, Oliphant TE, Haberland M, Reddy T, Cournapeau D, et al. *SciPy 1.0: fundamental algorithms for scientific computing in Python*. *Nat Methods* 2020;17:261–72. <https://doi.org/10.1038/s41592-019-0686-2>.