CHALMERS
UNIVERSITY OF TECHNOLOGY

Online geometry assurance in individualized production by feedback control and model calibration of digital twins

Journal of Manufacturing Systems 66 (2023) 71–81

Contents lists available at ScienceDirect

Journal of Manufacturing Systems

journal homepage: www.elsevier.com/locate/jmansys

Technical paper

Online geometry assurance in individualized production by feedback control and model calibration of digital twins

Anders Sjöberg [a,c], Magnus Önnheim [a,c], Otto Frost [a,c], Constantin Cronrath [b], Emil Gustavsson [a,c], Bengt Lennartson [b], Mats Jirstrand [a,b,c,*]

[a] Fraunhofer-Chalmers Centre, Gothenburg, SE-412 88, Sweden
[b] Department of Electrical Engineering, Chalmers University of Technology, Gothenburg, SE-412 96, Sweden
[c] Fraunhofer Center for Machine Learning, Sweden

ARTICLE INFO

Keywords:
Digital Twin
Calibration
Unscented Kalman filter
Smart assembly 4.0

ABSTRACT

In this paper, we consider online calibration of a Digital Twin and its use for control and optimization in the assembly process of sheet metal parts. This calibration is done based on a feedback signal received by calculating the quality of the simulated assemblies as compared to the prediction made by the Digital Twin.

We develop a Kalman filter-based approach for online calibration of the Digital Twin, which in turn is used by a one-step look-ahead optimizer to define an online control scheme. This control scheme balances directly predicted quality gains against reduced uncertainty whose purpose is to enable long-term quality gains.

The usage of a calibrated model in a one-step look-ahead optimizer as a controller allows to incorporate the benefits of the usage of Digital Twins for *individualized* control, where the control parameters of a production cell are optimized in a Digital Twin based on measured properties of the production inputs, over *nominal* control, where control parameters are set with respect to some reference production inputs, in an approach which is able to use measured final production quality for feedback control.

The proposed approach is evaluated by computer simulations of two industrial product assembly use cases. In the first case, it demonstrates significant gains in quality of the produced assemblies, while in the second case it shows negligible to small improvements. The second case is, however, rather insensitive to miscalibration, which enables only small gains.

1. Introduction

The concept of a Digital Twin in manufacturing [1,2] has emerged in the light of the advances of the Internet of Things, big data, and cloud computing [3]. A Digital Twin is essentially a real-time digital counterpart of a physical object or system [4], that breaks the barriers between the physical world and the virtual world [5,6]. The Digital Twin concept is a core enabling technology for realizing smart and autonomous manufacturing systems [7] to improve productivity and optimality [8,9].

However, an end-to-end Digital Twin of a manufacturing system necessarily consists of many interacting parts. Thus, *calibration* of the Digital Twin to reality becomes increasingly important, as miscalibration of interacting components may nullify any gains. Indeed, convergence and co-evolution of digital models and physical systems are key characteristics of Digital Twins [10–12]. These key characteristics are provided by state estimation of the system and parameter identification of the Digital Twin model [13]. Yu et al. for instance, use Gaussian Particle Filtering to estimate system states and parameters [14]. Lugaresi and Matta address this research challenge by generating and updating digital production system models from manufacturing event logs [15]. Wang et al. combine data-driven and physics-based methods to predict tool wear [16], while, Liu et al. present a mechanism to evaluate uncertainty in Digital Twin-based decision making [17].

In this paper, we perform a simulation-based study on *Smart Assembly 4.0* [18]—a Digital Twin framework for a sheet metal welding production cell in an assembly line. It includes geometric path planning of industrial robots, measurement of individual part variation before welding, physical models for the deformation of assemblies during welding, modelling of locator adjustments, and modelling of feedback control and data driven process improvement [19,20]. Detailed modelling of the end-to-end assembly process has shown promising results, see, e.g., [21–25].

We especially focus on the clamp and welding simulation in Smart-Assembly 4.0, where the predicted quality of a particular weld is

* Corresponding author at: Fraunhofer-Chalmers Centre, Gothenburg, SE-412 88, Sweden.
E-mail address: jirstran@chalmers.se (M. Jirstrand).

https://doi.org/10.1016/j.jmsy.2022.11.011
Received 6 May 2022; Received in revised form 17 October 2022; Accepted 16 November 2022
Available online 10 December 2022

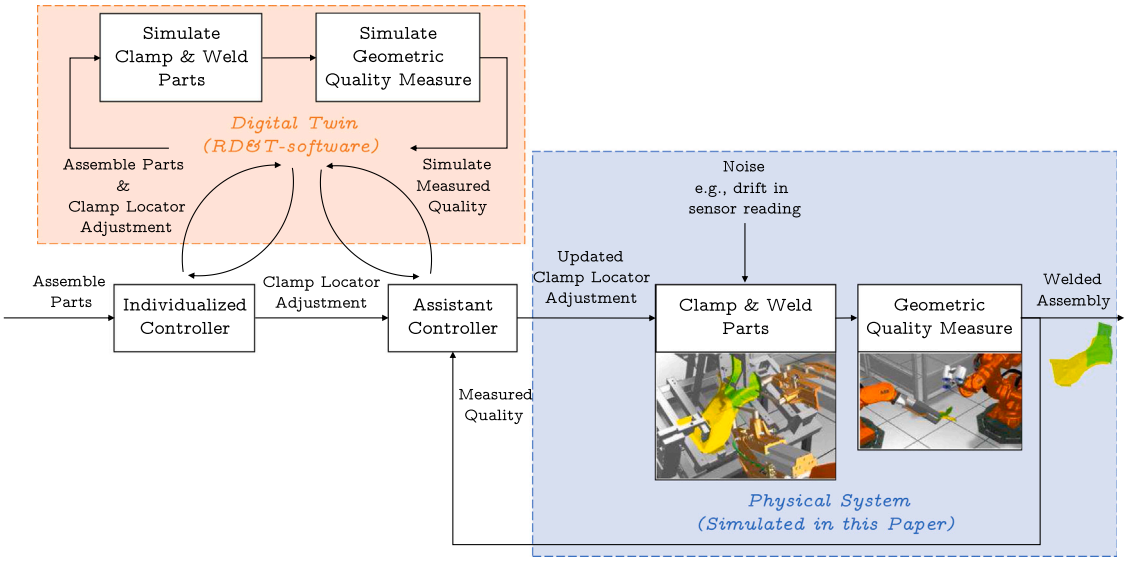


Fig. 1. An overview of the whole chain of events. Given a set of assembly parts, which are about to be welded, the individualized controller computes the optimal clamp locator adjustment with help of the Digital Twin, i.e., the RD&T software, and proposes that as a control signal. Due to noise, e.g., drift in sensor reading, in the welding cell the clamp locator might need to be adjusted according to that offset, and that is the purpose of our proposed (assistant) controller. After welding the assembly is scanned to measure the geometric quality and that is in turn fed back to the controller. This controller incorporates both the Digital Twin and the feedback signals of welded assemblies' quality. In this simulation based study the physical system is simulated in accordance with the Digital Twin.

computed by the Robust Design & Tolerance (RD&T) software package [26]. It applies statistical variation simulation for tolerance analysis and uses Finite Element Methods (FEM) to predict deformations of a considered assembly. Amongst many other parameters, the weld quality is dependent on the placement of clamp locators. These clamp locator adjustments are seen as a control signal and the software RD&T is considered as a black box function, from which a quality measure is obtained given a set of assembly parts to join and the clamp locator adjustments. We make no additional assumptions on this function. RD&T constitutes the Digital Twin in this paper and in the sequel these terms are used interchangeably.

We further assume access to an individualized controller, which is able to optimize the black box function to find the optimal clamp locator adjustment for each individual assembly and for any fixed configuration of the parameters of RD&T. This individualized controller assumes, however, that there is no noise or other deviations in the clamp locator adjustment measurements. Such noise could, for instance, correspond to a drift in the sensor readings or a perturbation of one of the assembly machines. In practice, the presence of such noise may affect quality greatly.

For that reason, we examine in this paper whether it is possible to identify and track such deviations between the Digital Twin model and a physical system, where the only discrepancy between them both is that particular deviation. That is, we assume that the physical system can be fully represented by our Digital Twin using a certain parameter configuration. We examine further how to use the tracking information in order to assist the individualized controller in, over time, finding the highest quality welds for sequence of assemblies, see Fig. 1 for an illustrative overview. To acquire an evaluation environment with ground truth, i.e., known discrepancy between the physical system and the Digital Twin, we also simulate the physical system equivalently as our Digital Twin.

We provide a problem formulation for online calibration and control of a Digital Twin that takes into account black-box characteristics, individual part geometries, and noisy system measurements. A general objective of this formulation and Smart Assembly 4.0 is to enable the Digital Twin concept for practical applications. To the best of the authors' knowledge, we developed a novel Kalman filter-based approach that balances a trade-off between *exploration* (tracking ability) and *exploitation* (single-assembly quality improvements) in order to tackle the calibration and control problem over a sequence of assemblies. In our simulated experiments, we evaluate the exploration–exploitation trade-off and show that significant quality improvements are possible with our proposed approach.

The rest of the paper is organized as follows: in Section 2, we develop the problem formulation and the proposed Kalman filter-based solution approach. The results of the simulated experiments on an industrial relevant assembly task are presented in Section 3. Sections 4 and 5 presents a discussion of the results and the conclusion for this paper.

2. Methods

This section is organized as follows: in Section 2.1 we formalize the problem, providing the necessary assumptions and terminology required to establish a rigorous framework. In Section 2.2 we present a dynamic stochastic model that will be used to solve the problem. In Section 2.3 we present the proposed control scheme for the online calibration. Finally, in Section 2.4, we describe the experimental setup for the simulation study.

2.1. Formalizing the problem

One main underlying assumption is that the Digital Twin, in our case the RD&T software, is able to provide highly accurate predictions of the final quality, but that due to process errors it may be inaccurately parameterized, in particular in terms of clamp locator offsets. Let us define the quality function family of all possible instantiations (given a parameterization) of a Digital Twin assembly process model as the set of functions

$$\mathcal{Q} := \left\{ Q(\boldsymbol{x}, \boldsymbol{u}; \boldsymbol{\theta}) : \mathbb{R}^m \times \mathbb{R}^d \to \mathbb{R} : \boldsymbol{\theta} \in \mathbb{R}^n \right\}, \tag{1}$$

denoting all the possible models, where $\boldsymbol{x} \in \mathbb{R}^m$ denotes the geometrical shape of the constituent parts of an assembly, $\boldsymbol{u} \in \mathbb{R}^d$ is the clamp locator adjustment for the corresponding assembly, and $\boldsymbol{\theta} \in \mathbb{R}^n$ is the parameterization of the model. During an episode consisting of T assemblies, assembly parts $\boldsymbol{x}_t$, $t = 1, \ldots, T$ are revealed and a corresponding control signal $\boldsymbol{u}_t$ is generated by some control scheme, yielding a final assembly with a measurement of geometric quality z_t.

The quality is defined by the deviation of some representative points of the overall geometry, and is then summarized by the root mean square (RMS) measure [27]. Our main assumption is, that for any assembly $\boldsymbol{x}_t$ and any control signal $\boldsymbol{u}_t$, the real (unknown) quality function Q_t is a member of $\mathcal{Q}$, i.e., at time t there exists some parameterization $\boldsymbol{\theta}_t$ such that the real observations z_t can be accurately represented.

From this assumption we formulate our two research questions:

1. Given a realization $(\boldsymbol{x}_t, \boldsymbol{u}_t, z_t)$, is it possible to find an adequate estimate $\hat{\boldsymbol{\theta}}_t$ of $\boldsymbol{\theta}_t$?
2. Can we design (and validate) a control scheme for $\boldsymbol{u}_t$ that takes into account the uncertainty of the estimate $\hat{\boldsymbol{\theta}}_t$ in order to improve quality?

For later reference, we here give two examples of control schemes not taking the uncertainty of $\hat{\boldsymbol{\theta}}_t$ into account.

Definition 1 (*Nominal Control*)**.** The control signal for every assembly t is given by solving

$$\boldsymbol{u}_t := \boldsymbol{u}^0 \in \operatorname{argmin}_{\boldsymbol{u}\in\mathbb{R}^d} Q(\boldsymbol{x}^0, \boldsymbol{u}; \boldsymbol{\theta}^0), \tag{2}$$

where $\boldsymbol{x}^0$ is some nominal or reference assembly and $\boldsymbol{\theta}^0$ is some nominal parameter.

Definition 2 (*Individualized Control*)**.** The control signal for every assembly t is given by solving

$$\boldsymbol{u}_t := \boldsymbol{u}_t^0(\boldsymbol{x}_t) \in \operatorname{argmin}_{\boldsymbol{u}\in\mathbb{R}^d} Q(\boldsymbol{x}_t, \boldsymbol{u}; \boldsymbol{\theta}^0), \tag{3}$$

where $\boldsymbol{\theta}^0$ is some nominal parameter.

The benefit of individualized control as compared to nominal control was explored in the related paper [21], using the same case data. An individualized controller will both serve the purpose of assisting our approach and be a baseline to which we compare our approach.

For concreteness, we will in the remainder of this paper focus on a specific, simplified, structure of the parameters $\boldsymbol{\theta}$, manifesting itself as translation offsets in the clamp locator adjustments $\boldsymbol{u}$.

Assumption 1. There exists a nominal parameter $\boldsymbol{\theta}^0$, such that for every parameter $\boldsymbol{\theta}$, there exists a $\boldsymbol{\delta u}^*$ such that for every assembly $\boldsymbol{x}$ and every clamp locator adjustment $\boldsymbol{u} \in \mathbb{R}^d$

$$Q(\boldsymbol{x}, \boldsymbol{u}; \boldsymbol{\theta}) := Q(\boldsymbol{x}, \boldsymbol{u} - \boldsymbol{\delta u}^*; \boldsymbol{\theta}^0). \tag{4}$$

Assumption 1 models the important special case where miscalibration is due to defects in the locator fixture, due to, e.g., inexact mounting of a fixture in a production cell, or drift caused by wear and tear of the actuators inside the fixture, which affects the desired control signal $\boldsymbol{u}$. That is, we effectively identify the set of possible $\boldsymbol{\theta}$ with the clamp locator offset $\boldsymbol{\delta u}^*$. Note that this assumption may not be entirely realistic, as it cannot model, e.g., miscalibrations of the physical characteristics of materials of the constituent parts. It is however important to point out that assuming a specific form of the effect of the miscalibration, as in (4), has the significant advantage that its implementation is completely separated from the implementation of Q, which enables our approach to work with a black-box model for Q. We refer the reader to related work with the same case material [22] for an approach without Assumption 1 utilizing a contextual bandits approach.

2.2. Stochastic model

To develop our Kalman filter based approach, we use the following uninformative stochastic dynamic model for the process of welding and measurement in the production cell:

$$\boldsymbol{u}_t = F\left(\boldsymbol{x}_t; \mathcal{D}_{t-1}\right), \tag{5a}$$

$$Z_t \sim \mathcal{N}(Q(\boldsymbol{x}_t, \boldsymbol{u}_t - \boldsymbol{\delta u}_t^*; \boldsymbol{\theta}^0), \sigma_z^2), \tag{5b}$$

$$\boldsymbol{\delta u}_{t+1}^* \sim \mathcal{N}(\boldsymbol{\delta u}_t^*, \Sigma_u). \tag{5c}$$

Here, $\mathcal{D}_{t-1} := \{(\boldsymbol{x}_s, \boldsymbol{u}_s, z_s),\ s = 1, \dots, t-1\}$ represents the data collected so far and $\mathcal{N}$ denotes a normal distribution. (5a) denotes a control scheme mapping collected data to some control signal by a function F, e.g., from Definition 1, 2, or by the control scheme to follow in the sequel. (5b) models the measured final assembly quality as a noisy measurement with measurement error variance σ_z^2. (5c) models the dynamics of $\boldsymbol{\delta u}^*$ as a discrete Brownian motion and if Σ_u is small, $\boldsymbol{\delta u}^*$ can be thought of as a slowly changing constant. That is, the dynamic of $\boldsymbol{\delta u}^*$ is deliberately modelled in an uninformative way—no specific or strong assumptions are made about the dynamics of parameter drift in the production cell.

Our overall aim is to minimize the expected average of quality defects over an episode of assemblies, i.e., find the control scheme F^*, where

$$F^* \in \operatorname{argmin}_F \mathbb{E}\left[\frac{1}{T}\sum_{t=1}^{T} Z_t\right] = \operatorname{argmin}_F \frac{1}{T}\sum_{t=1}^{T} Q(\boldsymbol{x}_t, \boldsymbol{u}_t - \boldsymbol{\delta u}_t^*; \boldsymbol{\theta}^0) \tag{6}$$

and T is the number of assemblies.

Further note, with $\boldsymbol{u}_t^0$ as in (3), we have that

$$\boldsymbol{u}_t^0 + \boldsymbol{\delta u}_t^* \in \operatorname{argmin}_{u_t} Q(\boldsymbol{x}_t, \boldsymbol{u}_t - \boldsymbol{\delta u}_t^*; \boldsymbol{\theta}^0) \tag{7}$$

and if we could achieve $F(\boldsymbol{x}_t; \mathcal{D}_{t-1}) = \boldsymbol{u}_t^0 + \boldsymbol{\delta u}_t^*$, then F is a solution to (6). However, this is typically not possible, since $\boldsymbol{\delta u}_t^*$ is not directly measurable. We may however define a surrogate to approximate the ideal F^*, by the calibration problem for $\boldsymbol{\delta u}_t^*$, i.e., the estimation of $\boldsymbol{\delta u}_t^*$ from observed values z_t by using (5b)–(5c). In effect, the above discussion shows that a better calibration of $\boldsymbol{\delta u}_t^*$ should lead to a better controller F.

2.3. Online calibration

As noted earlier, if the ideal F^* is approximated by estimating $\boldsymbol{\delta u}_t^*$, a control signal could be chosen to directly compensate for the estimated offset. The performance of such an approach clearly depends on the accuracy of that estimation. However, the accuracy of the estimation itself depends on the control signals. It may therefore be worthwhile to excite the system, by injecting noise signals into (5a), in order to get a more well-conditioned calibration problem. Thus, there is an exploitation–exploration trade-off, the balance between choosing a control signal that is expected to result in high quality according to the current estimation or an exploratory control signal that excites the system in order to ease the calibration, in the hope of improved quality over an entire episode.

In this paper we consider an *online* calibration approach, where the deviation error $\boldsymbol{\delta u}_t^*$, for every assembly t, is estimated and compensated for. A block-diagram of the online controller is provided in Fig. 2. The controller consists of a state estimator and a state-feedback controller, which we describe separately in the following subsections.

2.3.1. State estimator

The stochastic dynamic model (5b)–(5c) can be seen as a state–space model with external inputs, with $\boldsymbol{\delta u}_t^*$ as the state variable, $\boldsymbol{x}_t$ and $\boldsymbol{u}_t$ as the external input and Z_t as a measurement, given by a non-linear measurement function. It then follows naturally to incorporate an unscented Kalman filter (UKF) [28] as the estimation algorithm for the state $\boldsymbol{\delta u}_t^*$ of the stochastic dynamic model (5b)–(5c). The usage of a Kalman-like filter allows us to maintain a probability distribution of the true state of (5b)–(5c) given past observed measurements, giving not only an estimate of the true state but also a quantitative measure on the uncertainty of the current estimate. This measure will be used to define a term promoting exploration of the controller in the sequel (12). Note that due to the black-box nature of Q, the classical Kalman filter [29] cannot be used. However, the unscented Kalman filter is able to provide Gaussian approximations of the probability distributions, even in the

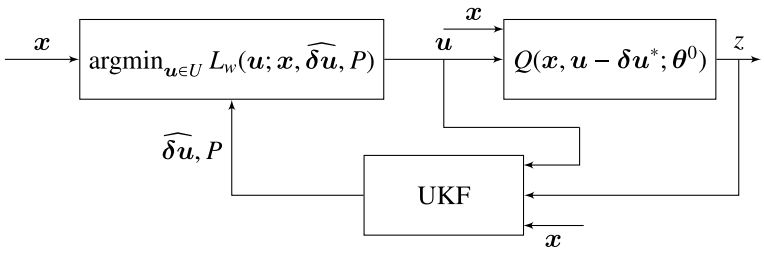


Fig. 2. A block-diagram of the proposed online control scheme. The UKF-block denotes the Unscented Kalman Filter as a state estimator, providing a state estimate $\widehat{\delta u}$ along with an uncertainty measure in the form the posterior covariance P of the UKF. The controller has access to the current assembly, x, along with the state estimate $\widehat{\delta u}$ and uncertainty measure P, and is implemented as a one-step lookahead minimizer of a loss function L_w, defined in (13), that balances the exploitation–exploration trade-off mentioned in Section 2.3. The rightmost box corresponds to the physical system in Fig. 1, while the other two boxes in this figure are together another representation of the controllers and the Digital Twin in Fig. 1.

presence on non-linearity. Throughout the remainder of this paper, we denote the estimate of the state by $\widehat{\delta u}_t$, and the corresponding posterior state covariance as P_t. That is, the UKF provides a probabilistic estimate of δu_t^* given by

$$\delta u_t^* \sim \mathcal{N}(\widehat{\delta u}_t, P_t). \tag{8}$$

We may then consider the posterior covariance P_t as a quantitative measure of calibration quality, in the sense that large covariances indicates uncertainty in the estimation of δu_t^*.

2.3.2. Control signal

By definition, the optimal control signal is $u_t = u_t^0 + \delta u_t^*$. However, since δu_t^* is unknown a sensible replacement is our currently best estimate of this quantity given information available up to assembly t, i.e., $\widehat{\delta u}_t$, resulting in the control signal

$$u_t = u_t^0 + \widehat{\delta u}_t := \hat{u}_t^*. \tag{9}$$

Note that as described in beginning of Section 2.3, this control signal may lead to a ill-conditioned calibration problem, which manifests in the posterior state covariance P_t in (8) growing large, possibly unnecessarily worsening the future losses of (6), written as

$$J_t = \frac{1}{T-t}\sum_{s=t+1}^{T} Q(x_s, u_s - \delta u_s^*; \theta^0). \tag{10}$$

An intuitive approach to minimizing (10) is to exploit the current information $(\widehat{\delta u}_t, P_t)$ to try and directly optimize $Q(x_{t+1}, u_{t+1} - \widehat{\delta u}_{t+1}; \theta^0)$. However, this choice of control signal may affect future values of P_t adversely, which may cause degradation of quality in the long run. Thus, we would ideally like to also keep P_t small.

We therefore consider a one-parameter family of controllers, implemented as controllers minimizing a one step look-ahead loss, as a weighted combination of the expected next-step quality along with the next-step uncertainty P_t of the UKF-estimator, denoted as the *exploitation loss* and *exploration loss*, respectively.

We define the exploitation loss $L_t^{\text{exploit}}(u) := L^{\text{exploit}}(u; x_t, \widehat{\delta u}_t, P_t)$ as

$$L_t^{\text{exploit}}(u_t) := \sum_{s_t \in S_t} \mathcal{W}(s_t) Q(x_t, u_t - s_t; \theta^0), \tag{11}$$

where S_t is the set of sigma points [28] in the unscented Kalman filter, $Q(x_t, u_t - s_t; \theta^0)$ is the estimated quality at $u_t - s_t$, and $\mathcal{W}(s_t)$ is the corresponding weight. That is, we define the exploitation loss as the estimated expected next-assembly quality as calculated by the UKF.

The exploration loss is designed to promote the calibration of the deviation error, as measured by the (future) state covariance P_{t+1}. We define the exploration loss as

$$L_t^{\text{explore}}(u_t) := \frac{1}{d}\sqrt{\text{Tr}(P_{t+1}(u_t))}, \tag{12}$$

where P_{t+1} is the state covariance matrix from the unscented Kalman filter at time $t+1$, Tr is the trace operator, and d is the dimension of u_t. Note, however, that there are many other reasonable definitions of loss functions to be used for minimizing the covariance matrix P_{t+1}.

A one parameter family of loss functions can then defined by a convex combination of the exploitation and exploration objectives, L_t^{exploit} and L_t^{explore},

$$L_w(u_t) := (1-w)\cdot L_t^{\text{exploit}}(u_t) + w \cdot L_t^{\text{explore}}(u_t), \tag{13}$$

where $w \in [0, 1]$. That is, $w = 0$ and $w = 1$ corresponds to full exploitation and full exploration, respectively. The exploration–exploitation trade-off discussed in the introduction of this section is then handled by finding a weight w that best balances the next-step quality and the ability to calibrate well, as measured by (6). The empirical experiments in this paper explore this trade-off.

Solving the problem

$$\text{argmin}_{u_t \in U_t} L_w(u_t), \tag{14}$$

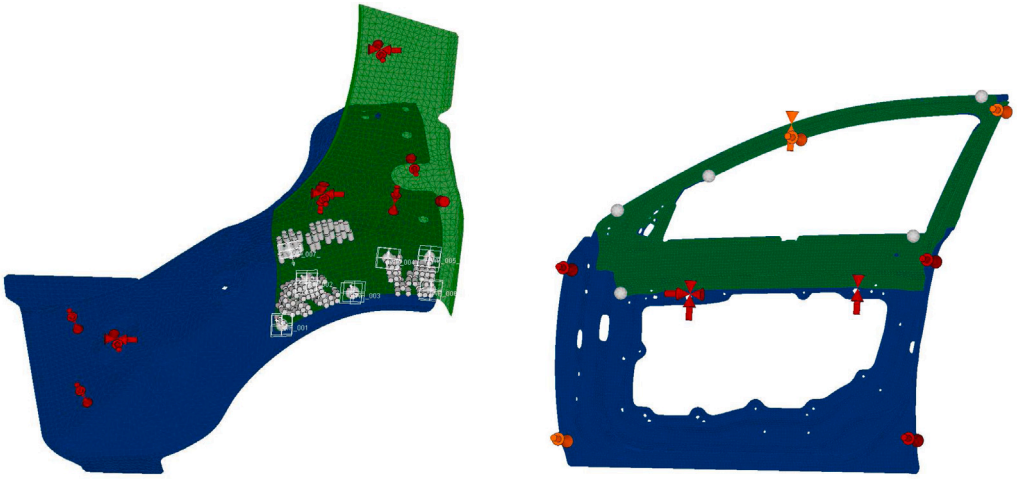
where U_t is a chosen set, for a given w comes with a significant computational burden, since even a single evaluation of (14) requires many calls to the function Q, which in itself is time-consuming to evaluate due to expensive FEM computations. We propose to approximate (14) by simply evaluating candidate values of u_t generated from a low-discrepancy sequence, e.g., a Sobol-sequence [30], and transform it into a sequence from a Gaussian distribution with mean $\hat{u}_t^*$ in (9) and with covariance matrix given by the posterior state distribution P_t from the unscented Kalman filter. There is a sensible dynamical nature of this approach, that the degree of exploration in the minimization is decided by the certainty of the estimation. The computational budget then decides the length of the proposed sequence. Points that are outside the chosen set U_t are disregarded.

2.4. Experimental setup

In order to evaluate the proposed framework, it is applied to simulations of two industrial sheet metal assembly cases. In both cases two sheet metal parts are clamped into position for a subsequent spot-welding operation. In the first case, the two sheet metal parts constitute a car body, and in the second case, the two parts form a car door, see Fig. 3. In the sequel we refer to these two cases as the *car body*-case and the *car door*-case. Further, for both cases there are 10 individually scanned sheet metal parts of the upper and the lower parts. Thus, there are 100 possible assembly combinations. An experimental run is defined as an episode consisting of the 100 unique assemblies that are randomly permutated.

All the units in the sequel of this paper are in millimetres, except for the variances that are in square millimetres.

The observed quality measurement z_t is generated by $Q(x_t, u_t - \delta u_t^*; \theta^0)$ in a noise-free environment, except for the part variation that is already induced by the individually scanned parts. The offset δu_t^* is updated as a periodic piece-wise linear function—a continuous drift



(a) (b)

Fig. 3. Illustrations of the two industrial sheet metal assembly cases (a) and (b) where there are two sheet metal parts, blue and green, clamped into position for a subsequent spot welding operation. There are 12 and 19 locators, respectively, of the fixture, red and orange, that can be adjusted along their axis to improve geometry of the welded assembly. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

that remains in reasonable magnitude; it is an example of slowly varying disturbances. We define the update scheme as

$$\boldsymbol{\delta u}_t^* := \boldsymbol{p}_{t-1} + \boldsymbol{\delta u}_{t-1}^*, \tag{15}$$

where initial offset $\boldsymbol{\delta u}_0^*$ is set to 0 and $\boldsymbol{p}_0 \in \mathbb{R}^d$ is a randomly chosen initial direction for the drift, that is normalized to a length of 0.03. The update scheme of $\boldsymbol{p}_0$ is in turn

$$\boldsymbol{p}_t := \begin{cases} \boldsymbol{p}_{t-1}, & \text{if } |(\boldsymbol{p}_{t-1} + \boldsymbol{\delta u}_{t-1}^*)_i| \le \boldsymbol{b}_i,\ i = 1, \dots, d, \\ -\boldsymbol{p}_{t-1}, & \text{otherwise,} \end{cases}$$

where $\boldsymbol{b} \in \mathbb{R}^d_{\ge 0}$ is a chosen boundary equal to 0.5.

We let the variance σ_z^2 in (5b) be small, 0.0001 and 0.000001 for respectively case, and Σ_u in (5c) be equal to 0.001. As initialization we take $\widehat{\boldsymbol{\delta u}}_0 \sim \mathcal{N}(0, 0.001)$.

We consider the following control schemes:

- An individualized controller that always apply the control signal $\boldsymbol{u}_t^0$ in (3) and is referred to as the *baseline controller*.
- A controller that applies the control signal $\hat{\boldsymbol{u}}_t^*$ in (9), provided partly by a UKF. This controller is referred to as the *default controller*.
- 11 different controllers that apply the control signal given by (14) for 11 evenly distributed values of w. This family of controllers is referred to as the *w-controllers*. The number of points in the Sobol sequence, in order to approximate (14), are chosen to be 2^{2+d}, where d is the dimension of the control signal $\boldsymbol{u}$. Let the chosen set U_t in (14) be

$$\{\boldsymbol{u} \in \mathbb{R}^d : -0.1 \le (\boldsymbol{u})_i - (\hat{\boldsymbol{u}}_t^*)_i \le 0.1,\ i = 1, \dots, d\}.$$

The individualized control signal $\boldsymbol{u}_t^0$ in (3) is approximated by a global optimizer that have run an extensive number of iterations to ensure near optimal values for each assembly considered. Simultaneously the optimal quality z_t^* for the corresponding assembly is approximated.

The implementation is made in Python. The unscented Kalman filter is provided by the software library filterpy [31]. Furthermore, in order to generate the low-discrepancy sequence that attempts to solve (14) we used the QMCPy package [32]. All the computer simulated experiments were made on a desktop computer with a 4-core 4 GHz i7-6700K processor.

3. Results

We consider two evaluation metrics in order to analyse the results—one describing the quality defects and the other is a performance measure of the offset estimation. The first evaluation metric is the average relative error residual with respect to the approximated optimal quality over an episode, i.e.,

$$\frac{1}{T}\sum_{t=1}^{T} \frac{z_t - z_t^*}{z_t^*},$$

where z_t is the measured quality and z_t^* is the approximated optimal quality. The second evaluation metric is the average Euclidean distance between $\widehat{\boldsymbol{\delta u}}$ and $\boldsymbol{\delta u}^*$ over an episode, i.e.,

$$\frac{1}{T}\sum_{t=1}^{T} \left|\widehat{\boldsymbol{\delta u}}_t - \boldsymbol{\delta u}_t^*\right|.$$

We consider that a controller diverges if the distance between the estimate $\widehat{\boldsymbol{\delta u}}_t$ and the real offset $\boldsymbol{\delta u}_t^*$ exceeds 10.0 during a run, because after exceeding that magnitude controllers start to propose unreasonable control signals and do not regain a good state estimation.

3.1. Test case A—Car body

Computer simulated experiments on test case A (see Fig. 3(a)) comprise of three experiments in which only one of the three most significant locators is drifting and one experiment in which all three are drifting simultaneously.

3.1.1. One drifting locator

Figs. 4–6 are box plots that exhibit the results of the car body case for all the considered controllers over 10 different runs, where only one clamp locator is drifting and consequently the only clamp locator considered by the controllers. A run is a realization of an episode with deterministic randomness—same randomness (e.g., sequence order of assemblies and offset drifting direction) for all the controllers. For each of the three figures there is a different clamp locator drifting, we call the locators A, B, and C, and those are the ones that affect the quality the most. The upper part of each figure shows the average relative error residual with respect to the approximated optimal quality, while the lower part of each figure shows the average Euclidean distance between $\widehat{\boldsymbol{\delta u}}$ and $\boldsymbol{\delta u}^*$ over an episode. The number of times a controller diverges among the 10 runs can be seen in the figures as well, visualized as red circles where the degree of redness reflects the number of divergences.

The average distance to $\boldsymbol{\delta u}^*$ is considerably smaller for higher values on w in all three cases, with the smallest value at $w = 1.0$, i.e., full exploration. The smallest average relative error residual quality is around $w = 0.7$.

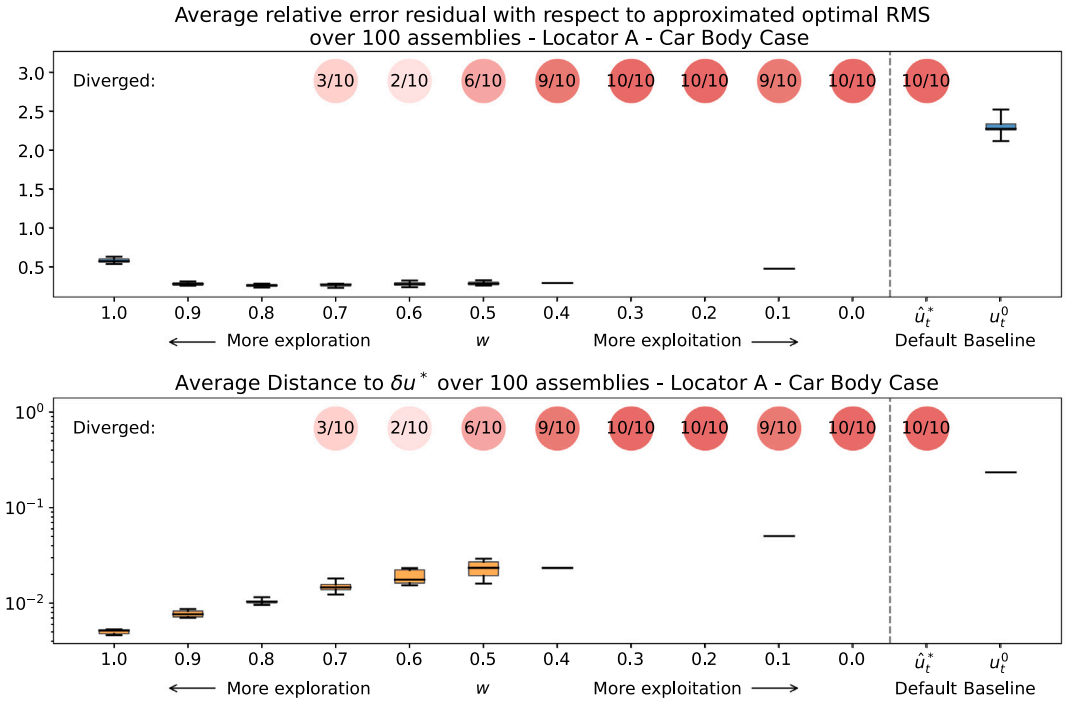


Fig. 4. (Car body case with locator A drifting.) Box plots (upper and lower) showing the spread of the average relative residual error quality of an episode and the average Euclidean distance between the estimation and offset error of an episode for all considered controllers over 10 different runs for the car body case. A run is a realization of an episode with deterministic randomness—same randomness (e.g., sequence order of assemblies and offset drifting direction) for all the controllers. Each run consists of 100 assemblies where clamp locator A is drifting. A controller diverged during a run if the state estimation $\widehat{\delta u}$ is far from the offset δu^*.

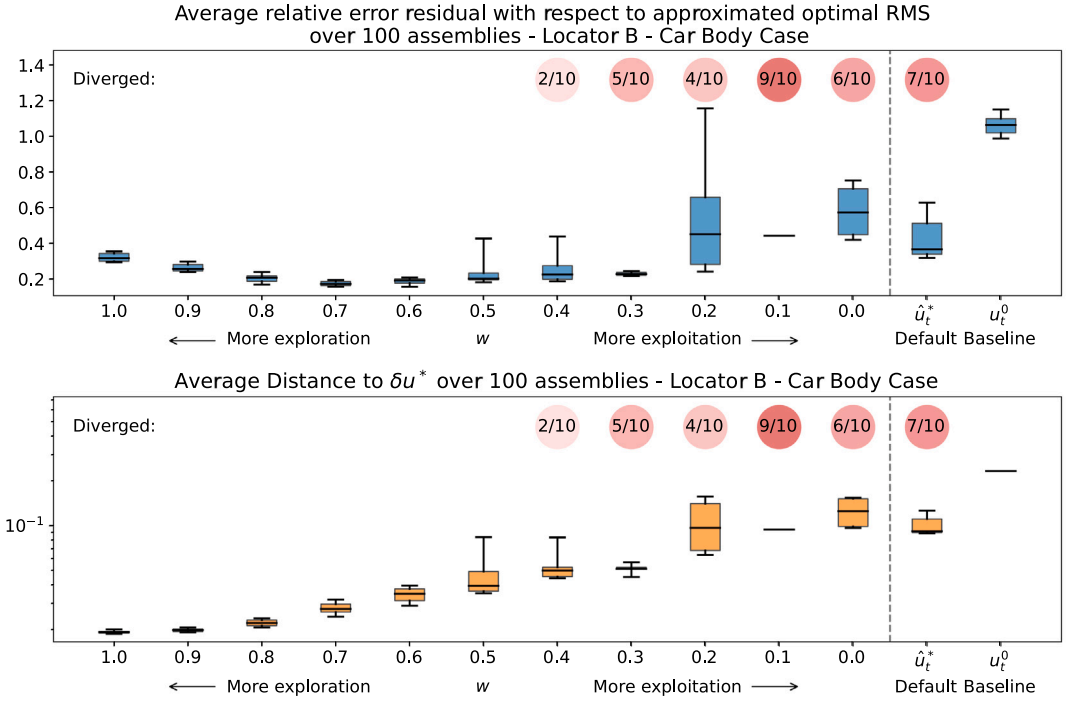


Fig. 5. (Car body case with locator B drifting.) Box plots (upper and lower) showing the spread of the average relative residual error quality of an episode and the average Euclidean distance between the estimation and offset error of an episode for all considered controllers over 10 different runs for the car body case. A run is a realization of an episode with deterministic randomness—same randomness (e.g., sequence order of assemblies and offset drifting direction) for all the controllers. Each run consists of 100 assemblies where clamp locator B is drifting. A controller diverged during a run if the state estimation $\widehat{\delta u}$ is far from the offset δu^*.

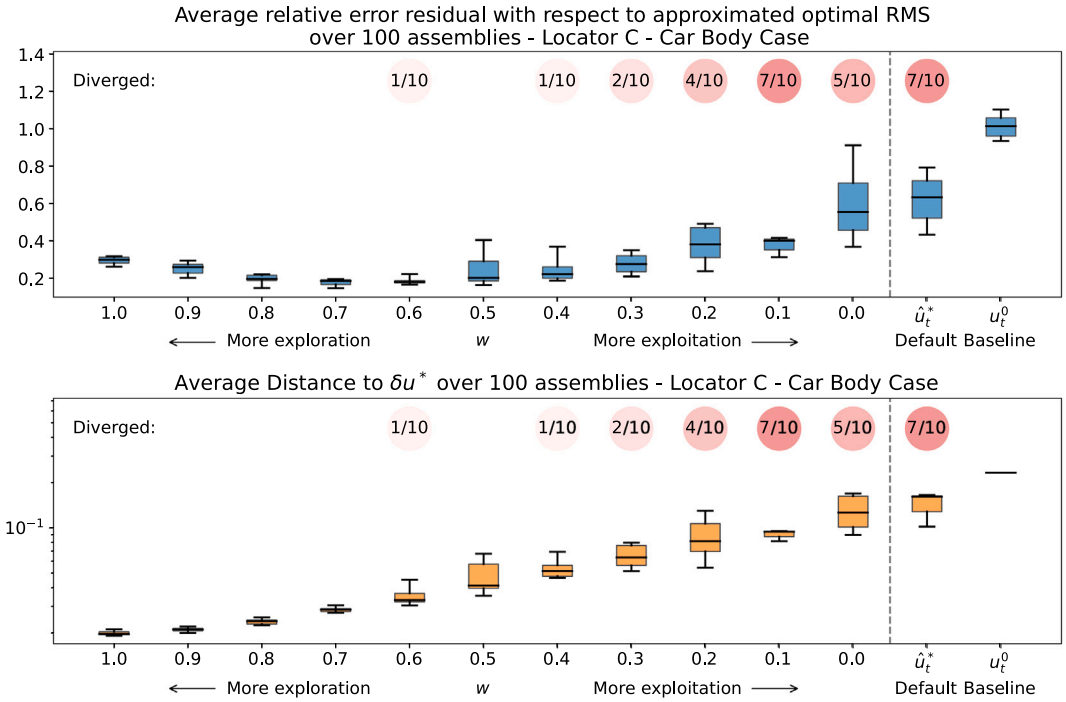


Fig. 6. (Car body case with locator C drifting.) Box plots (upper and lower) showing the spread of the average relative residual error quality of an episode and the average Euclidean distance between the estimation and offset error of an episode for all considered controllers over 10 seeded runs for the car body case. Each run consists of 100 assemblies where clamp locator C is drifting. A controller diverged during a run if the state estimation $\widehat{\delta \boldsymbol{u}}$ is far from the offset $\delta \boldsymbol{u}^*$.

3.1.2. Three drifting locators

Fig. 7 shows the results of the car body case for all considered controllers over 10 different runs, where all the three clamp locators A, B, and C are drifting simultaneously. The only controllers that did not diverge during these 10 runs are the proposed w-controller with $w = 1.0$ and the baseline controller, which cannot diverge by definition. However, the proposed control scheme with $w = 1.0$ outperforms the baseline controller with a large margin, since the average relative error residual quality is considerably smaller.

3.2. Test case B—Car door

Computer simulated experiments on test case B (see Fig. 3(b)) follow the same experiment plan as on test case A (three individually drifting locators, and three simultaneously drifting locators).

3.2.1. One drifting locator

Similarly for the car door case, the Figs. 8–10 exhibit the results for all the considered controllers over 10 different runs, where one clamp locator is drifting and consequently the only clamp locator considered by the controllers. In this case the clamp locator adjustments have a much smaller impact on the quality, and therefore there is a much smaller gain by utilizing our proposed controllers. Clearly, it is more beneficial with a high rate of exploitation, since the average relative error residual quality is lower at low values of w. In Fig. 9 we can see that the approximated optimal quality z_i^* is not correct for some of the assemblies since the average relative error residual quality has a lower value than zero. That is, on several assemblies the controller applied control signals that resulted in better quality than the approximated optimal quality and therefore the value is below zero.

3.2.2. Three drifting locators

Fig. 11 shows the results of the car door case of an episode when all three locators A, B, and C are drifting simultaneously, for all considered controllers over 10 different runs. The w-controller with $w = 0.5$ performs best, with no divergence and a low average relative error residual on the quality.

Table 1
The average amount of needed computational time, in seconds, per assembly for the different considered controllers to propose a control signal $\boldsymbol{u}$ and updating the state estimate $\widehat{\delta \boldsymbol{u}}$ and the uncertainty measure P.

Controllers	Car body		Car door	
	1 locator	3 locators	1 locator	3 locators
Default controller	1.6	7.9	2.0	16.2
w-controllers	3.1	100.1	5.4	178.2

3.3. Computation time

Table 1 shows the average amount of needed computation time in seconds per assembly for the different considered controllers. For the w-controllers we can see that there is a massive increase in computation time when considering three clamp locators compared to one.

4. Discussion

All the Figs. 4–11 show that higher exploration results in better estimations. This result was expected, based on the discussion in Section 2.3.2. However, we can see that the adequate estimations come with a cost since the quality is negatively affected for higher values of exploration rate. On the other hand, by only optimizing the expected next-assembly quality the model loses track of the offset. That in turn may result in extremely poor quality and often ultimately in the controller diverging, most likely due to ill-condition of the calibration problem. Thus, we see a clear trade-off between exploration and exploitation in the quality. Often a combination of both is most beneficial.

When considering the runs where controllers did not diverge, the different proposed w-controllers did on average perform better than

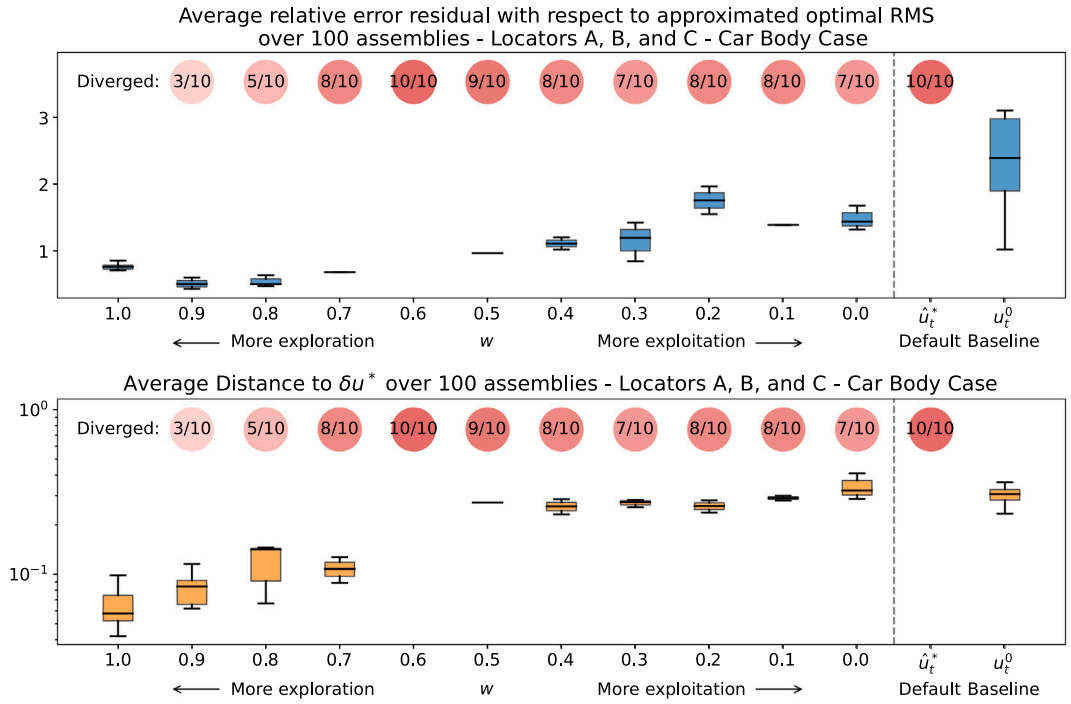


Fig. 7. (Car body case with locators A, B, and C drifting.) Box plots (upper and lower) showing the spread of the average relative residual error quality of an episode and the average Euclidean distance between the estimation and offset error of an episode for all considered controllers over 10 different runs for the car body case. A run is a realization of an episode with deterministic randomness—same randomness (e.g., sequence order of assemblies and offset drifting direction) for all the controllers. Each run consists of 100 assemblies where clamp locators A, B, and C are drifting. A controller diverged during a run if the state estimation $\widehat{\delta \boldsymbol{u}}$ is far from the offset $\delta \boldsymbol{u}^*$.

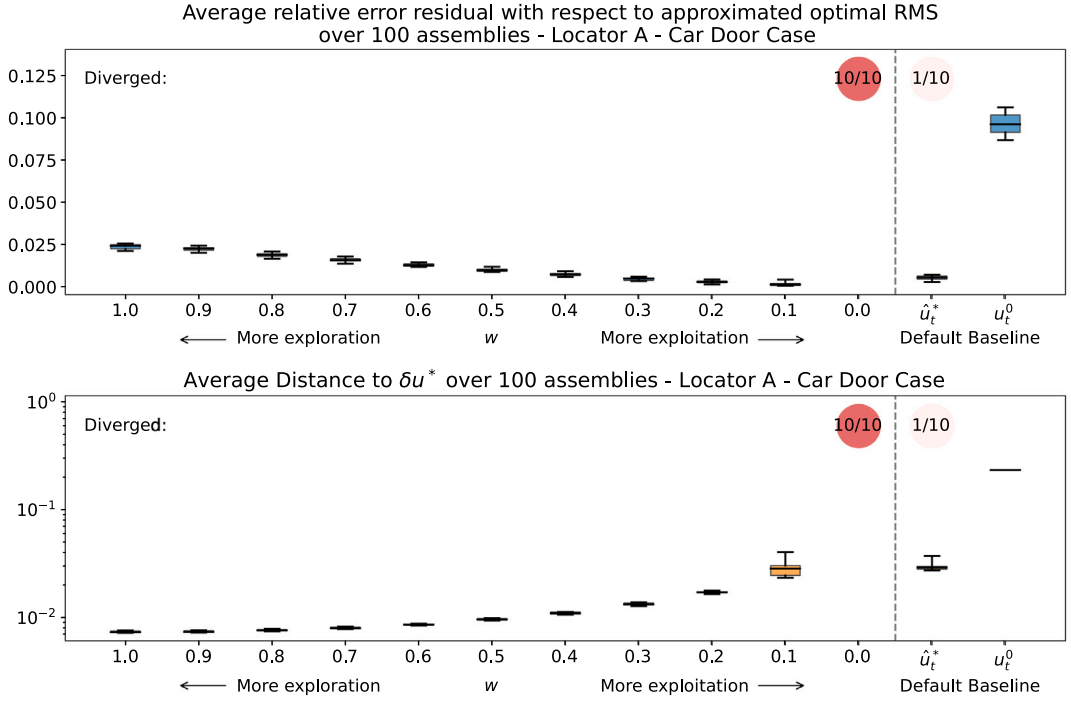


Fig. 8. (Car door case with locator A drifting.) Box plots (upper and lower) showing the spread of the average relative residual error quality of an episode and the average Euclidean distance between the estimation and offset error of an episode for all considered controllers over 10 different runs for the car door case. A run is a realization of an episode with deterministic randomness—same randomness (e.g., sequence order of assemblies and offset drifting direction) for all the controllers. Each run consists of 100 assemblies where clamp locator A is drifting. A controller diverged during a run if the state estimation $\widehat{\delta \boldsymbol{u}}$ is far from the offset $\delta \boldsymbol{u}^*$.

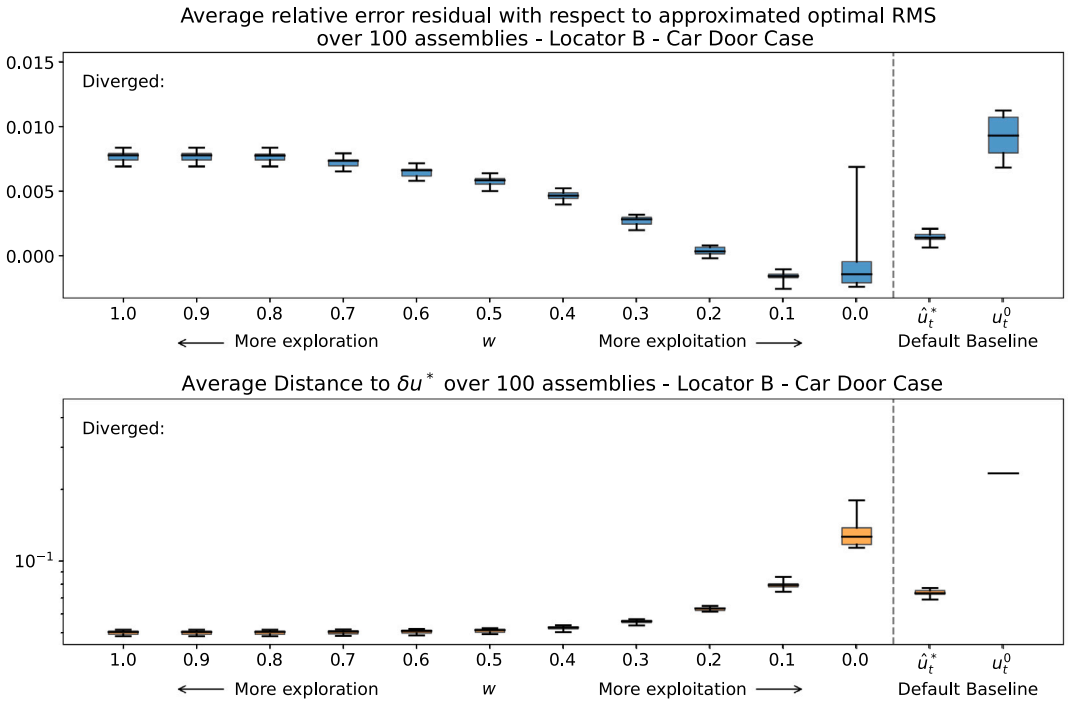


Fig. 9. (Car door case with locator B drifting.) Box plots (upper and lower) showing the spread of the average relative residual error quality of an episode and the average Euclidean distance between the estimation and offset error of an episode, respectively, for all considered controllers over 10 different runs for the car door case. A run is a realization of an episode with deterministic randomness—same randomness (e.g., sequence order of assemblies and offset drifting direction) for all the controllers. Each run consists of 100 assemblies where clamp locator B is drifting. A controller diverged during a run if the state estimation $\widehat{\delta \boldsymbol{u}}$ is far from the offset $\delta \boldsymbol{u}^*$.

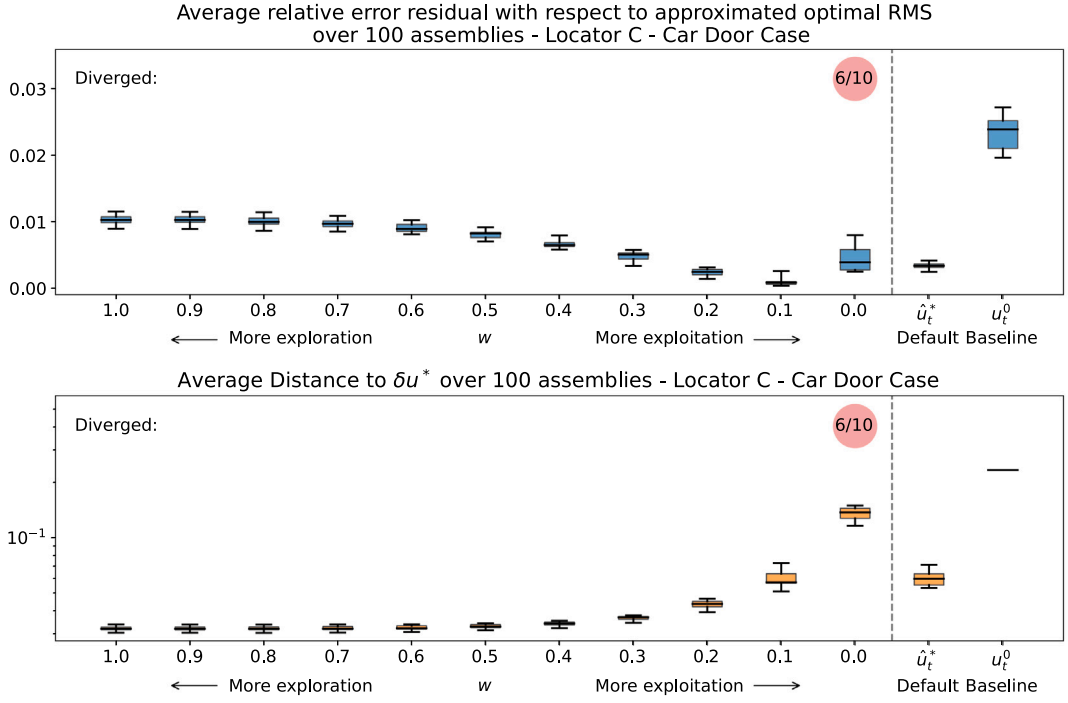


Fig. 10. (Car door case with locator C drifting.) Box plots (upper and lower) showing the spread of the average relative residual error quality of an episode and the average Euclidean distance between the estimation and offset error of an episode for all considered controllers over 10 different runs for the car door case. A run is a realization of an episode with deterministic randomness—same randomness (e.g., sequence order of assemblies and offset drifting direction) for all the controllers. Each run consists of 100 assemblies where clamp locator C is drifting. A controller diverged during a run if the state estimation $\widehat{\delta \boldsymbol{u}}$ is far from the offset $\delta \boldsymbol{u}^*$.

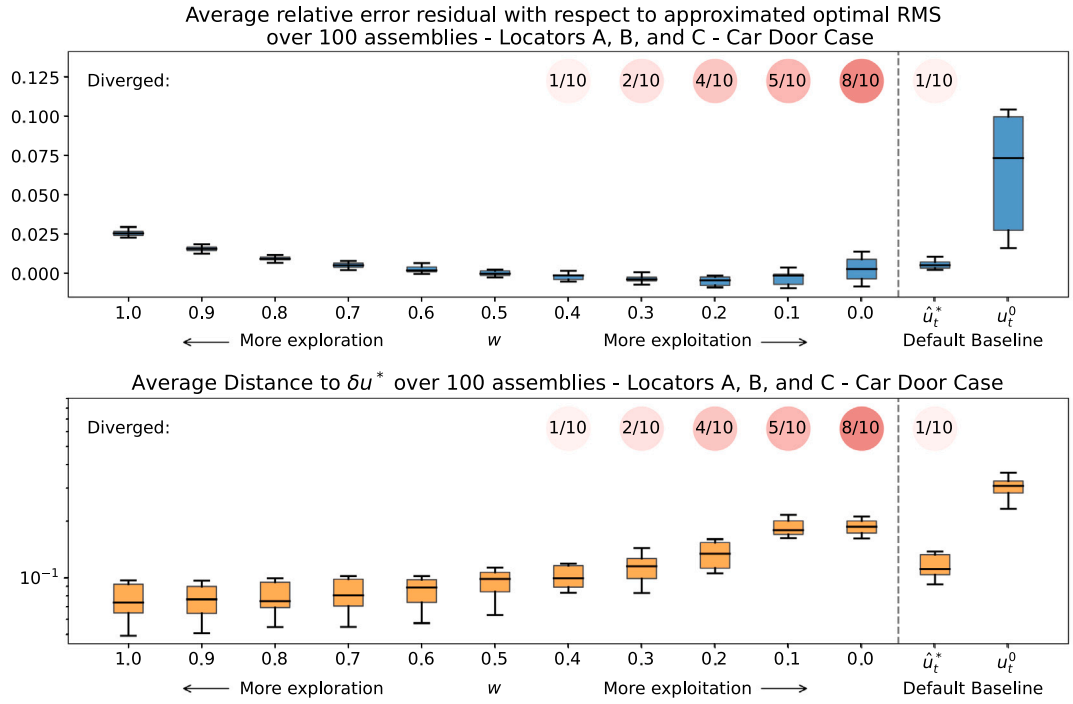


Fig. 11. (Car door case with locators A, B, and C drifting.) Box plots (upper and lower) showing the spread of the average relative residual error quality of an episode and the average Euclidean distance between the estimation and offset error of an episode, respectively, for all considered controllers over 10 different runs for the car door case. A run is a realization of an episode with deterministic randomness—same randomness (e.g., sequence order of assemblies and offset drifting direction) for all the controllers. Each run consists of 100 assemblies where clamp locators A, B, and C are drifting. A controller diverged during a run if the state estimation $\widehat{\delta \boldsymbol{u}}$ is far from the offset $\delta \boldsymbol{u}^*$.

the baseline controller, therefore stability is crucial. That is, emphasize having a high exploration rate. Especially, in the three dimensional car body case, where the only controller that was successful in all 10 runs was the full-exploration w-controller with $w = 1.0$, see Fig. 7.

It is not a trivial task how to choose the weight w. The limit where controllers start to diverge differs depending on which locators that are considered, which can be seen Figs. 4–11. This is likely in part due to the unweighted choice of (12). That is, the uncertainty corresponding to each component of $\widehat{\delta \boldsymbol{u}}$ is emphasized equally regardless of how easy or hard that component is to estimate, due to having a large or small effect on Q. By introducing a diagonal weight matrix W in (12),

$$L_t^{\text{explore}}(\boldsymbol{u}_t) := \frac{1}{d}\sqrt{\text{Tr}(W P_{t+1}(\boldsymbol{u}_t))},$$

it is possible that it could lead to a more generally valid choice for w. Such a weighting would however have to be inferred either from the inner structure of Q, or from observed data, which is an interesting avenue for future work.

The proposed w-controllers come with significant computational overhead, especially as the dimension of the state–space increases, which may limit the practical usefulness of our approach, as can be seen in Table 1. It does not scale well due to the curse of the dimensionality—both as a consequence of the increased number of Sigma points needed in the UKF and the increased number of evaluation points needed to approximate (14). An interesting line of future research would be to investigate the performance and stability of simplifications of the full control-scheme, such as, e.g., using parallel independent controllers for each component in the full state space.

5. Conclusions

Our computer simulated experiments demonstrated that it is possible to obtain an adequate estimate of the parameters for each assembly t. Further, we proposed the w-controllers that incorporate the uncertainty of the estimation in order to consider possible long-term quality gains. The proposed controllers demonstrated significant quality gains in the car body case, while in the car door case the gains were negligible to small. However, the car door case is rather insensitive to miscalibration which enables only small gains. As a final remark, it is important to emphasize having a high exploration rate, i.e., w close to one, in order to increase stability and thereby minimize the risk of divergence of the state estimation.

Declaration of competing interest

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

Acknowledgements

This work was financed by The Swedish Foundation for Strategic Research, through the Smart Assembly 4.0 project, within the Winquist Laboratory and was also in part developed in the Fraunhofer Cluster of Excellence *Cognitive Internet Technologies*.

References

[1] Shafto M, Conroy M, Doyle R, Glaessgen E, Kemp C, LeMoigne J, et al. Modeling, simulation, information technology & processing roadmap. Natl Aeronaut Space Adm 2012;32:1–38.
[2] Kritzinger W, Karner M, Traar G, Henjes J, Sihn W. Digital twin in manufacturing: A categorical literature review and classification. IFAC-PapersOnLine 2018;51(11):1016–22.
[3] Qi Q, Tao F. Digital twin and big data towards smart manufacturing and industry 4.0: 360 degree comparison. Ieee Access 2018;6:3585–93.
[4] Hochhalter JD, Leser WP, Newman JA, Glaessgen EH, Gupta VK, Yamakov V, et al. Coupling damage-sensing particles to the digitial twin concept. National Aeronautics and Space Administration, Langley Research Center; 2014.
[5] Grieves M, Vickers J. Digital twin: Mitigating unpredictable, undesirable emergent behavior in complex systems. In: Transdisciplinary perspectives on complex systems. Springer; 2017, p. 85–113.

[6] Fei T, Jiangfeng C, Qinglin Q, Zhang M, Zhang H, Fangyuan S. Digital twin-driven product design, manufacturing and service with big data. Int J Adv Manuf Technol 2018;94(9–12):3563–76.
[7] Rosen R, Von Wichert G, Lo G, Bettenhausen KD. About the importance of autonomy and digital twins for the future of manufacturing. IFAC-PapersOnLine 2015;48(3):567–72.
[8] Cunbo Z, Liu J, Xiong H. Digital twin-based smart production management and control framework for the complex product assembly shop-floor. Int J Adv Manuf Technol 2018;96(1–4):1149–63.
[9] Ding K, Chan FT, Zhang X, Zhou G, Zhang F. Defining a digital twin-based cyber-physical production system for autonomous manufacturing in smart shop floors. Int J Prod Res 2019;57(20):6315–34.
[10] Tao F, Zhang M. Digital twin shop-floor: a new shop-floor paradigm towards smart manufacturing. Ieee Access 2017;5:20418–27.
[11] Alam KM, El Saddik A. C2PS: A digital twin architecture reference model for the cloud-based cyber-physical systems. IEEE Access 2017;5:2050–62.
[12] Rosen R, Fischer J, Boschert S. Next generation digital twin: An ecosystem for mechatronic systems? IFAC-PapersOnLine 2019;52(15):265–70.
[13] Cronrath C, Ekström L, Lennartson B. Formal properties of the digital twin–Implications for learning, optimization, and control. In: 2020 IEEE 16th international conference on automation science and engineering. IEEE; 2020, p. 679–84.
[14] Yu J, Song Y, Tang D, Dai J. A digital twin approach based on nonparametric Bayesian network for complex system health monitoring. J Manuf Syst 2021;58:293–304.
[15] Lugaresi G, Matta A. Automated manufacturing system discovery and digital twin generation. J Manuf Syst 2021;59:51–66.
[16] Wang J, Li Y, Zhao R, Gao RX. Physics guided neural network for machining tool wear prediction. J Manuf Syst 2020;57:298–310.
[17] Liu S, Sun Y, Zheng P, Lu Y, Bao J. Establishing a reliable mechanism model of the digital twin machining system: An adaptive evaluation network approach. J Manuf Syst 2022;62:390–401.
[18] Söderberg R, Wärmefjord K, Carlson JS, Lindkvist L. Toward a digital twin for real-time geometry assurance in individualized production. CIRP Ann 2017;66(1):137–40.
[19] Wärmefjord K, Söderberg R, Lindau B, Lindkvist L, Lorin S. Joining in nonrigid variation simulation. In: Computer-aided technologies–applications in engineering and medicine. InTech Rijeka, Croatia; 2016.
[20] Bohlin R, Hagmar J, Bengtsson K, Lindkvist L, Carlson JS, Söderberg R. Data flow and communication framework supporting digital twin for geometry assurance. In: ASME international mechanical engineering congress and exposition. Vol. 58356, American Society of Mechanical Engineers; 2017, V002T02A110.
[21] Rezaei Aderiani A, Wärmefjord K, Söderberg R, Lindkvist L. Individualizing locator adjustments of assembly fixtures using a digital twin. J Comput Inf Sci Eng 2019;19(4).
[22] Cronrath C, Aderiani AR, Lennartson B. Enhancing digital twins through reinforcement learning. In: 2019 IEEE 15th international conference on automation science and engineering. IEEE; 2019, p. 293–8.
[23] Pei F-Q, Tong Y-F, Yuan M-H, Ding K, Chen X-H. The digital twin of the quality monitoring and control in the series solar cell production line. J Manuf Syst 2021;59:127–37.
[24] Grégorio J-L, Lartigue C, Thiébaut F, Lebrun R. A digital twin-based approach for the management of geometrical deviations during assembly processes. J Manuf Syst 2021;58:108–17.
[25] Claus F, Hamann B, Leitte H, Hagen H. Decomposing deviations of scanned surfaces of sheet metal assemblies. J Manuf Syst 2021;61:125–38.
[26] RD&T Technology AB. RD&T. 2021, http://rdnt.se/, (Accessed 9 September 2021).
[27] Söderberg R. Robust design by tolerance allocation considering quality and manufacturing cost. In: International design engineering technical conferences and computers and information in engineering conference. Vol. 97676, American Society of Mechanical Engineers; 1994, p. 219–26.
[28] Wan EA, Van Der Merwe R, Haykin S. The unscented Kalman filter. Kalman Filter Neural Netw 2001;5(2007):221–80.
[29] Welch G, Bishop G, et al. An introduction to the Kalman filter. Chapel Hill, NC, USA; 1995.
[30] Sobol' IM. On the distribution of points in a cube and the approximate evaluation of integrals. Zhurnal Vychislitel'noi Matematiki i Matematicheskoi Fiziki 1967;7(4):784–802.
[31] Jahangiri A, Katthe A, Sohrabi A, Liu X, Pulagam S, Balali V, et al. Developing a computer vision-based decision support system for intersection safety monitoring and assessment of vulnerable road users. 2020.
[32] Choi S-CT, Hickernell FJ, McCourt M, Sorokin A. QMCPy: A quasi-Monte Carlo python library. 2020, URL https://github.com/QMCSoftware/QMCSoftware.