On-line anomaly detection with advanced independent component analysis of multi-variate residual signals from causal relation networks.

LUGHOFER, E., ZAVOIANU, A.-C., POLLAK, R., PRATAMA, M., MEYER-HEYE, P., ZÖRRER, H., EITZINGER, C. and RADAUER, T.

2020

Manuscript Details

Manuscript number	INFFUS_2018_706
Title	On-line Anomaly Detection with Advanced Independent Component Analysis of Multi-Variate Residual Signals from Causal Relation Networks
Article type	Research paper

Abstract

Anomaly detection in today's industrial environments is an ambitious challenge in order to detect possible arising faults/problems, which may turn into severe waste during production, into defects or even into damages of systems components, at an early stage. Data-driven anomaly detection from multi-sensor networks faces challenges in proper data (information) fusion methodologies to establish adequate modeling cycles, whose outcome are models characterizing the anomaly-free reference situation based on which new on-line data are compared with, i.e. how much these deviate from them (pointing to potential faults). In this paper, we propose a new approach which is based on i) causal relation networks (CRNs) representing the inner causes and effects between sensor channels (or sensor nodes) in form of partial sub-relations, which are modeled by non-linear (fuzzy) regression models for characterizing the (local) degree of influences of the single causes on the effects, ii) an advanced analysis of the multi-variate residual signals obtained from the partial relations in the CRNs, which employs independent component analysis (ICA) to characterize hidden structures in the fused residuals (a significant change in these indicates an anomaly) and iii) automatized control limits on the energy content of latent variables obtained through the demixing matrix from ICA. Suppression of possible noise content in residuals --- to decrease the likelihood of false alarms ---, is achieved by performing ICA-based residual analysis solely on the dominant parts. Our approach was successfully evaluated for a real-world manufacturing process in the context of micro-fluidic chip production, where customer complaints arose about the quality of the chips during a specific production cycle $\rightarrow$ our approach could detect the anomaly in the process (leading to the bad quality chips) with negligible delay based on the process data recorded by multiple sensors in two production phases (injection molding and bonding), while it produced lower false alarm rates than several related and well-known state-of-the-art methods for (unsupervised) anomaly detection and while it also caused much lower parametrization efforts (in fact, none at all) to produce reliable results.

Keywords	on-line anomaly detection; causal relation networks; advanced multi-variate residual analysis; dominant parts of independent component analysis; automatized control limits; on-line production systems

Corresponding Author	Edwin Lughofer
Corresponding Author's Institution	Johannes Kepler University Linz
Order of Authors	Edwin Lughofer, Alexandru-Ciprian Zavoianu, Robert Pollak, Mahardhika Pratama, Pauline Meyer-Heye, Helmut Zörrer, Christian Eitzinger, Thomas Radauer

Submission Files Included in this PDF

File Name [File Type]

anomaly_detection_advanced_residualanalysis_manufacturing_systems_v02.pdf [Manuscript File]

To view all the submission files, including those not included in the PDF, click on the manuscript title on your EVISE Homepage, then click 'Download zip file'.

Research Data Related to this Submission

There are no linked research data sets for this submission. The following reason is given:
The data that has been used is confidential

On-line Anomaly Detection with Advanced Independent Component Analysis of Multi-Variate Residual Signals from Causal Relation Networks

Edwin Lughofer[a], Alexandru-Ciprian Zavoianu[a], Robert Pollak[a], Mahardhika Pratama[b], Pauline Meyer-Heye[c], Helmut Zörrer[c], Christian Eitzinger[c], Thomas Radauer[d]

[a] *Department of Knowledge-Based Mathematical Systems, Johannes Kepler University Linz, Austria*
[b] *School of Computer Science and Engineering, Nanyang Technological University, Singapore*
[c] *Profactor GmbH, Steyr-Gleink, Austria*
[d] *Stratec Consumables, Anif, Austria*

Abstract

Anomaly detection in today's industrial environments is an ambitious challenge in order to detect possible arising faults/problems, which may turn into severe waste during production, into defects or even into damages of systems components, at an early stage. Data-driven anomaly detection from multi-sensor networks faces challenges in proper data (information) fusion methodologies to establish adequate modeling cycles, whose outcome are models characterizing the anomaly-free reference situation based on which new on-line data are compared with, i.e. how much these deviate from them (pointing to potential faults). In this paper, we propose a new approach which is based on i) causal relation networks (CRNs) representing the inner causes and effects between sensor channels (or sensor nodes) in form of partial sub-relations, which are modeled by non-linear (fuzzy) regression models for characterizing the (local) degree of influences of the single causes on the effects, ii) an advanced analysis of the multi-variate residual signals obtained from the partial relations in the CRNs, which employs independent component analysis (ICA) to characterize hidden structures in the fused residuals (a significant change in these indicates an anomaly) and iii) automatized control limits on the energy content of latent variables obtained through the demixing matrix from ICA. Suppression of possible noise content in residuals — to decrease the likelihood of false alarms —, is achieved by performing ICA-based residual analysis solely on the dominant parts. Our approach was successfully evaluated for a real-world manufacturing process in the context of micro-fluidic chip production, where customer complaints arose about the quality of the chips during a specific production cycle → our approach could detect the anomaly in the process (leading to the bad quality chips) with negligible delay based on the process data recorded by multiple sensors in two production phases (injection molding and bonding), while it produced lower false alarm rates than several related and well-known state-of-the-art methods for (unsupervised) anomaly detection and while it also caused much lower parametrization efforts (in fact, none at all) to produce reliable results.

Email addresses: `edwin.lughofer@jku.at (Corresponding Author)` (Edwin Lughofer), (Alexandru-Ciprian Zavoianu), (Robert Pollak), (Mahardhika Pratama), (Pauline Meyer-Heye), (Helmut Zörrer), (Christian Eitzinger), (Thomas Radauer)

Keywords: on-line anomaly detection, causal relation networks, advanced multi-variate residual analysis, dominant parts of independent component analysis, automatized control limits, on-line production systems

1. Introduction

1.1. Motivation and State-of-the-Art

In todays large-scale manufacturing systems or factories of the future, the supervision of the whole production chain for approaching *zero defect manufacturing* [1] and realizing predictive maintenance [2] with high performance can be established through the usage of *data-driven anomaly and fault detection methods* [3] [4]. Thereby, an utmost goal is to detect any possible arising fault (problem) in the system at an early stage [5] [6] based on the trend or course of the current production process [7] [8] in order to take action to avoid severe failures and/or damages. Faults can be of different nature such as downtrends in the quality of the production reaching to defective production parts/items inducing significant waste and thus costs for the company [9] and/or even annoying customers (as was the case in our use case, see Section 5), or machine/tool failures [10] reaching to system dropouts increasing risks for operators and enlarging production system down-times resulting in losses for the company [11]. Any anomal behavior in the current production process may thereby indicate a potential fault/problem. In a data-driven fusion sense, it is typically reflected by multiple (sensor) measurement recordings which deviate from the past regular behavior or do not fit into the characteristics (density, shape, spread, ...) of past (fused) sample representations/distributions [12] [4].

Compared to predictive and forecast modeling methods [7] [13] and approaches with the usage of anomaly/fault prognostics and RUL (remaining useful life) estimation [14] [15] [16], which are able to predict quality downtrends through health indicators or other types of problems (failures, degradations, ...) in the future with a certain horizon, the essential point is that anomaly detection operates in a fully *unsupervised manner* (typically based on process data/models), which means that no quality information about the current process/system/product states needs to be available to establish appropriate predictors or forecast models. Therefore,

1. The automatization capability of such an approach is expected to be very high: the on-going, regular measurement recordings can be immediately taken as representatives for the fault-free production process — and a characterization can be built upon these, which can be further used as a fault-free reference situation.
2. Annotation effort in terms of labeling costs for historic data samples [17] [18] can be completely avoided. Opposed to classification approaches for fault detection [19] [20], where ML classifiers are trained based on pre-labelled data, there is no necessity to collect data in advance and to divide it into faulty and non-faulty phases. Often such data is not available at all and cannot be simulated easily without significant risks and/or production costs (especially not for fault/anomal cases, i.e. to enforce faults or non-acceptable production parts). This leads to imbalanced problems, where anomaly/fault-free phases are clearly over-represented.

3. It also abandons the necessity to have a kind of product quality index or even a failure index permanently measured over time, as is necessary for prognostics and forecast modeling methods [21] [7]. Such measurements are in some systems costly to obtain (especially if manually taken), which often ends up in small data set sizes for model training (even when collecting data over weeks and months), in other systems they are not really profitable or possible to install at all.
4. It is still possible to detect faults at an early stage (although no concrete prediction horizon is given), as the inter-relations between process variables are often expected to be violated *before* the faults becomes apparent/significant on the industrial plant or on the production items themselves — as could be, e.g., verified before in [22] [23] [24] for several industrial applications.

These circumstances trigger an applicability to a wide range of (on-line) quality control systems, where quality or failure (label) information cannot be provided at all [25].

Anomaly detection has been thus widely proposed in literature, ranging from pure statistical approaches in form of recognizing outliers from regular data distributions representing fault-free modes [26] [27] through measuring the deviations in transformed data [28] and (model) parameter spaces [29] to one-Class classification methods such as one-Class SVMs [30] [31] or support vector data description [32] — a survey of various methods can be found in [4]. The point is that all of these methods are typically operating in the high-dimensional feature space where indeed raw samples may be transformed to lower dimensions (e.g., through the usage of PCA and variants [33] [34] or through kernel regression [35]), but there is little or no interpretation in the model outputs, neither in the models at all. This also makes the localization of (detected) anomalies/fault difficult. Moreover, deviations of new incoming samples to learnt characterizations (shapes, orientations etc.) of regular (anomaly-free) operation modes are checked versus statistically motivated thresholds on a sample-wise basis, whereas the time component is often completely neglected (e.g., by supervising the trends of the residuals over time).

Both situations have been improved in previous works such as [22] [23] or [36], where partial SysID models have been trained from data to explain relations, dependencies contained in the system when everything is working properly / goes smoothly. The modular structure of the SysID models defined in different but not necessarily disjoint subspaces can be nicely exploited for fault localization purposes [37]. They, however, neglect the analysis of the multi-variate residuals (obtained as deviations between predicted and observed targets) directly in their joint space, as they solely act on a uni-variate basis (i.e., each residual signal is analyzed independently) and amalgamate fault warning by a simple OR-operator. Thus, these approaches are not able to identify hidden structures possibly contained in the multi-variate residual time-series, neither they are able to characterize possible dependencies (correlations) among residual signals which may become 'violated' in the case of anomalies. Furthermore, they apply conventional variable selection methods, thus looking for ordinary correlations between targets and inputs and not for real cause-effect relations, which weakens the insight into the system process (and contained dependencies).

1.2. Our Approach

In order to omit these bottlenecks mentioned above, our new approach goes significantly beyond state-of-the-art in terms of the following aspects:

- Elicitation of real (=statistically dependent) causal relations in the system based on a modified version of the PC algorithm [38] as a preliminary filter step; this automatically yields the SysID step, identifying which (target) variables (channels) can be explained by which others, however, automatically omitting trivial and/or fake-correlations among the variables. The output is a (interpretable) directed graph where an edge from one node N_1 to another N_2 represents a real causal influence of the corresponding variable (reflected by the node N_1 from where the edge starts) onto another variable (reflected by the node N_2 where the edge ends).

- Building up non-linear regression models based on the causal network structure with the usage of generalized TS fuzzy systems [39]: for each causal relation in the network, a non-linear (fuzzy) model is established for characterizing the degree of influence of each cause onto a particular effect. This is achieved through the usage of a particular learning scheme, which is able to learn the inner (rule) structure (including the adequate number of rules) and parameters from scratch, based on regularized weighted least squares, modified quantization error and statistical tolerance regions — note that such models have been also established in [22] [23], but 1.) using *conventional* TS fuzzy systems [40] having a limited approximation capability (by employing a particular top-down approach for rule learning [41]), and 2.) using conventional variable selection methods to achieve a dimensionality-reduced input structure — so, no real interpretable causal relations have been sought there.

- An advanced residual signal analysis, where each causal relation model serves as residual generator: therefore, our approach performs a multi-variate residual signal analysis realized through an advanced independent component analysis (ICA) (which is able to characterize hidden structures in the multi-dimensional residual space and thus to indicate changes of these in case of anomalies) and it establishes a fully automated thresholding through kernel density estimation (KDE) and CDF, without requiring any assumption on the distributions of the residuals (as was the case in the previous works [22] [23], which assumed normally distributed residuals). Furthermore, suppression of possible noise content in residual signals is elegantly achieved by decomposing the demixing matrix from ICA (into dominant and non-dominant parts) and by monitoring the energy conent of the latent variables (demixed residuals) solely on the dominant parts.

Our approach will be evaluated within the scope of a micro-fluidic chip production system, where two production stages are supervised in terms of anomalies or undesired changes during on-line production: injection molding and bonding. Therefore, particular data sets (from on-line measurements) have been drawn from the process: one data set includes the regular process behavior under anomaly-free conditions, the other data set comprises the time range of chip production for a particular order, for which *real customer complaints* about the final quality of the chips took

place. Thus, the first data set is used for checking whether our advanced residual-based anomaly detection based on the causal relation networks (established from preliminary regular production cycles — historic data from a data-base) are robust against false alarms, the second one whether our approach can reliably recognize the anomaly leading to the bad quality chips. An additional data set has been extracted from a production phase during which no customer complaints occurred, but where different machine parameter settings were used. In this sense, we could verify how robust our approach (and related SoA works, see below) is against such variations which are definitively no anomalies, but intended changes (thus, ideally, no anomaly should be detected). As these data sets were available from both production stages, we can also check where the anomaly can be recognized better.

The performance of our approach is compared to several state-of-the-art methods in anomaly detection (Section 6), where it turned out that our new approach could outperform various related (unsupervised) SoA methods for anomaly detection as well as the previous uni-variate residual-based methods in [22] [23], in terms of detection capability as well as false alarm rates — only support vector data description with sigmoid kernels could compete with our methods, but they required high tuning efforts for two most sensitive parameters within extensive trial-and-error runs, which are usually not possible when being installed in an on-line system as a kind of plug-and-play method.

2. Problem Statement and Basic Framework

We are aiming for an anomaly detection approach which is able to sufficiently cover the detection of as many various anomalies as possible at different parts of an industrial (manufacturing) system. This is because any anomaly can typically lead to real severe failures in the system latter, which may cause significant down-trends in product quality, may induce machine breakdowns or even may become risky for operators (e.g., consider a leaky emission pipe of a machine). Thereby, we assume that a permanent collection of measurements for recording the state of the on-line process is carried out — as we are aiming for a purely data-driven approach, we exclude the usage of expert knowledge or analytical insights (which are typically costly to obtain and not really flexible with respect to different product types, machine settings etc.). The measurements are typically recorded with the usage of multi-sensor networks [42] [43], especially in large-scale industrial systems (in order to supervise several parts/chains and not to be restricted to one particular phase).

A typical sensor network example is shown in Figure 1, containing fives sensors located at five different sites; each of the sensors contains one or more channels, as indicated by the number of strokes going into them. We assume that the channels are dynamically and continuously measured over time and recorded through the central data base sink (bottom box). Sensors #2, #3 and #4 are connected, thus could exchange the channel data and perform a partial modeling on their channel views. The data is fused and stored at a central data base server (bottom box) which is connected to all sensors (and included channel), thus is able to process all 16 channels in parallel together with the event signals coming from a process control system. Synchronization and fusion of the data is a sophisticated

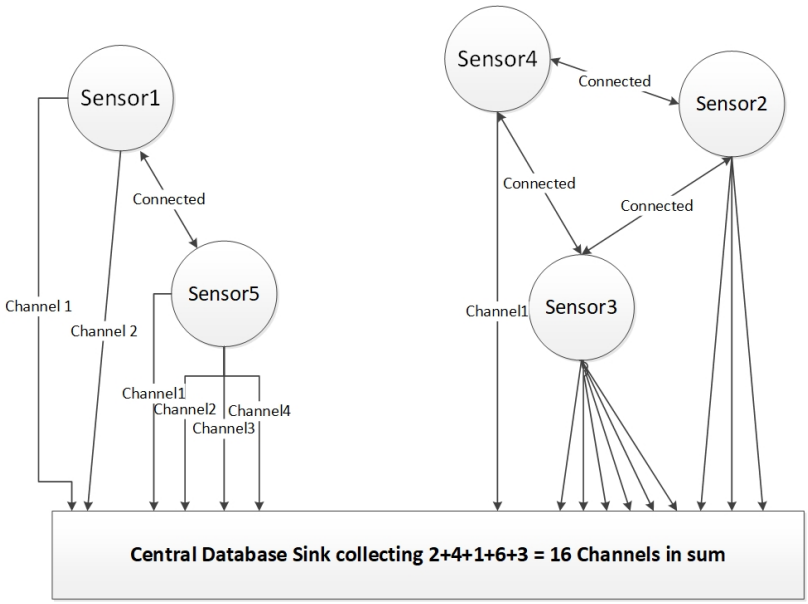

Figure 1: Example of a multi-sensor network with five spatially distributed sensor sites, each sensor measuring one or more channels, which are all collected in a central data sink.

challenge, especially in Big Data applications, but not the focus of this paper — we thus refer to [44], [45]. We assume that the data-driven modeling phase can be performed on the central data base server. In principle, the modeling and also the whole FD can be conducted in each sensor individually, which is for instance established in so-called smart or intelligent sensors embedding some artificial intelligence on chip devices [46] [47]. However, this typically leads to a smaller partial view of the whole possible interrelations between channels in the system than when performing the modeling at the central site, using the information from all sensor together — in our approach, a data and information fusion is achieved through model construction (CRNs + fuzzy regression).

In particular, in a such a network containing M sensors measuring M process values, the idea is to identify multivariate (causal) relations between these at the central site by including as many of these M process values in the cause-effect structures (see Section 3). These relations are catered to represent prevalent causal structures in regular, anomaly-free production mode, and any 'violations' of these in newly recorded (on-line) data may indicate a potential problem in the system, thus an anomaly. According to the degree and distribution of violations which are reflected in changes in the multi-variate residual signals, it can be decided whether a real anomaly occurred or whether it is because of a regular system dynamics or because of noise, see Section 4. The principle structure of our anomaly detection approach is shown in the framework in Figure 2, whose components and their methodological realizations will be detailed in the next two subsections. The upper part presents the off-line phase, where the causal relation network and its implicit (cause-effect) prediction models are established (Section 3, the lower part shows the on-line

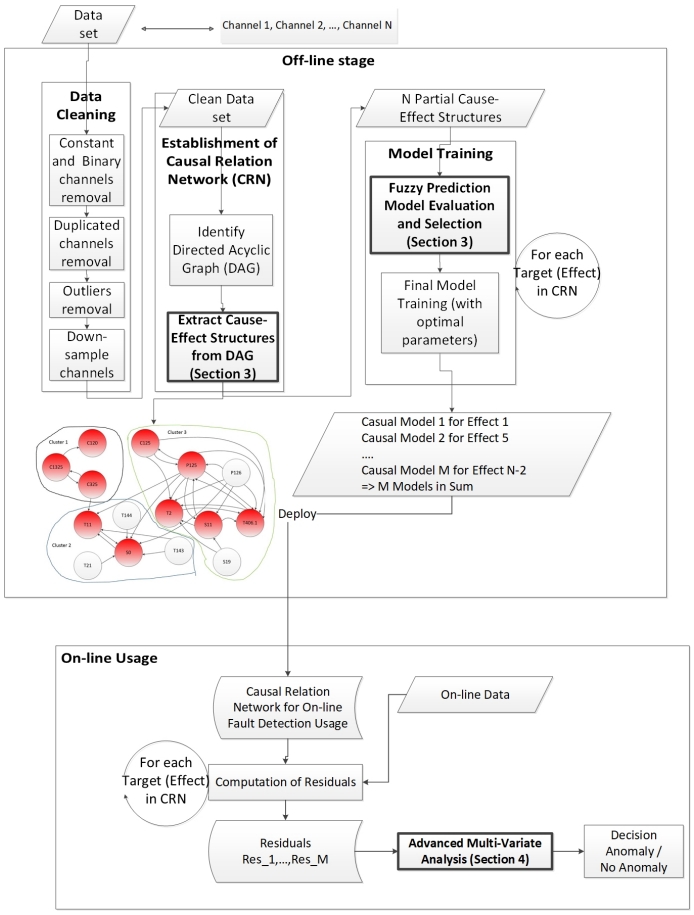

Figure 2: Framework of the Anomaly Detection Strategy; sections where components are described in more detail enumerated as such in bold font.

phase where anomaly detection is carried out based on the identified relations. Any calculation of statistical measures and of advanced multivariate analysis of the residual signals will be carried out in incremental, single-pass manner to guarantee high-speed on-line functionality, see Section 4. Data cleaning is carried out in advance for which we have applied standard algorithms, see also [22], and are thus not explained here further in detail.

3. Establishing the Causal Relation Structure and the Prediction Models

Our approach for setting up models from the multi-variate (eventually large-scale) sensor data for describing the internal dependencies contained within the system under regular, anomaly-free conditions, is based on two stages:

- In the first stage, causal relations are sought to explain the internal cause-effect structures between the system variables (often recorded through channels); this is done by seeking for real structures in a statistical sense (in

form of actual conditional dependencies), omitting any fake-correlations or trivial dependencies which do not actually show the real relations (thus, giving no new insights) and therefore usually also do not contribute to the performance of anomaly detectors based on the models.

- In the second stage, the partial causal structures between the variables — realized and represented through a directed graph, termed as *causal relation network (CRN)* — are used for establishing multi-variate (non-linear) regression models. Thereby, each effect in the CRN is used as target and each cause for each effect as input (with multiple causes for one effect possible). The regression coefficients for each cause then indicate the impact/weight of this cause on the effect, in relation to the other causes, rather than showing the conditional dependencies between causes and effects (as achieved in Bayesian networks [48] — thus, our approach differs from these).

The first stage is realized through the usage of a fast version of the PC algorithm [38] [49], which operates in two steps. In the first step, it learns from data a skeleton graph, which contains only undirected edges. In the second step, it orients the undirected edges to form equivalence classes of directed acyclic graphs (DAGs). Each equivalence class can then be interpreted as a cause-effect relation, where the nodes denote the variables (sensor channels) and the directed edges the cause-effects (each directed edge starting at a cause and ending at an effect). The theoretical foundation of the PC algorithm [38] [49] thereby is that if there is no link (edge) between nodes (variables) x_i and x_j, then there is a set of vertices Z that either are neighbours of x_i and x_j such that x_i and x_j are independent conditioning on Z. In other words, Z disconnects x_i and x_j. Hence, in a first step a fully connected graph (all variables are connected with each other is formed) is formed, which is iteratively 'out-sparsed' by removing edges whenever there is a subset of variables in the neighborhood of x_i and x_j, which relieves an independence between x_i and x_j whenever being conditioned on it. Such a conditioned independence can be decided at the light of independence statistical tests based on the data set. PC algorithm thereby uses the chi-square test based on the cross entropy statistics. After 'out-sparsing', the edges are directed due to a so-called *orientation step*. The orientation step will proceed by looking for sets of three variables $\{x_i, x_j, Z\}$ such that edges $x_i \rightarrow Z$, $x_j \rightarrow Z$ are in the graph by not the edge $x_i \rightarrow x_j$. Then, if $Z \notin S_{x_i,x_j}$, it orients the edges from x_i to Z and from x_j to Z creating a v-structure: $x_i \rightarrow Z \leftarrow x_j$. Figure 3 shows an example of a causal relation network (from a real-world production process with masked variable names) which embeds a kind clustered structure (indicated by surrounded convex hull drawings), showing which variables and thus parts of the system are more closely related and linked. This may even serve as additional information for experts/operators for gaining further insights into the production process.

In the second stage, the concrete causal relation models between all causes and effects are established. Therefore, each variable appearing as effect in the CRN (as shown by a red node in the example in Figure 3), is used as a target variable and each cause flowing into the same effect is used as an input variable for establishing a relation model. We aim for both, i) robustness with respect to noise and ii) multi-variate characteristics of the relation model,

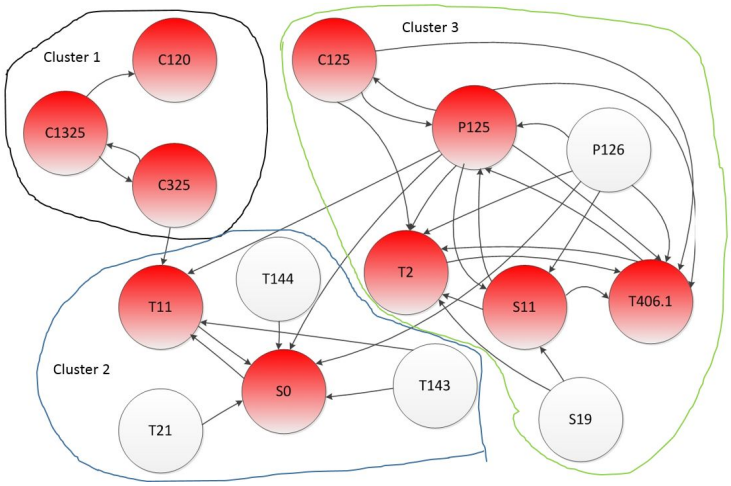

Figure 3: A typical causal relation network from an industrial production system; dark (red) nodes denote the effects in the relations, all inflowing vertices indicate the causes.

the latter basically in order to achieve a relative weight/impact of each cause among all causes for one particular effect. The combination of both can be best achieved with a regularized multi-variate regression approach, where the regression coefficients directly represent the relative weights and a regularizer in the objective function guarantees for robustness in case of noise and high-dimensional input space. Furthermore, the causal relation may have a different characteristics with different levels of influences between causes and effects in different parts of the feature space. Thus, a local partitioning of the regression model into different sub-models may be an important issue as well.

Therefore, we employ the structure of Takagi-Sugeno (TS) fuzzy systems, which are able to decompose the input space into partial local regions (rules) and to represent local relations in the context of (weighted) multi-variate regression. This is because in each rule one local linear regression model is established to form the consequents. In functional form, a TS fuzzy system is defined as:

$$\hat{f}(\vec{x}) = \sum_{i=1}^{C} \Psi_i(\vec{x}) \cdot l_i(\vec{x}) \quad \Psi_i(\vec{x}) = \frac{\mu_i(\vec{x})}{\sum_{j=1}^{C} \mu_j(\vec{x})} \quad , \tag{1}$$

with C the number of rules (local partitions) and $l_i(\vec{x}) = w_{i0} + w_{i1}x_1 + ... + w_{ip}x_p$ a linear hyper-plane describing the local regression trend of the i-th rule (we assume an input dimensionality of p, which can vary among different cause-effect relations). When using the generalized version, i.e. arbitrarily rotated rules in the input space (rather than axis-parallel ones as in the classical case proposed in [40]), the membership degree $\mu_i(\vec{x})$ to the ith rule is estimated by

$$\mu_i(\vec{x}) = \exp(-\frac{1}{2}(\vec{x} - \vec{c}_i)^T \Sigma_i^{-1} (\vec{x} - \vec{c}_i)) \tag{2}$$

with $\vec{c}_i$ the center and Σ_i^{-1} the inverse covariance matrix of the ith rule, describing its shape and orientation. As

examined in [50] [39], the generalized variant of TS fuzzy systems has some advantages regarding compactness of the rule base and better approximation capabilities compared to classical TS fuzzy systems (as having been used in our previous works in [22] [23]).

For learning the non-linear antecedent parameters and the adequate number of rules, we interpret the data set as a pseudo-stream and apply *Gen-Smart-EFS* learning engine [39] (a previous development by the main author of this paper), which is a single-pass method embedding incremental merging [51] and splitting operations [52] for establishing homogenous and compact rules (omitting strong rule overlaps and thus redundant local models as well as blown-up heterogenous rules). Additionally, specific fine-tuning iterations are applied to optimize the positioning and shapes of the rules according to a modified version of expected quantization error [53]. Once the centers and covariance matrices are estimated from data by *Gen-Smart-EFS*, the following regularized objective function is optimized to find the regression coefficients of each local linear model l_i separately:

$$J_i = \sum_{k=1}^{N} \Psi_i(\vec{x}(k)) e_i^2(k) + \lambda \sum_{j=1}^{p} (\alpha w_{ij}^2 + (1-\alpha)|w_{ij}|) \longrightarrow \min_{\vec{w_i}} \tag{3}$$

where $e_i(k) = y(k) - \hat{y}_i(k)$ represents the error of the local linear model in the kth sample (N samples in sum), λ the regularization parameter and α a parameter in $[0, 1]$, steering the degree of influence of the Lagrangian term stemming from the classical Lasso approach, i.e., $\sum_{j=1}^{p} |w_{ij}|$, versus the Lagrangian term inspired by classical ridge regression, i.e., $\sum_{j=1}^{p} w_{ij}^2$. In combination, both terms are also used in the elastic net approach [54] (for classical global linear regression), thus by solving the weighted formulation in (3) by using cyclical coordinate descent method along regularization paths [55], and this with weights given by the rule membership degrees, we term our approach as *fuzzily weighted elastic net*. The separate weighted optimization of consequent parameters, no matter whether used in plain or regularized form, has several advantages over a joint global one regarding robustness and interpretability — as deeply examined in [56] (Chapter 2).

4. Advanced Residual Signal Analysis

Once all partial relations embedded in the CRNs have been established, it is possible to check new incoming on-line samples how well they 'fit' into the network — or, in other words, whether any of the causal relations are 'violated', thus not valid any longer for new on-line samples → if this would be the case, an anomaly may be reported or at least a potential anomaly candidate is found. In our case of having established partial regression fits, violations can be checked by inspecting the residuals between real (measured) and predicted effects:

$$res_i = y_i - \hat{y}_i \quad \forall i = 1, ..., M \tag{4}$$

The predicted effects $\hat{y}_i$ are obtained by predicting the effects using the single causes from the on-line sample, the real effects are part in the on-line samples, as all sensor signals are assumed to be permanently measured (→ no supervised

information, thus no annotation needed). M is the number of causal relations found and used for on-line processing due to sufficient quality of the regression fit (as, e.g., calculated by the R^2 measure on observed versus predicted targets from left-out folds in the training data set, e.g., during cross-validation [57]).

Thus, for each causal relation with sufficient quality, a residual signal $res_i(k)$ (with k the time instance) can be elicited and tracked over time to check for atypical appearances. This has been established in a univariate manner (inspecting each signal separately and independently) in the previous works in [22] [23] with the usage of a statistical tolerance band, which could be dynamically and recursively (exactly) updated upon and with fault-free on-line data. However, this approach had three major drawbacks:

- It assumes Gaussian normal distribution of residuals to establish a tolerance band, i.e. a threshold for deciding whether new samples are abnormal or not. In some applications, this may be not necessarily the case, trending or skewed residuals have been observed before, e.g., [58].
- It establishes the tolerance band for each residual signal separately and independently and then performs a decision based on winnner-takes-all approach: if one residual signal lies over the tolerance band, an anomaly (fault) is reported. Thus, it is not able to actually characterize the multi-variate characteristics of the residual time series (and to identify any hidden structures/dependencies among these which should be valid).
- It applies a fixed value of a multiplication factor n (for σ) in the tolerance band — indeed, this value can be tuned through so-called ROC (receiver-operating characteristics) curves [59], but only in an a posteriori manner to evaluate the achievable bounds of the whole approach — as conducted in [22] [23].

Thus, in order to make the approach independent from the distribution of the residuals, to expand it to a full and multi-variate characterizing analysis of residuals and to omit any manually tuned a-priori threshold (usually a vague default guess due to past experience), we propose two alternative ways: i) a normalized version of SPE (squared prediction error) statistics [60] where the statistically ideal control limits on SPE are estimated from training data using kernel density estimation (KDE) [61], and ii) automatized control limits on the energy content in independent components obtained through de-mixing matrices from independent component analysis (ICA). We further apply a special decomposition of demixing (into dominant and non-dominant parts) to automatically reduce noise effects in the signals. We explain both in more detail in the following subsections.

4.1. Normalized SPE Statistics with Fully Automated Threshold

As a light and thus also very fast variant (ready-made for on-line and real-time analysis) of a joint multi-variate residual analysis, we propose the usage of the SPE statistics, which at time instance k (for which residuals $\vec{res} = [res_i(k)], i = 1, ..., M$ are produced) is defined by:

$$SPE(k) = \vec{res}(k)^T \vec{res}(k) = \sum_{i=1}^{M} (res_i(k))^2 \tag{5}$$

The problem with this original definition (also used in other fault detection approaches) is that it over-weights residuals from models with either larger ranges of the targets (effects) or also with larger ranges of the residuals themselves — for instance, a model producing very low residuals in as range of $[-0.001, 0.001]$ (for the regular, anomaly-free case) will be completely 'masked out' in the calculation of (5) when (regular) residuals from other models range in $[-1, 1]$ — this can be also the case when a significant deviation is observed (e.g., a 10 times higher value of 0.01 will become not visible in SPE). Even though when performing a normalization of all targets in a causal relation network, e.g, to the interval of $[0, 1]$, and thus to assure the same ranges across all targets, the ranges of the residuals during the normal, anomaly-free phase may vary from model to model — this may depend on the prediction accuracy (or quality) of the model on fault-free data. Therefore, we suggest the usage of normalized SPE statistics (NSPE), which is defined by:

$$SPE(k) = \vec{resn}(k)^T \vec{resn}(k) = \sum_{i=1}^{M} (resn_i(k))^2 \tag{6}$$

where $resn_i(k) = \frac{res_i(k)}{max_{j=1,...,N}(res_i(j)) - min_{j=1,...,N}(res_i(j))}$, where N is then number of training samples, thus the actual residual in the kth on-line sample is normalized with the ranges of the residuals obtained on the training matrix. A more robust normalization by omitting outliers, anomalies in the training data itself can be used by using above 90th and below 10th percentiles instead of maximum and minimum, respectively.

The problem now remains how to extract control limits as violation thresholds above which the SPE value can be seen as untypical and thus may indicate a potential anomaly. Therefore, we adopt the idea of non-parametric kernel density estimation (KDE) [61] [62] in order to estimate the cumulative distribution function (CDF) of the SPE statistics as achieved when applying it to the training data, resulting in a vector of SPE values over all residuals obtained from the training data. In KDE, the cumulative distribution function is estimated by [62]:

$$\hat{F}(x) = \frac{1}{N} \sum_{i=1}^{N} V(\frac{x - x_i}{\sigma}) \tag{7}$$

where σ is a smoothing parameter, N the number of (training) samples and $V(x) = \int_{-\infty}^{x} K(u)du$ with K a kernel function. The determination of the smoothing parameter can be done by following the guidelines mentioned in [62] and [63]. However, it is usually not so sensitive when requiring a significance threshold level that a time-series/signal stay within its allowed limits — the principal monotonic trend of the CDF is then sufficient. An example of CDF estimation on an empirical distribution of ascending SPE values (along a data set from a real-world manufacturing process, see Section 5) is provided in Figure 4.

Once having the CDF estimated, it is easy to extract an anomaly detection threshold as a confidence limit $1 - \alpha$, based on a significance level α (typical values are 0.05, 0.025 and 0.01 — we used $\alpha = 0.01$ in all our experiments to avoid false alarms as much as possible). The 0.99 confidence limit is indicated by a vertical line in Figure 4: all SPE values occurring above 1.15 are thus indicating a potential anomaly.

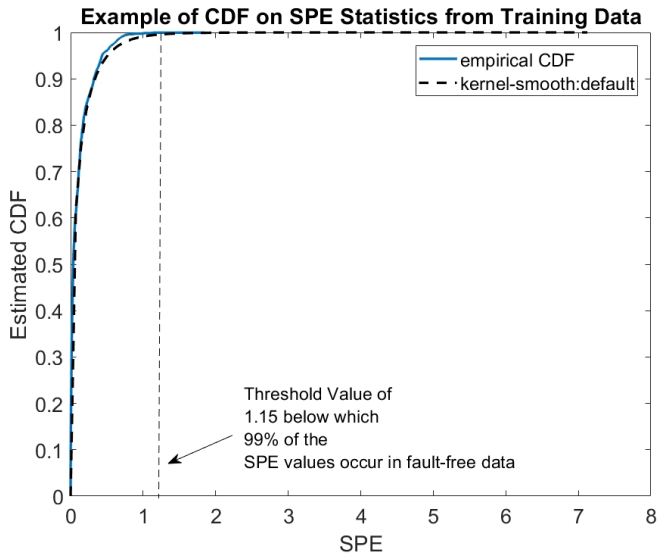

Figure 4: An example of a CDF for SPE statistics extracted from fault-free training data set; the vertical line denotes the SPE value (of 1.15) above which only 1% of the values achieved on the training data set lies, thus usable as confidence limit on SPE values extracted for new on-line samples.

4.2. Control Limits through Independent Component Analysis (ICA)

One major problem with the SPE statistics is that it is a naive sum over multi-variate residual samples, thus any (single) peaks and significant atypical residuals arising due to significant noise in the data recordings (but not due to a real anomaly) are directly reflected there, easily exceeding the tolerance limits as calculated through cumulative distribution estimation (7) — explicit filtering of the residuals (peaks) would be an option to reduce false alarms (as proposed in [22]), but requires additional parameters for filter design and leaves pretty much possibilities regarding the selection of the filter type (which is hardly possible to do reliably in advance for a new application scenario/data stream etc.). Real-world examples of such peak-type SPE statistics in the fault-free case will be shown in the results section.

Another shortcoming of SPE statistics is that it only reflects the amount of residual energy from all causal relations (in a direct measurable way), but it does not exhibit any hidden fundamental structure possibly occurring in multi-variate time-series — for instance, certain (in)dependencies between several residuals (e.g., arising due to partial behavior sharings among causal relations) which hold in the nominal anomaly-free case. When anomalies happen, such (more complex) hidden structure may become ”violated” and thus detectable. Independent component analysis (ICA) is a technique which addresses this problematic issue by attempting to find latent variables without assuming to have Gaussian distributed data at hand. The LVs are linear combinations of observed variables and statistically independent as possible of each other [64]. In our setting of multi-variate residual signals R, the relation between the

original signals and the latent variables is thus described as:

$$R = A * S + E \tag{8}$$

where $R \in \mathbb{R}^{M \times N}$ is the residual matrix composed of N residual vectors (=multi-variate signal samples with M residual signals from M causal relations observed in parallel), which are assumed to reflect a kind of mixture or superpositions of (hidden) independent (latent) variables stored in $S \in \mathbb{R}^{D \times N}$ — this assumption is realistic because usually several variables may appear as either causes or effects in different partial causal relations (within the CRN); hence, the residuals from different relation-based regression fits are usually not independent per se. M is the number of causal relations found and used for residual extraction due to sufficient quality of the regression fit. $A \in \mathbb{R}^{M \times D}$ (with D the reduced dimensionality of LVs) denotes the mixing matrix, containing the basic information about the hidden dependence structure contained in the multi-variate residuals. Therefore, its counterpart, the demixing matrix $W \in \mathbb{R}^{D \times M}$ contains the basic information about the *hidden independence structure*, i.e. the knowledge how to demix a multi-variate residual to achieve independent components S. The purpose of ICA is thus also to obtain such a demixing matrix in order to obtain reconstructed S: $\hat{S} = WR$. This can usually not be solved in exact manner, thus an approximative way is followed within iterative optimization cycles. One of the most famous procedure for obtaining A and W is the *fast ICA* method [65], which is a fixed point algorithm, solving the problem via the Newton method (we used its MATLAB implementation for all evaluation purposes).

The idea for the purpose of anomaly detection is now to use the demixing matrix W in order to demix new residual vectors $\vec{r}_{new}$ into the (nearly) independent LVs $\vec{s}_{new}$: $\vec{s}_{new} = W\vec{r}_{new}$. Thereby, W is determined once during the training phase, based on fault-free training data (representing regular operation modes etc.). Whenever $\vec{s}_{new}$ shows an untypical appearance compared to the vectors $\vec{s}_1, ..., \vec{s}_N$ obtained on the training data itself, it may point to a change in the hidden independence structure (as W determined from the training data does not match the real structure any longer). Such an untypical appearance can be recognized due to a significant increase of the energy of the s-vectors, which can be measured in terms of squared L^2-norm of the vectors (termed as I^2 statistics), thus by:

$$I^2(new) = \vec{s}_{new}^T \vec{s}_{new} \tag{9}$$

A significant increase of $I^2(new)$ can be again elicited through a confidence limit, which is extracted based on the cumulative distribution estimation through KDE — as discussed in the previous section for the SPE statistics, see also Figure 4 for an example.

Finally, we address the first problem discussed in the first paragraph of this section (noise problematic) by not using the whole matrix W, but only the most *dominant parts* in W. Thereby, our idea is to re-arrange the rows of W (each one containing the mixing weights for one LV) in descending order according to their L^2 norms and take the first X% stored in W_d when multiplying with $\vec{r}_{new}$, thus

$$\vec{s}_{new} = W_d * \vec{r}_{new} \tag{10}$$

This means that we omit non-dominant parts (with low LV loadings) for the reconstruction step of a new residual vector; the non-dominant parts are typically more intensively affected by noise in the data than the dominant parts (due to higher sensitivity in smaller changes). In this sense, we suppress untypical noise level in testing data, while assuming that real anomalies contain a more intensive changing behavior onto signal reconstruction (and are thus still (more) reflected in $\vec{s}_{new}$ when being reconstructed from the dominant part). We will verify this claim empirically in the results section, where we will check the performance of I^2 in (9) when using both, dominant and non-dominant parts of W for reconstructing $\vec{s}_{new}$.

4.3. Integration of Model Uncertainty in Residuals

Finally, we describe how we integrated possible uncertainties in predicted model outputs into the residuals, obtaining 'normalized' residual signals $resn_i(k)$, in order to give residuals higher weights in case of more certain model outputs. The approaches described above with the control limits etc. can be directly applied to the normalized residuals. This is achieved by dividing the original signal through the confidence level of the output, thus by

$$resn_i(k) = \frac{y_i(k) - \hat{y}_i(k)}{conf_i(k)} \quad \forall i = 1, ..., M. \tag{11}$$

If $conf_i(k)$ is calculated as a model error measure (e.g., mean absolute error, mean squared error, ...) and thus serving as global band then $conf_i(k_1) = conf_i(k_2)$ for all k_1, k_2 time instances, but usually $conf_i(k_1) \neq conf_j(k_1)$. Locality of $conf_i(k)$, i.e. $conf_i(k_1) \neq conf_i(k_2)$, can be achieved by so-called *local error bars*. For linear regression, these are given through the uncertainty of the inverse covariance matrix, hence by:

$$conf_i(k) = \sqrt{diag(cov\{\widehat{y_i}(k)\})} \tag{12}$$

with

$$cov\{\widehat{y_i}(k)\} = \sigma^2 \vec{x}_k (X^T X)^{-1} \vec{x}_k^T \tag{13}$$

with X the regression matrix obtained during training phase, $\vec{x}_k$ the current input sample = cause for which the effect y is predicted, and σ^2 and unbiased estimator of the noise level. For TS fuzzy systems, we exploit the statistically motivated derivation in [66] by using an extended (local) noise estimator $\hat{\sigma}_j$ from [67], which yields:

$$conf_i(k) = \sum_{j=1}^{C} t_{\alpha,\Sigma(k)-deg} \hat{\sigma}_j \sqrt{(\vec{x}_k \Psi_j(\vec{x}_k))^T P_j (\Psi_j(\vec{x}_k)\vec{x}_k)} \tag{14}$$

where $t_{\alpha,\Sigma(N)-deg}$ stands for the percentile of the t-distribution for $100(1-2\alpha)$ percentage confidence interval (default $\alpha = 0.025$) with $\Sigma(N) - deg$ degrees of freedom and P_j the inverse Hessian matrix of the jth rule (C rules in sum), which is given by $P_j = (X^T Q_j X)^{-1}$, with $Q_j = diag(\Psi_j(\vec{x}(k))$ (Ψ_j the normalized membership degree to the jth rule) and X the regression matrix obtained during training phase. deg denotes the degrees of freedom in one local model, thus $p + 1$ with p the dimensionality of the input feature space (+1 for the intercept of the local hyper-plane); $\Sigma_j(N)$

the support of the jth rule over past samples as a measure of significance of the jth rule, thus $\Sigma_j(N) = \sum_{n=1}^{k} \Psi_j(\vec{x}_n)$. Clearly, if a sample $\vec{x}_k$ has a very low membership degree to the jth rule, the contribution in the sum in (14) is little (and should be little because it lies far away from the local linear model of this rule, hence it becomes unimportant).

5. Application Scenario and Experimental Setup

In our case study, we deal with the inspection of micro-fluidic chips used for sample preparation in DNA (deoxyribonucleic acid) sequencing. On the chip, the DNA and primers are packed into aqueous droplets in oil phase. Currently, they are checked in the diagnostic instrument by means of image inspection in a closed loop. This is done in an a posteriori manner, where bad chips are sorted out once they have already been produced (in order to not deliver bad parts to customers), based on machine learning classifiers as developed during a preliminary project, see [68] [69]. This, however, typically does not prevent unnecessary waste and can even induce greater complications and risks at the production system.

Therefore, the idea in this project (see Acknowledgements below) was to supervise the process data which are directly recorded at production time of chips and which may already reflect untypical occurrences when something goes ”out of the rudder”, much before it can be (manually) realized or even (automatically) predicted in the quality of the chips itself. An alternative possibility to circumvent such a delay causing unnecessary waste is to predict chip quality (defined through several measures) at an early point of time (or even early stage) — as pursued in our previous work in [7]. However, not for all quality measures reliable prediction models could be established with sufficiently high accuracy (defined by company experts), and sometimes anomalies may happen in the system which are not sufficiently affected in the (production-based) chip quality criteria, but still lead to unsatisfactory chip behavior (as happened in our case where customer complaints arose, see below).

5.1. Data Characteristics

In fact, our test case under empirical study contains process data which happened during the production of chips which passed the manual tests (i.e, which were in-line the limits of several important quality criteria such as flatness and void events, thus the prediction models could not detect any violations or anomalies), but latter the customer complained about the functionality of these chips (when using it in a bio-medical application)! This opened the question whether any anomalies in the process data could be realized, which occurred during production of those chips (chip orders) for which the customer complaints arose. Therefore, we traced back the corresponding chip order in our data base, which stores the collected data from the production processes over several years and extracted the data occurring during the production of this order. This data thus served as test data whether our approach is able to detect anomalies properly. Furthermore, we extracted process data occurring during the production of another chip order, which were explicitly labelled as OK by the customer without any complaints and which were produced with the *same machining parameters* as the chip order which were not OK (part of this OK data set was used as training

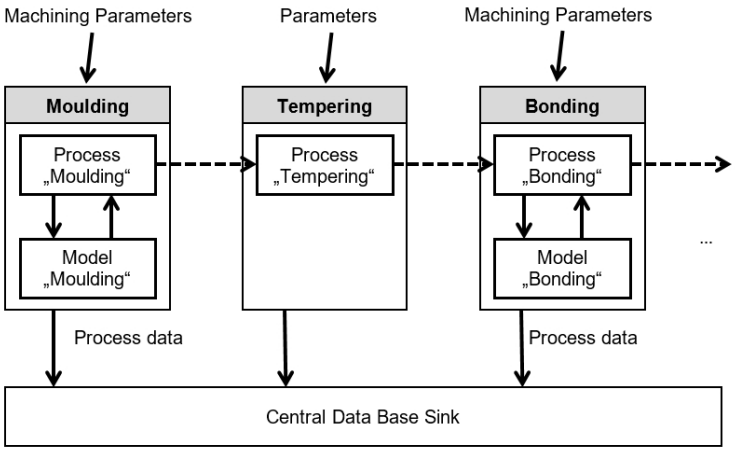

Figure 5: Multi-stage chip production process including injection molding and bonding as essential stages, which should be monitored and supervised to assure sufficient chip quality.

data set). In this way, we can actually check whether our approach is actually able to detect the anomaly during production (leading to functional problems of the chip) and not a possible intended drift/shift due to a change in the machining/production parameters. Furthermore, we also extracted another data set from a different production order, which was also confirmed by the customer as OK, but which was produced with a *different machining parameter* setting (for a different chip type). In this way, we can check how robust our approach is against changes in the parameters (ideally it should not detect any anomaly in this OK data set with just changed settings).

As the chip production system contains two essential production stages where the quality of the chips is basically determined, namely injection molding and bonding, where injection molding operates as the first, early stage and bonding as the last stage of production (before the chip is shipped to the customers), we extracted the appropriate data from both production stages. In this sense, we could realize how early our approach is able to detect (severe) anomalies: a detection of an anomaly already at the stage of the injection molding process would have a high impact on the practical usage of our approach, as the operator can intervene much earlier and thus prevent waste and save time for the latter production stages; clearly, a detection of an anomaly already at the stage of bonding liner would be also beneficial. Figure 5 shows the three production stages in our chip manufacturing system, where the intermediate stage (tempering) plays a minor role for quality control and is thus ignored in our empirical study (also, for this stage there are no process data recorded so far).

Table 1 summarizes the characteristics of all data sets which have been used for training and testing purposes of our approach (after deleting some rows with missing values).

Table 1: Characteristics of the applied data sets from the two different production stages injection molding (IE03) and bonding (BL02).

Data Set	# of Proc. Values (Dim)	# of Samples	Setting	Time Frame
Order **OK** 34374610, IE03	64	2974	AI	5 days in a row
Order **NOT-OK** 35026840, IE03	64	5360	AI	5 days in a row
Order **OK** 34374610, BL02	15	3881	AB	5 days in a row
Order **NOT-OK** 35026840, BL02	15	2440	AB	3 days in a row
3 Orders **OK**, BL02	15	6548	BB	2, 8 and 1 days in row

5.2. Experimental Setup for Training and Evaluation

For the OK data sets with standard parameter settings (AI and AB) — see Table 1 — we used the first half for establishing the causal relation networks and training the partial sub-models using generalized TS fuzzy systems as well as linear regression for performance comparison → 1487 samples in case of IE03 and 1940 samples in case of BL02. We used the second half as anomaly-free test data in order to check whether our approach is able to produce low false alarms on new on-line data samples (as all the data sets are timely ordered). This is in fact a real on-line simulation, as the (in time latter occurring) test data is loaded and processed sample-wise through the calculation of SPE statistics and I^2 statistics for dominant and non-dominant parts. Furthermore, we even used the OK data set with different parameter settings 'BB' for checking the stability of the various method(s) (to be compared with, see below) versus false alarm rates. We then also used the data sets which correspond to the NOT-OK orders in order to check how well our control limits are exceeded and thus how reliably our approach can detect the fault.

The most sensitive training parameter for fuzzy systems training procedure *Gen-Smart-EFS* [39], i.e. the factor in the statistical tolerance interval for deciding whether to evolve a new rule or not (thus steering the number of rules), is optimized by a grid-search technique coupled with 10-fold cross-validation procedure [57]: that factor value leading to the minimal cross-validation error (average error over the hold-out folds) is selected and a final TS fuzzy model is trained on the whole training data set, which also yields the help measures (inverse covariance matrix per rule etc.) for calculating the confidence bands as discussed in Section 4.3; this is repeated for each partial causal relation in the network found by the PC algorithm. Therefore, we also applied the fast PC algorithm to different combinations of folds within the CV-procedure to check its sensitivity on training data selection, and found out that we achieved pretty stable results, i.e. more or less the same causal relation dependence structure was obtained for all combinations and also in case when using the whole training data set.

For comparison purposes with related state-of-the-art works, we also report the results achieved with several well-known unsupervised anomaly detection methods, which do not identify causal relations or inter-dependencies between process values, but rather treat them as data sample vectors in a joint high-dimensional feature space. Almost all

of these can be seen as renowned methods having been heavily used for the purpose of anomaly/fault detection in preliminary publications. In particular, we applied the following methods:

- Cross-sample distance-based outlier detection: this approach is based on our own outlier detection methods, which we have originally used before in past projects for cleaning outliers in training data sets; its basic idea is to characterize an outlier/anomaly to have an untypical occurrence in the feature space in terms of a distance from the *main trend in the training data*. For representing the main trend, the average distance between each sample pair in the training data is calculated. Then, the cardinality of the set of distances lying above the main trend is calculated for each sample and thus samples with high cardinality are outlier candidates. A threshold is found according to the knee point in the sorted list of cardinalities. Hence, this approach is a simple unsupervised distance-based approach seeking for untypical distances of samples to other ones (and to the main trend of distances) in the high-dimensional sample space.

- Q-statistics and Hotelling (T^2) statistics based on principal components analysis (PCA) [70]: the classical way to characterize outliers based on principal components which represent the directions in the multi-dimensional feature space which best explain the (most significant) variances in the (training) data. The Q-statistics represents the distance from a (new) full-dimensional data sample to the principal components projection subspace, whereas the Hotelling statistics represents the distance of a (new) data sample inside the projection space to the center of the ellipsoid characterized by the covariance of the data — both have been used in several fault and anomaly detection approaches, see, e.g., [71], [28]; automatized significance thresholds (above which an anomaly is likely) can be extracted from the data for both statistics [72]. The number of principal components used has to be parameterized, but a good strategy is to extract as many principal components as necessary to explain 95% to 99% of the variance contained in the (training) data (thus, little parametrization effort is required).

- Q-statistics and Hotelling (T^2) statistics based on probabilistic principal components analysis (PPCA) [73]: principal components are extracted locally to achieve a mixture of them and thus to cope with non-linearities in the data (especially with different local variance behavior of the data). Thereby, starting from a probabilistic model, (the mixture of) PCA projections are derived within a maximum-likelihood (ML) procedure. Q- and T^2 statistics are then calculated in the same way as in case of classical PCA by using the extracted (mixture of) principal components (i.e., the nearest local PCA space for each sample) — again, little parametrization effort is required for choosing the number of principal components (see above).

- Q-statistics and Hotelling (T^2) statistics based on kernel PCA [74]: thereby, the trick is to project the data into a high-dimensional kernel space by applying a Mercer kernel between each (training) sample pair in order to 'linearize' the (non-linear) data space and then to apply standard linear PCA on the kernel matrix. Mean centering the data (always required for conducting PCA correctly) in the kernel space is not straightforward

(because the actual feature space due to projection ϕ is not given [34]) and thus desires a pseudo mean centering operation, which we have used based on the derivations in [75]. After the $N \times N$ kernel matrix K is spanned (with N the number of training samples), principal components directions are extracted through using the eigenvector decomposition of the kernel matrix (as done in classical PCA). Q- and T^2 statistics are then calculated in the same way as in the case of classical PCA by using the extracted principal components — again, little parametrization effort is required for choosing the number of principal components (see above); additional parametrization effort is induced when trying out different parameter values, i.e. different widths of the Gaussian kernel, which we used per default as included in MATLAB's kernelPCA_tutorial implementation.

- One-class support vector machines (OC-SVMs) [76]: it estimates the support of arbitrary sample distributions in high-dimensional feature spaces. The main idea behind OC-SVMs is that it is intended to create a kind of hull around the data of a single labeled class. In this sense, one-class SVMs can be exploited to characterize the fault-free nominal situation, based on the (fault-free) training data (the same data which is used for training our regression-based CRNs). Thus, one-class SVMs have been widely (and successfully) used for anomaly and fault detection, e.g., see [29] [31] [30]. We applied three commonly used kernels for conducting the same kernel trick as in the case of PCA (see above), namely sigmoid kernel, Gaussian kernel and linear kernel and report separate results among these. Thereby a kernel parameter (width in case of Gaussian and sigmoid kernels) can be tuned additionally to the most sensitive ν-parameter, which is a re-parametrization of the cost parameter C, which in turn is a regularization parameter (to improve stability in case of noise etc.) [77]. We used the lib-SVM implementation [78] and had to test various parameter combinations (ν and widths over ranges as suggested in [79]) to achieve useful results at all, see also results below.

- Support vector data description (SVDD) [80]: it differs from one-Class SVMs in terms of directly finding the smallest hyper-sphere that contains all (training samples) samples, except for some (more) outlying samples (which is achieved by introducing slack variables), whereas one-class SVMs finds a minimal enclosing sphere for (training) samples by finding a maximal margin hyperplane between the samples and the origin [81]. It has been thus also successfully used for anomaly and fault detection, e.g., see [32] [82]. Again, as in case of one-Class SVM, we used lib-SVM implementation (SVDD extension — as parameterizable by '-s 5' in version 3.22), applied three kernels (linear, sigmoid, Gaussian) and had to tune various combinations of the most important parameters ν and kernel width to achieve reliable results.

- Fault detection using k-nearest neighbor rule (FD-kNN) [26]: it uses the k-nearest neighbors of a (training) sample to estimate a 'regular' distance statistics D, which turns out that it approximately follows a non-central chi-square distribution. Based on this knowledge, a control limit can be derived automatically from the training data (as in the case of our approach), which is then applied to the test data, where for each test data sample the

k-nearest neighbors from the fixed training set are elicited. Parameter k needs to be tuned adequately in our use case (ranging from 1 to $\sqrt{N}$) to achieve reliable results, see below.

- Fault detection using weighted k-nearest neighbor rule (FD-wkNN) [27]: it is the same as FD-kNN but includes sample weights when calculating the test statistics D — these are elicited by the reciprocal squared distances to k-nearest neighbors of each of the k-nearest neighbors of one data sample. Thus, it performs a double k-nearest neighors approach with weights from the second k-NN stage. Thus, it is expected to take better into account local differences of different modes [27].

Furthermore, we compare the performance with the related residual-based approach published in [22], where only univariate residual analysis is conducted independently (for each signal separately) based on tolerance bands assuming Gaussian distributed residuals and for which a multiplication factor for the width of a tolerance band has to be hand-tuned (which is not needed in the approach proposed in this paper). Several data plots of residual signals and control limits will be shown to compare our own preliminary method with the new one proposed in this paper in order to clearly show what the new on really brings in terms of stability and detection capability (in our previous approach, the control limits are statistically motivated tolerance bands). The comparisons with all methods mentioned above will be also made based on the percentage of false alarms in the OK data sets and the detection capabilities in the NOT-OK data sets, especially i) whether an approach is able detect the anomaly or not (Yes/No) and ii) if so, the delay of detection, i.e. how many samples need to be supervised such that control limits are exceeded; this is very important for an early recognition of anomalies during the real on-line case.

6. Results

6.1. Causal Relation Networks

Figure 6 visualizes the causal relation network obtained by the PC algorithm for the bonding liner data (BL02), whereas only partial causal relations are shown for which regression fits with (cross-validated) qualities of higher than 0.3 could be achieved. Thereby, the red nodes represent those process values which appear as effects (targets) in any of these, whereas white nodes represent process values which only appear as causes (inputs) (however, some effects may also appear as causes for other effects). The arrows between the nodes represent the cause-effect direction: starting point of each arrow always corresponds to a cause, an end point always to an effect.

In functional form, we are thus ending up with the following relations whose adequate structures (and associated non-linearity degree) and parameters are estimated by fuzzy regression fits:

- $ABT = f(ACT, BUT, AMT)$ with model qual 0.63
- $ACT = f(ABT)$ with model qual 0.50

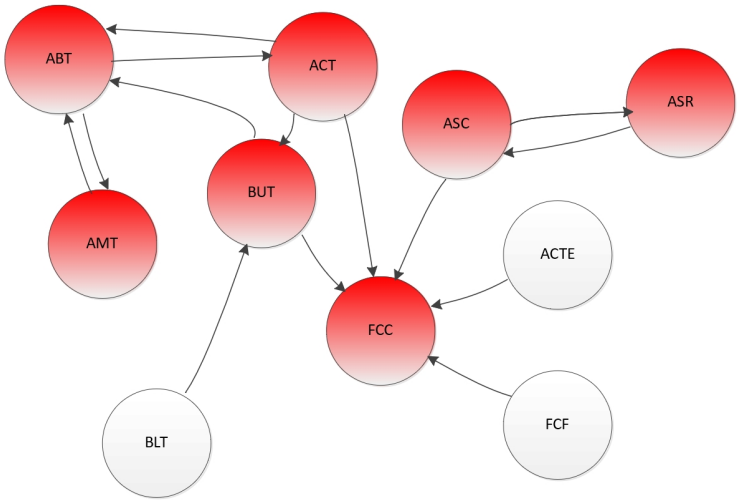

Figure 6: Causal relation network achieved by PC algorithm showing only cause-effect structures due to a high model quality from regression fits.

- $AMT = f(ABT)$ with model qual 0.90
- $ASC = f(ASR)$ with model qual 1.00
- $ASR = f(ASC)$ with model qual 1.00
- $BUT = f(ACT, BLT)$ with model qual 0.38
- $FCC = f(ACTE, FCF, ASC, BUT, ACT)$ with model qual 0.49

In case of linear regression fits, slightly worse model qualities could be obtained, pointing to a quasi-linear problem. The process values are named by three-letter abbreviations, whose full names we are not allowed to report due to security reasons.

For the injection molding data, we ended up with 31 causal relations achieving a regression model quality above 0.3; due to space reasons, these are not listed here, neither a directed graph produced, but they can be provided on demand in an appendix or as supplementary material In both cases, the causal relation structures helped the experts of the company to gain some valuable additional knowledge and insight into the process, which they were not aware of so far.

6.2. Residual Analysis for OK and NOT-OK Data

6.2.1. For BL02 (bonding liner)

Figures 7 a and 7 b show three completely different residual signals obtained for three partial causal relations at BL02, whose effects are represented by ASC, BUT and FCC. Thereby, in all figures the black color denotes the residual signals obtained on the training data, the green color the signal obtained on the test data of the OK order (with the same machining parameter setting) and the red color the signal obtained on the test data of the NOT-OK order. Additionally, also the static and dynamic tolerance bands are shown in dashed and dotted lines, which are obtained by a native univariate residual analysis, see [22].

It becomes immediately clear that for the ASC model (upper image in Figure 7 a) the residual continues in a similar manner over the complete time frame (for all three data sets), thus it cannot contribute to the detection of the anomaly contained in the NOT-OK data; on the other hand, neither it triggers any false alarms. This is different for the BUT model, where the residual signal shows several distinct peaks exceeding the tolerance bands several times significantly (lower image in Figure 7 a). In this case, a standard univariate approach independently inspecting each signal (as, e.g., in [22]) would have produced several false alarms. For the FCC model, the peaks are less intense during the green OK data set (Figure 7 b), whereas a clear shift can be observed in the NOT-OK phase, but exceeding the tolerance band only marginally and partially.

The situation regarding false alarms and detectability of the anomaly seems to improve much when inspecting SPE and I^2 statistics (as multi-variate measures over all residuals), visualized in Figure 8 a. Still, when applying the SPE statistics (upper image) there occurred a distinct peak exceeding the threshold (shown as dashed horizontal line) significantly at the end of the anomaly-free test data, however, most of the intermediate false alarms (as produced by the BUT model) are filtered away — thus, it improved false alarm rates, compared to univariate residual analysis using ordinary tolerance bands (see Figure 7) and OR connections over all residual signals (as done in [22]). The lower image shows the I^2 statistics on the dominant parts and indicates that it in fact acted as a kind of noise filter, as the two extreme peaks (producing false alarms) were completely gone. This is also an empirical confirmation of our discussion in Section 4.2 about the noise suppressing capabilities when only using the dominant parts. Furthermore, during the NOT-OK data set several consecutive samples (surrounded by ellipsoids) exceeded the control limit, thus the anomaly could be detected as significant with I^2 statistics. In case when applying this during real on-line sample-wise processing mode, the delay of anomaly detection turned out negligible due to the first significance exceed of the control limit after a few samples (compare also with Table 2). Figure 8 b shows the I^2 statistics on the non-dominant parts from the demixing matrix and signal reconstruction: as expected due to the discussion in Section 4.2, the noise impacts the statistics significantly (much more than that one extracted from the dominant parts) such that several false alarms are triggered during the OK data (green).

6.2.2. For IE03 (injection molding)

Figure 9 shows two different residual signals obtained for three partial causal relations at IE03, whose effects are represented by ENCSE and PLHWZ2. The color codings and the additional dashed and dotted lines (for the tolerance bands) mean the same as in the case of BL02. It becomes immediately clear that for both residual signals significant exceeds of the tolerance bands already during the OK phase (green) occurred. Thus, many false alarms can be expected when applying a univariate approach with classical tolerance bands from statistical process control (as, e.g., in [22]). In the first case (ENCSE model), the peaks even continue during the NOT-OK phase with the same amount and intensity level as during the OK phase; thus, no real distinction between the two phases can be made. In the second case (PLHWZ2) there is a clear shift of the residuals, which could be detectable, but the problem during the OK phase with many peaks over the tolerance band remains. Similar to the BL02 results, the situation regarding false alarms and detectability of the anomaly seem to improve much when inspecting SPE and I^2 statistics (as multivariate measures over all residuals), visualized in Figure 10. Still, when applying the SPE statistics (upper image) there occurred a distinct peak exceeding the threshold (shown as dashed horizontal line) significantly at the end of the anomaly-free test data, however, most of the intermediate false alarms are filtered away — thus, it improved false alarm rates. The lower image shows the I^2 statistics on the dominant parts and indicates that it in fact acted as a kind of noise filter, as false alarms and the two extreme peaks were completely gone. This is also again an empirical confirmation of our discussion in Section 4.2 about the noise suppressing capabilities when only using the dominant parts. Furthermore, during the NOT-OK data set nearly all! samples exceeded the control limit, which is a very strong indicator that something went pretty wrong. Compared to the BL02 results, the exceed of the control limits is much more intense in terms of number of samples and in terms of amplitude of I^2 statistics. This is even better for the "business", because injection molding is conducted typically days or even weeks before the bonding process of the same orders, thus a very early detection and furthermore a longer reaction time and a higher waste reduction can be achieved. As in case of BL02, the I^2 statistics on the non-dominant parts has similar problems as the SPE statistics with noise (peaks) during the OK phase; furthermore, it could not detect the anomaly at all as staying way below the control limits during the NOT-OK phase.

6.3. Residual Analysis for Additional OK Data with a Different Machining Parameter Setting

For the bonding process, three additional data sets were extracted lasting over 2, 8 and 1 separate days for orders showing no customer complaints at all, but with different machining parameter settings. Hence, we exchanged the NOT-OK data set shown in the preliminary result plots with this OK data sets (thus appearing as in red color at the right end of the signals) and left the remaining data (original training data and OK test data) unchanged. In this way, we could establish a one-to-one comparison with the results achieved between NOT-OK data and OK data with different parameter settings (ideally the latter should not trigger any alarms, while the former did as verified in the previous section). Figure 11 compares the SPE and I^2 statistics on dominant parts (upper and lower image). Obviously, the

I^2 statistics produced much lower false alarms (4 only in sum over a period of 6548 samples = 0.06%) than the SPE statistics (3.26%), which makes it more attractive to engineers and appliers who do not want to get wrongly disturbed too much by false warnings during an on-line production phase. Also, the 4 distinct peaks could be somehow ignored (when not confirmed with more consecutive samples), which is not possible in case of SPE (due to phases with more consecutive, nearby lying samples appearing over the control limit). Interestingly, the I^2 statistics extracted from the non-dominant parts looks very similar to the SPE statistics. Thus, both are effected by the noise in the data (or changes parameters) in a similar way to produce significant false alarms.

Concrete values regarding anomaly detection capability, delay of detection as well as false alarm rates are reported in the next section when comparing the performance of the control limits from our approach with those from several well-known related works in anomaly detection.

6.4. Comparison with Results Achieved by Related Anomaly Detection Methods

Table 2 shows the comparison of the performance of the various related SoA methods as used for evaluation purpose and which have been discussed in Section 5.2. Thereby, for both production stages, bonding liner and injection molding, three measures are presented:

- Detection capability: its value is either 'Yes' or 'No' to indicate whether the anomaly in the order which were not OK for the customer can be could have been detected with the method or not.
- Delay of detection: this value indicates the delay of the detection if the method would have been used in an on-line learning context, i.e. after how many samples, once the production problem started, the method was able to detect the anomaly. The unit is in turns of the number of samples, where in case of injection molding the sampling frequency is approximately 20 seconds and in case of bonding liner 50 seconds.
- False alarm (FA) rate: this value indicates the percentual amount of samples which were detected as anormal, but in fact were samples recorded from the OK phases. The percentage is made possible as the total number of samples recorded during OK production is known in the data sets.

Additionally, in the last column the parametrization effort of the methods is reported in terms of 'Low', 'Medium' and 'High, or even 'None', if there is no parameter to be tuned for the on-line anomaly detection phase. The latter is the case for our approach, as the full tuning takes place during the off-line model building process (within an automatized CV- and model selection procedure). During the on-line test phase, our approach just amalgamates the multi-dimensional residual information to statistical measures such as SPE and I^2 statistics from the independence component analysis and applies a threshold on these, which is automatically determined through KDE and CDF, see Section 4.1.

The latter technique has been also applied to the PCA-based variants and to the two FD-kNN methods, but for the PCA a possibility still is to define the dimensionality of the reduced sub-space (typically $p << m$), which can

Table 2: Performance comparison of our control limits with several related unsupervised SoA methods in anomaly detection.

Methods	**Bonding Liner**			**Injection Molding**			**Parametrization**
	Det. Capability	**Delay of Det.**	**FA Rate (%)**	**Det. Capability**	**Delay of Det.**	**FA Rate (%)**	**Effort**
Outlier Card.	No	∞	4.63 (8.98)	Yes	1	1.28	None
PCA Q-Stat	No	∞	0 (0)	Yes	25	1.78	None-Low
PCA T^2	No	∞	0 (0)	Yes	159	1.14	None-Low
PPCA Q-Stat	No	∞	3.81 (9.72)	No	∞	1.28	None-Low
PPCA T^2	No	∞	0 (0)	Yes	25	1.64	None-Low
kernel PCA Q-Stat	No	∞	3.44 (4.36)	Yes	1	1.7	None-Medium
kernel PCA T^2	No	∞	3.18 (4.36)	Yes	1	1.7	None-Medium
one-Class SVMs Linear	No	∞	2.76 (1.33)	Yes	1	9.58	High
one-Class SVMs Gaussian	No	∞	23.38 (19.45)	Yes	1	47.76	High
one-Class SVMs Sigmoid	No	∞	2.55 (1.75)	Yes	1	100	High
SVDD Linear	No	∞	2.78 (2.60)	Yes	1	1.14	High
SVDD Gaussian	No	∞	8.62 (11.16)	Yes	1	48.97	High
SVDD SVMs Sigmoid	**Yes**	**1**	1.51 (1.0)	Yes	1	1.0	High
FD-kNN	No	∞	4.52 (9.46)	Yes	1	5.18	Medium
FD-wkNN	No	∞	4.57 (9.72)	Yes	1	3.34	Medium
Uni-variate Res. Analysis as in [23]	Yes	1	4.12 (10.41)	Yes	1	0.92	Medium
SPE from CRNS (ours)	**Yes**	**2**	3.26 (9.56)	Yes	11	1.56	**None**
I^2 from ICA with CRNs (ours)	**Yes**	3	**0.05 (0)**	Yes	1	**0.43**	**None**

be done through accumulated explained variance of the data, but still a threshold is needed there (typically between 0.95 and 0.99) — however, we have reported the results on the full spanned PCA space (with marginal difference to the performance when reducing space), thus in sum we see the parametrization effort as 'None-Low'. In case of kernel PCA, someone could tune the width of the Gaussian kernel, which we used to be set 1 per default (as also was contained in the original MATLAB script). For the FD-kNN methods, k remains an essential parameter, which we optimized during several runs to obtain best performance. For one-Class SVMs and SVDD, the parametrization effort was pretty high to achieve any reliable results at all, because basically two parameters, namely ν and γ defining the widths and shapes of the Gaussian and sigmoid kernels, had to be optimized in larger grids; as their default values in the lib-SVM implementation lead to unusable results.

From Table 2, it becomes clear that our new approach could outperform various related (unsupervised) SoA methods for anomaly detection as well as the previous uni-variate residual-based method in [22] [23], in terms of detection capability as well as false alarm rates. This is especially true for the bonding liner data set, where almost all methods (except support vector data description) failed to detect the anomal behavior at all (thus, the delay of detection in the third column is set to a value of ∞). In case of injection molding (right part of the table), most methods can in fact detect the anomaly, but their false alarm rates (last but one column) are significantly higher than our variant using I^2 statistics from the ICA with CRNs (last line in the table). These higher false alarm rates are also the case for the

bonding liner, except PCA with either Q-statistics or T^2-statistics which produces a rate of 0 (compared to a rate of 0.05 when using I^2 statistics from ICA) — however, these methods cannot detect the anomaly (reflected in thousands of samples) at all. The only exception in the case of bonding liner is support vector data description, which can detect the anomaly with the slightly lower delay (1 vs. 3 samples), however, has a higher false alarm rate (1.51% vs. 0.05%) and especially a much higher parametrization effort: in fact, it required high tuning efforts for two most sensitive parameters within extensive trial-and-error runs in order to achieve reliable results at all (the same was the case of one-Class SVMs) — this is usually not (realistically) possible when being installed in an on-line system as a kind of plug-and-play method.

The same is true for the previous (partial) residual-based method in [23], where a significant hand-tuning of the multiplication factor n for the width of the tolerance bands had to be conducted to achieve useful results — optimal values in terms of the best tradeoff between detection delay and false alarm rate were 6 for bonding liner and 10 for injection molding, thus not even a direct transfer of this parameter from one scenario to the other could hold. But even with these optimally tuned values, the performance was still weaker (especially for the bonding liner) than using I^2 statistics (on the dominant parts of the demixing matrix) from ICA.

7. Conclusion and Outlook

We proposed a new approach for anomaly detection which is based on a combination of causal relation networks establishment through partial causal influence models employing a non-linear model architecture and an advanced analysis of multi-variate residual signals through independent component analysis and the usage of non-dominant parts of the demixing matrix in order to automatically suppress noise and to achieve a robust control measure whose ideal limits are automatically determined through KDE and CDF. The approach is successfully evaluated based on high-dimensional process data from micro-fluidic chip production, where a severe anomaly occurred during the on-line production process (leading to customer complaints). It could be detected with almost no delay and without producing significant false alarms on anomaly-free production phases where even different machining parameter settings were used. Widely used and renowned related works in anomaly detection showed significantly weaker performance than our proposed approach in terms of detection capability and false alarm rates, and this even with significant parametrization efforts in various trial and error test runs (whereas our approach did not require any).

Finally, we want to point out that our method has the realistic potential to be also elegantly used for *anomaly localization*, where the major goal lies in the identification of process values (channels) which are most likely affected by the anomaly, which furthermore can be used as indicator where the anomaly most likely lies (important for reason finding). This is because it offers a modular structure including several partial relations (in the CRN) and induced sub-spaces, which are typically not disjoint and thus can be exploited to check for redundant, overlapping causes and effects and their influence (weights) in those partial relations that lead to violations of the control limits – probably in

a similar manner as having been successfully conducted in [37] for a collection of SysID models — which thus will be one of our future research directions and evaluations. Such an anomaly localization may be difficult to realize with the use of the unsupervised SoA methods having been compared in Table 2, as they operate on a single model in the full space, where all input variables (channels) contribute with the same weight (so, no variable influence weights in particular ('violated') sub-spaces can be amalgamated, compared, processed etc.).

Acknowledgements

The authors acknowledge the Austrian research funding association (FFG) within the scope of the 'IKT of the future' programme, project 'Generating process feedback from heterogeneous data sources in quality control' (contract # 849962). The first and second authors also acknowledge the support by the "LCM — K2 Center for Symbiotic Mechatronics" within the framework of the Austrian COMET-K2 program.

References

[1] O. Myklebust, Zero defect manufacturing: A product and plant oriented lifecycle approach, Procedia CIRP 12 (2013) 246–251.

[2] J. Levitt, Complete Guide to Preventive and Predictive Maintenance, Industrial Press Inc., New York, 2011.

[3] V. Hodge, J. Austin, A survey of outlier detection methodologies, Artificial Intelligence Review 22 (2) (2004) 85–126.

[4] V. Chandola, A. Banerjee, V. Kumar, Anomaly detection: A survey, ACM Computing Surveys 41 (3).

[5] S. Aumi, B. Corbett, P. Mhaskary, Model predictive quality control of batch processes, in: 2012 American Control Conference, Fairmont Queen Elizabeth, Montral, Canada, 2012, pp. 5646–5651.

[6] R. Mobley, An Introduction to Predictive Maintenance — Second Edition, Elsevier Science, Woburn, Massachussetts, U.S.A., 2002.

[7] E. Lughofer, A.-C. Zavoianu, R. Pollak, M. Pratama, P. Meyer-Heye, H. Zörrer, C. Eitzinger, J. Haim, T. Radauer, Self-adaptive evolving forecast models with incremental PLS space updating for on-line prediction of micro-fluidic chip quality, Engineering Applications of Artificial Intelligence 68 (2018) 131–151.

[8] Z. J. Viharos, J. Csanaki, J. Nacsa, M. Edelenyi, C. Pentek, K. B. Kis, A. Fodor, J. Csempesz, Production trend identification and forecast for shop-floor business intelligence, Acta Imeko 5 (4).

[9] E. Lughofer, C. Eitzinger, C. Guardiola, On-line quality control with flexible evolving fuzzy systems, in: M. Sayed-Mouchaweh, E. Lughofer (Eds.), Learning in Non-Stationary Environments: Methods and Applications, Springer, New York, 2012, pp. 375–406.

[10] M. Pratama, E. Dimla, T. Tjahjowidodo, E. Lughofer, W. Pedrycz, Online tool condition monitoring based on parsimonious ensemble+, IEEE Transactions on Cybernetics on-line and in press (doi: 10.1109/TCYB.2018.2871120).

[11] M. Kano, Y. Nakagawa, Data-based process monitoring, process control, and quality improvement: Recent developments and applications in steel industry, Computers and Chemical Engineering 32 (2008) 12–24.

[12] M. Moshtaghi, J. Bezdek, C. Leckie, S. Karunaseker, M. Palaniswami, Evolving fuzzy rules for anomaly detection in data streams, IEEE Transactions on Fuzzy Systems 23 (3) (2015) 688–700.

[13] M. Chen, A hybrid ANFIS model for business failure prediction utilizing particle swarm optimization and subtractive clustering, Information Sciences 220 (2013) 180–195.

[14] D. Acuna, M. Orchard, R. Saona, Conditional predictive bayesian cramer-rao lower bounds for prognostic algorithms design, Applied Soft Computing https://doi.org/10.1016/j.asoc.2018.01.033.

[15] W. Liao, Y. Wang, Data-driven machinery prognostics approach using in a predictive maintenance model, Journal of Computers 8 (1) (2013) 225–231.

[16] S. Ekwaro-Osire, A. Gonalves, F. Alemayehu, Probabilistic Prognostics and Health Management of Energy Systems, Springer, New York, 2017.

[17] E. Lughofer, On-line active learning: A new paradigm to improve practical useability of data stream modeling methods, Information Sciences 415–416 (2017) 356–376.

[18] E. Lughofer, R. Richter, U. Neissl, W. Heidl, C. Eitzinger, T. Radauer, Explaining classifier decisions linguistically for stimulating and improving operators labeling behavior, Information Sciences 420 (2017) 16–36.

[19] D. Liu, Y. Zhang, Z. Yu, M. Zeng, Incremental supervised locally linear embedding for machinery fault diagnosis, Engeineering Applications of Artifical Intelligence 50 (C) (2016) 60–70.

[20] D. Dou, S. Zhou, Comparison of four direct classification methods for intelligent fault diagnosis of rotating machinery, Applied Soft Computing 46 (2016) 459–468.

[21] M. Orchard, D. Acuna, On prognostic algorithm design and fundamental precision limits in long-term prediction, in: E. Lughofer, M. Sayed-Mouchaweh (Eds.), Predictive Maintenance in Dynamic Systems, Springer, New York, 2018, p. to appear.

[22] F. Serdio, E. Lughofer, A.-C. Zavoianu, K. Pichler, M. Pichler, T. Buchegger, H. Efendic, Improved fault detection employing hybrid memetic fuzzy modeling and adaptive filters, Applied Soft Computing 51 (2017) 60–82.

[23] F. Serdio, E. Lughofer, K. Pichler, T. Buchegger, H. Efendic, Residual-based fault detection using soft computing techniques for condition monitoring at rolling mills, Information Sciences 259 (2014) 304–320.

[24] L. Mendona, J. Sousa, J. S. da Costa, An architecture for fault detection and isolation based on fuzzy methods, Expert Systems with Applications 36 (2) (2009) 1092–1104.

[25] L. Wang, R. Gao, Condition Monitoring and Control for Intelligent Manufacturing, Springer Verlag, London, UK, 2006.

[26] Q. He, J. Wang, Fault detection using the k-nearest neighbor rule for semiconductor manufacturing processes, IEEE Transactions on Semiconductor Manufacturing 20 (4) (2007) 345–354.

[27] C. Zhang, X. Gao, Y. Li, L. Feng, Fault detection strategy based on weighted distance of k nearest neighbors for semiconductor manufacturing processes, IEEE Transactions on Seminconductor Manufacturing on-line and in press (DOI 10.1109/TSM.2018.2857818).

[28] P. Odgaard, B. Lin, S. Jorgensen, Observer and data-driven-model-based fault detection in power plant coal mills, IEEE Transactions on Energy Conversion 23 (2) (2008) 659–668.

[29] H. Chen, P. Tino, X. Yao, A. Rodan, Learning in the model space for fault diagnosis, IEEE Transactions on Neural Networks and Learning Systems 25 (1) (2014) 124–136.

[30] D. Fernndez-Francos, D. Martnez-Rego, O. Fontenla-Romero, A. Alonso-Betanzos, Automatic bearing fault diagnosis based on one-class v-svm, Computers & Industrial Engineering 64 (1) (2013) 357–365.

[31] S. Mahadevan, S. Shah, Fault detection and diagnosis in process data using one-class support vector machines, Journal of Process Control 19 (10) (2009) 1627–1639.

[32] L. Duan, M. Xie, T. Bai, J. Wang, A new support vector data description method for machinery fault diagnosis with unbalanced datasets, Expert Systems with Applications 64 (2016) 239–246.

[33] T. Nakamura, A. Lemos, A batch-incremental process fault detection and diagnosis using mixtures of probablistic pca, in: Proceedings of the Evolving and Adaptive Intelligent Systems (EAIS) Conference 2014, IEEE press, Linz, Austria, 2014.

[34] R. Samuel, Y. Cao, Nonlinear process fault detection and identification using kernel pca and kernel density estimation, Systems Science & Control Engineering 4 (1) (2016) 165–174.

[35] J. Yu, J. Jang, J. Yoo, J. Park, S. Kim, Bagged auto-associative kernel regression-based fault detection and identification approach from steam boilers in thermal power plants, Journal of Electrical Engineering & Technology 12 (4) (2017) 1406–1416.

[36] P. Angelov, V. Giglio, C. Guardiola, E. Lughofer, J. Luján, An approach to model-based fault detection in industrial measurement systems with application to engine test benches, Measurement Science and Technology 17 (7) (2006) 1809–1818.

[37] F. Serdio, E. Lughofer, K. Pichler, M. Pichler, T. Buchegger, H. Efendic, Fuzzy fault isolation using gradient information and quality criteria from system identification models, Information Sciences 316 (2015) 18–39.

[38] P. Spirtes, C. Glymour, R. Scheines, Causation, Prediction, and Search, 2nd Edition, MIT Press, Cambridge, 2000.

[39] E. Lughofer, C. Cernuda, S. Kindermann, M. Pratama, Generalized smart evolving fuzzy systems, Evolving Systems 6 (4) (2015) 269–292.

[40] T. Takagi, M. Sugeno, Fuzzy identification of systems and its applications to modeling and control, IEEE Transactions on Systems, Man and Cybernetics 15 (1) (1985) 116–132.

[41] E. Lughofer, S. Kindermann, SparseFIS: Data-driven learning of fuzzy systems with sparsity constraints, IEEE Transactions on Fuzzy Systems 18 (2) (2010) 396–411.

[42] L. Cohen, G. Avrahami-Bakish, M. Last, A. Kandel, O. Kipersztok, Real-time data mining of non-stationary data streams from sensor networks, Information Fusion 9 (3) (2008) 344–353.

[43] W. Dargie, C. Poellabauer, Fundamentals of wireless sensor networks: theory and practice, John Wiley and Sons, Hoboken, New Jersey, 2010.

[44] B. Khaleghi, A. Khamis, F. O. Karray, S. N. Razavi, Multisensor data fusion: A review of the state-of-the-art, Information Fusion 14 (1) (2013) 28–44.

[45] S. Agarwal, D. Starobinski, A. Trachtenberg, On the scalability of data synchronization protocols for PDAs and mobile devices, IEEE Network 16 (4) (2002) 22–28.

[46] P. Boltryk, C. J. Harris, N. M. White, Intelligent sensors - a generic software approach, Journal of Physics, Conference Series (15) (2005) 155–160.

[47] L. Fortuna, S. Graziani, A. Rizzo, M. Xibilia, Soft Sensor for Monitoring and Control of Industrial Processes, Springer Verlag, London, 2007.

[48] F. V. Jensen, Bayesian Networks and Decision Graphs, Springer Verlag, New York, 2001.

[49] T. D. Le, T. Hoang, J. Li, L. Liu, H. Liu, S. Hu, A fast PC algorithm for high dimensional causal discovery with multi-core PCs, Journal of LaTeX Class Files 13 (9) (2014) 1–13.

[50] A. Lemos, W. Caminhas, F. Gomide, Multivariable gaussian evolving fuzzy modeling system, IEEE Transactions on Fuzzy Systems 19 (1) (2011) 91–104.

[51] E. Lughofer, J.-L. Bouchot, A. Shaker, On-line elimination of local redundancies in evolving fuzzy systems, Evolving Systems 2 (3) (2011) 165–187.

[52] E. Lughofer, M. Pratama, I. Skrjanc, Incremental rule splitting in generalized evolving fuzzy systems for autonomous drift compensation, IEEE Transactions on Fuzzy Systems 26 (4) (2018) 1854–1865.

[53] Y. Xu, S. Furao, O. Hasegawa, J. Zhao, An online incremental learning vector quantization, in: T. Theeramunkong (Ed.), Proceedings of the PAKDD 2009 Conference, Vol. 5476 of Lecture Notes in Artificial Intelligence, Springer, Berlin Heidelberg, 2009, pp. 1046–1053.

[54] H. Zou, T. Hastie, Regularization and variable selection via the elastic net, Journal of the Royal Statistical Society, Series B (2005) 301–320.

[55] T. Hastie, R. Tibshirani, J. Friedman, Regularized paths for generalized linear models via coordinate descent, Journal of Statistical Software 33 (1).

[56] E. Lughofer, Evolving Fuzzy Systems — Methodologies, Advanced Concepts and Applications, Springer, Berlin Heidelberg, 2011.

[57] M. Stone, Cross-validatory choice and assessment of statistical predictions, Journal of the Royal Statistical Society 36 (1) (1974) 111–147.

[58] C. Zongwu, T. Chih-ling, W. Xizhi, The examination of residual plots, Statistica Sinica 8 (1998) 445–465.

[59] T. Fawcett, Roc graphs: Notes and practical considerations for data mining researchers, Tech. rep., HP Laboratories, 1501 Page Mill Road Palo Alto, CA 94304 (2003).

[60] L. Chiang, E. Russell, R. Braatz, Fault Detection and Diagnosis in Industrial Systems, Springer, London Berlin Heidelberg, 2001.

[61] B. W. Silverman, Density Estimation for Statistics and Data Analysis, CRC Press, Boca Raton, 1986.

[62] M. P. Wand, M. C. Jones, Kernel Smoothing, CRC Press, Boca Raton, 1994.

[63] J. M. Lee, C. Yoo, I. B. Lee, Statistical process monitoring with independent component analysis, Journal of Process Control 14 (5) (2004) 467–485.

[64] P. Comon, Independent component analysis: a new concept?, Signal Processing 36 (3) (1994) 287–314.

[65] A. Hyvärinen, E. Oja, Independent component analysis: algorithms and applications, Neural Networks 13 (4–5) (2000) 411–430.

[66] I. Skrjanc, Confidence interval of fuzzy models: An example using a waste-water treatment plant, Chemometrics and Intelligent Laboratory Systems 96 (2009) 182–187.

[67] E. Lughofer, M. Pratama, On-line active learning in data stream regression using uncertainty sampling based on evolving generalized fuzzy models, IEEE Transactions on Fuzzy Systems 26 (1) (2018) 292–309.

[68] E. Lughofer, E. Weigl, W. Heidl, C. Eitzinger, T. Radauer, Integrating new classes on the fly in evolving fuzzy classifier designs and its application in visual inspection, Applied Soft Computing 35 (2015) 558–582.

[69] E. Weigl, W. Heidl, E. Lughofer, C. Eitzinger, T. Radauer, On improving performance of surface inspection systems by on-line active learning and flexible classifier updates, Machine Vision and Applications 27 (1) (2016) 103–127.

[70] I. Jolliffe, Principal Component Analysis, Springer Verlag, Berlin Heidelberg New York, 2002.

[71] D. Garcia-Alvarez, Fault detection using principal component analysis (pca) in a wastewater treatment plant (wwtp), in: Proceedings of the 62-th Int. Student's Scientific Conference, Saint-Peterburg, Russia, 2009, pp. 13–17.

[72] C. Cernuda, E. Lughofer, G. Mayr, T. Röder, P. Hintenaus, W. Märzinger, J. Kasberger, Incremental and decremental active learning for optimized self-adaptive calibration in viscose production, Chemometrics and Intelligent Laboratory Systems 138 (2014) 14–29.

[73] M. Tipping, C. Bishop, Probabilistic principal component analysis, Journal of the Royal Statistical Society: Series B (Statistical Methodology) 61 (3) (1999) 611–622.

[74] B. Schölkopf, Nonlinear component analysis as a kernel eigenvalue problem, Neural Computation 10 (1998) 1299–1319.

[75] M. Welling, Kernel principal components analysis, Small TutorialHttps://www.ics.uci.edu/ welling/classnotes/classnotes.html.

[76] B. Scholkopf, Estimating the support of a high-dimensional distribution, Neural Computation 13 (7).

[77] V. Vapnik, Statistical Learning Theory, Wiley and Sons, New York, 1998.

[78] C. Chih-Chung, L. Chih-Jen, LIBSVM: A library for support vector machines, ACM Transactions on Intelligent Systems and Technology 2 (2011) 27:1–27:27, software available at http://www.csie.ntu.edu.tw/ cjlin/libsvm.

[79] C.-W. Hsu, C.-C. Chang, C.-J. Lin, A practical guide to support vector classification (2006).

[80] D. Tax, R. Duin, Support vector data description, Machine Learning 1 (2004) 45–66.

[81] C. Lampert, Kernel methods in computer vision, Foundations and Trends in Computer Graphics and Vision 4 (3) (2009) 193–285.

[82] D. Wang, P. Tse, W. Guo, Q. Miao, Support vector data description for fusion of multiple health indicators for enhancing gearbox fault diagnosis and prognosis, Measurement Science and Technology 22 (2) (2011) 025102.

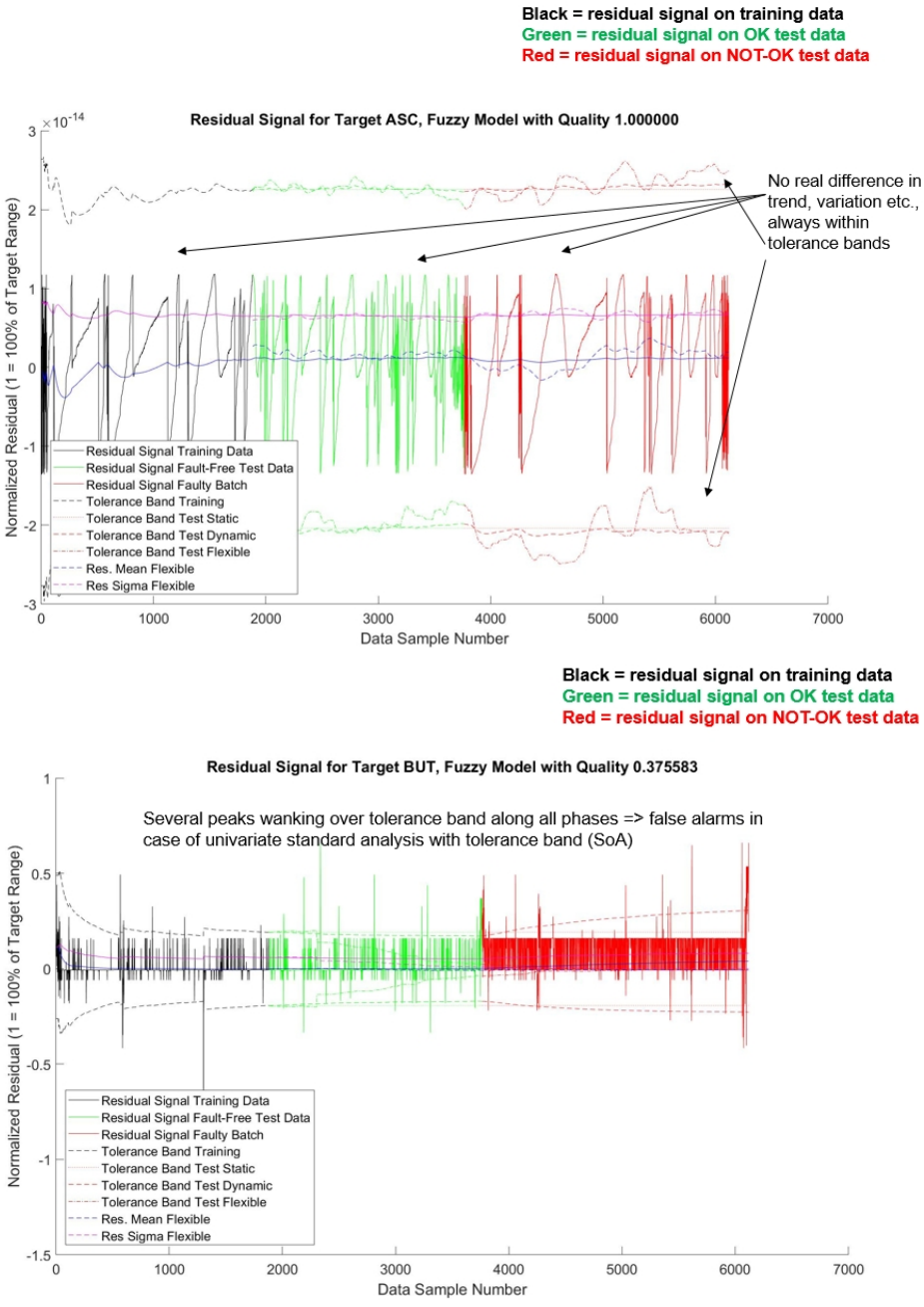

Figure 7a: Residual signal for ASC and BUT (two of the 7 models obtained at bonding liner and used for residual analysis); additionally, static and dynamic tolerance bands are shown in dashed and dotted lines as calculated in a native univariate residual analysis approach [22].

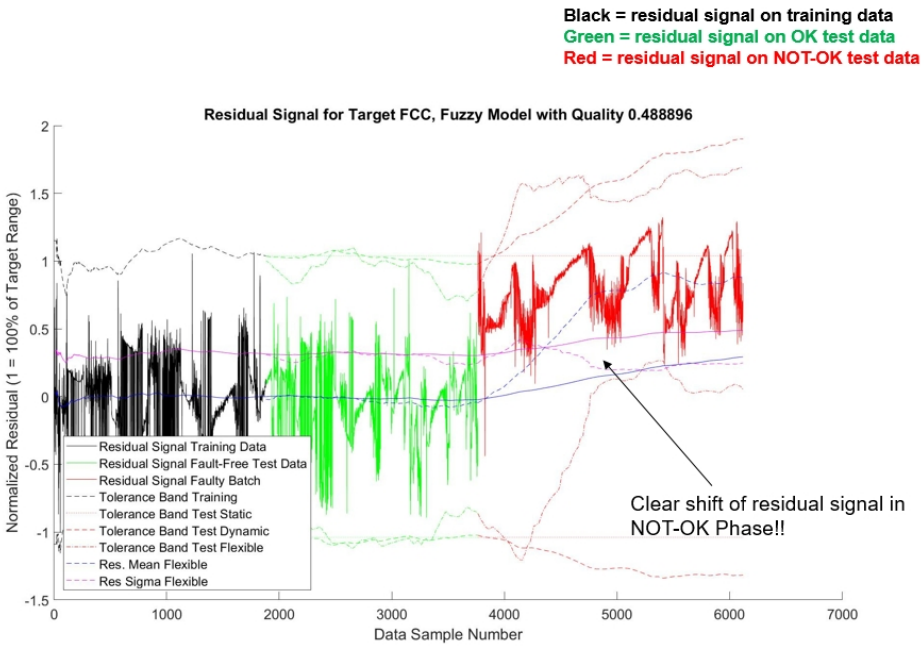

Figure 7b: Residual signal for FCC (another one of the 7 models obtained at bonding liner and used for residual analysis); additionally, static and dynamic tolerance bands are shown in dashed and dotted lines as calculated in a native univariate residual analysis approach [22].

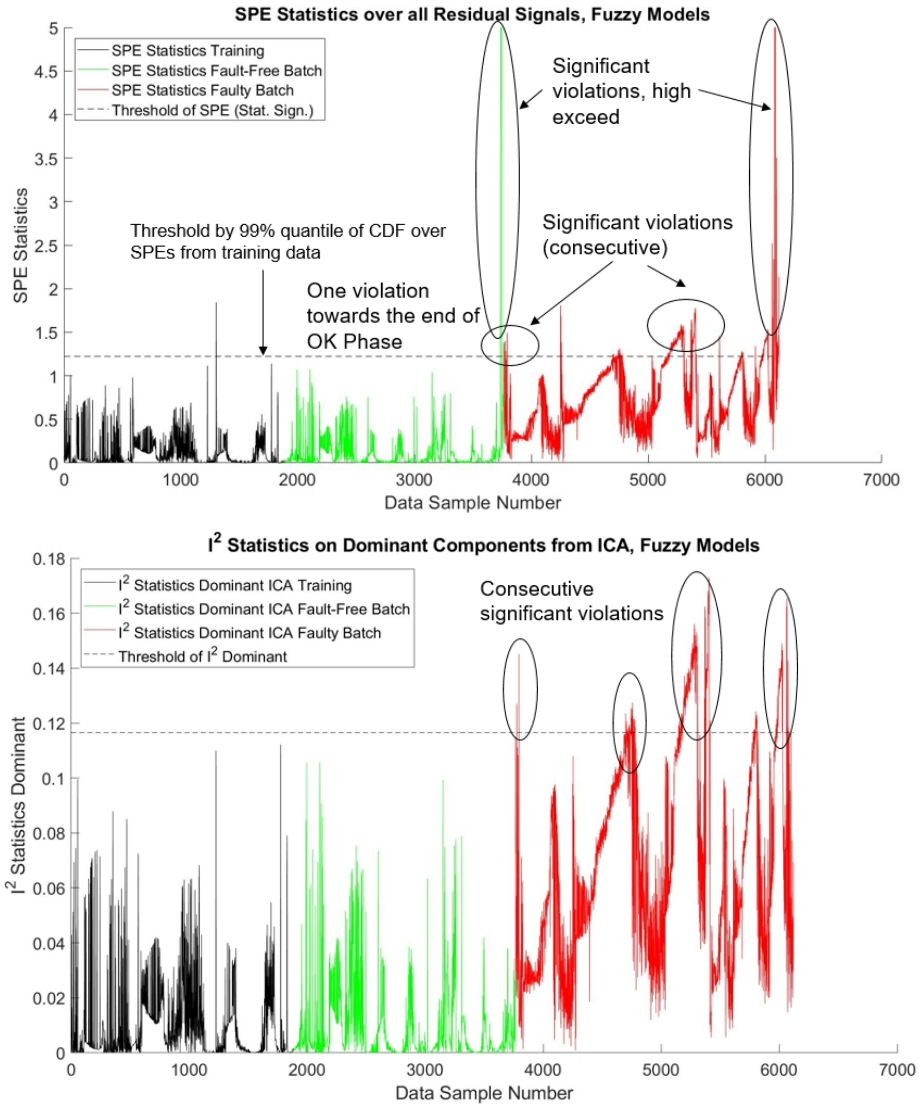

Figure 8a: upper: SPE statistics over the training data set (black line), test data set of the OK order (green) and test data set of the NOT-OK order (red) in case of BL02; consecutive and significant exceed of control limits are indicate as such; lower: I^2 statistics over the same data set portions.

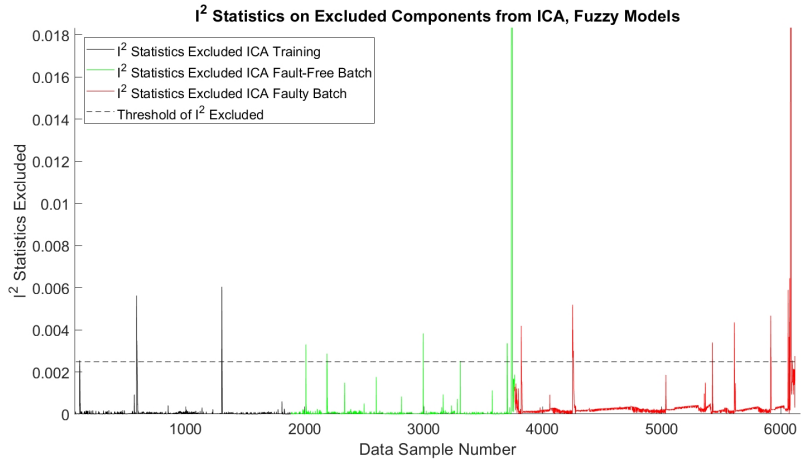

Figure 8b: The same as in the lower image in Figure 8, but using only the non-dominant parts in the I^2 statistics — please mind the difference in terms of several false alarms during the OK data set.

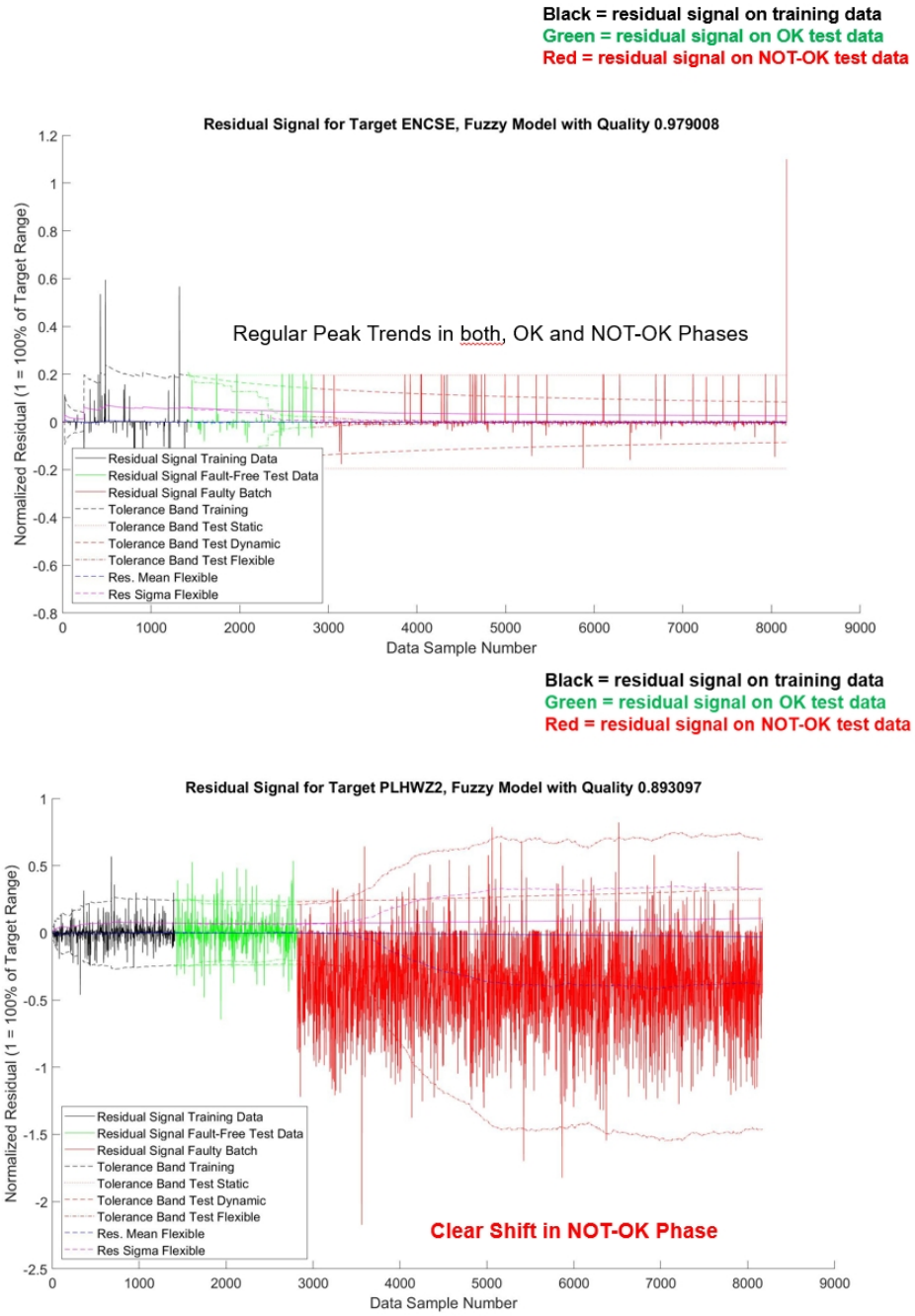

Figure 9: Residual signal for ENCSE and PLHWZ2 (two of the 31 models obtained at bonding liner and used for residual analysis); additionally, static and dynamic tolerance bands are shown in dashed and dotted lines for comparison purposes with native univariate residual analysis [22].

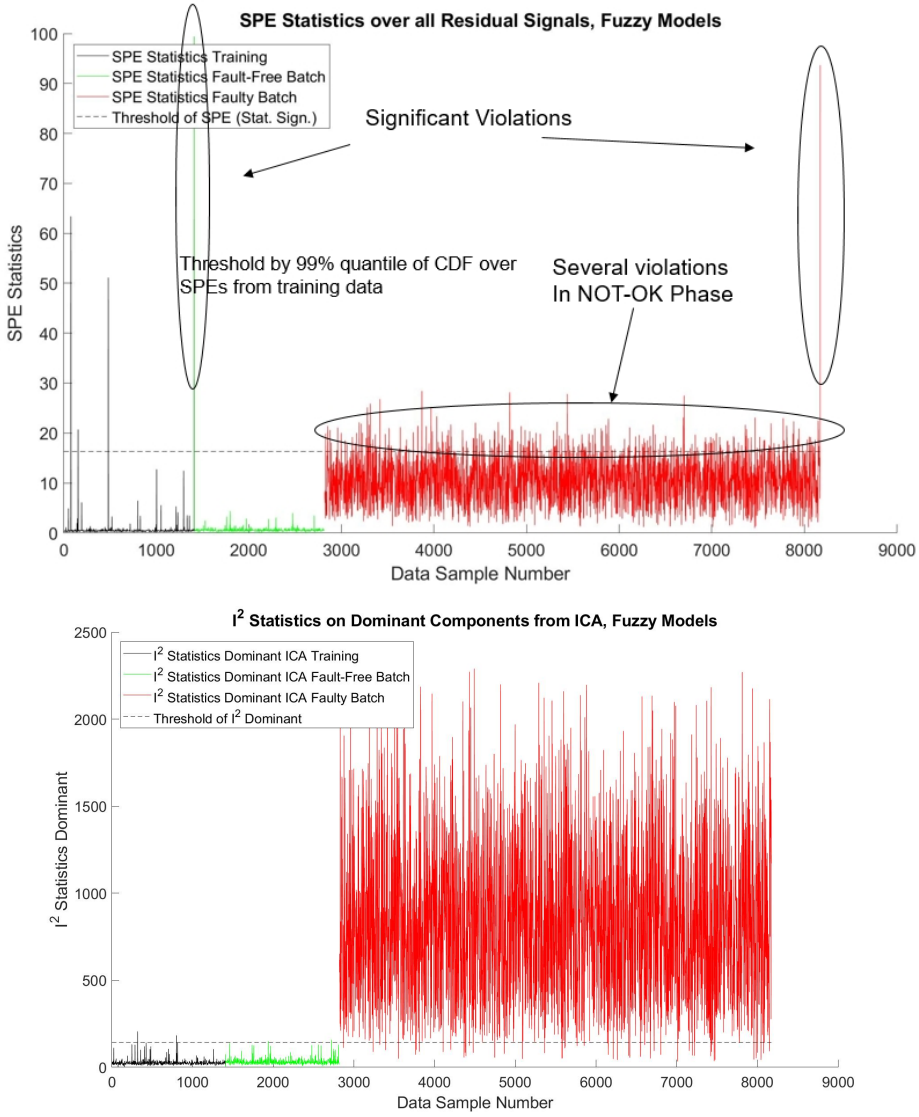

Figure 10: upper: SPE statistics over the training data set (black line), test data set of the OK order (green) and test data set of the NOT-OK order (red) in case of IE03; consecutive and significant exceed of control limits are indicate as such; lower: I^2 statistics over the same data set portions.

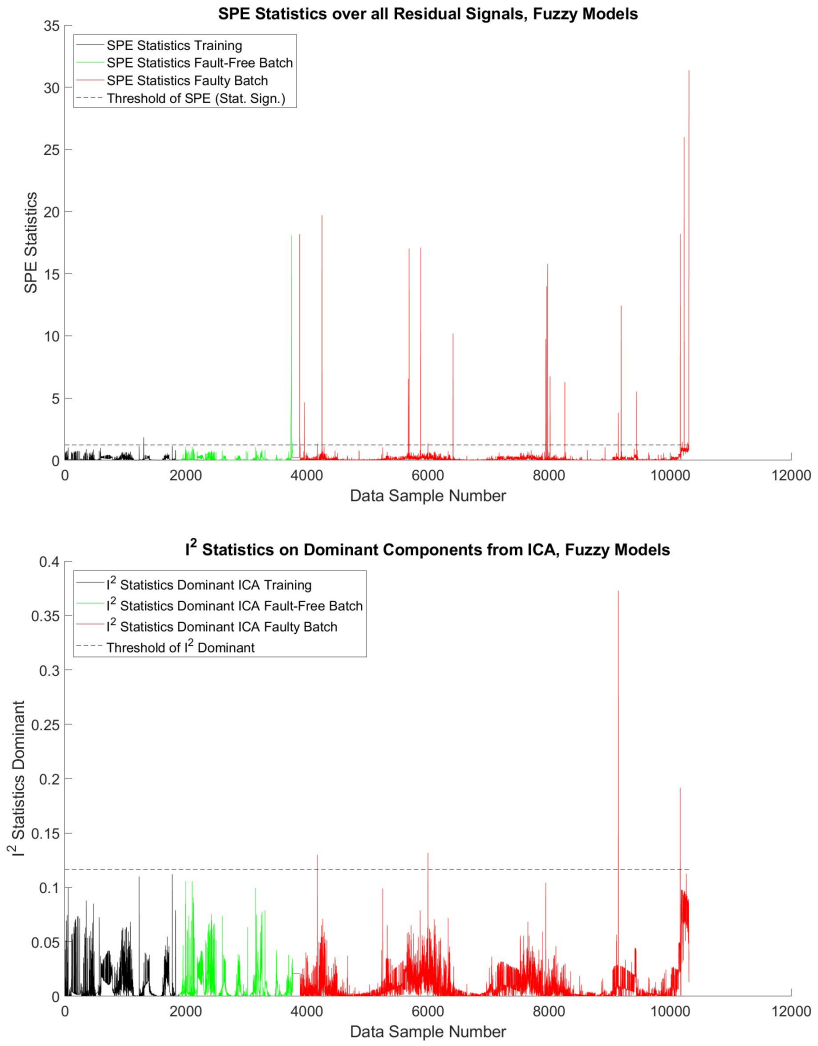

Figure 11: upper: SPE statistics over the training data set (black line), test data set of the OK order (green) and test data set of the OK order with different machining parameter (red) in case of BL02; consecutive and significant exceed of control limits are indicate as such; lower: I^2 statistics over the same data set portions.