Automated Diagnosis of Anomalies via Sensor-Step Data Outlier Detection: An Application in Semiconductors

MD Ridwan Al Iqbal
Rensselaer Polytechnic Institute
Troy, NY USA
iqbalm@rpi.edu

Andrés Vargas
Rensselaer Polytechnic Institute
Troy, NY USA
varga5@rpi.edu

John S. Erickson
Rensselaer Polytechnic Institute
Troy, NY USA
erickj4@rpi.edu

Kristin P. Bennett
Rensselaer Polytechnic Institute
Troy, NY USA
bennek@rpi.edu

ABSTRACT

The diagnosis of potential mechanical defects and sources of variation impacting wafer quality is an essential task in semiconductor manufacturing. During manufacturing, semiconductor fabrication tools are monitored by sensors that capture critical environmental information including temperature, power, and pressure. Our goal is to create an online and unsupervised anomaly detection system able to detect potential mechanical faults and other significant anomalies by monitoring sensor data from semiconductor process tools and then automatically generating a comprehensive report to help define putative causes of these anomalies for process engineers. Our framework represents the raw multivariate time series data as a tensor of sensor-step pairs. Exploiting information at recipe step granularity enables the framework to be easily deployed to new tools and recipes. The system finds instances that significantly differ from a normal model of the sensor-step pairs. A moving window Principal Component Analysis (PCA) model is used to capture the normal model, and departures from this model indicate potential anomalies. The system is modular, so different anomaly detection algorithms can be used. The dynamically-generated reports produced by the system contain lists of influential sensor-step pairs and also include visualizations and statistical tests explaining and validating the detected anomalies. The generated report includes the processed data and sample analysis code process engineers need to expedite further exploration of the anomalies. The anomaly detection and diagnosis system has been demonstrated by experimenting on wafer trace data generated by three different process nodes or types, with data from 13 different chambers. Our experimental results show that the detected sensor-step anomalies are indeed of engineering significance.

ACM ISBN 978-1-4503-2138-9.
DOI: 10.1145/1235

1. INTRODUCTION

During the manufacture of integrated circuits, wafers pass through a series of tools that perform different semiconductor processes. Each of these processes, including deposition, etching, and interconnection, implements a recipe consisting of a sequence of steps. Many sensors continuously monitor the state of each tool in order to control the process and potentially to detect faults.

Since semiconductors are manufactured at the nanometer scale, any process variation can lead to reduced quality and yield. Better methods for detecting changes and anomalies across an array of sensors are needed. But simply detecting the occurrence of anomalies is not sufficient; since the goal is to improve process yield, detection methods must also expedite the diagnosis of each anomaly to help process engineers identify potential root causes of the variations. The ideal deconstruction of detected anomalies has the following goals:

1. Indicate *sensors* associated with the anomalies;
2. Indicate the *recipe steps* in which the anomalies occurred;
3. Provide *explanations* as to why the anomalies were detected, including visualizations highlighting the detected sensor changes;
4. Prepare data and provide *sample analysis workflows* to support deeper investigations of the anomalies.

The fourth requirement is critical because highly skilled engineers currently spend significant time preparing sensor data for analyses and conducting analyses to diagnose source of faults. Tools such as dynamically generated "notebooks" can facilitate their investigations. An additional important requirement is *adaptability.* During manufacturing wafers may undergo hundreds of unique fabrication processes each implementing a different recipe. It is therefore important to have an adaptable anomaly detection method that is general enough to assist with the diagnosis of problems for any type of tool and any type of recipe.

This paper demonstrates how these deconstruction goals can be accomplished with a general-purpose anomaly detec-

Process node	Valid # Wafers	# of Recipe steps	# of Chambers	samples per wafer (mean)
Node α	31754	15	2/4	1293
Node β	4610	10-30	2/4	897
Node γ	5224	10-30	3/5	987

Table 1: Statistics about the data sets

tion system called *Anomaly Identification for Process Monitoring Online Framework* (AIPMOF) [4]. AIPMOF is a comprehensive modular machine learning framework that includes all aspects of fault detection including feature construction, unsupervised parameter tuning, real-time anomaly detection, unsupervised validation, supervised validation, and diagnosis[4]. AIPMOF is designed for rapid adaptation for different tool and recipe types with varying step lengths. For this paper, we focus on the diagnostic capabilities of AIPMOF when used with moving window PCA[5].

The key distinguishing features of AIPMOF include its core data representation and the way the framework performs diagnosis. AIPMOF takes as input variable-length time series sensor data annotated with recipe steps. AIPMOF transforms this raw data into a *sensor-step tensor representation*. Specifically, for each wafer, the data from each sensor-step pair is represented by summary statistics. Based on this tensor representation, AIPMOF signals when a significant change in the sensor data has been detected and then automatically generates a report that deconstructs the anomalies to facilitate diagnosis.

The report is in the form of an R Notebook[1] which enables process engineers to easily investigate the source of the anomaly using R [8]. These AIPMOF-generated notebooks combine R code and narrative text in a single self-contained document which includes data structures (specifically R data frames) containing various statistics supporting the anomaly detection process. The notebook's flexible structure allows engineers to easily include their own custom analysis and visualizations. The notebook also identifies the most influential features associated with each anomaly or change, leading to actionable insight into potential root causes.

For each wafer, the data associated with each sensor-step pair is represented by summary statistics. In this work we report results using the mean, but alternative statistics such as standard deviation, min, and max can be used. As discussed in the Related Work section, previous approaches have typically used time warping or interpolation to handle different segments of trace data for each wafer and they do not explicitly represent each step. By representing each step we can capture anomalies within and across steps, and we can localize each anomaly to specific sensors and steps in the process. This novel representation also allows the system to be highly adaptive to different process tools.

This paper describes our experiments with AIPMOF on three production data sets, each corresponding to a different deposition process. The data sets we studied are proprietary, so we anonymized the nodes, sensors and certain other identification information. All data sets have 16 sensors from power and gas flow sub-systems measuring power, temperature, voltage, current, pressure and throttle valve position possibly at multiple locations. Ultimately, our experiments with production data demonstrated that faults can be diagnosed by finding patterns in influential features and reported within automatically-generated R Notebooks.

In the following sections, we provide a literature review of related approaches and an overview of the deposition data sets. We discuss the details of AIPMOF including anomaly detection and diagnosis. We demonstrate the diagnosis algorithm and provide a description of the automatically generated R Notebook reports. We provide a detailed demonstration of how the system can be used for anomaly detection and diagnosis on three chambers and discuss different anomaly patterns. Finally, we show results of running the system on all of the chambers.

2. RELATED WORK

Anomaly detection has been an important research topic for application domains involving critical systems and requiring a level of consistency in operation. Applications of anomaly and change detection have included faults and failure detection in complex industrial systems [9] and sensor networks [13].

Aggarwal noted several methods for performing anomaly detection through unsupervised outlier detection techniques [1]. Hodge and Austin [3] surveyed fault detection techniques with statistical machine learning approaches. They have identified three categories of algorithms. In the first type, anomalies and outliers are identified with no labels; this can also be referred to as unsupervised clustering. The second type uses supervised classification and requires labeled data. The third approach uses one class classification and only determines normal behavior.

Outlier detection is another important aspect of our research. Knorr et. al. and Micenkova et. al. used subspace as possible explanation of outliers [7, 6]. They define a general algorithm that works with many outlier detection methods and tries to extract subspaces from the outliers.

AIPMOF is a modular framework based on unsupervised clustering that covers the full lifecycle of anomaly detection and deconstruction in smart manufacturing. Any of these prior detection and explanation algorithms could potentially be used as components in this framework.

Many papers have reported success with versions of the moving window PCA algorithm used in AIPMOF[11, 2, 12, 5]. As with AIPMOF, the T^2 and/or the SPE/Q statistics are used for triggering anomalies. Unlike AIPMOF, this prior work has typically used interpolation or time warping. AIPMOF uses a step-based data transformation in order to incorporate stepwise information.

Most papers lack an investigation or diagnosis of potential root causes of anomalies. A notable exception is [10], which uses an approach based on the T^2 statistic, but they only identify the sensor associated with the anomaly, whereas

[1]See *R Notebooks*, https://rmarkdown.rstudio.com/r_notebooks.html

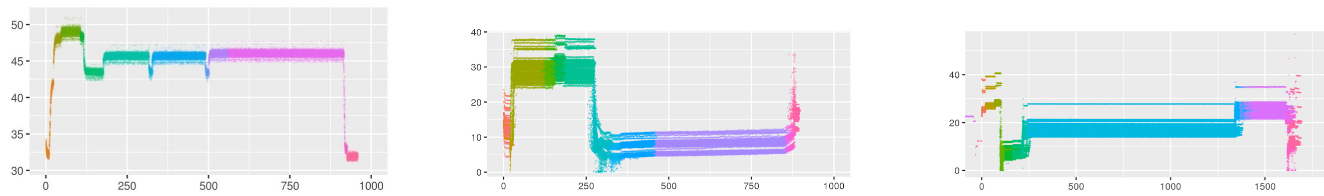

Figure 1: Throttle Sensor samples over time on one chamber from Node α, Node β & Node γ respectively. Colored by recipe step

AIPMOF uniquely identifies the sensor-step pair associated with anomalies. We could not find any indication in prior work on the automatic generation of an anomaly report that includes visualizations and statistical tests to confirm which sensors and steps are associated with anomalies.

3. DATA DESCRIPTION

Our experiments focused on three semiconductor deposition data sets. Each of these data sets corresponds to a different process node, or recipe, and contains sensor data gathered over an approximately fourth month period. In this discussion we identify these nodes as **Node α, Node β, Node γ**. Node α is a mature process node that has been fine-tuned over time to have a consistent performance behavior. Node β and Node γ are new processes showing more variation from wafer to wafer. Each node may be executed in different physical chambers (tools). The chambers (tools) used by the process nodes are all performing deposition but still vary significantly in their behavior.

All the tools we studied contained 16 sensors monitoring temperature and pressure control subsystems. Each process node has its own recipe and that recipe is divided into several steps; common steps include ramping up the temperature, introducing various gasses for the deposition chemical reaction, and cooling down the chamber. Figure 1 shows the raw sensor data for the sensor throttle valve for a single chamber from all three nodes colored by recipe step for all wafers processed in that chamber. It can be clearly seen that the number of recipe steps and the sensor behavior within recipe steps vary significantly which makes it necessary to divide the data into sensor-step pairs.

Node α has 15 steps in its process recipe; this number of recipe steps is consistent across all the wafers. The number of recipe steps for other nodes, is less consistent; the number of recipe steps varies across wafers in Node β and Node γ. Each data set exhibits a completely different run behavior.

The three nodes provide a large amount of "raw" time-series data. The sensor data is sampled at 10-second intervals. A typical wafer run consists of over 1000 units. The raw data was exported from the manufacturing system in over 55,000 XML-formatted files. Since the control system determines when to end a step based on the recipe definition, the number of samples per sensor-step per wafer varies.

The raw data is collected from multiple tools. Each tool is composed of two simultaneous running chambers (A & B). The data from each chamber of a tool can be considered independent of the other. Our analysis therefore is focused on the chamber rather than the tool level. As the wafer is being processed, raw sensor readings are taken at intervals. The data is then provided as a series of samples along with associated context information. The number of samples per step varies since the steps are of variable length. Some recipe steps, however, may have very few samples since they can be very short. This represents a data representation challenge as not all recipe steps are of the same length and then length does not determine the importance of the step.

4. ANOMALY DETECTION & DIAGNOSIS FRAMEWORK

In this section, we describe AIPMOF (*Anomaly Identification for Process Monitoring Online Framework*). AIPMOF is a flexible-modular system that is designed to be deployed in manufacturing various processes that have the recipe/step/sensor form. While AIPMOF is flexible, the result reported here are use the moving window PCA algorithm for the actual anomaly detection, since it is a well accepted approach.

We begin by explaining the tensor-based data representation/preprocessing framework. Then we explain the Moving window PCA based algorithm. With this background, we can described the diagnosis process. Note that AIPMOF also contains modules to perform model section of all hyperparameters and validation, but they are beyond the focus of the paper and can be found in [4].

4.1 Feature Construction

The data that we have been provided is a set of sensors that is sampled continuously over a time interval. Since not all wafers run the same length of time, this poses a challenge as the number samples is variable between different wafers. Another important factor is the existence of recipe steps. Each recipe step represents a different mode of operation as shown in Figure 1. Capturing the differences between recipe steps is important for anomaly detection.

Our solution to this challenge is a data conversion technique that converts the data of a particular wafer run into a set of variables that represents the pattern of that particular sensor at that particular step.

The per wafer data forms a 4 mode data tensor (multidimensional matrix), with the modes being wafer ID, sensor, step and statistics (features) such as mean. More precisely, $T_{i,j,s,f}$ is the feature f calculated for wafer i over step j for sensor s. We further make the assumption that the wafers are ordered chronologically, so that wafer i is processed immediately before wafer $i + 1$. We assume that a distinct tensor is formed for each chamber. The tensor is augmented with per-wafer data that captures the properties of each wafer including the chamber ID and start time. The flexibility of this representation comes from being able to add various types of statistics for each step/sensor pair. Different transformations of the raw time-series will only result in different features in the f index. Some potential

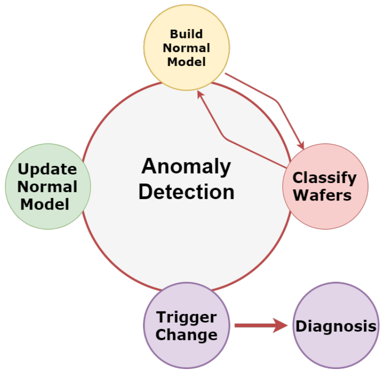

Figure 2: Online anomaly detection algorithm life-cycle

future statistics that could be added are the parameters of a function, dynamic time warping features, and other anomaly measures. In this work, we use the mean.

The next step is to matricize the tensor into an array, i.e. we convert the multi-dimmensional matrix into a two-dimensional matrix for analysis. Assume there are N wafers, J steps, S sensors, and F features. Then the data set, which lives in $R^{N \times J \times S \times F}$, is matricized into $X \in R^{N \times D}$ where $D = J \times S \times F$. We refer to the ith row of X as x_i.

4.2 Unsupervised Anomaly Detection with Moving Window PCA

Our basic anomaly detection strategy is an online anomaly detection system that can run continuously and trigger any unusual changes in process sensor data. This is done by maintaining a model that captures the normal behaviors and variations in the manufacturing process. This model can then be used in statistical test to identify and flag outlier wafers. If sufficient outliers occur, then a change has been detected. If new samples behave normally then we can update our model to incorporate this new data. Once a significant anomaly has been detected, then the model of normal behavior is completely recreated with a new set of observations as the new normal. The normal model is created based on a moving window of presumably normal samples.

Our online anomaly detection algorithm, Moving PCA for Anomaly Detection (MOPAD) uses this general structure as well. The basic idea is to capture the normal expected variance of a process using a principal component analysis (PCA) model.

Let $W = \{w_1, \ldots, w_n\}$ be the indices of the set of normal samples in a window of size n. This is considered the normal window. Let the window be $\hat{X} \in R^{n \times D}$. Note

$$\hat{X} = \begin{bmatrix} x_{w_1} \\ \cdots \\ x_{w_n} \end{bmatrix}. \tag{1}$$

$\hat{X}$ is scaled so that the data is standardized. PCA computes the eigendecomposition of the covariance matrix

$$\frac{1}{n-1}\hat{X}'\hat{X} = V\Lambda V' \tag{2}$$

where $V \in R^{D \times D}$ are the eigenvectors of the Covariance and Λ is a diagonal matrix with the corresponding eigenvalues, $\lambda_1 \geq \lambda_2 \geq \cdots \lambda_d$, in decreasing magnitude along its main diagonal.

Let P, which we also refer to as the *loadings*, be a matrix whose columns are the first r columns of V. Here, r is a parameter of the algorithm. Note that the columns of P are orthogonal. The scores are $T = \hat{X}P$.

There are two popular measures for detecting anomalies in using PCA [5]. The T^2 statistic, or Mahalonobis distance, is the most popular measure. For a new point x, the Hotelling T-square statistics measures the Mahalanobis distance of x from the mean (here scaled to be 0):

$$T_x^2 = xP\Lambda^{-1}P'x'. \tag{3}$$

Recall that x is a row vector in equation (3). The sensitivity of the T^2 statistics to the changes in feature j is

$$\frac{\partial T^2}{\partial x_j} = 2(P\Lambda^{-1}P'x')_j \tag{4}$$

where $(a)_j$ indicates the jth entry of vector a. T^2 follows a χ^2 distribution with r degrees of freedom.

An alternative measure of anomalies is the squared prediction error (SPE), also known as the Q statistic, which measures the residual reconstruction error:

$$Q_x = \|x' - PP'x'\|^2 = \|(I - PP')x'\|^2 = x(I - PP')x' \tag{5}$$

where the third equality stems from the fact that P has orthogonal columns. The Q-statistic above is distributed as a weighted sum of chi-squared random variables [11], which has an ambiguous pdf. This is fine for our algorithm because we don't select the firing value for the Q-statistic based on confidence regions of the pdf, but rather based on the raw value.

The sensitivity of the SPE statistics to the changes in feature j is

$$\frac{\partial Q^2}{\partial x} = 2((I - PP')x_j' \tag{6}$$

The overall structure of the moving window PCA algorithm is given in Algorithm 1. The algorithm has 4 parameters: Window Size n, Firing Value for T statistic, F_T and Firing Value for Q statistic F_Q, and number of PCA components k which are tuned automatically.

The algorithm starts with selecting a starting window which is a vector of observations with window size n, and initializing the list of abnormal wafers, L, to an empty column vector.

For each new wafer x_k. The algorithm would calculate the PCA loadings and score using equation 2 with W_{k-1} as the data matrix. Then using equation 5 or 3, the algorithm would determine the monitoring T^2 statistic and Q statistic for the observation x_k. Determining if an observation is anomalous is done by comparing the observed monitoring score with the associated firing value. This leads to four possible alternatives defining anomalousness for an observations x_k: (1) the T statistic fires, (2) the Q statistic fires, (3) either the T statistic fires or the Q statistic fires, or (4) both the T and Q statistics fire. The behavior of the algorithm will change based on which of these definitions are chosen. Note that option (2) is not recommended, since using only the reconstruction error does not incorporate the distance from the normal model. Options (1), (3), and (4) are given mathematically by

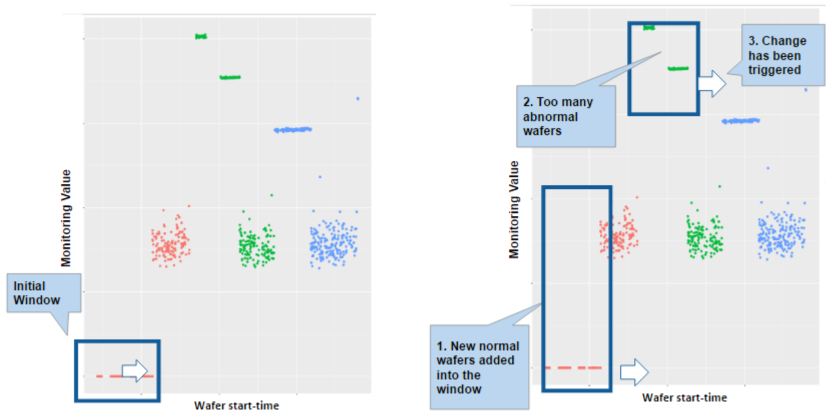

Figure 3: Moving Window PCA Illustration

$$Anom(x_k) \quad = \quad \begin{cases} T_x^2 \geq F_T \\ (T_x^2 \geq F_T) & \vee (Q_x \geq F_Q) \\ (T_x^2 \geq F_T) & \wedge (Q_x \geq F_Q) \end{cases} \tag{7}$$

Algorithm 1 Moving PCA for Anomaly Detection (MOPAD)

Algorithm: Moving PCA for Anomaly Detection
Input(Data matrix $X = R^{N \times D}$, window-size n, Firing-Value F_T, F_Q, PCA components k)
Output(Anomalies CP)

Select an initial normal window of n wafers $W_0 = \begin{bmatrix} w_1 \\ \vdots \\ w_n \end{bmatrix}$, where w_i is the ith index of the wafers in the normal window.
Initialize L, the abnormal window, to an empty column vector.
for k=1,2,... **do**
  if $Anom(x_k)$ is false **then**

$$W_k \leftarrow \begin{bmatrix} W_{-1,k-1} \\ w_k \end{bmatrix} \tag{8}$$

    Note: $W_{-i,k-1}$ is W_{k-1} with the i^{th} wafer removed.
  else
    $W_k \leftarrow W_{k-1}$ and $L \leftarrow \begin{bmatrix} L \\ w_k \end{bmatrix}$.
  end if
  if $length(L) > n$ **then** then
    1. Trigger Fault and add location of x_k into list of anomalies, CP.
    2. Update normal window $W_k \leftarrow L$ and reset L to an empty column vector.
  end if
end for

After the determination of $Anom(x_k)$, if x_k is normal, then the observation will be included into the current window W_k and the least recent observation will be removed. The list of anomalous observations L is always updated to empty if x_k is normal. This ensures that an anomaly is triggered when a consecutive list of wafers are considered outliers. If the observation is anomalous, then the observation will be included in another list of anomalous observations, L. If L becomes larger than window size then a fault is triggered and the window is updated with the faulty observation list becoming the new normal window, $W_k \leftarrow L$. This cycle will continue until the data ends triggering anomalies whenever too many anomalies occur sequentially.

The first stage of the anomaly detection algorithm is the selection of hyper parameters. The full description of our validation and hyper parameter selection scheme is provided in [4]. We used a training and testing set with each chamber belonging to either training or test set. This chamber wise split is then used for training of hyper-parameters. We used grid search to find the best parameters with an unsupervised clustering metric as the optimization criterion.

4.3 Influential Feature Identification

The algorithm can also determine the contribution of each feature in the determination of the monitoring value. The T^2statistic of x is the overall contribution of all the features in the feature space. We can calculate contribution of individual features as the sensitivity of T statistic to feature j, $Contrib_j(x) = \frac{\partial T^2}{\partial x_j}$

$$Contrib_j(x) = \frac{\partial T^2}{\partial x_j} = 2(P\Lambda^{-1}P'x')_j \tag{9}$$

By aggregating the contribution score of both sides of Fault/Change point Trigger, f. We can create a feature score for feature j and anomaly f,

$$Feature - score(j, f) = \sum_{x=f-n,f+n} Contrib_j(x) \tag{10}$$

In equation 10, we are using the the data on both sizes of a change point to aggregate the feature score. The moving window size n is used as size of the feature contribution for that particular change point.

Feature-score for each sensor-step pair is then sorted to get the list of top influential features $S_{k,f}$ where $S_{1,f}$ is the top most feature for the anomaly point f and $S_{k,f}$ is kth most influential feature for anomaly point f.

Based on this feature score, it is possible to identify the

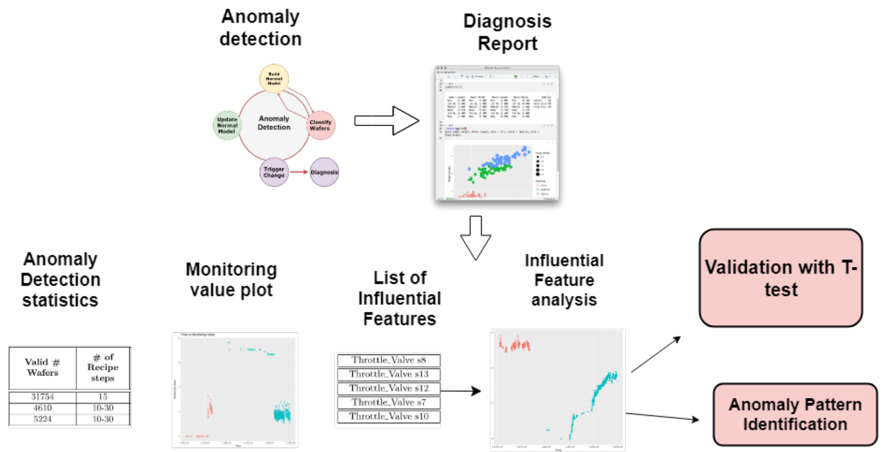

Figure 4: R Notebook dynamic diagnosis report components

Algorithm 2 Validating top features via Welch's t-test

Algorithm: Calculate T-test Accuracy
Input(Data matrix X, Top Feature List S, window-size n, Change Points CP)
Output(T-test Accuracy TAccu)

```
for anomaly point f ∈ CP do
    Get the list of top 5 features ŝ = {s_{1,f} ... s_{5,f}}
    for each feature tf ∈ ŝ do
        Calculate a T-test between two sides of f:
        t_success(tf) = T − test(X^{tf}_{f−n,f−1}, X^{tf}_{f,f+n})
        if t_success is True then
            TAccu(c) = True
        end if
    end for
end for
```

features with the highest feature scores. These features can be considered root cause candidates.

4.4 Validation of Influential Features

Our main goal in this validation method is the verification of significant anomalous behavior. Ideally, when there is an anomaly, we expect that change will also happen in some meaningful sensor-step pairs. Top contributing features detected by our method can be considered these meaningful sensor-step features. If we find a statistically significant change in the top influential features, then this can be considered a meaningful diagnosis.

This can be measured by calculating a test of statistical significance. We are using Welch's T-test as the test of significant difference between the two sides of a change point. The t-test validation algorithm works by calculating a **T-test Accuracy** measurement. We are making the assumption that at least one of top 5 features would have a significant change in order for the algorithm to be performing well. If any of the top 5 features are not changing in statistically significant test, then that particular change could be considered a false positive. This can be considered a method to calculate true and false positives.

The t-test compares between two groups observations. In our case, we compare the two sides of a change point $f \epsilon CP$. The size of each group would be the window size n of the change detection algorithm.

5. DYNAMIC DIAGNOSIS REPORT

The key output of the anomaly detection framework is dynamically-generated reports of each run of the anomaly detection algorithm over batches of consecutive wafers. These reports allow engineers to quickly understand the current state of tools where individual wafers are being processed. If there had been a significant change in behavior and an alarm was triggered, the report will contain a list of most influential features as well. The generated report also includes various other plots and statistics for understanding the anomaly that may have occurred.

The report is generated as an R Notebook file. The anomaly detection algorithm is implemented as a separate R script that generates its output as data structures meant to be consumed by R code in the notebook. Code blocks in the R Notebook load the data and generate the associated statistics, figures and other information. The key advantage of this approach is the ability to dynamically change how the results of the anomaly detection can be displayed and processed further. The notebook code can be modified by process engineers to generate further analyses as needed.

The first step in fault detection and diagnosis is running the anomaly detection algorithm on the sensor data set for wafers over a select period of time. Here we considered four months of data in each run. The dynamically-generated R Notebook is produced at the end of the run. Figure 4 shows an illustration of the detected anomaly leading to the diagnosis report. Notice that the R Notebook includes the following items:

1. **Run statistics:** The initial part of the report shows the different run statistics such as the number of wafers run during the process operation, average sample length, and number of anomalies alerted.

2. **Monitoring value analysis**: The next part of the re-

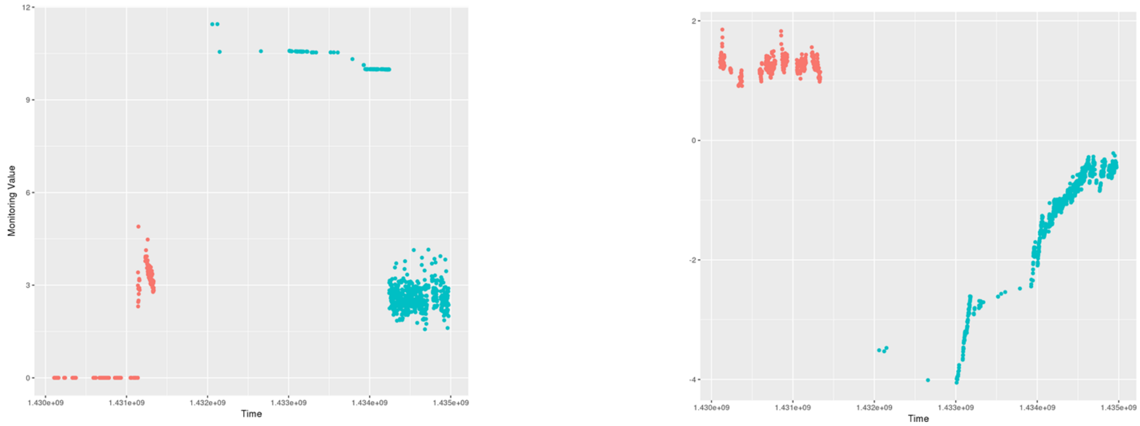

Figure 5: Node α Ch1B - Monitoring value and influential feature “Throttle Valve s8” over time

port shows how the monitoring value varied over time and statistics such as mean and standard deviation. This can be used to look for patterns in the chamber behavior. The monitoring value distribution can be used to check if there are any time period with unusual variance. There could be certain time periods when the number of anomalous wafers were higher than normal but not enough to trigger a change. These kind of situations could be observed using the monitoring value. Figure 5 shows the time vs monitoring value plot for chamber Ch1B from Node α.

3. **List of influential features:** This part of the report shows the most influential features found by the algorithm. We report the top five for this study, but other criteria such as statistical tests of significance could be used. These influential features are sensor-step pairs that point at potential root causes for anomalies. They help the engineers focus on the appropriate steps and sensors and can indicate the potential anomaly type. Table 4 shows the top five influential from one chamber per Node type. Looking at these feature lists, we see a pattern in the list of influential step-sensors. In the case of Ch1B from Node α, the influential features all come from throttle valve sensor across many steps which provides strong evidence that the throttle valve could be the main root cause.

4. **Influential feature behavior plots:** The report provides the raw sensor values of the influential features during the anomalous state using this plot. The relationship between the monitoring value and how influential features changed during a triggered anomaly can be examined. This allows engineers to see if influential features show significant changes in behavior in these wafers.

5. **Validation of influential features:** This part of the report validates the diagnosis suggestion using statistical significance test. The goal is to verify if the influential features do vary significantly across the change point.

6. DIAGNOSIS PERFORMANCE ANALYSIS

In this section we discuss the results of our experiments using the three process node data sets. The results were generated via a chamber-wise split of training and testing data. Our hyper-parameter selection method is discussed in a full-length paper presenting the AIPMOF framework [4].

Tables 2 & 3 show the mean t-test accuracy and P value for all data sets. The T-test accuracy is 1.00 for all data sets. The p-value in every chamber was also low as indicated by the average p-value. This demonstrates that the top five influential features found by the algorithm show statistically significant change when an anomaly is alerted. This also validates our algorithm performance by showing that we are indeed finding anomalies represent significant changes.

In the case of diagnosis, we have seen a few common type of patterns of potential root cause among the chambers. These are:

- **Sensor Anomaly:** Influential features come from one sensor over multiple steps. Therefore, the most probable cause of a mechanical problem could be the sensor.
- **Step Anomaly:** Influential features come from multiple sensors from one single step. The problem then could mostly be isolated to that particular step.
- **Consecutive Step Anomaly:** Influential features come from consecutive steps over multiple sensors.
- **Subsystem Anomaly:** Influential features come from multiple sensors from the same subsystem (Power or Gas flow).

We will now show one detailed example of diagnosis per data set in order to demonstrate how the diagnoses are used. The list of influential features for one chamber per node is shown in Table 4. The first chamber is from node α Ch1B which shows quite high variance before a change was triggered. Figure 5 shows the monitoring value and influential feature behavior side by side. Looking at the right plot, it is clearly visible that the influential feature Throttle Valve shows a significant shift. The list of influential features as shown in Table 4 shows the top 5 influential features of this chamber. All the top 5 influential features belong to Throttle Valve and therefore follow the sensor anomaly pattern. Thus, the engineers can suggest investigating the throttle valve for potential problems. For example, the throttle valve may become clogged due to debris over time.

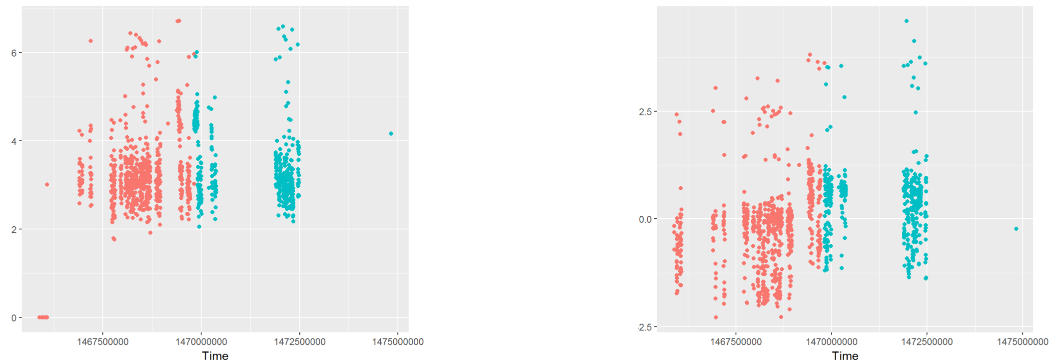

Figure 6: Node β Ch2A - Monitoring value and influential feature "Voltage 1 s2" over time

Cham.	# anom.	T-Test Accu.	T-test P
Ch1B	2	1.00	2.72E-78
Ch2A	4	1.00	2.74E-07
Ch2B	2	1.00	3.58E-01
Ch3B	2	1.00	6.58E-300
Mean	**2.5**	**1.00**	**8.94E-2**

Cham.	# anom.	T-Test Accu.	T-test P
Ch1B	2	1.00	3.93E-01
Ch1B	2	1.00	3.07E-04
Ch2A	2	1.00	4.27E-07
Ch2B	4	1.00	3.47E-01
Mean	**2.5**	**1.00**	**1.85E-01**

Table 2: Anomaly detection results on Node α & Node β for each chamber. The #anom is number of changes found. T-test accuracy indicates % of number of chambers with significant change in 5 most influential features and T-test P is the average p-value found.

We next look at chamber Ch2A on Node β. This chamber shows much higher variance than node α which is an older and more fine-tuned process. Node β consistently shows higher variance throughout the process. This can attributed to the fact that this is a newer process with less fine tuning.

The monitoring value can be used to view this information about chamber variances at a glance. The diagnosis pattern in Node β - Ch2A shows that the influential features all belong to the same step and they are all from the power subsystem (Voltage, Current, Power). This shows that the potential venue for investigation is step 02.

Cham.	# anom.	T-Test Accu.	T-test P
Ch1A	2	1.00	1.41E-01
Ch2A	3	1.00	1.32E-02
Ch2B	2	1.00	1.77E-02
Ch3A	2	1.00	5.72E-01
Ch3B	2	1.00	1.35E-01
Mean	**2.2**	**1.00**	**2.41E-01**

Table 3: Anomaly detection & Diagnosis γ

The final demonstration is also from Node γ - Ch2B. Comparing this chamber monitoring value report to the other two chamber reports, one can see a clear difference between the variances of the two chambers. This is another way to use the monitoring value to compare multiple chambers. The chamber Ch2B shows variation in a subsystem as a potential root cause. The influential features all belong to the same subsystem but not the same steps. Both step 6 and step 10 appear in the influential feature list.

7. CONCLUSIONS

We have presented a novel system for detecting and diagnosing anomalies in semiconductor manufacturing. The key insight is that by representing the data using summary statistics for each sensor-step pair, the system can identify both the sensors and individual process steps causing an anomaly in a form that is natural to the engineer. For example, a sticky throttle valve will cause an anomaly across many steps, whereas an incorrectly specified control parameter will effect many sensors in a single step. The underlying framework, AIFMOF, is adaptable, allowing these methods to be rapidly deployed for new tools and process recipes with minimal tuning, a critical requirement in semiconductor manufacturing since a single wafer may experience hundreds of different processes.

AIPMOF is modular, enabling different sensor-step representations and anomaly detection algorithms to be plugged into the system. In the work presented here, we used the mean to summarize sensor-step data per-wafer, and moving window PCA for the normal model since it is a robust method requiring minimal tuning with clear interpretations.

AIPMOF embodies a new approach to anomaly detection with many possibilities for future work. Computationally, our methods have focused on anomaly detection within the context of a single chamber. However, our per wafer sensor-step tensor representation enables the identification of sensor-step variations across multiple tools running the same process that may not be apparent in a single chamber.

Alternative computational methods for anomaly detection including deep learning could be used with our sensor-step representation, but appropriate interpretation strategies would need to be developed for process engineers. AIPMOF currently generates R Notebooks with an embedded data representation and analysis pipeline for engineering analysis and interpretation. This represents an improvement over the current manual methodologies for investigating complex anomalies, but requires significant data analytics skill to explore further. Eventually, a natural language

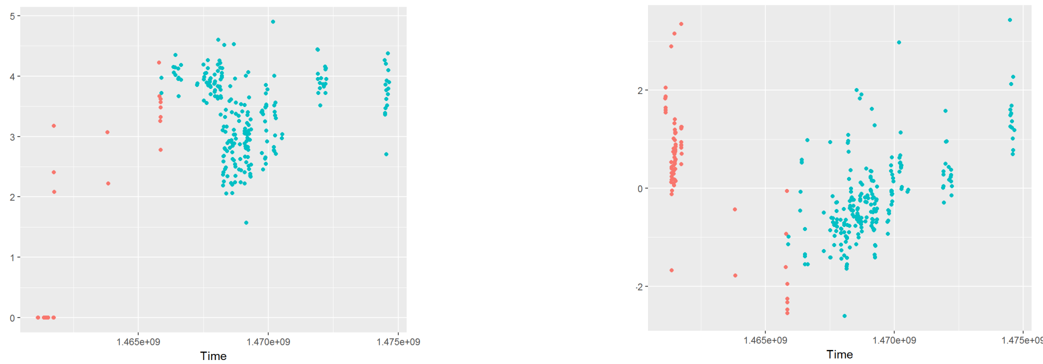

Figure 7: Node γ Ch2B - Monitoring value and influential feature "Temp 1 s6" over time

Node α- **Ch1B**	Node β- **Ch1B**	Node γ- **Ch2B**
Throttle_Valve s8	Voltage_1 s2	Temp_1 s6
Throttle_Valve s13	Power_1 s2	Temp_1 s7
Throttle_Valve s12	Voltage_2 s2	Current_2 s10
Throttle_Valve s7	Power_3 s2	Power_04 s10
Throttle_Valve s10	Voltage_3 s2	Voltage_2 s10

Table 4: Top 5 features of 3 chambers from all three datasets

artificial intelligence system could allow the engineer to more naturally interact with the system to delve into the analysis.

Finally, the results presented here model each sensor-shape pair by its mean in order to identify anomalous shape-sensor pairs. This may be acceptable in manufacturing processes in which steps achieve a set point, but is not sufficient for steps with complex waveforms. We are currently developing a "shape-signatures" approach for summarizing the sensor-step data, in which a parametric model of the process step is created from the data, informed by the ordinary differential equations modeling the PI-control systems implementing the process. Shape signatures will provide an even more informative representation of the observed process, capturing more subtle anomalies, providing deeper diagnoses, identifying the specific class of faults, and potentially even suggesting changes to control parameter to improve processes.

8. ACKNOWLEDGEMENTS

Partial funding and semiconductor process data was provided by GLOBALFOUNDRIES. We would like to thank GLOBALFOUNDRIES for their considerable support. We also thank Mr. Mark Reath of GLOBALFOUNDRIES for his support, mentoring and encouragement. This work was also supported by NSF Grant 1331023 and the Rensselaer Institute for Data Exploration and Applications.

9. REFERENCES

[1] C. C. Aggarwal. *Outlier Analysis*. Springer, 2013.

[2] I. Applied Materials. *Epi Centura Process Manual*, Sep 1995.

[3] V. J. Hodge and J. Austin. A survey of outlier detection methodologies. *Artificial intelligence review*, 22(2):85–126, 2004.

[4] M. R. A. Iqbal, A. Varga, J. Erickson, and K. P. Bennett. A framework for unsupervised online anomaly detection for semiconductor process tools. Manuscript in review for release for publication, 2018.

[5] S. Joe Qin. Statistical process monitoring: basics and beyond. *Journal of Chemometrics*, 17(8-9):480–502, 2003.

[6] E. M. Knorr and R. T. Ng. Finding intensional knowledge of distance-based outliers. In *Proceedings of the 25th International Conference on Very Large Data Bases (VLDB)*, pages 211–222, Edinburgh, Scotland, 1999.

[7] B. MicenkovÃ¡, R. T. Ng, X. H. Dang, and I. Assent. Explaining outliers by subspace separability. In *2013 IEEE 13th International Conference on Data Mining*, pages 518–527, Dec 2013.

[8] R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2017.

[9] L. Tarassenko, D. A. Clifton, P. R. Bannister, S. King, and D. King. Novelty detection. *Encyclopedia of Structural Health Monitoring*, 2009.

[10] A. Thieullen, M. Ouladsine, and J. Pinaton. Application of principal components analysis to improve fault detection and diagnosis on semiconductor manufacturing equipment. In *2013 European Control Conference (ECC)*, pages 1445–1500, July 2013.

[11] D. X. Tien. Moving pca for process fault detection - a performance and sensitivity study. Master's thesis, National University of Singapore, 2005.

[12] S. Wold. Exponentially weighted moving principal components analysis and projections to latent structures. *Chemometrics and intelligent laboratory systems*, 23(1):149–161, 1994.

[13] Y. Zhang, N. Meratnia, and P. Havinga. Outlier detection techniques for wireless sensor networks: A survey. *IEEE Communications Surveys & Tutorials*, 12(2):159–170, 2010.