CROSS-PROCESS DEFECT ATTRIBUTION USING POTENTIAL LOSS ANALYSIS

Tsuyoshi Idé[1], Kohei Miyaguchi[2]

[1]IBM Semiconductors, IBM Thomas J. Watson Research Center, New York, USA.
[2]IBM Research – Tokyo, Tokyo, JAPAN (currently affiliated with LY Research, JAPAN).

ABSTRACT

Cross-process root-cause analysis of wafer defects is among the most critical yet challenging tasks in semiconductor manufacturing due to the heterogeneity and combinatorial nature of processes along the processing route. This paper presents a new framework for wafer defect root cause analysis, called Potential Loss Analysis (PLA), as a significant enhancement of the previously proposed partial trajectory regression approach. The PLA framework attributes observed high wafer defect densities to upstream processes by comparing the best possible outcomes generated by partial processing trajectories. We show that the task of identifying the best possible outcome can be reduced to solving a Bellman equation. Remarkably, the proposed framework can simultaneously solve the prediction problem for defect density as well as the attribution problem for defect scores. We demonstrate the effectiveness of the proposed framework using real wafer history data.

1 INTRODUCTION

The latest technology nodes in semiconductor manufacturing involve more than one thousand process steps across about a dozen process types such as deposition and etching. Cross-process root-cause analysis of wafer defects spanning the entire processing sequence is among the most critical yet challenging tasks in semiconductor manufacturing. Particularly during process integration and yield ramp-up stages, classical design-of-experiment methodologies, which analyze process outcomes across systematically varied parameters, are often impractical due to the prohibitively large number of adjustable parameters along the processing route. Although a wide range of off-the-shelf machine learning tools are publicly available, most of these tools are designed for prediction tasks, such as estimating real-valued outputs (regression) or categorical outcomes (classification). As a result, fab-wide defect diagnosis still heavily depends on manual, ad hoc analysis by domain experts.

To support more systematic analysis, three major directions have been pursued in the semiconductor analytics literature to date. The first approach treats defect diagnosis as a by-product of cross-process virtual metrology (VM) modeling. Regularized linear regression combined with variable selection techniques is commonly used (e.g., (Susto, Pampuri, Schirru, Beghi, and De Nicolao 2015; Jebri, El Adel, Graton, Ouladsine, and Pinaton 2016; Kim, Kim, Jun, Chong, and Song 2018)). However, linear models struggle to capture complex nonlinear relationships across heterogeneous fabrication processes. Furthermore, defect attribution based on linear models is essentially reduced to variable-wise correlation analysis, which is known to yield only weak attribution signals (Miyaguchi, Joko, Sheraw, and Idé 2025a).

The second direction is to leverage recurrent neural networks (RNNs) and Transformers to replace conventional VM models. While they can capture complex nonlinear dependencies in sequential processes (e.g., (Yella, Zhang, Petrov, Huang, Qian, Minai, and Bom 2021; Han, et al. 2023; Dalla Zuanna, Gentner, and Susto 2023; Lee and Kim 2020; Hsu and Lu 2023)), handling different processes requires significant feature engineering effort. Additionally, they typically operate as black boxes, making input attribution a non-trivial task. These models are also generally data-intensive, and it is often difficult to collect enough data to cover the combinatorial complexity of the fabrication process.
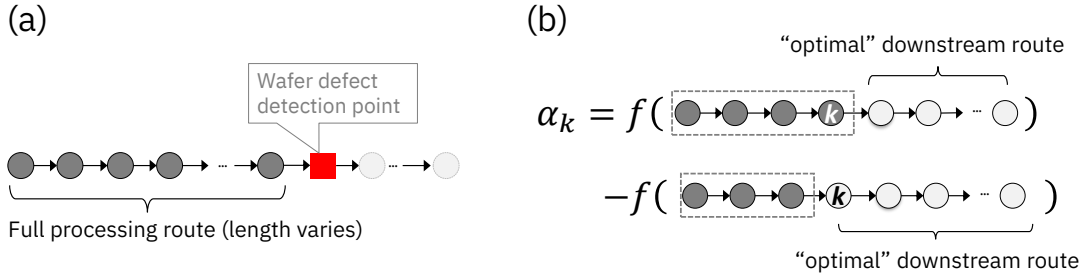
(a)

Wafer defect detection point

Full processing route (length varies)

(b)

"optimal" downstream route

$$\alpha_k = f(\ \boxed{\bullet\!\to\!\bullet\!\to\!\bullet\!\to\! k}\ \bullet\!\circ\!\to\!\circ\!\to\cdots\to\circ\ )$$

$$-f(\ \boxed{\bullet\!\to\!\bullet\!\to\!\bullet\!\to\! k}\ \circ\!\to\!\circ\!\to\cdots\to\circ\ )$$

"optimal" downstream route

Figure 1: Problem setting and the key idea. (a) We are interested in identifying upstream processes responsible for wafer defects detected at a specific detection point. The number of process steps along the route can vary. (b) The key idea of the PLA approach. Instead of zeroing out the process embeddings on the downstream path, it solves an optimization problem for optimal downstream routes.

The third direction involves leveraging explainable artificial intelligence (XAI) techniques applicable to black-box prediction models. This is a promising direction in that it can potentially enhance expressive prediction models with interpretability, helping to identify which process steps should be adjusted to improve defect rates. However, for cross-process defect attribution, existing methods, such as those based on Shapley values (e.g., (Torres, Kissiov, Essam, Hartig, Gardner, Jantzen, Schueler, and Niehoff 2020; Senoner, Netland, and Feuerriegel 2022; Lee and Roh 2023; Guo and Chen 2024)), pay limited attention to key characteristics of semiconductor processes, such as the sequential nature of fabrication steps. In addition, they often rely on assumptions that may not be fully justifiable in semiconductor manufacturing, such as dependence on arbitrarily selected baseline inputs. Ironically, these XAI approaches are often used as yet another form of black-box reasoning without careful justifications.

Recently, a new framework called the partial trajectory regression (PTR) was proposed to address these issues, such as capturing the sequential nature of fabrication and the lack of direct attribution capability (Miyaguchi, Joko, Sheraw, and Idé 2025b). However, similar to Shapley-value-based approaches, its attribution mechanism still suffers from potential biases due to inappropriate model assumptions, as discussed in detail later.

In this paper, we propose a novel framework called *potential loss analysis* (PLA), as a significant enhancement over PTR. Figure 1 illustrates the key idea. Our goal is to attribute an observed wafer quality issue by computing a responsibility score (or attribution score) for each upstream process. To evaluate the influence of the k-th process, we compare counterfactual outcomes based on partial process routes with and without the target process. The key idea is to use *optimal* downstream routes and compare the outcomes from the best possible continuations. To identify such optimal routes, we formulate wafer processing as a sequential decision-making problem and solve a Bellman equation. To the best of our knowledge, this is the first work to introduce the notion of path optimization into wafer defect attribution. We demonstrate the effectiveness of the proposed framework using real wafer history data from a state-of-the-art FEOL (front-end-of-line) process.

2 RELATED WORK

This section provides more detailed context of the problem we address with a particular focus on cross-process virtual metrology and explainable AI.

2.1 Cross-Process Virtual Metrology

A key characteristic that distinguishes semiconductor manufacturing from other industrial domains is its process complexity. A typical semiconductor process involves more than hundreds of intricate and highly specialized physical operations, including photolithography, thermal annealing, polishing, wet and dry etching, ion implantation, electroplating, sputtering, and chemical vapor deposition, among others.

For cross-process root-cause analysis, these heterogeneous operations must be mapped to a shared representation space to enable meaningful comparisons. Existing literature offers three general approaches for this. The first approach utilizes process trace data (Xu, Zhang, Sun, Chen, Qin, Lv, and Zhang 2024; Fan, Hsu, Tsai, Chou, Jen, and Tsou 2022). While effective within individual process tools, this method requires extensive tool-specific preprocessing, and the quality of analysis heavily depends on the chosen preprocessing strategy, making it less suitable for cross-process analysis. The second approach uses inline measurements as proxies for physical processes. Since these measurements partially absorb the physical heterogeneity across processes, this method has become common practice in recent studies (Senoner, Netland, and Feuerriegel 2022; Guo and Chen 2024; Wang and Chen 2024; Ni, Rui, Zhuo, Li, Wen, and Nie 2025). However, these approaches typically disregard the sequential order of processes and perform root-cause analysis as a by-product of virtual metrology (VM), often via univariate correlation analysis, which is known to yield only weak attribution signals (Miyaguchi, Joko, Sheraw, and Idé 2025a).

The third approach involves embedding techniques, where data objects (e.g., process steps) are transformed into numerical vector representations. For instance, Fan et al. (Fan, Lin, and Jen 2022) use one-hot encoding to unify categorical and numerical data in the VM setting. Schulz et al. (Schulz, Jacobi, Gisbrecht, Evangelos, Chan, and Gan 2022) propose defining a fab state vector using known interdependencies among processing tools under an unsupervised setting, without the context of wafer defect analysis. More recently, Miyaguchi et al. (Miyaguchi, Joko, Sheraw, and Idé 2025b) proposed proc2vec and route2vec algorithms that think of process attributes as synthetic words and capture their similarity using kernel embedding. The proposed PLA framework uses their approach as a building block.

2.2 Explainable AI (XAI)

Numerous methods have been developed to improve the interpretability of machine learning models under the umbrella of XAI (Xu, Uszkoreit, Du, Fan, Zhao, and Zhu 2019). One widely used category is additive explanation methods (Lundberg and Lee 2017; Ribeiro, Singh, and Guestrin 2016), which provide mathematically justified decompositions of a model's output into individual contributions of input variables. Other common XAI techniques include gradient-based methods (Selvaraju, Cogswell, Das, Vedantam, Parikh, and Batra 2020) and attention-based methods (Ali, Schnake, Eberle, Montavon, Müller, and Wolf 2022).

While most XAI studies have traditionally assumed vector inputs whose dimensions can be arbitrarily reordered, a growing—though still relatively limited—body of research is beginning to address the unique challenges posed by sequential data, particularly time-series data (Rojat, Puget, Filliat, Del Ser, Gelin, and Díaz-Rodríguez 2021). In the specific field of semiconductor analytics, the integration of model-agnostic XAI techniques with advanced VM models is emerging as a research trend, aiming to balance predictive power with interpretability. Among the wide variety of XAI methods (see, e.g.,(Molnar 2020) for an overview), the majority of recent studies adopt the Shapley value (Torres, Kissiov, Essam, Hartig, Gardner, Jantzen, Schueler, and Niehoff 2020; Senoner, Netland, and Feuerriegel 2022; Lee and Roh 2023; Guo and Chen 2024), possibly due to the availability of a well-designed Python implementation (Lundberg and Lee 2017). Ironically, despite its widespread adoption, most studies apply XAI methods as black boxes, without critically examining their modeling assumptions. In fact, mainstream attribution algorithms, such as Shapley values and integrated gradients, provide attribution scores *relative to* an arbitrary reference point. A similar issue arises in the PTR framework, as discussed later.

In contrast, the proposed PLA framework, which is based on our recent unpublished work (Miyaguchi 2025), completely eliminates the need for the arbitrary reference point, which we believe presents a major step forward in XAI research.

3 PRELIMINARIES

This section provides a formal definition of the attribution problem and an overview of partial trajectory regression as the baseline approach.

3.1 Problem Setting

Our main goal is to develop a method for computing the attribution score of each process in a wafer's processing route, given an observed process outcome. To formulate the attribution model, we assume a training dataset consisting of N pairs of (process outcome metric, processing route):

$$\mathscr{D} \triangleq \{(y^{(n)}, \xi^{(n)}) \mid n = 1, \ldots, N\}, \quad \xi^{(n)} = \left((\boldsymbol{x}_1^{(n)}, t_1^{(n)}), \ldots, (\boldsymbol{x}_{L^{(n)}}^{(n)}, t_{L^{(n)}}^{(n)})\right), \tag{1}$$

where N is the number of wafers and the superscript $^{(n)}$ indicates that the quantity belongs to the n-th wafer. The symbols ξ and y denote a processing route (or trajectory) and the corresponding process outcome metric, respectively. For y, we use log defect density in empirical evaluations. L denotes the number of processes in route ξ. Note that L may vary across wafers due to reworks or different route definitions. Therefore, treating ξ as a fixed-dimensional object is generally not appropriate.

We assume that process k ($k = 1, \ldots, L$) has a vector representation $\boldsymbol{x}_k \in \mathbb{R}^D$ and an associated timestamp t_k, where D is the dimensionality of the representation space. While finding a common representation space across all the processes is a nontrivial task, the kernel embedding method described in the next subsection provides a practical solution.

Because the length of each trajectory varies, the index k only refers to the position of a process within a given trajectory ξ. For instance, the 10th process in $\xi^{(1)}$ and the 10th process in $\xi^{(2)}$ may correspond to entirely different physical operations.

Formally, the task of process attribution is defined as follows:

Definition 3.1 (Process attribution). *Find a function $\alpha_k(\xi)$ that computes the responsibility score for the k-th process on a process route instance ξ ($k = 1, \ldots, L$), which is generally not included in $\mathscr{D}$.*

3.2 Process Embedding

We assume that processes are represented by numerical vectors $\{\boldsymbol{x}_k\}_{k=1}^L$. As discussed in the previous section, this is a nontrivial task due to the heterogeneity of semiconductor processes, which include etching, polishing, ion implantation, and more. Here, we adopt the kernel embedding approach proposed by (Miyaguchi, Joko, Sheraw, and Idé 2025b), assuming that high-level process attributes such as process ID and recipe ID are available from the manufacturing execution system (MES).

We first extract common attributes from the MES across different processes, such as equipment IDs, recipe IDs, tool types, photo layer IDs, route IDs, and others. We then create a synthetic 'token' for each process by concatenating these strings as follows:

$$(\text{process token}) = \texttt{eqp} \oplus \texttt{recipe} \oplus \texttt{tool\_type} \oplus \texttt{photo\_layer} \oplus \texttt{route} \oplus \cdots,$$

where $\oplus$ denotes string concatenation with a suitable separator.

Based on these string representations, we construct a dictionary of tokens. Let V_d denote the size of the vocabulary, i.e., the number of unique tokens. For this dictionary, we compute a kernel matrix $\mathsf{K} \in \mathbb{R}^{V_d \times V_d}$, where $K_{i,j}$ is the similarity between tokens i and j computed using a variant of the substring kernel (Lodhi, Saunders, Shawe-Taylor, Cristianini, and Watkins 2002; Shawe-Taylor and Cristianini 2004). Once the kernel matrix K is computed, the vector representation of token i is obtained as:

$$\boldsymbol{x}_i = (\sqrt{\lambda}_1 v_{1,i}, \ldots, \sqrt{\lambda}_k v_{k,i}, \ldots, \sqrt{\lambda}_D v_{D,i})^\top, \tag{2}$$

where λ_k is the k-th largest eigenvalue of K, and $v_{k,i}$ is the i-th element of the eigenvector corresponding to λ_k. D is a user-defined embedding dimensionality.
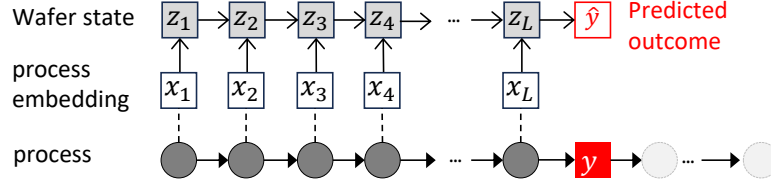
Figure 2: Assumed state-space model of PLA. Each process has a vector representation called the process embedding. The wafer state evolves based on the process embedding and the previous state.

3.3 Partial Trajectory Regression

With the process embeddings $\{\boldsymbol{x}_k\}$ now represented in a D-dimensional space, we aim to build a predictive model for defect density y as a function of the process vector sequence. This remains a nontrivial task due to the variable length of trajectories, making it an instance of *trajectory regression* (Idé and Kato 2009; Idé and Sugiyama 2011), distinct from standard regression problems.

   In this work, we employ a state-space model as illustrated in Fig. 2. A general form of the model is defined by an RNN architecture:

$$\boldsymbol{x}_k = \text{Embed}(\text{process}_k), \quad \boldsymbol{z}_k = \text{Cell}(\boldsymbol{z}_{k-1}, \boldsymbol{x}_k, t_k), \tag{3}$$

where $\boldsymbol{z}_k$ denotes the wafer state vector after applying the first k processes, and $\boldsymbol{x}_k$ is the process embedding from the previous section. t_k denotes the timestamp associated with $\boldsymbol{x}_k$. The function $\text{Cell}(\cdot)$ is typically implemented as a neural network that computes a new wafer state from the previous state and current input $(\boldsymbol{x}_k, t_k)$. While standard RNN cells such as Long Short-Term Memory (LSTM) or Gated Recurrent Unit (GRU) can be used, it has been suggested that a custom lightweight architecture

$$\boldsymbol{z}_k = \psi(t_k, t_{k-1})\boldsymbol{x}_k + \boldsymbol{z}_{k-1}, \tag{4}$$

is preferable in wafer root-cause analysis scenarios due to the lack of sufficient wafer samples covering the entire process variability (Miyaguchi, Joko, Sheraw, and Idé 2025b). Here, $\psi(\cdot, \cdot)$ is a temporal weighting function, set to $\log_{10}(1 + \cdot)$ in our experiments.

   One notable property of the PTR architecture in Fig. 2 is its ability to make predictions based on *partial trajectories*. This is achieved by training a regression model of the form:

$$f_\phi(\boldsymbol{z}_k) = \text{MLP}_\phi(\boldsymbol{z}_k), \tag{5}$$

where MLP_ϕ is a multi-layer perceptron with parameters ϕ, trained to minimize the loss:

$$L_{\text{PTR}}(\phi) \triangleq \frac{1}{2N} \sum_{n=1}^{N} \sum_{k=1}^{L^{(n)}} \frac{1}{L^{(n)}} \left(y^{(n)} - f_\phi(\boldsymbol{z}_k) \right)^2 + \eta \|\phi\|_1, \tag{6}$$

where $\|\cdot\|_1$ denotes the ℓ_1 norm and η is its regularization strength as a hyperparameter.

4   POTENTIAL LOSS ANALYSIS FRAMEWORK

This section begins by discussing the limitations of the PTR-based attribution strategy and proceeds to introduce the proposed optimized potential outcome framework.

4.1 Issues with PTR-based Attribution

The PTR framework provides a useful approach to cross-process defect attribution. As illustrated in Fig. 1, the PTR attribution method computes the attribution score of process k by comparing two potential outcomes defined by partial trajectories. Specifically, the attribution score of the k-th process in ξ is computed as:

$$\alpha_k(\xi) = f(z_k) - f(z_{k-1}), \qquad \text{(PTR)} \tag{7}$$

where $f(\cdot)$ is a regression function used to predict the process outcome, as discussed in Sec. 3.3. Unless $f(z_k)$ is an additive function over different ks, the score $\alpha_k(\xi)$ should include not only single-process effects but also multi-process correlations with upstream processes. This approach aligns with Rubin's potential outcome framework (Rubin 2005), as it simulates a counterfactual comparison with and without the target process k, similar to the situation in randomized controlled trials.

A critical question, however, is whether the PTR attribution method (7) is able to isolate the effect of process k alone, independent of all other factors. Notably, one can show that the prediction on a partial trajectory is equivalent to that of the full trajectory with the downstream process vectors set to $\mathbf{0}$. This follows from the equivalent form of Eq. (4):

$$z_k = z_0 + \sum_{i=1}^{k} \psi(t_i, t_{i-1}) x_i. \tag{8}$$

Thus, the full trajectory z_L equals a partial trajectory z_k if $x_{k+1} = \cdots = x_L = \mathbf{0}$.

Since the scale and origin of the process vectors can be arbitrary, this "zeroing out" approach may introduce biases into the attribution score. This is true, for example, when a dimension of x_i represents a binary indicator variable. While process-wise standardization might help mitigate this issue, such an approach results in unwanted dependency of the attribution score on the population means, depending on the prediction model adopted, as shown in Sec. 5 (see Fig. 4). This is the same type of issue encountered with Shapley values, whose attribution scores are defined relative to an arbitrary reference point. We next discuss how to eliminate this arbitrariness.

4.2 Defining Optimal Expected Cumulative Loss

The key idea of the proposed Potential Loss Analysis (PLA) framework is to use an optimized downstream route rather than arbitrarily zeroing out the process embeddings.

Let $F(z_k \mid x_{k+1}, \ldots, x_L)$ denote the predicted outcome starting from wafer state z_k when the wafer follows a downstream path $x_{k+1}, \ldots, x_L$ afterwards. We model this as a sequential decision-making problem, where the next process x_{k+1} is chosen based on z_k. Each action transitions the wafer state to a new state z_{k+1} and incurs a loss $C(z_{k+1})$. We define the cumulative expected loss from state z_1 as:

$$F(z_1 \mid x_1, x_2, \ldots) = \mathbb{E}\left[\sum_{t=1}^{\infty} C(z_t) \mid z \right], \tag{9}$$

where we used the numbering from one as the subscripts specify the relative positions within a trajectory. Each transition $(z, x) \to z'$ may be affected by random factors. We model such randomness using a probability distribution $p(z' \mid z, x)$ and the expectation $\mathbb{E}[\cdot]$ is taken with respect to this distribution. We assume that the transition terminates at $z_L \in \mathscr{S}_T$, where $\mathscr{S}_T$ denotes the set of terminal states. The wafer state remains in an absorbing state $s_\perp$ after the terminal state:

$$p(z' \mid z, x) = \delta(z' - s_\perp) \quad \text{for } z \in \mathscr{S}_T \text{ or } z = s_\perp, \tag{10}$$

where $\delta(\cdot)$ is the Dirac delta function.

In our problem setting, the instantaneous loss function $C(\mathbf{z})$ represents the defect density observed at the terminal state. Formally, it is given by:

$$C(\mathbf{z}) = \begin{cases} y(\mathbf{z}), & \mathbf{z} \in \mathscr{S}_T \\ 0, & \text{otherwise,} \end{cases} \tag{11}$$

where $y(\mathbf{z})$ is the observable defect density at the terminal state. From Eqs. (9) and (10), we have

$$F(\mathbf{z} \in \mathscr{S}_T \mid \mathbf{x}_1, \mathbf{x}_2, \ldots) = \mathbb{E}\left[y(\mathbf{z})\right] \quad \text{and} \quad F(\mathbf{z} = \mathbf{z}_\perp \mid \mathbf{x}_1, \mathbf{x}_2, \ldots) = \mathbb{E}\left[0 + 0 + \cdots\right] = 0. \tag{12}$$

We now define the *optimal expected cumulative loss*, which serves as a replacement for $f(\cdot)$ in Eq. (7):

$$F^*(\mathbf{z}_1) \triangleq \min_{\mathbf{x}_1, \mathbf{x}_2, \ldots} F(\mathbf{z}_1 \mid \mathbf{x}_1, \mathbf{x}_2, \ldots) = \min_{\mathbf{x}_1} \left\{ C(\mathbf{z}_1) + \sum_{\mathbf{z}_2} p(\mathbf{z}_2 \mid \mathbf{z}_1, \mathbf{x}_1) F^*(\mathbf{z}_2) \right\}. \tag{13}$$

This represents the *best possible process outcome* achievable by following an optimal process trajectory from the specified initial state. Recurrent functional equations of this form are generally referred to as the Bellman equation in control theory (Bertsekas 2012).

4.3 Deriving a Tractable Optimization Problem

The definition in Eq. (13) involves optimization over downstream trajectories. Fortunately, we can derive a tractable alternative that, perhaps unexpectedly, solves both the regression and trajectory optimization problems simultaneously.

One of the challenges with Eq. (13) is how to handle the nested min operator. We first note that

$$F^*(\mathbf{z}) \leq C(\mathbf{z}) + \sum_{\mathbf{z}'} p(\mathbf{z}' \mid \mathbf{z}, \mathbf{x}) F^*(\mathbf{z}') \tag{14}$$

holds in a transition $(\mathbf{z}, \mathbf{x}) \to \mathbf{z}'$ from any $(\mathbf{z}, \mathbf{x})$. Assuming deterministic transitions, we approximate $F^*(\mathbf{z})$ using a parametric model $F^\theta(\mathbf{z})$, where θ is a set of model parameters to be learned from the training data $\mathscr{D}$. Under the deterministic setting, the above inequality becomes:

$$F^\theta(\mathbf{z}) \leq C(\mathbf{z}) + F^\theta(\mathbf{z}'), \tag{15}$$

where $\mathbf{z}'$ is the next state resulting from applying process $\mathbf{x}$ to state $\mathbf{z}$. The tightest fit is achieved by maximizing $F^\theta(\mathbf{z})$, where the optimal θ may vary across the state $\mathbf{z}$. To find the best fit overall, we seek θ that maximizes the expected value of $F^\theta(\cdot)$ under the constraint (15):

$$\max_\theta \sum_{\mathbf{z}} \rho(\mathbf{z}) F^\theta(\mathbf{z}) \quad \text{s.t.} \quad F^\theta(\mathbf{z}) \leq C(\mathbf{z}) + F^\theta(\mathbf{z}'), \quad \forall (\mathbf{z} \to \mathbf{z}'), \tag{16}$$

where $\rho(\mathbf{z})$ denotes the empirical distribution of states in $\mathscr{D}$. This approach is based on the linear Bellman inequality formulation introduced in (De Farias and Van Roy 2003), and was discussed in the present context in (Miyaguchi 2025) for the first time.

4.4 Solving the Optimization Problem

Incorporating Eq. (12), the constraint can be rewritten as:

$$F^\theta(\mathbf{z}) \leq C(\mathbf{z}) + F^\theta(\mathbf{z}')(1 - \mathbb{I}(\mathbf{z}')), \tag{17}$$

where $\mathbb{I}(z')$ is an indicator that equals 1 if $z' = s_\perp$, and 0 otherwise. This constraint can be incorporated into the objective as the TD (time difference)-style penalty:

$$\max_\theta R(\theta \mid \mu), \quad R(\theta \mid \mu) = \sum_z \left[\mu \rho(z) F^\theta(z) - \frac{1}{2} \left\{ F^\theta(z) - C(z) - F^\theta(z')(1 - \mathbb{I}(z')) \right\}^2 \right], \quad (18)$$

where μ^{-1} is a regularization hyperparameter. Exploiting the fact that the second term in the parenthesis can be written as

$$\frac{1}{2} \{\dots\}^2 = \left\{ (1 - \mathbb{I}(z'))[F^\theta(z) - C(z) - F^\theta(z')] + \mathbb{I}(z')[F^\theta(z) - C(z)] \right\}^2 \quad (19)$$

$$= \left\{ (1 - \mathbb{I}(z'))[F^\theta(z) - F^\theta(z')] + \mathbb{I}(z')[F^\theta(z) - y(z)] \right\}^2 \quad (20)$$

and that $\mathbb{I}(z')(1 - \mathbb{I}(z')) = 0$ always holds, the final objective to be maximized is:

$$R(\theta \mid \mu) = \frac{1}{N} \sum_{n=1}^N \left[\frac{\mu}{L^{(n)}} \sum_{t=1}^{L^{(n)}} F^\theta(z_t^{(n)}) - \frac{1}{2} \left\{ y^{(n)} - F^\theta(z_{L^{(n)}}^{(n)}) \right\}^2 - \frac{1}{2} \sum_{t=1}^{L^{(n)}-1} \mu_i \left\{ F^\theta(z_{t+1}^{(n)}) - F^\theta(z_t^{(n)}) \right\}^2 \right], \quad (21)$$

where μ_i is a hyperparameter that can be adjusted according to data quality. More detailed mathematical analysis shows that the solution must satisfy $F^\theta(z_{t+1}) \geq F^\theta(z_t)$. To ensure this, we employ a positive output neural network for the difference:

$$G^\theta(z_t, z_{t+1}) \triangleq F^\theta(z_{t+1}) - F^\theta(z_t) = \text{ReLU}_\theta(z_t \oplus z_{t+1}), \quad (22)$$

where $\oplus$ denotes vector concatenation and ReLU_θ is the rectified linear unit activation function with a neural network parameter set θ. Equations (21) and (22) define the main optimization problem in the proposed PLA framework.

4.5 Process Attribution with PLA Framework

The PLA framework defined by Eqs. (21) and (22) offers several unique advantages.

First, it directly provides the attribution score:

$$\alpha_k(\xi) \triangleq G^\theta(z_{k-1}, z_k) \quad \text{(PLA)}, \quad (23)$$

which is guaranteed to be non-negative. Once we learned the model parameter θ, this formula can be used for any process route instance ξ.

Second, the attribution score is derived from the comparison of best possible downstream outcomes with and without the process of interest, eliminating the arbitrariness of zeroing out. As shown in the next section (Fig. 4), this allows providing informative signals to wafers whose defect density is close to the population mean.

Finally, the term $\left\{ y^{(n)} - F^\theta(z_{L^{(n)}}^{(n)}) \right\}^2$ in Eq. (21) ensures that $F^\theta(z)$ is a reasonable estimator of terminal loss at $z \in \mathscr{S}_T$, allowing the PLA framework to solve not only the downstream path optimization problem but also trajectory regression simultaneously.

5 EMPIRICAL EVALUATION

We applied the PLA framework to conduct root-cause analysis for a specific defect type in a state-of-the-art FEOL process. The training dataset $\mathscr{D}$ was collected at the NY CREATES Albany NanoTech fab, which consists of process histories for $N = 787$ wafers, covering hundreds of process steps, along with corresponding defect density measurements obtained at a process-limited yield (PLY) evaluation point. We implemented PLA in Python, where PyTorch 2.3.1 was used for the attribution function in Eq. (23).
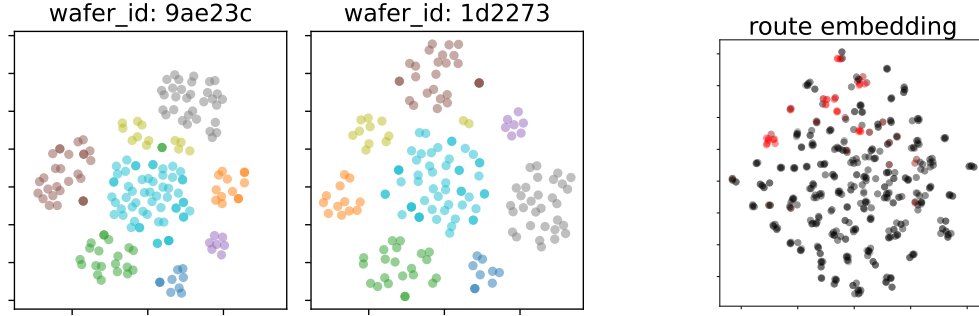
Figure 3: Process and route embeddings. **Left**: Distribution of process embeddings $\{x_k\}_{k=1}^{L}$ for two wafer instances. Each process belongs to one of eight process types (wet process, rapid thermal processing, inspection, lithography, reactive ion etching, ion implantation, furnace, chemical mechanical polishing), which are color-coded. **Right**: Distribution of route embeddings $\{z_{L^{(n)}}^{(n)}\}_{n=1}^{N}$ in the training dataset $\mathcal{D}$, with the top 10% highest-defect-density routes highlighted in red.

5.1 Process and Route Embedding

One of the critical challenges in cross-process defect attribution is the limited sample size relative to the combinatorial complexity of processing routes. Process and route embedding aims to leverage partial commonalities among routes to enable more robust prediction.

Figure 3 (a) shows the scatter plot of process embeddings for two wafer examples, with each point corresponding to a process step along the route. The visualization was generated using t-SNE as implemented in scikit-learn (Van der Maaten and Hinton 2008) with perplexity 30. Clear clustering structures are observed, suggesting that the string kernel embedding captures similarities between processing conditions effectively. Points located in close proximity typically correspond to slightly different recipe versions applied on the same tool type.

Figure 3 (b) shows route-level embeddings for all $N = 787$ wafers, where each point represents the entire process history of a wafer. The figure exhibits numerous micro-clusters, many of which may originate from lot-based processing patterns. Routes in the top 10% of defect density are highlighted in red. Notable clusters in the upper region of the plot indicate the potential presence of systematic issues.

5.2 Process Attribution

Next, we compared the proposed PLA framework with PTR. As a baseline, we used a linear model in Eq. (5), which yielded a moderate cross-validated correlation of 0.61 between predicted and ground truth log defect density. In contrast, PLA, using a two-hidden-layer neural network in Eq. (22), achieved a correlation efficient of 0.87, demonstrating a significantly improved capability for modeling the process outcomes.

Figure 4 compares cumulative attribution scores between PTR and PLA for a wafer from the held-out test set. Specifically, it plots the following quantities at each timestamp τ such that $t_k \leq \tau$:

$$\sum_{i=1}^{k} \alpha_i + f(z_0) \quad \text{(PTR)} \quad \text{or} \quad \sum_{i=1}^{k} \alpha_i + F^{\theta}(z_0) \quad \text{(PLA)}. \tag{24}$$

For PTR, the cumulative scores at the initial and terminal processes correspond to $f(z_1)$ and $f(z_L)$, respectively. Similarly, for PLA, they correspond to $F^{\theta}(z_1)$ and $F^{\theta}(z_L)$. This visualization illustrates how the predicted defect density, referred to as 'badness' in the figure, accumulates throughout the process sequence.
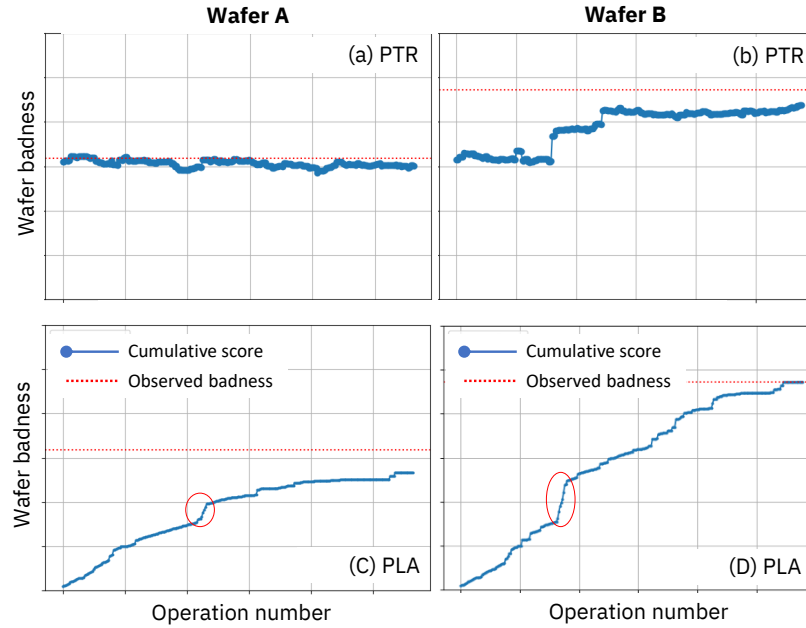
Figure 4: Comparison of PTR and PLA in terms of cumulative attribution scores. For Wafer A, the PTR-based score in (a) fails to provide informative signals, while the PLA result in (c) clearly shows how the wafer defectiveness accumulates along the route. For Wafer B, the negative attribution values in (b) complicate interpretation, whereas PLA yields consistently interpretable scores thanks to the guaranteed non-negativity.

PTR-based attribution exhibits ups and downs due to the absence of a monotonicity constraint. This is particularly problematic for wafers with near-average defect density in Fig. 4 (a), as the score is computed relative to the population mean due to its linear construction. In contrast, Figs. 4 (c) and (d) show that PLA produces more stable and interpretable attribution curves.

In Fig. 4 (c) and (d), the marked upward jumps in the attribution curves were found to correspond to unusually long wait times at specific tools, the latter of which was suspected to be the main contributor to the defect type of interest. This result highlights how PLA can effectively pinpoint problematic processes and provide actionable insights for root cause analysis.

6 CONCLUSION

We have proposed a new cross-process defect attribution framework, Potential Loss Analysis (PLA). The PLA framework addresses a fundamental challenge in sequential manufacturing: how to evaluate the responsibility of each process along the processing route.

We formalized the attribution task as a comparison of best-possible outcomes across two counterfactual partial trajectories. To the best of our knowledge, this is the first work to show that the problem can be reduced to solving a Bellman equation, enabling simultaneous regression and attribution.

Empirical evaluation on a state-of-the-art FEOL process demonstrated that PLA overcomes critical limitations of the prior partial trajectory regression approach. PLA not only improves prediction accuracy but also yields more reliable and interpretable attribution scores, making it a promising tool for data-driven, cross-process root cause analysis in semiconductor manufacturing.

ACKNOWLEDGEMENT

REFERENCES

Ali, A., T. Schnake, O. Eberle, G. Montavon, K.-R. Müller, and L. Wolf. 2022. "XAI for Transformers: Better Explanations through Conservative Propagation". In *Proceedings of the 39th International Conference on Machine Learning (ICML)*, 435–451.

Bertsekas, D. 2012. *Dynamic Programming and Optimal Control*, Volume I. Athena Scientific.

Dalla Zuanna, F., N. Gentner, and G. A. Susto. 2023. "Deep Learning-Based Sequence Modeling for Advanced Process Control in Semiconductor Manufacturing". *IFAC-PapersOnLine* 56(2):8744–8751.

De Farias, D. P., and B. Van Roy. 2003. "The Linear Programming Approach to Approximate Dynamic Programming". *Operations Research* 51(6):850–865.

Fan, S.-K. S., C.-Y. Hsu, D.-M. Tsai, M. C. Chou, C.-H. Jen, and J.-H. Tsou. 2022. "Key Feature Identification for Monitoring Wafer-to-Wafer Variation in Semiconductor Manufacturing". *IEEE Transactions on Automation Science and Engineering* 19(3):1530–1541.

Fan, S.-K. S., W.-K. Lin, and C.-H. Jen. 2022. "Data-Driven Optimization of Accessory Combinations for Final Testing Processes in Semiconductor Manufacturing". *Journal of Manufacturing Systems* 63:275–287.

Guo, P., and Y. Chen. 2024. "Enhanced Yield Prediction in Semiconductor Manufacturing: Innovative Strategies for Imbalanced Sample Management and Root Cause Analysis". In *Proceedings of the 2024 IEEE International Symposium on the Physical and Failure Analysis of Integrated Circuits (IPFA)*, 1–6.

Han, et al., S. 2023. "Deep Learning-Based Virtual Metrology in Multivariate Time Series". In *Proceedings of the 2023 IEEE International Conference on Prognostics and Health Management (ICPHM)*, 30–37.

Hsu, C.-Y., and Y.-W. Lu. 2023. "Virtual Metrology of Material Removal Rate using a One-Dimensional Convolutional Neural Network-Based Bidirectional Long Short-Term Memory Network with Attention". *Computers & Industrial Engineering* 186:109701.

Idé, T., and S. Kato. 2009. "Travel-Time Prediction using Gaussian Process Regression: A Trajectory-Based Approach". In *Proceedings of the 2009 SIAM International Conference on Data Mining*, 1185–1196.

Idé, T., and M. Sugiyama. 2011. "Trajectory Regression on Road Networks". In *Proceedings of the AAAI Conference on Artificial Intelligence*, Volume 25, 203–208.

Jebri, M. A., E. El Adel, G. Graton, M. Ouladsine, and J. Pinaton. 2016. "Virtual Metrology on Semiconductor Manufacturing Based on Just-In-Time Learning". *IFAC-PapersOnLine* 49(12):89–94.

Kim, K.-J., K.-J. Kim, C.-H. Jun, I.-G. Chong, and G.-Y. Song. 2018. "Variable Selection under Missing Values and Unlabeled Data in Semiconductor Processes". *IEEE Transactions on Semiconductor Manufacturing* 32(1):121–128.

Lee, K. B., and C. O. Kim. 2020. "Recurrent Feature-Incorporated Convolutional Neural Network for Virtual Metrology of the Chemical Mechanical Planarization Process". *Journal of Intelligent Manufacturing* 31(1):73–86.

Lee, Y., and Y. Roh. 2023. "An Expandable Yield Prediction Framework Using Explainable Artificial Intelligence for Semiconductor Manufacturing". *Applied Sciences* 13(4):2660.

Lodhi, H., C. Saunders, J. Shawe-Taylor, N. Cristianini, and C. Watkins. 2002. "Text Classification Using String Kernels". *Journal of Machine Learning Research* 2(Feb):419–444.

Lundberg, S. M., and S.-I. Lee. 2017. "A Unified Approach to Interpreting Model Predictions". In *Proceedings of the 31st International Conference on Neural Information Processing Systems (NeurIPS)*, 4768–4777.

Miyaguchi, K. 2025. "Path Learning with Trajectory Advantage Regression". *arXiv preprint arXiv:2506.19375* https://doi.org/10.48550/arXiv.2506.19375.

Miyaguchi, K., M. Joko, R. Sheraw, and T. Idé. 2025a. "Sequence-Aware Inline Measurement Attribution for Good-Bad Wafer Diagnosis". In *Proceedings of the 2025 SEMI Advanced Semiconductor Manufacturing Conference (ASMC)* https://doi.org/10.1109/ASMC64512.2025.11010308.

Miyaguchi, K., M. Joko, R. Sheraw, and T. Idé. 2025b. "Wafer Defect Root Cause Analysis Using Partial Trajectory Regression". In *2025 SEMI Advanced Semiconductor Manufacturing Conference (ASMC)* https://doi.org/10.1109/ASMC64512.2025.11010733.

Molnar, C. 2020. *Interpretable Machine Learning*. Lulu.com.

Ni, T., W. Rui, C. Zhuo, Y. Li, X. Wen, and M. Nie. 2025. "A Novel Approach to Reducing Testing Costs and Minimizing Defect Escapes Using Dynamic Neighborhood Range and Shapley Values". *ACM Transactions on Design Automation of Electronic Systems* https://doi.org/10.1145/3716826.

Ribeiro, M. T., S. Singh, and C. Guestrin. 2016. "'Why Should I Trust You?': Explaining the Predictions of Any Classifier". In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 1135–1144.

Rojat, T., R. Puget, D. Filliat, J. Del Ser, R. Gelin, and N. Díaz-Rodríguez. 2021. "Explainable Artificial Intelligence (XAI) on Timeseries Data: A Survey". *arXiv preprint arXiv:2104.00950*.

Rubin, D. B. 2005. "Causal Inference using Potential Outcomes: Design, Modeling, Decisions". *Journal of the American Statistical Association* 100(469):322–331.

Schulz, B., C. Jacobi, A. Gisbrecht, A. Evangelos, C. W. Chan, and B. P. Gan. 2022. "Graph Representation and Embedding for Semiconductor Manufacturing Fab States". In *Proceeding of the 2022 Winter Simulation Conference (WSC)*, 3382–3393 https://doi.org/10.1109/WSC57314.2022.10015297.

Selvaraju, R. R., M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra. 2020. "Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization". *International Journal of Computer Vision* 128:336–359.

Senoner, J., T. Netland, and S. Feuerriegel. 2022. "Using Explainable Artificial Intelligence to Improve Process Quality: Evidence from Semiconductor Manufacturing". *Management Science* 68(8):5704–5723.

Shawe-Taylor, J., and N. Cristianini. 2004. *Kernel Methods for Pattern Analysis*. Cambridge Univ. Press.

Susto, G. A., S. Pampuri, A. Schirru, A. Beghi, and G. De Nicolao. 2015. "Multi-Step Virtual Metrology for Semiconductor Manufacturing: A Multilevel and Regularization Methods-Based Approach". *Computers & Operations Research* 53:328–337.

Torres, J. A., I. Kissiov, M. Essam, C. Hartig, R. Gardner, K. Jantzen, *et al.* 2020. "Machine Learning Assisted New Product Setup". In *Proceeding of the 2020 31st Annual SEMI Advanced Semiconductor Manufacturing Conference (ASMC)*, 1–5 https://doi.org/10.1109/ASMC49169.2020.9185215.

Van der Maaten, L., and G. Hinton. 2008. "Visualizing Data Using *t*-SNE.". *Journal of Machine Learning Research* 9(11):2579–2605.

Wang, S., and Y. Chen. 2024. "Improved Yield Prediction and Failure Analysis in Semiconductor Manufacturing with XGBoost and Shapley Additive Explanations Models". In *Proceedings of the 2024 IEEE International Symposium on the Physical and Failure Analysis of Integrated Circuits (IPFA)*, 01–08.

Xu, F., H. Uszkoreit, Y. Du, W. Fan, D. Zhao, and J. Zhu. 2019. "Explainable AI: A Brief Survey on History, Research Areas, Approaches and Challenges". In *Proceedings of the 8th CCF international conference on Natural language processing and Chinese computing, part II 8*, 563–574.

Xu, H.-W., Q.-H. Zhang, Y.-N. Sun, Q.-L. Chen, W. Qin, Y.-L. Lv *et al.* 2024. "A Fast Ramp-up Framework for Wafer Yield Improvement in Semiconductor Manufacturing Systems". *Journal of Manufacturing Systems* 76:222–233.

Yella, J., C. Zhang, S. Petrov, Y. Huang, X. Qian, A. A. Minai *et al.* 2021. "Soft-Sensing Conformer: A Curriculum Learning-Based Convolutional Transformer". In *Proceedings of the 2021 IEEE International Conference on Big Data (Big Data)*, 1990–1998.