

IBM Research Report

Data Analytics and Stochastic Modeling in a Semiconductor Fab

Sugato Bagchi¹, Robert J. Baseman¹, Andrew Davenport¹,
Ramesh Natarajan¹, Noam Slonim², Sholom Weiss¹

¹IBM Research Division
Thomas J. Watson Research Center
P.O. Box 218
Yorktown Heights, NY 10598

²IBM Research Division
Haifa Research Laboratory
Mt. Carmel 31905
Haifa, Israel



Research Division

Almaden - Austin - Beijing - Cambridge - Haifa - India - T. J. Watson - Tokyo - Zurich

Data Analytics and Stochastic Modeling in a Semiconductor Fab

Sugato Bagchi¹, Robert J. Baseman¹, Andrew Davenport¹, Ramesh Natarajan¹,
Noam Slonim², and Sholom Weiss¹

¹IBM Thomas J. Watson Research Center, Yorktown Heights, NY, USA 10598.

²IBM Haifa Research Labs, Haifa University Campus, Haifa, Israel, 31905.

October 21, 2009

Abstract

The scale, scope and complexity of the manufacturing operations in a semiconductor fab provide some unique challenges in ensuring product quality and production efficiency. We describe various analytical techniques, based on data mining, process trace data analysis, stochastic simulation and production optimization, that have been used to address these manufacturing issues, motivated by the following two objectives. The first objective is to identify sub-optimal process conditions or tool settings, that potentially affect the process performance and product quality. The second objective is to improve the overall production efficiency through better planning and resource scheduling, in an environment where the product mix and process flow requirements are complex and constantly changing.

1 Introduction and Background

This paper describes certain analytical techniques that have been used for process-quality and production-efficiency applications in semiconductor fabs. These techniques, although broadly applicable, have been motivated, developed and applied in specific projects at the IBM 300mm fab located in East Fishkill, New York.

A semiconductor fab is a highly capital-intensive and technologically-sophisticated manufacturing facility, where the “front-end” wafer processing operations in the overall chip making process are carried out. The corresponding “back-end” operations, which involve further packaging, assembly and testing of the individual chips on these processed wafers before final product release, may take place in a separate manufacturing facility.

Specifically, a fab consists of ultra-clean rooms and mini-environments, where a series of processing steps are performed, layer by layer, on the surface of cylindrical silicon wafers of size up to 300mm in diameter, in order to produce a collection of integrated circuit (IC) devices on each wafer. The industrial operations involved in the wafer processing are numerous, and to provide

context, we list a few of these operations here, along with their oft-used or standard acronyms: cryogenic aerosol cleaning (AERP), atomic layer deposition (ALD), chemical mechanical planarization (CMP), chemical vapor deposition (CVD), furnace heating (FRN), insulator deposition (INS), ion-implantation (ION), liner deposition (LNR), photo-lithography (LTH), molecular beam epitaxy (MBE), metal layers (MTL), plasma etching (PE), plating (PLT), reactive ion etching (RIE), rapid thermal processing (RTP) and ultra-violet processing (UVP).

For each individual wafer, the overall processing in the fab typically involves hundreds of steps, whose sequence and settings will often be modified *ex tempore*, based on the results of the intermediate metrology and electrical tests carried out on the wafers between the processing steps. Once the wafer processing is completed, the final testing of the IC devices on the wafer is performed, and the wafer yield is characterized in terms of the proportion of the devices on the wafer that meet the design specifications. This wafer yield is the most important product-quality metric for evaluating the overall success of the sequence of processing steps performed on the individual wafer.

From a manufacturing operations perspective, a semiconductor fab is characterized by a high degree of complexity, with respect to short-term and long-term factors, including for example, the specific sequence of manufacturing operations for each wafer, the processing conditions for each such operation, the optimal allocation of constrained resources such as process tools and technicians in a dynamic production environment, and the pro-active and retro-active analysis of the data that is collected on a massive scale from the individual process tools and wafer test devices.

One important concern in the fab is the operational monitoring and control of the product quality. The individual wafer processing steps are themselves complex, and they also have complex interactions with other processing steps in the manufacturing sequence. Therefore, discerning the impact of the various processing conditions in each individual step on the final product yield, can be a highly uncertain exercise involving intangibles, and the required engineering understanding often emerges through significant trial-and-error and accumulated expertise on each product line. As a result, the product yields are typically much lower and more variable during the early stages of a new product line, when compared to a more mature product line.

Another important concern in the fab is the overall production efficiency. Even for a stable and well-characterized product line, the overall processing time for wafer lots is invariably much greater than the sum of the individual raw processing times for the steps in the prescribed manufacturing sequence. The factors responsible for this low production efficiency include, for example, the sub-optimal scheduling of critical equipment in a complex production environment, the overheads of excessive defect testing and rectification, and the manufacturing bottlenecks created by planned and unplanned tool maintenance events.

These two important concerns, viz., the individual product quality and the overall production efficiency, which parenthetically may often lead to competing objectives that require careful operational trade-offs, are in general, managed in the fab environment in a myriad ways; e.g., by determining optimal combinations of process tools and process settings, by flagging quality control measurements for engineering attention, by relating the operating conditions of process tools with aberrant quality or maintenance issues, and by scheduling processes, process tools, and maintenance to improve plant productivity.

The analytical projects described in this paper are motivated by the two concerns listed above, and their broad objective is to augment and inform the abilities of both the process engineers to improve and maintain product yields, and the production managers to reduce and eliminate

manufacturing bottlenecks.

The summary of this paper is as follows. Section (2) describes the Enhanced Data Mining Solution (EDMS) which supports the diagnosis of the fab processing operations and identifies manufacturing scenarios involving tools and processes that are associated with systematic variations in test performance or device yield.

Section (3) describes Trace Data Analysis Solution (TRACER), for extracting insight from massive amount of process trace data, and to develop predictive models for the local and downstream performance of process tools, often in conjunction with the test performance or device yield data.

Section (4) describes the Work-In-Progress simulator (WIPSim), which models many aspects of the fab to provide operational line support, and to assess the various options for operations management in consonance with the demand requirements, capacity constraints and other process-specific manufacturing rules.

Section (5) describes the Maintenance Scheduling Solution (MSS), which uses the results from WIPSim in Section (4) to generate maintenance schedules for the process tools in the fab, so that the planned as well as the estimated unplanned maintenance events can be carried out with minimal tool downtime and disruption of the production efficiency.

Section (6) concludes with a summary, along with our perspective on the important future research directions.

2 Enhanced Data Mining Solution (EDMS)

2.1 Motivation and Overview

In this section, we describe EDMS, which is an automated system for improving product yield and quality based on the continuously-collected manufacturing data in the fab, such as the tool usage history profile and the process and test measurements, and which *in toto* comprises of tens of thousands of features. The use of this manufacturing data is an enhancement that considerably extends the traditional product yield and quality considerations in the fab, which are typically measured in terms of final-product testing metrics, such as the power consumption or the operating frequency of the chips in each wafer or wafer lot.

A unique aspect of the EDMS methodology, is the inference of patterns in terms of binary regression rules that isolate significantly higher or lower production performance values, relative to the overall mean, for certain tools or combinations of tools used in a particular manufacturing step, followed by a subsequent filtering of these rules by knowledge-based constraints, which greatly increases the possibility that the empirically-validated rules will be interesting enough to warrant further investigation by the process engineers. The installation of this system in the IBM 300mm fab has lead to numerous opportunities for product yield and process improvement, with a significant return on investment.

Given the process complexity and the long manufacturing times for each batch of semiconductor wafers, it is not surprising that considerable effort has been made to collect and analyze manufacturing data to identify patterns and root causes for improving productivity [12, 13]. For acute tool failures, manufacturing engineers have numerous techniques to immediately determine the source of failure, but for process yield improvement, many of the opportunities are far more

subtle [14], [8], and there is great interest in mining the process data prior to final testing [1, 11]. As noted in Section 1, the fraction of the manufactured chips that will fail at the intermediate and final test stages, may be especially large during new product introduction, or during process modifications in existing product workflows. The large volume of collected data, which consists of tens of thousands of measured values for each wafer, suggests the possibility of using automated analytical techniques for identifying and extracting interesting patterns, which include for example, any unusually high or low product or process yields, or any superior or substandard performance characteristics in terms of product chip speed or power consumption.

The central theme of the EDMS approach is fault diagnosis, and the important requirement that the results of the EDMS analysis be transparent and understandable to the process engineers who maintain and monitor the production line, is achieved by using a specialized form of a binary-clause regression rule. For example, as illustrated in Figure 1, a typical binary-clause regression rule shows a condition in which there is a marked contrast the mean value of the wafers in the data sample. In Figures 1 and 2, Implanter A1 and Furnace B1 are tools; Extension Implant and Oxidation are process steps; n-ion is a measurement characteristic of chip power consumption and speed, and both low and high values of n-ion are indicative of aberrant performance. Figure 1 has only one condition, while Figure 2 has multiple conditions; however, in practice, some multiple conditions can be difficult to rationalize in terms of tool behavior, and therefore EDMS currently limits rules to a maximum of two conditions.

```
Average median-n-ion for all 292 wafers is 823.4.
-
IF (Implanter A1 is used for Extension Implant)
THEN (Average median-n-ion for 58 wafers is 801.9)
OTHERWISE (Average median-n-ion for 234 wafers is 828.7)
```

Figure 1: Typical Rule

```
Average median-n-ion for all 292 wafers is 823.4.
-
IF (Implanter A1 is used for Extension Implant)
AND (Furnace B1 is used for Oxidation)
THEN (Average median-n-ion for 43 wafers is 793.6)
OTHERWISE (Average median-n-ion for 249 wafers is 828.6)
```

Figure 2: Typical Paired Rule

The rules have several descriptors, such as the number of wafers covered, or the contrast in the target value from the overall population mean. These descriptors are further used as constraints and filters for identifying the set of acceptable or potentially actionable rules. While predictive modeling methods, such as decision trees [7], can generate an implied set of covering rules from the manufacturing data, however, the generated rules are usually inadequate for the application for a variety of reasons. For example, the rules may be too complex; they may cover too few wafers to be of practical interest; or they may just reflect the standard operating environment in the fab

(which may be transient and non-stationary, with substantial probabilistic effects). Finally, there are also significant time and personnel constraints in the fab, which limit the number of diagnostic rules that can be followed up on by the process engineers. In general, the rules that will be of greatest interest, are therefore those that cover a substantial number of wafers, and furthermore, which can be backed up and supported by a graphical display of the relevant time-series data on the candidate tool or process to facilitate the further evaluation and resolution of the rule diagnosis. For example, Figure 3 illustrates the supporting time series in support of the rule in Figure 1.

A problem may be detected when a relatively large number of wafers deviate significantly from the overall mean value. If a pattern is discovered relative to specific tools, then the induced rule may be a diagnosis of a potential problem with those tools. The resolution of this problem may then lead to opportunities for improving the overall target, such as product quality and yield. For example, the operational parameters of a specific tool, such as under-performing Implanter A1 in Figure 1, may be compared to similar tools to isolate a difference in process settings that can be adjusted, after a detailed investigation to ascertain if this will lead to clear-cut improvements.

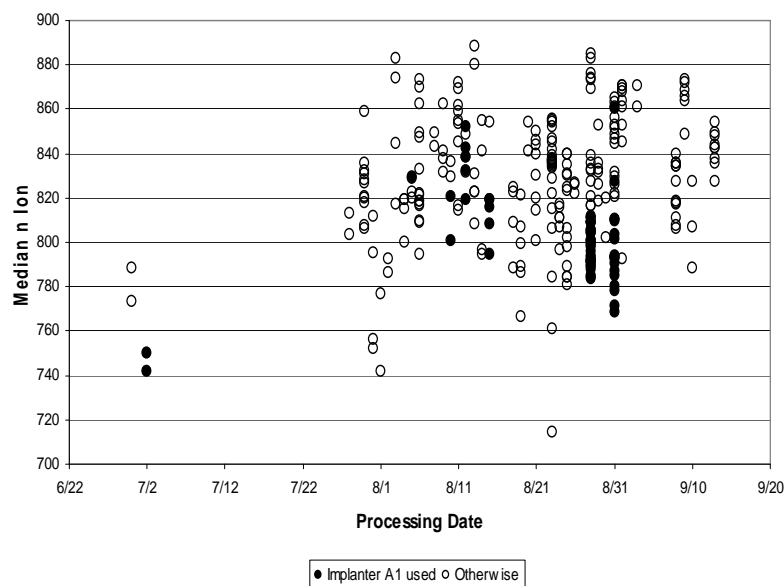


Figure 3: Time Series for Pattern in Figure 1

2.2 Analytical Methods - Binary Regression Rules

To reiterate, EDMS is based on generating binary regression rules to distinguish patterns of high or low target values, and these rules, which are limited to a specified maximum number of conditions, are then filtered by constraints on the rule coverage (e.g., minimum numbers of wafers or wafer lots in the rule), or significance (e.g., minimum allowable deviations of target values from the overall mean values). The limitation on the maximum number of rule conditions, and typically only singletons or pairs are allowed, is because of the difficulty in operationally exploiting patterns

of greater complexity. The general form of the binary regression rule is given by

$$\text{If } X \text{ and } Y \text{ are satisfied; Then } value = a; \text{ Otherwise } value = b. \quad (1)$$

A data mining method is used to induce these rules from sample data, and Figure 4 provides an overview of the specific learning method that we have used in EDMS, which is similar to regression trees in the CART algorithm [5].

1. *Grow binary regression tree to depth k (shallow tree).*
2. *Cross-validate to get best size tree.*
3. *Each path to a terminal node is a potential rule.*
4. *Generalize rules by extracting subsets of paths.*
5. *Filter rules.*

Figure 4: Overview of the learning method for rule induction

In the semiconductor fab, the opportunities associated with larger numbers of wafers are generally more significant than those associated with a few outliers. Therefore, in growing the regression tree, the absolute-deviation criterion is used in the tree splitting, rather than the usual squared-error criterion which tends to emphasize outliers. Similarly, the regression trees are restricted to relatively shallow depth, e.g., 4 levels of splits, since the various constraints on the eventual rules derived from this tree, such as the length of the rules, or the minimum numbers of wafers and wafer lots that are covered by the rules, typically will not be satisfied at the deeper levels of the tree. Tree induction methods have been extensively studied, and a standard algorithm can be used for Steps 1 and 2 in Figure 4 to produce a tree with a tested and stable performance. However, since the objective of EDMS is diagnosis rather than prediction, the resulting tree is only an intermediate structure from which binary regression rules are inferred.

Each path from the root to a leaf node of the induced tree is a potential regression rule which is comprised of the conjunction of the node split conditions on that path. However, as mentioned above, depending on the application, it is often beneficial to restrict the rules to 2 or fewer conditions. Figure 5 describes a procedure for extracting a rule R from the path traversed beginning at root node 1 and ending at node k , where k may be a terminal or non-terminal node, which potentially generating rules for every leaf and non-leaf node in the tree. However, instead of starting at the root node, the rule can also be assembled by starting at the last node on the path and gradually adding parent nodes, which is the reverse of how the path was generated, and in each case, when the target value in rule R is *close* to the value in the complete path to node k , the procedure halts. A candidate rule's *deviation* is the difference between the mean target value for all wafers in the sample (the root node) and the target value for the rule (the conclusion node), and heuristically, a rule is deemed to be close when its deviation is within 10% of the full path's deviation. When the maximum length is exceeded, the procedure halts and no rule is extracted.

In summary, the application of these procedures results in stable, empirically-tested rules, which are also compact enough to be easily understandable by the process engineers who must be convinced of the "usefulness" of the rule, in order to justify the investment of time and effort in further investigation. The preliminary set of rules obtained in this fashion are further refined by

1. Number the nodes on the path from root node 1 to last node k
2. $i = k; R = \phi; j = \text{max length}$
3. $R = \{\text{Node } i\} \text{ AND } R$
4. if (deviation(R) is within 10% of full-path deviation) Stop
5. $i = i - 1$; if ($i = 1$) Stop
6. if ($k - i > j$) { $R = \phi$; Stop}
7. Go to 3

Figure 5: Method for rule extraction and generalization (ϕ denotes the empty rule; {Node i } denotes the condition at the Node i ; the AND operator adds conditions to the rule).

a set of filters, and the most interesting rules are those that meet all the thresholds posed by the filters. Table 1 lists some of these rule filters.

<i>Filter</i>	<i>Threshold</i>
Minimum number of wafers covered	25
Minimum number of lots covered	5
Target deviation units above global mean	5
Target deviation units below global	5

Table 1: Rule filters

2.3 Cost Benefits of EDMS

From the cost-saving perspective, the value of an EDMS diagnostic rule can be measured much more objectively than in most applications. Any reductions in product yield and quality for a period of time directly translates into losses that can be quantified in terms of the total lost sales. Thus the early detection and diagnosis of a highly significant opportunity using EDMS can be evaluated in terms of the actual amount of money saved. The aggregate savings from using EDMS to date, have been of the order of many millions of dollars.

3 Trace Data Application (TRACER)

3.1 Motivation and Overview

In this section, we provide a conceptual overview of the capabilities of TRACER, which is a simple yet coherent framework for developing applications based on the off-line analysis of process trace data (PTD). A more detailed exposition of the TRACER framework with further technical details may be found in [20]. In addition, the figures in this section are actual outputs or views generated by the the TRACER framework utilities, but the details of the process/sensors, which are incidental to the discussion, are suppressed.

Modern manufacturing tools are equipped with numerous sensors that record a variety of chemical, physical, and mechanical process measurements, typically to provide feedback to the operators or the process control mechanisms on the tools. Consider, for example, a manufacturing process for a given wafer on a RIE tool consisting of multiple steps, e.g., Step 1: Over the course of 20 seconds, raise the chamber’s temperature to 200 deg.; Step 2: Keep the chamber’s temperature at 200 deg. for 30 seconds; Step 3: Over the course of 60 seconds raise the temperature to 300 deg; etc.; in this case, the PTD consists of the time series that are recorded in each sensor during each individual step.

The aggregate volume of PTD collected in this way can be quite large. For example, let N_1 denote the average number of processes conducted on a single wafer in a manufacturing day, N_2 the average number of sensors in the chambers that execute these processes, and N_3 the average number of steps in each such process. The total number of time series traces collected per wafer per manufacturing day, can be estimated as $N_1N_2N_3$, which is $O(10^3)$ or more under typical operating conditions in a fab. Furthermore, this time-series data, as a consequence of being compiled from a variety of sensors and tool processes, is also heterogeneous with disparate units and scales of measurement. Therefore, the task of identifying the trace signals that are of potential interest from an applications viewpoint is a formidable challenge for any statistical analysis framework.

The two important characteristics of PTD from an applications context are, first, this data is generally available for all wafers from all processes, and second, this data provides a window into the fundamental physico-chemical processes that the wafers undergo in the corresponding manufacturing step. These two characteristics should be contrasted with the testing and metrology data, which are typically only available for only a small fraction of the wafers, due to the cost and time associated with obtaining this data for the entire product batch. However, the complementary nature of the product measurement data suggests the potential for combining it with the PTD to obtain further insights into the tool status, tool operating characteristics, and product quality performance, as a function of the given manufacturing step.

A major challenge in analysing the PTD is its inherent stochastic nature, which is a complex consequence of multiple source factors, such as the random drifts and failures in the associated tool components; the pre-conditioning effects of the previous processes in the workflow, and previous manufacturing usage of the tool; the status of the tool in its routine maintenance cycle; the idiosyncratic behaviors with respect to particular products; the errors in setup and configuration of the control software, recipe specification and installed hardware; and any drifts, failures, or calibration errors in the sensors themselves. The identification and interpretation of these source factors from the PTD is complex, due to this large multiplicity of source factors. Furthermore, the operational significance of these individual source factors depends on the application context, and this significance can range from normal maintenance events to critical failures requiring immediate attention. Despite these inherent challenges, there are a host of applications that can greatly benefit from the analysis of the PTD, as described below.

3.2 The TRACER Framework

3.2.1 Preliminaries

Let $\mathcal{C} = \{C_1, C_2, \dots\}$ denote the set of tool chambers under consideration, with $\mathcal{C}_x = \{C_{1,x}, C_{2,x}, \dots\}$ denoting the set of sensors in chamber C_x . Further, let $\mathcal{P} = \{P_1, P_2, \dots\}$ denote the set of pro-

cesses under consideration, with each chamber executing a given subset of these processes, and let $\mathcal{P}_y = \{P_{1,y}, P_{2,y}, \dots\}$ denoting the set of steps associated with process P_y . Then, for each wafer being processed in chamber C_x via process P_y , the set of associated trace data is the Cartesian product $\mathcal{C}_x \times \mathcal{P}_y$, and each component in this Cartesian product represents a distinct set of time series measurements reported for each wafer being processed. For example, if $C_{1,x}$ represents the temperature sensor in chamber C_x , and if $P_{1,y}$ represents the first step in process P_y , then the relevant trace data in this component is the time series of temperature measurements obtained at some appropriate sampling rate from the temperature sensor during this first step.

Although the entire time series data can also be considered, for simplicity and practicality currently in the TRACER framework, each time series is converted to a single summary statistic, e.g., the time series median. Therefore, for a specified time period denoted by the index d (e.g., the first week of August 2009), the trace data associated with sensor $C_{i,x}$ and process step $P_{j,y}$ consists of a vector of these summary statistics for all wafers processed by this chamber via this process during this time period, which is termed the *Trace Data Vector* (TDV), and denoted by $V^{d,x,i,y,j}$. The size of $V^{d,x,i,y,j}$ is denoted by $N^{d,x,y}$, which is the number of wafers that are processed in the same chamber, process and time context.

Finally, consider a specific product measurement (e.g., CD thickness) denoted by the index k , that is taken after the completion of process P_y (as mentioned earlier, these measurements may only be taken over some subset $N^{d,x,y,k}$ of the processed wafers), and denote the corresponding vector of product measurements by $M^{d,x,y,k}$. For simplicity, we will assume that the vector $M^{d,x,y,k}$ has the same dimension as $V^{d,x,i,y,j}$, namely $N^{d,x,y}$ elements, and the wafers for which the k -th product measurement is not available, are represented by missing values in $M^{d,x,y,k}$.

3.2.2 Objectives and Scoring Framework

The TRACER project has the goal of developing a versatile framework to address various application objectives in PTD analysis. Originally, three general objectives were defined; improving tool stability, improving tool matching, and gaining new insights on tool operation. For example, tool stability may be identified from the TDVs, which reflect instabilities in various ways, ranging from gradual drifts to abrupt transitions in the process statistics, as shown in Figure 6. Similarly, tool matching may be identified by comparing the reported TDV on a given tool with the TDVs for nominally identical tools executing the same process, which again may be reflected in various ways, as shown in Figure 7. Finally, gaining new insights on tool operation can be carried out by examining the dependencies between TDVs and the associated product quality measurements, from a similar perspective.

The core observation in the development of the TRACER framework is that at the abstract level, all three objectives mentioned above (as well as some other objectives not discussed here) involve scoring each TDV in terms of its importance in the context of the given objective. For example, in the context of improving tool stability, a scoring function, $S_1(V^{d,x,i,y,j})$ should quantify the stability of chamber C_x for process P_y during the time period d . Similarly, for gaining insights regarding tool operation, the scoring function $S_2(V^{d,x,i,y,j}, M^{d,x,y,k})$ should quantify the level of dependency between $V^{d,x,i,y,j}$ and the associated product measurements $M^{d,x,y,k}$.

Therefore, this notion of a scoring function on the TDV is simple yet general enough to support a variety of applications, although, two concerns should be emphasized. First, this approach is focused on univariate statistical analysis; namely, each TDV is analyzed independently, and in-

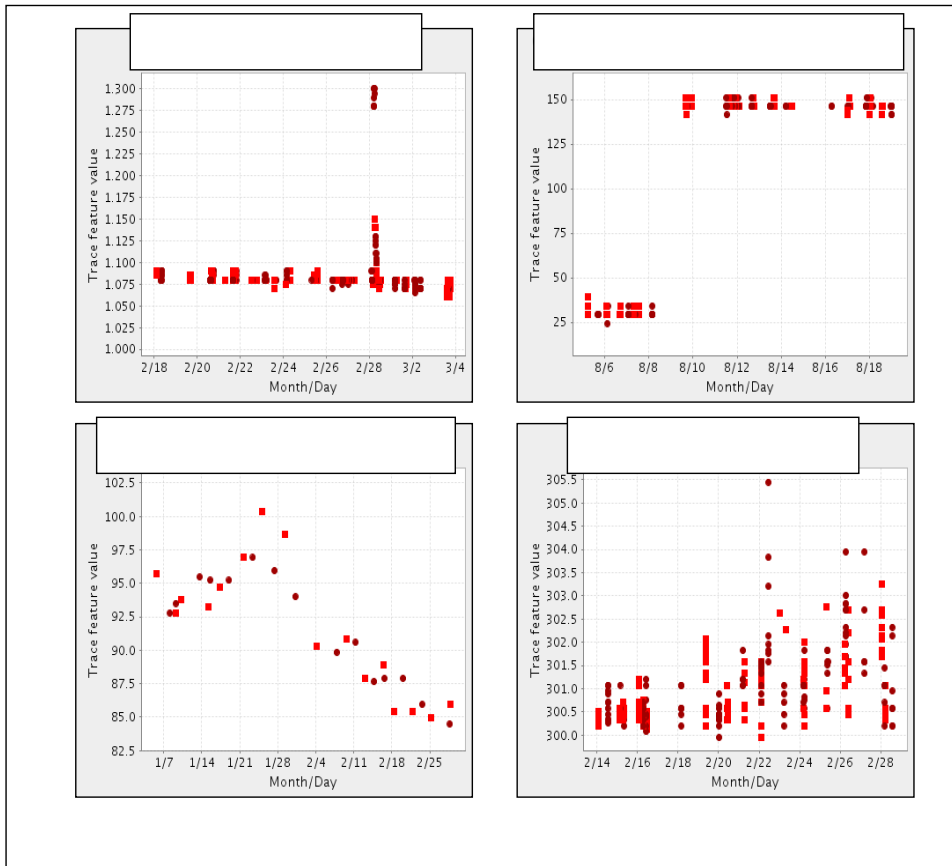


Figure 6: TDVs may reflect instabilities in various ways including gradual drifts and abrupt transitions.

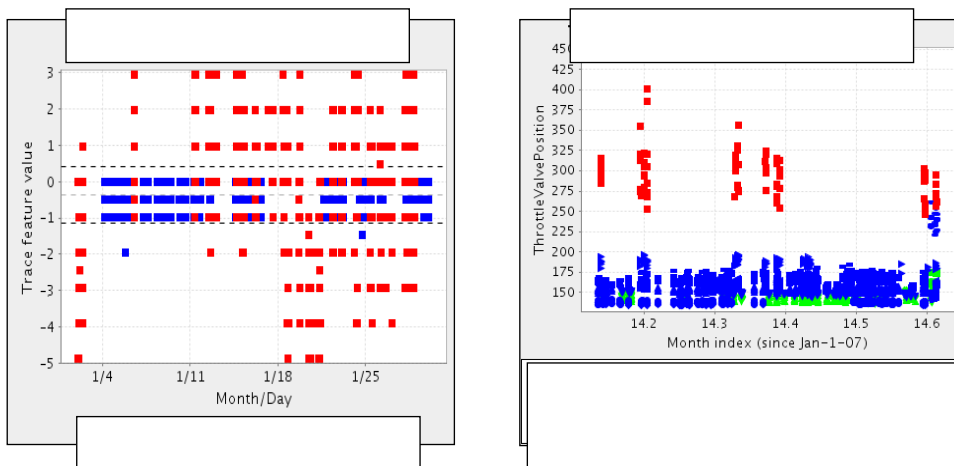


Figure 7: TDVs may reveal various types of tool mismatches between nominally identical tools in terms of mean and variance.

teractions effects involving multiple TDVs are not considered. Second, even with this limitation of univariate analysis, if the chamber C_x is equipped with 50 different sensors and the process P_y involves 20 different steps, then 1000 different TDVs must be scored for this chamber-process pair. (This assumes that we use a single summary statistic per trace time series data, e.g., the median. However, it is often important to consider multiple summary statistics, e.g., the mean and the variance of the time series data, which will add another constant factor to the number of TDVs being analyzed.) The supervising process engineer, who may be responsible for 100 such chamber-process pairs, may however only be able to manually review a tiny fraction of the 100,000 TDVs in this simple example.

The key ideas in the TRACER methodology are therefore designed to address these two concerns outlined above.

3.2.3 TRACER Chamber/Process Smart Heat Maps

A primary observation in the development of the TRACER framework was that the overall status of chamber C_x executing process P_y can be communicated via a single heat-map table where the rows represent C_x sensors and the columns represent P_y steps, as shown in Figure 8. Each cell in this heat map is color coded to indicate the level of interest that should be associated with the corresponding combination of a chamber's sensor and process-step. The actual color is determined in terms of the underlying analytics, which are tailored to the particular objective under consideration. For example, if the underlying analytics are designed to detect unstable signals the corresponding heat map will highlight TDVs that reflect presumed tool stability issues. Alternatively, if the underlying analytics are designed to reveal TDV mismatches, then the corresponding heat map will highlight presumed issues associated with tool matching.

Assuming the effectiveness of the analytics layer, the TRACER chamber/process report then enables the supervising engineer to rapidly review thousands of underlying TDVs in a single view, with the operationally significant signals clearly highlighted. By clicking through the cells in the heat map, the associated detail reports for the relevant TDVs can be accessed, as illustrated in Figure 9.

Although the TRACER framework typically uses only univariate analysis (with each TDV being analyzed independently) the heat-map representation implicitly reveals some of the data interactions. For example, if the detected problem is related to a particular process step, the relevant heat-map column is highlighted, while if a problem is associated with particular set of sensors, the relevant heat-map rows are highlighted. In addition, each detail report contains links to other detail reports related to the same process step, in case the two corresponding TDVs were found to have a statistically significant dependency. Finally, the ordering of the rows and columns in the heat maps can reflect the use of cluster analysis and/or domain knowledge.

The perspective in this heat-map report is from the chamber/process point of view, not from the wafer point of view. However, an immediate concern is that statistically significant signals highlighted in this heat-map view may not necessarily be of operational significance. For example, an unstable TDV detected by this framework may either represent significant tool stability issues that require immediate attention, or some normal behavior related to a recent maintenance event. Thus, it seems crucial to provide some means of distinguishing, at least approximately, between the statistically significant signals that are also operationally significant, and those that are not. This classification is carried in the TRACER framework as described in Section 3.2.4 below.

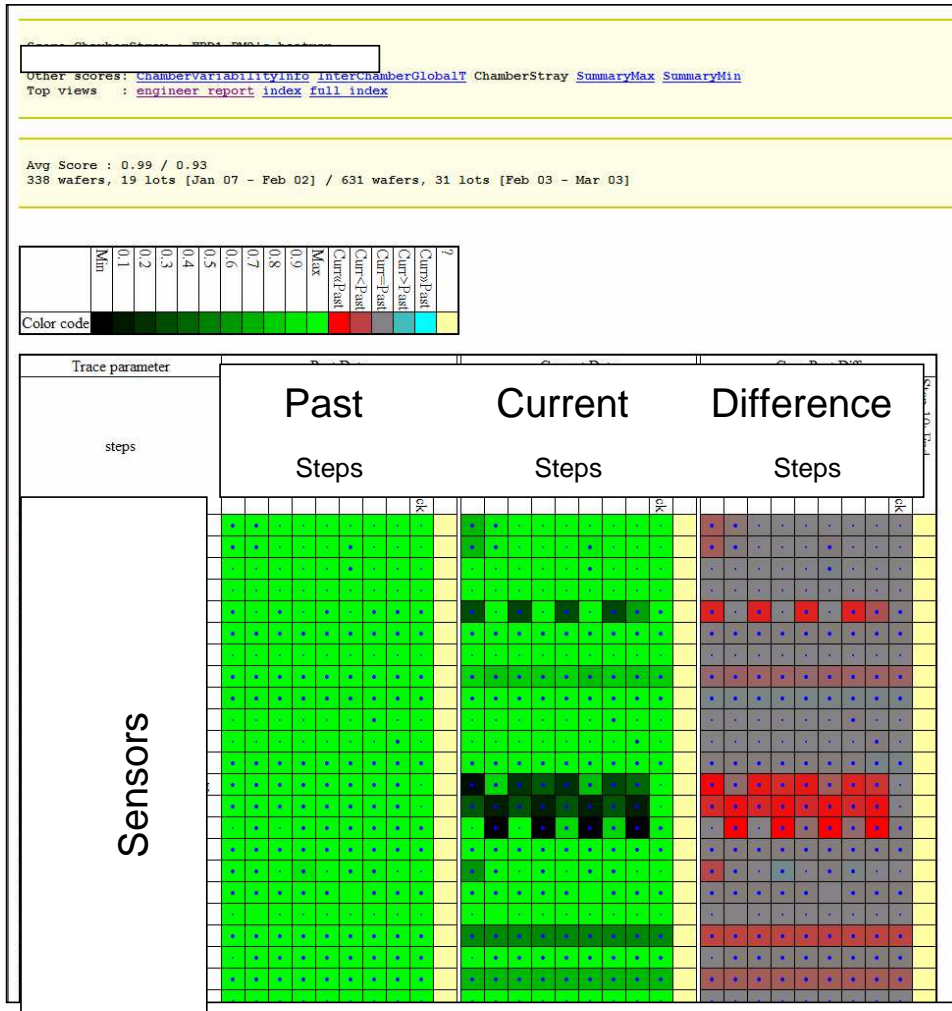


Figure 8: The TRACER chamber/process heat-map report, in which rows represent chamber sensors while columns represent process steps. Each cell is color-coded according to the underlying analytic score (scaled to a finite range, typically 0 to 1). The underlying score determines the tool issues highlighted in this report – tool stability issues, tool matching issues, relationship of trace data to performance data, or any other meaningful objective. The TRACER analysis is carried out over two time periods – “current data” versus “past data” and three heat maps are generated: one for the past data, one for the current data, and a third for the difference between the two. Thus, the TRACER difference heat map is designed to highlight TDVs with some statistically abnormal behaviors that are only observed in the current data, and therefore more likely to reflect ongoing problems with tool operation.

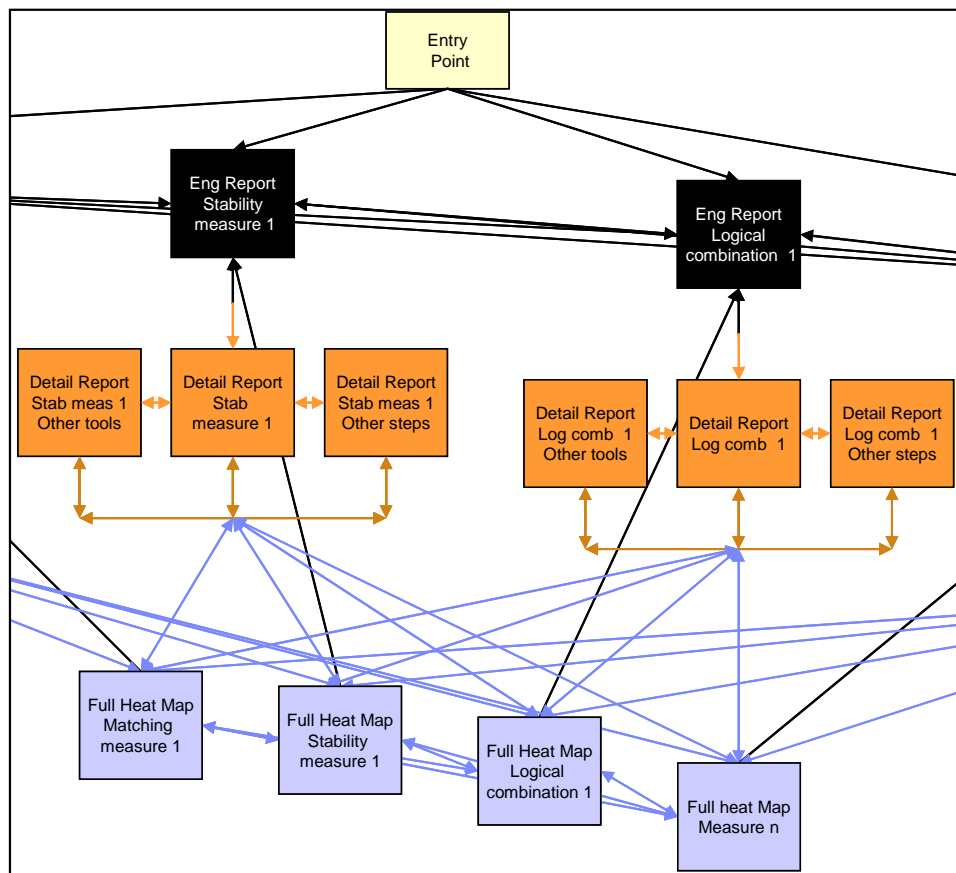


Figure 9: By clicking on a particular cell in the heat map, the detail report of the underlying PTD is obtained. Each detail report also contains links to other detail reports in the same step, if the dependency between the respective TDVs is statistically significant.

3.2.4 The TRACER difference heat map

One possible approach for detecting operationally-significant signals is to compare the examined PTD to the PTD under ideal manufacturing conditions - the latter is often termed the “golden” PTD in the literature [19]. However, such a characterization is non-trivial and time-consuming, since it is required for each and every process.

In the TRACER framework, an alternative approach was taken, which is based on the notion of *dynamic reference data*. Specifically, the TRACER analysis is carried out over two time periods, usually in succession; e.g., the data collected over the last two weeks (“current data”) versus data collected over the two weeks preceding that (“past data”). Thus, for each chamber/process pair, three heat maps are generated: one for the past data, one for the current data, and a third for the difference between the two, as shown in Figure 8. Specifically, let $S_{past}(V^{d,x,i,y,j})$, $S_{curr}(V^{d,x,i,y,j})$ respectively denote the scores associated with the i, j cell in the past and current heat maps, and assume for simplicity, that the relevant scores are in the range 0 to 1. Then, the score of the corresponding cell in the difference heat map is simply given by $[S_{curr}(V^{d,x,i,y,j}) - S_{past}(V^{d,x,i,y,j}) + 1] / 2$.

As a result of its construction, the TRACER difference heat map is designed to highlight TDVs with some statistically abnormal behaviors that are only observed in the current data, and which may reflect problems with tool operation. For example, if the trace data associated with some of the sensors are constantly drifting, this phenomenon will be detected by the TRACER analysis in both the past and the current data, hence the corresponding cells in the difference heat map will not be highlighted. However, if some abnormal trace behavior is observed solely in the current data the corresponding cell(s) in the difference heat map will be highlighted, as presumably this trace behavior reflects a truly novel event of operational significance.

3.2.5 The TRACER hierarchy of linked heat maps

While the chamber/process heat map report mentioned above can summarize large volumes of data through a single view, the sheer volume of the trace data collected in the fab requires even more summarization and filtering. For example, a process engineer may be responsible over $O(100)$ chamber/process pairs, and it is unrealistic to expect this individual to review all the corresponding TRACER heat maps.

To address this issue, the TRACER output was designed as a hierarchy of linked reports, composed at various levels of granularity, that exploit the same heat map notion at all levels to swiftly draw the attention of the end user to the most important signals. One example of such a TRACER high level report is a heat map in which rows represent different processes while columns represent chambers. Hence, highlighted cells indicate a potential problem in particular chamber/process pairs, and clicking over a specific cell will reveal the relevant underlying chamber/process heat map report, from which the user may click over any highlighted cells to the potentially most-relevant detail reports. In this high-level report as well, the heat map representation provides the results in context, so that for example, a problem associated with a given chamber will probably be reflected by a highlighted column; alternatively, a problem associated with a particular recipe will be reflected through a highlighted row. Such a high-level heat map report may cover up to $O(10^6)$ TDVs in a single view, thereby allowing the end user to rapidly review and focus on a handful of signals that are of potential operational significance.

3.2.6 Summary

The TRACER framework is highly modular and versatile, and new scoring functions can be easily integrated into it. A major advantage of this framework is that it provides a common platform for multiple application objectives, such as tool stability, tool matching and tool operating insight.

The TRACER results are always presented in the context of the end-user application objectives. For example, in the basic heat map, the context consists of all the other sensors and process steps involved in the Cartesian product $\mathcal{C}_x \times \mathcal{P}_y$. In the difference heat maps, the context is provided by the previous related data.

In summary, the TRACER framework provides application views and reports that are ideally suited for large-volume PTD, since this data can be covered by a single, high-level view from which the signals of operational interest can be isolated and rapidly reviewed.

4 Work-In-Progress Simulator (WIPSim)

4.1 Motivation and Overview

A semiconductor fab is a complex manufacturing environment, in which, at any given time, there are hundreds of product routes, with each route involving thousands of process steps and hundreds of tools. This complexity, along with numerous other operating factors, is responsible for a high level of variability in the work-in-progress (WIP).

For example, the wafer processing often requires repeated sequences of similar processes, so that the WIP is re-entrant over the same set of tools, which is known to result in turbulence [16]. These instabilities are further compounded by unexpected changes in the tool supply and demand capacities. For example, the tool-supply availability is subject to quality-control measures on specific wafers, that can automatically inhibit certain tools from performing these process recipes. Similarly, the tool-demand availability can change when wafers are re-routed away from their planned processing routes, into branches and re-works.

There are additional sources of complexity in fabs that simultaneously cover multiple chip manufacturing technologies, ranging from technologies under development to those in high volume production. In order to reduce capital costs and shorten the transition times from development to production, these different chip manufacturing technologies will often share the same tools. The higher uncertainty and the greater requirement for manual intervention on the development routes, will often influence the relatively more-stable production routes, leading to greater variability on those high-volume routes as well. As a result, the fab is never at a “steady state” in terms of WIP throughput. For example, Figure 10 shows histograms of bottleneck process centers in the fab. Over a three month period, it shows that over half of the bottleneck process centers are temporary, lasting for two days or less. The throughput on any given day depends on which process centers are going to be the bottleneck for that day. An effect of shifting bottlenecks is the high variability of the average daily WIP at key process centers as shown in Figure 11. This implies that the productivity impact of tool availability in these process centers will vary greatly from day to day.

This high variability which is intrinsic to the nature of fab operations creates major challenges in terms of meeting customer commitments, since even the stable, high-volume products are processed in a dynamic, as opposed to a steady-state manner, through their product route. This is illustrated in Figure 12. This shows the distribution of wafers along a particular product route

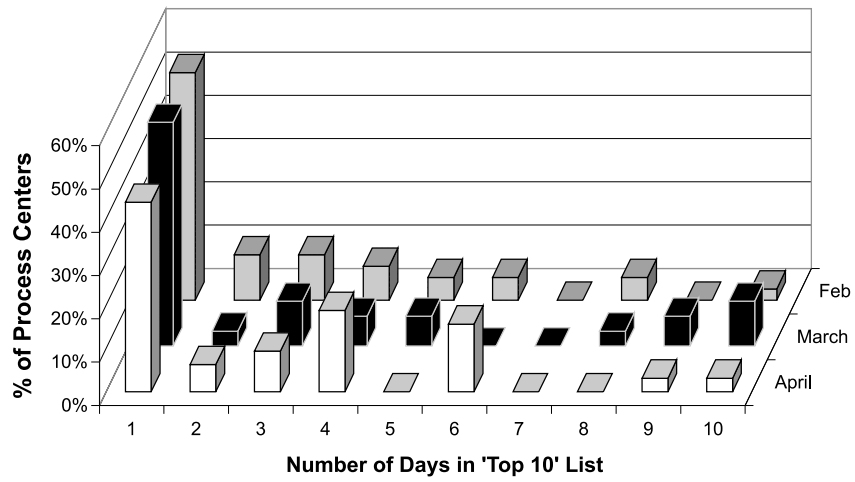


Figure 10: Histograms of Bottleneck Process Centers

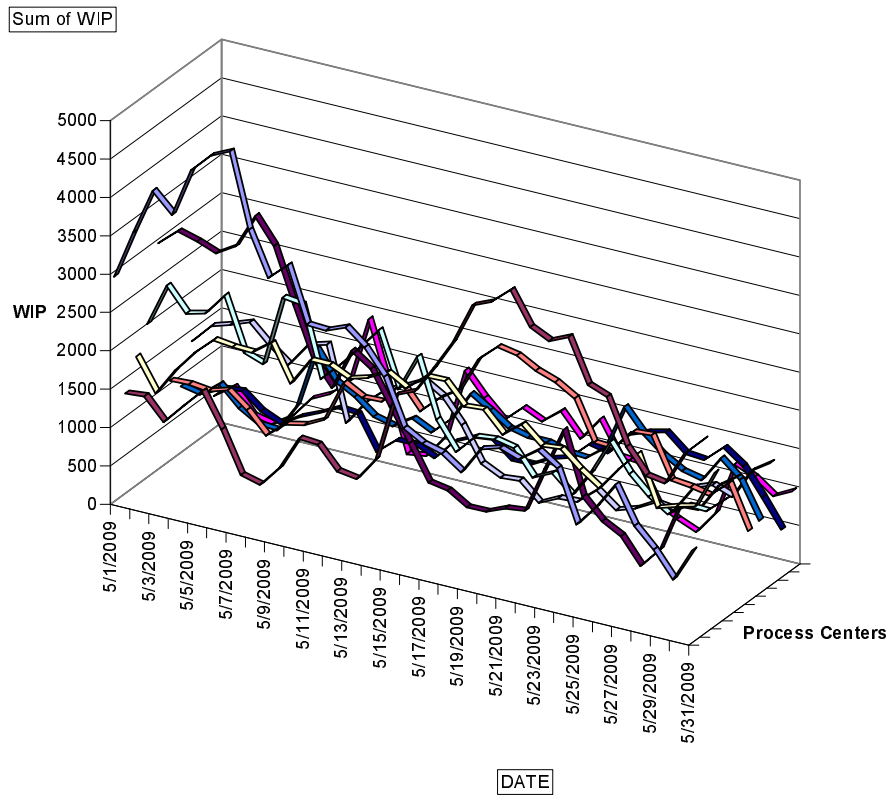


Figure 11: Average daily WIP in the top 10 Process Centers over one month

(shown along the ordinate in terms of fraction of operations completed) over a one month period (abscissa). The number of wafers is denoted by the shading (where the darkest shading denotes 1200 wafers). This figure allows the visualization of WIP “bubbles” and “holes”. Bubbles imply the presence of processing bottlenecks impeding the throughput of WIP while holes imply lack of WIP leading to idling of expensive tools. The figure shows how bubbles and holes appear and dissipate over days or even sometimes weeks, resulting in the intermittent production of the finished wafers.

Fab production managers are therefore faced with difficult choices for short-term operational decisions, such as whether to speed up a given product route, or whether to take certain tools offline for maintenance. The complexity of individual product routes, and the interactions between these routes, increases the difficulty of evaluating the effectiveness of operational decisions. Therefore, even when the direct effect of an operational decision is not in doubt, there is always uncertainty over unintended consequences, since for example, speeding up the processing on one product route may negatively impact all other product routes that share common tools with it.

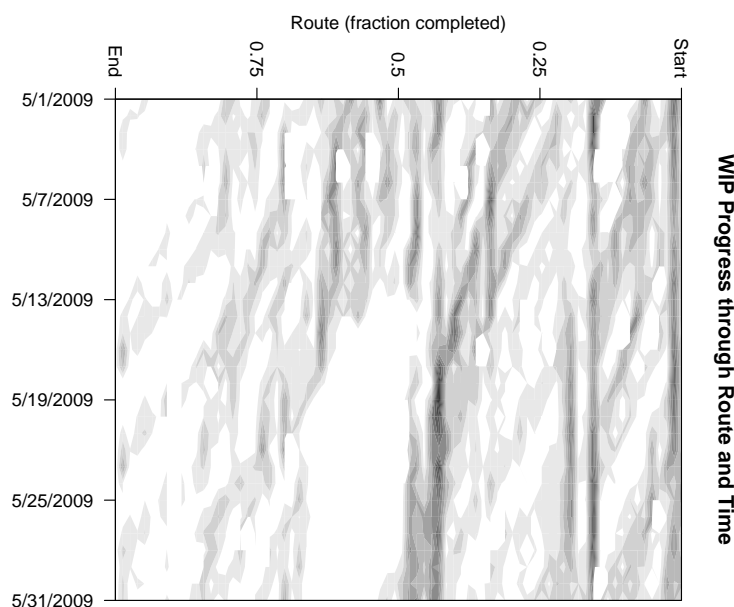


Figure 12: Flow of WIP, in terms of number of wafers, over a one-month period (abscissa) through a product route progression (from top to bottom along the ordinate). The darkest shade denotes 1200 wafers and white denotes 0 wafers.

Although fab simulation models have a long history [17, 6, 18, 10], those models are typically used to study fab-wide strategic issues over a steady-state, long-time horizon, such as for capacity planning, or for operational scheduling in a small section of the fab (such as a cluster of similar tools).

We have therefore developed a specialized simulation model for operational decision support in the fab [3]. In contrast to previous work, our modeling intent is to cover the entire fab, so as to capture the impact of any operational decisions across all product routes and process centers, and to predict operational metrics over very short time horizons of the order of hours and days.

For such a simulation model to be useful in decision-support applications, the following desiderata must be satisfied. First, the model should cover all the WIP and tools in the fab at the appropriate fine level of granularity, so that the model outputs have operational value. Second, the simulation should be based on a current snapshot of the fab state, without requiring an extended warm-up period, due to the short simulation horizon. Third, the statistical estimation of the large number of model parameters, which is based on the data in historical event logs, should be automated. Fourth and finally, the model creation and maintenance efforts should be minimal, so that frequent simulations can be performed, with the model outputs can be integrated into the planning and decision-making, perhaps even on a daily assessment cycle.

The proposed simulation model incorporates the following factors that are responsible for most of the WIP variability in the fab.

The first factor, which is most critical, is the processing time at a given tool, which is a function of the process recipe, the number of wafers, the tool setup requirements, and the number of parallel load ports and chambers in the tool. The historic data in the wafer processing logs for each tool is analysed to obtain the regression parameters for these dependencies, in order to estimate the Expected lot-processing time.

The second factor is the tool downtime, and for each tool, the available time and downtime are modeled as empirical distributions by analysing their historical event states. These distributions are often bi-modal, reflecting the presence of both short-term and long-term outages.

The third factor is the wafer-lot sampling in the product routes, since a significant fraction of the product route comprises of non-mandatory test and measurement operations. The decision to perform a given non-mandatory operation on a wafer lot depends on various rules associated with the product maturity cycle and recent yield levels. Therefore, the historic state data for the wafer lot is used to estimate the lot sampling probability, as well as the number of wafers per lot that will be measured or tested in these non-mandatory operations.

The fourth factor is the need to account for the holds, which may appear for various reasons, in the flow of WIP in the product routes. These holds, which are typically of variable duration, often require manual intervention on the product routes. The hold frequency and duration are obtained as empirical distributions from the historic state data on wafer lots.

In order to ensure that the simulation model tracks the changing conditions in the fab, the various parameters associated with the various factor models mentioned above are recomputed weekly, and depending on the specific factor model, up to 12 weeks of historical data may be used for the statistical estimation.

The WIPSim model has been deployed in the 300mm IBM fab for over a year, in a range of decision support activities [2], which include the following.

First, it provides daily projections of the incoming and average WIP at each of over 200 process centers in the fab, which is used to plan the tool maintenance activities, and the assignment of constrained labor to the maintenance of critical tools. In addition, longer term projections over a period of 1 to 3 months have been used to make tool-idling decisions that save millions of dollars in maintenance costs.

Second, it has been used to improve the productivity of specific product routes, with minimal impact on other key product routes, by allowing fab managers to evaluate the effect of adjusting various parameters, such as the daily production targets, and the dispatching priorities. Since the impact on the other production routes is affected by the day-to-day variability of WIP position in the fab, the model simulations provide the ability to evaluate and select the most effective option

from the alternatives.

Third, it has been used to evaluate the impact of production rules, that are invoked at hourly or daily time intervals, on the overall fab productivity. This feature has been used by fab managers to modify production rules, to automate decision-making, and to re-organize product groups, so that the key product routes have the resources for the best productivity.

These examples, given above, specifically illustrate the two broad classes of operational decision-support applications that are supported by the WIPSim methodology.

The first class of decision-support applications rely on making use of the WIPSim results for the projected WIP and process-center throughput based on the current observed state of the fab. A good example, described in Section 5, is a novel maintenance-scheduling application in which WIPSim is used to estimate the WIP profiles. One challenge in developing this class of decision-support applications, is the need to incorporate the gradual decrease in the accuracy of the WIPSim projections with increasing time horizon. For example, a coarsening of the decision time-scale with increasing time, can reduce the complexity of the decision evaluation, with the assumption that the WIPSim model will be run again in the future, when more accurate predictions are available.

The second class of decision-support applications rely on making use of the WIPSim results for the fab performance over some fixed period of time period. In order to increase the robustness of decisions that are based on the WIPSim results, multiple replications can be run from a single initial state of the fab, in addition to using replications from multiple start dates to obtain a variety of initial states. The impact of the reduction in variability in this way, depends on the specific metric that is being predicted, and for example, the throughput metric can be predicted more reliably for high-volume production wafers compared to low-volume development wafers, particularly since the latter are also subject to many more manual interventions and re-routing. The challenge in this class of decision-support applications is the need to obtain robust algorithms, which take into account the various aspects of variability in the product routes, in the simulation results.

5 Maintenance Scheduling Solution (MSS)

5.1 Overview

In this section, we consider the problem of generating optimal schedules for tool maintenance in the semiconductor fab, which is challenging because of the high degree of uncertainty in the operational-level details, such as the tool availability, product yield, and processing times. In order to take into account these operational-level details, we have developed a novel integrated approach, termed MSS, in which simulation results are used to estimate the Expected work-in-process (WIP) in the fab, and in which the scheduling optimization is carried out using a variety of techniques including goal programming, constraint programming and mixed-integer programming (the detailed technical aspects of the MSS formulation are described in [9]).

In a capital-intensive manufacturing facility, such as a semiconductor fab, the scheduling of the maintenance events, such as cleaning, calibration and safety checks on the individual tools, is critical for the following reasons. First, these maintenance events are expensive, and should be performed either according to the equipment needs, or according to the recommended maintenance schedule of the original-equipment manufacturer (OEM). Second, the recommended OEM maintenance schedule should not be compromised, in order to avoid potential sub-optimal functioning

or even malfunction of the tool, which would precipitate an even-more costly unplanned maintenance event. Third, any tool that is in maintenance, either partly or entirely, should not impact any critical production requirements. Fourth and finally, the maintenance of certain tools will require coverage from appropriately-qualified technicians, who are a scarce and expensive resource.

5.2 Problem formulation

5.2.1 Inputs and Primary Constraints

There are three different types of maintenance events in the fab that need to be considered: (a) regular preventive events, according to the recommended OEM schedule (e.g., every six months); (b) trigger events, based on the tool reaching a certain state (e.g., after it has processed certain number of wafers); and, (c) unplanned or unforeseen events, which leave no choice but to require the tool be taken down for repair.

The MSS system is primarily concerned with the first two types of events mentioned above, and unplanned events are considered only indirectly, as described below. However, in the future, we expect to incorporate the mean-time-before-failure (MTBF) statistics of various tools, to directly ensure a reserve capacity of maintenance technicians for periods when the likelihood of unplanned events is high.

For each tool maintenance event, we are given the following - a release date, a due date, a processing time, the relevant tool or the part of the tool under consideration, and the average number of technicians required to service the event. The scope of the maintenance event, whether limited to a part of the tool (e.g., a single chamber of a lithography tool), or applicable to the entire tool, is important, since it impacts the tool usage in the production operations, as well as the possibility of simultaneously scheduling multiple maintenance events on the tool.

The tools in a semiconductor fab can be partitioned into *toolsets*, with each toolset consisting of the set of tools with the same function (e.g., lithography) and manufacturing vendor. Thus, individual toolsets will have the same set of qualified maintenance technicians (however, very infrequently, certain technicians may be certified on multiple toolsets).

For each toolset, there is a timetable giving the technician capacity during each time period (or maintenance shift), and this capacity is the primary resource constraint for generating MSS schedules (although technician requirements that exceed capacity can occasionally be sub-contracted to external vendors, this is considered to be an expensive and avoidable alternative).

5.2.2 Objectives

In practice, the set of feasible schedules satisfying the technician capacity constraints are straightforward to obtain, and the real challenge is to manage the numerous complex production and maintenance objectives, listed below.

Resource Leveling The first objective is the uniform utilization of the available technicians for a given toolset, since this also increases the likelihood of having some surplus capacity to handle unplanned maintenance events in that shift (see Figure 13 for examples of good and bad utilization). This objective may also be stated as: *minimize the number of technicians that are utilized through-*

out the entire schedule of pending maintenance events, or alternatively: minimize the maximum number of technicians that are active in any given shift.

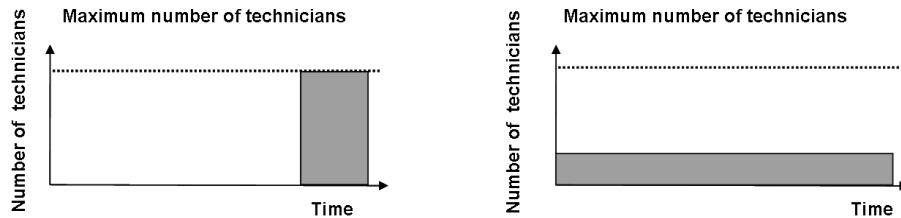


Figure 13: Bad (left) and good (right) examples of maintenance technician utilization

Production Disruption The second objective is to minimize the production disruption that results from removing tools for maintenance. As described earlier, the work-in-progress (WIP) at any given shift, consists of wafers are either being processed or waiting in a queue to be processed, by a given tool. During a maintenance event, when all production is stopped on this tool, in addition to the delays in its own queue, this tool downtime will also starve downstream processes, leading to a cascade of tool under-utilization. Ideally, these production disruptions should be minimized by scheduling the relevant maintenance events during shifts when there is little or no WIP on the given tool. Figure 14 illustrates a maintenance schedule for a set of tools in a toolset, in which the WIP levels for each tool and time period are plotted in the background.

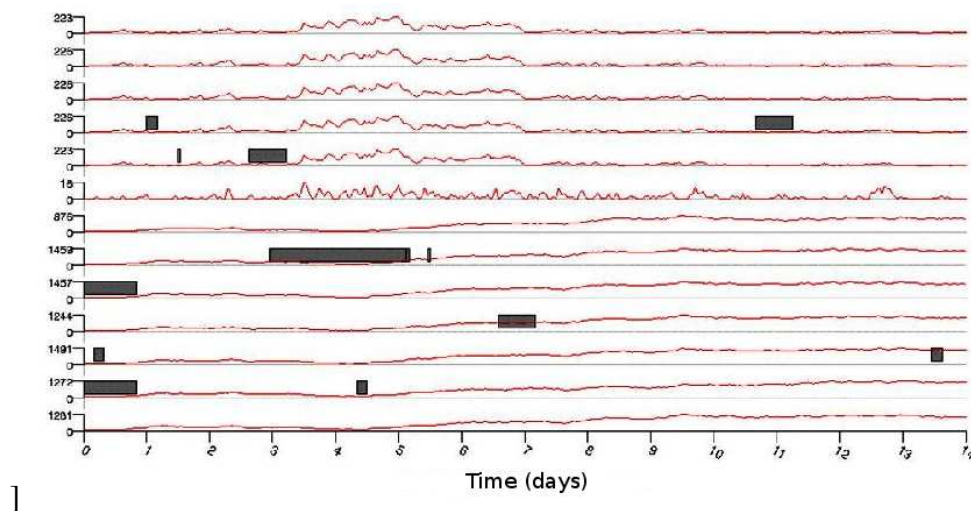


Figure 14: The maintenance schedule for a toolset: each row is a Gantt chart for a single constituent tool, and the lines denotes the corresponding projected WIP over each time period in the schedule, and the boxes denote the start and duration of the corresponding maintenance events.

The objective for minimizing the production disruption can be formulated in the following way: Given w_{mt} , the level of WIP on tool m in time period t , and maintenance event k on tool m which starts in time period s and finishes in time period e , the WIP disruption of this event is $\sum_{t \in [s,e]} w_{mt}$. We wish to minimize the total WIP disruption for all scheduled maintenance events.

However, the difficulty with using this objective function is the uncertainty in the WIP levels for each tool over the scheduling horizon, and as discussed in Section 4.1, this uncertainty is an intrinsic issue in semiconductor fabs. The detailed production scheduling in fabs is usually done using dispatch rules which are applied whenever a tool becomes available for processing, and there is no long-term production schedule that can be used to determine the WIP levels for each tool. Therefore, we use the Expected WIP for each tool and time period obtained from WIPSim (Section 4; see also [3]) as follows. We use 20 replications of WIPSim for the whole scheduling horizon, based on a division of the horizon into one hour time buckets. From the WIPSim output, we obtain the expected WIP level for each tool during each time bucket, and these estimates are usually quite accurate in the short term (1-3 days), although much less so over the long term (up to two weeks). Consequently, the maintenance scheduling is performed every day, with rescheduling taking place over several days, based on the most recent updates to the estimates for the Expected WIP.

Earliness/Tardiness The third objective is to minimize the long-term costs of periodic maintenance events. While there is some flexibility in determining when a maintenance event j can be performed in the schedule, as specified by the release and due dates, for periodic maintenance events, the elapsed interval between the completion time e_i of one event i and the start time s_j of the following event j on a tool should not exceed the recommended elapsed duration $D_{i,j}$ specified by the OEM manufacturer. This leads to earliness/tardiness costs for scheduling periodic maintenance events, where the tardiness cost comes from scheduling an event j to start at some time s_j such that $s_j > e_i + D_{i,j}$, and the earliness cost comes from scheduling an event j to start at some time s_j such that $s_j < e_i + D_{i,j}$. The due date d_j of an event j is calculated such $d_j = e_i + D_{i,j}$, so that scheduling an event precisely at the due date incurs zero earliness/tardiness costs.

Therefore, given an earliness penalty α_j and a tardiness penalty β_j for each event j , and given the completion time C_j of event j in a feasible schedule, the earliness-tardiness cost $\eta(C_j)$ for this event can be computed as $\eta(C_j) = \max(\alpha_j(d_j - C_j), \beta_j(C_j - d_j))$.

5.2.3 Side Constraints

In addition to the basic problem formulation presented above, there are a number of “side constraints” in MSS that arise from imposing other user preferences on the maintenance schedule, which MSS attempts to satisfy whenever possible, irrespective of their impact on the problem objective. For brevity, we limit the discussion below to a couple of possible side constraints.

Follow-up Maintenance Constraint It has been observed that whenever a maintenance event is completed on a given tool, there is a good likelihood of a follow-up maintenance event within six hours on the same tool. Ideally, this follow-up should be performed by the same technician who is responsible for the original maintenance event, and therefore, it is desirable to reserve this technician for this potential follow-up event. The side constraint can therefore be stated as: *maximise the number of maintenance events that satisfy this user preference.*

Separation Constraint For operational reasons, any two maintenance events i and j on a given tool, should preferably be scheduled in one of the following ways, in some decreasing order of

preference; (a) i and j separated by at least 24 hours; (b) i and j separated by at least 12 hours; (c) i and j scheduled continuously, so there is no gap between the end of i and the start of j or vice-versa. By assigning weights w_i to each of these preferences (a)-(c), this constraint may be stated as: *maximize the weighted sum of the satisfied preferences (a)-(c) for all the scheduled maintenance events.*

5.3 Solution Approach

The solution approach in MSS is motivated by the following two observations based on the typical problem data. First, generating a feasible maintenance schedule is usually straightforward, with the main resource bottleneck being the availability of maintenance technicians. Second, obtaining the optimal maintenance schedule is difficult, due to presence of the multiple objectives and preferences, as described above.

Several approximate methods (local search, genetic algorithms, dispatch rules) as well as exact methods (branch-and-bound search, constraint programming, mixed-integer programming). have been suggested in the literature for solving manufacturing scheduling problems. Since the goal in MSS is to find the optimal solution for the maintenance scheduling problem, the primary focus has been on exact methods, although these methods have different strengths and weaknesses, particularly for present class of scheduling problems with complex objectives and side constraints.

For example, constraint programming solvers [4] are able to compactly model scheduling problems using an event-based formulation, and are successful at finding good feasible solutions in highly resource-constrained problems; however, these solvers are not always suitable for problems with complex, non-convex objectives (e.g., consider the ILOG CP Optimizer [15], which has a flexible representation for modelling objective functions, but requires these to be in a semi-convex form). Similarly, mixed-integer programming solvers based on time-indexed formulations are effective for the modeling and solution of scheduling problems with complex, non-regular objectives, and yield good linear programming relaxations when the objective function is of the form $\sum_j f_j(C_j)$, where C_j is the completion time of job j , but the resulting formulations can be very large, since the number of decision variables depends on the length of the time horizon, and therefore, this approach is often restricted to small problems. Finally, a number of specialized branch-and-bound techniques have been developed for large-scale scheduling problems [21], but this formulation is not very suitable for incorporating side constraints.

In the MSS formulation, the five objectives and constraint preferences mentioned above, were ranked in the following order for the optimal solution (starting with the most important): (a) resource leveling, (b) separation constraints, (c) follow-up maintenance constraints, (d) production disruption, and (e) earliness-tardiness costs.

The MSS solution approach, which is inspired by lexicographic goal programming, is to first solve the scheduling problem with respect to the most important objective, ignoring all other objectives. Let f_1 denote the value for the first objective in the corresponding solution. A new *constraint* is added to the model, after fixing the value for the first objective at f_1 , and the scheduling problem is solved for the second objective only, but with the first objective now represented as a constraint in the model. Subsequently, we add a second constraint to the model based on the objective value found for the second objective. We continue cycling through this process until we have solved the problem for all objectives.

The objectives for resource levelling, separation constraints and follow-up maintenance constraints can be solved very efficiently using constraint programming (the details of this solver, which uses depth-first chronological backtracking, the SetTimes branching heuristic and the timetable resource constraint propagator can be found in [4]; we note, parenthetically, that stronger propagation than timetable was not found to be useful in practice). Finding a feasible solution to the maintenance scheduling problem in this way is very fast (less than a second of CPU time), so that the estimate of the minimum number of technicians required can be obtained by solving a series of feasibility problems in the following way: For each feasibility problem, the number of available technicians is set to a fixed value, and a binary search is used to find the smallest number of technicians for which a feasible solution can be found for scheduling all the pending maintenance events. A similar approach is used to determine the best value of the objective for the separation constraints and the follow-up maintenance constraints. The objective for disruption and earliness-tardiness is solved using mixed-integer programming, for which we use the time-indexed formulation with some additional cuts. In practice, the mixed-integer programming solver is much slower than the constraint programming solver, due to the large formulation. However, with time being discretized into 15 minute buckets over a 2-week horizon, the CPU times for the mixed-integer programming solver was of the order of 5-20 minutes (using CPLEX version 11).

5.4 Summary

The MSS-based system described here, is now being routinely used to generate maintenance schedules for IBM 300mm fab over two-week time horizons. Specifically, this system determines the timing of maintenance events for individual tools, taking into account the availability of the appropriately-qualified technicians, so as to minimize the production disruptions in the fab, as a consequence of the scheduling of planned and unplanned maintenance events.

6 Summary and Future Directions

In Section 1, we noted two important concerns in a semiconductor fab, which were the individual product quality and the overall production efficiency. The EDMS and TRACER methodologies generally address the product-quality concern, while WIPSim and MSS generally address the production-efficiency concern. It is often tacitly assumed that the various factors and outcomes associated with these two important concerns are unrelated to each other; however, this decoupling also reflects some of the organizational aspects of the fab, and in particular, the roles and responsibilities that are assigned for various facets of the the data collection, and for the actionability of the analytics-based operational recommendations.

The development of an integrated framework for analytical applications for a semiconductor fab should consider the functional requirements along two dimensions. The first dimension comprises the various solution requirements, such as aberrant behavior detection, root cause analysis, process and operations optimization, and predictive modeling. The second dimension comprises the appropriate data-delivery requirements, ranging from real-time to off-line, that are needed for effectively implementing each of the proposed solutions. The synergistic evolution of the solution and data-delivery requirements, will drive the emergence of of new analytics solutions, with increasingly more timely and comprehensive data sets.

We now describe some directions that we have identified as important for further research, based on our experiences to date.

The first direction is to further broaden the scope of the analytics and data perspectives. As noted above, the two areas of concern, viz., individual product quality and overall production efficiency are often addressed as separate and unrelated problem domains; however, in practice, there is often an interaction between these domains. For example, an operational decision on the allocation of certain tools to certain processes must take into account the resulting implications along both the product quality and manufacturing productivity dimensions. (In general, being able to use all tools for all tasks maximises manufacturing productivity, but minimises the product quality.) Furthermore, the specialized nature of the engineering expertise, and the fine-grained details of the organizational structure, leads to even further compartmentalization of the data and analytics perspectives within these two ostensibly-separate problem domains. As a result, the characterization of complete event and aberrant behavior, and the subsequent root cause diagnosis, is often compromised. Therefore, a broadening scope of the analytics and data perspectives is an important future direction, and this scope should consider all the disparate data inputs in the fab, including for example, process history modifications, in-line contamination, in-line chemical and physical analyses, in-line electrical, final product test, environmental, consumables, and field performance. In addition, there is often an additional compartmentalization between the groups responsible for factory operations and financial management. Financial management frequently relies on limited static models of factory operations, which may never be realized in practice. Therefore realistic models of factory operations and factory demands, along with explicit linkages between them, will provide improved financial management responsiveness, risk management, and financial outlooks.

The second direction is to extend the analytical techniques that have been developed to date, to better handle the numerous practical data challenges, such as the presence of heterogeneous data types; modeling requirements for non-standard distributions; data quality issues involving errors, outliers, and missing measurements; and data heterogeneity in terms of the variation of sample sizes and measurement scales across the data sets. Another related challenge is that there is often little consensus on what constitutes either routine or aberrant behavior in many of these data sets, as well as the need to detect specific aberrant behaviors in a more general setting than is being done currently. For example, the EDMS-induced rules in Section 2 are effective for identifying aberrant conditions in a single tool, but further work is required for the case when specific pairs of tools lead to aberrant conditions, although the the constituent tools may be individually non-aberrant.

The third direction is to expand the use of predictive models, in particular, for the characterization of the intermediate product yield and process performance, based on the process trace data. There is a significant interest in the semiconductor manufacturing industry in the so-called “virtual metrology” application, which has the potential for replacing the time-consuming and capital-intensive product testing events in the manufacturing workflow, by credible model predictions. Predictive modeling is also extensively used in WIPSim to obtain estimates of the daily incoming WIP projections at the individual processing centers.

The fourth direction is to develop methodologies for detecting aberrant conditions that are of genuine operational significance. The current approach of eliciting expert information is effective in reducing the number of false positives to the end user, but this approach is difficult to scale to very high-dimensional and high-throughput monitoring applications. The use of process trace data to predict product performance, as mentioned earlier, also holds the promise of inferring opera-

tionally significant thresholds for process aberrations based on individual product requirements.

In closing, we note that the challenges of improving product quality and production efficiency are not unique to semiconductor fabs, and these challenges are encountered in all manufacturing and service enterprises, albeit with differences in the respective functional requirements. The opportunities for new solutions and applications in these enterprises will increasingly be driven by the evolution of the instrumentation, data capture, and data-serving capabilities of their production operations.

Acknowledgement

The 300mm fab analytics project is a team effort involving a large number of our colleagues at IBM, whose contributions to the overall scope of the project, and to the development of the specific techniques described in this paper is deeply acknowledged.

References

- [1] C. Apte, S. Weiss, and G. Grout. Predicting defects in disk drive manufacturing: A case study in high-dimensional classification. In *IEEE CAIA (93)*, pages 212–218, 1993.
- [2] S. Bagchi, C. Chen-Ritzo, L. Burns, and S. Catlett. Experiences in implementing simulation-based support for operational decision making in semiconductor manufacturing. *European Journal of Industrial Engineering*, to appear, 2009.
- [3] S. Bagchi, C. Chen-Ritzo, S. T. Shikalgar, and M. Toner. A full-factory simulator as a daily decision-support tool for 300mm wafer fabrication productivity. In S. J. Mason, R. R. Hill, L. Moench, and O. Rose, editors, *Proceedings of the 2008 Winter Simulation Conference*, pages 2021–2029, 2008.
- [4] P. Baptiste, C. L. Pape, and W. Nuijten. *Constraint-Based Scheduling - Applying Constraint Programming to Scheduling Problems*. International Series in Operations Research and Management Science, Oxford, 2001.
- [5] L. Breiman, J. Friedman, R. Olshen, and C. Stone. *Classification and Regression Trees*. Wadsworth, Monterrey, CA., 1984.
- [6] M. S. Bureau, C. Dauzere-Peres, C. Yugma, L. Vermarien, and J. B. Maria. Simulation results and formalism for global-local scheduling in semiconductor manufacturing facilities. In S. G. Henderson, B. Biller, M.-H. Hsieh, J. Shortle, J. D. Tew, and R. R. Barton, editors, *Proceedings of the 2007 Winter Simulation Conference*, pages 1768–1773, 2007.
- [7] R. Chen, K. Yeh, C. Chang, and H. Chien. Using data mining technology to improve manufacturing quality - a case study of LCD driver IC packaging industry. In *Seventh ACIS International Conference on Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing*, pages 115–119, 2006.
- [8] W. Chen, S. Tseng, K. Hsiao, and C. Liu. A data mining project for solving low-yield situations of semiconductor manufacturing. In *Proceedings of IEEE Conference and Workshop on Advanced Semiconductor Manufacturing*, pages 129–134, 2004.

- [9] A. J. Davenport. Integrated maintenance scheduling for semiconductor manufacturing. In *19th International Conference on Automated Planning and Scheduling: Scheduling and Planning Applications Workshop*, to appear, 2009.
- [10] C. D. DeJong and S. A. Fischbein. Semiconductor manufacturing material handling systems: integrating dynamic fab capacity and automation models for 300mm semiconductor manufacturing. In J. A. Jones, R. R. Barton, K. Kang, and P. A. Fishwick, editors, *Proceedings of the 2000 Winter Simulation Conference*, pages 1505–1509, 2000.
- [11] T. Fountain, T. Dietterich, and B. Sudyka. Mining IC test data to optimize VLSI testing. In *Proceedings of the Sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 18–25, 2000.
- [12] R. Goodwin, R. Miller, E. Tuv, A. Borisov, M. Janakiram, and S. Louchheim. Advancements and applications of statistical learning/data mining in semiconductor manufacturing. *Intel Technology Journal*, 8(4):325–336, 2004.
- [13] J. Harding, M. Shahbaz, Srinivas, and A. Kusiak. Data mining in manufacturing: A review. *Manufacturing Science and Engineering*, 128(4):969–976, 2006.
- [14] G. Kong. Tool commonality analysis for yield enhancement. In *Proceedings of IEEE Conference and Workshop on Advanced Semiconductor Manufacturing*, pages 202–205, 2004.
- [15] P. Laborie, J. Rogerie, P. Shaw, P. Vilim, and F. Wagner. Ilog cp optimizer: Detailed scheduling model and opl formulation. Technical Report 08-002, ILOG, xxxx.
- [16] Huiran Liu, Zhibin Jiang, and Richard Y.K. Fung. Modeling of large-scale complex re-entrant manufacturing systems by extended object-oriented petri nets. *The International Journal of Advanced Manufacturing Technology*, 27(1):190–204, 2005.
- [17] S. J. Mason and P. A. Jensen. A comparison study of the lofic of four wafer fabrication simulators. In J. M. Charnes, D. J. Morrice, D. T. Bruner, and J. J. Swain, editors, *Proceedings of the 1996 Winter Simulation Conference*, pages 1031–1038, 1996.
- [18] O. Rose. A comparison study of the lofic of four wafer fabrication simulators. In S. G. Henderson, B. Biller, M.-H. Hsieh, J. Shortle, J. D. Tew, and R. R. Barton, editors, *Proceedings of the 2007 Winter Simulation Conference*, pages 1078–1712, 1996.
- [19] A. Skumanich, J. Yamartino, D. Mui, and D. Lymberopoulosn. Advanced etch applications using tool level data. *Solid State Technology*, 47(6):47–52, 2004.
- [20] N. Slonim, R. Baseman, E. Aharoni, S. Bagchi, D. Baras, D. Bickson, A. Ghoting, Y. M. Lee, A. Lozano, O. Margalit, H. Neuvirth-Telem, A. Niculescu-mizil, C. Reddy, M. Rosen-Zvi, S. Siegel, F. Tipu, S. Weiss, and E. Yashchin. The TRACER - a general framework for process trace data analysis. *The International Journal of Advanced Manufacturing Technology*, to appear, 2009.
- [21] F. Sourd. New exact algorithms for one-machine earliness-tardiness scheduling. *INFORMS Journal of Computing*, 21:167–175, 2009.