Simulation in Produktion
und Logistik 2017
Sigrid Wenzel & Tim Peter (Hrsg.)
kassel university press, Kassel 2017

Transformation of Real-time Reporting Event Data to Long-term Simulation Models

Transformation von Echtzeit-Reporting Event Datenelementen in Langfrist-Simulationsmodelle

Mike Gißrau, Torsten Gereke, SYSTEMA, Dresden (Germany),
mike.gissrau@systemagmbh.de, torsten.gereke@systemagmbh.de

Abstract: Data collection and aggregation is one of the master challenges in creating valuable simulation models. The complexity of this task still increases in typical inhomogeneous and historically grown IT infrastructures even in older wafer fabs. This paper describes an advanced way to use an online and real-time reporting source to generate all necessary data for a long-term simulation model. Within the paper, a short introduction to the ability of real-time data aggregation, the data pre-processing and simulation model transformation and generation is given. The main objective is to offer a new common interface where different simulation solutions can access the required simulation data without additional integration effort.

1 Introduction

Today simulation is one of the still growing techniques to identify and analyze complex production environments. Banks (2001) proposed simulation to be "consistently one of the top three methodologies used by industrial engineers, management scientists, and operations research".

In semiconductor environment, simulation becomes more and more a major element on short-term and long-term planning and improvement of wafer fabs. Typical fields of investigation are production planning, ramp-up scenarios or the improvement of dispatching and scheduling approaches (Andersson et al. 2008). Even in Europe, complexity of production processes is continuously increasing. At the same time, customer specific variations of products are transforming classical mass production operations into sensitive production networks with a wide range of different production classes. Statistical methods for production planning become more and more imprecise and unstable.

In general, the data acquisition, verification and validation of simulation models is hard work. The complexity of these tasks still increase in typical inhomogeneous and historically grown IT infrastructures even in older wafer fabs. Banks (2008) underlined this problem by the following statement:

"If there is no data available, not even estimates, simulation is not advisable"

One of the major problems of today's simulation projects is the acquisition of data for a valid and valuable simulation experiment and results. Typical elements are product flows, processing times or equipment dedications. Often, this information is distributed at different systems at typical semiconductor manufacturing sites. Besides the manufacturing execution system, also enterprise resource planning tools or tools for recipe management, equipment maintenance planning and others are in place. Sometimes, the information is not accessible.

In this paper, we want to propose a new consistent way to generate important master data elements from manufacturing events within an event-oriented data aggregation real-time reporting environment to create valuable master data for long-term simulation experiments and analysis. Today typical approaches request all necessary data from a database or defined file format producing a lot of load and validation effort (Son and Wysk 2001). An efficient data collection and generation approach is also mentioned as one of the challenges in simulation of complex manufacturing systems (Noack and Rose 2008).

2 Long-term vs. Short-term Simulation Models

In today's facility environments, two general model types (and all in-between) are known and mentioned by Noack (2012) and Gißrau (2013).

Within long-term models, the production process under investigation is modeled with more general assumptions to generate a valuable output for questions regarding capacity planning or long-term resource allocation. This type of model is widely used even in semiconductor environment. Typical, many statistical distributions, e.g. for equipment availability or operator availability are used. For short-term prediction, these models are often not applicable due to their lower level of granularity. In this case, many more data like planned maintenance schedules, the operator work schedule and other detailed data is necessary. Without these data, the prediction is often not sufficient to fulfill typical use cases like work-in-process (WIP) forecast or estimation of flow factors.

In this paper, we mainly address the long-term simulation model type as a first step of our investigations. We will move to the real-time forecast later in our research program.

3 Real-time Data Aggregation via Basic Manufacturing Events

In this section, we give a rough overview about the motivation and the idea of real-time data aggregation via basic manufacturing events.

The real-time aggregation approach is motivated by the relationships between the logical descriptions of objects of the real-world system like a manufacturing system and the real-time information system. For example, a specific movement of a part of a machine may correspond to the engineering concept "process start". Another movement or event may correspond to a logistical concept called "cycle time". Such concepts may be used in different kinds of systems (Manufacturing Execution

Systems, Enterprise Resource Planning, etc.). However, all these different kinds of systems could theoretically share the same basic ontology.

The problem domain is to be described as the availability of information (in a most general logical, qualitative and quantitative sense) in order to monitor, supervise, and qualify any kind of industrial/business/financial process, to steer, control, drive and optimize such processes. Assuming different objects or processes (like business processes, financial processes, engineering processes), which are characterized through specific and well-defined figures. Typically, such figures are given as performance indicators, engineering measurements (for example: physical measurements (within semiconductor industry termed inline-data), functional measurements (within semiconductor industry termed test-data), derived measures (example from the semiconductor industry: yield)), or logical associations/ attributions in the general and abstract case.

We have conducted analysis of the mathematical structure describing how information is derived from these scenarios. This study has concluded that any corresponding system model in all the different domains and applications must incorporate the structure of the same basic system model. That is, because of the compositional characteristics, any parameter or data component, which describes the behavior of subsystems on the lowest level of granularity, can be grouped and aggregated with corresponding parameters using historical records, and within the mathematical concept of linear information spaces. The system model preserves the linearity of the overall model, and defines the corresponding linear relations of the historical records. It can be conceptualized as a nondeterministic system according Lee and Seshia (2011).

A fundamental premise of the newly developed methodology (see Fig. 1) is that all incoming data elements or events are immediately captured and transformed into components of information. For example, out of the event “production lot process step terminated”, will result in ‘real-time’ calculation of the information component, which represents the proportionate partial data value for the desired characterization of the production process (kinds of performance indicator, or any other desired aggregation).

Each event can contain multi-dimensional additional context values like product name or current operation. In addition, this information is part of the information components.

This approach is practicable, since the corresponding information is relatively small and is being processed as soon as it is available. Such minimized computational effort is not possible within common solutions even for approaches utilizing small-scale aggregation strategies (i.e. data to be aggregated is split into small batches, which fit in memory).

For more information to the topic, we refer to Luhn et al. (2014, 2015a, 2015b, 2016).

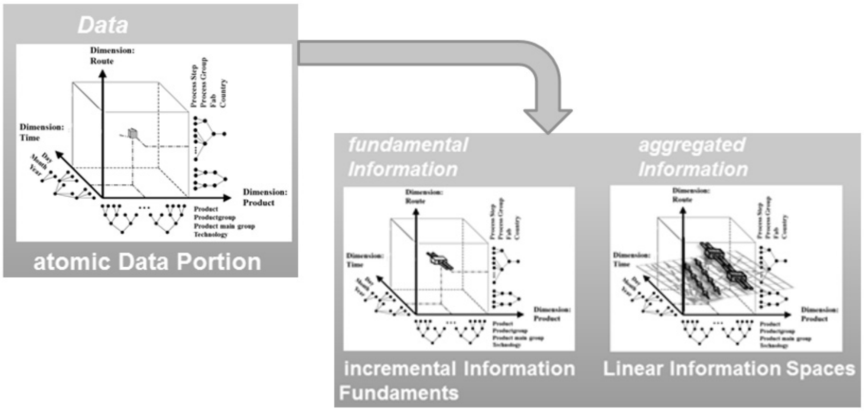

Figure 1: Schematic representation of the new methodology: any Fab event (atomic data portion) gets transformed immediately into incremental Information Fundaments

4 Master Data Transformation Algorithms

In this section, two different examples for master data generation algorithms from the basic manufacturing events are explained in detail. Of course, several more algorithms are used in order to produce the full set of required data for a simulation model.

4.1 Example 1 – Product Flow

One of the most important elements at a manufacturing model is the description of the product flow of each production entity, e.g. the lot. Besides the right order of operations, also the corresponding production resources and their processing time statistics is necessary. Also possible rework information should be taken into account. Each production entity, also of the same product or technology node, can change its production flow due to different events like changing flow specifications, rework or context changes over time. The proposed algorithm takes the individual flow of each lot under defined context information like product name to form a common flow for each entity of these classes. Figure 2 illustrates the problem by visualizing the summary operation flow of a couple of different lots of the same product from real fab data. The black line marks the reference flow (also called Golden Flow), which is calculated by the different lot flow occurrences.

Figure 2: Exemplified visualization of different lot routes of the same product

To find the right combination of steps for the mass of entities, we used an adapted diff algorithm described by Hunt and McIllroy (1976) to update the flow information by each incoming lot event. It bases on the longest common subsequence (LCS) problem where the objective is to find the longest subsequence common to all sequences of a given set. In this case, the operational information of two production entities are compared. Each operation element O is assigned a weight W describing the frequency a production entity passes this operation in past. Newer occurrences of an operation gets a higher weight than older ones to allow a fast adaption to changes at the flow, e.g. in case of a change of a defined route in a product flow. At each comparison, the weights are calculated as follows:

- Equal elements: $W_{New} = W_{Old}+S_L$
- New element: $W_{New} = S_L$
- Removed element: $W_{New} = W_{Old}-S_L$

The parameter S_L describes the current number of sub-elements (e.g. wafer) of the production entity. With the usage of the weights, it is possible to remove unwanted influences like rework or special operation flows of production entities. Figure 3 illustrates one example, where Lot 2 executed two operations in past, whereas the current production flow has three operations stored. For operation Y, the new weight is reduced in order to reduce the importance of this step.

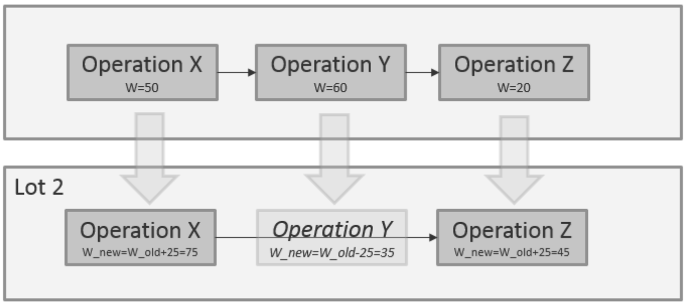

Figure 3: *Example of operation weight recalculation*

With the calculated weights, it is possible to generate a general flow of operations for different contexts of the production entities, like the product name.

The algorithm is tested with different industrial and synthetic data sets. The synthetic data sets contain of different simulation experiments with a complex semiconductor facility model (see Tab. 1). Within the simulator experiments, different rework rates are simulated to generate variable lot routes within the same product. The rework rates range between 10 % and 50 %. Our experiments shows a valuable output until a rework rate of 30 %. Above this level, the calculated workflows are not stable and usable. At our experiments, the calculated deviation D of the calculated flows to the references is defined by

$$D = \frac{N_{Diff}}{N} \tag{1}$$

where N is the total number of operations and N_{Diff} the number of operations which are not calculated in the right way.

Table 1: *Example of complex simulation parameters*

Domain	Property	Value
	Count	189
Equipment	Setup	Yes
	Batch	Yes
	Count	9
Products	Raw Process Time	273 h-441 h
	Process Steps	234-355
Setup	Run Length	360 d
	Warm Up	90 d

Some results are illustrated in Figure 4, which shows an example report of the resulting calculation of the product flow.

Figure 4: *Example analysis of operation flow deviation per lot*

In general, the algorithm performs within an average error rate of lower than 5 % in comparison to the reference flow until a reasonable non-standard-flow rate of about 30 %. This behavior could also be investigated by our real fab data experiments from typical semiconductor foundries. With this algorithm, it is now possible to generate the product flows and tool dedications without request the corresponding master data from the warehouse of the customer.

4.2 Example 2 – General Production Control Information

In today's manufacturing environments, several different production control techniques like Dispatching or Scheduling are quite common in daily business (Gißrau 2013). In reality, a wide range of different simple to complex heuristics and algorithms are used to support the manufacturing performance optimization.

Bringing this data and information to a simulation model is very difficult (Noack 2012). Besides the technical problems of how to transform such an algorithm into the simulation language, often also key aspects are not known. Some of the information is stored only in the head of the operating personnel. For simulation purposes, it is quite important to know the main production schedule aspects per equipment.

As a first step, we decided to implement a simplified view on dispatching strategies per equipment by using the available data of the real-time reporting environment without having any access to the scheduling and dispatching systems of the factory. The basic idea is to identify lot groups according to their attributes and contextual information, and then identify the sorting criteria per group. In the most manufacturing environments, we see a group based prioritization and sorting. For this, additional statistics per lot and equipment are gathered and aggregated while the basic manufacturing events comes in. One main idea is to collect the number of preceding processing operations – the lot has to wait on an equipment until the process of the lot itself starts. This idea is transformed in different statistical values describing the weight calculation in different ways. One output is the simple dispatching weight $W_{Equipment}$ that describes the absolute number of preceding processing operations until the lot is produced. Figure 5 illustrates the general idea with a simple example.

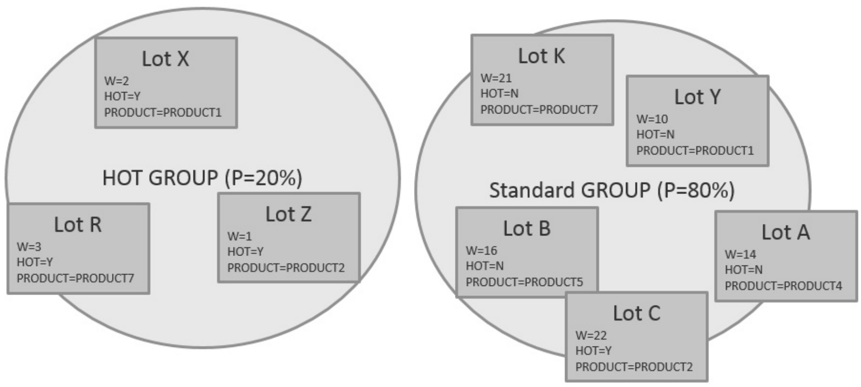

Figure 5: *Example for dispatch group classification*

We suppose several different lots with two attributes, the hot flag and the product name. Furthermore, this example is addressed to one defined equipment. When the different lots comes to the equipment, for each processing of one lot, the waiting lots weight is increased by one. After a while, the lots have the weights illustrated in the figure. Now it is possible to classify the different lots into several groups via grouping over the weights. Lots with a very low weight are classified in one group (in the example the HOT group), whereas the rest of the lots are classified in another group. In addition, each group can be assigned the amount of all lots passing over time, e.g. 20 % for the HOT group.

The grouping is done with classical statistical methods of clustering. Within the group, it is tried to find out the right sorting criteria (according to available lot attributes), otherwise FIFO is supposed. Several experiments are executed to verify and validate the supposed algorithm. Figure 6 illustrates experiments with a complex semiconductor model with different dispatching policies and static priorities. Besides the FIFO policy, the static prioritization via three groups and a mixed experiment of both are illustrated.

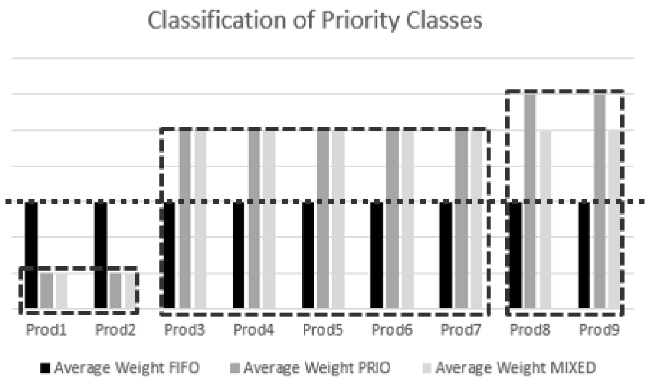

Figure 6: *Experimental output with a complex semiconductor simulation model*

Figure 7 shows the grouping for a real world example, where six basic main dispatching groups are identified.

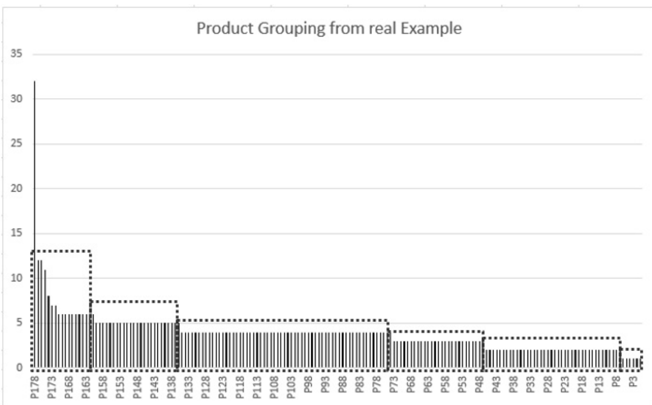

Figure 7: *Product grouping for one equipment with real-world data*

With this information, it is possible to setup the simulation model with a generic and generalized view on the production control techniques applied in a manufacturing area. Our experiments show a very sophisticated result in detection of the right lot groups and their internal sorting criteria like due date.

5 Simulation Model Generation Workflow

With the exemplified algorithms, it is possible to offer a wide range of different master data required for simulation purposes. Figure 8 illustrates the general workflow.

The start is the empty simulation model, where all necessary objects and its logic is predefined in the simulation environment. This includes the logic of the object flow and its states and behavior. After/during collection of the event data, the real-time reporting environment offers a wide range of different master data sources, which can be accessed by the simulation system in real-time. The simulation environment imports the master data and is able to generate the simulation model. To illustrate this, the figure above shows a generated model without tool layout information from a complex semiconductor simulation environment. The next step is to validate and verify the generated simulation model. The real-time reporting environment also supports this by offering a historical view on the data and allows the system to initialize with a historical set of data (e.g. product distributions, tool down behavior, etc.).

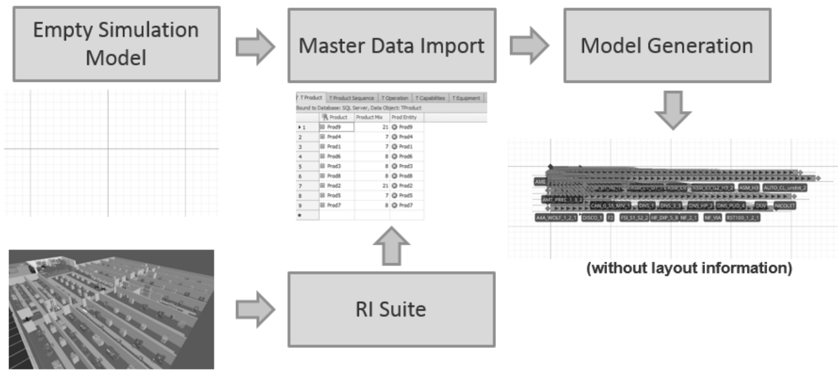

Figure 8: *General workflow of model generation*

6 Conclusion and Outlook

Within this paper, we describe a possible way to generate simulation model data from an event-driven approach rather than collect all necessary data from different customer sub systems. The main idea is to reuse the event driven data aggregation approach of the real time reporting to generate a sufficient level of master data for the generation of simulation models. The results show valuable outcome in the data conformity and speed. The master data is immediately available after request so that also future online simulation approaches can be supported. Using manufacturing events rather than explicit data calls reduces the complexity of the data calls for model generation, but needs also a sufficient level of availability of the events and its contextual data. The next step within this project is the development of a common interface for a wide range of different simulation systems as well as the support of offline and online simulation approaches. Within this work, future demands of data delivery and aggregation can be realized in real-time, which is one of the key aspects to meet industrial requirements for the digital transformation of manufacturing.

References

Andersson, M.; Ng, A.; Grimm, H.: Simulation optimization for industrial scheduling using hybrid genetic representation. In: Mason S.J.; Hill, R.R.; Mönch, L.; Rose, O.; Jefferson, T.; and Fowler, J.W. (eds.): Proceedings of the 2008 Winter Simulation Conference, 2008, pp. 2004-2011.

Banks, J.: Discrete event system simulation. 3rd ed., Prentice Hall, 2001.

Banks, J.: Handbook of Simulation: Principles, Methodology, Advances, Applications, and Practice. John Wiley & Sons, Inc. 2007.

Gißrau, M.: Development and Introduction of Semiconductor Production Control Enhancements for Customer Oriented Manufacturing. PhD Thesis, Universität der Bundeswehr München, 2013.

Hunt, J.; McIlroy, M.D.: An Algorithm for Differential File Comparison (PDF). Computing Science Technical Report, Bell Laboratories. June 1976.

Lee, E.A.; Seshia, S.A.: Introduction to Embedded Systems – A Cyber-Physical Systems Approach. LeeSeshia.org 2011.

Luhn, G.; Ertelt, M.; Zinner, M.: Real-Time Information Systems and Methodology based on Continuous Homomorphic Processing in Linear Information Spaces; patent application pending, placed as US Provisional PA, 7.3.2014.

Luhn, G.; Bartl, K.: Real-Time Information System as key enabler for "Industry 4.0" inspired Semiconductor Manufacturing – Turning Data Into Fundamental Information. Proceedings of the 15th European Advanced Process Control and Manufacturing Conference, Freising (Germany) 2015a.

Luhn, G.; Bartl, K.; Habich, D.: Real-Time Information Base as key enabler for Manufacturing Intelligence and Industrie 4.0 – Building the bridge between "real" Real-time Exploitation and Big Data via mathematically grounded Information Fundamentals. Proceedings of the 2015 ASMC, Saratoga Springs NY (USA), 2015b.

Luhn, G.; Orbanz, J.: "Deep" Production Complexity Management. Real-Time Manufacturing in Fractal Dimensions. Proceedings of the 16th European Advanced Process Control and Manufacturing Conference, Reutlingen (Germany), 2016.

Noack, D.; Rose, O. A simulation based optimization algorithm for slack reduction and workforce scheduling. In: Mason, S.J.; Hill, R.R.; Mönch, L.; Rose, O.; Jefferson, T.; Fowler, J.W. (eds.): Proceedings of the 2008 Winter Simulation Conference, 2008, pp. 1989-1994.

Noack, D.: Towards Realization of a high-fidelity simulation model for short-term horizon forecasting in wafer fabrication facilities. In: Johansson, B.; Jain, S.; Montoya-Torres, J.; Hugan, J.; Yücesan, E. (eds.): Proceedings of the 2010 Winter Simulation Conference, 2010. pp. 2563-2574.

Noack, D.. Online Simulation in Semiconductor Manufacturing. PhD Thesis. Universität der Bundeswehr München, 2012.

Son, Y.; Wysk, R.: Automatic simulation model generation for simulation-based, real-time shop floor control. Journal Computers in Industry 45 (2001) 3, pp. 291-308.

Wand, Y.; Weber, R.: Toward a theory of the deep structure of information systems. Information Systems Journal 5 (1995) 3, pp. 213-223.