Proceedings of the 2003 Winter Simulation Conference
S. Chick, P. J. Sánchez, D. Ferrin, and D. J. Morrice, eds.

CONSTANT TIME INTERVAL PRODUCTION PLANNING WITH APPLICATION TO WIP CONTROL IN SEMICONDUCTOR FABRICATION

Kazuo Miyashita

National Institute of Advanced Industrial
Science and Technology (AIST)
Tsukuba, Ibaraki 305-8564, JAPAN

Kazuyuki Senoh
Hiroyuki Ozaki
Hirofumi Matsuo

Institute of Policy and Planning Sciences
University of Tsukuba
Tsukuba, Ibaraki 305-0006, JAPAN

ABSTRACT

We develop a constant time interval production planning and control methodology, called CONSTIN, and its associated simulation system for a large-scale and unstable semiconductor manufacturing process. CONSTIN moves work-in-process inventories (WIP) between processes only at a constant time interval, and consequently maintains a desirable level of WIP. Our theoretical and experimental analysis shows a clear relationship between WIP levels and the time interval in CONSTIN. Computational experiments with realistic wafer fabrication process data demonstrate that CONSTIN is comparable in simulation accuracy to a popular event-driven simulator and can run much faster. Additional experiments also manifest that, with appropriate control rules, CONSTIN can restore the desired levels of WIP from extreme deviations and maintain them. Therefore, we conclude that CONSTIN is a promising methodology of production planning and WIP control for the semiconductor manufacturing process.

1 INTRODUCTION

Semiconductor manufacturing process is among the most complicated manufacturing processes. For example, the number of production steps for semiconductor manufacturing are usually not less than a few hundred with a large number of repetitive reentrant loops, and its lead time extends over a couple of months (Atherton and Atherton 1995). Therefore, developing an optimal production plan for the semiconductor manufacturing process in an exact sense is computationally intractable. Hence, many simulation-based methods using various releasing heuristics and dispatching rules have been investigated extensively in the literature, and have been applied to the factories (Wein 1988, Rose 2001). However, the conventional simulation-based methods have the following drawbacks in general:

1. **Speed of simulation:** In order to determine many production parameter values such as product-mix and job release rate in highly stochastic manufacturing environments, it is necessary to conduct many trial-and-error simulation runs. The computation time required to run such simulation is too long to be used for making a short-term (e.g., hourly or daily) production plan for semiconductor manufacturing.
2. **Feasibility of results:** The simulation-based methods produce very detailed decisions on releasing and dispatching jobs based on a fixed manufacturing scenario, which tend to be easily deviated from the real manufacturing situations due to disruptive events such as machine failures. Hence, most of those decisions produced by the simulations cannot be readily used for practical production control purpose.

Furthermore, the fierce competition in the global market place and short technology life cycles require the semiconductor industry to always deploy state-of-the-art manufacturing technologies. It causes their manufacturing processes to be unstable and unpredictable because they most of the time operate early in the learning curves of manufacturing. In order to hedge disruptions in manufacturing and avoid the ripple effects of exceptional events such as machine breakdowns and yield loss, they tend to keep a large number of work-in-process inventory (WIP) and end up with unnecessarily long production lead times. The costs of having large WIP are high because of rapid product obsolescence associated with short product and technology life cycles. Consumers of semiconductors also demand short lead times and customized products. Hence, a production planning and control methodology that can always maintain the lowest possible levels of WIP to keep high output levels are of critical importance in the semiconductor manufacturing industry.

For addressing the above-mentioned problems, a faster simulation method that can produce more robust and executable production decisions is indispensable. In this paper, we propose an innovative production planning and control methodology and its computer simulation system, and demonstrate its applicability in WIP control for the semiconductor manufacturing processes.

2 PROPOSED METHODOLOGY

We propose the "CONSTIN" (CONStant Time INterval) production planning and control methodology for achieving robust and stable production in large-scale and complicated manufacturing environments with highly fluctuating factors. In CONSTIN, a manufacturing process comprises a sequence of *operations* which are run in a synchronized manner, and WIP is transfered between operations only at the end of a fixed time interval. Hence, the WIP level at each operation changes periodically as shown in Fig. 1. As an additional restriction on WIP movements in CONSTIN, WIP is assumed not to move beyond its next operation at a single time interval. Hereafter, we use a term *period* to represent a fixed time interval used to synchronize operations in CONSTIN.

For an intuitive comprehension of CONSTIN, assume that you are an operator in charge of a workstation (i.e., machine) in an imaginary fab. At the beginning of every period, you see WIP in front of your workstation. Since the workstation can process several operations for various products, the WIP is classified for several operation types. CONSTIN provides you with an order that you should produce how much of each type of WIP and move to its next workstation in the period. Thus, in CONSTIN, dispatching decisions (i.e., sequencing for processing each type of WIP) do not have effects on succeeding operations as long as you can process all the designated WIP to be processed within the period.

In CONSTIN, it is possible to prevent ripple effects of machine failures and yield losses in the manufacturing process, if those deviational events are fixed within a period or if a sufficient level of WIP is kept to allow continuous production in the neighboring operations of the deviations. Hence, CONSTIN can be a suitable method to accomplish stable and robust production in the face of fluctuating manufacturing capabilities and unpredictable product yields.

Such robustness of CONSTIN, however, imposes restrictions on conventional WIP handling. Without an appropriate control of WIP level and setting of the time interval, CONSTIN might waste manufacturing capacity and increase WIP not as intended. As will be shown by theoretical analysis and extensive computational experiments in this paper, CONSTIN can avoid such problems and produce promising results for large scale manufacturing problems.

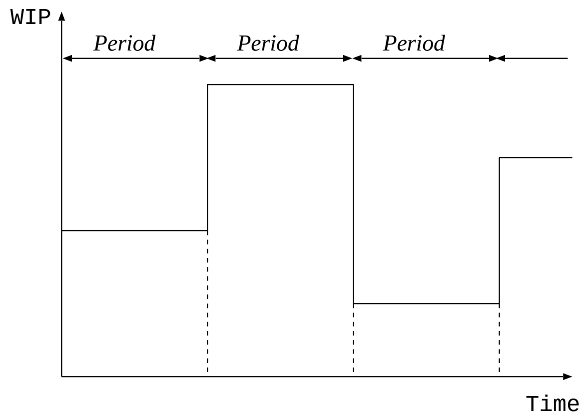

Figure 1: Periodic Transition of WIP Level in CONSTIN

2.1 Notation and Model

m = number of workstations;

g = number of products;

n_p = number of operations for product p;

n = total number of operations for all products;

c = $(c_1, c_2, \ldots, c_m)^T$, vector of production capacity of workstation per period;

s_i = processing time per unit for operation i;

S = $m \times n$ unit processing time matrix; the (k, i)-th element of S is s_i if operation i is performed at workstation k, and 0 otherwise, for $k = 1, 2, \ldots, m$ and i$= 1, 2, \ldots, n$;

$r_p(t)$ = job release units of product p at period t;

$x(t)$ = $(x_1(t), x_2(t), \ldots, x_n(t))^T$, vector of production start units for operation i in period t;

$w(t)$ = $(w_1(t), w_2(t), \ldots, w_n(t))^T$, vector of inventory levels of operation i at the beginning of period t;

$z(t)$ = $(z_1(t), z_2(t), \ldots, z_n(t))^T$, vector of production units completed for operation i in period t

$u(t)$ = $(u_1(t), u_2(t), \ldots, u_n(t))^T$, vector of rework units for operation i in period t;

$v(t)$ = $(v_1(t), v_2(t), \ldots, v_n(t))^T$, vector of scrap units for operation i in period t.

Based upon the assumption of periodical WIP transfer to the succeeding operation, CONSTIN is modeled using the above notations. Approximate mathematical analysis of the model similar to CONSTIN was done in (Gong and Matsuo 1997) although their model lacks the details required for realistic implementation.

```
initializeData();
t = 0;
while (t <= EndOfSimulation)  {
    runForPeriod(t);
    t += Period;
}
```

Figure 2: Simulation Loop

In CONSTIN, WIP transition in each period is calculated as follows:

- when operation i is the first operation of product p

$$w_i(t+1) = w_i(t) + r_p(t) - (z_i(t) - u_i(t)).$$

- otherwise

$$w_i(t+1) = w_i(t) + (z_{i-1}(t) - u_{i-1}(t) - v_{i-1}(t)) - (z_i(t) - u_i(t)).$$

Since in CONSTIN the WIP transfer is not allowed during a period, the production start unit and the production unit completed at each operation is limited by the WIP available at the beginning of the period.

$$x_i(t) \leq w_i(t)$$
$$z_i(t) \leq w_i(t)$$

Since the production capacity of each workstation is limited, it is impossible to produce the products beyond the capacity constraint.

$$Sx(t) \leq c$$

3 COMPUTER SIMULATION

In a conventional event-driven simulation, every event that occurs in the manufacturing process must be carefully calculated and its cascading effects need to be propagated through the entire manufacturing process. On the other hand, in CONSTIN, the manufacturing process can be simulated only by calculating the WIP transitions for every operation in each period. Therefore, in CONSTIN, simulation can be executed much faster than the traditional event-driven method, and thus CONSTIN has a great advantage as a simulation method to be applied to the large-scale manufacturing process.

3.1 Simulation Algorithm

Computational simulation of the CONSTIN method is done by executing the simulation loop shown in the Fig. 2. In the codes of Fig. 2 there are two parameters, $Period$ and $EndOfSimulation$, whose values need to be determined before running simulation. Simulation runs till $EndOfSimulation$ at a fixed interval of $Period$. How to determine the value of the $Period$ parameter is explained later in this section. In setting the value of the $EndOfSimulation$ parameter, it must be large enough for the simulation results to be in the steady state. When a large value is set for the $Period$ parameter, CONSTIN takes long time to cumulate enough size of WIP at each workstation for steady production and the $EndOfSimulation$ value also needs to be set large.

```
for (each workstation in the fab)  {
  for (each operation at the workstation)  {
    wip -= processed WIP in the previous period;

    if (operation is the first operation)
      wip += releaseRule(operation);
    else
      wip += WIP inflow from the preceding operation;

    demand = wipTranferRule(wip, operation);
  }
  sortingRule(operations at the workstation);

  for (each operation in the sorted order)  {
    calcProduction(operation);
  }
}
```

Figure 3: Calculation of WIP Transition

In the main part of the simulation codes, the `runForPeriod` function, WIP transition at each workstation is calculated as shown in Fig. 3.

The WIP level of an operation is first subtracted by the amount of WIP that was processed by the workstation in the previous period. Then, if the operation is the first operation of the product, new release in the period is added to the WIP. The quantity of the release is determined by a rule (i.e., the `releaseRule` function in Fig. 3). By changing this rule, CONSTIN can simulate the push-type manufacturing system like MRP and also the pull-type manufacturing system such as JIT and CONWIP (Hopp and Spearman 2000).

For each operation the `wipTransferRule` function in Fig. 3 determines the amount of WIP to be processed in the present period by the workstation. To achieve leveled production among products, this function needs to strike a balance between production levels for all the products by taking into consideration the manufacturing process parameters such as inventory levels of neighboring operations, the amount of completed products, and resource utilization of adjacent operations among others.

When the amount of WIP to be processed in the present period is determined for every operation of the workstation, the `sortingRule` function in Fig. 3 sorts the operations based on their priorities and thus develops the sequence of the operations to be processed by the workstation. For determining a sequence of operations, traditional dispatching rules such as SPTF can be used in the `sortingRule` function.

When the `wipTransferRule` function determines the production levels that do not satisfy capacity constraints, the jobs in the latter position of the sequence may not be processed in the present period due to the capacity limitation of the workstation. Thus, there are complicated interactions among the above three rules used in the simulation.

Finally, the `calcProduction` function in Fig. 3 calculates the time necessary to process the designated amount of WIP for the workstation and updates utilization of the workstation. In the current implementation, the `calcProduction` function can cope with several types of semiconductor manufacturing practices such as batch processing.

3.2 Period Parameter

The most significant parameter to be set before running simulation of CONSTIN is $Period$. When a value of $Period$ is set large, CONSTIN shows high robustness against several variabilities in the manufacturing process such as machine failures. But the robustness comes with large WIP to continue processing. On the other hand, when the value of $Period$ is set small, CONSTIN's robustness decreases, and computational time for simulation increases. There is a clear trade-off to be considered in setting a value of $Period$. The value of $Period$ needs to be determined appropriately according to the purpose of simulation. The queueing theory provides a procedure for setting a reasonable value of $Period$ as explained in the following.

Assuming that there is only one type of product to be processed in the steady state of CONSTIN simulation, production at a workstation i in the duration of $Period$ is calculated as

$$z_i = rl_i d,$$

where z_i is production completed in the workstation i per period, r is a release rate of the product, l_i is the number of operations to be processed at the workstation i, and d is a value of $Period$.

In CONSTIN, the amount of production completed is always less than the WIP level before production (i.e., $\sum_{i=1}^{m} z_i \leq \sum_{j=1}^{n} w_j$). That is,

$$\sum_{i=1}^{m} rl_i d \leq \sum_{j=1}^{n} w_j.$$

Since the throughput rate of the product in the steady state is equal to the release rate r, Little's Law (Little 1961) in the queueing theory implies that

$$\sum_{j=1}^{n} w_j = ry,$$

where y is the expected cycle time of producing the product. Hence, from the above inequality, it is shown that

$$d \leq y / \sum_{i=1}^{m} l_i.$$

Although the value of l_i is evident from the modeled manufacturing process, a precise value of the expected cycle time y is difficult to estimate because the cycle time should include not only processing times but also waiting times, which are not explicitly known a priori. Hence, the expected cycle time tends to be managed so that the following equation holds,

$$y = \alpha \sum_{i=1}^{n} s_i.$$

Although α is always greater than 1.0, its value varies depending on the characteristics of the manufacturing process to be simulated. Then, from the above discussion, we have a provisional and reasonable estimate of d, the value of $Period$, as

$$d \leq \alpha \sum_{i=1}^{n} s_i / \sum_{j=1}^{m} l_j.$$

4 SIMULATING WAFER FABRICATION PROCESS

We conducted computational experiments to test validity of CONSTIN and its simulation system using realistic data of semiconductor manufacturing processes. As test data of wafer fabrication processes, we used the MIMAC (Measurement and Improvement of MAnufacturing Capacities) testbed datasets (Fowler and Robinson 1995), which are now maintained and made public by MASM lab in Arizona State University. For further details and downloads of dataset, see <www.eas.asu.edu/~masmlab/home.htm>.

As a preliminary investigation, we executed several simulation runs of CONSTIN using dataset 1 of the MIMAC testbed. Table 1 shows the properties of the test problem. It has basic characteristics of a semiconductor manufacturing process such as lengthy process flow with many repetitive reentrant loops and a couple of bottleneck workstations.

Table 1: Test Problem

Type of product	Non-volatile memory
Process flows	2
Products	2 (i.e., 1 per process)
Workstation groups	83
Workstations	265
Operations	210 (Product A) 245 (Product B)
Raw processing time (hour)	313.4 (Product A) 358.6 (Product B)
Demand rate (wafers/day)	380.95 (Product A) 190.48 (Product B)
Lot size (wafers)	48 (Product A, B)

4.1 Simulation Conditions

In this experiment, we made the following assumptions to focus our investigative attentions to the basic properties of CONSTIN and its simulation: (1) no WIP exists at the beginning of simulation, (2) there is no variabilities in processing times of operations, (3) no setup time is considered, (4) operators are not considered in the model, and (5) there is neither machine failure, product rework, nor scrap. That is, we have no stochastic factor in this simulation.

In the simulation codes for this experiment, `releaseRule` uses a constant releasing policy based on the fixed demand rate, `wipTransferRule` adopts a policy of processing all the remaining WIP at a workstation, and `sortingRule` assigns a higher priority to the operation that has more WIP units to be processed.

In this problem, the average raw processing time of a single wafer is about 8, 862 minutes and the average number of operations is 221.7. Then, by assuming $\alpha \geq 2.0$, we set a value of the $Period$ parameter as 80 minutes, which is a result of $8,862 \div 221.7 \times 2.0$ as is explained in Sec. 3.2). And for a value of the $EndOfSimulation$ parameter, we use 6 months in order to make sure that the simulation results reach a steady state.

4.2 Simulation Results and Analysis

Figs. 4, 5 and 6 show the results of the experiment in terms of the average output rate of wafers per day, average WIP level and average resource utilization, respectively. These simulation results show that CONSTIN satisfies the specified demand rates in a steady manner after about 1000 periods (i.e., 55 days) of production.

In order to verify the validity of CONSTIN, we compared the simulation results of CONSTIN with those of a commercial event-driven simulator, AutoSched AP ver.7.1 which has been widely used in the various semiconductor manufacturing fabs. Comparison of the simulation results is

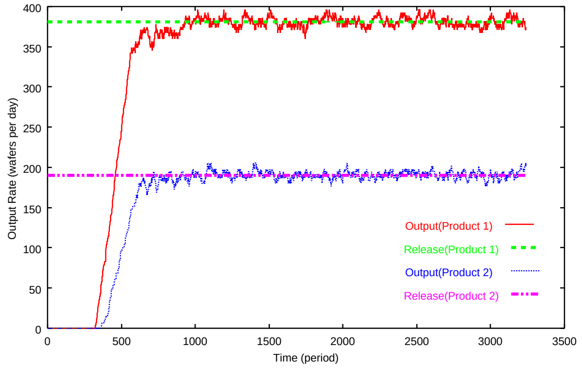

Figure 4: Simulation Result (Output Rate)

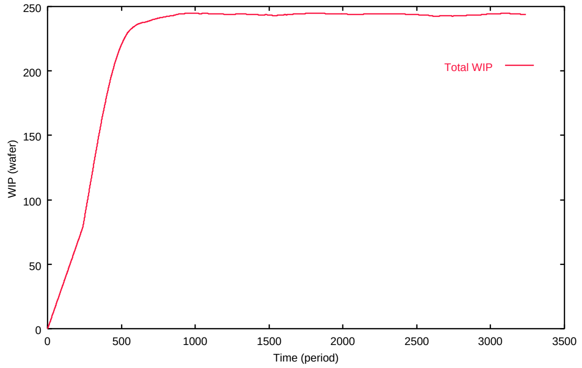

Figure 5: Simulation Result (WIP)

summarized in Table 2. Table 2 shows that CONSTIN and AutoSched are compatible in terms of the output rate and the resource utilization. However, as is expected, CONSTIN has a higher WIP level than AutoSched AP and runs quite faster than AutoSched AP.

The CPU times in Table 2 are the results of 6 months long simulation using a PC with 1.2GHz Pentium III CPU. In the experiment, CONSTIN is more than 20 times faster than AutoSched in required CPU time. CONSTIN's simulation speed increases almost linearly with a value of the $Period$ parameter. For example, when a value of the $Period$ parameter is set at 480, CPU time for the simulation is

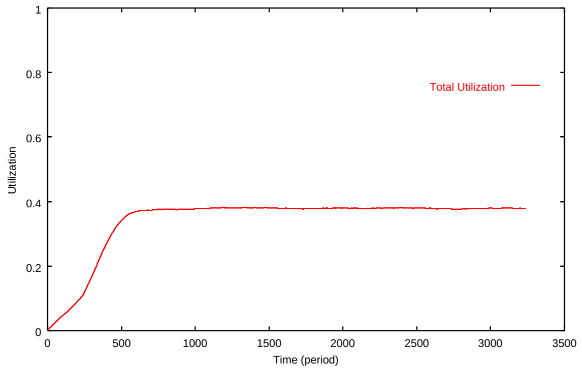

Figure 6: Simulation Result (Resource Utilization)

Table 2: Results of Preliminary Experiments

		CONSTIN	AutoSched
Output (wafers)	Product A	237	239
	Product B	122	120
WIP (wafers)	Product A	157	101
	Product B	85	62
Resource Utilization(%)		37.9	38.0
CPU time (sec.)		4.5	106

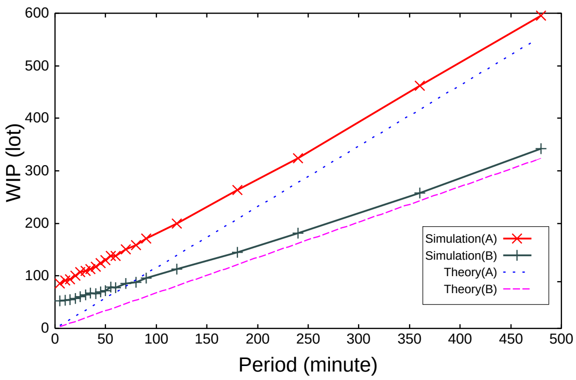

Figure 7: Relationship between $Period$ and WIP (Both Theoretical and Experimental Results)

about 1.0 second. Thus, it is possible to use the CONSTIN simulation system for the purpose of real-time scheduling.

As for the WIP level, it is inevitable that CONSTIN cumulates more inventory than the event-driven simulator since, in CONSTIN, WIP is allowed to move only at the end of each period. The cumulated WIP can be thought as a source of robustness in CONSTIN. Therefore, in using CONSTIN, it is important to set a value of the $Period$ parameter that strikes a balance between the allowable WIP level and the desirable production robustness.

Concatenated solid lines in Fig. 7 show the simulation results of the WIP levels for Products A and B with several $Period$ values. The results show that the WIP level increases almost linearly with the value of $Period$ parameter.

For mathematical analysis, let $W_p(t)$ be the WIP level of the product p in the period t, then we have

$$\begin{aligned} W_p(t) &= \sum^{n_p} w_i(t) \\ &= \sum^{n_p} (w_i(t-1) + z_{i-1}(t-1) - z_i(t-1)) \\ &= \sum^{n_p} z_{i-1}(t-1) + \sum^{n_p} (w_i(t-1) - z_i(t-1)). \end{aligned}$$

Suppose that the value of t is large enough for the simulation to reach a steady state. Since, in the steady state, a production output rate should be equal to a release rate, the value of $\sum^{n_p} z_{i-1}(t-1)$ is sum of the release at every operation of the product p in the period. And, since WIP levels remain constant in a steady state, the value of $\sum^{n_p}(w_i(t-1) - z_i(t-1))$ is a small value as compared with the value of $\sum^{n_p} z_{i-1}(t-1)$ when the duration of the period d is large. Therefore, we have

$$\bar{W_p} \approx \bar{r_p} n_p d,$$

where $\bar{W_p}$ and $\bar{r_p}$ represent average values of $W_p(t)$ and $r_p(t)$ over the periods.

Two dotted straight lines in Fig. 7 show the plots of the above equation for Products A and B. The graph exhibits the good correspondence between computational simulation and mathematical analysis.

Hence, a user of CONSTIN can easily set a reasonable value for the $Period$ parameter by estimating the resultant WIP level corresponding to the $Period$ value through preliminary simulation runs.

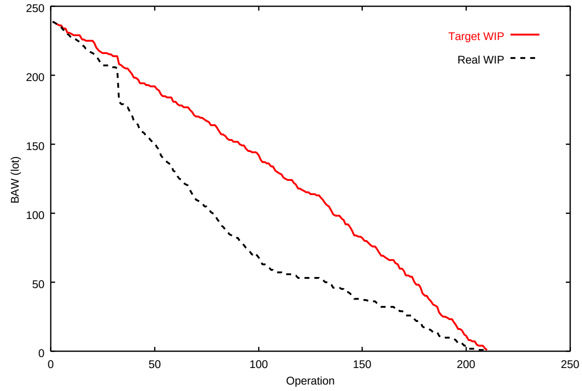

Figure 8: Backward Accumulated WIP

5 CONTROLLING WIP LEVEL

CONSTIN can flexibly adapt its behavior to designated purposes by tuning its three control rules. To show an example of such capability of CONSTIN, we apply it for controlling the WIP levels in semiconductor manufacturing.

5.1 Backward Accumulated WIP

In order to recognize the levels of WIP and its distribution over production process intuitively, we define a concept, *backward accumulated WIP* (BAW), as shown in Fig. 8. BAW of product p in its i-th operation in the period t is calculated as

$$\sum_{j=i+1}^{n_p} w_p^j(t),$$

where n_p is the number of operations for product p and $w_p^j(t)$ is a WIP level of the j-th operation of product p in the period t.

Target WIP in Fig. 8 plots the desirable BAW levels for each operation of the product. A desirable BAW level for the i-th operation of product p, W_p^i, is defined as

$$W_p^i = \sum_{j=i+1}^{n_p} \alpha s_p^j r_p,$$

where α is the parameter used to estimate the cycle time in Sec. 3.2 and s_p^j is processing time for the j-th operation of product p. Since α decides the levels of desirable WIP, it is also considered as a safety stock factor. And it is important to find an appropriate value for α that strikes a balance between lean and robust manufacturing. In the semiconductor manufacturing fabs we have observed, the value of α is often set between 3 and 5.

Real WIP in Fig. 8 is an example plot of actual WIP levels in a certain period. It shows a situation in which less than desirable levels of WIP is maintained in the latter operations of the product. A big gap in the Real WIP plot around the 30th operation indicates the existence of a bottleneck machine with large WIP. We assume that stable manufacturing is realized when real WIP at each operation of the product is maintained close to its target WIP value.

5.2 Control Rules for WIP Maintenance

In CONSTIN we have three rules, `releaseRule`, `wipTransferRule` and `sortingRule`, to control its behavior as explained in Section 3. To maintain the level of WIP as desired, we tuned those rules as follows:

1. `releaseRule` is determined to keep the total amount of WIP constant. New release does not occur until a product is completed. Therefore, BAW at the initial operation of a product always has the same value as the target BAW value of the product. The size of release for product p in the period t is calculate as

$$max(0, \sum_{j=1}^{n_p} W_p^j - \sum_{j=1}^{n_p} w_p^j).$$

2. `wipTransferRule` is tuned to compensate a gap of WIP levels in the succeeding operations of a product. The size of WIP to be processed in the i-th operation of product p in the period t is calculated as

$$\min(w_p^i(t), \max(r_p, r_p + \frac{\sum_{j=i+1}^{n_p}(W_p^j - w_p^j(t))}{n_p - i})).$$

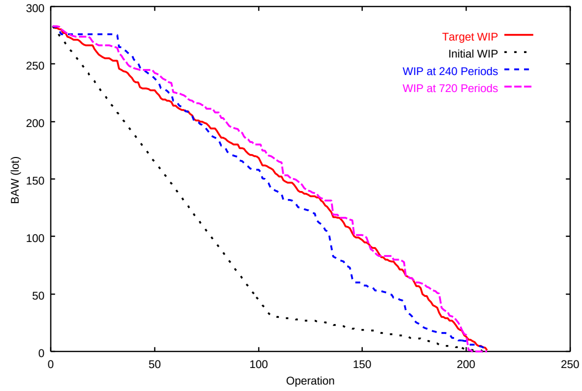

Figure 9: Backward Accumulated WIP of Product A

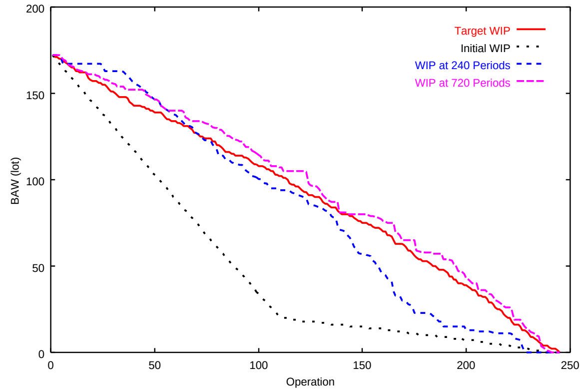

Figure 10: Backward Accumulated WIP of Product B

3. `sortingRule` is same as a default definition: it gives a higher priority to the operation with more WIP units to be processed.

Since the algorithm of CONSTIN is simple and its control rules are clearly modularized, a user of CONSTIN can modify the behavior of CONSTIN without complicated customization efforts.

5.3 Experimental Results

Simulation conditions for the experiments are almost same as those in Section 4.1. The differences from the previous experiments are (1) a value of *Period* is determined as 60 minutes, (2) a safety stock factor α is set as 5.0, (3) stochastic machine failures are modeled using an exponential distribution, and (4) demand rates are reduced to 360.54 wafers/day for Product A and 180.28 wafers/day for Product B. The demand rates are tuned to realize 98% resource utilization for a bottleneck machine.

Figs. 9 and 10 show the results of backward accumulated WIP of Product A and Product B. Initial WIP levels of the products are set as depicted in the graphs. The initial WIP levels represents the situation commonly seen in many semiconductor fabs, where WIP in the latter operations are exhausted by previous emergent, or *hot* orders. In order

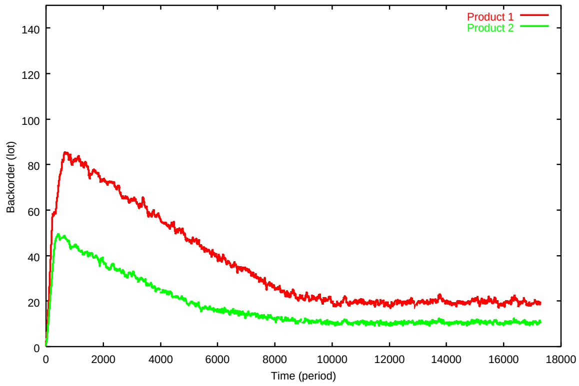

Figure 11: Backorders of Product A, B

to keep stable production, WIP level needs to be restored to the desirable level as soon as possible. Figs. 9 and 10 indicate that CONSTIN successfully rebuilds the desirable levels of WIP as the simulation goes on. In the experiments, at the period of 720 (i.e., after 30 days), the levels of WIP recovered to the desirable levels, and thereafter they remain steady.

Fig. 11 shows the transition of backorders for Products A and B. The graph indicates that at the beginning of simulation backorders grow in numbers due to the WIP shortage in the latter operations of the products. But, from the certain period (about 700 periods) the number of backorders starts to decrease and keeps on reducing until they s gets saturated at about 10,000 periods. This corresponds well with the results of backward accumulated WIP shown in Figs.9 and 10 in that restored WIP contributes to the reduction of backorders. And the reason why backorders are resolved only slowly and partially is extremely high utilization of the bottleneck machine (i.e., 98%). There is little capacity buffer available for catching up backorders in the experiments.

6 CONCLUSION

This paper shows that the constant time interval production planning and control methodology, CONSTIN, and its simulation system are an effective solution to large-scale and complicated manufacturing processes such as semiconductor fabrication.

For manufacturing processes comprising of stochastic elements such as machine failures and yield loss, it is difficult to keep stable outputs without a high level of WIP. However, unless its size is properly controlled, WIP results in lengthy lead times and increased obsolescence stocks. CONSTIN proposed in this paper forces each process to have WIP by restricting inventory movements between processes. Mathematical analysis and computer simulation show that in CONSTIN the size of WIP can be controlled linearly with respect to a value of the *Period* parameter, which is a duration of the time interval prohibiting inventory movements. For CONSTIN, a reasonable size of WIP can be autonomously set and kept for the manufacturing process by fixing a value of the *Period* parameter. Therefore, CONSTIN with a properly set *Period* value can realize robust manufacturing in the face of various variabilities of the manufacturing process.

Since simulation of CONSTIN needs to update simulation states only at the intervals of simulation time, it can be executed much faster than a conventional event-driven simulation, which is required to calculate a new state whenever any new event occurs. Hence, a user of the CONSTIN simulator can execute frequent what-if analysis to find preferable values of manufacturing parameters such as release rates and product-mix rates for her/his manufacturing processes.

Application of CONSTIN to the WIP control verifies that, with combination of appropriate rules, CONSTIN is effective to maintain the desirable levels of WIP. The three rules used in the simulation codes (i.e., `releaseRule`, `wipTransferRule`, `sortingRule`) are shown to have significant impacts on quality of the simulation results. The rules used for the experiments in this paper are of the simplest forms and their influence on the results are considered to be interrelated complexly. In a future study, it is necessary to develop more sophisticated rules to meets several requirements in the semiconductor manufacturing.

Since CONSTIN is to be used for production planning (i.e., to decide appropriate release rates, WIP levels and dispatching rules through runs of what-if analysis), its effectiveness should be further evaluated by detailed behaviors of the manufacturing processes executing CONSTIN's planning results. We are now developing the integrated system of CONSTIN with distributed shop floor controllers to verify that a large scale semiconductor factory can be controlled smoothly based on the CONSTIN methodology in the realistic situations.

ACKNOWLEDGMENTS

The authors would like to thank Mr. Hiroyuki Kashiwase at Hitachi Co., Ltd. for his efforts in the early phase of this research. And the authors are indebted to valuable discussions with Mr. Tsukasa Okazaki at Hitachi Higashi-Nihon Solutions Co., Ltd.

REFERENCES

Atherton, L. F., and R. W. Atherton. 1995. *Wafer fabrication: Factory performance and analysis*. Kluwer Academic Publishers.

Fowler, J., and J. Robinson. 1995. Measurement and improvement of mafucaturing capacities (MIMAC): Final report. Technical Report 95062861A-TR, SEMATECH, Austin, TX.

Gong, L., and H. Matsuo. 1997. Control policy for manufacturing system with random yield and rework. *Journal of Optimization Theory and Applications* 95 (1): 149–175.

Hopp, W. J., and M. L. Spearman. 2000. *Factory physics*. Second ed. McGraw-Hill.

Little, J. D. C. 1961. Proof of the queueing formula $L = \lambda W$. *Operations Research* 9:383–387.

Rose, O. 2001. The shortest processing time first (SPTF) dispatching rule and some variants in semiconductor manufacturing. In *Proceedings of the 2001 Winter Simulation Conference*, 1220–1224. INFORMS.

Wein, L. M. 1988. Scheduling semiconductor wafer fabrication. *IEEE Transaction on Semiconductor Manufacturing* 1 (3): 115–130.

AUTHOR BIOGRAPHIES

KAZUO MIYASHITA is a Senior Researcher of Intelligent Systems Institute at the National Institute of Industrial Science and Technology, and an Associate Professor of Institute of Computer Science at the University of Tsukuba. Prior to his current position, he spent ten years working for Production Engineering Laboratory of Panasonic. Dr. Miyashita received his B.E. and M.E. from the University of Tokyo, and his Ph.D. from Osaka University, respectively. He is a member of AAAI and IEEE CS. His e-mail address is <k.miyashita@aist.go.jp>, and his web page is <http://staff.aist.go.jp/k.miyashita/>.

KAZUYUKI SENOH is a graduate student in the Institute of Policy and Planning Sciences at the University of Tsukuba. You can send e-mail to him at <senoh90@shako.sk.tsukuba.ac.jp>.

HIROYUKI OZAKI is a graduate student in the Institute of Policy and Planning Sciences at the University of Tsukuba. You can send e-mail to him at <ozaki90@shako.sk.tsukuba.ac.jp>.

HIROFUMI MATSUO is a Professor of Institute of Policy and Planning Sciences at the University of Tsukuba. Dr. Matsuo received his B.E. and M.E. from Kyoto University and his Ph.D. from Massachusetts Institute of Technology. He is a member of INFORMS. His e-mail address is <mastuoh@shako.sk.tsukuba.ac.jp>.