A SCHEDULING RULE FOR JOB RELEASE IN SEMICONDUCTOR FABRICATION *

C. Roger GLASSEY and Mauricio G.C. RESENDE

University of California, Berkeley, CA 94720, USA

Received July 1987
Revised June 1988

This paper introduces a closed-loop job release mechanism for stochastic job shops where the main source of randomness is due to machine failure and repair. The release policy adapts concepts of inventory control to the context of job shop scheduling. The control mechanism, called starvation avoidance, is compared in simulation studies with other input control mechanisms producing favorable results.

production/scheduling * simulation * stochastic models

1. Introduction

The manufacturing of very large scale integrated (VLSI) circuits [6, 9] is perhaps the most complex manufacturing process found today. The first VLSI manufacturing stage is wafer fabrication, in which many devices are simultaneously built up in layers on wafers of silicon. Partly because of contamination problems, some of these facilities are among the few existing examples of *paperless factories* in which the status of machines and inventories is available in real-time from the factory's computer integrated manufacturing (CIM) system. Hence, real-time decisions about dispatching (which job to do next when a machine becomes available) and lot release (of raw wafers into the factory) could be based on the global factory state. In much of the literature on job shop scheduling [2, 10], it is assumed that dispatching decisions are based on local information about machine status and queues from which the job is to be selected. The possibility of making decisions based on an expanded information set raises two research questions: How should global information be summarized and used for decisions; and how much improvement over traditional methods can one expect as a result?

Our objective is to find decision policies that will be efficient on the frontier of delay (or inventory level) and throughput. By Little's Law, the expected inventory of work awaiting processing (WAP) is the product of average delay and flow rate. Consequently, for a fixed output rate, reducing WAP reduces time spent in the shop. However, reducing WAP will tend to reduce throughput by reducing the buffering between work stations so that machine downtime at any station will have a high probability of forcing idle time elsewhere due to lack of work and reducing batch sizes so a greater fraction of machine time is spent on set ups. In this paper we are concerned only with the first tradeoff.

Some features of wafer fabrication are not found in most other job shops. Because each layer of the wafer requires exposure of a photoresistive material through a mask (the process is known as photolithography) each batch of wafers makes many visits to the photolithography work station. Since this equipment is very expensive, this work station is often the bottleneck. A scheduling policy

* This research was partially funded by the Brazilian Council for the Development of Science and Technology (CNPq), Semiconductor Research Corporation, Philips Research Laboratories, Fairchild Semiconductor Corporation, Cray Research, and the State of California through PROJECT MICRO – Microelectronics Innovation and Computer Research Opportunities.

should focus on the queue of work at the bottleneck because it will on average be the biggest queue in the factory and furthermore, whenever the bottleneck work station is forced to become idle due to lack of work, that lost time represents an unrecoverable loss of final output.

For the remainder of this paper we require some assumptions about the production system we are studying. They are as follows.

- All random variables (i.e. time between machine failures and time to repair) are stationary.
- The desired output rate is constant (at least for intervals long compared to mean flow time).
- A single work station is the unique bottleneck for the shop and machine time is the limiting resource. In other words, there is a single work station that has the minimum proportion of idle time. For work station j, the proportion of idle time is given by

$$I_j = 1 - F \times \frac{W_j}{N_j A_j} \tag{1}$$

  where F is the average flow rate of new work into the shop (lots per hour), W_j is the expected work load (machine hours) at station j per lot, N_j is the number of machines at station j and

$$A_j = \frac{\text{MTBF}_j - \text{MTTR}_j}{\text{MTBF}_j} \tag{2}$$

  is average machine availability, where MTTR_j is the mean time to repair machine j and MTBF_j is mean time between failures of machine j.
- A single product is manufactured. This assumption is merely to simplify the notation and is otherwise irrelevant.

One conclusion we have reached from our work and that of others [3, 13] is that dispatching decisions are not as important as release decisions. Effective flow control requires a sufficiently large inventory of raw material so that new work can be introduced whenever desired. While such policies shift inventory from WAP to raw material, it is much less expensive to hold inventory in that form. The raw wafers can be made into any product, so the risk of obsolescence due to demand shifts or engineering changes is lower. They are not as subject to contamination and yield loss.

2. Work release control

The most common release control used in VLSI fabrication is the open loop strategy of *uniform starts* (UNIF), i.e. release a new lot of work into the shop at a constant rate equal to the desired output rate, and independent of current inventory levels or machine status. Throughput is controlled by varying the interval duration. A second release strategy is the *constant-WIP* (C-WIP) rule which starts a new lot of wafers whenever a lot of wafers is completed. Throughput, in this strategy, is controlled by adjusting the WIP level. Wein [13] has proposed a variation of the C-WIP strategy in which inventory is measured, not by counting wafers, but by summing the remaining work to be performed at the bottleneck work station. He calls this strategy *workload regulating* (WR) release. WR [13] monitors the sum of remaining processing time at the bottleneck resource for all jobs in the network. When this sum falls below a critical value a new job is released into the system. Throughput can be controlled by changing the critical value. Our proposed strategy is similar to Wein's but is derived from a somewhat different approach. We start with a simple idea: To reduce inventory, do not start new work. The difficulty with that idea is that eventually the bottleneck work station starves and no finished work leaves. Instead, we propose the *starvation avoidance* (SA) rule: Start new work just in time to avoid idling the bottleneck work station due to lack of work. Fredericks [5] suggests that controlling the number of lots in the pipeline from start to the bottleneck resource can lead to a reduction in work-in-process when compared to uniform starts, but gives no details of how this could be accomplished.

The problem of avoiding starvation of the bottleneck is one of inventory control, and the tradeoff is between inventory level in front of the bottleneck and stock out probability. In the classical inventory models for this tradeoff, the optimal policy is characterized by a single critical number, and replenishment orders are issued to keep the net stock position (on-hand plus on-order inventory) equal to that number. The lead time for replenishment L is the time required for new wafers to complete all operations before the first visit to the bottleneck. Since there are reentrant flows, we need to consider work that could arrive from other work stations in addition to new wafers. Hence we define *virtual inventory* at the bottleneck as the expected time required for the bottleneck station to process all lots in the set S of actual bottleneck inventory and all work-in-process at other work stations expected to arrive there within the lead time. We approximate virtual inventory by

$$W = \frac{D_S + E(R)}{N_B} \tag{3}$$

where D_S is the total bottleneck processing time of the next operation on all the wafers in the set S of wafers in the virtual inventory. $E(R)$ is the expected time to repair all bottleneck machines that are currently failed and N_B is the number of machines at the bottleneck work station. To keep the calculations simple. we take the elapsed time for a wafer to arrive at the bottleneck to be the sum of lot processing times of all intervening operations. This ignores queueing and machine down time at the intervening stations. Consequently L is the sum of processing times of all operations preceding the first visit to the bottleneck. and S is the set of lots whose processing times. summed from the current operation to the next visit to the bottleneck. is less than L.

If $W > L$. we would expect a newly released lot of wafers would reach the bottleneck before it runs out of work. Because of uncertainties in lead times and arrival of work from other stations we propose the *starvation avoidance* release rule: If $W < \alpha \times L$ then release a batch of new wafers into the shop. The positive control parameter α can be adjusted to attain different points along the tradeoff curve: larger values of α result in higher inventories (more safety stock) and lower starvation rates. For a detailed discussion of starvation avoidance see Glassey and Resende [7] and Resende [12].

3. Results

We report results of comparing SA with several scheduling policies on a number of wafer fab queueing networks. A scheduling policy is defined by a release control and dispatching policy pair (e.g. UNIF/SRPT is a scheduling policy that uses uniform (UNIF) release control and the shortest remaining processing time (SRPT) dispatching rule). Policies are compared on the delay/throughput tradeoff curve. Simulations are carried out using FabSim [11] a VLSI fab simulator, on the Cray X-MP/14 supercomputer at U.C.-Berkeley. All queueing networks process a single product and have a single bottleneck resource. Table 1 summarizes the fab networks tested in this study. For a detailed description of these networks, the simulation experiments and all dealy/throughput tradeoff curves see Resende [12].

Table 1
Test fab queueing networks

Fab network name	Number of stations	Number of Recipe steps
Small-Fab	5	12
Dayhoff-87	18	24
Bipolar	24	46
SSi-implant	41	182
SSi-aligner	41	182

The dispatching rule used in all simulation runs of Small-Fab was SRPT. Four release strategies were compared. namely UNIF, WR, C-WIP and SA. Within its range of throughput values (94–100% of expected capacity) SA release outperformed the other three scheduling policies with respect to the delay/throughput tradeoff. The improvement was most remarkable when compared to UNIF, where SA mean delay varied from 25% to as lows as 5% of the corresponding UNIF delay. Negligible differences were registered in the coefficient of variation (CV) of inter-departure times. CV of inter-departure times is a measure of the regularity of output. When compared to C-WIP and WR. improvements were not as marked as when compared to UNIF. Still. near capacity SA mainained its overall good performance with mean delays of 83% and 50% of those produced by WR and C-WIP. respectively.

UNIF release with first-in-first-out (FIFO) dispatching was compared to SA scheduling (SA release and SA dispatching) on Dayhoff-87 [4]. The SA dispatching rule combines two simple rules by means of weights, as in [8], and dynamically changes the weights according to the state of the system. If the bottleneck is in no danger of starvation the dispatching rule gives more weight to the shortest remaining processing time (SRPT) rule. while if there is imminent danger. more weight is given to a rule that gives high priority to lots that are headed for the bottleneck station. This rule is described in detail in Resende [12]. SA scheduling produced schedules having mean delays up to 3 times shorter than those produced by UNIF/FIFO in the throughput range of SA scheduling (87–97% of expected capacity). UNIF/FIFO. however. produced inter-departure times having about 5–8% less CV than those of SA scheduling.

SA scheduling was compared to UNIF/FIFO on Bipolar [1]. Once again. SA outperformed UNIF/FIFO in its range of throughput values (96–99% of expected capacity) with reductions in mean delay varying from 50 to 73%. SA scheduling. however. produced inter-departure time CV up to 30% greater than those of UNIF/FIFO.

SA scheduling was compared to UNIF/SRPT on both SSi-implant and SSi-aligner. In SSi-implant. where. unlike the other examples. the first visit to the bottleneck only occurs at step number 21. SA scheduling only shows a clear improvement over UNIF/SRPT for throughput values greater than 94% of expected capacity. Because of sampling variability there were no statistically significant differences between the two policies below that throughput rate. In the high end of this range (i.e. close to capacity) SA is able to produce schedules with approximately 40% less dealy than those produced by UNIF/SRPT. Inter-departure time CV differences are negligible throughout. For SSi-aligner the range of throughput values generated went from 98 to 100.2% of expected capacity. In this range SA scheduling produced mean delays varying from 88 to 65% of those produced by UNIF/SRPT.

4. Conclusions

Starvation avoidance was extensively tested on several fab networks and the following ranking of release strategies in order of increasing effectiveness: UNIF. C-WIP. WR and SA seems to hold for all the experiments we performed. We conclude the following about SA.

- SA is an effective scheduling policy in that it produces near-capacity throughput while maintaining the average job delay considerably lower than when traditional job release policies are used.
- Near capacity. SA outperformed all other policies in all cases tested. However, it should be noted that SA was not compared with WR and C-WIP on the larger networks-Bipolar, Dayhoff-87. SSi-implant and SSi-aligner.
- As is the case with inventory control, SA is sensitive to the randomness of the lead time. If the lead time from new wafer starts to the bottleneck work station increases in length or variability. not only will the tradeoff curve shift upwards (longer delays for any output rate) but the relative reduction in delay achieved by SA compared with UNIF will be not as dramatic. This is clearly seen in comparing the models SSi-aligner and SSi-implant. This is an example of a common phenomenon in control systems: Introducing time lags or noise in the feedback loop of a stochastic system will degrade performance. Randomness of lead time can be reduced by having the first bottleneck visit occur early in the process recipe.
- SA is easy to implement. provided a CIM system with up-to-date shop floor information is available. In practice. the throughput control parameter α used in SA must be set. This parameter controls the factory throughput. One way to set α is with simulation. This, however, requires a validated simulation model which may not be available. Another approach is as follows: Run the fab for a period of time releasing work with the UNIF control policy and measure the virtual inventory W. By using $\alpha = W/L$ as the control parameter setting (where L is the lead time for replenishment) then with high probability the throughput will be larger than the average throughput measured under the UNIF release policy. Since near capacity the slope of the delay/throughput curve for SA is steep. by reducing α one can still maintain a high throughput rate. while reducing considerably the mean delay. In practice, α can be decreased slowly. with careful monitoring of the corresponding delay and throughput rates, until a desired point on the delay/throughput curve is reached.

Future research should determine if the results for the single process. single bottleneck case described in this paper hold for the several multi-process. multi-bottleneck cases. One way to treat multi-process flows is as follows. Let L_i be the expected time required for a lot of type i ($i = 1, \ldots, p$) to reach the bottleneck for the first time. Let the lead time for replenishment

$$L = \max(L_i \mid i = 1, \ldots, p). \qquad (4)$$

Furthermore, define a start slack for product i to be $\delta_i = L - L_i$. Whenever the virtual inventory falls below the safety stock level, select the product (k) that is furthest behind its cumulative starts schedule for release. The cumulative starts schedule is derived from the outs schedule by shifting back by the lead time and scaling by an appropriate value to account for yield fallout. Release the lot after δ_k time units, but include it in the virutal inventory count immediately. In this paper we describe only one interpretation of the SA principle. With other definitions of virtual inventory (e.g. making use of improved lead time and equipment downtime estimates) other interpretations of SA can be investigated.

References

[1] M. Barry, Private Communication, 1986.

[2] J.H. Blackstone Jr., D.T. Phillips and G.L. Hogg, "A state-of-the-art survey of dispatching rules for manufacturing job shop operations", *International Journal of Production Research* **20** (1), 1982.

[3] D.Y. Burman, F.J. Gurrola-Gal, A. Nozari, S. Sathaye and J.P. Sitarik, "Performance analysis techniques for IC manufacturing lines", *AT&T Technical Journal* **65** (4), 1986.

[4] J. Dayhoff, "New techniques for simulation modeling and analysis", *CIM Review*, Winter, 1987.

[5] A.A. Fredericks, "Performance analysis modeling for manufacturing lines", *AT&T Technical Journal* **65** (4), 1986.

[6] P. Gise and R. Blanchard, *Modern Semiconductor Fabrication Technology*, Prentice-Hall (Reston), Englewood Cliffs, NJ, 1986.

[7] C.R. Glassey and M.G.C. Resende, "Closed-loop job release control for VLSI circuit manufacturing", Technival Report ORC 87-8, Operations Research Center, University of California, Berkeley, CA 94720, 1987.

[8] J.C. Hershauer and R.J. Ebert, "Search and simulation selection of a job shop sequencing rule", *Management Science* **21** (7), 1975.

[9] W.G. Oldham, "The fabrication of microelectronic circuits", *Scientific American* **237** (3), 1977.

[10] S.S. Panwalker and W. Iskander, "A survey of scheduling rules", *Operations Research* **25** (1), 1977.

[11] M.G.C. Resende, "Computer simulation of semiconductor wafer fabrication", Technical Report ORC 86-14, Operations Research Center, University of California, Berkeley, CA 94720, 1985.

[12] M.G.C. Resende, "Shop floor scheduling of semiconductor wafer manufacturing", Technical Report ESRC 87-1, Engineering Systems Research Center, University of California, Berkeley, CA 94720, 1987. Ph.D. Dissertation.

[13] L.M. Wein, Private communication, 1986.