Optimization of Work-Center Cycle Time Target Setting in a Semiconductor Wafer Fab

Hermann Gold and Hannah Dusch

Infineon Technologies AG, Wernerwerkstrasse 2, 93049 Regensburg

***Abstract*—In this paper the problem of assigning target cycle times at operation level in a semiconductor wafer fab, where target end-to-end-delays are given, is considered. In the original position allowed waiting times are assigned at processing stations proportional to the square root of processing times. We apply the fairness principle which claims that waiting times should be proportional to processing times at so-called machine resource pools. To match overall cycle time targets the normalization constants are adjusted using LP and QP methods.**

***Index Terms*—Queuing Networks, Optimization**

I. Problem Description

We consider a semiconductor manufacturing facility, called FAB, which resembles a traditional job shop with recirculation and in which single servers are replaced by complex work centers. There is a variety of products (with a corresponding index set $g \in G$, g mnemonic for good) which are produced in wafer lots, each lot being released into FAB with an associated process flow, or recipe, that consists of a prescribed ordered set of operations.

FAB is part of a complex supply chain (SC). At the interface between SC and FAB managers negotiate a loading for FAB for a certain time period, typically week, thus defining the number of wafer starts per week ($wspw$) and per product. With the acceptance of the loading, which may be only part of what SC managers would demand, the manager of FAB promises to deliver the loaded wafer lots (or jobs) with a pre-specified due-date lead time derived from fab cycle time. Let the sum of all processing times for product g arising in FAB be $b_g^{(tot)}$ (total processing time for product g). Herein processing times are considered to be fixed at their expected values under an optimal routing scheme. Assume that there is a number of exogenous priority corridors with index set P, each external demand being assigned to a particular priority corridor according to negotiation. Then the following should hold.

R1: The fab cycle time $S_{p,g}^{tot}$, defined by the total time spent in Fab by a wafer lot, for some product g released in a given priority corridor p, is proportional to its respective total processing time $b_g^{(tot)}$, with a proportionality factor c_p dependent on priority corridor p, but independent of product index g, hence $S_{p,g}^{(tot)} = c_p \cdot b_g^{(tot)}$.

Typically, there are few priority corridors, around 3 to 6, e. g. rocket lots, hot lots, normal lots, and as many as a few hundred products, in the different priority corridors. Different quantities of the same product can be assigned to different priority corridors in a given time period. In this research we consider the problem of breaking these fab cycle time targets down to cycle time targets for the individual operations of each combination of product and priority corridor. Normalized cycle time and waiting times at operations step level are mathematically denoted FF_l and $FF_l^{(W)}$, respectively, $l \in L_g$ the set of operation steps of product g. Hereby, as usual in semiconductor manufacturing, FF stands for flow factor.

II. Original Target Setting and Fairness

The original position from where we started our work on cycle time and flow factor target setting was established by purely inductive reasoning. Its essence is that normalized waiting time FF_W at a given process step should be inverse proportional to the processing time of the step. Mathematically, this original position is formulated as follows:

$$\textbf{OP:} \qquad \mathrm{FF}_W \sim \frac{1}{\sqrt{b}} \Rightarrow \text{waiting time } w \sim \sqrt{b} \qquad (1)$$

Relation (1) is extended by an offset, which is not considered important for this report. However, it is important to note that further linear scaling of the set $\{\mathrm{FF}_{pgl}^{(W)},\ p \in P, g \in G, l \in L_g\}$ for each product with a distinctive overall target X-Factor is required, since (1) is oblivious of the effect of folding the series of individual step cycle times resulting from individual step flow factors for each such product and for the different priority corridors.

There are numerous arguments to defy the original position, all brilliantly speculative, but one - this is a truth: The relaxation time as a measure for the speed of approach to the stationary situation for a stable queueing system is proportional to the service time divided by $(1-\sqrt{\rho})^2$, ρ being the utilization level in a single server approximation of the work center serving the process step under consideration. In attempt to smoothen material flow in FAB, (1) was originally designed to give higher priority weight to longer jobs as it would occur in a time-dependent priority system according to [6]. However, in the meantime it became clear, that prioritization of long jobs can significantly increase variance and decrease efficiency in terms of average waiting times with a negative impact on on-time-delivery by FAB, when service times are highly variant (see [1]). This is an argument against **OP**.

We conclude that the principle (1) is a robust guideline for cycle time target setting from managers' viewpoint, but it should be compromised for the counterargument given above and for the sake of reduction of complexity. In this work we follow a rigorous approach given in the following.

R2: For some fixed pair of customer priority corridor p and work center r we require that in any operation step of corridor-p-customers visiting work center r the expected waiting time is proportional to the expected service time with a common proportionality factor $c_{p,r}$, which is denoted as (target) flow factor for priority class p at work center r, $\mathrm{FF}_{p,r}$ being the associated variable. That is to say, at the work center level we deploy the same proportionality principle as at the fab level (see **R1** of Section 1).

The rigor of our approach becomes obvious under the categorization of the regimes presented with respect to fairness. To the authors' best knowledge there are three fairness measures which can be considered the best investigated ones: i) variance of stationary waiting times, minimum variance is best. ii) waiting time proportional to service time and iii) quality of load sharing (see [1]). Obviously, the fairness measure invoked by ii) is underlying requirements **R1** and **R2**. A closer look reveals that the original position **OP** has its closest acquaintance with fairness measure i) since smaller target flow factors for processes with larger processing times tend to flatten out the differences in absolute target waiting times, though not rigorously. Requirement **R2** is achieved in expectation in symmetric queues and it is this property which the load sharing concept deployed in FAB aims at (see [4]).

III. FF Targets Optimization

A. Pooling of Resources

We are now going to construct a Kelly type network of FAB. Central to this construction is a load sharing algorithm which is designed in the following way. To begin with, we build sets of communicating servers, which we call closed machine sets (CMS). A closed machine set is a minimal set of machines from which load neither can be shifted from its inside to its outside nor in the reverse direction. To formalize this, we consider aggregations of jobs, called job classes, characterized as follows. Two jobs belong to the same job class if and only if they have all parameters relevant for the determination of machine utilization levels in common. We call these parameters first-order parameters in accordance with stochastic processing network modelling practice. First-order parameters do not allow to capture the effects of statistical variability. In a semiconductor manufacturing network they include: 1) the subset of machines on which a given job class can be processed. This is referred to as dedication by practitioners. 2) the machine-dependent processing times of a job class. 3) the maximum sizes of batches of wafer lots that can be served during one service period in case of batch service, typically occurring at furnace operation steps, which can also be machine-dependent. 4) machine internal process flow alternatives, described by Boolean expressions. Machines featuring this possibility are called cluster tools. Job classes are associated with an index set $j \in J$. For easier notation we make job class indices unique over all CMS in the sequel. Hence with each $l \in L_g$, for some $g \in G$ a unique job class j is associated.

Now, two machines x and y are in the same closed machines set if and only if there is a chain of job classes $z_1, \dots, z_n$ and a set of machines $m_1, \dots, m_{n-1}$ such that z_1 can be processed on x and m_1, z_i can be processed on $m_{\imath-1}$ and $m_\imath$, for $\imath = 2, \dots, n-1$, and z_n can be processed on m_{n-1} and y. Within each CMS we build disjoint subsets of resources, called resource pools, inside each of which load can be distributed in a way such that all its servers are homogeneously loaded. The load levels are determined according to the well-known lexical difference principle defined by J. Rawls, where in our setting the gain for which machines compete is idling. This is favourable for waiting time reduction in queueing systems. The corresponding algorithm is described in [4] and references therein. Henceforth resource pools are associated with an index set $r \in R$. According to fairness measure iii) the resource pool concept provides a guideline to achieve efficiency in terms of average waiting times since ”it tends not to allow any server [of a given resource pool] to be idle while there are jobs awaiting processing in front of it”. Pooling of resources along these lines gives rise to approximate FAB by a Kelly type network. Resource pools generally coincide with work centers.

B. Work Conservation Laws

Let us recall, that in the original position the flow factors for the different operation steps, products and priority classes are considered to be completely static over all time, independent of product mix and volume release and the consequent machine utilization levels.

Implicitly, by observing FAB dynamically in time, these static numbers result in a target profile of unfinished work, since each lot of the work in progress (WIP) is tagged with a specific waiting time allowance. To improve target setting this profile should not be taken as a surprise, instead we need to acknowledge the amount of unfinished work as an important characteristic in the planning phases. A natural choice for the aggregation of unfinished work is the set of resource pools. Considering the system in the original position as a time-dependent priority system we apply the $M/G/1$-conservation law (see [5]), hereby approximating resource pools as one-server systems.

Some of the conditions which are prerequisite for the validity of M/G/1 and GI/G/1 conservation laws are crucial in semiconductor manufacturing. Firstly, a (resource pool) server must always be busy if there are jobs queueing in front of it. The optimal nominal plan is considered to provide a guideline for putting this fairly good into scheduling practice. Secondly, we need to be careful about the condition of arrivals to be Poisson. From measurement and statistical analysis we know that this condition is sufficiently fulfilled for large resource pools, e.g. like photolithography. For small resource pools with just one or two machines, arrival processes are typically more generally distributed, but from conservation law extensions as reported in [2] it is known that the law still holds in that case when service times for all classes are equal. If this is not the case the conservation law is an approximation. The condition that work can neither be generated nor destroyed is somewhat crucial when significant job-class dependent and sequencing dependent setups are required, i.e. at implantation work centers of FAB.

In the application we differentiate the conservation law with respect to exogenous priority corridors. Let P_g be the set of priority corridors having product g as a member. For some pair (p, r), $p \in P$, $r \in R$, let $T(p, r) = \{(g, j)/p \in P_g \wedge g$ visits r as job class $j\}$. Relating to the original position **OP** (indicated in the upper index of the flow factor, namely $\mathrm{FF}^{(W,OP)}$), we get

$$C_{p,r} = \sum_{T(p,r)} \rho_{gj} * \mathrm{FF}^{(W,OP)}_{pgjr} * \mathrm{b}_{gj} \tag{2}$$

With fairness principle ii) applied within priority corridors $\mathrm{FF}^{(W)}_{p,g,j,r}$ depends only on p and r, yielding $\mathrm{FF}^{(W)}_{p,r}$ according to the requirement **R1** of Section II.

$$\begin{aligned} C_{p,r} &= \mathrm{FF}^{(W)}_{p,r} * \sum_{T(p,r)} \rho_{gj} * b_{gj} \\ \mathrm{FF}^{(W)}_{p,r} &= \frac{C_{p,r}}{\sum_{T(p,r)} \rho_{gj} * b_{gj}} \end{aligned} \tag{3}$$

C. LP and QP Programs for Flow Factor Target Setting

With the flow factor unifications on the basis of the conservation law for a given exogenous priority corridor the sum of the cycle times of a given route is no longer perfectly adapted to its overall target cycle time, in general. Target flow factors will be recaptured by optimization via LP and QP. Using an LP formulation target cycle times per priority group can be fulfilled only with early delivery for some products. The advantage of LP solutions is that the burden of changes in target flow factors is put fairly on all shoulders. The basic LP program for some $p \in P$ is the following:

$$\begin{aligned} &\text{Minimize} && -\delta && \\ &\text{subject to} && && \\ &B \cdot (FF + \Delta) && \leq \quad S, && S = c_p(B \cdot \mathbf{1}) \qquad (4) \\ &-\Delta && \leq \quad -\delta\,\mathbf{1} && \qquad (5) \\ &FF + \Delta && \geq \quad \mathbf{1} && \qquad (6) \end{aligned}$$

Hereby B is a $|G| \times |R|$-dimensional matrix, with $b_{g,r}$ being the service time requirement of product g at resource pool r, FF is the vector of unified resource pool target flow factors, Δ is the vector of changes in flow factors and $\mathbf{1}$ is an $|R|$-dimensional vector of ones. The optimization goal is to minimize the maximum change (5) while avoiding any lateness (4). Constraint (6) guarantees that waiting times are greater or equal 0. The final solution will be found by a cascade of such LPs leading to a set of δ's values each of which is assigned to a set of resource pools. The algorithm is applied with exogenous priorities only (LPPRIO1) or with exogenous and endogenous priorities (LPPRIO2). Minimizing the sum of squared deviations from unified flow factors and replacing the $\leq$-sign in constraint (4) by an equality sign leads to a QP optimization problem (omitting (5)).

The results for a typical instance of FAB with just one priority corridor are summarized in Table 1. It contains statistics both on the resulting vector Δ and on the relative deviation D between sojourn time $S^{'}$ which is achieved in the final solution according to LHS-values of (4) and target sojourn time S, given by $(S - S^{'})/S$ for each $g \in G$. As can be seen in column Max regarding D, the match of $S^{'}$ with S is within 7.2% when algorithm LPPRIO1 is applied. Usually a deviation of more than 5% for any $g \in G$ would not be tolerable. Using LPPRIO1 this tolerance is slightly violated, but the main reason for LPPRIO1 to be disqualified for practical use are the huge enlargements for flow factors for some resource pools which are proposed through it, as can be seen in the results regarding Δ. The reason is that with no further endogenous priorities for some given exogenous priority p too many resource pools are imposed with a relatively high negative δ. Relieving this by using LPPRIO2 the target match between $S^{'}$ and S is easily achieved at the cost of some increase in complexity (number of endogenous priorities en), but complexity, abbreviated with CO in Table 1, is still only 10.4% of the complexity associated with the original position. Finally, using QP the minimum value of Δ of -1.47 is extremely unfair and therefore the QP solution is of no practical use. More details on algorithms and results presented in this section are reported in [3].

		LPPRIO1	LPPRIO2	QP
Δ	Mean	0.882	$7.06 \cdot 10^{-4}$	0.349
	Median	0.125	0	0.295
	Max	48.2	0.0379	0.802
	Min	-0.0673	0	-1.47
D	Mean	2.91	0.985	0
	Median	1.7	0.809	0
	Max	7.22	3.68	0
Prios	en	1	58	1
CO over OP		0.004	0.104	0.004

Table 1. Summary of results for LP and QP optimzation

IV. Conclusion and Outlook

In this paper we considered the problem of cycle time target setting in a semiconductor fab. This kind of target setting is a device to cope with the stochasticity of the related queueing network so as to achieve due date delivery. The aim was to reduce the complexity of this system with its over 50000 subtargets for a wafer fab in order to provide easier guidance and managability of the system.

The analysis of the system at the start of this work revealed that a main lever towards this aim would be to derive work load estimates at the level of resource pools which are easier to control and evaluate than the numerous individual subtargets. Though it was yet possible to aggregate flow factors for individual lot arrival streams via averaging them within pre-defined categories in multiple dimensions in the initial position, tagging them with their individual load contribution factors has made them amenable to the application of Kleinrock's work conservation law for queueing systems. Thus, given that individual subtargets were achievable, we could provide correct resource pool work load estimators. In cases where they were not, we could at least improve those estimators significantly.

Our main idea realized in this work was to homogenize global and local cycle time targets by applying the same fairness principle locally and globally, which is formalized mathematically in a corresponding optimization scheme. By use of a cascaded Linear Programming approach we provided a new target setting system which has huge advantages over the old one. First, it reduces the complexity in terms of the number of subtargets to only 10% in comparison to the old system and thus features higher transparency from the very beginning. Second, since the system is designed around resource pools, it lends itself to easier, computationally efficient, queueing network analysis.

In future work the work load formula used in Section III-B and the optimization scheme will be refined, in that characteristcs from resource pool operating curves are incorporated in the related procedures. The main challenge here is to adapt the analysis of multi-server-multiple-queue systems for use in multi-server systems with dedicated and discretionary traffic streams. Given this, we will focus on cycle time ditributions analysis. In particular we need to provide the 95%-Percentiles of product cycle times which are very important in practical use.

V. Additional Information

The work presented here has been done while the co-author was employed at Infineon Technologies AG. Her contribution to this paper is part of her master thesis done at OTH Regensburg and Infineon Technologies AG. She is now working with Osram Opto Semiconductors GmbH, Leibnizstrasse 4, 93055 Regensburg.

E-Mail:
Hannah.Dusch@osram-os.com
Hermann.Gold@infineon.com

References

[1] Avi-Itzhak B., Levy H.: On Measuring Fairness in Queues. Adv. Appl. Prob., Vol. 36, 919-936 (2004).

[2] Bolch, G., Greiner, S., de Meer, H., Trivedi, K.S.: Queueing networks and Markov chains, John Wiley & Sons (1998).

[3] Dusch, H.: Komplexitätsreduktion und Optimierung der Durchlaufzeitziele in einer Halbleiterfabrik [Master Thesis], Regensburg: Technische Hochschule, 2017.

[4] Gold, H.: Why our company uses programming languages for mathematical modeling and optimization, In: Kallrath J., Algebraic modeling systems, 161-169, Springer, 2012.

[5] Kleinrock, L.: A Conservation Law for a Wide Class of Queueing Disciplines, Naval Research Logistcs Quarterly, Vol. 12, 181-192 (1965).

[6] Kleinrock L., Finkelstein R.P.: Time Dependent Prioritiy Queues, Operations Research, Vol. 15, 104-116 (1976).