Proceedings of the 2017 Winter Simulation Conference
W. K. V. Chan, A. D'Ambrogio, G. Zacharewicz, N. Mustafee, G. Wainer, and E. Page, eds.

QUALITY BASED SCHEDULING FOR AN EXAMPLE OF SEMICONDUCTOR MANUFACTORY

Dirk Doleschal
Elisa Sophie Schöttler

Electronics Packaging Laboratory
Technische Universität Dresden
D-01062 Dresden, GERMANY

ABSTRACT

Quality is an important measurement within a semiconductor manufactory. Due to the fact that yield is directly affected by quality of the manufacturing process, in this paper a quality based scheduling approach will be presented which compares different methods like dispatching, MIP and CP, regarding different objectives. To test the different used methods a benchmark model of a semiconductor manufactory is build up. Here a lithography work center is used in detail where the rest of the fabrication is only build up as a delay station. With this model the repeatability for the example of a lithography step is investigated. Thereby in this investigation it is assumed, that each lithography tool has an offset which is transferred to the structure. Now the quality of a product should be best, if the offset from one layer to the next layer is minimized.

1 INTRODUCTION

Mathematical methods are used in operations research to find best possible decisions exemplarily for applied business economics. Due to this they are used for optimization of different problems. One kind of problems are scheduling problems where typically existing jobs have to be allocated to available resources within a given time period.

The semiconductor industry is a manufacturing with a high potential for optimization and furthermore the importance of scheduling has increased. The modern semiconductor manufacturing process is described by May and Spanos (2006) as “the most sophisticated and unforgiving volume production technology that ever has been practiced successfully”. During the production of a chip, which could last for up to three months, several hundred process steps are passed. High investigation costs for the production tools require a high utilization and an efficient manufacturing. Short innovation cycles and an increasing mix of products also increase the cost pressure for the production (Chen and Wu 2007, Doleschal et al. 2015).

Due to the high complexity and lot of restrictions within the production (i.e. cycles, batch tools, …) it is understandable that the scheduling of semiconductor manufactories is found relatively often in literature. It is interesting that typically only one of the two direct influences to the production costs is investigated: the throughput. Especially in combination with the scheduling an optimization of the second factor – the yield – is rarely found in literature. Also the integration of quality, a further success factor, is given little attention (May and Spanos 2006). Due to this fact it is investigative to find out, in which ways the aspects quality and yield can be integrated into scheduling to optimize basic objectives (i.e. low cost and high quality) for semiconductor manufacturing. May and Spanos (2006) divides quality into “quality of design” and “quality of conformance”. Thereby quality of design is influenced by the right selection of object characteristics whereby the quality of conformance defines how well an object agrees with its design specifications. In this investigation the quality of conformance is used as objective. This quality

978-1-5386-3428-8/17/$31.00 ©2017 IEEE

parameter is influenced, among others, by the production process as well as the state of the producing machines and other secondary resources.

As already mentioned the integration of quality aspects into scheduling of a semiconductor manufacturing is only rarely found in literature. Yugma et al. (2015) presented a literature study where they come to the result that the consideration of different quality and process data is important for production planning and will become more important. However, hardly any attempts have been made to bring these aspects together. In their investigation they describe a "wafer quality index" which is defined by the wafer and machine condition for each machine allocation. A quality based scheduling investigation can be found in Doleschal et al. (2015), where each machine out of a parallel machine work center gets a "health factor" and the products are differently influenced by this health factor. Here a capacitive based mathematical model was compared to different dispatching rules. Further investigation can be found in Obeid et al. (2012), where also a parallel machine work center is investigated. Starting from a mathematical model a "yield centric heuristic" and a recursive version of this is developed. The goal was to optimize, among others, a weighted sum of the cycle time and the expected yield. Already in 1995 Srinivasan et al. investigated the correlation between yield and waiting time. A similar investigation can be found in Colledani et al. (2015). Further work regarding time restrictions can be found for example in Qiao et al. (2014) and Li and Li. (2007). Klemmt and Mönch (2012) for example take five different classes of time restrictions into account and compared a list based heuristic with a mixed integer based optimization for several randomly generated test data. A sole empirical relationship between cycle time and yield could not be proven by Cunningham and Shanthikumar (1996). This implies that there are further criteria for the improvement of quality.

In this work an investigation on a quality based repeatability is done. For this different methods are compared against each other. These methods are different dispatching rules, mixed integer programming (MIP) and constraint programming (CP). The paper is structured as follows. In section 2 the problem is described in detail. Section 3 presents the used simulation model and investigated methods and is defined as the main part of the work. In section 4 the buildup of the test data is described and after this in section 5 the results of this investigation are presented. A short conclusion and outlook is given in section 6.

2 PROBLEM DESCRIPTION

During the manufacturing process of a semiconductor chip 300 – 700 production steps on more than 100 machines are performed (Mönch et al. 2013). In general the semiconductor manufacturing process can be divided into three parts: the frontend, wafer test and backend. Thereby in the frontend the functional layers are deposited. In the wafer test each chip on a wafer is tested on functionality and finally in the backend the separated chips are packed und furthermore tested. Whereas the frontend is dominated by cycles in which the main process steps layer production, photo lithography, doping and structuring are carried out, the wafer test and the backend process are rather linear. The main process steps in the frontend are accompanied by further process steps like measuring, cleaning and planarization steps. This general process flow in the frontend is shown in Figure 1.

For a quality based investigation, many quality factors could be assumed. Some of these factors are:

- Repeatability within cycles like shown in Figure 1.
    - Here we assume that two adjacent layers are positioned as exactly as possible above the other on a wafer and the producing machines have an specific offset
- Health factors for each machines of a parallel machine work center like investigated in Doleschal et al. (2015).
    - Here it is important that the different products are influenced different by this health factor
- Time couplings at special operations within the production process
    - The quality effect may happen due to chemical processes which occur during the waiting time or due to impurities.

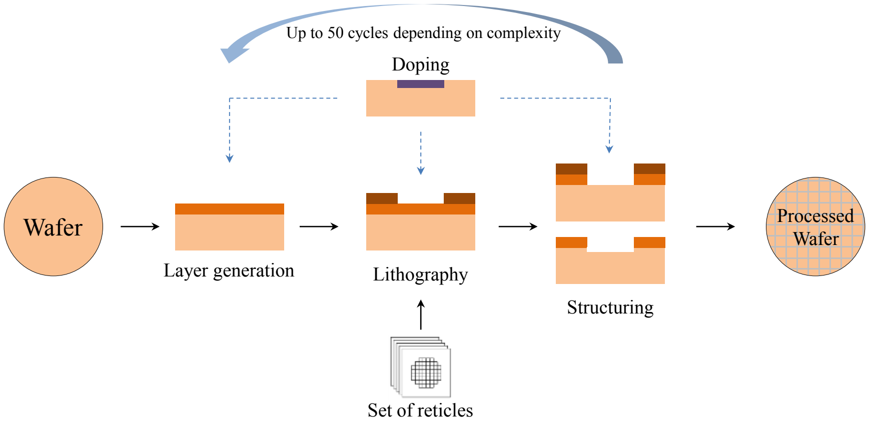

Figure 1: General production process for the frontend

As already mentioned in this investigation the repeatability is used as a quality factor. So here we use the lithography step as basis. Additional process restrictions in this area are the availability of secondary resources like reticles. These reticles are needed to transfer the structure to the wafer and typically these reticles are very limited due to the high costs. Furthermore, setups can occur if the reticle has to be changed on a machine. Also a lithography tool is assumed as a single processing tool, which means that only one job can be processed at a time.

Due to the fact that this investigation is focused on the quality aspect of the repeatability within a lithography work center, the rest of the frontend process is only considered as a delay station and therefore not described more in detail. Additional to the lithography tools jobs from different products exists. These products have predefined routes with a specific number of cycles. Also the type of a reticle and the process time within a cycle of a route are defined for each product and process step. Furthermore, the number of available reticles have to be defined in the input data. Each job consists of a release date which defines the earliest possible start and a due date which should be held. Furthermore, jobs can have different numbers of wafers included (max 25). This number has an effect on the processing time in the lithography step.

Therefore the considered problem includes:

- Set of machines M
  - Offset of machine $m \in M$
- Set of secondary resource types R
  - Number of available secondary resources of type $r \in R$
- Set of different products P
  - Route for product $p \in P$
  - Number of cycles/process steps I_p for product $p \in P$
  - Process times for each step $t_{i,p}$ ($i \in [1,\ldots,I_p]$) of the route for product $p \in P$
  - Type of limited secondary resource $r \in R$ needed for process step p_i
- Set of jobs J
  - Release date r_j for job $j \in J$
  - Due date d_j for job $j \in J$
  - Count of wafers w_j for job $j \in J$
  - According product p_j for this job

2.1 Quality Parameter

Typically the offset of a lithography tool can be described by the following parameters:

- X-Y offset – This value describes the offset in the X-Y plane
- Z offset – This value has an influence on the sharpness and the size/scale of the resulting structure
- Rotation – This value describes the rotation of the structure in the X-Y plane
- Tilt – Here the distance between reticle and wafer is not equal for all corners and edges

For this investigation mainly the X-Y offset and the rotation has an impact. The other two parameters could be rather classified into a health factor orientated scheduling.

For simplification the offset for the repeatability is normalized in the range between [-1;1]. In this case this simplification should be accurate enough. Also the offset for a machine can be varied over time. Now the goal is to process each layer of a job according to his previous layers.

For this each job $j \in J$ gets a history h_j of offsets. This history is updated each time when the job passes a machine.

In the model each machine has an offset O_m. This offset is transferred to a job $j \in J$ if this job is processed on this machine. So each job also gets an average offset O_j during processing. This offset is calculated by the average offset of all layers $O_{j,z}$ (z=1,…,n_j), where n_j is the number of cycles I_p of the according product p_j. This offset is equal to the offset of the processing machine (O_m) for this layer of a job. So the average offset O_j for each job $j \in J$ can be calculated by:

$$O_j = \frac{1}{n_j} \cdot \sum_{z=1}^{n_j} O_{j,z} \tag{1}$$

Also the maximum deviation D_j of all offsets of a job $j \in J$ could be calculated by:

$$D_j = \max(O_{j,1},...,O_{j,n_j}) - \min(O_{j,1},...,O_{j,n_j}) \tag{2}$$

From the offset of each layer of a job $j \in J$ the average fluctuation F_j could be described as a further quality parameter:

$$F_j = \frac{1}{n_j - 1} \cdot \sum_{z=2}^{n_j} | O_{j,z} - O_{j,z-1} | \tag{3}$$

Now the overall quality objectives are the average maximum Deviation D and the average fluctuation F:

$$D = \frac{1}{|J|} \cdot \sum_{j \in J} D_j \tag{4}$$

$$F = \frac{1}{|J|} \cdot \sum_{j \in J} F_j \tag{5}$$

2.2 Objectives

Further objectives in this investigation are the average flow factor FF, the average tardiness T and the overall setup time S. Thereby the flow factor is only calculated for the lithography work center due to the fact that the rest of production is denoted as a delay station. So the flow factor FF_j for a job can be described as:

$$FF_j = \frac{TPT_j}{RPT_j} \tag{6}$$

Where RPT_j is the raw processing time of job j and TPT_j is the total processing time (waiting time + RPT_j). Then the overall flow factor is the average over all flow factor values for each job:

$$FF = \frac{1}{|J|} \cdot \sum_{j \in J} FF_j \tag{7}$$

The overall tardiness T is defined as:

$$T = \frac{1}{|J|} \cdot \sum_{j \in J} \max(C_j - d_j, 0) \tag{8}$$

Here C_j is the completion time of job j.

S is defined as the average setup time over all machines for the whole considered time horizon.

These non-quality based objectives are necessary to evaluate the quality based methods and parameter from the viewpoint of scheduling.

3 INVESTIGATED METHODS

For this problem a simulation model is built up. The scheme of this model is described in Figure 2.

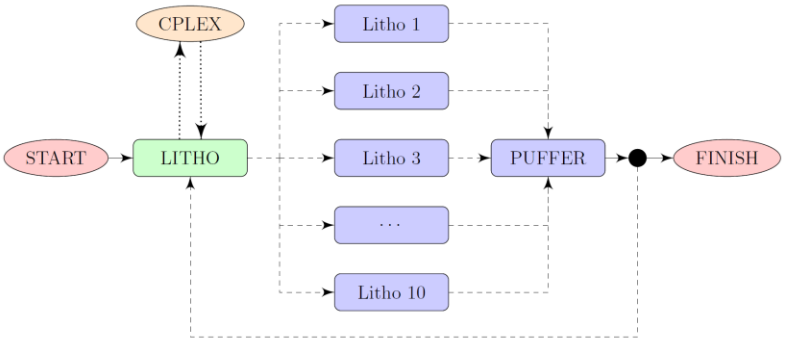

Figure 2: Scheme of the simulation model

The simulation model was built up using the simulation system simcron MODELLER 3.2. Within this simulation model all restrictions described in section 2 are built in. Furthermore the simulation model is coupled with scheduling methods like MIP and CP. In this cases the scheduling methods are executed cyclic from the simulation model and the result from this methods is again used within the simulation model. These methods also gain a forecast which is equal to their cyclic execution time. This is due to the fact that otherwise a machine could be empty even if available jobs are waiting for processing. The implementation of MIP and CP was done using IBM ILOG CPLEX Optimization Studio.

3.1 Dispatching Rules

Typically a discrete event simulation model needs dispatching rules. In the simplest cases these are rules like First In - First Out or the reverse Last In – First Out. In this investigation the dispatching rules ODD – an operational due date based rule, a quality based rule and a combination of these both rules with a minimized setup rule are used. All rules are described more in detail in the following subsections.

3.1.1 Operational Due Date - ODD

The ODD rule is first published in Rose (2003). The main idea of this rule is, that each job gets an own operational due date for each operation within the manufacturing process. In this investigation the ODD rule is implemented as follows:

1. Calculate the gap between (d_j - r_j) and the total raw processing time for all steps including delay
2. Divide the calculated gap by the number of cycles the job j has in its product route → cyclic gap
3. Adds the calculated cyclic gap as an allowed waiting time in front of the lithography step

This rule leads to an optimization of tardiness without taking quality or setup minimization into account. This rule can be enhanced by a setup minimization rule as described in section 3.1.3.

3.1.2 Quality Based Rule - Quality

For this dispatching rule an additional parameter "maximum offset" $O_{\max}$ is needed. This parameter describes the maximum allowed difference between the average job offset O_j and the machine offset O_m. The rule works in the following way:

1. For the first cycle all machines are allowed for a job $j \in J$ → After this, job j has the offset O_m of the allocated machine
2. For all other cycles: if job j wants to be processed on machine m:
    a. If $|O_j - O_m| > O_{\max}$ → job j is declined to be processed on this machine, otherwise:
    b. Calculate the difference between O_j and the current offset for all other available machines
    c. If another machine as m has a better difference → job j is declined on machine m
    d. If no other machine has a better offset difference → job j is allowed on machine m

Again this dispatching rule ignores the setup state of the machines and therefore it may happen, that the setup duration is relatively high. Also the operational due date is used for job ordering.

3.1.3 Setup Minimization Rule - MinSetup

This dispatching rule is an extension for the ODD rule and the Quality rule. With this extension it is ensured, that a job j is only allowed on a machine m if

- The setup state (reticle) which is currently allocated on the machine m is equal to the requested state of the job j OR
- There are no other available jobs with the same setup state as currently allocated on the machine m AND
- There is no other free machine with the same setup state as requested by the job j.

This implementation of the MinSetup rule may be in conflict with the quality based dispatching rule. Here a machine which is preferred by the quality rule can be rejected by the MinSetup rule and vice versa. To resolve this conflict in the last parts of these rules the search for other machines and jobs is restricted to those objects which are in the maximum offset gap $O_{\max}$ and according to their setups. This leads to a more complex extension of the MinSetup rule compared to the extension of the ODD rule.

3.2 Mixed Integer Programming - MIP

Another integrated method is the mixed integer programming method. In this investigation a capacitive based approach is coupled with the described simulation model. Thereby the MIP model is executed cyclic within the simulation model. The simulation model generates the input data for the MIP model. This input data consists of:

- Set of available machines M
    - Current setup state s_m ($m \in M$)
    - Remaining run time r_m ($m \in M$)
    - Current offset o_m ($m \in M$)
- Set of available jobs J
    - Needed setup state t_j ($j \in J$)
    - Current average offset q_j ($j \in J$)
    - Process time p_j

- Set of secondary resources R

Now the goal of the MIP model is to allocate all jobs to machines with respect to different objectives. Here three objectives are defined: makespan $C_{\max}$, quality / offset O and setup minimization S. The mixed integer model is built up in a multistage way: in the first stage the makespan is minimized and the result is used in the second stage with a factor ω_1. The second stage tries to minimize the quality by assigning jobs to the best machines and this result including the result from the first stage is used again in a third stage were the setup should be minimized. For this, the MIP model needs two relaxing factors ω_1 and ω_2 for the results of stage 1 and 2. This implementation is shown in Figure 3.

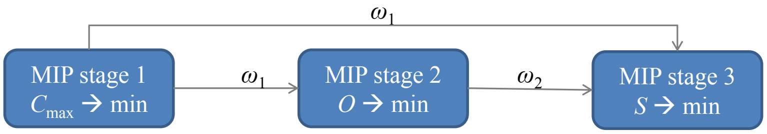

Figure 3: MIP model divided into three stages

For the MIP model the following decision variables are needed:

$X \in \{0,1\}^{J\times M}$ - Defines if a machine $m \in M$ is used for job $j \in J$

$Y \in \{0,1\}^{R\times M}$ - Defines if a secondary resource $r \in R$ is used on a machine $m \in M$

$C_{\max} \in \mathbb{N}$ - Defines the maximum workload over all machines

With the defined input data and variables the following equations can be formulated:

$$\sum_{m\in M} X_{j,m} = 1 \qquad ;\forall j \in J \tag{9}$$

$$r_m + \sum_{j\in J} X_{j,m} \cdot p_j \le C_{\max} \qquad ;\forall m \in M \tag{10}$$

$$\sum_{\substack{j\in J\\ t_j = r}} X_{j,m} \ge Y_{r,m} \qquad ;\forall m \in M, r \in R \tag{11}$$

$$\sum_{\substack{j\in J\\ t_j = r}} X_{j,m} \le K \cdot Y_{r,m} \qquad ;\forall m \in M, r \in R \tag{12}$$

$$O = \sum_{\substack{j\in J\\ m\in M}} | o_m - q_j | \cdot X_{j,m} \tag{13}$$

$$S = \sum_{\substack{r\in R\\ m\in M}} Y_{r,m} \tag{14}$$

Equation (9) ensures that each job is assigned to exactly one machine. Equation (10) limits the maximum makespan over all machines. With equations (11) and (12) X and Y are coupled. This means a secondary resource is used on a machine if a corresponding job is allocated to a machine and vice versa. Here, K is a big integer number. Equations (13) and (14) are the definitions for the objectives for each stage (c.f. Figure 3). Additionally in stage 2 the following equation is needed:

$$C_{\max} \le \omega_1 \cdot C_1 \tag{15}$$

Here C_1 is the result from stage 1. In the same way in the third stage equation (15) and the following equation are needed, where O_2 is the result of the second stage:

$$O \le \omega_2 \cdot O_2 \tag{16}$$

The result from this MIP method is an allocation of all jobs to machines without consideration of time. Due to this fact in the simulation model the ODD rule is further used to assign the jobs in sequence.

3.3 Constraint Programming – CP

The last investigated method is a constraint programming approach. This method generates a detailed schedule from the information gained by the simulation model. The input data from the MIP model is extended by the following information:

- Release date r_j and due date d_j for each $j \in J$
- Limit l_r for secondary resource $r \in R$
- Setup matrix U

Due to the fact that a constraint programing model does not have mathematical equations from which it is built up, in this case the CP model is described with typical CP expressions. First for each job j a task called $jobtask_j$ has to be defined. This task is assigned to a possible domain where the start of the domain is equal to the release date r_j and the task has a length which is defined by the process time p_j. Additional for each job-machine combination an optional worker task $worktask_{j,m}$ has to be defined which is used to allocate the job tasks to machines. Furthermore, a cumulative function $LimitSR_r$ has to be defined which ensures the limit l_r for each secondary resource $r \in R$. The last needed expression variable is a sequence variable called $MachineSetup_m$ which is used to integrate the setup matrix and therefore the setup times. With these basics the CP model can be described by:

$$\text{alternative}(jobtask_j, \text{all}(m \in M)\ worktasks_{j,m}) \qquad ;\forall j \in J \qquad (17)$$

$$\text{noOverlap}(MachineSetup_m, U) \qquad ;\forall m \in M \qquad (18)$$

$$LimitSR_r \leq l_r \qquad ;\forall r \in R \qquad (19)$$

With the constraint (17) each job is assigned to exactly one machine and the tasks are coupled regarding start and end date. Constraint (18) is used to integrate the setup matrix U into the model and additionally due to the fact that the sequence variable for the machine setup is coupled with the worker tasks this constraint also ensures that the assigned jobs on a machine cannot overlap. The last constraint (19) ensures that the number of used secondary resources in parallel cannot exceed the limit l_r. The objective functions can be defined as:

$$\text{Cycle time } C = \sum_{j \in J} \text{endOf}(jobtask_j) \qquad (20)$$

$$\text{Offset } O = \sum_{\substack{j \in J \\ m \in M}} \text{presenceOf}(worktask_{j,m}) \cdot | q_j - o_m | \qquad (21)$$

$$\text{Tardiness } T = \sum_{j \in J} \max(0, \text{endOf}(jobtask_j - d_j)) \qquad (22)$$

The minimization goal for this CP approach can be described as a weighted sum of all these objectives whereby only one weight ω_1 is used for the offset. This is due to the complexity and therefore the runtime of this investigation:

$$\text{minimize } \omega_1 \cdot O + C + T \qquad (23)$$

The result from this CP approach is an exact allocation of jobs to machines including their starting date. This result is used exactly in the simulation system. So here no ODD rule is needed.

4 TEST SETS

To test the described methods a test environment is built up. The underlying scheme of the scheduling problem is already described in Figure 2. The investigated model consists of five products. The products thereby differ in the number of cycles as well as in their process times and the needed secondary resource (reticle). The number of cycles thereby variates between 5 and 12 cycles. In summary 15 different reticles exists. The number of available reticles per type is between 1 and 4. These product and secondary resource data is generated once and then used for each test instance. The rest of the used parameters are defined in the following table:

Table 1: Experimental setup (UD - uniform distribution).

Factor	Values used	Total values
Number of machines *m*	10	1
Number of jobs	2500; 2600	2
Considered time horizon	80d	1
Machine offset	static, dynamic	2
Release dates *rdd*	UD ~ [0,80d]	1
	Number of independent instances	100
	Total number of problems	400

For calculating the results a settling phase of 10 days is implemented. This means only jobs which are released and finished between day 10 and day 80 are considered for the objectives. Furthermore the machine offset can be static or dynamic. In the static case the offset is the same for the whole 80 days and in the dynamic case this offset can vary. For this for each machine a time course of the offset is generated. Here the offset can vary each hour within an interval of [-0.2;0.2]. If the offset drifts out of the bounds of [-1;1], a new offset for this machine is calculated which is within these bounds. With the different number of jobs within these test instances a different utilization of the machines should be reached to test the methods under different circumstances.

These test instances are used to build up scheduling instances for the problem described in section 2. For each test instance schedules are generated with each described method (Dispatching, MIP and CP) within the underlying simulation environment. After this for each schedule the objectives are calculated and evaluated. The results can be found in the next section.

5 RESULTS

In this section a few results of this investigation are shown. The following rules are used:

- ODD – Operational due date rule like described in section 3.1.1
- ODD+SetupMin – ODD rule combined with the setup minimization of section 3.1.3
- Quality($O_{\max}$) – Quality rule as described in section 3.1.2 with $O_{\max}$ as additional parameter
- Quality+SetupMin($O_{\max}$) – Quality rule combined with setup minimization
- MIP(ω_1, ω_2) – Mixed integer programming method as described in section 3.2
- CP(ω_1) – Constraint programming approach described in section 3.3

The rules are orientated on the *x*-axis and for each objective the results are shown as a separate bar. The used objectives are defined in section 2.1 and 2.2. Figure 4 shows the result for a test set of 2500 jobs and a static machine offset. This means the offsets of the machines keep equal for the whole time horizon of 80 days. As already mentioned all results are gained from a “stable” phase between day 10 and day 80.

The results show that regarding tardiness and flow factor the ODD rule and especially the ODD+SetupMin rule are best. Having an eye on the quality parameter the other methods perform better. The MIP method as well as the CP method gains best results regarding the setup time. This may also due to the fact that these methods have a forecast. Also the results show that the MIP method is very sensitive to the both factors ω_1 and ω_2.

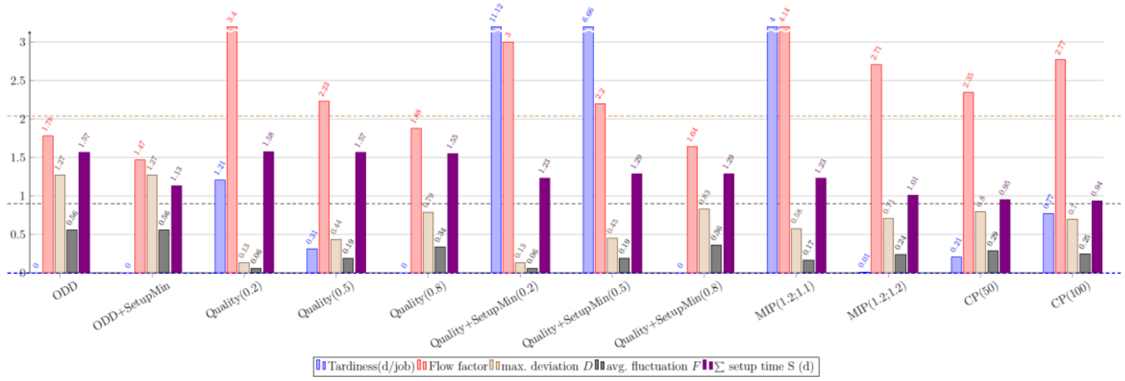

Figure 4: Results for static machine offset and 2500 jobs within 80 days

Figure 5 shows the result for the test instances with 2600 jobs and again a static machine offset.

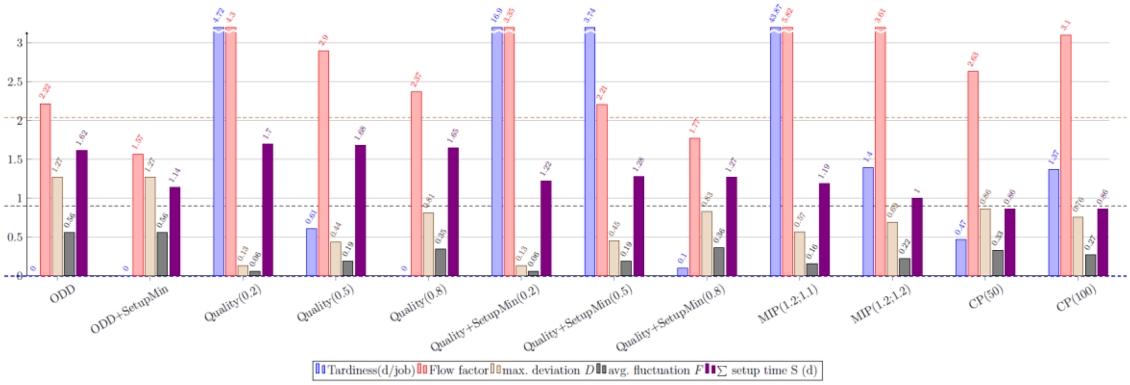

Figure 5: Results for static machine offset and 2600 jobs within 80 days

The results are similar to the results shown in Figure 4. Only due to the higher load the effect in tardiness and flow factor is higher. Again the quality based dispatching rules gain best results for the quality objectives, depending on their maximum offset. The disadvantage of these dispatching rules are the high flow factor and setup time.

The last Figure shows the result for 2600 jobs and a dynamic machine offset. This means in this case the machine offset can vary over time.

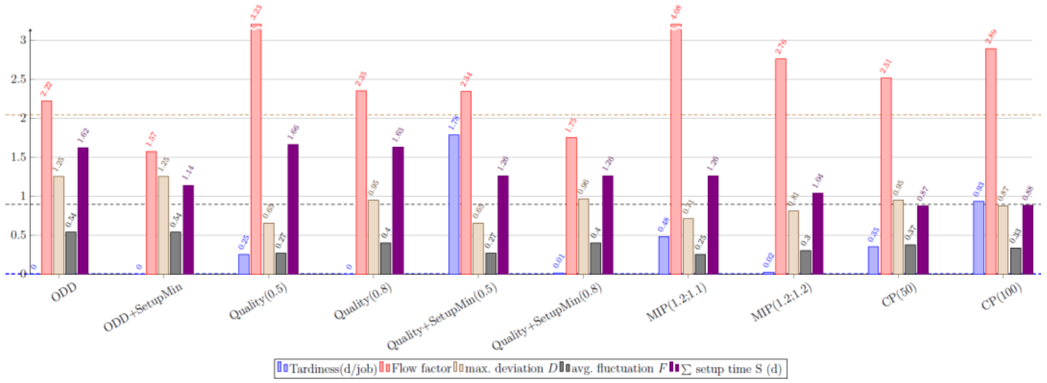

Figure 6: Results for dynamic machine offset and 2600 jobs within 80 days

For these results in contrast to Figure 4 and Figure 5 the tardiness and flow factor especially for the MIP methods are much lower. This may be due to a problem occurred during the MIP optimization, where the last (or second) stage of the MIP sometimes could not found a solution and then the result from the second (or first) stage is used.

Overall all results show a similar trend of the methods. Taking all objectives into account the CP(50) method seems to be a good compromise for the investigated methods.

6 CONCLUSION AND OUTLOOK

In this investigation different quality based optimization methods are compared. These methods are tested on an artificially generated benchmark which is orientated on a lithography step within a frontend of a semiconductor manufacturing. This benchmark is based on cycles which typically occur within a frontend fabrication. The different used products have between 5 and 12 cycles for the lithography step. The investigated quality parameter is based on repeatability on the lithography step. Therefore the used lithography tools get an offset which is normalized within the interval [-1;1]. This offset can be static or dynamic over time. Also the offset of a machine is equal for all products in this investigation. The overall objective is to minimize this offset over all jobs. Further objectives are the tardiness, flow factor and setup minimization. Setups occur if a production step on a machine needs another secondary resource (reticle) than the last used reticle on this machine. Overall in this investigation 15 reticles with an amount between 1 and 4 exist. The needed reticle depends on the process step of each product. Two mainly different dispatching rules are implemented. The first is the tardiness orientated dispatching rule ODD and the second is a quality based dispatching rule. Both rules are also coupled with a setup minimization. Furthermore a multi stage mixed integer programming model as well as a constraint programming method are implemented. The results show that the MIP model is very sensitive to the used parameters and there also seems to be an error which results in an unsolvable third (and partly second) stage. The results for the CP optimization shows good performance over all objectives. The ODD rule results in an optimized schedule regarding flow factor and tardiness whereby the quality based rule reaches best results for the quality parameters. Overall the CP method seems to be the best method taking all objectives into account.

Due to the fact that quality parameter are only rarely investigated in literature so far in this field of optimization a high potential further exists. This cycle based quality optimization should be enhanced for further implementations of dispatching rules and more parameter combinations for the constraint programming approach. Also the problem within the MIP model should be solved.

ACKNOWLEDGEMENT

This project has received funding from the Electronic Component Systems for European Leadership Joint Undertaking under grant agreement No 737459. This Joint Undertaking receives support from the European Union's Horizon 2020 research and innovation program and Germany, Austria, France, Czech Republic, Netherlands, Belgium, Spain, Greece, Sweden, Italy, Ireland, Poland, Hungary, Portugal, Denmark, Finland, Luxembourg, Norway, Turkey.

REFERENCES

Chen, A. and G. Wu. 2007. "Real-time health prognosis and dynamic preventive maintenance policy for equipment under aging Markovian deterioration". In: *Internal Journal of Production Research*, 45(15). 3351 - 3379.

Colledani, M., A. Horvarth and A. Angius. 2015. "Production quality performance in manufacturing systems processing deteriorating products". In: *CIRP Annals - Manufacturing Technology*, 64. 431 - 434.

Cunningham, S. P. and J. G. Shanthikumar. 1996. "Empirical results on the relationship between die yield and cycle time in semiconductor wafer fabrication". In: *IEEE Transactions on Semiconductor Manufacturing*.

Doleschal, D., G. Weigert and A. Klemmt. 2015. "Yield integrated scheduling using machine condition parameter". In: *Proceedings of the 2015 Winter Simulation Conference*, edited by L. Yilmaz, W. K. V. Chan, I. Moon, T. M. K. Roeder, C. Macal, and M. D. Rossetti. 2953-2963. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

Klemmt, A. and L. Mönch. 2012. "Scheduling jobs with time constraints between consecutive process steps in semiconductor manufacturing". In: *Proceedings of the 2012 Winter Simulation Conference*, edited by C. Laroque, J. Himmelspach, R. Pasupathy, O. Rose, and A.M. Uhrmacher. 2173-2182. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

Li, T. and Y. Li. 2007. "Constructive Backtracking Heuristic for Hybrid Flowshop Scheduling with Limited Waiting Times". In: 2007 International Conference on Wireless Communications, Networking and Mobile Computing. 6671 - 6674.

May, G. S. and C. J. Spanos. 2006. *Fundamentals of semiconductor manufacturing and process control*. New Jersey: John Wiley & Sons, Inc.

Mönch, L., J. W. Fowler and S. J. Manson. 2013. *Production Planning and Control for Semiconductor Wafer Fabrication Facilities*. New York: Springer Science + Business Media.

Obeid, A., S. Dauzère-Pérès and C. Yugma. 2012. "Scheduling on Parallel Machines with Time Contraints and Equipment Health Factors". In: *8th IEEE Conference on Automation Science and Engineering*. 401-406. Seoul, Korea.

Qiao, Y., N. Wu and M. C. Zhou. 2014. "Scheduling of Dual-Arm Cluster Tools With Wafer Revisiting and Residency Time Constraints". In: *IEEE Transaction on industrial informatics*, 10(1). 286 - 300.

Rose, O. 2003. "Accelerating Products under Due-Date Oriented Dispatching Rules in Semiconductor Manufacturing." In Proceedings of the 2003 Winter Simulation Conference, edited by S. Chick, P. J. Sánchez, D. Ferrin, and D. J. Morrice, 1346-1350. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

Yugma, C., J. Blue, S. Dauzère-Pérès and A. Obeid. 2015. "Integration of scheduling and advanced process control in semiconductor manufacturing: review and outlook". In *Journal of Scheduling*, 18(2). 195-205.

AUTHOR BIOGRAPHIES

DIRK DOLESCHAL studied mathematics at Dresden University of Technology, Germany. He obtained his degree in 2010 in the field of optimization. He has been a Research Assistant at Electronics Packaging Laboratory of the Dresden University of Technology since 2010 and works in the field of production control, simulation & optimization of manufacturing processes. His email is Dirk.Doleschal@tu-dresden.de

ELISA SOPHIE SCHÖTTLER studied industrial engineering at Dresden University of Technology, Germany with focus on operations logistics and electrical engineering. She obtained her degree in 2016 in the field of optimization. Her email is Elisa_Spohie.Schoettler@tu-dresden.de