

## **SIMULATIONS-BASED AND SOLVER-BASED OPTIMIZATION APPROACHES FOR BATCH PROCESSES IN SEMICONDUCTOR MANUFACTURING**

Andreas Klemmt  
Sven Horn  
Gerald Weigert

Electronics Packaging Laboratory  
Technische Universität Dresden  
01062 Dresden, GERMANY

Thomas Hielscher

Qimonda

01099 Dresden, GERMANY

### **ABSTRACT**

Scheduling is one of the key factors for semiconductor fabrication productivity. Objectives like lot cycle time and throughput must be optimized to push the technological development and secure the existence on the rapid growing global market. But especially in the frontend the manufacturing process is dominated by cluster-tools and reentrance flows which makes a production planning and optimization very hard. The workflow here is mostly controlled only by dispatch rules. To get a further improvement in manufacturing planning strategies, there is an increasing request of exact or simulation-based solution methods for specified work centers or bottleneck machine groups. One example of this is the semiconductor oven process. Here, complex batch processes with a lot of restrictions have to be scheduled. A reduction of cycle time in this section by optimized manufacturing strategies has a great influence on all global optimization objectives. Two approaches are investigated in this paper.

### **1 INTRODUCTION AND LITERATURE SURVEY**

The semiconductor manufacturing process is usually divided into two parts: the frontend and the backend. In the frontend the wafers are fabricated, in backend the wafers are sawed and the resulting chips are assembled to components or modules and finally tested (Potoradi et al. 2002). The frontend wafer fabrication process is thereby divided into four main steps:

- Lithography
- Etching
- Furnace
- Polishing

To bring all different structures (transistors, capacities, etc.) onto the wafer, these four steps have to be carried out repeatedly, so a frontend process planning proves to be notably difficult.

A lot of scheduling techniques for semiconductor manufacturing have been stated in publications over the last years. A good overview of them is given in (Gupta and Sivakumar 2006). Here the authors classify the different scheduling techniques into four groups: dispatching, mathematical programming, search methods and artificial intelligence techniques. In this paper approaches concerning two of this groups are investigated in detail: mathematical programming and search methods.

A special scheduling task in semiconductor manufacturing is the planning of the oven process step after etching operation. Here batch processes are carried out with a lot of restrictions which have great influence on the total lot cycle time. So approaches to scheduling batch processes are a matter of particular interest and also recorded in several publications. Hereby the solution methodologies could be grouped into exact approaches, approximate approaches and simulation approaches. A very detailed collection about them is given in the literature review of (Mathirajan and Sivakumar 2003).

Batch problems are often considered being two sub-problems: batch formation and the sequencing of the formed batches. Different exact and heuristic solution approaches to this are described in (Erramilli and Mason 2006) or (Mönch et al. 2007) for example.

Concerning the oven batch scheduling problem described in this paper, the following publications are significant: (Akcali et al. 2000) compared different scheduling strategies of a semiconductor furnace process by simulation experiments. The technological background of the oven furnace process is presented very clearly, too. In (Tajan, Sivakumar, and Gershwin 2006) different batch strategies concerning to diffusion ovens under comprehension of the predecessor operation are studied. (Mönch 2005) developed different heuristics as well as simulation-based approaches to the diffusion oven problem. (Werner 2007) presented an exact and a heuristic graph-based algorithm for makespan optimization of semiconductor oven batch processes.

## 2 PROBLEM DESCRIPTION

Following, an optimization approach for operative scheduling of a special oven process machine group is described. The machine group consists of  $n_M$  oven machines with partly different properties. The task is to schedule a pool of lots which have already been waiting for processing or are reaching the oven process step in the next hours (currently on etching equipment). The number of all lots to be scheduled is  $n_J$ . Every lot consists of  $w_i$  wafers and has a so called recipe  $r_i$  which defines the technological oven process program. Thereby  $n_R$  defines the number of different currently existing recipes. Every lot has a supply date  $e_i$  which defines the time period till it arrives at the oven process step ( $e_i < 0$  if the lot is already waiting for oven operation). Also some lots can have a time coupling restriction  $t_{is}$  which defines the maximum allowed queuing time before starting the oven process. A lot can be processed separately in an oven machine or merged together with other lots to a batch. A batch may include only lots of the same recipe. Also a maximum batch size  $W_k$  is given for every machine  $k=1, \dots, n_M$  determining the maximum number of wafers which can be merged together. The time  $p_i$  for processing a lot  $i$  in a batch depends only on the recipe. So  $p_i$  is equal for all lots  $i$  with the same recipe  $r_i$ . An oven machine can initially be down or busy. Then  $d_k$  defines the time period till machine  $k=1, \dots, n_M$  is ready for processing the next batch ( $d_k=0$  if machine  $k$  is already ready for processing). In addition not all recipes can be processed by every machine. Therefore  $k$  sets  $MR_k$  are defined describing which recipes  $r \in \{1, \dots, n_R\}$  can be processed by machine  $k$ . The task now is to schedule all lots under the given constraints by minimizing optimization objectives. Two objectives will be investigated in this paper: the makespan and the lot cycle time.

## 3 SOLVER-BASED OPTIMIZATION

In this section a mathematical mixed integer programming (MIP) approach to the described problem is developed. Several of these approaches have been studied in literature for related (mostly single machine) batch problems, for example (Koh et al. 2005). But for the oven batch problem with the described specific constraints no matching formulation was found.

For other batch problems it was shown that exact approaches like MIP can only handle comparative small problems because the complexity of the MIP is strongly increasing with the problem dimension. So it was the first task to find out up to which problem dimension the problem could be solved in near real time. This marks the practical relevance of the MIP-solver.

For all MIP-formulations shown in this paper some further sets and parameters have to be defined. First the sets  $M_i$  specifying which machines  $k \in \{1, \dots, n_M\}$  can be

used for processing lot  $i=1, \dots, n_J$ . Second the sets  $J_k$  including all lots  $i \in \{1, \dots, n_J\}$  which can be carried out on  $k=1, \dots, n_M$ . Both could be derived directly from  $MR_k$ . The maximum number of batches which have to be scheduled on each machine is restricted to  $n_B$ .

All time depending parameters or variables (like  $t_i$ ,  $e_i$ , etc.) are hour-based. The start of planning interval lies at time 0, the end is defined as the finishing time of the last lot. At last a very large number  $K$  has to be defined.

### 3.1 MIP for makespan optimization

First a basis model for makespan optimization of the described problem is developed. For this purpose the following variables have to be defined:

- $s_{jk}$  ...start time of batch  $j$  on machine  $k$ ;  $s_{jk} \in \mathbf{R}_+$   
( $j=1, \dots, n_B$ ;  $k=1, \dots, n_M$ )
- $x_{ijk}$  ...=1, if lot  $i$  is scheduled in batch  $j$  on machine  $k$ ;  
=0, otherwise;  $x_{ijk} \in \{0, 1\}$   
( $i=1, \dots, n_J$ ;  $j=1, \dots, n_B$ ;  $k \in M_i$ )
- $y_{jkr}$  ...=1, if recipe  $r$  is scheduled in batch  $j$  on machine  $k$ ;  
=0, otherwise;  $y_{jkr} \in \{0, 1\}$   
( $j=1, \dots, n_B$ ;  $k=1, \dots, n_M$ ;  $r \in MR_k$ )
- $C_{max}$ ...makespan;  $C_{max} \in \mathbf{R}_+$

Now the MIP basis model can be formulated:

$$C_{max} \rightarrow \min \text{ subject to} \quad (1)$$

$$\sum_{k \in M_i} \sum_{j=1}^{n_B} x_{ijk} = 1 \quad i=1, \dots, n_J \quad (2)$$

$$\sum_{i \in J_k} w_i x_{ijk} \leq W_k \quad k=1, \dots, n_M; j=1, \dots, n_B \quad (3)$$

$$y_{jkr_i} - x_{ijk} \geq 0 \quad i=1, \dots, n_J; j=1, \dots, n_B; k \in M_i \quad (4)$$

$$\sum_{r \in MR_k} y_{jkr} = 1 \quad j=1, \dots, n_B; k=1, \dots, n_M \quad (5)$$

$$x_{ijk} e_i \leq s_{jk} \quad i=1, \dots, n_J; j=1, \dots, n_B; k \in M_i \quad (6)$$

$$d_k \leq s_{1,k} \quad k=1, \dots, n_M \quad (7)$$

$$s_{jk} + p_i x_{ijk} \leq s_{j+1,k} \quad i=1, \dots, n_J; j=1, \dots, n_B-1; k \in M_i \quad (8)$$

$$s_{jk} - e_i - t_i + K x_{ijk} \leq K \quad i=1, \dots, n_J; j=1, \dots, n_B; k \in M_i \quad (9)$$

$$s_{n_B,k} + p_i x_{i,n_B,k} \leq C_{max} \quad i=1, \dots, n_J; k \in M_i \quad (10)$$

Here the constraints set (10) restricts the objective function (1). (2) ensures that every lot can only be processed in one batch on one machine. The maximum batch size is restricted by (3). Equations (4) and (5) force that only lots of the same recipe can form a batch. (6) guarantees that a lot can only be processed after it has reached the oven operation. Equation (7) ensures that batches can not be started on a machine until the initial down/busy time has ended. (8) forces that a batch can be started only after the last one is finished. Equation (9) accomplishes the time coupling restriction.

Different modifications of this MIP-formulation are now possible. For example, it is not necessary to define the variable  $y_{jkr}$  by using

$$x_{ijk} + x_{hjk} \leq 1 \quad i, h = 1, \dots, n_J; r_i \neq r_h; j = 1, \dots, n_B; k \in M_i \cap M_h \quad (11)$$

instead of (4) and (5). However, tests have shown that for the investigated oven problem this formulation is not as powerful as formulation (1)-(10). The number of binary unknowns is indeed lower than in (1)-(10) but the high number of additional constraints negate this advantage.

### 3.2 MIP for cycle time optimization

In this subsection a modification of problem (1)-(10) is developed which allows an optimization concerning the objective lot cycle time. The reason is that the objective makespan is not very significant under real production planning conditions (rolling forecast, see section 6) because not observed new lots enter the system in the planning horizon. To modify the shown optimization problem concerning cycle time minimization variable  $C_{max}$ , equation (1) and (10) can be substituted. Instead of them the new objective function

$$\sum_{k=1}^{n_M} \sum_{j=1}^{n_B} \sum_{i \in J_k} x_{ijk} s_{jk} \rightarrow \min \quad (12)$$

has to be optimized under the constraints (2)-(9). This new objective function is equivalent to the minimization of

$$\sum_{k=1}^{n_M} \sum_{j=1}^{n_B} \sum_{i \in J_k} x_{ijk} (s_{jk} - e_i) \rightarrow \min$$

which describes the minimization of the queuing time of all lots. In the case of equal (recipe depending) process times on all machines, the lot queuing time minimization is equivalent to lot cycle time minimization.

The disadvantage of the objective function (12) is that it is quadratic and so it defines a MIQP (mixed integer quadratic program). Because of the non-convexity of function (12) not every MIP solver can handle such a problem. In some cases it can be sensible to work with the objective function

$$\sum_{k=1}^{n_M} \sum_{j=1}^{n_B} s_{jk} \rightarrow \min \quad (13)$$

instead (results see section 5) which minimizes the sum of the batch start times. To give an exact MIP-formulation for cycle time minimization, it is necessary to define  $n_J$  new variables:

$z_i$  ...starting time of lot  $i$ ;  $z_i \in \mathbf{R}_+$  ( $i=1, \dots, n_J$ )

Then the MIP formulation for cycle time minimization can be written as:

$$\sum_{i=1}^{n_J} z_i \rightarrow \min \quad \text{subject to} \quad (14)$$

$$K(1 - x_{ijk}) + z_i \geq s_{jk} \quad i = 1, \dots, n_J; j = 1, \dots, n_B; k \in M_i \quad (15)$$

and constraints (2)-(9).

In section 5 the following three MIPs will be investigated on some example models and tested for practical relevance.

MIP-1: makespan optimization (1)-(10)

MIP-2: batch start optimization (13) and (2)-(9)

MIP-3: cycle time optimization (14), (15) and (2)-(9)

## 4 SIMULATION-BASED OPTIMIZATION

If the complexity of practical problems in production control is too high to be solved by exact methods, heuristic optimization algorithms in connection with simulation systems become suitable alternatives. Figure 1 shows the basic principle of a simulation-aided optimization system. See (Klemmt et al. 2007) for more details.

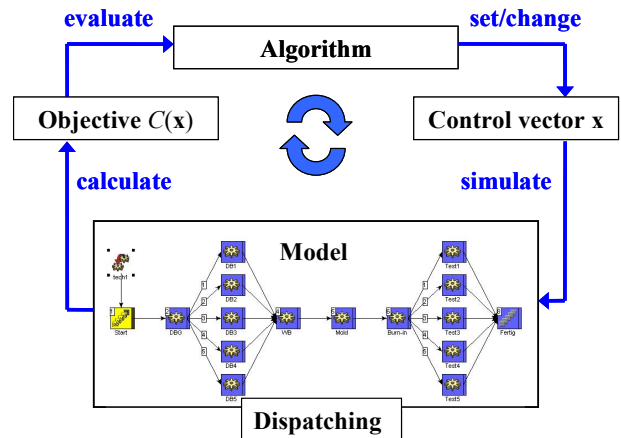


Figure 1: Simulation-based optimization

The problem is described by a simulation model which includes a set of control variables  $x$  and responds with an objective value  $C$  after a simulation run is completed. The control vector  $x$  consists of several variables  $x$  which influence the behavior of the simulation model, e.g. job permutations, buffer and machine capacities or release dates. So the optimization system is divided into an evaluation part on the one hand and a separate optimization algorithm on the other hand. Both subsystems communicate by  $x$  and  $C$  while the optimization cycle is running. A heuristic search algorithm evaluates the objective function after a simulation run is completed and modifies the control variables. Then the model is simulated again (under slightly changed conditions). So the simulation-based optimization approach

can be understood as a heuristically evaluated comparison of alternatives. In contrast to this a (real time) dispatching approach, as used in the manufacturing line, normally delivers only one result on the basis of a rule evaluation.

A simulation model with all described conditions for the oven batch process can be easily built. For this the discrete event simulation system *simcron MODELLER* was used. With the aid of this model it is possible to reproduce the current manufacturing dispatching strategy. It is also possible to investigate different dispatching strategies with regard to their effect on the existing objectives. Furthermore dispatching parameters can be applied as control variables  $x$  in the simulation-based approach.

So a comparison of the current manufacturing strategy (real time dispatching rules), simulation-based optimization approaches and the results of the three MIP-formulations is possible. A dispatching strategy for the oven batch process is exemplarily defined as follows:

**SIM-1:** The simulation model is simulated once. Thereby all available lots are prioritized by their waiting time. If an oven machine is available for processing, the lot with the highest priority is started and merged with all other lots of the same recipe (beginning with highest priority until maximum batch size was reached). No batch will be retarded until at least one lot is still available.

Two simulation-based optimization approaches for the oven batch process are realized as follows:

**SIM-2:** The simulation model is simulated repeatedly, until a time limit is reached. For optimization a Genetic Algorithm is used. The control variables in the simulation model for the optimization are:

- lot sequence of the first six lots (concerning lot waiting time)
- dispatch parameter: minimum batch size
- dispatch parameter: max. waiting time for incoming lots

Optimization objective is the makespan.

**SIM-3:** Same conditions as in SIM-2. Optimization objective is the lot cycle time.

As control variables of the simulation-based optimization three points of influence are chosen: First, in each case after a previous batch is finished, the selection of the next batch recipe and the corresponding lots can be changed by the search algorithm. Second, the parameter “minimum batch size” is variable. This parameter defines that a batch starts immediately if the specified batch size is reached. And third, the “maximum waiting time” can be adjusted. This value retards the start of a new batch, if additional lots of the same recipe are arriving in this period.

## 5 RESULTS ON EXAMPLE MODELS

In this section a comparison of the six developed solution approaches (MIP-1, MIP-2, MIP-3, SIM-1, SIM-2 and SIM-3) for optimizing the oven batch problem described in section 2 is shown. For this four representative example problems (OVE-1, OVE-2, OVE-3 and OVE-4) are generated and illustrated in Table 1. All listed values are altered and do not allow conclusions of real manufacturing data. However, the problem dimension like the number of investigated machines and lots as well as the number of products, the number of allowed products per machine and the relationship of the processing times are derived from real production data.

The task was to develop an optimized scheduling strategy for a special oven group consisting of 4 machines with an incoming lot rate of maximum 25 lots in 12 hours. An exemplary model for this real manufacturing problem is shown in Table 1 below and denoted as OVE-1. As shown later (Table 2), this initial problem could be solved exactly. So the problem dimension (number of lots, products and machines) of OVE-1 was now increased in 3 steps (problem OVE-2, OVE-3 and OVE-4) to get an impression how the quality of the solution depends on the problem size and the used optimization method. All lot data of a problem is also used in the next higher problem dimension (see data coloring in Table 1). For all lots  $i$  a time coupling restriction of 24 hours was set.

The results of the different approaches are shown in Table 2. The termination criteria for MIP-solver and Genetic Algorithm was a time limitation of maximum 5 minutes to simulate online conditions. If the exact solution was found earlier (only possible by MIP-approaches) the optimization termination time is shown in parentheses in the column “problem dimension”. The problem dimension (for MIP-approaches) defines the number of constraints  $\times$  the number of unknowns which are needed to describe the respective MIP formulated in section 3. For solving the MIPs the CPLEX 11.0 optimization library of TOMLAB was used. The simulation-based optimization approaches are realized by the discrete event simulation and optimization system *simcron MODELLER*.

To get an impression about the solution quality of the different approaches it was tried to calculate the exact optimum concerning makespan and cycle time by method MIP-1 and MIP-3 respectively. Sometimes the exact solution could be found after several hours. If not, the optimization was terminated after 24 hours. Then the best found value and the gap to the last lower bound is shown in column “optimum”.

For every solution approach and every problem the results concerning makespan, cycle time (queuing time) and the number of needed batches are shown in Table 2.

Table 1: Four examples for an oven batch problem

Problem	$n_J$	$n_M$	$n_R$	$n_B$	Problem data	Oven	1	2	3	4	5	6	7	8
OVE-1:	25	4	7	3	OVE-1,-2,-3,-4	$W_k$	100	100	50	50	100	100	100	50
OVE-2:	30	5	7	3	OVE-2,-3,-4	$d_k$	2	0	0	3	4	1	0	2
OVE-3:	40	6	8	3	OVE-3,-4	$MR_k$	1,2,	1,2,	4,5,	1,6,	1,2,	1,2,	5,8,	9,
OVE-4:	60	8	10	3	OVE-4		3,4	3,4	6,7	7	3	8	9	10

Lot	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
$e_i$	0	0	0	0	0	1	1	2	2	2	3	3	3	4	5	5	5	5	5	6
$r_i$	1	1	2	2	1	5	5	5	5	3	1	3	1	2	2	1	1	1	6	6
$w_i$	25	25	20	25	25	25	25	25	20	20	5	25	5	25	25	20	20	20	25	25

Lot	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40
$e_i$	6	8	8	8	8	8	9	10	10	10	10	10	11	11	11	11	12	12	12	12
$r_i$	6	1	4	7	7	1	1	3	1	3	8	8	8	6	6	1	1	7	1	7
$w_i$	25	25	25	25	20	25	25	25	25	20	25	20	5	25	25	10	20	5	20	15

Lot	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60
$e_i$	-3	-3	-1	2	2	3	6	6	7	7	7	8	9	10	10	10	10	11	12	12
$r_i$	6	1	4	9	9	1	1	3	1	3	4	4	4	6	6	10	10	7	10	7
$w_i$	25	25	25	25	20	25	25	25	25	20	25	20	5	25	25	10	20	5	20	15

Recipe	1	2	3	4	5	6	7	8	9	10
$p_i$	9	7	8	12	5	7	8	10	9	11

Table 2: Results of the batch optimization approaches

Problem	Optimum	Lot Cycle time	Approach	Best solution			Problem Dimension
	Makespan			Makespan	Cycle time (queuing)	Batches	
OVE-1	25 h	11.24 h (optimum proved after 11 h optimization Time)	SIM-1	29 h	12.76 h (5.0 h)	13	557x226 (290 s) 501x225 (38 s) 669x250
			SIM-2	25 h	11.64 h (3.9 h)	11	
			SIM-3	28 h	11.64 h (3.9 h)	11	
			MIP-1	25 h	14.68 h (6.9 h)	10	
			MIP-2	28 h	11.72 h (3.7 h)	10	
			MIP-3	30 h	11.24 h (3.5 h)	10	
OVE-2	22 h (optimum proved after 5 h optimization time)	10.26 h (best solution after 24 h optimization time 19% gap)	SIM-1	28 h	11.46 h (3.6 h)	14	866x337 777x366 (120 s) 1044x366
			SIM-2	25 h	11.53 h (3.6 h)	11	
			SIM-3	26 h	10.60 h (2.7 h)	12	
			MIP-1	22 h	11.53 h (3.6 h)	11	
			MIP-2	23 h	10.93 h (3.0 h)	11	
			MIP-3	26 h	10.26 h (2.4 h)	11	
OVE-3	26 h	10.1 h (best solution after 24 h optimization time 17% gap)	SIM-1	34 h	12.03 h (3.8 h)	18	1288x484 (90s) 1154x483 1556x523
			SIM-2	27 h	11.05 h (2.8 h)	17	
			SIM-3	32 h	11.73 h (3.5 h)	18	
			MIP-1	26 h	12.15 h (3.9 h)	14	
			MIP-2	27 h	12.02 h (3.8 h)	15	
			MIP-3	30 h	11.00 h (2.8 h)	15	
OVE-4	28 h (optimum proved after 0.5 h optimization time)	10.76 h (best solution after 24 h optimization time 22% gap)	SIM-1	33 h	12.36 h (3.8 h)	24	1880x691 1684x690 2272x750
			SIM-2	28 h	11.32 h (2.8 h)	19	
			SIM-3	28 h	11.13 h (2.6 h)	19	
			MIP-1	28 h	14.43 h (5.8 h)	20	
			MIP-2	30 h	12.18 h (3.6 h)	19	
			MIP-3	29 h	11.30 h (2.7 h)	20	

Evidently, the simulation-based as well as the solver-based optimization approaches show clearly better results concerning all objectives than the original dispatching strategy on the investigated example problems. So there is a significant optimization potential which can be exploited by extensive and iterative operating optimization systems in comparison to a single step rule based dispatching system. This is a consequence of the high number of batches which have to be usually assorted by the dispatching system.

It is very interesting that the exact solution approach by MIP delivers almost as good results for problems which are much bigger than the original manufacturing problem. The objective function improvement in the first five minutes is of such a high percentage that either the exact solution is already met or the objective function difference to the optimum is marginal. Here also a large difference between cycle time and makespan optimization can be seen. A beneficial or optimal cycle time solution doesn't have to correlate with an favorable makespan result. As a compromise between exact cycle time (MIP-3) and makespan (MIP-1) optimization, method MIP-2 can be designated.

Here the number of batches and the sum of the batch start times respectively, is minimized. Depending on the underlying manufacturing problem it must be chosen which method should be used. For example for rolling forecast (see next section) a makespan optimization is not expedient.

The simulation-based optimization approaches show continuously good results for each problem and for both objectives. Indeed, the exact optimum concerning cycle time was not reached in any model but the results are very close to them. Especially for higher problem dimensions (OVE-4) the best found value after 5 minutes optimization time was already better than the MIP-3-solution. So if the problem dimension is much bigger than in OVE-4, simulation-based approaches are advisable only. In such cases the problem dimension of the MIP is too high for finding good results under the given time constraints. Even the finding of feasible solutions is then already a problem.

A visualization of two different batch strategies (SIM-1 and MIP-3) for problem OVE-1 is shown in Figure 2.

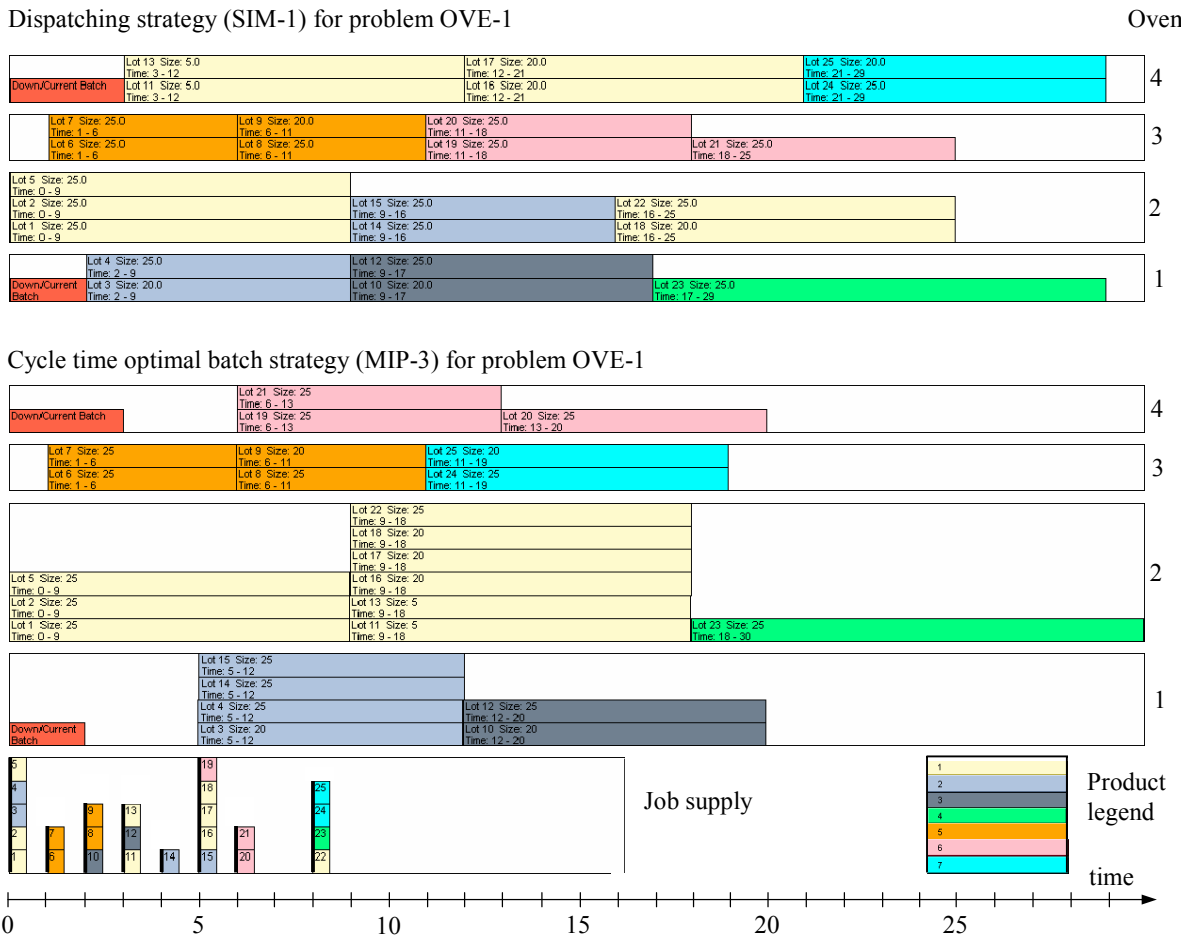


Figure 2: Comparison of two different batch strategies

The difference between both methods is shown comprehensively. In the original dispatching strategy there are no idle time intervals. Every machine is loaded as soon as a batch can be started. In contrast to this the optimal batch strategy (concerning cycle time) is sometimes retarded to form bigger and altogether less batches. So only then the lot cycle time is minimized which is calculated as:

$$\frac{1}{n_j} \sum_{i=1}^{n_j} (z_i + p_i - e_i) \quad (16)$$

However, the problem is that sometimes a retardation and sometimes an immediate batch start leads to the optimal batch strategy. This makes a rule based decision for cycle time optimization very hard or even impossible.

### 6 ROLLING FORECAST

In the previous section the potential of simulation-based and solver-based optimization approaches was shown on some representative static example models. But in the real production scenario continuously new lots are arriving at the oven operation within the planned time period. So oven 1, 3 and 4 will not be idle in 20 hours (see Figure 2 MIP-3) but rather processing these new lots. Unfortunately, because of the complexity of semiconductor manufacturing, nobody can say exactly when these new lots arrive. Only lots on the preceding operation etching can be exactly planned.

A solution for this is to describe a modification of the MIP-3 optimization approach which simulates a practical execution of this method under conditions of rolling forecast. So the real benefit of this approach (compared to the realized strategy in the manufacturing line) can be evaluated. Therefore a real production data set of the oven group consisting of four machines is investigated over a time period of one month. All lot- and machine information  $e_i$ ,  $w_i$ ,  $t_i$ ,  $r_i$ , etc. are known for the whole month.

A solver-based optimization approach (repeated execution) for the real production data can be drafted as follows: At time 0 (start of the month) the problem MIP-3 is solved for all lots which are already in operation or will entering oven operation in the next  $\Delta e$  hours (already known lots from etching operation). After a defined time period of  $\Delta t$  hours rescheduling is started. All lots which have been finished until  $0+\Delta t$  are further disregarded. All batches which are currently (on  $0+\Delta t$ ) in process, define the new machine down times  $d_k$  (see Figure 3). The lots of these batches are also disregarded. All lots which have not been started until  $0+\Delta t$  as well as all lots entering oven operation in the time period  $0+\Delta t+\Delta e$  represent the new lot pool. Start time 0 is now moved and represents  $0+\Delta t$  (denoted as

$0'$  in Figure 3). All supply dates of the still remaining lots are adjusted ( $e_i=e_i-\Delta t$ ). Now the MIP which represents this new problem is solved. The process is repeated until all lots are planned and the end of the planning horizon (one month) is reached.

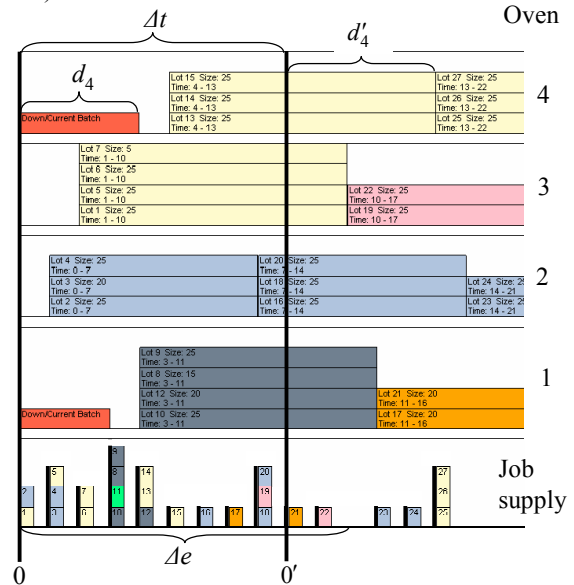


Figure 3: Rolling forecast (altered data)

With the shown optimization approach the solver has only the same information as the line staff at the representing date. So no information is used which is not also available for line staff or the dispatching system. Thereby for rolling forecast only the objective lot cycle time (16) represented by minimizing (14), the queuing time, is investigated. The use of a makespan optimization is as already mentioned not expedient in this application scenario.

On representative example models (see Table 2) a theoretical benefit in queuing time reduction of 30%-40% in comparison to a single step dispatch rule was shown. In the next section it will be investigated if this is also realistic in the real production environment which is also controlled by a rule based dispatching system.

### 7 COMPARISON TO REAL DATA

The investigated data set of reported data of the real process (one month of an oven group) was explicitly chosen in a time period without significant down times on the oven equipment to ensure comparability. When calculating the real cycle time (queuing time), the lots with significant problems (e.g. hold) were not taken into consideration. Only lots without problems, more than 1000, are used for this calculation because the lots in the MIP approach have

also “no problems”. The high number of lots and the long observation period guarantee a good average value for the objective function by different loading scenarios. Also an average transportation time from etching to oven process is considered in the queuing time of all lots planned by the MIP approach. Furthermore time coupling restriction of some products and high priority lots are considered. So the basis for an impartial comparison is made as well as possible.

To prevent any conclusions to real objective values, the cycle time of the manufacturing line is subsequently scaled to 1 (also queuing time). So only the percental improvement of the objective function by the repeated MIP approach is shown. The data set is thereby investigated several times under different parameter settings. First, an interesting investigation can be made by studying  $\Delta t$ . This is a relatively arbitrary parameter ( $0 < \Delta t < \Delta e$ ). So the question whether a high frequent rescheduling makes sense, can be answered. Secondly, the forecast parameter  $\Delta e$  can be changed. Usually it is clearly defined by the manufacturing system itself. But in some cases a purposeful reducing of the forecast parameter is sensible.

The results of the comparison of real data and solver-based optimization approach are shown in Figure 4. On the y-axis the scaled objective function concerning cycle time and queuing time is shown for different settings of the forecast parameter  $\Delta e$ . The rescheduling interval is displayed on x-axis. The average oven processing time is calculated and the x-axis is scaled according to this value.

For all rescheduling optimization runs a time restriction of maximal 5 minutes was defined. So the calculation of each data point in Figure 4 takes several hours which is equivalent to the optimization time which would be needed in the whole month under practical use.

The investigation in parameter  $\Delta t$  shows clearly that it is better to reschedule in relative short time intervals. This is very intuitive, too. Because of the high rescheduling frequency the optimizer has always the “latest” information about the progress of the current lot inventory.

More interesting are the results concerning parameter  $\Delta e$ . It is clear that, if the forecast is relatively small (<average process time), the objective function improvement is not as good as in cases with a wide forecast (>average process time). Especially a retardation of lots to form bigger batches is not possible then. But when the forecast is larger than the average process time ( $\Delta e=1.2$  and  $\Delta e=1.8$ ), the results are relatively close together or even better for forecast horizon  $\Delta e=1.2$ . This can be interpreted in the following way. If the forecast is too wide, a lot of information which is not needed for the next batches on the oven equipment exists. Because of the fact that more lots have to be planned, the dimension of the MIP represented by  $\Delta e=1.8$  is much bigger than the dimension of a problem representing by  $\Delta e=1.2$ . So it can be possible that an exact (or even good) solution can not be found.

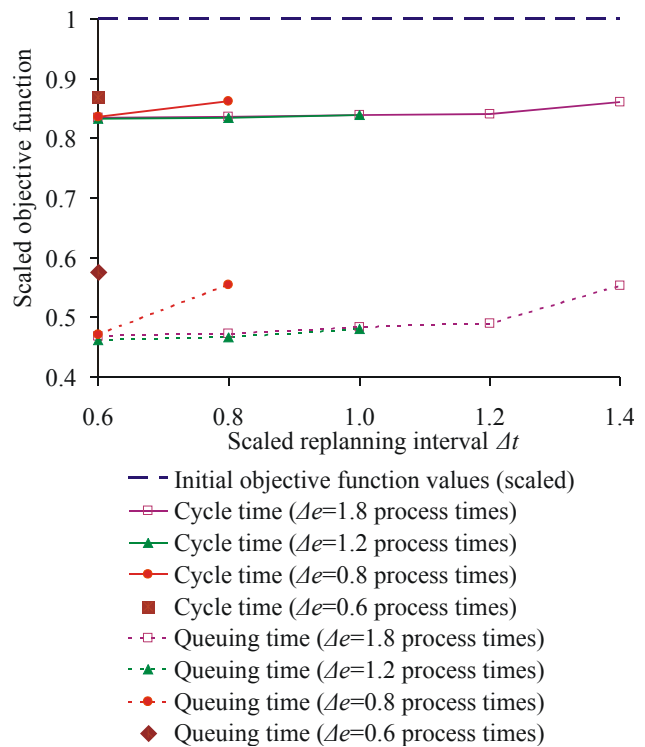


Figure 4: Theoretical cycle time (queuing time) optimization potential of the oven group

Summarizing it can be stated that the shown MIP approach for cycle time reduction (MIP-3) delivers comparable good results for real manufacturing data as for the investigated example models in section 5. So the optimization potential of the solver- or simulation-based approach also exists in the real manufacturing line. Of course, the theoretical benefit of 50% queuing time reduction will not be reached in practice because there are situations which cannot be planned (machine problems, personnel problems, technical problems, etc.). But because of the fact that sometimes a retardation of lots and sometimes an immediate batch start leads to a nearly optimal solution, this potential could not be exploited easily by a rule evaluation as usually realized in a dispatching system.

After all investigations of parameter settings it has to be considered that these are problem specific. For other batch problems, even if they are very related to the one shown in this paper, the dependencies could be different. Next to the number of lots and machines, the number of different products and the equipment dedication of the products have a great influence on the solvability and convergence of the MIP approach.

Also the forecast parameter  $\Delta e$  and the number of batches per machine  $n_B$  should be chosen problem adapted. This can reduce the optimization time and will improve the solution quality drastically.



## 8 CONCLUSION

In this paper different MIP and simulation-based optimization approaches to scheduling batch processes are presented. For the concrete example of an oven batch machine group for the semiconductor furnace process these methods are investigated and compared with a rule based dispatching approach. The background of this comparison was the information that the real manufacturing process is also controlled by a rule based dispatching system. For the representative example models as well as for the real process data it could be shown, that there is a still significant optimization potential to exploit by the developed methods.

## ACKNOWLEDGMENTS

Special thanks goes to Philipp Heimann (Qimonda Dresden) and Sebastian Werner for the excellent preliminary work as well as the data extraction and preparation which were the basis for this paper.

## REFERENCES

- Akcali, E., R. Uzsoy, D. G. Hiscock, A. L. Moser, and T. J. Teyner. 2000. Alternative loading and dispatching policies for furnace operations in semiconductor manufacturing: a comparison by simulation. In *Proceedings of the 2000 Winter Simulation Conference*, 1428-1435.
- Erramilli, V., and S. Mason. 2006. Multiple orders per job compatible batch scheduling. *IEEE Transactions on Electronics Packaging Manufacturing* 29:285-296.
- Gupta, A. K., and A. I. Sivakumar. 2006. Job shop scheduling techniques in semiconductor manufacturing. In *The International Journal of Advanced Manufacturing Technology* 27:1163-1169.
- Klemmt, A., S. Horn, E. Beier, and G. Weigert. 2007. Investigation of modified heuristic algorithms for simulation-based optimization. In *Papers of the 30th International Spring Seminar on Electronics Technology*, 24-29.
- Koh, S.-G., P.-H. Koo, D.-C. K., and W.-S. Hur. 2005. Scheduling a single batch processing machine with arbitrary job sizes and incompatible job families. *International Journal of Production Economics* 98:81-96.
- Mathirajan, M., and A. I. Sivakumar. 2003. Scheduling of batch processors in semiconductor manufacturing – a review. *Innovation in Manufacturing Systems and Technology (IMST)*.
- Mönch, L., R. Schabacker, D. Pabst, and J. W. Fowler. 2007. Genetic algorithm-based subproblem solution procedures for a modified shifting bottleneck heuristic for complex job shops. *European Journal of Operation Research* 177:2100–2118.

Mönch, L. 2005. *Ablaufplanung für Batchmaschinen in der Halbleiterproduktion – Modellierung und Lösungsverfahren*. Available via [http://www.pm.tu-berlin.de/fileadmin/assistenten/martin/mainz\\_scm\\_moench\\_final.pdf](http://www.pm.tu-berlin.de/fileadmin/assistenten/martin/mainz_scm_moench_final.pdf).

Potoradi, J., O. S. Boon, S. J. Mason, J. W. Fowler, and E. M. Pfund. 2002. Using simulation-based scheduling to maximize demand fulfillment in a semiconductor assembly facility. In *Proceedings of the 2002 Winter Simulation Conference*, 1857-1861.

Tajan, J. B. C., A. I. Sivakumar, and S. B. Gershwin. 2006. Performance of a serial-batch processor system with incompatible job families under simple control policies. *Innovation in Manufacturing Systems and Technology (IMST)*.

Werner, S. 2007. *Entwicklung und Bewertung von Methoden zur Optimierung ausgewählter Fertigungsprozesse in der Halbleiterindustrie*. Thesis, Department of Mathematics, Technische Universität Dresden.

## AUTHOR BIOGRAPHIES

**ANDREAS KLEMMT** studied Mathematics at Dresden University of Technology, Germany. He obtained his degree in 2005 in the field of optimization and data classification. He has been a Research Assistant at Electronics Packaging Laboratory of the Dresden University of Technology since 2006 and works on the field of production control, simulation & optimization of manufacturing processes, especially in the field of electronics and semiconductor industry. His e-mail address is [<klemmt@avt.et.tu-dresden.de>](mailto:klemmt@avt.et.tu-dresden.de)

**SVEN HORN** studied Computer Science at the University of Leipzig, Germany. He was also employed as a Software Engineer for developing telecommunications and financial software. Since 2003 he is a Research Assistant at Electronics Packaging Laboratory of the Dresden University of Technology. His main interests are simulation and optimization of manufacturing processes as well as database systems and their practical application. His e-mail address is [<horn@avt.et.tu-dresden.de>](mailto:horn@avt.et.tu-dresden.de).

**GERALD WEIGERT** is an Assistant Professor at Electronics Packaging Laboratory of the Dresden University of Technology. Dr. Weigert works on the field of production control, simulation & optimization of manufacturing processes, especially in electronics and semiconductor industry. His e-mail address is [<gerald.weigert@tu-dresden.de>](mailto:gerald.weigert@tu-dresden.de).

**THOMAS HIELSCHER** is a Staff Engineer for Process and Productivity Mastering at Qimonda Dresden. His e-mail address: [<thomas.hielscher@qimonda.-com>](mailto:thomas.hielscher@qimonda.-com).