Scheduling batches with time constraints in a job shop system: developing two approaches for semiconductor industry

Federica Ciccullo*, Margherita Pero, Giovanni Pirovano, Andrea Sianesi

*Politecnico di Milano – Department of Management, Economics and Industrial Engineering
Via Lambruschini 4B, 20156 Milano, Italy (federica.ciccullo@polimi.it)

Abstract:
This paper stems from an EU-funded project called INTEGRATE. The project aims at developing solutions to improve semiconductors manufactures efficiency. In this contribution, we propose two distinct algorithms to support (ST) managers of Catania’s plant to create the batches and to dispatch them among three consecutive operations (i.e. one cleaning and two diffusions). The case study presents a problem that is similar to a scheduling problem in a flexible job shop system, but with several additional elements of complexity. In particular, two of them are keystones of our approach, namely: time constraints between operations (i.e. the time between the end of an operation $n$ and the start of the operation $n+q$ must be lower than a time-limit, in order to guarantee the lots’ quality) and the absence of batching affinity between operations (Dabbas and Fowler, 2003). Researchers have proposed various approaches dealing with scheduling with time constraints (e.g. Chen and Tang, 2012) and with batching problems (e.g. Cerekci and Banerjee, 2010) separately. Nevertheless, the analysed literature is falling short in merging the two problems with the above-mentioned elements of complexity, to support the decisions of which lots to process and in which order. In addition to that, all the detected models are not suitable to ST plant situation. To fill this gap we propose two heuristic models that, in line with the literature, are based on different prioritisation indexes. The two models differ in their focus, which is on the sole time constraint for the first one and on flow time as well for the second one. Some preliminary and qualitative considerations are provided.

Keywords: Time constraint, scheduling, batch machines, wafer fab

1. Introduction
Electronics industry is one of the largest industries in the world. Key players in this industry are the semiconductor manufacturers, which produce integrated circuits on silicon wafers. According to a report by the Semiconductor Industry Association (SIA), the worldwide sales of semiconductors reached $25.53$ billion for the month of July 2013, the highest total of 2013, and an increase of 5.1% over July 2012. Wafer fabrication is a very complex manufacturing process. It takes place in fabs, which are very large systems with tens to hundreds machines moving hundreds to thousands wafers (Moench et al., 2011). The scheduling problems in a wafer fab is similar to scheduling flexible job shop but it is more complicated, since there are several elements which have to be considered, i.e. (Dabbas and Fowler 2003): (1) Re-entrance of lot in some stages, i.e. lots must be processed by the same machine during multiple visits; (2) Different types of machines depending on the object processed, i.e. wafer, lot (a set of wafers) or batch (a set of lots); (3) Setup times; (4) Auxiliary resources: in some stages, auxiliary resources are required, such as reticles in the photolithography stage; (5) Multiple orders per lot: due to technology innovation and increase in the wafer size, customer order is reducing, so various customer orders might be grouped in a lot; (6) Uncertainties and machine availability: the arrival of new orders, the cancellation of orders, and long machine failure are common unplanned events; (7) Time constraints: in some cases, the time between the end of an operation $n$ and the start of the operation $n+q$ must be lower than a time-limit, in order to guarantee the lots’ quality.

Scheduling in such a context is challenging for managers. Therefore, this work presents the preliminary results of a research carried on in the framework of an EU-funded project, INTEGRATE, aimed at proposing solution to support wafers manufactures in managing the shop floor. The studied case is the one of the plant of STMicroelectronics in Catania. In particular, the proposed approach attempts to solve the practical issue experimented by the production management of the company, namely: defining the batches and dispatching the lots among three consecutive operations (i.e. one cleaning and two diffusions) characterized by time constraints between them and no batches affinity between the first and the second operation. Moreover, the batches might be composed of lots of wafers of different families. By following the classification of Bucker (2007) and considered $j$ as the lot’s index, the problem can be described as in formula (1):

$$ R \left[ \min_p \left\{ p_{j}, d_{j}, p - batch, incompatible - batch, T_{lags} \right\} \left\{ \min(T), \min(R_{work}), S_{\text{scrap}} = 0 \right\} \right] \left( 1 \right) $$
Where the first part of (1) defines the machine environment, which is unrelated parallel machines (R) with different operations (J). The second part describes the product characteristics in the studied situation: different processing time of the lots \( p_j \), relevant and different due dates of the lots \( d_j \), the time for processing a batch on a machine is the longest processing time among all the lots in the batch (p-batch), not all the lots can be batched together (incompatible-batch), and presence of time constraints between operations (Tlagmin). The last part represents the objectives, which are: minimizing the average flow time of the lots (T) and the number of re-cleaned lots (Rwork), and avoid scrapped lots (Sscrap). To this aim, two algorithms have been developed. The former leverages on both the Apparent Tardiness Cost (ATC)-Index (Vepsalainen and Morton 1987) and the Batch Apparent Tardiness Cost (BATC)-Index (Cereíci and Banarjee, 2010), which are calculated in correspondence of specific trigger events. This method is greatly focused on the respect of the time constraint between cleaning and first diffusion operations and between the two consecutive diffusions. The latter algorithm relies on other priorities indexes closed to the one developed by Tu and Chen (2008), but with a different batching policy compared to the former. The keystone of this model consists in managing differently the lots that have to perform time-constrained operations and the others. For the lots type that are subjected to time constraints the model seeks to respects them, while for the others it aims to reduce flow time.

The remainder of this paper is organized as follows. In section 2 the literature related to batching and scheduling approaches with time constraints is reviewed. The analysed problem is described in Section 3. Section 4 presents Model 1, while section 5 Model 2. Section 6 presents conclusions and directions for future research.

2. Literature Review

Researchers have addressed the problem of composing and dispatching batches in different ways. Proposed methods can be classified according to various elements: (i) machine environment, e.g. work area, parallel machines; (ii) process restriction, e.g. re-entrant flow, time constraints, sequence-dependent; (iii) objectives, e.g. related to cycle time, throughput, on time delivery (Mönch et al., 2011). The presence of so many elements has brought to different solving techniques, i.e. mathematical programming, simulation and heuristic algorithms (Mathirajan and Sivakumar, 2006).

Batching and dispatching problems are often solved by means of a rule based on the value of an index that takes into account lot's features. Vepsalainen and Morton (1987) develop an index called ATC, which assigns the priority on the basis of the expected tardiness cost per immediate processing requirements. The aim is to minimize the sum of weighted tardiness, in a job shop. An evolution of this index is presented by Cereíci and Amarnath (2010). They focus on mean tardiness performance of a batch machine in a two-stages system by including an upstream unit capacity machine. They propose two new control strategies, called BATC-I and BATC-II, in order to compose and dispatch the batches. Akçali et al. (2000) focus on a loading policy based on the minimum batch size, and different dispatching policies, based on an index called critical ratio of lots, in diffusion area of a wafer fab. They suggest that by setting a variable minimum batch size, which depends on the production volume, the flow time can be reduced. They use simulation in order to test the different approaches, and the results show that the loading policy has a significant influence on the flow time and on overall cycle time of the products, while dispatching policy has less significant effect. Some approaches incorporate information about the state of the fab. Solomon et al. (2002) develops a dispatching policy to be used in batch-processing machines that incorporates information about future arrivals and the status of critical machines, so to balance the time that lots spend waiting at the batch-processing machines and the time spent in setups, in order to improve the makespan. Also Cigolini et al. (2002) consider the future arrivals for scheduling several products on parallel batching machines. Especially they focus on the problem, when there is less than a full load batch queuing at a batching machine. The procedure they adopted compares the total delay of lots in the queue, if they start immediately an incomplete batch, versus the one they would have if they wait for the next arrival. A simulation model tests the method, which preforms better than the actual industry practice. Flower et al. (2000) present a dispatching policy, base on NACH (Fowler, 1992) strategy, which considers the future arrivals. They develop the policy by considering the case of multi-family and parallel batch machines. The strategy has a pull decision point, when a machine becomes available and a push one, when lots arrive in the queue. Furthermore, Neale and Duenyas (2000) present a new heuristic algorithm, which aims to minimize the average makespan by considering the upstream and downstream machines in the network. Others authors consider more objective functions or problems in the same time. Sup and Young (2003) develop three efficient polynomial time algorithms for minimizing three corresponding due date related measures including maximum tardiness, number of tardy jobs and total tardiness. On the other hand Yeong-Dae et al. (1998) develop a scheduling approach in order to face three problems at the same time: i) lot release control; determining the time and quantity of wafers to release, ii) mask scheduling in the photolithographic workstations and iii) batch scheduling, determining the size and sequence of the batches. Another influence factor is the number of considered lots family. Mönch et al. (2005) present an approach, which considers incompatible lot family and unequal lot ready time in a diffusion area with parallel batch machine. The aim is to minimize total weighted tardiness.

In a fab, there are different types of batch machines, which are able to perform different operations and have different batch capacity. Yang et al. (2013) develops three stages algorithm in order to scheduling batch processing machines and minimize the total tardiness. The first stage collects the information from the fab; in the second part the batch are formed and dispatched while in the third the
super-hot batch are considered and loaded first. Some works do not focus on scheduling lots and batches in a specific section, but aim to plan the dispatching for the entire fab. Caumond et al. (2008) introduced a framework based on a disjunctive graph to model the problem and on a mimetic algorithm for generating the lots scheduling on the machines. The aim is to minimize the total makespan of lots. The result on a medium scale instances proving that a high quality solution can be obtained in short computational time. Chen and Tang (2012) propose a new heuristics algorithm to solve the problems of flexible flow shop with re-entrant flows and queue time constraints. The algorithm is based on four priority rules, which are applied at the different stage of the system in order to minimize the number of tardy lots. The results show better performance than the other tested heuristic methods. Attar et al. (2013) present an efficient metaheuristic algorithm, called biogeography-based optimization (BBO). The method is compared with two other algorithms, showing that it outperforms the others.

As far as time constraints are concerned, both practitioners and researchers have recognized the relevance of the topic. Solutions have been proposed in industries other than semiconductors and considering both tardiness and makespan as objectives. For instance, Gicquel, et al. (2012) study hybrid flow-shop scheduling problem arising from a bio-process industry, where a variety of constraints have to be taken into account, such as limited waiting time between processing stages. They propose an exact solution approach for minimizing the total weighted tardiness, based on a discrete time representation and a mixed-integer linear programming formulation. Li and Li (2007) study the hybrid flow-shop scheduling problem with limited waiting time constraint in a multi stage process, characterized by parallel unit capacity machines. The objective is to minimize the makespan for a given set of jobs. They propose a recursive backtracking algorithm, which schedules each job from the first stage to the last.

Some works finally consider both batch-processing machines and time constraints. For example, Su (2003) proposes a heuristic algorithm to minimize the makespan on a two stages process, where at the first stage there is a batch machine and considering only lots belonging to the same family. Yugma et al. (2012) propose a solving method based on disjunctive graph representation. In addition they develop a constructive algorithm based on iterative sampling and Simulated Annealing, in order to improve the initial solution. The approach is implemented in a real fab and has brought improvements for the interactive scheduling. These contributions are few, and none proposes a model applicable to STMicroelectronics’s plant situation.

3. Problem description and model assumptions

The aim of this work is to provide two models based on heuristic algorithms, which address both the batching and the dispatching problems, taking into account the presence of a time constraints between two sequential stages involved in the wafer fabrication. The stages considered by the model are three: one cleaning operation and two diffusion operations. The algorithm needs to address these 3 problems simultaneously: (i) Which lots should compose a batch; (ii) Which batch should be dispatched first and (iii) When a batch should be dispatched. These decisions need to be taken to minimize the average flow time of lots, to avoid scrapped lots and to reduce the number of re-cleaned lots. The model should be easily implemented in the existing information system.

There are three lots’ flows under study. The first considers all lots that have to perform only a cleaning operation (hereafter called Type A), the second consider lots that have to perform a cleaning operation and a diffusion operation (Type B), while the last one consist of a cleaning operations and two consecutive diffusion operations (Type C). In addition, lots can have different recipe. Two generic lots can have the same recipe at the first stage and different recipe at the second stage. Therefore, it is not necessarily possible to load on the diffusion operation in the same batch the lots processed together in a batch at the cleaning operation. While the batches composed at the second stage can be loaded and worked directly at the third stage, without any recombination of lots, if the capacity limit of the machine is respected. Furthermore, at each stage, the lots can be processed by a set of parallel batch machines, which can have different maximum batch size.

In the plant under study, the lots can wait in the queue in front of the second and the third stage no longer than a threshold. The queue time is calculated from the moment, when the lots finish to be processed at a stage, to the instance when the lots are loaded on a load-space at the next stage. The threshold is equal to 6 hours between the cleaning and diffusion operation while is equal to 0 hours between the first and the second diffusion operation. If a lot exceeds the time constraint between the first and second stage, it must be reworked, while if a lot exceeds the time constraint between the second and the third stage it must be scrapped.

4. Model 1

The proposed algorithm is based on two main concepts, which are consistent with the major goals and imposed constraints described above: (i) To avoid lots waiting at the batch-machines at the second and third stages, due to lack of others lots to be included in the batch, the method seeks to form, right at the cleaning operation, batches with lots sharing the same flow and same batch affinity in all the stages, and (ii) in order to avoid to overstep the time constraint, the queues at the second and the third stage are monitored. The model is composed of three elements: (i) Prioritization: which aims to compose the batches and calculate the priority index to prioritize the batches; (ii) Trigger events and dispatching: which aims to define the instant when prioritization should be performed, and then, based on the prioritization results, the batch must be dispatched; and (iii) Queues control: which aims to limit the number of batches waiting at the second stage in order to avoid to overstep the time constraint.
The main parameters that are inputs and decision variables for the model are presented in table 1. To set the prioritization at the cleaning work area two indexes are tested. The first one is proposed by Vepsäläinen and Morton (1987) and is called ATC (see formula 2), while the second is the BATC (see formula 3) index developed by Cerecki and Banerjee (2010).

\[
ATC_j = \frac{V_j}{p_{ij}} e^{-\left(\frac{(d_j-p_i-t_{now})}{k}\right)}
\]

\[
BATC_k = \sum_{i=1}^{n_i} \frac{V_i}{p_{ii}} e^{-\left(\frac{(d_i-p_i-t_{now})}{k}\right)} \left(\frac{n_k}{b_{im}}\right)
\]

Formula (2) is calculated for each lot. It is composed of two parts. Firstly, a ratio between the cost for delay \((V_j)\) and the process time \((p_{ij})\), which represents the weighted shortest process time and gives priority to the lots with shorter process time, assuming equal cost for delay. The second part represents the index slack per remaining process time, which gives priority to the lots by comparing their remaining process time with the one of the queuing lots. The second index \((3)\), is the sum of the ATC index for the lots composing the batch weighted by the ratio of number of lots in the batch \((n_k)\) maximum machine capacity \((b_{im})\).

The second task is to define the trigger event. For starting the heuristic for forming and dispatching a batch at the cleaning work area it is: (i) A cleaning machine is empty, or (ii) A new lot arrives in queue at the cleaning and a cleaning machine is empty, or (iii) A machine at the diffusion work becomes free and a cleaning machine is empty. Whereas for the dispatching of the batches at the diffusion work area the trigger event is that there is at least one complete batch in front of the diffusion machine and at least one load-space of the diffusion machine is empty. The last task is the queue control between the cleaning and the first diffusion operation, which is accomplish by the following equation (4):

\[
TC \times b_l < \frac{WIP_i}{a_i} \times \mu_i
\]

(4) aims to check if a specific machine \(i\) can process all the lots in its queue without overstep the time constraint \((TC)\), by considering the waiting batches \((Wip)\), the service rate \((ui)\) and machine availability \((ai)\), in line with Little’s law. The Wipi is updated every time a new batch is dispatched to the cleaning operation and when a batch ends the diffusion operation. When the machine is broken, \(bi\) is set to zero and no lots are allowed to enter in the queue. In addition to the two different index types, the model considers three other decision variables: Wtmax, \(\Delta T\), and Tf. Definitions are provided in table 1.

### 4.1 Algorithm at the cleaning and diffusion work areas

After the index calculation (being either ATC or BATC index), the algorithm then orders the lots or batches according to its value. If the chosen lot has to perform only cleaning, the procedure looks for all the lots in the queue or arriving within a lapse of time \(\Delta T\), which have the same recipe and flow of the selected one. The algorithm then, starting from the lot with higher priority index, checks if it can be processed on the empty machine at cleaning and, if it is a lot of flow \(B\) or \(C\), checks if the queue in front of the diffusion machine is lower than the time constraint, by means of equation (4). If both checks are passed (case of types \(B\) and \(C\), then the algorithm looks for similar lots in terms of flow and recipe on cleaning and diffusion and group them in a batch, including lots arriving within \(\Delta T\) (note that if in the batch, there are lots not yet arrived, but arriving within \(\Delta T\) the machine is considered busy by the batch).

<table>
<thead>
<tr>
<th>J</th>
<th>Lot index</th>
<th>Wtj</th>
<th>Waiting time of lot (j) at first stage [hours]</th>
</tr>
</thead>
<tbody>
<tr>
<td>I</td>
<td>Machine index</td>
<td>Wmax</td>
<td>Maximum waiting time allowed for the lot at first stage [hours]</td>
</tr>
<tr>
<td>O</td>
<td>Stage index</td>
<td>K</td>
<td>Look-ahead factor ((1,5&lt;K&lt;4,5))</td>
</tr>
<tr>
<td>k</td>
<td>Batch index</td>
<td>ai</td>
<td>Average availability of the machine (i)</td>
</tr>
<tr>
<td>nk</td>
<td>Number of lots in a batch</td>
<td>bi</td>
<td>Boolean variable: 0 if the machine is broken 1 in the other case</td>
</tr>
<tr>
<td>ui</td>
<td>Average rate of the machine [equivalent batches/hours]</td>
<td>WIP i</td>
<td># of waiting batches in front of the machine (i) [equivalent batches]</td>
</tr>
<tr>
<td>plj</td>
<td>Process time of lot (j) at stage 1 [hours]</td>
<td>Vj</td>
<td>Cost of the delay on delivery for the lot (j) [€/hours]</td>
</tr>
<tr>
<td>Tno w</td>
<td>Time when the priority indexes are calculated [hours]</td>
<td>Tf</td>
<td>The machine for the second diffusion is booked for the batch hours after the cleaning operation [hours]</td>
</tr>
<tr>
<td>dj</td>
<td>Due date of the lot (j) [hours]</td>
<td>TC</td>
<td>Time constraint between two stages [hours]</td>
</tr>
<tr>
<td>(\Delta T)</td>
<td></td>
<td>bim</td>
<td>Capacity of the machine (m)</td>
</tr>
</tbody>
</table>

In case that the batch is still not saturated, the algorithm checks if there are lots that have been waiting in queue for more than Wtmax. If yes, the batch is loaded, despite the fact that it is not complete. Otherwise, the algorithm checks the next lot with the highest priority index. After having performed the cleaning operation, the lots arrive physically in the queue in front of the diffusion, here they have to wait for all the lots having the same batch index. This second part starts when a load-space of a machine for diffusion is empty. If here is at least one batch that can be processed on that machine in the queue, than the batches are processed in FIFO logic. The last part begins when a machine at diffusion is empty. If there is at least one batch that can be processed on that machine, than the batches are processed in FIFO logic. In order to ensure that the time constraint, between the first and second diffusion operation, will be respected, the batch will book
the load–space on the machine, which has to perform the second diffusion operation. This happens when the difference between $T_{low}$ and the end of the cleaning operation finishes is smaller than a threshold ($T_f$).

5. Model 2

In Model 1 lots of type A might experienced a high flow time, due to their waiting time in the queue in front of the cleaning machine, since we impose to wait for lots with batch affinity. Therefore, Model 2 is developed in order to overcome the likely deterioration of flow time performance, respecting the time constraint imposed between the consecutive stages.

Before explaining the model, we have to make an important specification regarding the queue in front of the cleaning work area.

This queue remains physically unique, but is logically split into two parts: (i) The former contains the B and C-type lots that are not allowed to be processed at the cleaning work area. (ii) The latter contains all the A-type lots, which need to perform just the cleaning step, B and C-typology lots which have been already assigned to a furnace of the diffusion work area. They are those lots that have been authorized to enter in this queue (for the authorization procedure for B and C-typology lots we will reserve further explanations). Among the lots in this latter logic queue, batches are assigned to the cleaning work area when a cleaning machine becomes idle. As the “Model 1”, also the “Model 2” consists of two main parts referred to cleaning and diffusion work areas, but it includes also a sub-algorithm to assign batches containing B and C-types lots to furnaces. Table 2 lists the variables and parameters that will be used.

<table>
<thead>
<tr>
<th>Table 2- Parameters of Model 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>l</td>
</tr>
<tr>
<td>t</td>
</tr>
<tr>
<td>DD&lt;sub&gt;i&lt;/sub&gt;</td>
</tr>
<tr>
<td>$T_{R1}$</td>
</tr>
<tr>
<td>$T_{D1}$</td>
</tr>
<tr>
<td>$Dim_b$</td>
</tr>
<tr>
<td>$Dim_{b, max}$</td>
</tr>
<tr>
<td>$T_{c, max}$</td>
</tr>
<tr>
<td>$\Delta t_c$</td>
</tr>
<tr>
<td>bD1</td>
</tr>
<tr>
<td>bD2</td>
</tr>
<tr>
<td>$\lambda$</td>
</tr>
<tr>
<td>$\rho$</td>
</tr>
<tr>
<td>$\rho_{D1}$</td>
</tr>
<tr>
<td>$\rho_{D2}$</td>
</tr>
<tr>
<td>$\psi$</td>
</tr>
</tbody>
</table>

In this second model, as in the first one, there are some decision variables to set and test. In addition to the $I_c$ index, the model considers some other decision variables: (i) $\rho$, which is the weight for the batch saturation in $I_b$ formula (and which is complementary with $\lambda$, the weight of the due date in the $I_b$ formula); and (ii) $\rho_{D1}$ and $\rho_{D2}$, which are the weights for the batch saturation on the first or the second diffusion respectively in $I_b$ formula ($\rho_{D1} + \rho_{D2} = 1$).

5.1 Sub-algorithm for assigning lots to diffusion machines

When one furnace at the diffusion stage within a $\Delta t_c$ becomes free, it triggers the following sub-algorithm. This includes a series of steps here listed:

1. Choose all those lots that can be processed on the selected machine (lots of type B and type C);
2. Batch them considering the recipe at the first diffusion operation;
3. For the lots of type C, the algorithms checks if there would be an empty machine that can process them at the second diffusion, after a slack of time equal to the average cleaning time plus the diffusion process time. If this condition becomes true, they are included in the batch.
4. For all the batches the algorithm calculates a priority index ($I_b$) by considering the lots forming the batch (this index, as explained afterwards has a different expression depending from the fact that it considers solely lots of type B, or lots of type B and C);
5. The algorithm chooses the batch with highest priority index ($I_b$) and moves the lots to the cleaning queue.

5.1 Algorithm at the cleaning and diffusion work area

The second model relies on two types of priorities index, one for the cleaning stage ($I_b$) and one for the diffusion stage ($I_d$).

$I_b$ can be computed according to the performance that should be achieved (e.g.: FIFO, COVERT, Critical Ratio). $I_b$ is similar in its conceptualisation to the priority index proposed by Tu and Chen (2008) and instead computed by considering two variants. In case the batch considered for the calculation includes only lots of type B, $I_b$ is computed as the following expression (5):

$$I_b = \lambda \cdot \frac{\sum_{l \in b_{B1}} t + r_l}{Dim_{b_{B1}}} + \rho \cdot \frac{Dim_{b_{D1}}}{Dim_{b_{D1,max}}}$$

If instead the batch considered for the calculation includes at least one lot of type C, the priority index is computed as the following expression (6):

$$I_b = \lambda \cdot \frac{\sum_{l \in b_{B1}} t + r_l}{Dim_{b_{B1}}} + \rho \cdot \left( \rho_{D1} \cdot \frac{Dim_{b_{D1}}}{Dim_{b_{D1,max}}} + \rho_{D2} \cdot \frac{Dim_{b_{D2}}}{Dim_{b_{D2,max}}} \right)$$

The trigger event for starting the dispatching at the cleaning work area is represented by the presence of an empty machine within the cleaning work area. The lots to be dispatched are the ones that belong to the logical part of the queue, which is made up by lots of type A, and of types B and C for which the cleaning have been allowed by the sub-algorithm.

For each lot l the algorithm allows to compute the priority index $I_c$ according to a priority rule (e.g.: FIFO, COVERT).
A batch is then formed by including those lots that can be processed on the available machine with higher Lc. Lots in queue at the diffusion have to wait for all the lots having the same index (Ib). The waiting time of these lots has an upper limit equal to (7):

$$T = (T_C - T_{c,max}) * \psi$$  

(7)

Thus the batches already in the first diffusion queue can wait for the lot with the maximum cleaning processing time among the lots in their batch, without exceeding the time constraint. By doing so, the algorithm checks if the missing lots are arriving within the TC. If yes, lots wait for the others with batch affinity. Otherwise, the batch is processed incomplete. If one machines at either the first or the second diffusion is down and the TC is going to be overtaken, the batch in front of these machines seeks for another eligible machine in order to avoid the re-cleaning or the scrapping.

6. Conclusion
Two models to define batches to be dispatched along a three stages work area in a semiconductor industry when there is time constraints to be considered have been developed. The presented models share some major characteristics, e.g. the use of priority indexes, the respect of the time constraint, by controlling the length of the queue in front of the first diffusion work area and by booking in advance a chamber in the second diffusion operation. The major difference in the second model is the decision to not batch in front of the cleaning work area and so to prioritize flow time instead of batch saturation as performance outcome. The performance of the two models will be tested with an experimental campaign. To this aim, a simulation model of the analysed system will be developed in Arena™ using ST Catania's plan data.

Acknowledgments
This work has been funded by the EU-project INTEGRATE. We would like to thank Dino Colonna and Daniele Pagano of STMicroelectronics for their support and all the WP4 team for their suggestions.

7. References
Tu, Y.-M. and Chen, H.-N. (2008), Shop-Floor Control Model in Batch Processes of Wafer Fabrication with Time Constraints, La Jolla, California.
Yang, D., Jia, W., Jiang, Z., & Li, Y. (2013). Variable Time Windows-Based Three-Phase Combined Algorithm for On-Line Batch Processing Machine Scheduling with
