Lima, A., Borodin, V., Dauzère-Pérès, S., & Vialletelle, P. (2019). Sampling-based release control of multiple lots in time constraint tunnels. *Computers in Industry, 110*, 3-11. doi:https://doi.org/10.1016/j.compind.2019.04.014

Sampling-Based Release Control of Multiple Lots in Time Constraint Tunnels

Alexandre Lima[1,2] Valeria Borodin[1] Stéphane Dauzère-Pérès[1,3] Philippe Vialletetelle[2]

[1]Mines Saint-Etienne, Univ Clermont Auvergne
CNRS, UMR 6158 LIMOS
CMP, Department of Manufacturing Sciences and Logistics
F-13541 Gardanne, France
E-mails: alex.lima@emse.fr, valeria.borodin@emse.fr, dauzere-peres@emse.fr

[2]STMicroelectronics Crolles
F-38926 Crolles, France
E-mail: philippe.vialletelle@st.com

[3]Department of Accounting, Auditing and Business Analytics
BI Norwegian Business School
0484 Oslo, Norway

Abstract

Semiconductor wafer fabrication probably includes the most complex and constrained manufacturing processes due to its intricate and time-varying environment. This paper focuses on Time Constraint Tunnels (TCTs), which can have a very high impact on the yield and reliability of final products. More precisely, the original problem faced by managers of controlling the release of multiple lots in a TCT is addressed in the context of a wafer facility operating in a High-Mix Medium-Volume manufacturing environment. To support the management of TCTs in an industrially acceptable context, a scheduling-based sampling method is proposed to estimate the probability that multiple lots released at the entrance of a given TCT leave this TCT on time. In order to investigate the industrial viability and identify the limitations of the probability-estimation approach, numerical experiments are conducted on real-life data and analyzed through the prism of several relevant performance criteria. Insights gathered from this numerical analysis are then used to discuss the specific management requirements that stem from the criticality of TCTs in semiconductor manufacturing facilities.

Keywords: Production control; Time constraints; Probability estimation; Semiconductor manufacturing; Multiple lots.

1. Introduction

In semiconductor manufacturing, a Time Constraint (TC) can be seen as a set-up between wafer fabrication steps required to guarantee the quality and yield of final products. Being defined by a start process step and an exit process step, a TC restricts the time spent between process steps to a maximum time not to be exceeded, in order to preserve the expected chemical and physical properties of wafers.

Time constraint management is a critical task, since exceeding TCs may lead to significant reworks

or even scrapping the lot altogether. Both the production yield and the time required to process a wafer lot in a semiconductor manufacturing facility (commonly called fab) may be drastically affected because:

- Wafers which do not respect the recommended time constraints are reprocessed if possible, or scrapped,
- Damaged wafer lots reinserted through rework channels burden and slow down the production process.

With the rapid development of technologies, the number of emergent time constraints in product routes (sequence of process steps to complete a product) grows continuously. Moreover, time constraints often follow each other in close succession to the point of overlapping. This induces the composition of so-called *Time Constraint Tunnels* (TCTs, see Figure 1). Against an already intricate production framework, the management of TCTs has become increasingly challenging, notably when operating in High-Mix Medium-Volume manufacturing environments.

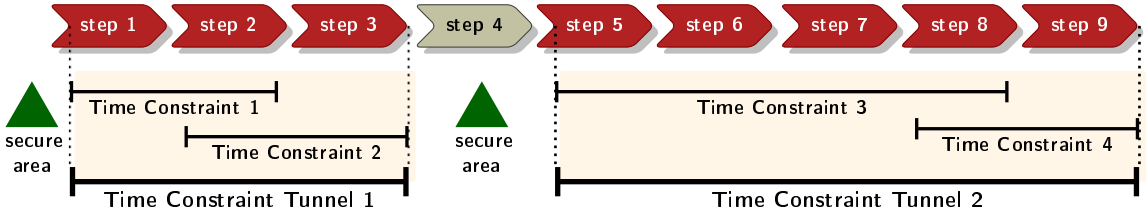

Figure 1: An illustrative example of two successive TCTs in the route of a lot (Lima et al., 2017)

In this paper, we consider the production control problem, never studied before to our knowledge, faced by TCT managers of deciding in real time the release of multiple lots in a TCT. As shown in the industrial instances of Section5, this may lead to considering more than 100 machines and thousands of process steps of hundreds of lots. To support the related decisions in a realistic industrial High-Mix Medium-Volume context, a novel appropriate and tractable probability-estimation approach is proposed to regulate wafer lots to be released in a given TCT. Based on the time-varying state of the wafer facility under study and by mimicking how lots are scheduled in the shop-floor, a series of binary response experiments are generated and observations are gathered for a subsequently statistical inference of the probability distribution, exhibited by the current full fab snapshot. Our first contribution lies in the new approach that extends to multiple lots in a non-trivial way our previous related work (Lima et al., 2017), that only deals with a single lot under different dispatching policies. Second, computational experiments performed on industrial data are presented and analyzed, highlighting the relevance of the proposed approach. Finally, managerial insights are provided, that discuss both the impact of TCTs on Operations Management and how the management of multiple lots differs from and complements the management of a single lot.

The remainder of the article is structured as follows. Section 2 provides a state-of-the-art related to time constraint management and emphasizes the originality of our problem, which is formalized in Section 3, of releasing multiple lots in a time constraint tunnel. In Section 4, a joint probability estimation approach is developed based on a list scheduling algorithm. In Section 5, computational experiments are conducted and performance indicators are analyzed in order to validate the decision-helping potential of

the proposed approach and to identify its limitations. Managerial implications are discussed in Section 6 to offer some insights for managers and decision makers in realistic and specific semiconductor manufacturing settings. Finally, some concluding remarks are presented in Section 7 including an outline of issues for future research.

2. Literature Review

With semiconductor manufacturing increasingly going towards smaller transistor sizes and more stringent quality requirements, wafers have become particularly sensitive to time constraints. As duly noted by Pappert et al. (2016) and Wang et al. (2018), the problem of managing TCs has evolved and is now an essential task in line with the pursuit of higher yield and quality specifications. Although time constraints are often found in the framework of semiconductor manufacturing, this is not the only industry which is confronted with these types of constraints. Time constraint management can also be encountered in the context of glass processing (Behnamian and Zandieh, 2011), the iron and steel industry (Chen and Yang, 2006; Li and Li, 2007) or even in bio processing plants (Gicquel et al., 2012).

The terminology surrounding time constraint varies from paper to paper with no established convention. The most frequent terms, that can be identified in the related literature, are the following:

- *Limited waiting times*: See e.g. Chen and Yang (2006); Li and Li (2007); Joo and Kim (2009); Behnamian and Zandieh (2011); Attar et al. (2013),
- *Time constraints* or *waiting time constraints*: See e.g. Robinson (1998); Kitamura et al. (2006); Chen and Yang (2006); Tu and Chen (2011); Klemmt and Mönch (2012); Sadeghi et al. (2015); Knopp et al. (2017); Lima et al. (2017); Kim and Lee (2019),
- *Time windows*: See e.g. Jung et al. (2014); Wang et al. (2018),
- *Queue times* or *Queue loops*: See e.g. Lee et al. (2005); Yurtsever et al. (2009); Wu et al. (2010, 2012b); Van Sickle and Hertzler (2006); Cho et al. (2014),
- *Time lags*: See e.g. Zhang and van de Velde (2010); Knopp et al. (2017).

Relative to each other, time constraints can form different configurations. Klemmt and Mönch (2012) distinguished five classes of TCs: **(1)** Adjacent or continuous TCs, i.e. time constraints between two immediately consecutive process steps, **(2)** TCs between two process steps that are consecutive and non-adjacent, **(3)** TCs that belong to the first two classes, **(4)** Overlapped TCs, and **(5)** TCs that belong to the third class and the fourth class. In the related literature, most of the studies have been designed and conducted for configurations with adjacent or non-adjacent consecutive time constraints in the first three classes. On the opposite, overlapped and general-structured configurations of TCs have received little scientific attention despite their importance in real-life applications (see e.g. Klemmt and Mönch (2012); Cho et al. (2014); Sadeghi et al. (2015); Knopp et al. (2017); Lima et al. (2017); Wang et al. (2018); Kim and Lee (2019)).

Time constraints are preponderantly handled in three different ways:

1. *Scheduling problems*, whose goal is to assign lots (jobs) to resources subject, amongst other constraints, to TCs, all the while optimizing a given criterion, see e.g. Yurtsever et al. (2009); Klemmt and Mönch (2012); Yugma et al. (2012); Jung et al. (2014); Knopp et al. (2017); Kohn et al. (2013); Attar et al. (2013); Chen and Yang (2006); Li and Li (2007); Gicquel et al. (2012); Behnamian and Zandieh (2011); Chien and Chen (2007); Zhang and van de Velde (2010); Cho et al. (2014); Wang et al. (2018); Kim and Lee (2019),
2. *Production control problems*, which investigate the production rate for production flows respecting TCs, see e.g. Lee et al. (2005); Wu et al. (2010, 2012b); Yu et al. (2013); Van Sickle and Hertzler (2006),
3. *Capacity planning problems*, which seek to determine the maximum number of Work-In-Process lots without violating TCs, see e.g. Robinson (1998); Kitamura et al. (2006); Tu and Chen (2011). Due to the heavily time-varying aspect of the problem of time constraint management, it appears that capacity planning methods have been losing in popularity, since a single tool breakdown can potentially shake up the whole capacity landscape in a very short amount of time.

With regard to solution methods, exact resolution approaches, usually based on Mixed Integer Programming (MIP), suffer from the curse of dimensionality and struggle to solve even small instances in acceptable computational times (Cho et al., 2014; Chen and Yang, 2006; Yu et al., 2013; Joo and Kim, 2009; Kim and Lee, 2019). Likewise, complex metaheuristics such as genetic algorithms (Chien and Chen, 2007), colonial competitive algorithms (Behnamian and Zandieh, 2011), or bio-geography based optimization (Attar et al., 2013), slightly mitigate the drawbacks posed by exact solution methods, by leaving significant room for improvement.

Real-life industrial applications are often tackled via heuristic techniques, sometimes coupled with local search approaches, e.g. variable neighborhood search (Kohn et al., 2013) or simulated annealing (Knopp et al., 2017). Overall, constructive heuristics seem to remain popular, as they offer tractable and scalable resolutions, which suitably conciliate computational efforts and the solution quality in industrial settings (Klemmt and Mönch, 2012; Lee et al., 2005; Yurtsever et al., 2009; Li and Li, 2007; Zhang and van de Velde, 2010; Jung et al., 2014).

Queuing models, while being computationally effective, come at the cost of a trade-off between solution quality and computational resources. In most of the cases, they are constructed based on very strong assumptions for the industrial world, where stationary states are never reached (Wu et al., 2010; Tu and Chen, 2011; Wu et al., 2012b,a).

To sum up, given the highly time-varying nature and large-scale size of problem instances, the management of TCTs turns out to be an intractable problem by classical scheduling approaches. As far as planning and production rate regulation methods are concerned, they are, by design, tailored for linear production lines with high redundancy. These methods prove to appropriately meet TCT management problems found in large-scale semiconductor production environments.

More precisely, the previously discussed state-of-the-art approaches do not answer our problem, i.e. how to support TCT managers in deciding *in real time* how many lots in a given set of lots should be released in a given TCT. Our goal is not to actually schedule lots to minimize a given objective function, or to study the stationary case in terms of production rates or capacity in TCTs.

In High-Mix Medium-Volume manufacturing facilities under study in this paper, dozens of different

products are often routed through production lines including heterogeneous machines with low equipment redundancy. In this context, a critical question that TCT managers need to answer in real time is: How many lots waiting in front of a TCT can be released?

To partly answer this question in complex industrial environments, Sadeghi et al. (2015) and Lima et al. (2017) propose a sequential sampling-based method to estimate the probability that a given lot leaves a given TCT on time. Driven by the industrial interest and viability of this novel approach, this paper generalizes it in a non-trivial way to multiple lots, illustrates the applicability of the approach through numerical experiments on industrial data, and discusses the managerial implications of integrating TCT decision-support tools in actual wafer facilities.

3. Problem Statement and Modeling

At a given point in time, let us consider a set $\mathscr{L}$ of wafer lots that are waiting in front of a Time Constraint Tunnel (TCT). For the sake of yield and quality considerations, we aim at quickly evaluating if and how the lots in $\mathscr{L}$, before being released in the TCT, can be processed in compliance with all time constraint requirements.

Multiple questions must be answered by TCT managers, in particular:

1. What is the maximum number of wafer lots which can be released in the TCT without violating time constraints? This is the main operational question that TCT managers are facing, and for which they wanted our help.
2. If lots have different priorities, how to evaluate and construct the best successful lot mixture? In this case, not only the number of lots to release should be decided, but also which ones.
3. How to quantify and what is the magnitude of the correlation between the lots released in the same TCT? A conditional probability estimation would be interesting to determine, as discussed in the perspectives in Section 7.

Let us investigate in this paper the first question posed above, by formulating it as follows: What is the maximum number of wafer lots that can be released in the TCT so that the completion on time of each lot is guaranteed by a given individual reliability threshold $\alpha \in (0,1]$?

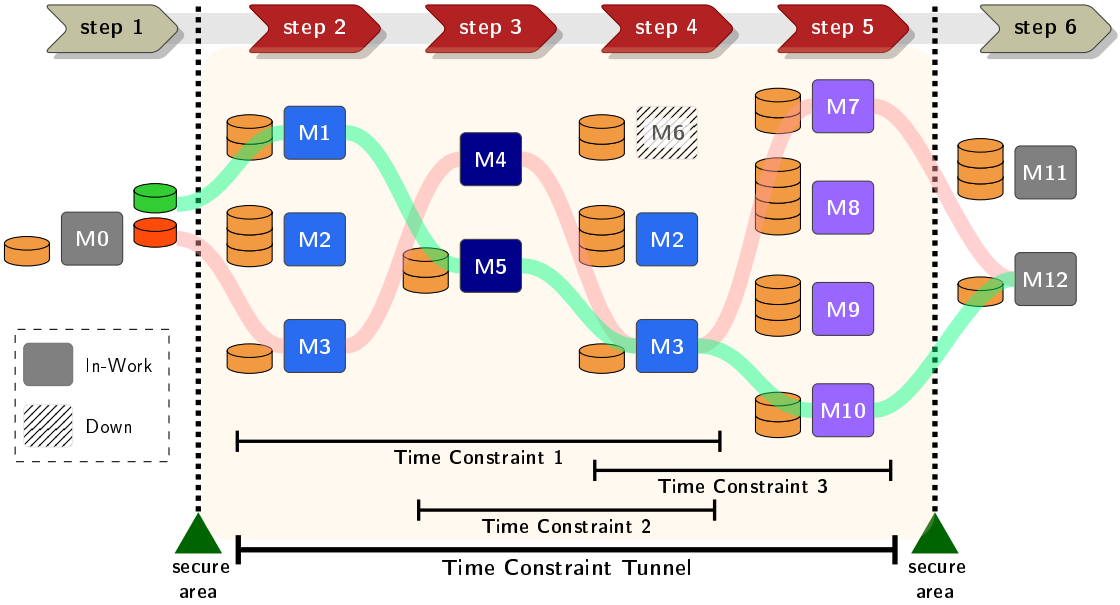

Figure 2: An example of a TCT with three TCs and two lots waiting to be released

It is important to note that this question needs to be answered in real time, i.e. quickly and by considering the current state of the wafer manufacturing facility. This makes the problem difficult to be handled analytically in a High-Mix Medium-Volume configuration. Figure 2 depicts the complexity involved in managing TCTs in such an environment. More precisely, one can see that TCTs include multiple features:

- *Intra- and inter- TCT resource sharing:* The lots to evaluate may share machines with several other lots that are not constrained by TCTs or are subject to other TCTs.
- *Re-entrant flows:* Lots may be processed by the same machine several times, e.g. the repetition of Machines M2 and M3 in different steps in Figure 2.
- *Heterogeneous machines and process step qualifications:* Each step can be potentially performed by different qualified (i.e. allowed to perform the process step) machines. The selected machines are represented by the highlighted arcs in Figure 2. Hence, not all machine sequencing configurations might be available.
- *Time-varying tool breakdowns:* As an example, Machine M6 might be down at run time. Note that the state of all machines could potentially change in a comparatively short amount of time.

Hence, in order to deal with this problem and support TCT managers, we develop a predictive algorithm, which uses as input the state of the fab when the analysis is performed. Specifically, real-time information, related to the machines and wafer lots in the area concerned by the TCT under evaluation, is used to design the system defined by the set of wafer lots $\mathscr{L}$ and the given TCT. To determine the joint empirical probability distribution of the successful processing of all lots $l \in \mathscr{L}$ in the TCT, a set of binary response experiments are performed and observations are gathered until the desired degree of sampling accuracy is achieved.

Before proceeding to the problem modeling, let us consider the following assumptions:

- Lots in $\mathscr{L}$ are of the same type, i.e. they have an equivalent sequence of process steps (route) to perform,
- The order in which the lots are released is predetermined according to a global priority factor detailed in the next section,
- Wafer lots are introduced by decreasing order of priority until a successful joint outcome occurs,
- The maximum number of lots allowed in the given TCT is determined by the last successful joint outcome,
- The generated observations (samples) are supposed to be independent,
- No correlation between the wafer lots under evaluation is taken into account.

Considering lots in $\mathscr{L}$ of the same type makes sense for two reasons. First, a given TCT is often associated to a single product type. Second, TCT managers usually decide to release lots of a given product type before moving to lots of another product type. Note that only lots to release in the TCT, i.e. lots in $\mathscr{L}$, are of the same product type, but that lots already in the WIP considered in our approach, i.e. lots in $\mathscr{L}_{WIP}$, are of different product types. Considering simultaneously lots of different product types to release in a TCT is an interesting perspective, which is discussed in Section 7

To evaluate in real time whether lots in $\mathscr{L}$ can satisfy the TCT, it is necessary to consider the lots currently in the Work-In-Process (WIP) that could compete for the same machines than lots in $\mathscr{L}$ in the TCT. Let us denote by $\mathscr{L}_{WIP}$ the set of these WIP lots. More precisely, lots in $\mathscr{L}_{WIP}$ might be processed in the near future on the same machines that lots in $\mathscr{L}$ might use in their process steps (also called operations). The relationships between process steps in the routes of lots in $\mathscr{L}$ and of lots in $\mathscr{L}_{WIP}$ are modeled via a disjunctive graph representation (Lima et al., 2017; Sadeghi et al., 2015). A disjunctive graph is defined by: **(i)** A set of nodes, with one node for each process step, **(ii)** A set of conjunctive arcs, with a conjunctive arc between every two consecutive process steps in a route, and **(iii)** A set of disjunctive arcs, with a disjunctive arc between every two process steps that may be processed on the same machine.

To convert the disjunctive graph into a conjunctive graph, lots are assigned and scheduled on the machines by the list scheduling algorithm given in (Lima et al., 2017; Sadeghi et al., 2015), which has the advantage to be computationally affordable and to mimic the way lots are scheduled in the shop-floor. For each resulting feasible schedule, the cycle time for any lot in $\mathscr{L} \bigcup \mathscr{L}_{WIP}$ is determined by using the properties of the graph. As such, we then can verify if any of the evaluated wafer lots has respected or not any of its associated time constraints.

4. A Multi-Lot Approach based on List Scheduling

Let us now introduce the procedure to determine the maximum number of wafer lots, which can be introduced in a given TCT without violating time constraints, i.e. so that each individual lot respects an imposed reliability threshold $\alpha \in (0,1]$:

- For each schedule, we determine whether or not lots in a subset of $\mathscr{L}$ have respected all their associated times constraints. More precisely, for each lot $l \in \mathscr{L}$, an estimation of the probability $\hat{\mathbb{P}}_l$ of respecting its time constraints is calculated as follows:

$$\hat{\mathbb{P}}_l = \frac{s}{N}, \ \forall l \in \mathscr{L} \tag{1}$$

  where s is the number of times when all time constraints of lot l are satisfied, and N is the sampling record length, i.e. the total number of generated schedules.

- A successful joint outcome is achieved when, for all lots:

$$\hat{\mathbb{P}}_l \geq \alpha, \ \forall l \in \mathscr{L} \tag{2}$$

  where $\alpha \in (0,1]$ is the required individual reliability threshold.

- The procedure is then iterated for an increasing size of subsets of $\mathscr{L}$ until an unsuccessful outcome occurs. The last successful joint outcome corresponds to the maximum number of lots that can be introduced into the system with an individual threshold level α.

Hence, this procedure determines the maximum number of wafer lots to be introduced in the TCT without violating time constraints, where each lot individually satisfies an imposed desired threshold level α. Note that the subset of lots $\mathscr{L}' \subseteq \mathscr{L}$ found in this way jointly guarantees a threshold level $\alpha^{|\mathscr{L}'|}$, since no correlation between evaluated lots is taken into consideration in this paper.

The proposed predictive multi-lot approach is based on a list scheduling algorithm, which is successively applied on subsets with increased cardinality of $\mathscr{L}$. A formal detailed description of the list scheduling algorithm can be found in Lima et al. (2017). The reader is also referred to (Lima et al., 2017) and (Sadeghi et al., 2015) for more information related to its use.

One of the key elements of the proposed approach is the selection policy. For each lot $l \in \mathscr{L}$, let us define a base priority factor $\pi^l_{\texttt{base}}$ and a global priority factor $\pi^l_{\texttt{total}}(\pi^l_{\texttt{base}}, \texttt{delay}^l)$, where $\texttt{delay}^l$ is the time during which lot l has been waiting at its current process step. The function governing $\pi^l_{\texttt{total}}(\pi^l_{\texttt{base}}, \texttt{delay}^l)$ is called the *standard dispatching rule* and can be formalized as follows:

$$\pi^l_{\texttt{total}}(\pi^l_{\texttt{base}}, \texttt{delay}^l) = \pi^l_{\texttt{base}}(1 + \texttt{delay}^l) \tag{3}$$

This function is based on the most common dispatching rule existing in the studied fab. In the same spirit, the selection of lots, which are waiting to be processed, is determined by the ranking of their respective $\pi^l_{\texttt{total}}$ values. In real-life settings, the lot selected for processing at a given machine most often corresponds to the lot with the largest priority factor $\pi^l_{\texttt{total}}$.

However, in order to generate a large number of schedules and to take the overall inherent variability of the fab into account, we propose to operate with a slightly modified version of the actual selection policy, where the probabilities of a lot scale proportionally with the global priority. Let $\mathbb{P}^l{}_{\texttt{select}}$ be thus the probability of lot selection in the scheduling step:

$$\mathbb{P}^l{}_{\texttt{select}} = \frac{\pi^l_{\texttt{total}}(\pi^l_{\texttt{base}}, \texttt{delay}^l)}{\sum_{k \in \mathscr{L}_{\texttt{ready}}} \pi^k_{\texttt{total}}(\pi^k_{\texttt{base}}, \texttt{delay}^l)} \tag{4}$$

where $\mathscr{L}_{\texttt{ready}}$ is the set of lots ready to be processed at a given process step in the list scheduling algorithm. By bringing together all aspects and elements discussed above, the pseudo-code given in Algorithm 1 formalizes the proposed multi-lot sampling approach.

Algorithm 1 Scheduling based multi-lot approach (TCT, α)

$\mathscr{L} = \emptyset$
Get `firstProcessStep` of TCT
Get `lotList` located at `firstProcessStep`
for $l \in$ `lotList` **do**
  if l is blocked at its step and unassigned **then**
    Calibrate the base priority $\pi^l_{\texttt{base}}$ to account for current fab rules
    Add l to $\mathscr{L}$
  end if
end for
Sort $\mathscr{L}$ by global priority $\pi^l_{\texttt{total}}(\pi^l_{\texttt{base}}, \texttt{delay}^l)$
$i = 0$
repeat
  $i = i + 1$
  Let $\mathscr{L}'$ be a subset of the first i lots in $\mathscr{L}$, i.e. $|\mathscr{L}'| = i$
  Release `ListSchedulingAlgorithm`($\mathscr{L}'$, TCT) given in (Lima et al., 2017)
  Compute $\hat{\mathbb{P}}_l, \forall l \in \mathscr{L}'$
until $i = |\mathscr{L}|$ or $\exists l \in \mathscr{L}'$ such that $\hat{\mathbb{P}}_l < \alpha$

Note that the considered priority-based random dispatching policy allows us to better reproduce the priority system commonly adopted in most wafer fabs. At first sight, the priority-based policy seems to be more computationally expensive than a pure random scheduling one, since it requires additional checking and updating process steps for every lot available for scheduling. Integrated in a sampling-based procedure, this policy accelerates the algorithm convergence by virtue of its definition. The interested reader is referred to our previous work (Lima et al., 2017) for a comparative study between different dispatching policies.

Note that when a lot is introduced into the set of lots ready to be processed by the list scheduling algorithm (Lima et al., 2017), all the samples are regenerated. Since lots are arbitrarily selected in subset $\mathscr{L}'$, the verification of the hypothesis on the imposed reliability threshold for the $|\mathscr{L}'|$ lots given below:

$$H_{|\mathscr{L}'|} : \hat{\mathbb{P}}_l > \alpha, \ \forall l \in \mathscr{L}' \tag{5}$$

assumes the re-evaluation of the TCT satisfaction of $|\mathscr{L}''|$ lots, $\forall \mathscr{L}'' \subseteq \mathscr{L}'$, i.e.:

$$H_{|\mathscr{L}''|} : \hat{\mathbb{P}}_l > \alpha, \ \forall l \in \mathscr{L}'' \subseteq \mathscr{L}' \tag{6}$$

In industrial practice, the implication (5) $\Longrightarrow$ (6) ensures an additional layer of safety towards the decision support in releasing any combination of lots in $\mathscr{L}'$. In other words, each new sampling iteration for $\mathscr{L}' \subseteq \mathscr{L}$ comforts the reliability level of releasing any subset of lots $\mathscr{L}'' \subseteq \mathscr{L}'$.

5. Numerical Experiments

5.1. Instance Description

To evaluate the resilience and the industrial viability of the proposed approach against different representative states of the manufacturing facility under study, 16 heterogeneous industrial instances have been selected over different periods of time separated by at least one month, whose general characteristics and description can be found in Tables 1 and 2. They are characterized by the following features:

- They are fairly large with up to 1,000 wafer lots and 150,000 nodes per disjunctive graph in some extreme cases,
- They include from 1 to 5 different TCTs,
- They correspond to different periods, spread over two years.

Table 1: General characteristics of TCTs in industrial instances

TCT	# TCs	# Machines	Average maximum time in TCs	# Process steps
1	3	57	8 hours to 24 hours	10
2	4	71	8 hours to 15 hours	8
3	4	69	2 hours to 12 hours	8
4	2	12	8 hours to 24 hours	3
5	5	112	8 hours to 24 hours	15

The selected time constraint tunnels come from different critical working areas of the manufacturing facility. Each TCT can include up to 5 time constraints and 15 process steps corresponding to different product types. In TCT 1, TCT 2, TCT 4 and TCT 5, the average maximum time in a TC varies from 8 hours to 24 hours, whereas it is shorter in TCT 3, varying from 2 hours to 12 hours.

Since each instance is recorded as a snapshot of the whole fab, it is not possible to control the state of all tunnels at any moment in time. As such, instances might not provide results for all of the selected time constraint tunnels. This is the case when either **(i)** A machine breakdown prevents the processing to be fully executed for a particular route, or **(ii)** No machine is qualified to perform a given process step. Both these cases result in production flow interruptions, commonly named *line stops* in the shop floor.

A special mention has to be made regarding TCT 1 for the period of time covering instances 4 and 8. The size of these instances is exceedingly large, due to the critical location of TCT 1 in the process flow, and an unusually high workload in the corresponding workshop when instances were recorded. It is particularly important to emphasize this specific case, as it explicitly highlights the linear scalability of the CPU time required by Algorithm 1 according to the graph size (see Table 3).

5.2. Parameter Selection and Tuning

In a preliminary study, Lima et al. (2017) has shown that the single lot sampling-based approach generalized in this paper, weakly converges after around 60 iterations. Based on this observation phase, the sampling record length has been fixed to $N = 30$. As a matter of fact, 30 generated feasible schedules prove to be sufficient to sample the industrial representative solution space with enough confidence, while remaining computationally efficient within the framework of a replication intensive approach.

Table 2: Description of industrial instances

Instance	TCT 1		TCT 2		TCT 3		TCT 4		TCT 5	
	# Lots	# Nodes	# Lots	# Nodes	# Lots	# Nodes	# Lots	# Nodes	# Lots	# Nodes
3	-	-	403	2,751	208	1,060	437	2,451	567	4,215
4	720	141,868	-	-	243	1,077	434	2,856	521	4,101
8	855	157,882	509	3,426	318	1,320	401	1,381	532	3,919
9	-	-	-	-	387	1,771	606	2,887	535	3,045
10	-	-	642	4,097	453	1,896	639	2,885	607	3,699
11	-	-	596	4,126	482	2,067	350	2,051	697	5,063
12	-	-	680	5,093	479	1,867	-	-	723	5,455
13	-	-	632	4,739	410	1,809	345	1,871	723	5,178
14	-	-	-	-	333	1,106	-	-	-	-
15	-	-	-	-	429	1,414	-	-	-	-
18	332	2,787	869	7,540	391	1,586	-	-	-	-
19	421	3,562	1,017	7,814	-	-	-	-	-	-
21	281	1,209	894	5,977	501	1,512	-	-	-	-
22	229	1,097	896	6,398	-	-	-	-	-	-
23	322	1,713	961	6,414	-	-	-	-	-	-
24	577	2,641	-	-	-	-	-	-	-	-

Algorithm 1 has been tested for two reliability threshold values $\alpha \in \{0.8, 0.9\}$. These values correspond to a low risk approach to managing TCTs, commonly adopted in production. Note that choosing values of α strictly smaller than 0.8 is not relevant from a production standpoint, since the involved compound risk becomes drastically lower than the required individual threshold level.

A maximum of 10 lots is allowed to be introduced in a TCT at the same time, i.e. $|\mathscr{L}'| \in \{1, 2, \ldots, 10\}$ or $|\mathscr{L}| = 10$. Algorithm 1 being conceived for reaching the saturation of a given TCT, a maximum of 10 lots is thus a very large number from a production standpoint.

Computational experiments were carried out on a machine with a 4-core Intel Xeon E3-1240 CPU clocked @ 3.50 GHz. The numerical results can be found in Table 3.

5.3. Result Analysis

Consistent with the industrial requirements, the proposed sampling approach proves to be affordable, being able to provide an answer in an acceptable amount of time (less than 5 minutes for almost all instances and time constraints with the exception of a few notable cases), even though sampling-based approaches are known to be inherently computationally expensive. Note that, in every other case (except for TCT 1, instances 4 and 8) the CPU time can be brought down to below 5 minutes by adjusting the value of $|\mathscr{L}|$. As aforementioned, the maximum number of lots to release is considered large in terms of production standards. An illustrative example is TCT 2.

Three distinct TCT profiles can be identified after a cross and in-depth analysis of Table 3, each corresponding to a different state of the TCT:

1. *Non passing, high difficulty tunnel state.* This profile is apparent in all TCTs, except for TCT 2, being very well illustrated by TCT 3 and TCT 5. Across most of their instances, $\mathbb{P}_l < \alpha$ when $|\mathscr{L}| = 1, l \in \mathscr{L}$, e.g TCT 3, instances 9 to 19 and TCT 5, instances 10 to 13. That is to say, for the given value of the risk threshold α, it is already not acceptable to release even a single lot in these TCTs.

The occurrence of this profile is determined and can be explained by several factors:

- The selected tunnels are highly critical, and thereby require a human manual dispatch, release and follow up of the lots throughout the entirety of the tunnel. In particular, this is the case for TCT 3 and TCT 5.
- The tunnel is currently at a reduced throughput capacity due to machine breakdowns. This state explains the more sporadic cases such as TCT 1 (instances 21 and 22), or TCT 4 (instances 9 to 11).
- Not all dispatching mechanisms existing in the fab have been included in the algorithm. Apart from the sampling convergence weakness, a coarse problem modeling leads to more pessimistic outlooks against configurations encountered in real life. On the upside, handling the problem at a lower layer of granularity also adds an additional level confidence on the validity of results when a successful outcome is obtained.
- For a given TCT and instance, the generated sampling set may not sufficiently represent the reality. While this is theoretically possible, the odds of it happening with the selected sample size are fairly low, according to the preliminary study conducted by Lima et al. (2017).

Let us also discuss the impact of the reliability threshold parameter α through the prism of some cases. For example, consider TCT 1 (instances 4 and 8). The lower value of $\alpha = 0.8$ makes a difference against $\alpha = 0.9$, by allowing a single lot to go through the tunnel. Hence, concluding about the state of a given TCT strongly depends on the value selected for α. This is a critical parameter that must be tuned by the decision makers in a production context.

2. *Non saturated, clearly passing, tunnel state.* Certain TCTs and instances have a 100% acceptance rates of the evaluated lots. In other words, even when simultaneously releasing 10 lots, all lots are able to go through their routes without incurring any TCT violations. This is notably the case for TCT 1 (instances 19 and 24), as well as for TCT 4 (instance 8) for both considered values of $\alpha \in \{0.8, 0.9\}$. The algorithm response means that the tunnel is fully unsaturated or in a high capacity configuration. For example, a high capacity configuration can occur when all the available machines required for the processing of the lots to release are up and running. However, note that a high machine availability remains a rare event, since process quality standards impose frequent maintenance operations on machines in semiconductor manufacturing facilities.

Table 3: Computational results

Instance	α	TCT 1		TCT 2		TCT 3		TCT 4		TCT 5	
		Lots	**CPU** (s)	**Lots**	**CPU** (s)	**Lots**	**CPU** (s)	**Lots**	**CPU** (s)	**Lots**	**CPU** (s)
3	0.8	-	-	10	179	2	7	3	64	3	117
	0.9	-	-	5	45	2	3	2	20	2	35
4	0.8	1	2,181	-	-	0	3	1	16	1	29
	0.9	0	1,066	-	-	0	1	0	8	0	13
8	0.8	1	2,195	10	421	1	9	10	141	1	49
	0.9	0	1,069	10	198	0	4	10	66	0	22
9	0.8	-	-	-	-	0	8	0	22	8	168
	0.9	-	-	-	-	0	8	0	22	5	103
10	0.8	-	-	1	30	0	10	0	25	0	2
	0.9	-	-	1	30	0	10	0	24	0	30
11	0.8			1	27	0	9	0	8	0	43
	0.9	-	-	1	27	0	9	0	8	0	43
12	0.8	-	-	1	37	0	7	-	-	0	44
	0.9	-	-	1	37	0	7	-	-	0	47
13	0.8	-	-	9	309	0	6	2	13	0	42
	0.9	-	-	8	269	0	6	1	6	0	42
14	0.8		-	-	-	0	3	-	-	-	-
	0.9	-	-	-		0	3	-	-	-	-
15	0.8	-	-	-	-	0	4	-	-	-	-
	0.9	-	-	-	-	0	4	-	-	-	-
18	0.8	5	34	5	399	0	4	-	-	-	-
	0.9	0	6	3	239	0	4	-	-	-	-
19	0.8	10	106	3	303	-	-	-	-	-	-
	0.9	10	112	4	527	-	-	-	-	-	-
21	0.8	0	2	7	572	0	7	-	-	-	-
	0.9	0	3	7	655	0	8	-	-	-	-
22	0.8	0	2	6	494	-	-	-	-	-	-
	0.9	0	2	5	471	-	-	-	-	-	-
23	0.8	3	13	1	77	-	-	-	-	-	-
	0.9	0	4	1	90	-	-	-	-	-	-
24	0.8	10	111	-	-	-	-	-	-	-	-
	0.9	10	124	-	-	-	-	-	-	-	-

3. *Passing tunnel state, with possible saturation.* This case is the most interesting, as it emphasizes the ability of Algorithm 1 to assess the current maximum capacity of the tunnel. A time-varying maximum lot release threshold is a very valuable information from a production standpoint.
   Additionally, this state reveals the crucial importance of the value of α and its effect on the algorithm output. The 10% difference in the acceptance threshold is sufficient to clearly observe its non-linear impact on the quantity of lots to release. This is best exemplified by extreme cases, e.g. TCT 1, instances 18 and 23, where respectively 5 and 3 lots can be released for $\alpha = 0.8$, while no lot can be released for $\alpha = 0.9$. The non-linear relation between α and the number of lots to release can also be observed in TCT 2 (instance 3) and TCT 5 (instances 3 and 9).

The above discussed findings shed light on the intrinsic interactions at stake between lots within a TCT. Because there are several possible machines that might be able to process certain process steps, a

slightly lower acceptance threshold can allow for a comparatively larger number of lots to go through, for an appropriate value of α.

6. Managerial Implications and Industrial insights on Real-life Applicability

6.1. The impact of Time Constraint Tunnels on Operations Management

The nature of time constraint tunnels makes their management complex in the framework of traditional organization structures, encountered in semiconductor manufacturing. TCTs tend to span several workshops, while operations management usually splits the responsibility of manufacturing, processes and tools by workshops or areas. This makes the question of who is eventually responsible for the lots having to respect their TCTs, not obvious to answer.

While it may appear counter-intuitive at first, in the studied manufacturing facility, the responsibility of the TCT management belongs to the workshop which owns the last process step of a given time constraint tunnel. The reason for this is twofold:

- Certain tunnels have a very different subset of process steps and tools at the beginning, but finish in the same last subset of tools.
- Given that the workshop owning the first process step of a TCT is setting the throughput inside the TCT, such a distribution of responsibility ensures communication between the workshops. The last workshop must thus make sure that this throughput will not lead to the appearance of bottlenecks through the bullwhip effect and is appropriate for a proper handling of the product flow as a whole. This is particularly notable since one of the most important key performance indicators is expressed in terms of the number of moves out per time period (a *move out* is a wafer coming out of a process step). Consequently, the number of moves out in a TCT usually favors the last workshop rather than the first.

Beside this responsibility sharing, a dedicated fab level regulator has the task of supervising all TCTs and the communication with/between workshops in order to alert them of any potential issue, as mentioned in Section 3. Therefore, TCTs place themselves in an interesting contrast with classical operations management, as they belong and are managed locally at the workshop level, but also have to be tracked at the overall fab level.

6.2. From a Single Lot to Multiple Lots

This paper extends to multiple lots our previous related work, dealing with the release of a single lot in a TCT (Lima et al., 2017). Let us underline in this section the fundamental differences of the two approaches, and how they could coexist within the same industrial context.

A single lot release control helps to assist the transition from a human gate-keeping decision to whether or not to release a lot in a TCT, towards a fully automated decision support. After being properly configured in agreement with expert decision makers, such a tool could be used without any kind of human supervision for supporting TCT management at operational level, or in complement with a scheduler or other existing dispatching rules.

Releasing multiple lots comes directly as a reinforcement to existing human TCT management, and is designed to fit the thought process of a decision maker. While single lot releasing requires a regular

supervision of a TCT, that is not the case when releasing multiple lots. For example, in the framework of a single lot approach, one may evaluate whether a lot can be released at a given point in time $t = 0$, then would have to do it again at $t' = t + 5$ minutes, $t'' = t' + 5$ minutes, etc. It is thus impractical from a human decision support point of view, except for specific cases, such as very low throughput and high complexity tunnels (e.g. this is the case of TCT 2 in our industrial data, for a specific product type).

Meanwhile, the multiple lot approach is better adapted to a human response time. It can be consulted at a frequency that matches the existing level of vigilance of decision makers in managing TCTs. Thus, having, when possible, a list, or better yet, a sequence of lots to release, is an appropriate support to their existing natural work pattern. It will allow TCT managers to either:

- Saturate the TCT in such a way that all the lots satisfy their TCs. This means that decision makers might not have to worry about the TCT for even longer periods of time,
- Take immediate response to issues that may have arisen, or to plan ahead with the sequence they are given, if they decide not to release all lots simultaneously.

As a complement of the existing work-flow, the automated release of multiple lots is much easier to be implemented and tested out on the field than its single lot counterpart, without going through complex approval processes. It relieves the workload of the decision makers, while still being subject to human validation.

Additionally, there is another reason why the multiple lot approach, although more complex to solve, presents an industrial interest. A critical question in the shop floor when referring to a TCT is: *What is the capacity of the TCT?* This question is very difficult to answer in time-varying production environments. In this sense, the proposed multiple lot approach allows the maximum capacity of a TCT to be quantified at a given point in time. This output makes our approach very attractive to decision makers and top managers as it partially answers their existing expectations.

From a purely technical point of view, it is also much simpler to implement the proposed multiple lot approach as a standalone tool, and not to have it interfacing directly with critical systems of the fab, such as the Manufacturing Execution System (MES). On the contrary, this is a prime requirement for the single lot approach to be implemented in its fully automated state. Additionally, the multiple lot approach can, for example, be supported by a basic graphical user interface developed with office management tools.

7. Conclusions and Perspectives

This paper generalizes and extends our previous work (Lima et al., 2017), by proposing an approach that handles the case where multiple lots can be simultaneously released in a TCT. More precisely, an industrially resilient and tractable sampling-based method is proposed to determine *in real time* how many lots can be allowed in a TCT, while ensuring they exit the TCT on time.

Numerical experiments conducted on real-life instances have shown the industrial viability of the proposed probability estimation approach, despite the added layer of complexity induced by considering multiple lots. This complexity is intrinsically time-varying and sequence-dependent, and was mitigated in this preliminary work by assuming that all lots under evaluation have the same type and priority, and have to respect identical time constraints. Even limited by the aforementioned assumptions, the proposed

approach can serve as providing useful information on the state and capacity of TCTs in a time-varying context, and thus can support the management of TCTs.

Several aspects of the current approach are open to further analysis and studies:

- *Lot selection policy.* Having multiple lots to release in a given TCT not only makes more complex the intrinsic time-varying nature of the problem, but also introduces a sequence-dependent component, as the lots can behave in various ways. For a fixed point in time, a similar situation from a lot-to-machine perspective can have a completely different outcome according to the sequencing decisions taken on the lots to evaluate. In the considered framework, lots are arbitrarily selected. This selection policy does not necessarily guarantee that the best combination of lots will be finally obtained in terms of both the number of lots able to come out on time, and the cumulative priority of the selected lots. Trying out all the possible mixes for the initial input may lead to a combinatorial explosion.

  This raises the following question: How can we tackle the problem of determining the best combination of lots to release, when lots have different types, priorities and/or are subject to different TCs? To go further, assuming different lots at the entrance of a TCT implies that a selection policy, able to correlate the best possible output to the best input, must be designed.

- *Conditional probability estimation.* We implicitly made the assumption that all lots to be released are independent and uncorrelated. This is a strong assumption whose impact should be analyzed in industrial practice, in particular because the lots to be introduced, usually of the same product type, often compete for the same critical machines.

- *Modeling choices.* One can take into account batching constraints or a more sophisticated tool selection policy for scheduling, instead of the first available first served policy. Consistent with the targeted purposes, there is a balance to achieve between the level of reality abstraction and the computational increase induced by any additional feature integrated in the problem modeling.

  For instance, the implementation of the priority-based probability of lot selection described in Section 4 affects in a straightforward manner the complexity of the algorithm. The number of sequencing decisions in the fully random setting roughly depends on the number of lots and the graph size, in the worst case. However, the algorithm complexity entailed by the priority-based selection is multiplied once more by the number of lots, since it requires additional checking and updating of process steps for every lot available for scheduling.

  Furthermore, we assume that the tools have a set state during the entire length of the simulation. While this hypothesis may hold true for short time horizons, this is not necessarily the case for longer time horizons with TC times exceeding 12 hours. In order to better consider the potential stress that may be generated by critical tools going down, it would be interesting to include stochastic breakdowns into the approach. They could be based on the estimation of Mean-Time-Between-Failure (MTBF) for all machines, which can be derived from historical data.

Acknowledgments

This work has been partially financed by the ANRT (Association Nationale de la Recherche et de la Technologie) through the PhD number 2015/0899 with CIFRE funds and a cooperation contract between STMicroelectronics and Mines Saint-Etienne.

References

Attar, S., Mohammadi, M., Tavakkoli-Moghaddam, R., 2013. Hybrid flexible flowshop scheduling problem with unrelated parallel machines and limited waiting times. The International Journal of Advanced Manufacturing Technology 68 (5-8), 1583–1599.

Behnamian, J., Zandieh, M., 2011. A discrete colonial competitive algorithm for hybrid flowshop scheduling to minimize earliness and quadratic tardiness penalties. Expert Systems with Applications 38 (12), 14490–14498.

Chen, J.-S., Yang, J.-S., 2006. Model formulations for the machine scheduling problem with limited waiting time constraints. Journal of Information and Optimization Sciences 27 (1), 225–240.

Chien, C.-F., Chen, C.-H., 2007. A novel timetabling algorithm for a furnace process for semiconductor fabrication with constrained waiting and frequency-based setups. OR Spectrum 29 (3), 391–419.

Cho, L., Park, H. M., Ryan, J. K., Sharkey, T. C., Jung, C., Pabst, D., 2014. Production scheduling with queue-time constraints: Alternative formulations. In: 2014 Industrial and Systems Engineering Research Conference. Institute of Industrial and Systems Engineers (IISE), pp. 282–291.

Gicquel, C., Hege, L., Minoux, M., Van Canneyt, W., 2012. A discrete time exact solution approach for a complex hybrid flow-shop scheduling problem with limited-wait constraints. Computers & Operations Research 39 (3), 629–636.

Joo, B. J., Kim, Y.-D., 2009. A branch-and-bound algorithm for a two-machine flowshop scheduling problem with limited waiting time constraints. Journal of the Operational Research Society 60 (4), 572–582.

Jung, C., Pabst, D., Ham, M., Stehli, M., Rothe, M., 2014. An effective problem decomposition method for scheduling of diffusion processes based on mixed integer linear programming. IEEE Transactions on Semiconductor Manufacturing 27 (3), 357–363.

Kim, H.-J., Lee, J.-H., 2019. Three-machine flow shop scheduling with overlapping waiting time constraints. Computers & Operations Research 101, 93–102.

Kitamura, S., Mori, K., Ono, A., 2006. Capacity planning method for semiconductor fab with time constraints between operations. In: 2006 SICE-ICASE International Joint Conference. IEEE, pp. 1100–1103.

Klemmt, A., Mönch, L., 2012. Scheduling jobs with time constraints between consecutive process steps in semiconductor manufacturing. In: 2012 Winter Simulation Conference (WSC 2012). IEEE, pp. 2173–2182.

Knopp, S., Dauzère-Pérès, S., Yugma, C., 2017. A batch-oblivious approach for complex job-shop scheduling problems. European Journal of Operational Research 263 (1), 50–61.

Kohn, R., Rose, O., Laroque, C., 2013. Study on multi-objective optimization for parallel batch machine scheduling using variable neighbourhood search. In: 2013 Winter Simulation Conference (WSC 2013). IEEE, pp. 3654–3670.

Lee, Y.-Y., Chen, C., Wu, C., 2005. Reaction chain of process queue time quality control. In: Semiconductor Manufacturing, 2005. ISSM 2005, IEEE International Symposium on. IEEE, pp. 47–50.

Li, T., Li, Y., 2007. Constructive backtracking heuristic for hybrid flowshop scheduling with limited waiting times. In: International Conference on Wireless Communications, Networking and Mobile Computing 2007 (WiCom 2007). IEEE, pp. 6671–6674.

Lima, A., Borodin, V., Dauzère-Pérès, S., Vialletelle, P., 2017. Analyzing different dispatching policies for probability estimation in time constraint tunnels in semiconductor manufacturing. In: 2017 Winter Simulation Conference (WSC 2017). IEEE, pp. 3543–3554.

Pappert, F. S., Zhang, T., Rose, O., Suhrke, F., Mager, J., Frey, T., 2016. Impact of time bound constraints and batching on metallization in an opto-semiconductor fab. In: 2016 Winter Simulation Conference (WSC 2016). IEEE, pp. 2947–2957.

Robinson, J. K., 1998. Capacity planning in a semiconductor wafer fabrication facility with time constraints between process steps. Ph.D. thesis, Citeseer.

Sadeghi, R., Dauzère-Pérès, S., Yugma, C., Lepelletier, G., 2015. Production control in semiconductor manufacturing with time constraints. In: 26th annual SEMI Advanced semiconductor manufacturing conference (ASMC 2015). IEEE, pp. 29–33.

Tu, Y.-M., Chen, C.-L., 2011. Model to determine the capacity of wafer fabrications for batch-serial processes with time constraints. International Journal of Production Research 49 (10), 2907–2923.

Van Sickle, D. L., Hertzler, E. F., 2006. 300mm time constrained queue loop management. In: International Symposium on Semiconductor Manufacturing 2006 (ISSM 2006). IEEE, pp. 57–60.

Wang, M., Srivathsan, S., Huang, E., Wu, K., 2018. Job dispatch control for production lines with overlapped time window constraints. IEEE Transactions on Semiconductor Manufacturing 31 (2), 206–214.

Wu, C.-H., Cheng, Y.-C., Tang, P.-J., Yu, J.-Y., 2012a. Optimal batch process admission control in tandem queueing systems with queue time constraint considerations. In: 2012 Winter Simulation Conference (WSC 2012). IEEE, pp. 2284–2289.

Wu, C.-H., Lin, J. T., Chien, W.-C., 2010. Dynamic production control in a serial line with process queue time constraint. International Journal of Production Research 48 (13), 3823–3843.

Wu, C.-H., Lin, J. T., Chien, W.-C., 2012b. Dynamic production control in parallel processing systems under process queue time constraints. Computers & Industrial Engineering 63 (1), 192–203.

Yu, T.-S., Kim, H.-J., Jung, C., Lee, T.-E., 2013. Two-stage lot scheduling with waiting time constraints and due dates. In: 2013 Winter Simulation Conference (WSC 2012). IEEE, pp. 3630–3641.

Yugma, C., Dauzère-Pérès, S., Artigues, C., Derreumaux, A., Sibille, O., 2012. A batching and scheduling algorithm for the diffusion area in semiconductor manufacturing. International Journal of Production Research 50 (8), 2118–2132.

Yurtsever, T., Kutanoglu, E., Johns, J., 2009. Heuristic based scheduling system for diffusion in semiconductor manufacturing. In: 2009 Winter Simulation Conference (WSC 2009). IEEE, pp. 1677–1685.

Zhang, X., van de Velde, S., 2010. On-line two-machine open shop scheduling with time lags. European Journal of Operational Research 204 (1), 14–19.