Scheduling job families on non identical parallel machines under Run-To-Run control constraints

Ali Obeid, Stéphane Dauzère-Pérès, Claude Yugma, Ariane Ferreira

Scheduling Job Families on Non-Identical Parallel Machines under Run-To-Run Control Constraints

A. Obeid, S. Dauzère-Pérès, C. Yugma and A. Ferreira

École des Mines de Saint-Etienne, Centre Microélectronique de Provence
880 Avenue de Mimet F-13541 Gardanne, France
`{obeid, dauzere-peres, yugma, aferreira}@emse.fr`

Keywords: Scheduling, Parallel Machines, Advanced Process Control, Time Constraint, Integer Linear Programming.

1 Introduction

New challenges are arising when solving scheduling problems in semiconductor manufacturing plants (fabs) with Advanced Process Control (APC) constraints. In particular, a Run-To-Run (R2R) control loop for a given product on a machine requires to regularly collect data for the product on the machine. This paper aims at introducing and modeling a new scheduling problem in which there is a time constraint on jobs of the same product, i.e. the time interval between two consecutive jobs of the same product should be smaller than a given threshold. Two Mixed Integer Linear Programming models are presented for scheduling jobs on non-identical parallel machines with setup times.

In semiconductor fabrication plants (fabs), the diversity of products leads to numerous process types and manufacturing procedures. Advanced Process Control (APC) aims at controlling processes and equipment to reduce variability, to increase equipment efficiency, to collect and classify information on equipment, etc. APC is usually associated to the combination of Statistical Process Control (SPC), Fault Detection and Classification (FDC), Run to Run control (R2R), and more recently Virtual Metrology (VM). Scheduling and APC are usually studied separately although both have the objective to improve productivity. Integration of APC constraints when scheduling jobs needs to be studied to evaluate potential benefits but also new challenges.

In semiconductor manufacturing, lots containing wafers (25 or less in a lot) are processed in various work areas with different characteristics. Lots will be called jobs in this paper, and the lots of the same product will be called job family. A job comes back many times to the same work areas, and there are several hundreds of process steps in its routing. Given this complexity, scheduling is often optimized locally in each work area. Kubiak *et. al.* (1996) studied the problem of scheduling a reentrant job shop with different job families. They showed that the SPT job order was optimal for the single machine reentrant shop under certain assumptions. An example is scheduling in the photolithography area that can be seen as a scheduling problem on parallel machines with job family setups (also called s-batching). A setup is required before starting the first job of a family, but not setup is necessary between two jobs of the same family. Although research has been performed on this problem, very little has been done to integrate APC constraints. A R2R controller uses data from past process runs to adjust settings for the next run as presented for example in Musacchio *et. al.* (1997). Note that a R2R controller is associated to one machine and one job family. A machine can usually process a limited number of job families, that are said to be *qualified* on the machine. Machines are thus non identical. In addition, in order to keep its parameters updated and valid, a R2R control loop should regularly get data. This imposes an additional constraint on scheduling, since jobs of the same family have to be scheduled within a maximum time interval on each machine on which the family is qualified. The value of the time threshold depends on several criteria such as the process

type (critical or not), the equipment type, the stability of the control loop, etc. If this time constraint is not satisfied, a qualification run is required to be able to process again the job family on the machine. We assume in this paper that this qualification run cannot be performed within the scheduling horizon.

There are very few articles which deal with scheduling decisions while integrating APC constraints. The impact of APC on scheduling performances is analyzed by Li and Qiao (2008). They also study the scheduling of job families on parallel machines. However, they consider that machines are identical, that qualification runs can be scheduled and that the threshold between two jobs of the same family is given in number of jobs. We consider non-identical parallel machines, we assume that qualification runs cannot be scheduled and they will be performed after the scheduling horizon. The problem becomes more complicated, since the assignment of jobs to machines is critical to avoid qualification runs. Finally, we consider a threshold expressed in time instead of number of jobs. Both threshold types are actually relevant and are related. Yiwei *et al.* (2009) study the interaction between scheduling and APC on one machine, with setup times between two job families, and a qualification run when the R2R constraint is not respected. They show that a single machine makespan problem with multiple job types is NP-hard. Another example of integration of APC constraints in scheduling decisions can be found in Detienne *et al.* (2009), where metrology operations are optimally scheduled to minimize the risk of losing products in jobs.

The following section describes the problem in more detail, and proposes two mathematical programming models.

2   Problem Definition and Modeling

2.1   Definition and notations

The goal is to schedule, on a horizon discretized in T periods, a set N of jobs of different families (or job types) on a set M of parallel machines. The set of job families is denoted F, and $f(i)$ is the family of job i. We assume that the processing times p_f of all jobs in family f are equal. Machines are not qualified to process all jobs families. A setup time s_f on a machine is necessary to change from a job of a family f' to a job of family f, where $f \neq f'$. Finally, Run To Run control constraints are considered through a parameter γ_f, which corresponds to the maximum time (called time threshold in the sequel) between the process of two jobs of family f on a qualified machine. If this constraint is not satisfied, a qualification run will be required to qualify again the machine for f. The objective is to optimize a scheduling criterion, e.g. the sum of the completion times, while minimizing the number of qualification runs.

The notations are summarized below. The parameters are:

T: Number of periods in the horizon,

N: Set of jobs,

M: Set of machines,

F: Set of job families,

$M(f)$: Set of qualified machines to process jobs in family f $(M(f) \subset M)$,

p_f: Processing time of jobs in family f,

s_f: Setup time of jobs in family f,

γ_f: Time threshold for job family f,

$f(i)$: Job family of job i $(f(i) \in F)$.

It is important to recall that, if the time threshold γ_f is not satisfied for job family f on machine m, we assume that the qualification run required on machine m cannot be performed within the time horizon. In this case, we suppose that no job in family f can be processed on m.

The decisions variables are:

$x_{i,t}^m = 1$ if job i starts at period t on machine m, and 0 otherwise,
C_i: Completion time of job i,
$y_{f,t}^m = 1$ if the time threshold is not satisfied for family f on machine m at period t, i.e. a qualification run is required, and 0 otherwise.

2.2 Two mathematical programming models

A first model (M1) can be written as follows, where each job i is considered separately:

$$\sum_{m \in M(f(i))} \sum_{t=1}^{T-p_{f(i)}+1} x_{i,t}^m = 1 \quad \forall i \in N \tag{1}$$

$$\sum_{m \in M(f(i))} \sum_{t=1}^{T-p_{f(i)}+1} t.x_{i,t}^m + p_{f(i)} \leq C_i \quad \forall i \in N \tag{2}$$

$$\sum_{i \in N; f(i)=f} \sum_{\tau=t-p_{f(i)}+1}^{t} x_{i,\tau}^m \leq 1 \quad \forall t=1 \ldots T, \forall f \in F, \forall m \in M(f) \tag{3}$$

$$\sum_{\tau=t-p_{f(i)}-s_{f(j)}+1}^{t} x_{i,\tau}^m + \sum_{\tau=t-p_{f(j)}-s_{f(i)}+1}^{t} x_{j,\tau}^m \leq 1 \quad \forall t=1 \ldots T, \forall (i,j) \in N \times N \text{ s.t.} \tag{4}$$

$$f(i) \neq f(j), \forall m \in M(f(i)) \cap M(f(j))$$

$$\sum_{i \in N; f(i)=f} \sum_{\tau=t-\gamma_f+1}^{t} x_{i,\tau}^m + y_{f,t}^m = 1 \quad \forall f \in F, \forall m \in M(f), \forall t = \gamma_f \ldots T \tag{5}$$

$$y_{f,t-1}^m \leq y_{f,t}^m \quad \forall t=2 \ldots T, \forall f \in F, \forall m \in M(f) \tag{6}$$

$$x_{i,t}^m \in \{0,1\} \quad \forall t=1 \ldots T, \forall i \in N, \forall m \in M(f(i)) \tag{7}$$

$$y_{f,t}^m \in \{0,1\} \quad \forall f \in F, m \in M(f) \tag{8}$$

Constraint (1) guarantees that each job is scheduled once and only once on a machine in the scheduling horizon. Constraint (2) is used to determine the completion time of each job. Constraints (3) and (4) model the fact that only one job is processed at a time on a machine. Constraint (3) is written for jobs of the same family, i.e. for which no setup time is not required, whereas Constraint (4) is associated to pairs of jobs of two different families for which setup times are necessary. Constraint (5) ensures that either the time threshold is always satisfied for a job family f qualified on machine m, or a qualification run is necessary, i.e. $y_{f,t}^m = 1$. Constraint (6) assures that if a machine is disqualified at period t, then it is also disqualified in the following periods. Constraints (7) and (8) impose variables $x_{i,t}^m$ and $y_{f,t}^m$ to be binary.

When analyzing the problem, it is possible to see that, since there are no release dates on jobs and jobs of the same family have the same processing time, all jobs in a family can be interchanged in an optimal solution. Let us denote n_f the number of jobs in family f, and C_f the sum of the completion times of jobs in f. Model (M2) below is equivalent to (M1).

$$\sum_{m \in M(f)} \sum_{t=1}^{T-p_f+1} x_{f,t}^m = n_f \quad \forall f \in F \tag{9}$$

$$\sum_{m \in M(f)} \sum_{t=1}^{T-p_f+1} t.x_{f,t}^m + n_f(p_f - 1) \leq C_f \quad \forall f \in F \tag{10}$$

$$\sum_{\tau=t-p_f+1}^{t} x_{f,\tau}^m \leq 1 \quad \forall t = 1 \ldots T, \forall f \in F, \forall m \in M(f) \qquad (11)$$

$$\sum_{\tau=t-p_f-s_{f'}+1}^{t} x_{f,\tau}^m + n_f . x_{f',\tau}^m \leq n_f \quad \forall t = 1 \ldots T, \forall (f,f') \in F \times F \qquad (12)$$

$$\text{s.t. } f \neq f', \forall m \in M(f) \cap M(f')$$

$$\sum_{\tau=t-\gamma_f+1}^{t} x_{f,\tau}^m + y_{f,t}^m = 1 \quad \forall f \in F, \forall m \in M(f), \forall t = \gamma_f \ldots T \qquad (13)$$

$$y_{f,t-1}^m \leq y_{f,t}^m \quad \forall t = 2 \ldots T, \forall f \in F, \forall m \in M(f) \qquad (14)$$

$$x_{f,t}^m \in \{0,1\} \ \forall t = 1 \ldots T, \forall f \in F, \forall m \in M(f) \qquad (15)$$

$$y_{f,t}^m \in \{0,1\} \ \forall t = 1 \ldots T, \forall f \in F, m \in M(f) \qquad (16)$$

Constraint (9) ensures that n_f jobs are scheduled for family f, and Constraint (10) is used to determine C_f. Constraints (11) and (12) are equivalent to Constraints (3) and (4). The number of binary variables in Model (M2) is greatly reduced compared to the number of binary variables in (M1). For each family f, the number of the binary variables x is divided by n_f.

Two types of criteria can be considered. The first type corresponds to classical scheduling criteria such as minimizing the sum of completion times $\sum_{i \in N} C_i$ (or equivalently $\sum_{f \in F} C_f$ in Model (M2)) or the makespan C_{max}. The second type is associated to the number of qualification runs $\sum_{f \in F} \sum_{m \in M(f)} y_{f,T}^m$ (weights could be considered to differentiate between job families or machines). The objective function could be a weighted sum of both types of criteria. However, it seems more realistic to consider a lexicographical order where the number of qualification runs is prioritized over a pure scheduling criterion.

3 Conclusion

In this article, we presented an original and relevant scheduling problem on non-identical parallel machines, where each machine has Run-to-Run control loops. We proposed two Mixed Integer Linear Programming Models(M1 and M2), and our first numerical results show that the execution times of M1 on a standard solver are on average ten times larger than of M2. Further results will be presented in the conference. Our research perspectives include the development of exact and heuristic approaches, and the possible integration of another threshold type based on the number of jobs.

References

Detienne B., S. Dauzère-Pérès, and C. Yugma, 2009, "Scheduling Inspection Operations Subject to a Fixed Production Schedule", *4th Multidisciplinary International Conference on Scheduling: Theory and Applications (MISTA 2009)*, pp. 581-593, Dublin, Ireland, August 2009.

Li L., and F. Qiao, 2008, "The impact of the qual-run requirements of APC on the scheduling performance in semiconductor manufacturing", *4th IEEE Conference on Automation Science and Engineering*, pp. 242-246, Washington DC, USA, August 2008.

Musacchio J., S. Rangan, C. Spanos, and K. Poolla, 1997, "On the utility of run to run control in semiconductor manufacturing", *IEEE International Symposium on Semiconductor Manufacturing (ISSM 1997)*, pp. D9-D12, San Francisco, USA, October 1997.

Yiwei C., E. Kutanoglu, J. Hasenbein, and J. Qin, 2009, "Scheduling with Advanced Process Control Constraints", www.me.utexas.edu/~has/apc09.pdf.

Kubiak W., S. X. C. Lou, and Y. Wang, 1996, "Mean Flow Time Minimization in Reentrant Job Shops with a Hub", *Operations Research*, Vol. 44, No. 5, pp. 764-776, September-October 1996.