Optimizing multiple qualifications of products on non-identical parallel machines

Antoine Perraudat[1,2], Stéphane Dauzère-Pérès[1], Philippe Vialletelle[2]

[1] Mines Saint-Étienne, Univ Clermont Auvergne
CNRS, UMR 6158 LIMOS
CMP, Depart. of Manufacturing Sciences and Logistics
Gardanne, France
{antoine.perraudat,dauzere-peres}@emse.fr
[2] STMicroelectronics
Crolles, France
philippe.vialletelle@st.com

Keywords: *Semiconductor manufacturing, qualifications of products, non-identical machines, workload balancing.*

1 Introduction

In semiconductor manufacturing, machines must be qualified before applying *a recipe* (an operation on a product). A qualification certifies that a machine meets the necessary quality and yield requirements. As performing a qualification can be expensive and usually takes time, an efficient management of qualifications of *recipes* on machines is essential to the overall performance of the manufacturing facility, e.g., in terms of throughput or cycle time [2]. However, qualifications of machines are often dynamic, i.e. time-varying, due to quality losses and machine wear. At an operational decision level, work-center managers must propose *re-qualification* decisions to optimize the performance of the manufacturing facility. To support decision making and improve manufacturing performances, the qualification plan, i.e. the set of qualifications (recipe, machine) to perform, should be optimized to balance the workload on production machines.

In High Mix/Low Volume (HMLV) manufacturing facilities, determining an optimized (re-qualification) qualification plan to balance the workload between non-identical parallel machines in a work-center is complex because the number of recipes can be large (up to one thousand), the number of machines can be large (up to two hundred), and throughput rates can be very different from one recipe to another for a given machine, and from one machine to another for a given recipe. [2] proposes a nonlinear qualification management optimization model to determine a single optimal qualification in terms of workload balancing. [8] extends the optimization model proposed by [2] by considering the finite production capacity of each machine. These optimization models have interesting practical uses such as improving the throughput, reducing the cycle time and fixing

bottleneck machines with cross-qualifications. To the best of our knowledge, no efficient solution approach has been proposed to solve the qualification management optimization problem with multiple qualifications and finite production capacity. [3] proposes a potential Decision Support System (DSS) that uses a multi-qualification management model to propose qualification plans. However, no solution approach is discussed.

In this paper, we seek to pursue the work of [2] and [8] by considering multiple qualifications in workload balancing optimization. We propose and compare six solution approaches on *real industrial data* from a HMLV semiconductor manufacturing facility, characterized by shifting bottleneck work-centers, short product life cycles, frequent product mix changes, and a high production variability with frequent disqualifications. Our academic and industrial contributions are as summarized below:

- Academic: We propose and compare six solution approaches to solve the multiple qualification management optimization model.
- Industrial: Solution approaches are implemented in a fully functional DSS that is used in the Crolles site of STMicroelectronics in France to better anticipate and manage bottleneck machines and reduce cycle times.

In this paper, we recall the qualification management optimization problem in Section 2 to make the paper self contained. The six new solution approaches are presented in Section 3. In Section 4, a comparison of solution approaches is performed on industrial data from the considered 300mm HMLV manufacturing facility located at Crolles in France. Finally, we conclude and give perspectives in Section 5. The DSS is not presented in this paper for space limitations.

2 Problem statement

Consider a work-center in a semiconductor manufacturing facility. The work-center consists of M parallel non-identical machines. R different recipes need to be processed in the work-center by the end of the planning horizon. Each recipe has a positive demand and the throughput rate of recipe r on machine m is deterministic and known. Each machine has a finite production capacity over the planning horizon and can only run qualified recipes, and a qualification can only be performed if the machine is "qualifiable" for the recipe. Finally, the objective consists in determining a qualification plan of k qualifications (recipe, machine) that maximize each machine utilization so that the workload is balanced.

Indices:
m: Index for machines, $\in \{1, .., M\}$,
r: Index for recipes, $\in \{1, .., R\}$.

Parameters:
k: Number of qualification decisions to be made at the beginning of the planning horizon,
$q_{r,m}$: Is equal to 1 if machine m is initially qualified for recipe r. Is equal to 2 if machine

m is initially qualifiable for the recipe r. Is equal to 0 if machine m cannot be qualified for recipe r,
$tp_{r,m}$: Throughput rate (in seconds) of recipe r on machine m,
c_m: Production availability (in seconds) of machine m over the planning horizon,
d_r: Demand in number of wafers for recipe r over the planning horizon,
γ: Workload balancing parameter strictly greater than 1.

Decision variables:
$OQ_{r,m}$: Is equal to 1 if there is qualification procedure to start for recipe r on machine m at the beginning of the planning horizon, and 0 otherwise,
U_m: Capacity utilization rate of machine m,
$WIP_{r,m}$: Quantity of recipe r processed by machine m.

$$\min \quad \sum_m U_m^\gamma \tag{1}$$

$$\text{s. t.} \quad \sum_{r,m} OQ_{r,m} \leq k \tag{2}$$

$$U_m = \sum_r \frac{WIP_{r,m}}{tp_{r,m} c_m} \qquad \forall m \tag{3}$$

$$\sum_m WIP_{r,m} = d_r \qquad \forall r \tag{4}$$

$$WIP_{r,m} \leq d_r \qquad \forall r, \forall m \mid q_{r,m} = 1 \tag{5}$$

$$WIP_{r,m} \leq d_r OQ_{r,m} \qquad \forall r, \forall m \mid q_{r,m} = 2 \tag{6}$$

$$WIP_{r,m} \leq 0 \qquad \forall r, \forall m \mid q_{r,m} = 0 \tag{7}$$

$$WIP_{r,m} \geq 0 \qquad \forall r, \forall m \tag{8}$$

$$OQ_{r,m} \in \{0, 1\} \qquad \forall r, \forall m \tag{9}$$

Objective function (1) aims at balancing the workload on the machines, or equivalently at maximizing the machine utilization. Constraints (2) limits the size of the qualification plan to at most k. Constraints (3) compute the capacity utilization rate of each machine in the work-center. Constraints (4) ensure that for each recipe all the demand must be assigned to machines. Constraints (5)-(7) ensure that machine m can process recipe r only if it is qualified. Note that dual prices of constraints (5)-(7) indicate a potential gain in terms of workload balancing and will be used in an heuristic. Finally, constraints (8) are the non-negativity constraints. Constraints (9) are the binary constraints.

This optimization problem is NP-Hard [5]. When $k = 1$, determining the optimal solution is "easy" because every qualification can be evaluated separately. However, for large industrial instances, the number of qualifications to evaluate is large, up to 150,000 qualifications in some cases, which can be difficult to solve in a few minutes. To solve the continuous relaxation of nonlinear optimization model (1)-(9) (or when embedded in solution approaches), we consider a cutting-plane algorithm. This is motivated by the fact that practical workload upper bounds can be determined for each machine. For instance,

it is unlikely that a machine will have a workload greater than 800% of its production capacity (i.e. $U_m < 8$). Therefore, we initialize the linearization with a limited number of cuts (between $U_m = 0$ and $U_m = 8$) and a limited number of cuts are further needed to solve the optimization model with satisfactory precision.

3 Solution approaches

3.1 Heuristics inspired by discrete location problems

The first two solution approaches are inspired by heuristics for discrete location problems and were initially proposed in [5]. The first solution approach is a greedy heuristic (GH) inspired by "ADD" heuristics used for discrete location problems [4]. GH iteratively builds a feasible qualification plan. At each iteration, all possible qualifications (case $q_{r,m} = 2$) are evaluated, and the qualification that best improves the objective function (1) is selected. This operation is repeated until a qualification plan of k qualifications is determined. The second solution approach is a local search (LS) inspired by "ADD-REMOVE" used for discrete location problems [4]. LS is a best improvement local search. LS determines an initial feasible qualification plan with the GH, and then tries to replace one qualification at a time with an improving qualification. LS ends when the qualification plan can no longer be improved.

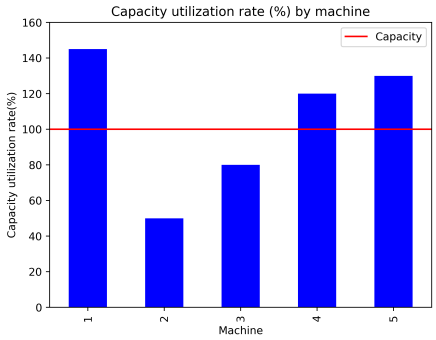
3.2 Dual prices

Although heuristics presented in Section 3.1 can be good starting points to determine satisfactory qualification plans, the number of qualifications to evaluate from one iteration to another can be substantial as the number of recipes and machines are large. On industrial instances, a few thousand qualifications have to be evaluated, which must be avoided in a DSS when short computational times are desired. Given the problem structure and the nature of data, we know from practical (industrial) experience that only a restricted set of qualifiable recipes and machines can lead to valuable qualification plans in terms of workload balancing. For instance, consider a work-center with five machines in parallel. The initial workload balance is presented in Figure 1. Machines 1, 4 and 5 are overloaded (i.e. $U_1, U_4, U_5 > 1.0$) while machines 2 and 3 are underloaded (i.e. $U_2, U_3 < 1.0$). Adding new qualifications to machines 1, 4 and 5 is irrelevant in terms of workload balancing because the machines would be even more loaded. Therefore, in this example, the search of good qualifications can potentially be restricted to machines 2 and 3. However, as there could still be many recipes, evaluating the qualifications of all qualifiable recipes on machines 2 and 3 could be too time-consuming when short computational times as desired in a DSS.

To identify a restricted but promising set of qualifications, we propose to use the dual variables of qualification constraints. If the decision variable $WIP_{r,m}$ is reformulated as the ratio of recipe r assigned to machine m, then the dual variable of the qualification constraint (6), when the optimization model (1)-(9) is solved for $k = 0$, can be interpreted as a potential gain in terms of workload balancing [6]. Then, it is possible to strongly restrict the search space in GH by evaluating qualifications associated to the smallest dual

variables (e.g., the eight smallest dual variables). Similarly, to GH and LS, two solution approaches, a greedy heuristic (GHDP) and a local search (LSDP) are proposed based on the values of dual variables. In both GHDP and LSDP, one qualification decision is taken at a time but qualifications are evaluated among a promising set of qualifications determined at the beginning of each iteration with the smallest dual variables. Another "instantaneous" greedy heuristic is proposed (IGH) and consists in building a feasible qualification plan made of the k qualifications associated to the k smallest dual variables.

FIG. 1: Initial workload balance.



3.3 Branch-and-bound approach

Similarly to the motivations that led to the use of dual variables, a branch-and-bound (BNB) approach is motivated by practical experience. In practice, we work on industrial data with preexisting qualifications, the qualification matrix is sparse, therefore, the overall number of possible qualifications is small. Among all possible qualifications, a small number of qualifications is relevant. In addition, we are interested in determining a qualification plan of limited size. Therefore, the continuous relaxation of the optimization problem (1)-(9) must be strong, and as we are able to quickly determine a feasible qualification plan with IGH, therefore pruning nodes in a branch-and-bound (BNB) approach should be "easy". In this paper, a best first branch-and-bound approach is explored. Bounding is performed by solving the continuous relaxation of the optimization model (1)-(9). Branching is performed on the qualification variable that is the closest to 1. A priority queue based on the smallest lower bound is implemented to explore the tree.

4 Computational study

The six solution approaches are compared on *real industrial data* from a 300mm HMLV manufacturing facility located in Crolles, France. Solution approaches are compared on 24 instances of a large work-center, where the number of recipes R varies between 700 and 800, the number of machines M is approximately equal to 200, and the initial number of new qualifications is on average equal to 0.7% of the product $R \times M$. Two different initial

qualification matrices are compared. The first matrix corresponds to the industrial data. To study the limit of the solution approaches, the second matrix is modified from the industrial one by allowing every qualification, i.e. all machines are "qualifiable" for all recipes. The computational time is limited to 180 seconds, $k \in \{0, 1, 2, 3, 4, 5, 6, 7, 8, 40, 100\}$, γ is set 4 and the numerical gap for BNB is set to 0.0001. In GHDP and LSDP, qualifications are evaluated from the 8 best dual variables. All solution approaches are implemented in Java 8 on a computer with an Intel(R) Xeon(R) CPU E3-1240 v5 @3.50GHz with 4 cores and 32 Go of RAM with Windows 10. Note that all solution approaches are parallelized (including BNB), i.e. up to 8 qualification plans can be evaluated at the same time. Linearized programs are solved with an open source solver [1]. Dual variables are extracted using the solver. Table 1 presents numerical results for the first qualification configuration. Table 2 presents numerical results for the second qualification configuration. Numerical results are reported in terms of mean gain (%) (with respect to the initial situation) and mean CPU (sec) over the 24 instances and for each value of k.

4.1 Numerical results

First qualification configuration: GH and LS perform poorly compared to GHDP. For instance, consider the case $k = 7$. The mean gain of GH is equal to 16.6% whereas the mean gain of GHDP is equal to 27.2%. Such a difference is explainable by the significant number of evaluated qualifications from one iteration to another in GH that prevents GH from determining a full qualification plan. On some instances, GH is actually unable to complete the first iteration. LSDP, based on the values of dual prices, is little relevant to improve GHDP. The computational time is multiplied by 2 to 4 but the mean gain is only increased by, at most, 0.2%. IGH has very short computational times but performs poorly compared to GHDP. BNB determines optimal solutions for all instances for $k = 1$ and $k = 2$, but loses in quality as soon as $k = 3$ due to the combinatorial explosion.

TAB. 1: Mean gain (%) and CPU (sec) over all instances for the first qualification configuration and a computational time of 180 seconds by solution approach.

	GH		GHDP		LS		LSDP		IGH		BNB	
k	Gain (%)	CPU (s)	Gain (%)	CPU (s)	Gain (%)	CPU (s)	Gain (%)	CPU (s)	Gain (%)	CPU (s)	Gain (%)	CPU (s)
1	15.9	182.0	15.8	2.7	15.9	187.3	15.8	4.9	15.1	2.3	15.9	7.5
2	16.5	186.7	20.8	5.0	16.5	189.7	20.8	10.1	17.6	2.3	20.9	35.5
3	16.5	186.4	23.0	7.5	16.6	188.3	23.1	15.5	18.9	2.4	23.2	81.1
4	16.5	185.9	24.6	9.8	16.6	188.4	24.7	20.9	19.9	2.3	24.8	128.8
5	16.6	185.8	25.6	12.1	16.6	188.5	25.8	27.2	20.4	2.3	25.6	164.7
6	16.6	187.3	26.5	14.5	16.6	186.5	26.7	32.5	20.9	2.4	25.4	170.0
7	16.6	187.4	27.2	17.3	16.6	186.9	27.3	39.8	21.6	2.3	26.0	179.3
8	16.6	185.8	27.7	19.6	16.6	188.9	27.8	47.6	21.9	2.5	25.8	180.9
40	16.6	188.9	29.5	97.8	16.6	188.0	29.5	181.2	25.9	2.5	25.9	180.8
100	16.6	187.0	29.6	181.4	16.6	186.2	29.6	181.3	28.3	2.7	28.3	180.8

Second qualification configuration: We can observe the same results for the second qualification configuration as for the first qualification configuration. However, differences between GH and GHDP and LSDP are exacerbated. We do not report LS in Table 2 because GH never completes its first iteration. This is because, as everything that

was not initially qualifiable is now qualifiable, the number of new possible qualifications explodes. About 150,000 qualifications must be evaluated on average in GH, which is intractable in short computational time. GHDP outperforms other solution approaches. LSDP is now more relevant to improve qualification plans determined by GHDP.

TAB. 2: Mean gain (%) and CPU (sec) over all instances for the second qualification configuration and a computational time of 180 seconds by solution approach.

	GH		GHDP		LSDP		IGH		BNB	
k	Gain (%)	CPU (s)	Gain (%)	CPU (s)	Gain (%)	CPU (s)	Gain (%)	CPU (s)	Gain (%)	CPU (s)
1	2.8	189.8	35.3	3.4	35.3	6.1	32.3	2.7	36.0	69.2
2	-	-	44.5	6.5	44.8	12.4	34.6	2.7	46.5	188.2
3	-	-	50.8	9.5	51.8	20.4	35.2	2.7	53.4	190.4
4	-	-	55.7	12.0	56.5	28.0	35.3	2.7	56.3	192.6
5	-	-	59.5	15.3	61.5	39.3	35.3	2.7	35.3	191.1
6	-	-	63.4	18.1	64.5	47.3	35.3	2.7	35.3	193.4
7	-	-	65.8	20.9	67.0	61.5	35.3	2.7	35.3	196.9
8	-	-	68.3	23.9	69.3	64.4	35.3	2.7	35.3	198.6
40	-	-	88.1	120.9	88.7	182.0	35.9	2.7	35.9	208.2
100	-	-	90.2	181.2	90.2	181.6	37.1	2.9	37.1	198.1

Generally, BNB approach is only for small values of k and the first qualification configuration. GHDP and LSDP are "immunized" against the combinatorial explosion (second qualification configuration) and outperform GH and LS. For a very small computational budget, instantaneous or of a few seconds, allowed in the decision support system, IGH is the best suitable approach, in particular for $k > 1$, because the computational time is independent of k, no matter the work-center and the qualification configuration. However, a qualification plan determined by IGH may be of poor quality, compared to GHDP, because one machine could be overqualified at the expense of other machines. Note that if many dual variables have the same value, or are very close, as in the second qualification configuration, the dual variable based solution approaches can lose in quality if a restricted number of qualifications is evaluated at each iteration. If the loss was substantial, the number of qualifications tested at each iteration of GHDP and LSDP could be increased to overcome the loss of quality. However, numerical results on industrial data show that this loss is very small.

Finally, this study shows that, although an optimization problem can be NP-Hard, studying the nature of data is primordial to design efficient solution approaches. For data from HMLV wafer fabs, using dual variables to guide the solution approach is shown to be efficient and robust against different qualification configurations. Analyzing dual variables to restrict the search space to few, but relevant, qualifications pays off.

5 Conclusions and perspectives

We propose six new solution approaches to solve a multiple qualification management optimization problem in semiconductor manufacturing. Solution approaches based on the values of dual variables of the qualification constraints are shown to outperform other solution approaches on industrial data of a specific work-center. Numerical experiments on

industrial data of another work-center, not presented in this paper for lack of space, confirm our conclusions. The nonlinear optimization model and solution approaches based on the values of dual variables are used in practice, in a fully functional decision support system, to propose qualification plans to work-center managers to improve workload balancing and therefore manufacturing performances. Each time work-center managers evaluate a scenario of maintenance operations, the DSS recomputes and optimizes the qualification plan and proposes it to work-center managers. The optimization approach helps work-center managers to rationalize and optimize their decision making. A feedback that work-center managers formulated is that the solving optimization model can lead to interesting qualification plans and that they are able to better anticipate potential problems such as bottleneck machines.

In the nonlinear optimization model proposed by [2] and [8], every parameter is deterministic. Studying the impact of the uncertainty of parameters on the quality of the solution approaches and qualification plans is interesting. Moreover, workload variables are continuous. However, units of demand are processed by batch of 25 units. Studying the relevance of considering integer variables is also relevant [7].

References

[1] Lougee-Heimer, R. The Common Optimization INterface for Operations Research: Promoting Open-source Software in the Operations Research Community. *IBM Journal of Research and Development* , 47(1):57–66. 2003.

[2] Carl Johnzen, Stéphane Dauzère-Pérès, and Philippe Vialletelle. Flexibility measures for qualification management in wafer fabs. *Production Planning and Control* , 22(1):81-90, 2011.

[3] Carl Johnzen, Stéphane Dauzère-Pérès, Philippe Vialletelle, and Claude Yugma. Optimizing flexibility and equipment utilization through qualification management. *IEEE/SEMI Advanced Semiconductor Manufacturing Conference* , 37-140. IEEE, 2009.

[4] Mark S Daskin. Network and discrete location: Models, algorithms, and applications. *John Wiley & Sons* , 2011.

[5] Carl Johnzen. Modeling and optimizing flexible capacity allocation in semiconductor manufacturing. *Doctoral dissertation* , 2009.

[6] Mokhtar S. Bazaraa, Hanif D. Sherali, and Chitharanjan M. Shetty Nonlinear programming: Theory and algorithms. *John Wiley & Sons* , 2013.

[7] Mehdi Rowshannahad, and Stéphane Dauzère-Pérès. Qualification management with batch size constraint. *In Proceedings of the 2013 Winter Simulation Conference* , 3707–3718. 2013.

[8] Mehdi Rowshannahad, Stéphane Dauzère-Pérès, and Bernard Cassini. Capacitated qualification management in semiconductor manufacturing. *Omega* 54: 50-59 , 2015.