OR Spectrum (2008) 30:431–452
DOI 10.1007/s00291-007-0093-4

REGULAR ARTICLE

Heuristic algorithms to solve the capacity allocation problem in photolithography area (CAPPA)

Shu-Hsing Chung · Chun-Ying Huang · Amy H. I. Lee

Published online: 1 June 2007

Abstract Wafer fabrication is one of the most complex and high competence manufacturing. How to fully utilize the machine capacity to meet customer demand is a very important topic. In this paper, we address the capacity allocation problem for photolithography area (CAPPA), which belongs to a capacity requirement planning scheme, with the process window and machine dedication restrictions that arise from an advanced wafer fabrication technology environment. Process window means that a wafer needs to be processed on machines that can satisfy its process capability (process specification). Machine dedication means that once the first critical layer of a wafer lot is processed on a certain machine, the subsequent critical layers of this lot must be processed on the same machine to ensure good quality of final products. We present six modified heuristics and a linear-programming-based heuristic algorithm (LPBHA) to solve the problem efficiently. The performance of the proposed algorithms is tested using real-world CAPPA cases taken from wafer fabrication photolithography area. Computational results show that LPBHA is the most effective one, and with a least

S.-H. Chung (✉)
Department of Industrial Engineering and Management, National Chiao Tung University, Hsinchu, Taiwan, ROC
e-mail: shchung@mail.nctu.edu.tw

C.-Y. Huang
Department of Business Administration, Ching Yun University, Jungli, Taiwan, ROC
e-mail: cyhuang@cyu.edu.tw

A. H. I. Lee
Department of Industrial Engineering and System Management, Chung Hua University, Hsinchu, Taiwan, ROC
e-mail: amylee@chu.edu.tw

average and a least standard deviation of deviation ratio of 0.294 and 0.085% compared to the lower bound of the CAPPA.

Keywords Photolithography area · Process window · Machine dedication · Heuristic · Linear programming

1 Introduction

In this paper, we consider the capacity allocation problem in photolithography (CAPPA), a variation of the capacity allocation problem considered by Leachman and Carmon (1992); Toktay and Uzsoy (1998); Chung and Huang (2001), and Hung and Cheng (2002), which has many real-world applications, particularly, in the semiconductor manufacturing and thin film transistor-liquid crystal display (TFT-LCD) manufacturing industries. Photolithography process uses stepper, a bottleneck workstation and the most expensive machine in a wafer fabrication factory, and the matched mask/reticle, a piece of glass with predefined circuit patterns, to transfer circuit patterns onto a wafer, and then forms tangible circuit patterns onto the wafer through etching operation (see Fig. 1). With the required number of processes in the photolithography, integrated circuitry products with preset functions are developed on the wafer. In practice, all fabrication steps before the next photolithography processes are classified to form a layer; hence the complexity of a product can be determined by the number of layers it requires.

As wafer fabrication technology advances to a higher precision level, the line width and the space between lines of IC diagrams (Fig. 2) copied from masks/reticles onto wafers in the photolithography area becomes smaller, and more stringent machine selection restrictions, the so-called process window control and machine dedication

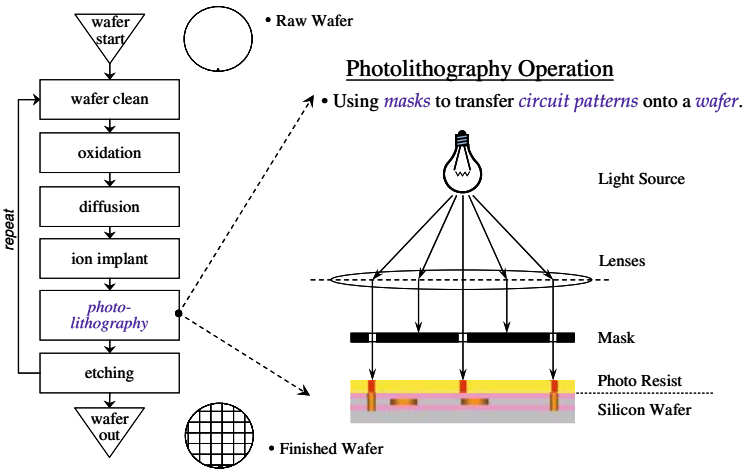


Fig. 1 The photolithography process

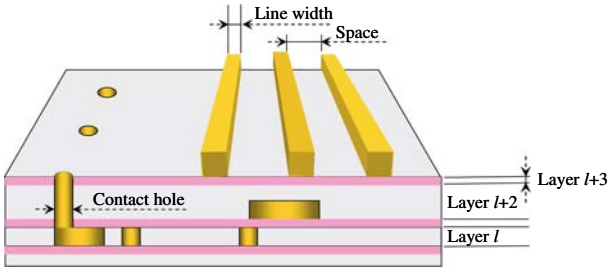


Fig. 2 An example of circuit pattern

control, are imposed on the photolithography area to ensure achieving the highest yield of wafer lots.

Process window constraint, also called equipment constraint or machine capability, is related to the stringent machine selection criterion to process high-end (higher precision level) fabrication technology so as to meet increasingly narrower line width, distance between lines, and tolerance limit. In other words, wafer lots could only be processed on machines that meet certain process capability (also called process recipe or process specification). Due to the difference among machines regarding to recipe processing, machines with different models in fact have varying functions to a certain extent even though they are grouped in the same workstation. Hence, the situation is that some machines can handle more process capabilities (simultaneously handle higher- and lower-end fabrication technology) while other stepper machines just handle less process capabilities (only handle lower-end fabrication technology). Figure 3 shows whether circuit patterns are properly transferred onto the wafer or not. Some related studies are as follows. Leachman and Carmon (1992) and Hung and Cheng (2002) develop a linear programming model to obtain a production plan for maximizing the profit with the consideration of machines' capability constraint. Toktay and Uzsoy (1998) transform the capacity allocation problem with machines' capability constraint into a maximum flow problem. However, only a single product type is considered in the study. Akçalı and Uzsoy (2000) study a shift scheduling problem arising from the photolithography area of wafer fabrication with the constraints of machines' capability, mask availability, and number of mask setup operations. They present a sequential procedure that separates the problem into capacity allocation and lot sequencing sub-problems. Chung and Huang (2001) propose a heuristic method, named COLA, to solve the capacity allocation problem by considering the process capability constraint in a single planning period, and the objective function is to balance the loading among machines. Chen et al. (2005) construct a capacity planning system to balance the load among fabs with the consideration of machine capability in a multiple-fabs environment. Their system is based on the pull philosophy and an assumption of infinite equipment capacity. Then, the release time and production fab of each lot can be determined.

The machine dedication constraint is set for layer-by-layer process on wafers so that the circuit patterns in such layers (named critical layers) can be correctly connected to provide particular functions. In other words, it will cause defective products

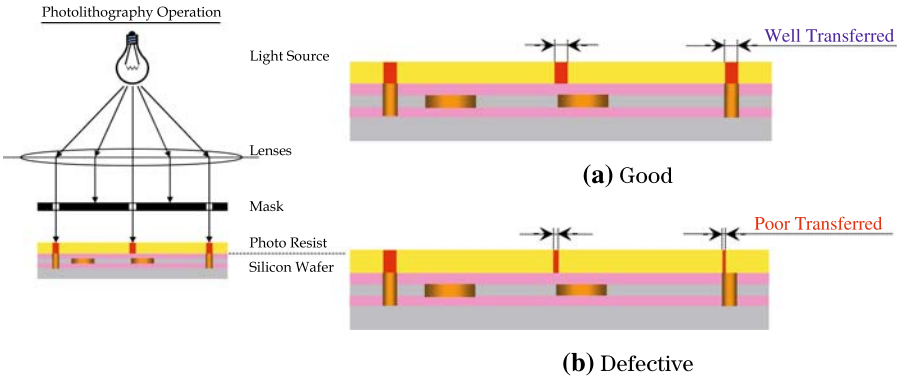


Fig. 3 Cases of process window characteristic

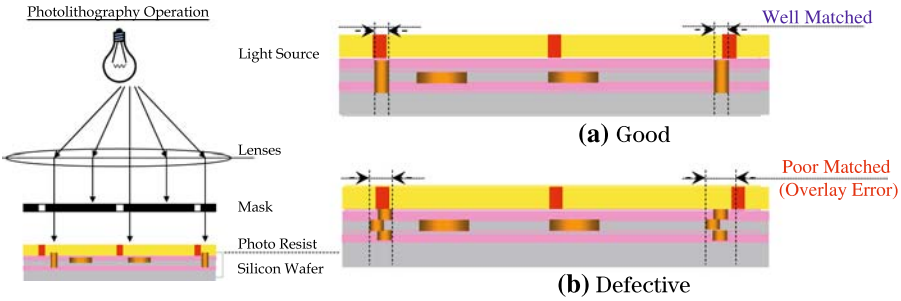


Fig. 4 Cases of machine dedication characteristic

if circuit patterns among these critical layers cannot be aligned and connected. Figure 4 shows two machine dedication cases. In Fig. 4b, the poorly matched circuit patterns result from the machine dedication problem. The spread of critical layer operations is different depending on product types. Usually, there is one critical layer operation after two to five non-critical layer operations. The alignment precision provided by different machines varies to a certain extent, even for machines of the same model type, which are referred to as machine difference. It has been restricted that when the first critical layer operation of a wafer lot is done on a particular machine, the rest of its subsequent critical processes need to be processed by the same machine to avoid the increase in defective rate due to machine difference. A related study is done by Akçalı et al. (2001), in which an investigation is conducted on the correlation between photolithography process characteristics and production cycle time by using a simulation model, and the machine dedication policy is set as one of the experiment factors. Experimental results indicate that the dedicated assignment policy has a remarkable impact on the cycle time.

With the use of advanced fabrication techniques, the impact of process window and machine dedication constraints on wafer fabrication is increasingly evident. Capacity requirement planning (CRP) is very difficult due to the fact that wafer fabrication has a special reentry characteristic and long cycle time, and the fact that the number of layers

contained in a product, the required process window, the number and distribution of critical layers are different for each product type. As a result, the effectiveness of the production planning and scheduling system is seriously impacted if the constraints of process window and machine dedication are not considered. In this paper, we tackle the capacity allocation problem in the photolithography area (CAPPA), which belongs to the CRP scheme, with considerations of process window and machine dedication that arise from an advanced wafer fabrication technology environment. Solving the CAPPA is to find a job allocation solution so that the process window constraint and machine dedication constraint are satisfied without violating the machine capacity. The objective function of a solution model for the CAPPA is to balance the load of machines in a workstation according to Chung et al. (2006). They showed that leveling the load among machines will help to maintain a stable production cycle time and lead to the accomplishment of the master production schedule (MPS) and detailed schedule (DS) on time.

Up to now, the CAPPA problem has not been tackled except by Chung et al. (2006), in which a mixed integer-linear programming (MILP) model is devised to solve the CAPPA problem. However, it usually requires a tremendous computational time for a large-scale CAPPA instance and thus loses the applicability in real environments. Therefore, we present six modified heuristics of Sule's algorithm (MSAs) and a linear-programming-based heuristic algorithm (LPBHA) for solving the CAPPA efficiently, and both new released work orders and in-process work orders in a planning horizon can be assigned to machines in a photolithography workstation. The performance of the proposed algorithms is tested by real-world CAPPA cases taken from the wafer fabrication photolithography area.

The remainder of the paper is organized as follows. Section 2 describes the CAPPA problem, analyzes its complexity and gives a demonstrative example. In addition, an MILP model is introduced to solve the CAPPA. Section 3, we present the proposed algorithms to solve the CAPPA efficiently. Section 4 uses a set of test problems, which are generated based on real-world cases taken from a wafer fabrication factory, to verify the feasibility and effectiveness of the proposed algorithms. In the last section, the research results are summarized.

2 Problem description and formulation

2.1 Problem description

The capacity allocation problem in wafer fabrication photolithography area (CAPPA) can be stated as follows. Given a wafer fabrication photolithography area with K machines and H types of process capabilities, there are I orders, L_i layers for each order, T time buckets in a planning horizon. Machines, the steppers, may have different types and numbers of process capability to process wafer lots. Solving the CAPPA is to find a job allocation solution so that the process window constraint and machine dedication constraint are satisfied without violating the machine capacity. The complexity of the CAPPA is NP-hard according to Low and Fang (2005). Low and Fang (2005) considered a load balanced demand points assignment problem (LBDPAP) in

Table 1 Process window of machines in the CAPA example

Machine no.	Process capability			
	1	2	3	4
1	1[a]	1	0	0
2	0	1	1	0
3	0	1	1	1

[a] 1 means that the machine has this certain process capability; 0 means that the machine does not have this certain process capability

a large scale wireless LANs environment. The LBDPAP concentrates on the issue of demand points assignment with the objective of minimizing the maximum load of all access points (APs). A demand point can only be assigned to an AP while the signal-to-noise ratio (SNR) value is greater than a certain threshold. The authors showed that the LBDPAP is NP-hard. When the number of planning period is one and none of the orders has critical layer operations, the CAPPA reduces to a LBDPAP. Thus, the CAPPA is NP-hard too.

Because of the reentry characteristic of wafer fabrication, we cannot merely plan the first or first several layers of process, and ignore the fact that such a machine assignment may have some impact on the future dispersion of the system load. The ignorance may lead to the derived planning results being a local optimum. Therefore, the load allocation of critical layer activities must cover the in-process and planned-to-release orders in a planning horizon that is long enough, i.e. the planning horizon must have at least the same length as the product cycle time. Actually, wafer fabricator usually uses a time horizon of 1 month to decide whether or not the short-term capacity is sufficient to accept an order. In addition, Chou and Hong (2000) consider a product mix planning problem in a wafer foundry factory, and design four types of time bucket sizes, which are 4, 2, 1 week, and daily, to identify the suitable granularity of such a problem. Their result show that using one week as the granularity is the most suitable one. Thus, in this paper, the planning horizon is set to be 4 weeks, and the planning period is set to be 1 week, i.e. the planning horizon comprises four planning periods.

2.1.1 A demonstrative example

Consider the following CAPPA example with three machines, M1, M2, and M3. Each machine possesses different process windows as shown in Table 1. Five orders are waiting for capacity allocation. The information of processing time (h), loading occurrence time (week), required process capability, and critical layer operation are shown in Table 2. The optimal solution is shown as in Fig. 5, where an underlined number represents a critical layer operation. The average load for the first week is 46.33 h, and the required load for M1, M2 and M3 is 48, 46, and 45 h, respectively. In the second week, the average load is 49.67 h, and the required load for M1, M2 and M3 is 50, 49, and 50 h, respectively. In the third week, the average load is 47.33 h, and the required load for M1, M2 and M3 is 47, 48, and 47 h, respectively.

Table 2 Processing time, loading occurrence time, required process capability and critical layer operation of orders in the CAPPA example

Order no.	Layer no						
	1	2	3	4	5	6	7
1	12,1,1,0[a]	15,1,3,1	19,2,2,0	12,2,3,1	14,3,2,0	–	–
2	11,1,1,0	16,1,3,1	18,2,2,0	11,2,3,1	9,3,2,0	15,3,3,1	17,3,1,0
3	13,1,2,0	15,1,3,0	10,2,4,1	13,2,2,0	20,3,4,1	12,3,3,0	–
4	12,1,2,0	14,2,4,1	14,2,2,0	13,2,4,1	12,3,3,0	15,3,2,0	–
5	13,1,2,0	13,1,3,0	19,1,4,1	12,2,3,0	13,2,4,1	12,3,4,1	16,3,2,0

[a] Processing time (h), load occurrence time (week), required process capability, whether a critical layer operation is included or not (1: critical layer operation; 0: non-critical layer operation), respectively

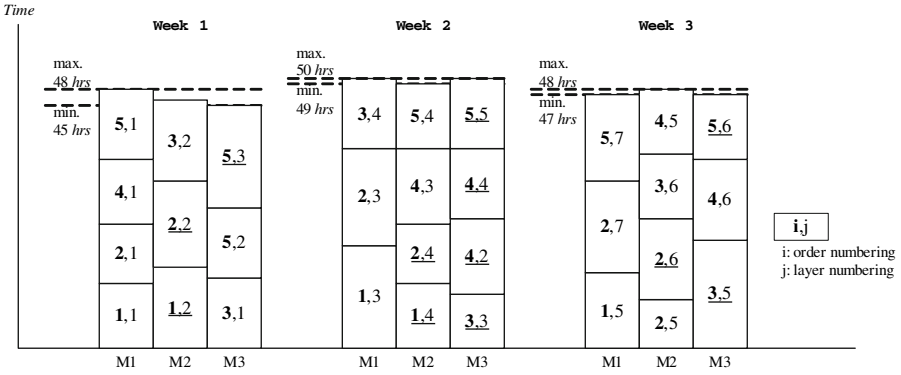


Fig. 5 The optimal solution for the CAPPA example

2.2 An mixed integer-linear programming formulation (MILP)

Notations and parameters

Indices

- i Index of order number, where $i = 1, \ldots, I$.
- l Index of layer number, where $l = 1, \ldots, L_i$.
- k Index of machine number, where $k = 1, \ldots, K$.
- h Index of processing capability number, where $h = 1, \ldots, H$.
- t Index of planning period, where $t = 1, \ldots, T$.

Parameters

- C_{hk} If machine k has processing capability h, then $C_{hk} = 1$; otherwise, $C_{hk} = 0$.
- AC_{kt} Available capacity of machine k in planning period t.
- CL_{il} If layer l of order i is a critical layer, then $CL_{il} = 1$; otherwise, $CL_{il} = 0$.
- CR_{ih} If the critical layer operations of order i require process capability h, then $CR_{ih} = 1$; otherwise, $CR_{ih} = 0$.

CR_{ilh} If layer l of order i has a load on processing capability h, then $CR_{ilh} = 1$; otherwise, $CR_{ilh} = 0$.
DC_{ht} Capacity requirement of process capability h in planning period $t (= \sum_i \sum_l (p_{il} CR_{ilh}\ LT_{ilt}))$.
DML_{it} Loading of critical layer operations of order i in period $t (= \sum_l (p_{il} CL_{il} LT_{ilt}))$.
L_i Number of photolithography operations for order i.
LT_{ilt} If layer l of order i has a load in planning period t, then $LT_{ilt} = 1$; otherwise, $LT_{ilt} = 0$.
p_{il} Processing time on layer l of order i.

Decision variables

dm_{ik} If the first critical layer of order i is assigned to machine k, then $dm_{ik} = 1$; otherwise, $dm_{ik} = 0$.
x_{ilk} If layer l of order i is assigned to machine k, then $x_{ilk} = 1$; otherwise, $x_{ilk} = 0$.
LS_{hkt} The loading level of process capability h assigned to machine k in planning period t.
ML_t The maximum loading level among machines in planning period t.

An MILP model (refer to as MILP_{OP} model) is constructed to solve the capacity allocation problem with constraints of process window and machine dedication as follows (Chung et al. 2006):

$$\text{Minimize} \quad \sum_t ML_t \tag{1}$$

Subject to

$$\sum_t \sum_k \sum_h \sum_l (x_{ilk} C_{hk} CR_{ilh} LT_{ilt}) = \sum_t \sum_h \sum_l (CR_{ilh} LT_{ilt}), \quad \text{for all } i \tag{2}$$

$$\sum_k x_{ilk} = 1, \quad \text{for all } i, l \tag{3}$$

$$\sum_t \sum_h \sum_l (x_{ilk} CL_{il} CR_{ilh} LT_{ilt}) = dm_{ik} \times \sum_t \sum_h \sum_l (CL_{il} CR_{ilh} LT_{ilt}), \quad \text{for all } i, k \tag{4}$$

$$\sum_i \sum_l \sum_h (x_{ilk} p_{il} C_{hk} CR_{ilh} LT_{ilt}) \le ML_t, \quad \text{for all } t \tag{5}$$

$$x_{ilk} \in \{0,1\}, \quad \text{for all } i, l, k \tag{6}$$

$$dm_{ik} \in \{0,1\}, \quad \text{for all } i, k \tag{7}$$

$$ML_t \ge 0, \quad \text{for all } t, k \tag{8}$$

The objective function (1) is to minimize the sum of ML_t, the maximum loading level among machines in planning period t. For a given CAPPA instance, a feasible solution with a smaller value of ML_t implies a smaller variance of machines' loading level in planning period t, while another one with a larger value of ML_t implies a larger variance of machines' loading level in planning period t. By minimizing the

sum of ML_t, the results will tend to balance the load among machines. Constraint (2) ensures that each layer of an order, including new release orders and WIP orders, must be assigned to a machine k if it has a capacity request in this planning horizon. In the machine assignment, the process window constraint must be considered. Constraint (3) is to make sure that each layer of an order can only be assigned to a single machine. Constraint (4) states the machine dedication control. Note that the orders in a planning horizon can either be orders planned to release or WIP orders that were released to the shop floor in the previous planning horizon. Therefore, dm_{ik} is a decision variable if the order is a planned-to-release order or a WIP order for which its first critical layer has not been assigned to a particular machine in the previous planning horizon; otherwise, dm_{ik} is a known parameter. Constraint (5) ensures that capacity loading of each machine in a period must be smaller than or equal to the maximum loading among machines in that planning period, ML_t.

3 Heuristic algorithms

3.1 Modified Sule's algorithms (MSAs)

Consider a variation of parallel machine scheduling problem in which n jobs must be scheduled on K parallel machines when some job j can be processed only on p machines ($p \leq K$). The objective is to assign jobs evenly among machines so that the numbers of operations processed on each machine are as evenly as possible. Hence, Sule (1997) proposes a heuristic method, called Sule's algorithm (SA), for solving such a scheduling problem with process capability restrictions. Following are the detailed procedures of SA, and an illustrative example is given in Table 3. The results are as shown in Table 4 and Fig. 6.

Step 1. Develop a job-machine relationship matrix, and calculate the job flexibility index (JFI) of each job and the machine flexibility index (MFI) of each machine, as shown in Table 3. JFI represents the number of machines that a job can be assigned without violating the process capability restriction, and MFI is the number of jobs that a machine can process without violating the process capability restriction.

Step 2. A candidate job is selected in the ascending order of JFI. If there is a tie, select the job with the smallest MFI of the corresponding suitable machines. Then, the candidate job is assigned to the machine for which the load balance could be achieved. The results are shown in Table 4.
In Table 3, job 1, 2, 4, 5 with the smallest JFI, i.e., 2, and the MFI of these jobs are 4, 4, 6, and 6, respectively. Thus, we will assign job 1 and 2 first. Here, we choose job 1 as the candidate and assign it to machine 1, the result is shown in row 3 of Table 4.

Step 3. Update the MFI for all machines according to the job assignment result. Repeat Step 2 until all jobs have been assigned.

The idea of Sule's algorithm is that a job with a smaller JFI implies having fewer suitable machines that can be processed. If a lower priority is given to the job, after higher priority jobs are assigned, the job may end up with only a few or even only

Table 3 Job-machine relationship matrix

Job no.	M1	M2	M3	M4	JFI
1	1[a]	1	0	0	2
2	1	1	0	0	2
3	1	1	1	0	3
4	0	1	1	0	2
5	0	1	1	0	2
6	0	1	1	1	3
7	0	1	1	1	3
8	1	1	1	1	4
MFI	4	8	6	3	21

[a] 1 means that the job can be processed on this machine; 0 means that the job can not be processed on this machine

Table 4 Result of job assignment by SA

Iteration no.	Job no.	Adjusted machine flexibility				Cumulative job assignment			
		M1	M2	M3	M4	M1	M2	M3	M4
0	–	4	8	6	3	0	0	0	0
1	1	**3**	7	6	3	**1**	0	0	0
2	2	2	**6**	6	3	1	**1**	0	0
3	4	2	5	**5**	3	1	1	**1**	0
4	5	2	**4**	4	3	1	**2**	1	0
5	6	2	3	3	**2**	1	2	1	**1**
6	7	2	2	**2**	1	1	2	**2**	1
7	3	**1**	1	1	1	**2**	2	2	1
8	8	0	0	0	**0**	2	2	2	**2**

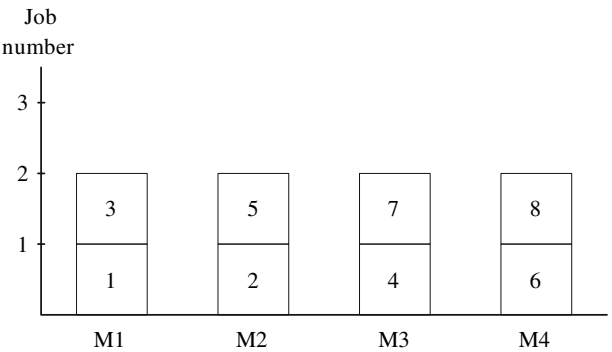


Fig. 6 Assignment solution by SA

one machine that can eventually be used. An unbalanced assignment of jobs among machines is resulted. Therefore, such a job should be given a higher priority in selecting a suitable machine in order to avoid load unbalance. In other words, a job with a larger JFI implies having more suitable machines, and there is a higher probability to achieve a balanced load among machines even if the job is assigned in the end.

Because the CAPPA not only has process capability restrictions but also faces multiple planning periods and machine dedication characteristics, it is obvious that the SA can not be directly applied to solve such a complicated problem, i.e., a modification of SA is required. Thus, we present six modified heuristics based on the SA by considering the characteristics of different loading levels among planning periods, process window and machine dedication, for solving the CAPPA in an efficient way.

In a planning period, jobs can be categorized into three types according to their operations: (1a) having a critical layer operation which has been dedicated to a specific machine, (1b) having a critical layer operation which has not been dedicated to a specific machine, and (2) having a non-critical layer operation without machine dedication restriction. For jobs that are machine-dedicated (i.e., critical layer operations), the JFI are smaller than those of non machine-dedicated jobs (i.e., non-critical layer operations), and these jobs may have a huge impact on the scheduling performance because the assignment of machine-dedicated jobs will influence the assignment results of other planning periods. As a result, in the assignment decision of jobs, we must first consider jobs in (1a) and (1b). In addition, critical layer operations of jobs in (1a) have been dedicated to specific machines; therefore, the JFI must be revised to 1, which is relatively lower than those of jobs in (1b). In consequence, the priority for assigning these three types of jobs in descending order is (1a), (1b) and (2). Thus, for a planning period, we design two kinds of assignment criteria of jobs:

3.1.1 Assignment criterion

C1. Select a candidate job according to the JFI and MFI (same as the SA). Jobs with smaller JFI are assigned first. If two or more jobs have the same JFI, the job with the smaller MFI is assigned first.

C2. Select a candidate job according to the job category [job type (1a) takes precedence over job type (1b), and job type (1b) takes precedence over job type (2)], JFI, and MFI. For jobs which belong to the same category, the candidate job is determined according to the assignment criterion C1 (same as the SA).

In addition, consider the CAPPA problem with multiple planning period characteristic, and the ratios of loading type (1a), (1b), and (2) to the overall loading in different periods are not the same. If we simply start the assignment from planning period 1 or in an arbitrary order of different planning periods, then we will lack an overall view to solve the CAPPA, and the solution quality may not be good. In other words, the objective of load balance among machines in every planning period is not achievable. Therefore, we design the following three types of period selection strategies; that is, the strategies for setting the priority order of planning periods, in an attempt to increase the solving quality of the proposed heuristic methods.

3.1.2 Period selection strategy

S1. Calculate the ratios of the load of job category (1a), (1b), and (2) to capacity supply ($\sum_k AC_{kt}$) for each planning period, and then rank the ratios in descending order. Assign a higher priority to the planning period with a higher ratio.

Table 5 Assignment criterion and period selection strategy for MSA-1 to MSA-6

Methods	Assignment criterion		Period selection strategy		
	C1	C2	S1	S2	S3
MSA-1	*		*		
MSA-2	*			*	
MSA-3	*				*
MSA-4		*	*		
MSA-5		*		*	
MSA-6		*			*

S2. Calculate the ratios of the load of job categories (1b) to the load of job categories (1b) and (2) for each planning period, and then rank the ratios in descending order. Assign a higher priority to the planning period with a higher ratio.

S3. Calculate the ratios of the load of job categories (1a) and (1b) to the load of job categories (1a), (1b) and (2) for each planning period, and then rank the ratios in descending order. Assign a higher priority to the planning period with a higher ratio.

The detailed calculations of loading type (1a), (1b), and (2) are given by Eqs. (9), (10) and (11), respectively.

$$\text{Loading of type (1a) in planning period } t\text{:}$$
$$\sum_i \sum_l \sum_k (p_{il} CL_{il} LT_{ilt} dm_{ik}) \tag{9}$$
$$\text{Loading of type (1b) in planning period } t\text{:}$$
$$\sum_i \sum_l \sum_k [p_{il} CL_{il} LT_{ilt} (1 - dm_{ik})] \tag{10}$$
$$\text{Loading of type (2) in planning period } t\text{:}$$
$$\sum_i \sum_l [p_{il} (1\text{-}CL_{il}) LT_{ilt}] \tag{11}$$

With the combination of two assignment criteria and three period selection strategies, we generate six heuristic methods, namely, MSA-1, MSA-2, MSA-3, MSA-4, MSA-5 and MSA-6, as shown in Table 5. Note that the process window and machine dedication restrictions must hold in the job assignment step under each heuristic algorithm.

3.2 Linear-programming-based heuristic algorithm (LPBHA)

The idea of the linear-programming-based heuristic algorithm (LPBHA), in a systematic view, is to avoid allocating a higher loading of process capability h on machine k than its ideal value and, instead, allocating a lower loading of process capability h' on machine k than its ideal value, due to the fact that different machines might not have the same process capabilities. LPBHA employs an MILP model (refer to as MILP_H model, see Eqs. (12)–(19) for details) to obtain the ideal loading level of process

capability h on machine k in planning period t (LS_{hkt}), and then uses the LS_{hkt} value as a parameter for job assignment since the decision variables x_{ilk} are not obtained in the MILP_H model. We note that the decision making of critical layer operations of a wafer lot will impact the loading level on machines for several periods. Therefore, the machine dedication characteristic must be considered in the MILP_H model to obtain the ideal loading level of each process capability on each machine in each planning period.

Considering the characteristic of CAPPA in handling the capacity allocation for multiple periods, we may need to decide the sequencing of planning periods to start the job assignment due to the machine dedication restriction. However, since the MILP_H model has derived the ideal loading level of each process capability on each machine in each planning period and has assigned critical layer operations of each job to a dedicated machine, the sequencing of planning periods in the LPBHA thus can be selected in an arbitrary manner. The jobs, which belong to the same planning period, are firstly ranked according to their CL_{il} value in descending order; and in case of ties, the tied jobs are ranked according to their processing time in descending order. A candidate job is picked from the top of the list. For a candidate job with process capability h, if after it is assigned to machine k (without violating the process capability restriction), the cumulative loading of process capability h for machine k in planning period t is lower than or equal to the LS_{hkt} value, then assign the job to machine k directly. Otherwise, assign the candidate job to a machine with the smallest positive difference from the LS_{hkt} value. Such a job assignment is considered since the ideal loading level of each process capability on each machine in each planning period (LS_{hkt}) obtained from the MILP_H model did not consider the indivisibility of jobs' loading, i.e., the LS_{hkt} value may be a fraction. Figure 7 depicts a flowchart of LPBHA.

The MILP_H model is as follows:

$$\text{Minimize} \sum_t ML_t \tag{12}$$

Subject to

$$\sum_k (C_{hk} LS_{hkt}) \geq DC_{ht}, \quad \text{for all} \quad h, t \tag{13}$$

$$\sum_h (C_{hk} LS_{hkt}) \leq ML_t, \quad \text{for all} \quad k, t \tag{14}$$

$$\sum_i (dm_{ik} DML_{it} CR_{ih} C_{hk}) \leq (C_{hk} LS_{hkt}), \quad \text{for all } h, k, t \tag{15}$$

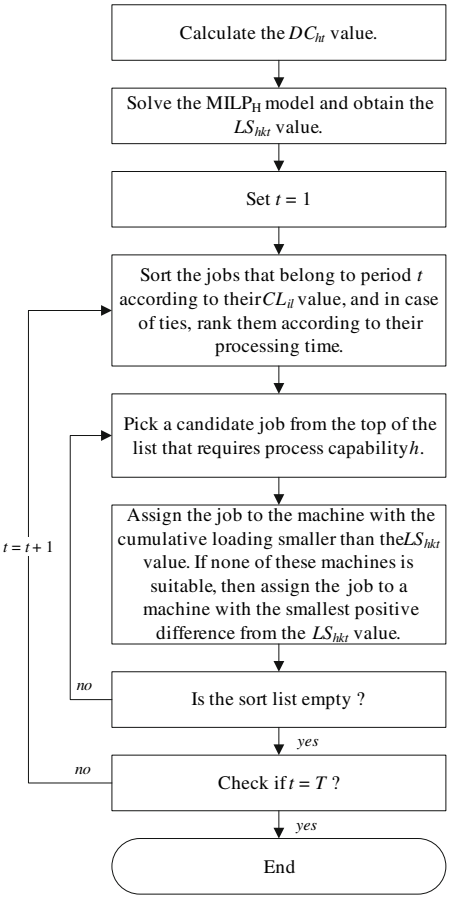
$$\sum_k dm_{ik} = 1 \quad \text{for all} \quad i, k \tag{16}$$

$$dm_{ik} \in \{0, 1\}, \quad \text{for all} \quad i, k \tag{17}$$

$$LS_{hkt} \geq 0, \quad \text{for all} \quad h, k, t \tag{18}$$

$$ML_t \geq 0, \quad \text{for all} \quad t \tag{19}$$

Fig. 7 Flowchart of LPBHA



where

$$DC_{ht} = \sum_i \sum_l (p_{il} CR_{ilh}\ LT_{ilt}), \quad \text{for each} \quad h, \text{each } t$$

The objective function of the MILP$_H$ model, represented by Eq. (12), is to balance the load among machines. Decision variable ML_t is the maximum loading among machines in planning period t. By minimizing the sum of ML_t over the periods, the result will tend to level the capacity loading among machines. Constraint (13) states the relationship between capacity supply of each machine and capacity demand of each process capability. It is to ensure that the total capacity supply of a specific process capability of all machines must be greater than the capacity demand of that process capability. Constraint (14) ensures that capacity loading of each machine in a period must be smaller than or equal to the maximum loading among machines in that planning period, ML_t. Constraint (15) ensures that the capacity supply of process capability h in machine k in planning period t must be greater than or equal to the sum

of capacity demand of process capability h for critical layer operations in machine k in planning period t. Constraint (16) ensures that critical layer operations of job i can only be assigned to one specific machine. Constraint (17) indicates that the dm_{ik} are binary variables. Constraint (18) and (19) indicate that the values of LS_{hkt} and ML_t respectively, are greater than or equal to zero.

4 Computational experiment

In this section, we design a comprehensive experiment of MSAs and LPBHA to evaluate its efficiency for solving large-scale CAPPA instances. The performances of the proposed algorithms are tested by real-world CAPPA cases taken from the wafer fabrication photolithography area. In addition, we show the performance comparisons between MILP_{OP} model and LPBHA, the most effective heuristic.

4.1 Test problem design

In order to analyze and compare the performance of MSAs and LPBHA on various capacity allocation problems, we generate 180 problems based on the data taken from the wafer fabrication photolithography area at the Science-Based Industrial Park in Taiwan. There are ten steppers with five different process capabilities. Five types of products: A, B, C, D, and E, are manufactured, and each product requires 17, 19, 16, 20, and 19 photolithography operations, respectively. The total photolithography operation time required for a product is in the range of 597–723 minutes. Product A and B require process technology of 0.17 μm, while Product C, D, and E adopt 0.14 μm process technology. Additionally, the design of these test problems highlight the key factors of the CAPPA, including product mix, loading level, and distribution of critical layer operations.

4.1.1 Product mix

Product mix has a major impact on the production system due to the different demand on process capabilities and different loading on critical layer operations. Ten kinds of product mix are considered here, as listed in Table 6.

4.1.2 Loading level

Under different system loading level situations, different numbers of orders need to be processed, and this has various impacts on the loading allocation among machines. Here, we consider three system loading levels: 95% (H), 85% (M) and 75% (L).

4.1.3 Distribution of critical layer operations

Critical layer operations require a stringent use of machines, that is, all critical layer operations of a job must be processed by the same machine. It is obvious that the number of critical layer operations has a major impact on the result of a CAPPA instance.

Table 6 Product mix level used in test problem

Product mix no.	Product mix level					Proportion				
	A	B	C	D	E	A	B	C	D	E
1	L[a]	L	L	H	H	2	3	3	6	6
2	L	L	H	H	L	3	3	6	5	3
3	L	H	L	H	L	3	7	1	6	3
4	L	L	H	L	H	4	3	5	2	6
5	L	H	L	L	H	4	6	2	2	6
6	L	H	H	L	L	4	6	5	3	2
7	H	L	L	H	L	6	3	3	5	3
8	H	H	L	L	L	6	7	2	2	3
9	H	L	L	L	H	7	3	3	2	5
10	H	L	H	L	L	7	3	5	3	2

[a] L means that the product has a low product mix level to other products; H means that the product has a high product mix level to other products

Table 7 Summarized information for the nine problem sets

Problem sets	Loading level	Dist. of critical layer	Number of orders	
			New release	WIP
1	H	1–3	328∼706[a]	194∼223
2		3–5		
3		5–7		
4	M	1–3	226∼548	161∼188
5		3–5		
6		5–7		
7	L	1–3	227∼389	140∼164
8		3–5		
9		5–7		

[a] Each order belongs to one of the five product types

We design three types of distributions of critical layer operations (the critical layer operations are present after every 1–3, 3–5 or 5–7 non-critical layer operations) and two instances for each type of distribution of critical layer operations.

With the factors considered, nine problem sets are designed, and each problem set has 20 sub-problems (two instances for one kind of distribution of critical layer operations and ten kinds of product mix for each instance). The difference among the nine problem sets is the loading level and distribution of critical layer activities. Summary for the nine problem sets is shown in Table 7.

4.2 Performance comparisons

The ILOG OPL 3.5 (2001) is used to implement the $MILP_H$ model. Microsoft Excel 2000 Visual Basic Application (VBA) is used to implement the MSAs and LPBHA because it is the most common development software used in companies and its

programming language is the same as the Visual Basic which is very easy to use. In addition, we adopt a Pentium IV 3.2 GH_Z PC as our test environment.

We note that the ML_t value obtained from the $MILP_H$ model is a lower bound for the ML_t value obtained from the $MILP_{OP}$ model, i.e., the optimal solution (ML_t^{OP}) will be greater than or equal to the lower bound (ML_t^{LB}). The reason is that the $MILP_{OP}$ model obtains decision variables x_{ilk} directly to determine the assignment of each layer of each order, while the $MILP_H$ model only concerns the assignment of critical layer of each order. Besides, the ML_t value obtained from the $MILP_{OP}$ model is computationally inefficient. Therefore, the deviation ratio between the maximum loading level among machines from a proposed algorithm and that from the $MILP_H$ model is employed to evaluate the performance of each algorithm (see Eq. 20).

$$\text{Deviation ratio} = \frac{\sum_t ML_t^H - \sum_t ML_t^{LB}}{\sum_t ML_t^{LB}} \times 100\% \tag{20}$$

where

ML_t^H The maximum loading level among machines in planning period t obtained from heuristic.

ML_t^{LB} The maximum loading level among machines in planning period t obtained from $MILP_H$ model.

Tables 8 and 9 show the average and the standard deviation of deviation ratio, respectively, among algorithms in problem sets 1–9. LPBHA has the best performance among algorithms with a least average and a least standard deviation of deviation ratio of 0.294 and 0.085%, respectively. The reason for its good performance is that ideal loading for each process capability of each machine is estimated first and is used as an assignment reference. In Table 10, we can see that the average computational time of LPBHA (i.e., 71.91 s) is longer than others. The reason is that the LPBHA needs extra time to solve the $MILP_H$ model in obtaining the ideal loading for each process capability of each machine. In fact, the computation time of the $MILP_H$ model used in LPBHA is in the range of 0.13– 9.98 s. Overall, the LPBHA shows a better performance than other myopic algorithms (MSAs) and can be used in large-scale CAPPA instances in a real-world situation (see Fig. 8).

From Table 11, the average deviation ratio of the assignment criterion C2 in problem sets 1–9 is calculated as 0.977%, compared to 1.364% of the assignment criterion C1. Table 12 shows that the assignment criterion C2 has a better performance with respect to the average deviation ratio than the assignment criterion C1. However, the discrepancy between the two assignment criteria decreases as the number of critical layer operations decreases. Additionally, Tables 11 and 12 show that both period selection strategies S2 and S3 outperform the period selection strategy S1 because they consider the loading profile of each planning period and select a planning period with the highest loading level of critical layer operations to start the job assignment.

Table 8 Average deviation ratio of MSAs and LPBHA algorithms in problem sets 1–9 (%)

Problem sets	Methods						
	MSA-1	MSA-2	MSA-3	MSA-4	MSA-5	MSA-6	LPBHA
1	1.879[a]	1.829	1.817	0.974	0.941	0.933	0.292
2	1.268	1.204	1.224	1.048	0.965	0.987	0.276
3	1.345	1.305	1.301	1.247	1.205	1.197	0.280
4	1.519	1.451	1.563	0.818	0.774	0.759	0.274
5	1.017	0.967	0.963	0.856	0.824	0.814	0.276
6	1.055	1.026	1.030	1.004	0.953	0.945	0.251
7	1.975	2.019	1.992	1.053	1.006	1.014	0.344
8	1.197	1.155	1.159	0.974	0.918	0.932	0.327
9	1.218	1.175	1.166	1.111	1.064	1.072	0.322
Avg.	1.386	1.348	1.357	1.009	0.961	0.962	0.294

[a] Average deviation ratios of 20 sub-problems

Table 9 Standard deviation of deviation ratio of MSAs and LPBHA algorithms in problem sets 1–9 (%)

Problem sets	Methods						
	MSA-1	MSA-2	MSA-3	MSA-4	MSA-5	MSA-6	LPBHA
1	1.739[a]	1.725	1.725	0.775	0.768	0.760	0.134
2	1.542	1.558	1.542	1.100	1.102	1.104	0.116
3	1.752	1.773	1.782	1.532	1.539	1.530	0.124
4	1.420	1.423	1.483	0.546	0.540	0.549	0.031
5	0.940	0.960	0.959	0.664	0.674	0.659	0.032
6	1.128	1.111	1.124	0.994	0.981	0.990	0.028
7	1.485	1.596	1.493	0.561	0.554	0.558	0.062
8	1.046	1.037	1.024	0.715	0.704	0.703	0.055
9	1.185	1.166	1.189	1.011	1.020	1.005	0.062
Avg.	1.395	1.413	1.408	0.914	0.913	0.911	0.085

[a] Standard deviation of deviation ratios of 20 sub-problems

Table 10 Computational time of MSAs and LPBHA algorithms in problem sets 1–9 (s.)

Problem sets	Methods						
	MSA-1	MSA-2	MSA-3	MSA-4	MSA-5	MSA-6	LPBHA
1	88.69[a]	88.93	89.61	70.63	70.59	71.03	85.12
2	83.97	84.04	83.91	73.88	72.24	74.28	86.64
3	84.14	83.75	83.46	75.40	76.24	77.61	90.67
4	75.07	74.85	75.69	58.98	59.14	59.60	72.93
5	70.57	70.77	70.64	61.49	60.44	61.96	73.08
6	70.58	70.46	70.16	63.04	63.77	64.06	76.09
7	52.07	52.05	52.84	41.75	41.68	42.02	53.29
8	49.42	49.49	49.26	43.39	42.66	44.27	53.54
9	49.38	49.43	49.11	44.56	45.11	45.43	55.84
Avg.	69.32	69.31	69.41	59.33	59.10	59.93	71.91

[a] Average computational time of 20 sub-problems

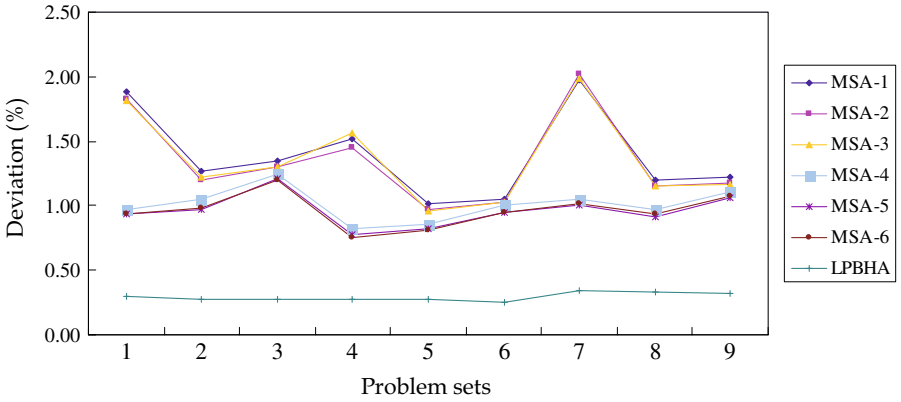


Fig. 8 Performance comparison among MSAs and LPBHA

Table 11 Average deviation ratio of assignment criteria and period selection strategies in problem sets 1–9 (%)

Problem sets	Assignment criterion		Period selection strategy		
	C1	C2	S1	S2	S3
1	1.842	0.949	1.048	1.021	1.014
2	1.232	1.000	0.864	0.815	0.829
3	1.317	1.217	0.957	0.930	0.926
4	1.511	0.784	0.871	0.833	0.865
5	0.982	0.831	0.716	0.689	0.685
6	1.037	0.967	0.770	0.743	0.742
7	1.995	1.025	1.124	1.123	1.117
8	1.170	0.942	0.833	0.800	0.806
9	1.186	1.082	0.884	0.854	0.853
Avg.	1.364	0.977	0.896	0.868	0.871

Table 12 Average deviation ratio of assignment criteria and period selection strategies in different types of distribution of critical layer operations (%)

Dist. of critical layer operations	Assignment criterion		Period selection strategy		
	C1	C2	S1	S2	S3
1-3	1.783	0.919	1.014	0.992	0.999
3-5	1.128	0.924	0.804	0.768	0.773
5-7	1.180	1.089	0.870	0.842	0.840
Avg.	1.364	0.977	0.896	0.868	0.871

4.3 Comparisons between $MILP_{OP}$ model and LPBHA

In this section, we compare the extent of achieving optimality of the LPBHA on five small-sized problems of the CAPPA, and the optimal solutions can be derived by $MILP_{OP}$ model that is described in Sect. 2.2. These problems, as shown in the

Table 13 Performance comparisons between $MILP_{OP}$ model and LPBHA

Problems	$MILP_{OP}$		LPBHA		
	(1) Objective function	Time (s.)	(2) Objective function	Time (s.)	Deviation (%)
M3J5[a]	146	0.28	146	0.06	0.00[b]
M3J10	291	71.56	297	0.12	2.06
M3J15	436	368.31	442	0.14	1.38
M4J5	114	17.25	119	0.06	4.39
M4J10	216	6, 757.39	226	0.11	4.63

[a] The notation denotes the number of machines and the number of jobs in the system. For example, M3J5 means that the system has three machines and five jobs. Note that the M3J5 is the CAPPA example described in Sect. 2.1

[b] The deviation is calculated as $[(2) - (1)]/(1) \times 100\%$

"problems" column in Table 13, are generated based on the system configuration of the CAPPA example described in Sect. 2.1. We did not consider the system configuration of M3J20, M4J15, M5J5, and M5J10 since M3J20 is infeasible, M4J15 and M5J15 can not be solved by $MILP_{OP}$ model within 86,400 s (i.e. one day long), and M5J5 is a too small problem case.

Table 13 shows that the $MILP_{OP}$ model generates optimal solutions in these test problems. LPBHA could generate a good feasible solution, due to the fact that the deviation ratio is in the range of 0.00–4.63%. Notice that the computational time of the $MILP_{OP}$ model reveals an exponential growth when the number of jobs and/or the number of machines increase, whereas that of the LPBHA is polynomial.

4.3.1 LPBHA for the CAPPA case

Table 14 is the summarization of the $MILP_{OP}$ model and LPBHA for solving the CAPPA case described in Chung et al. (2006). For the CAPPA case, the system configuration (the product type, the number of machines, the number of process capabilities) is the same as the test problem designed in Sect 4.1. The production planning and control department sets the planning horizon as 28 days, the planning period as 7 days, and the product mix ratio as 4:6:3:3:5 for Product A, B, C, D, and E, respectively. In the planning horizon, 474 lots are planned-to-release. Manufacturing execution system (MES) reveals that there are currently 204 lots of WIP on floor. In addition, the performance of both $MILP_{OP}$ model and LPBHA is evaluated with the objective function obtained from the $MILP_H$ model (a lower bound for the CAPPA) because of the optimal solution obtained from $MILP_{OP}$ model is computationally inefficient.

When LPBHA is applied to solve the CAPPA case, the deviation ratio is 0.14%, and the required solving time in such a large-scale CAPPA instance takes only 74.81 s. In contrast, the $MILP_{OP}$ model, by implementing depth-first search strategy incorporating strong branch rule, tends to obtain a feasible solution within a reasonable computational time, and, in this case, it results in a deviation ratio of 0.15% and a required computational time of 19,396.08 s.

Table 14 Comparisons between $MILP_{OP}$ model and LPBHA in the CAPPA case

$MILP_H$	$MILP_{OP}$			LPBHA		
(1) Objective function	(2) Objective function	Time (s.)	Deviation (%)	(3) Objective function	Time (s.)	Deviation (%)
24,246	24,283[a]	19,396.08[b]	0.15[c]	24,281	74.81	0.14[d]

[a,b] The $MILP_{OP}$ model used here did not give an optimal solution, it tends to obtain a feasbile solution within a reasonable computational time by implementing depth-first search strategy incorporating strong branch rule

[c] The deviation is calculated as $[(2) - (1)]/(1) \times 100\%$

[d] The deviation is calculated as $[(3) - (1)]/(1) \times 100\%$

5 Conclusion and future research

Wafer fabrication is both capital intensive and technology competent manufacturing. In this paper, we tackle the capacity allocation problem in CAPPA, which belongs to a CRP scheme, with the process window and machine dedication restrictions that arise from an advanced wafer fabrication technology environment. The process window constraint is caused by different process capabilities demanded on machines due to different existing fabrication technologies, while the machine dedication constraint is a control mechanism to improve the yield rate of wafer products. In order to effectively utilize the machine capacity to meet customer demand, we present six modified heuristics of Sule's algorithm (MSAs) and a linear programming based heuristic algorithm (LPBHA) to solve the problem efficiently. The performance comparison shows that LPBHA is the most robust one, due to the use of ideal loading level of each process capability in each machine in each planning period generated by $MILP_H$ model as a baseline for job assignment.

Acknowledgments This paper was supported in part by the National Science Council, Taiwan, R.O.C., for support under contract no. NSC94-2213-E-009-086. We thank the guest editor and the anonymous referees for their helpful comments, which improved the paper.

References

Akçalı E, Nemoto K, Uzsoy R (2001) Cycle-time improvements for photolithography process in semiconductor manufacturing. IEEE Trans Semicond Manuf 14(1):48–56

Akçalı E, Uzsoy R (2000) A sequential solution methodology for capacity allocation and lot scheduling problems for photolithography. In: 2000 IEEE/CPMT International Symposium on Electronics Manufacturing Technology, pp 374–381

Chen JC, Chen CW, Lin CJ, Rau H (2005) Capacity planning with capability for multiple semiconductor manufacturing fabs. Comput Indust Eng 48(4):709–732

Chou YC, Hong IH (2000) A methodology for product mix semiconductor foundry manufacturing. IEEE Trans Semicond Manuf 13(3):278–285

Chung SH, Huang CY, Lee AHI (2006) Capacity allocation model for photolithography workstation with the constraints of process window and machine dedication. Prod Plan Control 17(7):678–688

Chung SH, Huang HW (2001) Loading allocation algorithm with machine capability restriction for wafer fabrication factories. J Chin Inst Ind Eng 18(4):82–96

Hung YF, Cheng GJ (2002) Hybrid capacity modeling for alternative machine types in linear programming production planning. IIE Trans 34(2):157–165
ILOG Inc. (2001) ILOG OPL Studio 3.5. ILOG Inc., France
Leachman RC, Carmon TF (1992) On capacity modeling for production planning with alternative machine types. IIE Trans 24(4):62–72
Low CP, Fang C (2005) On the load-balanced demand points assignment problem in large-scale wireless LANs. Lect Notes Comput Sci 3391:21–30
Sule DR (1997) Industrial scheduling. PWS Publishing Company, Boston, pp 124–126
Toktay LB, Uzsoy R (1998) A capacity allocation problem with integer side constraints. Eur J Operat Res 109(1):170–182