Priority-Based Tool Capacity Allocation in the Foundry Fab

Chih-Yuan Yu*, Han-Pang Huang**

Robotics Laboratory, Department of Mechanical Engineering

National Taiwan University, Taipei, 10660, Taiwan.

TEL/FAX: (886) 2-23633875, E-mail: hphuang@w3.me.ntu.edu.tw

**Professor and correspondence addressee *Graduate student

Abstract

A virtual unit, tool group, is used to represent the group of tools in semiconductor foundry. Hence, a tool may belong to several tool groups because it may be handle several kinds of recipes. The coupling relation between the tools and the tool groups is a complex many-to-many relation. This paper aims to decide the tool-dispatching rule in tool capacity allocation. The term, "priority-based", is two-fold. One is to prioritize the lots in the fab, the other is to prioritize the tools in the correlated tool groups. The priority-based algorithm is used to estimate the tool utilization, bottleneck, tool-sharing in the tool groups, throughput of the tool groups and WIP (Work In Process) in a specific day. The proposed algorithm is applied to a real foundry fab and the results show that maximum fuzzy candidative ratio is better than other rules. Besides, a demonstration shows that the supervisor can find the bottleneck tools by the proposed algorithm and release the bottleneck by adding the right tools.

Key words: tool capacity allocation, neural network, cycle time estimation, priority

1. Introduction

The main purpose of this paper is focused to the optimization of daily tool capacity allocation. Once the tool capacity is available, the supervisors are eager to find the best configuration of on-time delivery and tool capacity allocations. The question arises from the coupling effects between tools and tool groups. Tools, which can handle the same recipe, will backup each other. The term, "tool group", is a group of backup tools. Usually, a tool group has more than one tool. Besides, a tool may process more than one recipe. It may belong to more than one tool group. Therefore, the coupling relation between the tools and the tool groups is a many-to-many relation. For convenient management, the route of a lot is not listed tool by tool. Instead, it is listed tool-group by tool-group. Any available tool in the tool group can serve the processing lots in this tool group. As a result, the supervisor wants to know the division of tool capacities for its tool groups. The tool capacity allocation is to decide the capacities of each tool for its tool group. In Fig. 1, the tools, "CETM03", "CMTM01", and "CETM02" are all enlisted in two tool groups, respectively. Then, the tool capacity allocation should optimize the tool utilization and on-time delivery of lots constrained with given tool capacities.

Wu et al. [9] proposed the Target Setting System (TSS) to calculate the daily target of each stage, a set of sequential operations. Chang et al. [1] proposed a formulation for optimizing long-term production and machine allocation for semiconductor manufacturing facilities. This paper shows a new approach to allocate the short-term capacities of tools for tool groups.
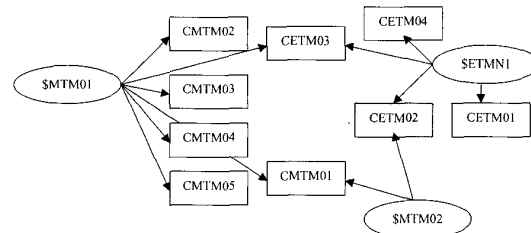

Fig. 1 The coupling effect between tools and tool groups

2. System Overview

2.1 On-time delivery performance in long-term planning

Assigning the release policy, allocating tools' capacities, and deciding a best lot-dispatching rule are the main decisions in the foundry fab. The order release plan depends on the due date of the lots and capacities of the fab and correlates with the WIP control level. A good order release plan can help to achieve the goals. However, the production control department gives the release policy. It is the first assumption in this paper. The dispatching rule adjusts the processing priorities for all lots in order to shorten the mean and variance of the cycle time and then fulfill the due-date [7]. In order to achieve on-time delivery, the least slack policy is usually used as a lot-dispatching policy [8] in the semiconductor foundry. Hence, the least slack policy is used as the lot-dispatching rule in this paper. This is the second assumption in the paper.

Suppose that the lot i arrives to the fab with a due date denoted as $\underline{f}\_(i)$. The estimated remaining cycle time for the lot i obtained by the estimator is denoted as ERCT(i), i.e., the slack s(i) of the lot i can be defined as

$$s(i) = \underline{f}\_(i) - t - ERCT(i) \qquad (1)$$

The quantity $\underline{f}\_(i) - t - ERCT(i)$ denotes the relative urgency of the lot i, therefore, the slack value, s(i), can tell how much time a lot has been ahead or behind its committed due date. If an idle tool has to decide which lot it should serve next, s(i) is reasonable to select the lot with the smallest s(i); i.e., the most urgent lot. As the degree of urgency is determined, on-time delivery performance can be improved by adjusting lots' priorities based on least slack policy.

The cycle time estimation [2, 3, 4, 5,] is the great topic in semiconductor industry. Yu et al. [11] shows three different algorithms to predict the product cycle time and lot remaining cycle time. The neural network approach shows good forecasting ability and short computation time.

Besides, it also provides the estimated cycle time of lots for least slack policy. After comparison, the on-line learning system [10] will be used to forecast the operation time and the estimated cycle time for lot in this paper.

2.2 Tool capacity allocation in short-term planning

The term "priority-based" is two-fold. One is to prioritize the lots in the fab by applying lot-dispatching rule, and the other is to prioritize the tools in the correlated tool groups by applying tool-dispatching rule.
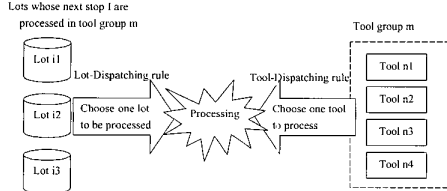

Fig. 2 The relation of lot-dispatching rule and tool-dispatching rule

In the first phase, each lot in the fab is assigned a priority class to identify its emergency and importance. The priority assignment of lots is based on least slack policy as mentioned in section 2.1. A lot with higher priority value is always selected first for processing. The priority classes used in this paper include Super Hot lot (SH), Hot lot (H), Rush lot (R), Normal lot (N), and Slow lot (S).

Given tools capacities, PM schedule, and release schedule, the capacity consumption of tools is assigned according to lots' priority classes. SH lots will be allocated first, then H lot and so on. The time horizon for the tool capacity allocation is one day. The estimated operation steps of a lot are based on the on-line learning system and will be further described in section 5.2. The first block in Fig. 3 shows the required time slot that the lot K completes operation 31. The time includes the waiting time and the processing time. The total flow in Fig. 3 denotes that if the tools of the fab are not finite capacity, then the lot can go through from operation 31 to 42 in the same day. Unfortunately, the capacity is finite.
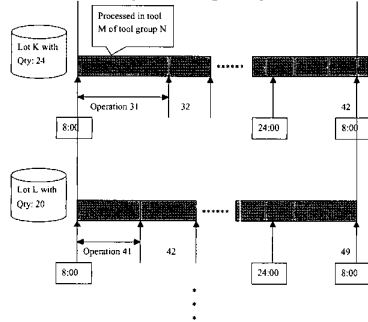

Fig. 3 The estimated steps of lot K, L in a specified day

However, the lot with the highest priority class can consume the capacities first. If the estimated throughput of the tool-group N in that day is 300 wafers, 24 (the number of wafers of lot K) is reserved for the lot K. If there is no more capacity in a tool group, the lot will be blocked from that tool group As a result, the lower the priority class the

lot has, the higher opportunity it is blocked due to the priority-based tool capacity allocation.

Which available tool in the tool group will be chosen to process the next lot is the second phase of prioritization. The tools in a specified tool group are assigned a priority value to indicate the service sequence of the tools. That is the tool-dispatching rule will be discussed in section 4. After reserving the tool capacity for lots, the blocked lots flag the bottleneck tools and tool groups. The shop floor manager can use this useful information to adjust the PM schedule or add tools to solve the bottleneck. After rearranging the tool capacity configuration, the manager should re-run the tool capacity allocation again until a balance among tool capacity, tool utilization and on-time delivery is achieved.

The symbols used are described below.

L(i)   the lot i, $L(i) \in L$, L is the set of all lots.
NL(i)   the quantities of the lot i.
s(i)   the slack value of the lot i.
W(i)   the waiting time of the lot i.
P(i)   the processing time of the lot i.
CT(i)   the cycle time of the lot i.
ERCT(i)   the estimated remaining cycle time of the lot i.
S(i)   the step that the lot i will go through in one day.
T(n)   the tool n, $T(n) \in T$, T is the set of all tools.
TG(m)   the tool group m, $TG(m) \in TG$, TG is the set of all tool groups.
NT(n)   the number of tool groups that the tool n is enlisted.
NTG(m)   the number of tools in the tool group m.
TGT(n)   the set of the tool groups that tool n is enlisted.
TTG(m)   the set of the tools in tool group m.
CT(n)(t) the capacities of the tool n at day t.
$e_{T(n)}(t)$ the utilization of the tool n at day t.
$UC_{T(n)}(t)$ the upper bound of the capacities of the tool n at day t.
$EC_{T(n)}(t)$ the estimated capacities of the tool n at day t.
$RC_{T(n)}$ the remaining capacities of the tool n.
$N_B(m)$ the number of blocked lots in the tool group m.
BR(n)   the bottleneck responsibility of the tool n.
$Move_{T(n)}$ the moves of the tool n.
$Move_{TG(m)}$ the moves of the tool group m.

3. Tool-Dispatching Rules

The coupling effect arises from the many-to-many relation between tools and tool groups. Due to the coupling effect, it is difficult to allocate the capacity of a tool to its tool groups. When a lot will consume the capacity of a tool group, the system should decide which tool in the tool group can serve the lot, i.e., the tool-dispatching rule in the tool groups. There are six types of tool-dispatching rules considered in this paper.

3.1 Maximum Remaining Capacity (MRC)

MRC means the tool with the maximum remaining capacity is chosen to serve the lot in the tool group m and can be expressed by

$$T^* = \arg\max_{T} \{RC_{T(n)}\} \quad \forall T(n) \in T_{TG(m)}, \quad \forall RC_{T(n)} > 0 \quad (2)$$

3.2 Maximum Remaining Capacity Ratio (MRCR)

Remaining Capacity Ratio of tool n at time t ($RCR_{T(n)}(t)$) is defined as $RC_{T(n)}(t)/ EC_{T(n)}(t)$ and $RC_{T(n)}(t)$ ranges from 0 to 1. MRCR can be expressed by

$$T^* = \arg\max_{T}\{RCR_{T(n)}\} \quad \forall T(n) \in T_{TG(m)}, \quad \forall RC_{T(n)} > 0 \quad (3)$$

3.3 Maximum Dedicative Ratio (MDR)

Dedicative Ratio of tool n at time t ($DR_{T(n)}$) is defined as $1/N_{T(n)}$. The less number of the tool groups a tool is enlisted (larger $DR_{T(n)}$), the more capacity the tool can contribute to each tool group. MDR can be expressed by

$$T^* = \arg\max_{T}\{DR_{T(n)}\} \quad \forall T(n) \in T_{TG(m)}, \quad \forall RC_{T(n)} > 0 \quad (4)$$

3.4 Any Available Capacity (AAC)

AAC can be expressed by

$$T^* = Random(T(n)) \quad \forall T(n) \in T_{TG(m)}, \quad \forall RC_{T(n)} > 0 \quad (5)$$

3.5 Sequence Available Capacity (SAC)

SAC can be expressed by

$$T^* = Seqence(T(n)) \quad \forall T(n) \in T_{TG(m)}, \quad \forall RC_{T(n)} > 0 \quad (6)$$

3.6 Maximum Fuzzy Candidative Ratio (MFCR)

In the first three approaches, MRC, MRCR, MDR are all the attributes to choose the candidate tool. As a result, the combined approach is usually used to include all these effects. Then the following relation can be found,

$$T^* = \arg\max_{T}\{\alpha \cdot RC_{T(n)} + \beta \cdot RCR_{T(n)} + \gamma \cdot DR_{T(n)}\} \quad (7)$$

$$\forall T(n) \in T_{TG(m)}, \quad \forall RC_{T(n)} > 0, \quad \alpha + \beta + \gamma = 1$$

The settings of $\pounds\backslash\pounds\,],\pounds^\wedge$ are usually subjective or by trial and error. However, fuzzy inference is suitable for consider multiple attributes with reasoning.

In the fuzzy system, there are two inputs and one output. Inputs are only chosen as $RC_{T(n)}$ and $DR_{T(n)}$ because $RC_{T(n)}$ and $RCR_{T(n)}$ are similar. The output is the fuzzy candidative ratio (FCR). MFCR means the tool with the maximum fuzzy candidative ratio is chosen to serve the lot in the tool group m and can be expressed by

$$T^* = \arg\max_{T}\{FCR_{T(n)}\} \quad \forall T(n) \in T_{TG(m)}, \quad \forall RC_{T(n)} > 0 \quad (8)$$

The fuzzy inference system contains the following properties.

- Linguistic variables {remaining capacity (RC), dedicative ratio (DR), candidative ratio (CR)}.
- $U(RC) = [0, \;], U(DR) = [0,1], U(CR) = [0,1]$
- Membership functions of RC, DR, CR are shown in Fig. 4, 5, 6, respectively
- Fuzzy rules are shown in Table. 1.
- The Mamdani fuzzy inference is adopted and uses product and max for T-norm and T-connorm operators, respectively, and Centriod of area (COA) is used as the defuzzification strategy.

The detailed descriptions of all procedures are described in next section.

Table 1 Fuzzy Rules

Dedicative Ratio	Remaining capacity			
	Too Small	Small	Large	Too Large
Too Low	Worst	Worse	Worse	Good
Low	Worst	Worse	Good	Good
Medium	Worse	Bad	Good	Better
High	Bad	Better	Better	Best

4. Priority-Based Tool Capacity Allocation

This paper discusses tool capacity allocation in short-term planning, while the on-time delivery in long-term planning follows the description given in section 2.1. This paper makes the following assumptions. (1) Release schedule and PM schedule are inputs of the system. (2) The lot-dispatching rule is least slack policy. (3) Step is adopted as the basic unit for describing the process flow. (4) All lots are assigned a priority class according to the least slack policy. (5) The proportion of five priority classes is fixed.

4.1 Procedures

Priority-based tool capacity allocation is a decision support system for the shop floor management. It runs daily in the beginning of a day in order to forecast the bottleneck of tool groups and tools in advance. The procedures of priority-based tool capacity allocation can be listed as follows.

(1) Keep running on-line learning system, (2) Obtain the PM (Preventive Maintenance) and release schedules, (3) Obtain the WIP information, (4) Adjust the priorities of lots, (5) Obtain the capacities of the tools, (6) Allocate the capacities of the tools, (7) Find the bottleneck tools and tool groups, (8) Take action to solve the bottleneck tools and tool groups and repeat procedures (1) to (8).

4.2 Running on-line learning system

Readers can reference [10] if interested in the on-line learning system.

4.2.1. Support for the tool capacity allocation

When the server program of on-line learning system keeps running, the information of the number of steps that the lot will go through in a day can be obtained by

$$S(i) = \left\{\arg\min_{s}\left\{\sum_{i=k+1}^{s}(W_i + P_i) > 1\,day\right\}\right\} - k \quad (9)$$

where k is the current step, and s is the step that the lot i will arrive one day later.

4.2.2. Support for least slack policy

The ERCT of the lot i can be obtained by

$$ERCT(i) = \sum_{i=m+1}^{N}(W_i + P_i) \quad (10)$$

where m is the current step and N is the total steps of the lot i.

4.3 Obtaining the PM (Preventive Maintenance) and the release schedules

The PM and release schedules in this paper are obtained from the automation department.

4.4 Obtaining the WIP information

In this paper, the WIP information is just the snapshot of the lots in the fab. Therefore, the WIP information is retrieved from the shop floor. It provides the information of lots' current location, current step, quantities, and priority class, etc.

4.5 Adjusting the priorities of lots

The least slack policy is used to prioritize all lots daily in the fab before executing the tool capacity allocation procedure. The proportion of each priority class

is fixed as 5% lots for SH, 15% for H, 30% for R, 45% for N, and 5% for S. The determination of the proportion is based on the research in [2] and the engineers' experiences from the real fab. The estimated remaining cycle time and the slack value of wach lot can be calculated from Eq. (10) and Eq. (1), respectively. If the slack $s(i) > 0$, it means that the lot commits its due date on time; otherwise, the lot will be delayed. The smaller the slack value of the lot has, the higher the priority class of the lot will be re-assigned.

The steps to re-assign the priority class of each lot are listed below.

Step 1: Calculate the slack values of all lots in the fab,

Step 2: Highlight some important lots by the managers,

Step 3: Sort the lots in terms of the larger slack value and importance,

Step 4: Reassign the priority class according to the proportion, (SH, H, R, N, S) = (5%, 15%, 30%, 45%, 5%).

4.6 Obtaining the capacities of the tools

The capacity of a tool is number of wafers that the tool can process in a day regardless the recipes. In general, the capacity of a tool depends on the PM schedule of the tool, the frequency of the setup change, recipes and the idle time of the tool. It is difficult to estimate the actual capacity of the tool due to the unpredicted interrupt and the idle time of the tool. The method for estimating the capacities of the tools in this paper is the dynamic moving average of the past seven days. Let $AC_{T(n)}(t)$ be the actual capacity of the tool n at day t, and $e_{T(n)}(t)$ be the utilization of the tool n at day t. The upper bound of the capacity of the tool n at day t, $UC_{T(n)}(t)$, can be obtained by

$$UC_{T(n)}(t) = \frac{AC_{T(n)}(t)}{e_{T(n)}(t)}, \qquad \forall T(n) \in T \qquad (11)$$

then the estimated capacity, $EC_{T(n)}(t)$, can be obtained by

$$EC_{T(n)}(t) = \frac{\sum_{i=1}^{7} UC_{T(n)}(t-i)}{7} \cdot \Omega_{T(n)}, \qquad \forall T(n) \in T \qquad (12)$$

$$\Omega_{T(n)} = \frac{24 hours - PM \ time \ period}{24 hours}$$

where $\Omega_{T(n)}$ is the portion of available time for processing.

4.7 Allocating the capacities of the tools

The main purpose of this section is to decide the amount of the tool capacities for each tool group. For example, the upper bound capacity of the tool n is 300 wafers and the tool n is enlisted in two tool groups. After running the capacity allocation module, the system will reserve 100-wafer processing capacity for the tool group 1, 150-wafer capacity for the tool group 2, and 50-wafer capacity is not used.

The capacity allocation algorithm is given below.

Step 1: Load all necessary data

The WIP information, the release schedule, the capacities of the tools, the mapping table of the tools and the tool groups are loaded into the system.

Step 2: Sort the lots by their priority class

Apply the method mentioned in section 4.5.

For i = 1 to L (L: the number of all lots in the fab plus new released lots)

Step 3: Calculate the steps that lot i will go through in the day

Applying eq (9).

For k = 1 to $S(i)$

Step 4: Find the next tool group (NTG)

Looking up the route of lot i. Let NTG = TG(m).

Step 5: Find the candidate tool in the tool group

Applying one tool-dispatching rule to find the candidate tool.

If the remaining capacity of T^*, $RC_{T^*} >$ the number of the lot i, $N_L(i)$, then reserve the capacity for the lot i and $RC_{T^*} = RC_{T^*} - N_L(i)$

else the lot is blocked in this tool group due to lack of the capacity, Exit the following steps of allocating

End *for-loop*

End *for-loop*

End of the procedure

The results of this procedure are the WIP information for one day later, reserved capacity of the tool for each tool group, the bottleneck tools, and the bottleneck tool groups.

4.8 Finding the bottleneck tools and tool groups

The bottleneck tools and bottleneck tool groups can be easily found from the capacity allocation algorithm. Let $N_B(m)$ = the number of blocked lots in the tool group m. It tells that $N_B(m)$ lots are blocked and can not continue their routes due to the lack of capacity of the tool group m. If $N_B(m)$ is too large, this tool group is called a bottleneck tool group.

A bottleneck tool group results in the blockage of all the tools enlisted in this tool group. The bottleneck responsibility (BR) of the tool n is defined by

$$BR(n) = \sum_{k \in TG}^{N_{T(n)}} \frac{N_B(k)}{N_{TG(k)}}, \quad for \ TG \ enlisting \ T(n) \qquad (13)$$

where $N_B(k)$ is the number of blocked lots in the tool group k; $N_{T(n)}$ is the number of tool groups that the tool n is enlisted; $N_{TG(k)}$ is the number of tools that the tool group k enlists; $N_B(k)/N_{TG(k)}$ means the bottleneck responsibility of the tool m in the tool group k. Similarly, if $BR(n)$ is too large, the tool n is called the bottleneck tool.

4.9 Action and re-allocation

The shop floor manager should take action to solve the bottleneck issue based on the information provided above. The action depends on the extra available resources, the experience of the managers, the PM schedule of the tools and the on-time delivery performance. Although it is not easy to make decision, the manager can adjust the capacities of the tools with the aid of the above results. Which tool should be added or which tool's PM schedule can be changed in order to solve the bottleneck can rely upon the list of the bottleneck tools and tool groups.

After adjusting the capacities and the usage of the tools, re-run the procedures (5) to (7) in 4.1 until the balance between the tool utilization and bottleneck is achieved

5. Results

The data used in this paper is retrieved from a real fab. The on-line learning system uses the on-line data from the real fab in real time. The time horizon of the data set is from 1998/11/1 to 1998/12/30 and the testing was performed on 1998/12/31. The system is executed at 8:00 AM. After running the system, the followings can be obtained for the manager.

5.1 WIP information

The total WIP distribution of all tool groups is shown in Fig. 7. The estimated total WIP for one day later is 20313 wafers.

5.2 Tool group moves

If the tool group move of the tool group m is $Move_{TG}(m)$, it means that there are $Move_{TG}(m)$ wafers flowed through the tool group m, i.e. the throughput of the tool group. The tool group move is defined as when a lot completes an operation, the lot contributes moves to this tool group. The moves that added to this tool group are equal to the number of the wafer of the lot. For example, if a lot with 24 wafers has been processed by the tool group m, then 24 tool group moves are added to the tool group m. Thus, we have

$$Move_{TG}(m) = \sum_{i \in \tilde{L}(m)}^{N} N_L(i) \qquad (14)$$

where N is the number of tools in $\tilde{L}(m)$; $N_L(i)$ is the quantity of $L(i)$; $\tilde{L}(m)$ is the set of lots which are processed in the $TG(m)$; L is the set of all lots. Note that $\tilde{L}(m)$ is the subset of L. The estimated total tool group moves in "1998-12-31" are 54597 wafers as shown in Fig. 8. The move of the tool group 33 is as high as 2850 wafer moves.

5.3 Tool moves

Similar to the tool group moves, the estimated tool moves in the tool n, $Move_T(n)$, is the number of wafers that are processed in the tool n. We have

$$Move_T(n) = \sum_{i \in L(n)}^{N} N_L(i) \qquad (15)$$

where $\tilde{L}(n)$ is the set of lots which are processed in $T(n)$. Fig. 9 shows not only the allocated capacity but also the utilization of the tools. From Eq. (11), the utilization of the tool m is defined as

$$e_{T(n)}(t) = \frac{AC_{T(n)}(t)}{UC_{T(n)}(t)} \qquad (16)$$

where $AC_{T(n)}(t)$ is the allocated capacity of the tool n at day t.

5.4 Bottleneck analysis

Only the tool groups, which block the lots, are listed in Fig. 10. It can be found that the tool groups 3 and 7 are critical to the system. Over 10 lots are blocked in these two tool groups and they are called the bottleneck tool groups. In order to resolve the bottleneck tool groups 3 and 7, three and two new tools, for example, are added to the tool groups 3 and 7, respectively. The result shows that the bottleneck tool group 3 is no more the bottleneck tool group, while the number of the blocked lots of the tool

group 7 is reduced from 15 lots to 6. Note that it is only a reference for the manager. The bottleneck responsibility of tools is shown in Fig. 11. The manager can identify the high-utilization tools and decide whether the PM schedule of the tool can be adjusted or not.

5.5 Tool-dispatching rule analysis

The six tool-dispatching rules described in section 4 are compared. Number of blocked lots, number of blocking tool groups, moves and WIP are the performance index to evaluate the rules. Fig. 12 shows the blocked lots after one day later. Fig. 13 shows the number of tool groups that block the lots. Fig. 14 shows the total moves in 1998/12/31. Fig. 15 shows the WIP one day later. In the above figures, it is obvious that MFCR is better than others, especially in "Blocked Lots" and "Moves". As a result, MFCR is used as the tool-dispatching rule.

6. Conclusion

This paper proposes a new approach to allocate the capacities of the tools for their tool groups. Besides, the maximum fuzzy candidative ratio is proved better than other tool-dispatching rules. Besides, the priority-based algorithm is used to estimate the tool utilization, bottleneck, tool sharing in the tool groups, throughput of the tool groups and WIP in a specific day.

With the support of the on-line learning system, which provides the cycle time information of lots, the computation time is small for simulating one-day tool capacity allocation. In addition, the accurate estimation of the remaining lot cycle time assures the correctness of the tool capacity allocation procedure. Based on the approach proposed in this paper, the shop floor manager can solve the bottleneck tools and reschedule the PM schedule of the tools.

References

[1] S.C. Chang, L.H. Lee, L.S. Pang, W.Y. Chen, Y.C. Weng, "Iterative Capacity Allocation and Product Flow Estimation for Scheduling Semiconductor Fabrication," *1995 IEEE/CPMT Int'l Electronics Manufacturing Technology Symposium*, pp.508-512, 1995.

[2] C. C. Cheng, "A Study of Lot Priority Setting for Wafer Fabrication," *Master Thesis*, Institute of Industrial Engineering, Chung Yuan Christian University, 1998.

[3] S. H. Chung, H. W. Huang, "The Design of Constraint-Oriented Target Planning System for Wafer Fabrication Factories," *Joint Conference of the 5th International Conference on Automation Technology*, Taiwan, 1998.

[4] B. Ehteshami, R.G. Petrakian, P.M. Shabe, "Trade-off in Cycle Time Management: Hot Lots," *IEEE Transaction on Semiconductor Manufacturing*, vol.5, no.2, pp.101-106, 1992.

[5] J.Y. Juang, H.P. Huang, "Queueing Network Analysis for an IC Foundry," *Proceedings of the 2000 IEEE International Conference on Robotics and Automation San Francisco, CA,*

pp.3389-3394, April, 2000.

[6] C.H. Lu, D. Ramaswamy, P.R. Kumar, "Efficient Scheduling Policies to Reduce Mean and Variance of Cycle-Time in Semiconductor Manufacturing Plants," *IEEE Transaction on Semiconductor Manufacturing*, vol.7, no.3, pp.374-388, 1994.

[7] M.A. Quaddus, "A Generalized Model of Optimal Due-Date Assignment by Linear Programming," *Journal of Operation Research Society*, vol.38, no.4, pp.353-359, 1987.

[8] K. S. Tsakalis, "Hierarchical Modeling and Control for Re-entrant Semiconductor Fabrication Lines: A Mini-Fab Benchmark," *IEEE/SEMI Advanced Semiconductor*

Manufacturing Conference, pp.508-513, 1997.

[9] G.L. Wu, K. Wei, C.Y. Tsai, S.C. Chang, N.J. Wang, R.L. Tsai, H.P. Liu, "TSS: a daily production target setting system for fabs," *Semiconductor Manufacturing Technology Workshop*, pp.86-98, 1998.

[10] C.Y. Yu and H. P. Huang, "Fab Model Based on Distributed Neural Network," *National Conference on Automation Technology*, ChiaYi, pp.271-277, 1999.

[11] C.Y. Yu, H.P. Huang, J.Y. Juang, C.F Yeh, "Cycle Time Estimation for Highly Product-Mixed Semiconductor Foundry," *The First Cross Strait Workshop on manufacturing*, Taipei, 2000.
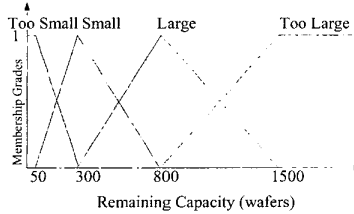
Fig.4 MF of Input 1 (remaining capacity)
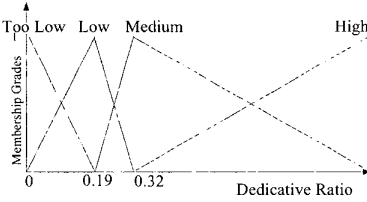
Fig.5 MF of Input 2 (dedicative ratio)
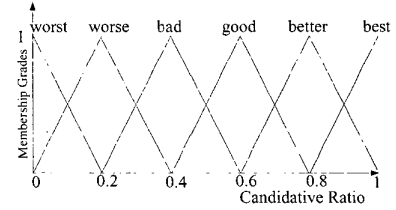
Fig.6 MF of Output (candidative ratio)
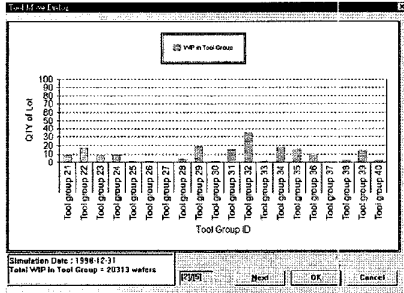
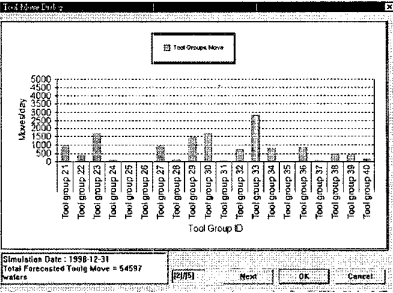Fig. 7 WIP in all tool groups

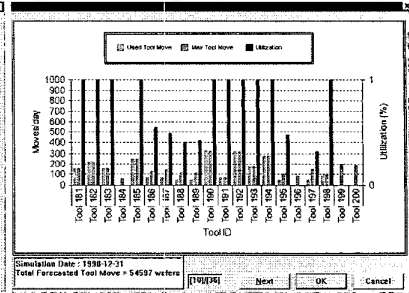

Fig. 8 Estimated tool group moves
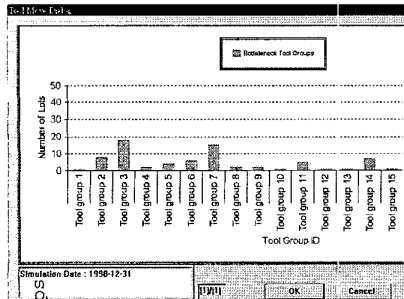

Fig. 9 Estimated tool moves
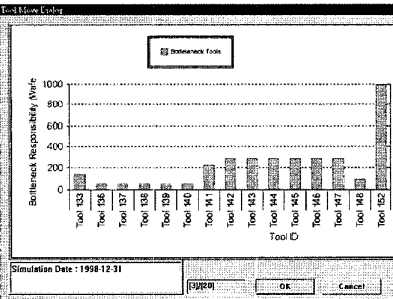

Fig. 10 Bottleneck tool groups
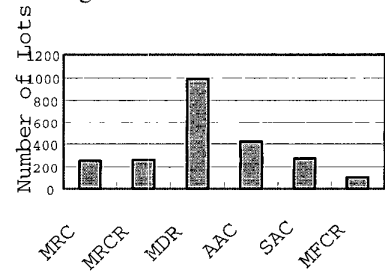

Fig. 11 BR of tools
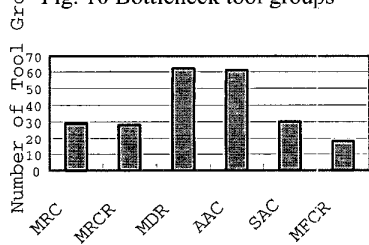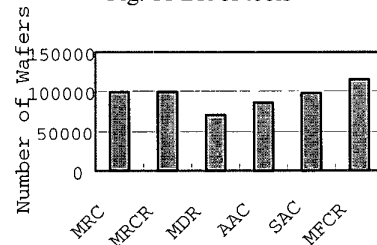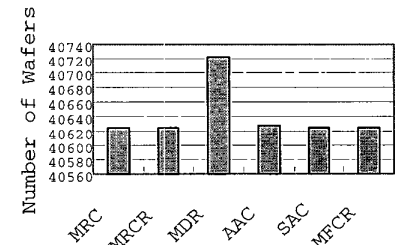

Fig. 12 Blocked Lots analysis

Fig. 13 Blocking Tool Groups analysis

Fig. 14 Moves analysis

Fig. 15 Blocked Lots analysis