

# A Queuing Network-based Method for Reconfiguration of Back-end Semiconductor Manufacturing Systems with Unreliable Equipment

Ying TANG and MengChu ZHOU

**Abstract**— Due to expensive, highly complex and time-consuming processes, semiconductor-manufacturing systems have been given a special attention. In our previous work (Tang *et al.*, 2003), a heuristic algorithm to design the reconfigurable automated production system was proposed. However, machine breakdown and planned/unplanned maintenance were not considered. This paper extends that work and addresses the related design issues in reconfigurable back-end semiconductor manufacturing systems considering failures and maintenance effects. Queuing network approaches are used to model the variance of production lines. The work enhances the proposed virtual production line (VPL) design methodology in (Tang *et al.*, 2003) where the deterministic machine processing time is used. A priority is also introduced to each idle machine and updated based on its past performance. The algorithms for Adaptive VPL configuration/reconfiguration are then proposed. The benefit of the proposed methodology and algorithms is illustrated through a simplified back-end semiconductor manufacturing example.

**Index Terms**—Reconfigurable Manufacturing Systems, Queuing Network Model, Heuristic Algorithm, Virtual Production Line

## 1. INTRODUCTION

A Reconfigurable Manufacturing System (RMS), identified as a priority technology by the National Research Council (National academy press, 1998), has become one of solutions to addressing manufacturing system flexibility, responsiveness, and integration issues. RMS includes reconfigurable machines and controllers, as well as methodologies for systematic design and diagnosis. Its production capacity can be rapidly adjusted to fluctuations in product demands, and functionality can be cost-effectively adapted to new products from a given product family (Koren *et al.*, 1998; Mehrabi *et al.*, 2000). Due to the mammoth needs of the semiconductor market, semiconductor-manufacturing companies have started to realize the opportunities for necessitating the use of RMS, which can be reconfigured and reprogrammed to provide manufacturers with a rapid response capability. From a production control point of view, the major part of flexibility of a manufacturing system is determined by its capability to make small batches of different part-mixes

and rapid changes from one part-mix to another, allowing the use of alternative machines, and accommodating dynamic machine selection among the alternatives during production (Tang and Qiu, 2004). The responsiveness of a manufacturing system is typically characterized by its product production cycle. The smaller the product production cycle time of a manufacturing system, the more responsive the manufacturing system is. The concept of a virtual manufacturing cell has been investigated allowing the set of workstations on the shop floor to be dynamically configured (Irani *et al.*, 1993; Simpson *et al.*, 1982), where cell formation according to machine capacity, processing time and product demands as well as its functionality is restrictedly undertaken. To address reconfiguration issues, the Virtual Production Line (VPL) concept was first introduced in (Qiu and Wysk, 1999). The fundamental issues for the discrete-event driven VPL design was presented in (Tang *et al.*, 2003). However, these methodologies were limited when applied to a system subject to machine failure and periodic maintenance. Actually, the variability due to such degraded behavior of manufacturing systems (e.g., unscheduled equipment downtime and maintenance) is a very important factor in effective production control (Tang and Zhou, 2001). In the recent academic literature, researchers use Petri nets (PN) to model such dynamics (Jeng and Xie, 2001; Kuo and Huang, 2000; Proth and Xie, 1994; Ramaswamy and Valavanis, 1994; Vinod and Altioik, 1986; Zhou and Jeng, 1998). Queuing theories are also adopted for this analysis and design (Chen *et al.*, 2000; Ciprut *et al.*, 1999; Tham *et al.*, 1999). Considering optimal production rate control in a failure prone manufacturing system, Hu and Xiang (1993) proved that under the hedging point policy the system is equivalent to an M/M/1 queue. Two Erlangian repairing models for a manufacturing system of identical unreliable machines were proposed by (Ching and Zhou, 1996). To solve the dynamic job shop scheduling problem, Tham *et al.* (1999) developed a minimum queuing delay dispatching algorithm that considers both the fairness of services to different jobs and the total queuing delays suffered by all the jobs. A number of maintenance dispatching policies were examined in (Mosley *et al.*, 1998) and the results indicated that the choice of maintenance scheduling policy significantly affects the system performance under restrictive staffing levels. Despite these relatively modest activities, it is strongly emphasized that the variability of VPLs is a decisive topic to investigate because of the complexity of the semiconductor manufacturing procedures. The primary goal of this paper is to present a method for the adaptive VPL design, which considers the performance of the failure prone machines

Manuscript received June 11, 2005; revised December 3, 2005 and March 4, 2006. This work is partially supported the National Outstanding Young Scientist Research Awards (Class B) from the National Natural Science Foundation of China under Grant No. 60228004. This paper is extended from "Design of Reconfigurable Semiconductor Manufacturing Systems with Maintenance and Failure" published at *IEEE International Conference on Robotic & Automation*, Seoul, Korea, May 21-26, 2001.

Y. Tang is with the Department of Electrical and Computer Engineering, Rowan University, Glassboro, NJ 08028, USA (email: tang@rowan.edu). M. Zhou is with the Department of Electrical and Computer Engineering, New Jersey Institute of Technology, Newark, NJ 07102, USA. (email: zhou@njit.edu).

and periodic maintenance. The rest of the paper is organized as follows. Section 2 considers a VPL with periodical maintenance; Section 3 analyses failure prone machines in a VPL; Section 4 focuses on adaptive algorithms for the VPL design; Section 5 gives an example, followed by our conclusion and future research directions in Section 6.

## 2. VPL WITH PERIODICAL MAINTENANCE

A VPL is organized as a sequence of workstations, each with one or more machines of the same type to handle a process in a stage (Tang *et al.*, 2003). When an order is placed, a VPL is configured based on the system status and the order information, and it is dedicated to making this product. Once the order is finished, its VPL can be retained for other orders of the same type; revised for a similar one; or dismissed so that the groups of equipment are reassigned to other VPLs. However, semiconductor-manufacturing equipment may be unreliable regardless of its age (Mosley *et al.*, 1998). The variability due to unscheduled equipment downtime is harmful to the system throughput and thus increases the production cost. Therefore, most of machines need to be maintained regularly for a period of time to achieve high availability. In this section, machines' periodic maintenance is considered first. The basic terminology and notation are introduced as follows:

### Basic terminology:

<b>System status</b>	The System status includes information of work-in-process (WIP) and machines' availability
<b>Order</b>	An order is a document providing authorization for a production to be performed. It specifies the type and size of the production. An order that has not been processed by a system is referred to a new order.
<b>Workstation</b>	A workstation is a segment of a VPL where a process is performed by one or more machines. Machines in a workstation are functionally identical, but may have different speeds.
<b>Desired speed</b>	The desired speed of a VPL is the minimal speed that guarantees the order processed on this VPL can be finished on time. It is described as the number of magazines per minute.
<b>Check time</b>	The check time is discrete-event time for the system to check certain VPLs in a system.
<b>Machine class</b>	Machine class is a set of machine instances, which have the same or similar properties.

### Notation:

$t$	The $t^{\text{th}}$ time buck (minute) from the start
$w$	VPL index.
$i$	Machine index.
$j$	Workstation index
$k$	Machine class index.
$s_w$	Number of workstations in the $w^{\text{th}}$ VPL.
$c_{ijw}$	Capacity of the $i^{\text{th}}$ machine within the $j^{\text{th}}$ workstation of the $w^{\text{th}}$ VPL
$\tau_{ijw}$	The $i^{\text{th}}$ machine's processing time of a magazine in the $j^{\text{th}}$ workstation of the $w^{\text{th}}$ VPL
$\beta_{ijw}$	Percentage of time that the $i^{\text{th}}$ machine is dedicated to the $j^{\text{th}}$ workstation of the $w^{\text{th}}$ VPL.
$v_{ijw}$	Speed of the $i^{\text{th}}$ machine in $j^{\text{th}}$ workstation of the $w^{\text{th}}$ VPL
$\epsilon_{jw}$	Number of machines in the $j^{\text{th}}$ workstation of the $w^{\text{th}}$ VPL
$p_{ik}$	Mean failure rate of the $i^{\text{th}}$ machine of the $k^{\text{th}}$ class
$P_{jw}$	Allowed maximum failure rate of machines in the $j^{\text{th}}$ workstation of the $w^{\text{th}}$ VPL without getting immediate repair
$\chi_{ijw}$	Maintenance/repair time for the $i^{\text{th}}$ machine in the $j^{\text{th}}$ workstation of the $w^{\text{th}}$ VPL
$r_{ijw}$	Repair rate of the $i^{\text{th}}$ machine in the $j^{\text{th}}$ workstation of the $w^{\text{th}}$ VPL
$\omega_{jw}$	Maintenance periodicity of machines in the $j^{\text{th}}$ workstation of the $w^{\text{th}}$ VPL
$t_{ijw}$	Maintenance rate of the $i^{\text{th}}$ machine in the $j^{\text{th}}$ workstation of the $w^{\text{th}}$ VPL
$\delta_{ik}$	Priority of the $i^{\text{th}}$ machine in the $M_k$
$N_w$	Size of an order (the number of magazines) processed in the $w^{\text{th}}$ VPL
$n_w(t)$	Number of magazines finished by the $w^{\text{th}}$ VPL at time $t$
$M_k$	Machine pool of the $k^{\text{th}}$ class
$R_k$	Repair machine pool of the $k^{\text{th}}$ class
$I_k$	Idle machine pool of the $k^{\text{th}}$ class
$\phi_k$	Number of idle machines of the $k^{\text{th}}$ class
$a_{jw}$	Average speed of the $j^{\text{th}}$ workstation in the $w^{\text{th}}$ VPL, which is measured as the number of magazine per minute
$a_w$	Desired speed of the $w^{\text{th}}$ VPL
$\underline{a}_w$	The minimal real-time speed of workstations in the $w^{\text{th}}$ VPL
$\zeta_w(t)$	Slack time of the $w^{\text{th}}$ VPL at time $t$
$\varphi_w(t)$	Tardiness of the $w^{\text{th}}$ VPL at time $t$

- $\psi_w(t)$  Earliness of the  $w^{th}$  VPL at time  $t$
- $E$  Set of VPLs with earliness ( $\psi_w(t) > 0$ )
- $D$  Set of VPLs with tardiness ( $\varphi_w(t) > 0$ )
- $ST(w)$  Start time of the  $w^{th}$  VPL
- $Due(w)$  Due time of an order processed in the  $w^{th}$  VPL
- $Date(w)$  Check time of the  $w^{th}$  VPL
- $Com(w)$  Completion time of an order processed in the  $w^{th}$  VPL
- $T_w$  Production time of an order processed in the  $w^{th}$  VPL (minutes)

In this paper, it is assumed that machines may have different maintenance time and the maintenance periodicity of workstations is independent. The speed and the mean maintenance rate of the  $i^{th}$  machine in the  $j^{th}$  workstation of the  $w^{th}$  VPL are calculated as:

$$v_{ijw} = \frac{c_{ijw}}{\tau_{ijw}} \quad (2.1)$$

$$t_{ijw} = \frac{\chi_{ijw}}{\omega_{jw}} \quad (2.2)$$

Then, the average speed of the  $j^{th}$  workstation is obtained.

$$a_{jw} = \sum_{i=1}^{\varepsilon_{jw}} v_{ijw} (1 - t_{ijw}) \beta_{ijw} \quad (2.3)$$

$\beta_{ijw}$  is initialized as 1. When a VPL is balanced, machines in that VPL are fully utilized. During the operation, workload unbalance may occur among workstations in a VPL. Certain workstations with higher frequencies will become bottlenecks, whereas machines in other workstations with low frequencies will often be idle. Thus,  $\beta_{ijw}$  changes with the machine's utilization. The machine utilization is defined as follows:

$$f = \frac{\text{Actual processing time of a machine}}{\text{Total production time}} \quad (2.4)$$

### 3. VPL WITH FAILURE AND MAINTENANCE

Although periodical maintenance is a useful mean to prevent machines from failure, some malfunctions and exceptions are still inevitable, which in turn cause line imbalance and decrease the speed of workstations. In this section, each machine with failure is modeled as a state machine shown in Figure 1. Note that the  $q^{th}$  workstation precedes the  $j^{th}$  one. Meanwhile, this paper assumes that machines may break down only when they are busy.

Let  $\pi_{ijw}^I$ ,  $\pi_{ijw}^B$  and  $\pi_{ijw}^F$  denote the probabilities of the idle, busy, and failure conditions of the  $i^{th}$  machine in the  $j^{th}$  workstation of the  $w^{th}$  VPL, respectively.

Then, the steady-state probabilities are obtained by solving the following linear equations (Zhou and Venkatesh, 1998):

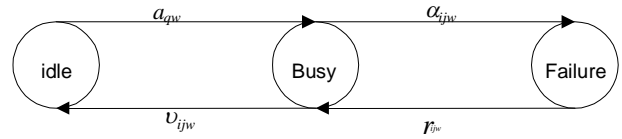


Fig. 1. State transition model of the  $i^{th}$  machine in the  $j^{th}$  workstation of the  $w^{th}$  VPL.

$$PID_{ijw} = 0 \quad (3.1)$$

$$\sum_{t \in \{I, B, F\}} \pi_{ijw}^t = 1 \quad (3.2)$$

where  $D_{ijw} = [(d_{yz})_{3 \times 3}]_{ijw}$  is the transition rate matrix.

$$D_{ijw} = \begin{pmatrix} -a_{qw} & a_{qw} & 0 \\ v_{ijw} & -(v_{ijw} + \alpha_{ijw}) & \alpha_{ijw} \\ 0 & r_{ijw} & -r_{ijw} \end{pmatrix}$$

Then  $\pi_{ijw}^I$ ,  $\pi_{ijw}^B$  and  $\pi_{ijw}^F$  are obtained by solving Equations. 3.1 and 3.2:

$$\pi_{ijw}^I = \frac{r_{ijw} v_{ijw}}{\Delta} \quad \pi_{ijw}^B = \frac{a_{qw} r_{ijw}}{\Delta} \quad \pi_{ijw}^F = \frac{a_{qw} \alpha_{ijw}}{\Delta}$$

where  $\Delta = r_{ijw} v_{ijw} + a_{qw} \alpha_{ijw} + a_{qw} r_{ijw}$  and  $\alpha_{ijw} = p_{hk}$  if the  $h^{th}$  machine in  $M_k$  serves as the  $i^{th}$  machine in the  $j^{th}$  workstation of the  $w^{th}$  VPL.

Due to periodic maintenance, the service speed of a machine is decreased from  $v_{ijw}$  to  $v_{ijw}(1 - t_{ijw})$  where  $t_{ijw}$  is the maintenance rate. Thus, considering the effects of both failure and periodic maintenance of machines, Equation 2.3 becomes:

$$a_{jw} = \sum_{i=1}^{\varepsilon_{jw}} v_{ijw} (1 - t_{ijw}) \beta_{ijw} \pi_{ijw}^B \quad (3.3)$$

### 4. ADAPTIVE RECONFIGURATION FOR VPL

From the above analysis, it is found that machine breakdown has adverse effects on system performance. Thus, to select a machine with higher speed and lower failure rate is very important in VPL design. For example, assume that machines 1 and 2 both can serve jobs in a certain stage  $j$ ,  $v_{1jw}$  is larger than  $v_{2jw}$ , but the failure rate of machine 1 is higher than that of machine 2. Then, the system may frequently spend resources to perform the operation associated with machine 1 without success, and then perform with machine 2 successfully. This will certainly increase the system's operational cost and decrease the throughput. In other words, if the system can select machine 2 prior to machine 1, the system throughput may be improved. Introducing priority for machines and integrating them into VPL design should enable the system to adapt for the best reconfiguration of VPLs.

Based on the past performance of machines, the values of  $\delta_{hk}$  assigned to the  $h^{th}$  machine in the  $M_k$  is decided as follows:  $\delta_{hk} = e$  if  $v_{ijw}(1 - p_{hk})$  is the  $e^{th}$  greatest value in  $\{v_{ijw}(1 - p_{hk}), h \in I_k\}$  (assuming the  $h^{th}$  machine in  $M_k$  is idle and may serve as the  $i^{th}$  machine in the  $j^{th}$  workstation of

the  $w^{\text{th}}$  VPL). It is clear that the priority of a machine may change depending on the job it serves.

When a new job comes or other events (e.g., breakdown) happen, the system will update the idle machine pools. If the mean failure rate of a machine is higher than a pre-set number  $P_{jw}$  (assuming this machine will serve jobs in the  $j^{\text{th}}$  workstation of the  $w^{\text{th}}$  VPL), remove this machine from its corresponding idle machine pool and add it into  $R_k$ , where machines receive immediate repairs.

Initially, all machines work in a good condition. During the operations, the mean failure rate of a machine is dynamically updated and assumed known in this paper. Thus, to configure a VPL for a new order, the system will sort machines in  $I_k$  with a decreasing order of their priority values. Machines will then be selected from  $I_k$  according to their priority values from the highest to the lowest. The algorithm that considers the variability due to machine breakdown and periodic maintenance is presented as follows.

#### Algorithm 1 (Adaptive reconfiguration):

(Assuming a new order is placed when  $u$  VPLs running in the system. And, the  $j^{\text{th}}$  workstation in a VPL requires the  $k^{\text{th}}$  class of machines, and  $\phi_k$  is the number of available machines in the  $k^{\text{th}}$  class.)

- (1) Set  $X = E$ , the set of VPLs with earliness.
- (2) Evaluate the information of the new order and calculate the desire speed of the  $(u+1)^{\text{th}}$  VPL:

$$a_{u+1} = \frac{N_{u+1} + s_{u+1} - 1}{Due(u+1) - ST(u+1)} \quad (4.1)$$

- (3) Update idle machine pools: if the failure rate of an idle machine in  $I_k$  is larger than  $P_{j(u+1)}$ , move it into its corresponding repair machine pool  $R_k$ ; assign priorities to the remaining machines in  $I_k$  and order them with a decreasing order of their priority values
- (4) Evaluate the system status and calculate the maximum speed of each workstation in the  $(u+1)^{\text{th}}$  VPL (assuming the arrival rate of jobs is larger than the speed of any machine in a VPL):

- i). Calculate the maximum speed of workstations with maintenance:

$$a_{j(u+1)} = \sum_{i=1}^{\phi_k} v_{ij(u+1)} (1 - t_{ij(u+1)}) \beta_{ij(u+1)} \quad (4.2)$$

- ii). Find the minimal speed of the  $(u+1)^{\text{th}}$  VPL only considering workstations with maintenance, that is:  $\underline{a}_{u+1} = \text{Min} \{a_{j(u+1)}, j \in [1, s_{(u+1)}]\}$  and the  $j^{\text{th}}$  workstation without failure}
- iii). Calculate the speeds of workstations with failures and maintenance. Note that  $\underline{a}_{u+1}$  is arrival rates of machines in such workstations.

$$a_{j(u+1)} = \sum_{i=1}^{\phi_k} v_{ij(u+1)} (1 - t_{ij(u+1)}) \beta_{ij(u+1)} \pi_{ij(u+1)}^B \quad (4.3)$$

- (5) Re-find the minimal speed of the  $(u+1)^{\text{th}}$  VPL according to  $\underline{a}_{u+1} = \text{Min} \{a_{j(u+1)}, j=1, 2, \dots, s_{(u+1)}\}$

- (6) If  $\underline{a}_{(u+1)} \geq a_{u+1}$ , keep this new VPL running with speed  $\underline{a}_{(u+1)}$ . Then, choose the minimum number of machines ( $\varepsilon_{j(u+1)}$ ) using Equation 3.3 such that  $a_{j(u+1)} \geq \underline{a}_{(u+1)}$  and they are the first  $\varepsilon_{j(u+1)}$  machines in  $I_k$ , then update  $\phi_k = \phi_k - \varepsilon_{j(u+1)}$ , and calculate  $\psi_{(u+1)}(t)$ . If  $\psi_{(u+1)}(t) > 0$ , add this new VPL into set  $E$  and go to Step (10)

- (7) If  $X \neq \Phi$ , do:

- a). Select the  $w^{\text{th}}$  VPL in the set  $X$ , remove it from set  $X$ , and adjust its  $\underline{a}_w$  as follows:

$$\text{Min } \varepsilon_{jw} \quad j = 1 \text{ to } s_w \quad (4.4)$$

Subject to:

$$\psi_w \geq 0 \quad (4.5)$$

$$a_w = \frac{N_w - n_w(t)}{Due(w) - Date(w)} \quad (4.6)$$

$$a_{jw} \geq a_w \quad j = 1 \text{ to } s_w \quad (4.7)$$

- b). If Step a) succeeds, update  $\varepsilon_{jw}$ . Exclude the redundant machines from this VPL and add them into idle machine pool  $I_k$ ;

- c). If  $\psi_w = 0$ , remove it from the set  $E$ ;

- d). Return to Step (2)

- (8) If there is at least one machine available at each stage

of the  $(u+1)^{\text{th}}$  order and  $\sum_{w=1}^u (\psi_w - \varphi_w) - \varphi_{u+1} > 0$ ,

configure this new VPL with a slow rate  $\underline{a}_{(u+1)}$ , put this new line into set  $D$  and go to Step (10)

- (9) Reject this new order, consider it later and exit.

- (10) Calculate the production time for the  $(u+1)^{\text{th}}$  order  $T_{u+1}$  and exit:

$$T_{u+1} = (N_{u+1} + s - 1) / \underline{a}_{(u+1)} \quad (4.8)$$

Degraded behaviors (e.g., tardiness, malfunctions and exceptions) of semiconductor manufacturing systems are not negligible in practice. Thus, how to reallocate resources to minimize the negative impact of these behaviors is extremely important. Taking this into consideration, an adjustment algorithm is presented based on our previous work (Tang *et al.*, 2003):

#### Algorithm 2 (Dynamical adjustment)

The system adjusts VPLs when tardiness, malfunctions or exceptions happen:

- (1). Repeat the following steps for all VPLs:

- a) Calculate  $\zeta_w(t)$ ,  $\varphi_w(t)$  and  $\psi_w(t)$ ;

- b) If  $\psi_w(t) > 0$ , update  $E = E \cup \{w\}$ ; otherwise if  $\varphi_w(t) > 0$ , update  $D = D \cup \{w\}$ ;

- (2). If  $D \neq \Phi$ , do:

- a) If  $E \neq \Phi$ , for each VPL in set  $E$ , Reallocate resources through Equations 4.4-4.7; and update  $\phi_k$  and  $\psi_w(t)$ . If  $\psi_w(t) = 0$ , remove it from set  $E$ ;
- b) Arrange VPLs in  $D$  with an ascending order of earliest due dates. For each VPL in  $D$ , do:
  - I) For  $j = 1$  to  $s_w$  (assuming the  $j^{th}$  workstation in the  $w^{th}$  VPL requires the  $k^{th}$  class of machines)
    - i) If  $\phi_k \neq 0$ , update  $I_k$  using the same way as Step 3 in Algorithm 1, then, assign the first  $\mathcal{G}_j$  ( $\mathcal{G}_j \leq \phi_k$ ) machines in  $I_k$  to the  $j^{th}$  workstation to increase its speed.  
for ( $i = \varepsilon_{jw}$  to  $\varepsilon_{jw} + \mathcal{G}_j$ ) do:  
if ( $\alpha_{ijw} = 0$ )  $a_{jw} = a_{jw} + v_{ijw}(1 - t_{ijw})\beta_{ijw}$   
else  $a_{jw} = a_{jw} + v_{ijw}(1 - t_{ijw})\beta_{ijw}^B$
    - ii) Otherwise, check other VPLs in the system (i.e.,  $x^{th}$  VPL and the  $v^{th}$  workstation in the  $x^{th}$  VPL requires the  $k^{th}$  class of machines). If there is machine  $i$  such that its utilization is less than 0.5, this machine can be shared by  $w^{th}$  VPL.  
for ( $x = 1$  to  $u$  ( $x \neq w$ ))  
for ( $i = 1$  to  $\varepsilon_{vx}$ )  
if ( $\alpha_{ivx} = 0$ )  
 $a_{jw} = a_{jw} + v_{ivx}(1 - t_{ivx})(1 - \beta_{ivx})$   
else  
 $a_{jw} = a_{jw} + v_{ivx}(1 - t_{ivx})(1 - \beta_{ivx})^B$
  - II) Calculate  $\underline{a}_w = \text{Min} \{a_{jw}, j = 1, 2, \dots, s_w\}$
  - III) If  $|\underline{a}_w - a_w| \leq 10^{-2}$ , remove this VPL from  $D$
  - IV) Otherwise, update its tardiness  $\phi_w$ . If  $\phi_w > 0$ , this order will be delayed  $\phi_w$  time to finish based on the present forecast.

**5. AN ILLUSTRATIVE EXAMPLE**

To better understand the above methods and algorithms, a practical implementation of the concept of VPLs is studied using a simplified back-end semiconductor line. In the exemplified back-end semiconductor manufacturing system, an order is sequentially made through some or all the following processes: Saw, 2/OP, Tape attach, Die attach, Plasma cleaning, Wire bond, 3/OP, Pre mold bake, Plasma cleaning, Glob topping, Mark, Plasma cleaning, Solder ball, Flux clean, Signulation, Final visual and Pack. A simplified process flow is given in Figure 2.

In the simulation run, three cases are analyzed and compared to illustrate the advantage of the proposed methodology.

- 1) In the baseline case, the system successively processes orders according to their order dates. In other words, the line is predefined to manufacture certain type of products at each time.
- 2) In the second case, whenever an order comes, the methodology presented in (Tang *et al.*, 2003) is

applied to strategically configure a VPL on the fly and monitor all VPLs in the system.

- 3) In the third one, the adaptive algorithms are used to configure and control VPLs.

For these three cases, this paper considers the following scenario: while the system is already running, two new orders are placed sequentially, both following the process flow in Figure 2. For the simplicity, it is assumed that except in ‘‘Saw’’ and ‘‘Wire bond’’ workstations, machines are reliable. Workstations are equipped with identical machines with different failure rates and machines are fully dedicated to its corresponding workstation. Before configuring a VPL for the first new order, there are ten machines unreliable in  $I_5$ , which can process jobs in ‘‘Wire bond’’ workstation. Their failure rates are  $\{0.004, 0.008, 0.01, 0.01, 0.01, 0.008, 0.008, 0.008, 0.04, 0.04\}$ . Based on field engineers’ experience,  $P_{jw}$  depends on  $\pi_{jw}$ . In this example, the threshold for machines in  $I_5$  is set as 0.004 (1/min), meaning that the machine with the failure rate higher than 0.004 gets immediate repair. On the fifth day of the first new order being processed, three machines in the ‘‘Saw’’ workstation break down. The next day after that, eighteen idle machines are added into  $I_5$  since an old order is finished. The input data for orders and workstations is shown in Tables 1 and 2, respectively.

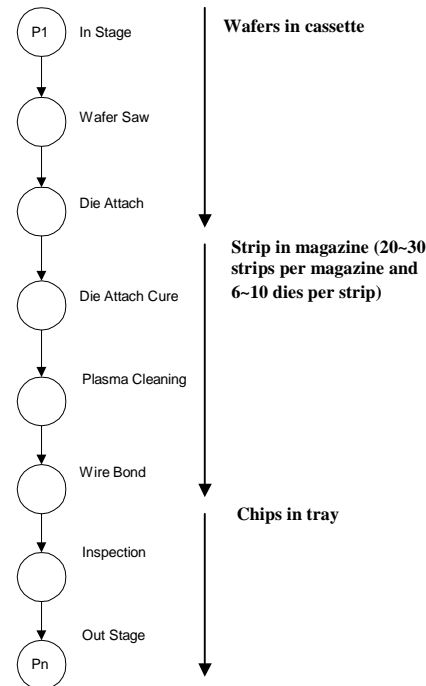


Fig.2. A simplified process flow.

Table 1. The input data for the two new orders

Order	EST	Due	$N_w$ (mag.)
1 <sup>st</sup> order	1/1/00	1/16/00	$10^5$
2 <sup>nd</sup> order	1/3/00	1/13/00	$10^4$

Table 2. The input data for workstations

	Saw	Die Attach	Die Attach Cure
$c_{ijw}$ (mag.)	1	1	24
$\tau_{ijw}$ (min.)	7	6	120
$\chi_{ijw}$ (hour)	1	1	1
$\varpi_{iw}$ (day)	7	7	7
$\phi_k$	50	50	50

To show the difference among these cases, the system throughput  $g$  is introduced, which is the total number of all completed magazines divided by the difference between maximum completion time and minimum start time of all the orders completed.

$$g = \frac{\sum_{w=1}^u N_w}{Max\{Com(w)\} - Min\{ST(w)\}} \quad (5.1)$$

Our computation results are then presented in Figure 3 and Table 3. The changes of VPL speeds, the system throughput and the utilization of “Wire bond” workstation are compared as the two orders go through the three cases. It is clear to see that in the last two cases, the system can concurrently handle multiple orders and adjust the speed of these VPLs according to production line condition and system status. Thus, the system’s throughput increases by 26.7%. In the third case, due to the priority introduced, the system can monitor machines’ performance and immediately repair machines with a high failure rate (i.e.,  $p_{ik} > 0.004$ ), instead of keeping such machines running until periodic maintenance. Thus, the second order was finished in advance by one day, and workstation utilization increases by 10.2% and 4.2% compared to those in the first and the second cases, respectively.

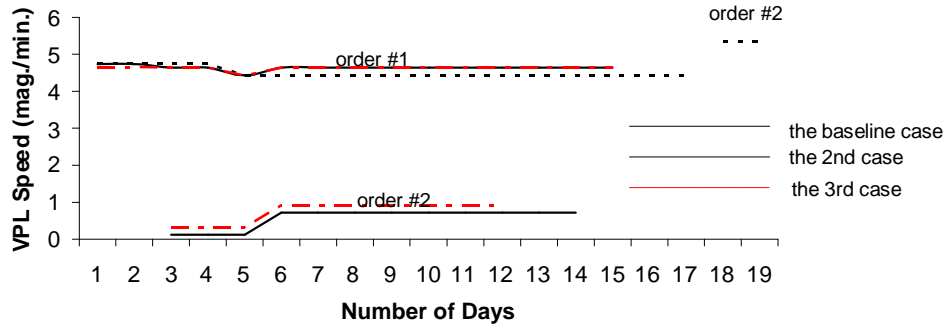


Fig. 3. VPL speeds for different cases when the system processes the two orders.

Table 3. The computational results

# of days	Baseline case		2 <sup>nd</sup> case (algorithms in (Tang <i>et al.</i> , 2003) )		3 <sup>rd</sup> case (adaptive algorithms)	
	1 <sup>st</sup> order	2 <sup>nd</sup> order	1 <sup>st</sup> order	2 <sup>nd</sup> order	1 <sup>st</sup> order	2 <sup>nd</sup> order
1	4.742		4.742		4.659	
2	4.742					
3	4.742					
4	4.742					
5	4.429					
6	4.429					
7	4.429					
8	4.429					
9	4.429					
10	4.429					
11	4.429					
12	4.429					
13	4.429					
14	4.429					
15	4.429					
16	4.429					
17	4.429					
18		5.358				
19		5.358				
$g$	5789		7333		7333	
$f$	90.2% (the 5 <sup>th</sup> workstation)		95.4% (the 5 <sup>th</sup> workstation)		99.4% (the 5 <sup>th</sup> workstation)	

## 6. CONCLUSION

In the competitive and risky environment of semiconductor manufacturing, production planning and control is quite complicated and difficult. In order to meet customers' needs and adapt to fluctuating market demands, the deployment of RMS has become critical in enhancing the system's flexibility and responsiveness. Moreover, degraded behaviors, such as failures and maintenance, of a semiconductor manufacturing system are not negligible in practice. This paper extends our previous work in reconfigurable back-end semiconductor manufacturing system design by considering the variability due to such degraded behaviors. A queuing network model is used to analyze a workstation's throughput due to machine downtime, unplanned and planned maintenance. In light of the adverse effects of unexpected machine breakdowns on system performance, a priority value is introduced to each idle machine, allowing the system to dynamically select the most reliable machines. The algorithms are then developed for adaptive VPL configuration and reconfiguration. From the example, the approach is found to be effective in increasing system throughput and machine utilization.

The research can be extended in several directions. For instance, more factory data is needed to further test the proposed methodology in the future. Different modeling methods for failure prone machines and maintenance scheduling in VPL operations to meet different productivity requirements are worthwhile investigating.

## REFERENCES

- [1] Chen, Y. J., Yu, G., Huang, K. S., and Wang, I., "A next queue algorithm in real time dispatching system of semiconductor manufacturing," *Proc. of the 9<sup>th</sup> International Symposium on Semiconductor Manufacturing*, Tokyo, Japan, Sept. 26-28 2000, pp. 67-70.
- [2] Ching W. K. and Zhou, X. Y., "Machine repairing models for manufacturing systems," *Proc. of IEEE Conf. on Emerging Technologies and Factory Automation*, Kauai, HI, Nov. 18-21, 1996 pp. 221-225.
- [3] Ciprut, P., Hongler, M. O. and Salama, Y., "On the variance of the production output of transfer lines," *IEEE Trans. on Robotics and Automation*, Vol. 15, No. 1, 1999, pp. 33-43.
- [4] Hu, J. Q. and Xiang, D., "The queuing equivalence to a manufacturing system with failures," *IEEE Trans. on Automatic Control*, Vol. 38, No. 3, March 1993, pp. 499-502.
- [5] Irani, S. A., Cavalier, T. M., and Cohen, P. H., "Virtual manufacturing cells: exploiting layout design and intercell flow for the machine sharing problem," *International Journal of Production Research*, Vol. 31, No. 4, 1993, pp. 791-810.
- [6] Jeng, M. D. and Xie, X. L., "Modeling and analysis of semiconductor manufacturing systems with degraded behavior using Petri Nets and Siphons," *IEEE Trans. on Robotics and Automation*, Vol. 17, No. 5, Oct. 2001, pp. 576-588.
- [7] Koren, Y., Hu, J.S., and Weber, T. W., "Impact of Manufacturing System Configuration on Performance," *Annals of the CIRP*, Vol. 47, pp. 369-372, 1998.
- [8] Kuo, C. H. and Huang, H. P., "Failure modeling and process monitoring for flexible manufacturing systems using colored timed Petri nets," *IEEE Trans. on Robotics and Automation*, Vol. 16, No. 3, June 2000, pp. 301-312.
- [9] Mehrabi, M.G., Ulsoy, A.G., and Koren, Y., "Reconfigurable manufacturing systems: key to future manufacturing," *J. of Intelligent Manufacturing*, 2000, Vol. 11, No. 4, pp. 403-419.
- [10] Mosley, S. A., Teyner, T., and Uzsoy, R. M., "Maintenance scheduling and staffing policies in a wafer fabrication facility," *IEEE Trans. on Semiconductor Manufacturing*, Vol. 11, No. 2, May 1998, pp. 316-323.
- [11] Proth, J. M. and Xie, X. L., "Cycle time of stochastic event graphs: evaluation and marking optimization," *IEEE Trans. On Automatic Control*, Vol. 39, No. 7, 1994, pp. 1482-1486.
- [12] Qiu, R. and Wusk, R., "Design and implementation of virtual production lines for discrete automated manufacturing systems," *The 14<sup>th</sup> Int. Federation of Automatic Control World Congress*, Beijing, P. R. China, July 5-9, 1999, pp. 455-460.
- [13] Ramaswamy, S. and Valavanis, K. P., "Modeling, analysis and simulation of failures in a materials handling system with extended petri nets," *IEEE Trans. on Systems, Man, and Cybernetics*, Vol. 24, No. 9, Sept. 1994, pp. 1358-1373.
- [14] Simpson, J. A., Hocken, R. J., and Albus, J. S., "The automated manufacturing research facility of the National Bureau of Standards," *Journal of Manufacturing Systems*, Vol. 1, No. 1, 1982, pp.17-32.
- [15] Tang, Y., Zhou, M. C. and Qiu, R., "Virtual production lines design for back-end semiconductor manufacturing systems," *IEEE Trans. on Semiconductor Manufacturing*, Vol. 16, No. 3, Aug. 2003, pp. 543-550.
- [16] Tang, Y. and Zhou, M. C., "Design of Reconfigurable Semiconductor Manufacturing Systems with Maintenance and Failure," *Proc. of 2001 IEEE Int. Conf. on Robotics and Automation*, Seoul, Korea, pp. 559-564, May 2001.
- [17] Tang, Y. and Qiu, R., "An Integrated Design Approach for Virtual-Production-Line-Based Reconfigurable Manufacturing Systems," Accepted by *International Journal of Production Research*, 2004.
- [18] Tham, C. K., Lou, R. A., and Ngin, H. T., "Minimum queuing delay: an adaptive algorithm for scheduling in a dynamic job shop," *Proc. of The 25<sup>th</sup> Annual Conference of the IEEE Industrial Electronics Society*, San Jose, CA, Nov. 29<sup>th</sup> - Dec. 3<sup>rd</sup> 1999, pp. 962-967.
- [19] Vinod, B. and Altiook, T., "Approximating unreliable queuing networks under the assumption of exponentially," *Journal of the Operational Research Society*, 37(3), 1986, pp. 309-316.
- [20] Xie, X. L., "Superposition properties and performance bounds of stochastic timed-event graphs," *IEEE Trans. On Automat. Contr.*, Vol. 39, No. 7, 1994, pp. 1376-1386.
- [21] Zhou, M. C. and Venkatesh, K., *Modeling, Simulation, and Control of Flexible Manufacturing Systems-A Petri net Approach*, World Scientific, Singapore, 1998.
- [22] Zhou, M. C. and M. D. Jeng, "Modeling, Analysis, Simulation, Scheduling, and Control of Semiconductor Manufacturing Systems: A Petri Net Approach," *IEEE Transactions on Semiconductor Manufacturing*, 11(3), 333-357, 1998
- [23] *Visionary Manufacturing Challenges for 2020*, National academy press, Washington, D. C., 1998.



**Ying Tang** (S'99- M'02) received the B.S. and M.S. degrees from the Northeastern University, P. R. China, in 1996 and 1998, respectively, and Ph. D degree from New Jersey Institute of Technology, Newark, NJ, in 2001.

Dr. Tang is currently an Assistant Professor of Electrical and Computer Engineering at Rowan University, Glassboro, NJ. Her research interests include modeling and scheduling of computer-integrated systems, Petri nets and applications, Artificial Intelligence, Reconfigurable systems

design, Networking and communication, and System-on-Chip design. She is a member of the IEEE Systems, Man and Cybernetics and Sigma Xi.



**MengChu Zhou** received his B.S. degree from Nanjing University of Science and Technology, Nanjing, China in 1983, M.S. degree from Beijing Institute of Technology, Beijing, China in 1986, and Ph. D. degree in Computer and Systems Engineering from Rensselaer Polytechnic Institute, Troy, NY in 1990. He joined New Jersey Institute of Technology (NJIT), Newark, NJ in 1990, and is currently a

Professor of Electrical and Computer Engineering, the Director of Discrete-Event Systems Laboratory, and the Director of the M.S. in Computer Engineering Program.

His research interests are in computer-integrated systems, Petri nets, semiconductor manufacturing, multi-lifecycle engineering, and system security. He has over 200 publications including five books, over 90 journal papers, and 14 book-chapters. He co-authored with F. DiCesare, *Petri Net Synthesis for Discrete Event Control of Manufacturing Systems*, Kluwer Academic, Boston, MA, 1993, edited *Petri Nets in Flexible and Agile Automation*, Kluwer Academic, 1995, co-authored with K. Venkatesh Modeling, *Simulation, and Control of Flexible Manufacturing Systems: A Petri Net Approach*, World Scientific, 1998, co-edited with M. P. Fanti, *Deadlock Resolution in Computer-Integrated Systems*, Marcel Dekker, 2005, co-authored with H. Zhu, *Object-Oriented Programming in C++: A Project-based Approach*, Tsinghua University Press, 2005.

He was invited to lecture in Australia, Canada, China, France, Germany, Hong Kong, Italy, Japan, Korea, Mexico, Taiwan, and US. He served as Associate Editor of IEEE Transactions on

Robotics and Automation from 1997 to 2000 and currently Managing Editor of IEEE Transactions on Systems, Man and Cybernetics: Part C, Associate Editor of IEEE Transactions on Automation Science and Engineering, and Editor-in-Chief of International Journal of Intelligent Control and Systems. He was General Co-Chair of 2003 and 2006 IEEE International Conference on System, Man and Cybernetics, 2004 and 2006 IEEE Int. Conf. on Networking, Sensors and Control. He was Program Chair of 1998 IEEE International Conference on System, Man and Cybernetics (SMC) and 1997 IEEE International Conference on Emerging Technologies and Factory Automation, and Guest Editors for IEEE Transactions on Industrial Electronics, and IEEE Transactions on Semiconductor Manufacturing. He organized and chaired over 80 technical sessions and served on program committees for many conferences.

Dr. Zhou has led or participated in twenty-six research and education projects with total budget over \$10M, funded by National Science Foundation, Department of Defense, Engineering Foundation, New Jersey Science and Technology Commission, and industry. He was the recipient of NSF's Research Initiation Award, CIM University-LEAD Award by Society of Manufacturing Engineers, Perlis Research Award by NJIT, Humboldt Research Award for US Senior Scientists, Leadership Award and Academic Achievement Award by Chinese Association for Science and Technology-USA, Asian American Achievement Award by Asian American Heritage Council of New Jersey, and Outstanding Contribution Award from IEEE System, Man and Cybernetics (SMC) Society. He is named Distinguished Lecturer of IEEE SMC Society for 2005-2006. He was the founding chair of Discrete Event Systems Technical Committee of IEEE SMC Society, and Co-Chair (founding) of Semiconductor Factory Automation Technical Committee of IEEE Robotics and Automation Society. He is Fellow of IEEE and a life member of Chinese Association for Science and Technology-USA.