Proceedings of the 2008 Winter Simulation Conference
S. J. Mason, R. R. Hill, L. Mönch, O. Rose, T. Jefferson, J. W. Fowler eds.

QUEUEING MODELS FOR SINGLE MACHINE MANUFACTURING SYSTEMS WITH INTERRUPTIONS

Kan Wu
Leon F. McGinnis
Bert Zwart

School of Industrial and Systems Engineering
Georgia Institute of Technology
Atlanta, GA 30332-0205, USA

ABSTRACT

Queueing theory is a well-known method for evaluating the performance of manufacturing systems. When we want to analyze the performance of a single machine, M/M/1 queues or approximations of G/G/1 queues often are considered a proper choice. However, due to the complex nature of interruptions in manufacturing, the appropriate model should be selected carefully. This paper proposes a systematic way to classify different kinds of interruptions seen in a single machine system. Queueing models for each category are proposed, and event classifications are compared from both the SEMI E10 and queueing theory points of view.

1 INTRODUCTION

Queueing theory plays an important role in evaluating the performance of a semiconductor fab. It gives quantitative measures of the trade-off between cycle time and throughput rate of a manufacturing system. During the past few decades, a significant literature on queueing theory applications to manufacturing systems has appeared, see for example Suri et al. (1993) and Buzacott and Shanthikumar (1993) and references therein.

However, as pointed out by Wu et al. (2007), when we attempt to apply queueing models to a real production system, even for a single machine, a number of issues are encountered. For example, real machines are subject to many kinds of interruptions, including breakdowns, setups and machine-operator interference. The issue is to choose the right queuing models when machines are subject to particular kinds of interruptions.

Hopp and Spearman (1996) describe how to apply G/G/m approximations to evaluate the performance of manufacturing systems by defining service time (ST) using the notion of effective process time (EPT), which accounts for theoretical process time, setup, breakdown, and all other operational delays due to variability effects.

Although these concepts are definitely useful, Wu and Hui (2007) point out that there is a systematic gap between effective process time and service time. When there are time-related events, the service time defined in an M/M/1 queue model cannot be measured precisely in practice, but is only statistically meaningful. Motivated by this issue, Wu et al. (2007) classify all the activities defined in SEMI E10 into Type-I and Type-II events, where Type-I events are WIP (Work-In-Progress) related events and Type-II events are time-related events. In this paper, based on the data hierarchy of FabSim, WIP-related events are called run-based events and time-related events are called time-based events.

This paper is organized as follows. We first classify different events from the perspective of queueing theory in Section II. We propose the corresponding queueing model to each specific category in Section III. In Section IV, the comparison of two classifications—SEMI E10 and queueing theory—is presented. Concluding remarks and directions for future research are given in Section V.

2 CLASSIFICATION OF EVENTS

Queueing theory predicts system performance under the influence of randomness. The randomness mainly comes from natural variability of inter-arrival and service times and from interruptions. Interruptions can be either preemptive or non-preemptive, and are defined as any event which prevents machines from being productive. When there is no interruption and times are exponentially distributed, the M/M/c model suffices. If times are not exponentially distributed, the G/G/c model may be appropriate.

Interruptions are inherent in any manufacturing system. They are caused by the interactions between a machine and an event, which have negative impacts on machine productivity. Based on SEMI E10, downtime is "the time when the equipment is not in a condition, or is not available, to perform its intended function." Examples are breakdowns, experiments, preventive maintenances (PM) and setups. There are basically two types of down states: unscheduled down time and scheduled downtime. Since

978-1-4244-2708-6/08/$25.00 ©2008 IEEE

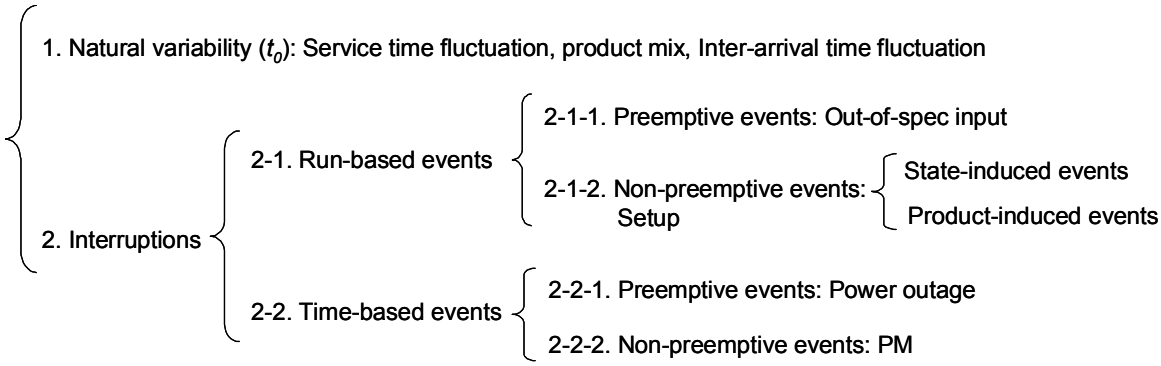

Figure 1: Classifications of events

scheduled downtime implies we have the ability to control when it happens, it is usually non-preemptive. Since unscheduled downtime implies a lack of control, it is often preemptive. Furthermore, failure is defined as "any unscheduled downtime event that changes the equipment to a condition where it cannot perform its intended function." Therefore, in this paper, we specifically define failures as preemptive events. By definition, failures will decrease the availability of machines.

Another approach, proposed by Wu and Hui (2007), is to classify interruptions as run-based and time-based events. As we will see later, this classification is the key to applying queueing models correctly. Run-based events are induced by the existence of WIP, and can only occur when WIP is present, while time-based events can occur anytime, whether or not WIP is present. For example, breakdowns caused by power outages appear to be time dependent, and should be classified as time-based events. On the other hand, setups due to differences in recipes constitute run-based events, as their impacts are only apparent when WIP exists. Some typical examples of Run-based events are out-of-spec input, and setups. Time-based event examples are power outages, PM's, and experiments.

It should be noted that product mix variation is not a run-based event, since it is not an interruption. It should be classified as part of natural variability. Natural variability counts all randomness which does not come from interruptions but from some natural properties designed or inherent in the system, such as release rules, customer product mix, or robot scheduling rules inside a machine.

Both run-based and time-based events can be further decomposed into preemptive and non-preemptive events. A preemptive event can occur anytime during processing, but a non-preemptive event can only occur before or after processing. Therefore, a Run-based non-preemptive event can only occur right before or after job processing, since it cannot interrupt processing and must be induced by the existence of WIP. Some examples are given in fig. 1.

Based on cause, run-based non-preemptive events are furthermore classified as state-induced and product-induced events. State-induced events correspond to machine changing either from busy to idle or idle to busy. For example, a machine goes into a "sleep mode" when it is idle, and requires some warm up time when it returns to production mode.

Product-induced events correspond to switching machine settings for different products. For example, a machine may need some setup time when switching from one recipe to another. There is a fundamental difference between these two types of events: state-induced setups will vanish when machine is fully utilized, since no state change occurs if a machine is always busy. However, product-induced setups are determined by external customer demands, thus, cannot be completely avoided. We may alter the frequencies of product-induced setups by changing scheduling rules, but we simply cannot run one product all the time if customers demand more than one product.

Since capacity is the maximum throughput rate of a machine, this fundamental difference leads to different kinds of impacts on capacity. Product-induced events have impacts on both cycle time and capacity, but state-induced events have impacts only on cycle time, as we will explain in detail in Section III.

In addition to the classifications in fig. 1, interruptions can be classified as: (a) downtime events, or (b) resource contention. Resource contention is caused by activities of other entities or machines. The machine is forced to be idle (but still in production mode) when the required resource is occupied by some other machines. Examples of such resources can be operators, engineers, mask sets, support tools or parts. If those resources are shared among multiple machines and we do not have full control of the occurrences of interruptions, resource contention model has to be considered.

3 QUEUEING MODELS FOR EACH CATEGORY

Mean service time ($1/\mu$), mean inter-arrival time ($1/\lambda$) and

number of servers are the three fundamental parameters of queueing models. While inter-arrival time and number of servers are clearly defined in manufacturing systems, service time (ST) sometimes causes confusion in the presence of interruptions. In the simplest queue, when no interruption exists, the reciprocal of service time is the capacity of a server. The above intuition can be easily verified in an M/M/1 queue. However, when interruptions exist, the definition of service time has to be modified.

When there is no interruption, the mean and variance of service time is simply determined by the events defined in the 1st category of Figure 1, natural variability. Mean service time is the average service time over all product mixes, and the variation of service time considers the impact from natural fluctuation and the differences in product mix.

An important concept is the generalized service time (GST), which reflects the capacity of a workstation under the influence of interruptions. Based on the above insight, GST is defined as

GST = Job departure time – The time epoch when the job first claims capacity of the machine (1)

where job departure time is the time that a job releases the machine capacity. A job claims capacity of a machine if:

1. the job is present at the machine,
2. the preceding job has released the machine, and
3. the machine is ready to process jobs.

Therefore, if a job arrives when the machine is down, it cannot claim capacity until the machine is ready for production. Furthermore, we define production mode strictly by capacity: A machine is in effective production mode if and only if the job consumes its capacity. Therefore, the setup times caused by product-induced events are counted into GST, but the setup times caused by state-induced events are not.

Based on the above definition, GST is the summation of service time, product-induced setup time, and the downtimes of all preemptive events occurring during that service time,

$$G = S + \sum_{i=1}^{N(S)} D_i + T, \qquad (2)$$

where G stands for GST, S stands for ST, N(S) is the number of preemptive events (e.g. breakdowns) during S, D_i is the i-th downtime, and T stands for duration of the run-based non-preemptive product-induced events. Although Eq. (2) is based on Eq. (1) and its related insights, this framework is similar to the concept of general processing time defined in Adan and Resing (2001), Chapter 10.2.

Furthermore, when both service time and downtime are generally distributed, and the mean time between preemptive events is exponentially distributed, the mean of GST is

$$E(G) = E(S) + E(S)\eta E(D) + E(T), \qquad (3)$$

where $1/\eta$ is the mean time between preemptive events, and D stands for downtime of the preemptive events which occur during the service period.

Availability (A) is defined as

$$A = \frac{m_f}{m_f + m_r}, \qquad (4)$$

where m_f is the mean time between failures (MTBF), and m_r is the mean downtime or mean time to repair (MTTR). Based on the assumption of Eq. (3), Eq. (4) becomes

$$A = \frac{m_f}{m_f + m_r} = \frac{1/\eta}{1/\eta + E(D)}. \qquad (4a)$$

Therefore, based on Eq. (4a), Eq. (3) can be restated as

$$E(G) = E(S)/A + E(T). \qquad (3a)$$

Furthermore, if both m_f and m_r are exponentially distributed, availability can be expressed as

$$A = \frac{m_f}{m_f + m_r} = \frac{1/\eta}{1/\eta + 1/\theta} = \frac{\theta}{\theta + \eta}, \qquad (4b)$$

where $1/\eta$ is the mean time between run-based preemptive events and $1/\theta$ is the mean time to repair those failures.

Using the definition of GST, cycle time (CT) can be explicitly defined as

$$CT = QT + GST, \qquad (5)$$

where QT stands for queueing time.

Another notable extension of service time is the concept of effective process time (EPT). Hopp and Spearman (1996) described EPT as follows: It is the total time "seen" by a job at a station. It does not matter whether the job is actually being processed or is being held up because machine is being repaired, undergoing a setup, rework, or waiting for its operators.

The definition of EPT is almost the same as the definition of GST defined in Eq. (1), except for the conditions under which machine capacity is claimed. For EPT, we only:

1. the job is present at the machine,
2. the preceding job has released the machine, but the machine may or may not be ready to process jobs.

Thus, EPT is exactly the same as GST only when:

(a) all interruptions are run-based,
(b) all run-based non-preemptive events are product-induced.

For situation (b), if there are state-induced events, EPT will incorporate the event time into its duration, which will lead to the miscalculation of capacity as explained before. A more detailed explanation will be given in section III.A-2.

For situation (a), if there are time-based interruptions, an arriving job may be blocked by an interruption. This waiting period will be counted into EPT (but not in GST). Since the interruptions have higher probability to block a job when the job arrival rate is high, the accuracy of EPT depends on the machine utilization, which complicates

our models, and contradicts the assumption that service time is independent of utilization.

When conditions (a) and(b) described above hold, EPT is the same as GST. Thus, from (3a), we have:

$$E(EPT) = E(S)/A + E(T), \qquad (3b)$$

In Factory Physics, Hopp and Spearman (1996) derive queueing models based on Eq. (3b). As argued here, those models are exact only when situation (a) and (b) hold. This could lead to errors if time-based events are common.

When there are time-based events, the concept of EPT should be used carefully, because in general,

$$CT \neq QT + EPT,$$

Throughout this remainder of this section, the models and methods we use are closely related to those presented in Adan and Resing (2001).

3.1 Models for Run-based Interruptions

Here we address models for situation 2-1 in Fig. 1.

3.1.1 Models for Run-based Preemptive Interruptions

For run-based interruptions, we analyze the preemptive events (2-1-1) first and start from the simplest case, i.e., M/M/1_Run-based preemptive event model. In this model, the machine can break down only when it is processing jobs. Jobs arrive according to a Poisson stream with rate λ, and service times are exponentially distributed with mean 1/μ. The up and down times are also exponentially distributed with means 1/η and 1/θ. For stability, we assume that

$$\rho = \lambda/(\mu A) < 1,$$

where A is defined in Eq. (4b). Based on the property of "Poisson arrivals see time averages" (PASTA) by Wolff (1982), an arriving job finds on average *E(L)* jobs in system and encounters η*E(L)*/μ breakdowns. Furthermore, each job, on arrival, sees the machine is already down with probability ρ(1-A) (We need ρ in front of (1-A), since it is run-based.). Therefore,

$$\begin{aligned} E(QT) &= \frac{E(L)}{\mu} + \eta\left[\frac{E(L)}{\mu}\right]\frac{1}{\theta} + \rho(1-A)\frac{1}{\theta} \\ &= \frac{E(L^q)+\rho}{\mu} + \eta\left[\frac{E(L^q)+\rho}{\mu}\right]\frac{1}{\theta} + \rho(1-A)\frac{1}{\theta} \\ &= \frac{1}{\mu}E(L^q)(1+\frac{\eta}{\theta}) + \frac{\rho}{\mu}(1+\frac{\eta}{\theta}) + \rho(1-A)\frac{1}{\theta} \\ &= \frac{E(L^q)}{\mu A} + \frac{\rho}{\mu A} + \rho(1-A)\frac{1}{\theta}, \end{aligned} \qquad (6)$$

where L is the number of jobs in the system (for both waiting and processing jobs), L^q is the number of jobs in queue and A is availability from Eq. (4b). Based on Little's law, which is proposed by Little (1961), QT can be further simplified as follows,

$$E(L^q) = \lambda E(QT), \qquad (7)$$

$$E(QT) = \frac{\rho}{1-\rho}\frac{1}{\mu A} + \frac{\rho}{1-\rho}\frac{1-A}{\theta}, \qquad (8)$$

and

$$E(CT) = E(QT) + E(G) = \frac{1}{1-\rho}\frac{1}{\mu A} + \frac{\rho}{1-\rho}\frac{1-A}{\theta}. \qquad (9)$$

A more general case is the M/G/1_Run-based preemptive event model. The assumptions are basically the same as above except the service time and down time are generally distributed. The first and second moment of the service time are denoted by *E(S)* and *E(S²)*. The up time between two breakdowns is exponentially distributed with mean 1/η. The first and second moment of the down time are denoted by *E(D)* and *E(D²)*. For stability, we assume

$$\rho_G = \lambda E(G) < 1.$$

While Adan and Resing (2001) dealt with the time-based interruptions, we have extended their results to run-based interruptions as follows: By PASTA, an arriving job finds on average $E(L^q)$ jobs in queue, and a working job with probability ρ_G. Therefore,

$$E(QT) = E(L^q)E(G) + \rho_G E(R_G), \qquad (10)$$

where

$$E(G) = E(S) + E(S)\eta E(D),$$

$$E(G^2) = E(S^2)\left[1+\eta E(D)\right]^2 + E(S)\eta E(D^2),$$

$$\rho_G = \lambda E(G),$$

$$E(R_G) = E(G^2)/2E(G),$$

By Little's law, QT can be further simplified as:

$$E(L^q) = \lambda E(QT), \qquad (11a)$$

$$E(QT) = \frac{\rho_G E(R_G)}{(1-\rho_G)} = \left(\frac{1+c_e^2}{2}\right)\left(\frac{\rho_G}{1-\rho_G}\right)E(EPT), \qquad (11b)$$

where c_e^2 stand for the squared coefficient of variation (SCV) of EPT, and

$$E(CT) = E(QT) + E(G).$$

Eq. (11b) holds because we are only dealing with run-based preemptive events. Since Eq. (8) is a special case of Eq. (11b), it can be shown that Eq. (8) and (11b) are both consistent with the results presented by Hopp and Spearman (1996).

In wafer fabs, Poisson arrivals is a reasonable assumption, especially when each workstation has multiple downstream servers to feed, and multiple upstream workstations to feed it. Furthermore, when there are multiple sources of failures, assuming that MTBF is exponentially distributed is also reasonable. Therefore, M/G/1 queues suffice for many situations in wafer fabs. For the more general situation, G/G/1_Run-based preemptive model, we need to resort to approximations. Together with the non-preemptive cases, the formulations are given in the next section.

3.1.2 Models for Run-based Non-Preemptive Interruptions

We will introduce two different formulations for the case of run-based non-preemptive events (2-1-2). The first model is introduced by Adan and Resing (2001), which assumes the machine needs a setup whenever it changes state from idle to production. For example, the machine is turned off when it is idle, and turned on again when a new job arrives, but restarting takes some time. Another example is the load activity of a machine. It is common that the load activity of the 1st job of a series of identical jobs needs to be considered explicitly, but for the subsequent jobs in the series, the load activity can be done in parallel with the GST of the active job. Therefore, the load time of the subsequent jobs can be ignored.

For this type of event, we start from the simplest case, M/M/1_Run-based non-preemptive state-induced event model. In this model, jobs arrive according to a Poisson process with rate λ, and service times are exponentially distributed with mean 1/μ. The setup time is exponentially distributed with means 1/θ. For stability, we assume

$$\rho = \lambda/\mu < 1.$$

By PASTA, an arriving job finds on average $E(L^q)$ jobs in queue and sees a working job with probability ρ. It arrives when the machine is not in operation and experiences a setup with probability 1-ρ. Therefore,

$$E(QT) = E(L^q)\frac{1}{\mu} + \rho\frac{1}{\mu} + (1-\rho)\frac{1}{\theta}, \tag{12}$$

combining Little's law,

$$E(L^q) = \lambda E(QT),$$

with (12) we get

$$E(QT) = \frac{\rho}{1-\rho}\frac{1}{\mu} + \frac{1}{\theta}, \tag{13}$$

and thus:

$$E(CT) = E(QT) + E(S) = \frac{1}{1-\rho}\frac{1}{\mu} + \frac{1}{\theta}. \tag{14}$$

This expression has also been derived by Adan and Resing (2001) in a different way. From Eq. (13), the queueing time of an M/M/1_Run-based non-preemptive state-induced event model is just the queueing time of an M/M/1 model plus an extra setup time.

A more general case is the M/G/1_run-based non-preemptive state-induced event model. The assumptions are the same as above, except the service time and setup time are generally distributed. The first and second moment of the service time are denoted by $E(S)$ and $E(S^2)$. The first and second moment of the setup time are denoted by $E(T)$ and $E(T^2)$. For stability, we assume

$$\rho = \lambda E(S) < 1.$$

Assuming PASTA, an arriving job finds on average $E(L^q)$ jobs in queue and sees a working job with probability ρ. Otherwise, it arrives either when the machine is idle or in a setup phase. Therefore,

$$E(QT) = E(L^q)E(S) + \rho E(R_S) + (1-\rho)\left[\frac{1/\lambda}{1/\lambda + E(T)}E(T) + \frac{E(T)}{1/\lambda + E(T)}E(R_T)\right], \tag{15}$$

where

$$E(R_S) = E(S^2)/2E(S),$$
$$E(R_T) = E(T^2)/2E(T).$$

Combining (15) and Little's law, we get:

$$E(QT) = \frac{\rho E(R_S)}{(1-\rho)} + \frac{1/\lambda}{1/\lambda + E(T)}E(T) + \frac{E(T)}{1/\lambda + E(T)}E(R_T), \tag{16}$$

and

$$E(CT) = E(QT) + E(S). \tag{16a}$$

See also Adan and Resing (2001) for a different derivation, where the expected cycle time is derived directly. From Eq. (16), queueing time of an M/G/1_Run-based non-preemptive state-induced event model is just the queueing time of an M/G/1 model plus the extra setup time.

Recall that the state-induced events impact only cycle time and not capacity. This can be verified by Eq. (13), (16) and the stability condition. Another interesting observation is that when λ approaches μ (i.e. ρ approaches 1), the queueing time of the original M/M/1 or M/G/1 queueing models increase without limit, but the extra setup times are bounded.

In the run-based non-preemptive product-induced event model, we assume the machine needs a setup for product changeovers. It is addressed by Hopp and Spearman (1996). The setups occur due to changes in the production process induced by switching products.

This model assumes the machine processes an average of N_t jobs between setups, and the probability of doing a setup after any job is equal (i.e. $1/N_t$). The setup times have a mean of t_t and a standard deviation of σ_t. Since GST is the same as EPT in run-based non-preemptive product-induced event model, we have

$$GST = S + T = EPT,$$
$$E(G) = E(S+T) = E(S) + E(T) = t_0 + t_t / N_t = t_e,$$

where T is the setup time experienced by a job, t_e is the mean of EPT and t_0 is the mean of service time. For stability, we assume

$$\rho = \lambda\, E(G) = \lambda\,(t_0 + t_t/N_t) < 1.$$

Comparing the stability conditions of state-induced model and product-induced model, it is clear that the setup times of product-induced models have direct impact on capacity, while the setup times of state-induced models do not.

By PASTA, an arriving job finds on average $E(L^q)$ jobs in queues and sees a working job with probability ρ. It experiences a setup with probability $1/N_t$. Similar to Eq.

(10), we have

$$E(QT) = E(L^q)E(G) + \rho_G E(R_G) = E(L^q)\big(E(S)+E(T)\big) + \rho_G E(R_G), \quad (17)$$

where

$$\rho_G = \lambda E(G),$$
$$E(R_G) = E(G^2)/2E(G),$$
$$E(G^2) = E(S^2) + 2E(S)E(T) + E(T^2).$$

The equations for QT and CT can be derived accordingly.

3.1.3 Generalizing the Poisson Arrival Assumption

Since we already know GST is the same as EPT in this case, we can just give the results directly from Factory Physics (1996). The mean and variance of EPT are as follows,

$$t_e = t_0 + t_t / N_t, \quad (18)$$

$$\sigma_e^2 = \sigma_0^2 + \frac{\sigma_t^2}{N_t} + \frac{N_t - 1}{N_t^2} t_t^2. \quad (19)$$

Together with Eq. (20), these equations are the foundations of the approximations for a G/G/1_Run-based non-preemptive product-induced event model. The exact M/G/1 and M/M/1 models can be obtained by assigning c_a^2 to 1 and both c_a^2 and c_e^2 to 1, respectively.

Despite a lack of analysis for time-based and state-induced events, Factory Physics has an extensive discussion of run-based preemptive events and non-preemptive product-induced events based on the concept of EPT.

By using the heavy traffic approximations of Whitt (1993), queueing time (QT) of G/G/1 queues can be estimated by Eq. (20) for both run-based preemptive events and non-preemptive product-induced events,

$$E(QT) = \left(\frac{c_a^2 + c_e^2}{2}\right)\left(\frac{\rho}{1-\rho}\right)E(EPT), \quad (20)$$

where ρ is utilization and c_a^2 and c_e^2 stand for SCV of arrival interval and EPT, respectively. Table 1 summarizes the formulations of EPT and c_e^2 used in Eq. (20).

Table 1. Parameters for computing QT under the existence of Run-based events (from Factory Physics)

Situation	Natural	Preemptive	Non-preemptive
t_e	t_0	t_0 / A	$t_0 + t_t / N_t$
σ_e^2	$t_0^2 c_0^2$	$\sigma_0^2/A^2 + \frac{(m_r^2+\sigma_r^2)(1-A)t_0}{Am_r}$	$\sigma_0^2 + \frac{\sigma_t^2}{N_t} + \frac{N_t-1}{N_t^2}t_t^2$
c_e^2	c_0^2	$c_0^2 + (1+c_r^2)A(1-A)m_r/t_0$	σ_e^2/t_e^2

3.2 Models for Time-based Interruptions

Here we address models for situation (2-2) in Fig. 1.

3.2.1 Models for Time-based Preemptive Interruptions

For time-based interruptions, we analyze the preemptive events (2-2-1) first and start from the simplest case, M/M/1_Time-based preemptive event model. In this model, the machine can break down anytime instead of only during processing. Jobs arrive according to a Poisson process with rate λ, and service times are exponentially distributed with mean $1/\mu$. The up and down times are also exponentially distributed with means $1/\eta$ and $1/\theta$. For stability, we assume

$$\rho = \lambda/(\mu A) < 1.$$

By PASTA, an arriving job finds on average $E(L^q)$ jobs in queues and sees a working job with probability ρ. Each job encounters $\eta(E(L^q)+\rho)/\mu$ breakdowns in total. Furthermore, each arriving job sees the machine is already down with probability (1-A). The above scenario is almost the same as the derivations of Eq. (6) except for the last term. Therefore,

$$E(QT) = \frac{E(L^q)}{\mu A} + \frac{\rho}{\mu A} + (1-A)\frac{1}{\theta}. \quad (21)$$

Combining Little's law and (21), we get

$$E(QT) = \frac{\rho}{1-\rho}\frac{1}{\mu A} + \frac{(1-A)}{1-\rho}\frac{1}{\theta}, \quad (22)$$

and

$$E(CT) = E(QT) + E(G) = \frac{1}{1-\rho}\frac{1}{\mu A} + \frac{(1-A)}{1-\rho}\frac{1}{\theta}. \quad (23)$$

See Adan and Resing (2001) for a different derivation.

Comparing Eq. (22) with Eq. (8), the gap between M/M/1_Time-based preemptive event and M/M/1_Run-based preemptive event models is

$$Gap = \left(\frac{\rho}{1-\rho}\frac{1}{\mu A} + \frac{1}{1-\rho}\frac{(1-A)}{\theta}\right) - \left(\frac{\rho}{1-\rho}\frac{1}{\mu A} + \frac{\rho}{1-\rho}\frac{1-A}{\theta}\right) = \frac{(1-A)}{\theta}. \quad (24)$$

This gap is also the gap between M/M/1_Time-based preemptive event model and the preemptive outage model introduced in Factory Physics, since the preemptive outage model is identical to the M/M/1_Run-based preemptive event model.

A more general case is the M/G/1_Time-based preemptive event model, which introduced by Adan and Resing (2001). The assumptions are the same as above, except the service time and down time are generally distributed. The first and second moment of the service time are denoted by $E(S)$ and $E(S^2)$. The up time between two break-

downs is exponentially distributed with mean $1/\eta$. The first and second moment of the down time are denoted by $E(D)$ and $E(D^2)$. For stability, we assume

$$\rho_G = \lambda E(G) < 1.$$

Similar to the derivations of Eq. (10), an arriving job finds on average $E(L^q)$ jobs in queues, and a working job with probability ρ_G. Furthermore, an arriving job has a certain probability, $(1-A_{NP})(1-\rho_G)$, to see the machine down in a non-processing period. The above scenario is the same as in the derivations of Eq. (10) except for the last term. Therefore,

$$\begin{aligned} E(QT) &= E(L^q)E(G) + \rho_G E(R_G) \\ &\quad + (1-\rho_G)(1-A_{NP})E(R_D), \end{aligned} \tag{25}$$

where A_{NP} is availability of the machine during non-processing period, and

$$E(G) = E(S) + E(S)\eta E(D),$$

$$E(G^2) = E(S^2)\left[1+\eta \mathrm{E}(D)\right]^2 + E(S)\eta E(D^2),$$

$$\rho_G = \lambda E(G),$$

$$E(R_G) = E(G^2)/2E(G),$$

$$E(R_D) = E(D^2)/2E(D),$$

$$A_{NP} = \frac{1/(\lambda+\eta)}{1/(\lambda+\eta) + E(D)\eta/(\lambda+\eta)} = \frac{1}{1+E(D)\eta}.$$

Combining Little's law and (25) we get

$$E(QT) = \frac{\rho_G E(R_G)}{(1-\rho_G)} + \frac{E(D)\eta}{1+E(D)\eta} E(R_D), \tag{26}$$

and

$$E(CT) = E(QT) + E(G). \tag{27}$$

Comparing Eq. (26) with Eq. (11), the gap between M/G/1_Time-based preemptive event model and M/G/1_ Run-based preemptive event models is

$$Gap = (1-A_{NP})E(R_D). \tag{28}$$

Theorem 1 (*Decomposition property of preemptive events).*

For any specific type of preemptive interruption, there is a gap between its run-based and time-based models. This gap is independent of machine utilization.

The gap in Eq. (24) is a degenerate case of Eq. (28) when the down time is exponentially distributed. Eq. (28) is very important, since the models given in Factory Physics can be improved by adding this term, when we have time-based preemptive events. Therefore, for the most general situation (the G/G/1_Time-based preemptive event model), a better approximation is obtained by combing Eq. (28) and Eq. (20).

3.2.2 Models for Time-based Non-Preemptive Interruptions

Identifying the role of the category (2-2-2) from Fig. 1 in the application of queueing theory to manufacturing systems is an important contribution of this paper. This type of event is very common in practice. Actually, most of the events listed in the "Summary of Time" of SEMI E10 are in this category.

Process experiments, equipment experiments, preemptive maintenance, tool modifications, and change of consumables are examples of this type of interruption. The modeling of this type of event becomes extremely complicated if we want to analyze the behavior of each interruption one by one, since the occurrences of each interruption have correlations with previous occurrences. The correlation could be based on a period of time, the usage rate (i.e. utilization), or both. For example, a part needs to be replaced every 3 months, but a chemical needs to be replaced after 800 usages.

Fortunately, what we usually care about is the overall system performance instead of the behavior of each interruption. Instead of looking at each specific event, if we look at this category of events, assuming a Poisson event rate is reasonable, since the arrivals are triggered by many different sources.

The overall behavior of this type of event can be classified as time-based non-preemptive. Furthermore, we usually have some level of control on the occurrence of each interruption. For example, we may postpone the execution of a PM until the machine is idle. If this is the case, this type of event can be modeled by the non-preemptive priority queues with two priorities, where the interruption has low priority and the job processing has high priority. For the simplest case, M/M/1_Non-preemptive priority queues with two priorities, Gross and Harris (1998) developed the following:

$$E(L_1^q) = \frac{\lambda_1 \hat{\rho}}{\mu_1} \frac{\lambda_1/\lambda + (\lambda_2/\lambda)(\mu_1^2/\mu_2^2)}{1-\lambda_1/\mu_1}, \tag{29}$$

$$E(L_2^q) = \frac{(\lambda_2/\mu_1)\hat{\rho}}{1-\lambda_1/\mu_1} \frac{\lambda_1/\lambda + (\lambda_2/\lambda)(\mu_1^2/\mu_2^2)}{1-\lambda_1/\mu_1-\lambda_2/\mu_2}, \tag{30}$$

$$E(L^q) = E(L_1^q) + E(L_2^q), \tag{31}$$

where

$$\lambda = \lambda_1 + \lambda_2,$$

$$\hat{\rho} = \lambda/\mu_1,$$

λ_1 and λ_2 are the arrival rates of high and low priority jobs, and μ_1 and μ_2 are the service rates of high and low priority jobs respectively.

If the service times are generally distributed and the discipline is FIFO, based on Adan and Resing (2001), the equations for M/G/1_Non-preemptive priority queues with two priorities are as follows:

$$E(QT_1) = \frac{\rho_1 E(R_1) + \rho_2 E(R_2)}{1-\rho_1}, \tag{32}$$

$$E(QT_2) = \frac{\rho_1 E(R_1) + \rho_2 E(R_2)}{(1-\rho_1-\rho_2)(1-\rho_1)}, \tag{33}$$

$$E(QT) = \frac{\lambda_1}{\lambda_1 + \lambda_2} E(QT_1) + \frac{\lambda_2}{\lambda_1 + \lambda_2} E(QT_2), \quad (34)$$

$$E(CT_i) = E(QT_i) + E(S_i), \quad i = 1, 2$$

$$E(CT) = E(QT) + \frac{\lambda_1}{\lambda_1 + \lambda_2} E(S_1) + \frac{\lambda_2}{\lambda_1 + \lambda_2} E(S_2),$$

where

$$\rho_1 = \lambda_1 / \mu_1 \text{ and } \rho_2 = \lambda_2 / \mu_2,$$

$$E(R_i) = E(S_i^2) / 2E(S_i), \quad i = 1, 2$$

λ_1 and λ_2 are the arrival rates of high and low priority jobs, and μ_1 and μ_2 are the service rates of high and low priority jobs respectively.

In reality, the control mechanism may be more complex than the above assumptions. For example, we may postpone the occurrence of a PM to some period when the machine is less busy, but may not postpone it too much without jeopardizing the quality of products. In this case, we may model it as follows: the interruption has low priority until its queueing time reaches a predetermined threshold, after which it switches to high priority. We call this scenario "delayed priority queues". However, the derivations involve the analysis of high dimensional Markov chains and is left as a direction for future research.

4 COMPARISON OF QUEUEING CLASSIFICATIONS AND SEMI E10

In this paper, we have proposed a systematic way to classify the shop floor events from a queueing theoretic point of view. The purpose of these classifications is to offer guidance for apply queueing models properly in practice. The classification is motivated on a key question: "How should we define service time under different circumstances?"

Service rate is one of the three fundamental elements of queueing models. While the other two, arrival interval and server count, are clearly defined and observable, in contrast, service time is not easily observed. The key idea is that service rate is the maximum throughput rate, or capacity, of a system.

In reality, people often mix raw processing time (RPT) with service time. Based on Wu and Hui (2007), a commonly acknowledged way to define raw process time is 'the total duration that a lot stays in a tool and is engaged in process related activities'. It is obviously different from service time (or GST), which is defined from the view point of capacity. For example, a preemptive breakdown, which occurs during processing, is counted into GST, but not RPT.

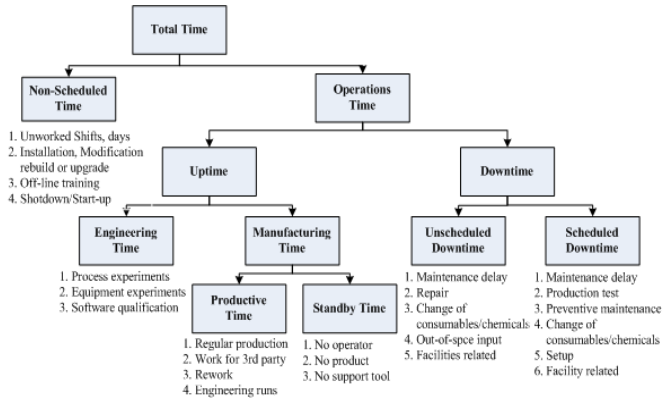

Figure 2: Summary of Time (from SEMI E10)

The purpose of SEMI E10 is to "establish a common basis for communication between users and suppliers of semiconductor manufacturing equipment by providing standards for measuring RAM performance of that equipment in a manufacturing environment," where RAM stands for reliability, availability, and maintainability. The classification of time proposed in SEMI E10 is summarized in Fig. 2. The purposes of these two types of classifications (Fig. 1 and Fig. 2) are very different. However, they both attempt to classify all the shop floor events.

The activity classifications of SEMI E10 are commonly adopted for productivity improvement projects in practice. By comparing these two classifications, we can see how each queueing model interacts with the activities in practice from the view point of productivity improvement.

Un-worked shifts, installation, modification, rebuild or upgrade, off-line training, shutdown/start-up, which belong to Non-Schedule Time are time-based non-preemptive events. Process and equipment experiments, software qualification listed under Engineering Time are also time-based non-preemptive events.

Activities under Productive Time, such as regular production, work for 3rd party, rework, and engineering runs can be viewed as product mix variability. However, a portion of these activities also belongs to run-based non-preemptive state-induced events, since load and unload are classified into Productive Time based SEMI E10.

Under Unscheduled Downtime, while maintenance delay is a resource contention problem, change of consumables/ chemicals, repair, and facilities related unscheduled downtime (e.g., power outage) are time-based preemptive events in general. Out-of-spec input can be viewed as a run-based preemptive event.

In the category of Scheduled Downtime, production test, preventive maintenance, change of consumable/chemicals, and facilities related scheduled downtime all are time-based non-preemptive events. However, maintenance delay is a resource contention problem. Setup is a run-based non-preemptive event.

Under Standby Time, "no product" does not belong to any category in Fig. 1, but simply corresponds to the idle

time of a machine. The situations with "no operator" and "no support tool" have to be modeled as resource contentions, but do not belong to any of the categories in Fig. 1. As we have discussed, they can only occur during the appearance of the above events, and their impact to the system must be modeled using those events. For example, a machine may wait for support tools during a process experiment or change of consumables, and a machine may wait for operators during a setup. Therefore, in SEMI E10 Summary of Time, the standby time caused by no operator and no support tool will occur together with one of the above events in other categories.

Through the above analysis, we conclude that the majority of the events on the shop floor are time-based, and most of the time-based events are non-preemptive. Because the purpose of SEMI E10 is to measure equipment performance, understanding the sources of the events is essential. However, the SEMI E10 classifications can cause confusion in applying queueing models. For the purpose of applying queueing theory, the classification we propose in this paper will avoid that confusion.

5 CONCLUSIONS

In this paper, we demonstrate how to model performance with different kinds of interruptions through different queueing models. Classifying events appropriately is key to correctly applying queuing models.

An important finding of this paper is the gap between models with time-based and run-based preemptive events. The gap is the product of unavailability of a machine during non-processing period and the residual downtime. This gap is independent of machine utilization. In section IV, we have also compared two different event classifications from SEMI E10 and queueing theory's point of view.

We have presented a newly developed classification method to apply queueing theory in manufacturing systems. Although we have proposed corresponding queueing models for each category, a number of questions remain to be answered, such as how to analyze the delayed priority queue and how to develop an integrated model of all types of events.

ACKNOWLEDGMENTS

The authors wish to thank Dr. Ron Billings for his helpful comments on run-based and time-based events.

REFERENCES

Adan, I., and J. Resing. 2001. *Queueing Theory*. Lecture Notes. <http://www.win.tue.nl/~iadan/queueing.pdf>

Buzacott, J. A., and G. Shanthikumar. 1993. *Stochastic Models of Manufacturing Systems*. New Jersey: Prentice-Hall.

FabSim: A Discrete-Event Simulation Model for Wafer Fabs <http://www2.isye.gatech.edu/~rbilling/courses/isye4803/Project/FabSim.pdf>

Gross, D., and C. M. Harris. 1998. *Queueing Theory*. New York: Wiley.

Hopp, W. J., and M. L. Spearman. 1996. *Factory Physics*. Chicago, IL: IRWIN.

Little, J. D. C. 1961. A Proof of the Queueing Formula: L=λW. *Operations Research* 9: 383–387.

SEMI E10, Specification for Definition and Measurement of Equipment Reliability, Availability, and Maintainability, *Book of SEMI Standards*. Mountain View, CA: SEMI.

Suri, R., J. L. Sanders and M. Kamath. 1993. Performance Evaluation of Production Networks. *Handbooks in OR & MS* 4: 199–286.

Wolff, R. W. 1982. Poisson Arrivals See Time Averages. *Operations Research* 30: 223-231.

Wolff, R. W. 1989. *Stochastic Modeling and the Theory of Queues*. New Jersey: Prentice-Hall.

Whitt, W. 1993. Approximations for the GI/G/m queue. *Production and Operations Management* 2: 114–161.

Whitt, W. 2003. *Stochastic Process Limits*. New York: Springer.

Wu, K., and K. Hui. 2007. The Determination and Indetermination of Service Times in Manufacturing Systems. *IEEE Trans. Semi. Manu.* 21:72–82.

Wu, K., L. F. McGinnis, and B. Zwart. 2007. Compatibility of Queueing Theory, Manufacturing Systems and SEMI Standards. In *Proceedings IEEE CASE 2007*.

KAN WU received the B.S. degree in nuclear engineering from National Tsing Hua University, Hsinchu, Taiwan. He received the M.S. degree in industrial engineering and operations research and the M.E. degree in nuclear engineering from the University of California, Berkeley in 1996. Afterward, he was an engineer with Tefen, Ltd., and with Taiwan Semi-conductor Manufacturing Company. During 2003 to 2005, he was an IE manager at Inotera Memories Inc. and also served as a reviewer for IEEE TSM. Currently, he is a Ph.D student at ISyE, Georgia Tech. His research interests include production planning, scheduling, and dispatching in semiconductor industry.

LEON MCGINNIS is the Gwaltney Professor of Manufacturing Systems at Georgia Tech. Professor McGinnis is internationally known for his leadership in the material handling research community and his research in the area of discrete event logistics systems. He has received several awards for his innovative research, including the David F. Baker Award from IIE, the Reed-Apple Award from the Material Handling Education Foundation, and the Material Handling Innovation Pioneer award from Material Handling Management Magazine. He is author

or editor of seven books and more than 110 technical publications. At Georgia Tech, Professor McGinnis has held leadership positions in a number of industry-focused centers and programs, including the Material Handling Research Center, the Computer Integrated Manufacturing Systems Program, the Manufacturing Research Center, and the newly-formed Product/Systems Lifecycle Management Center. His current research explores the application of PLM technologies to the design and management of highly capitalized factories.

BERT ZWART is Coca Cola Associate Professor of Industrial and Systems Engineering at Georgia Tech. Bert holds an M.A. in Econometrics from the Free University, Amsterdam, and a Ph.D in Applied Mathematics from Eindhoven University of Technology. He serves on several conference program committees, and serves on the editorial boards of Operations Research, Mathematics of Operations Research, Mathematical Methods of Operations Research, and Queueing Systems. His research is concerned with the modeling, analysis and simulation of stochastic systems arising in actuarial and financial mathematics, computer and communication systems, manufacturing systems, and customer contact centers.