Abstract. This paper considers a queueing network model for the performance analysis of a flexible manufacturing system composed of workstations with limited buffers, where a single automated guided vehicle is used for materials handling. An approximation algorithm is developed to obtain the system performance measures by using the reversibility, and some numerical examples are tested to show the effectiveness of the algorithm in comparison with simulation results.

Keywords: FMS, AGV, Queueing Network, Reversibility, Performance

1 Introduction

A flexible manufacturing system (FMS) can be described as a system comprised of a group of workstations integrated with an automated material handling system under computer control. Depending on the production amount and the variety of parts to be produced, the workstations are often composed of quality-different machining centers or various numerically controlled (NC) machines. The material handling system is employed to transport parts to and from the workstations, and parts are mounted on pallets that move through the system.

Strategic planning of an FMS requires a thorough understanding of the system performance so that many researchers have concentrated on the stochastic models to evaluate the performance of FMSs. Yao and Buzacott [5] have developed an open queueing network model with general service times and limited local buffers. Ma and Matsui [3] have evaluated the flexible machining systems for routing comparisons. Liu and Hung [2] have considered the real-time deadlock free control strategy for single multi-load automated guided vehicle (AGV).

As a material handling system with considerable flexibility, the AGV system continues to call dramatic attention to both warehousing and manufacturing operations. By the way, such an AGV employment may incur two major problems. One of them is the problem of possible collision between AGVs, and the other one is the problem of machine idling and blocking due to AGV’s capacity restriction. Thus, most existing FMSs have many automatic pallet changing facilities (APC) which are installed at each machine and play individually a role of limited local buffers.

In this paper, a queueing network model is considered for the performance analysis of an FMS composed of several workstations, each with both limited input and output.
buffers, where machine blocking is occurred and a single AGV is used for handling input and output materials. An approximation algorithm is developed to obtain the system performance measures such as system throughput, machine utilization and blocking probability. And, a variety of numerical examples are tested to show the effectiveness of the algorithm in comparison with simulation results.

2 The Model and Analysis

2.1 The Model Description

The FMS model consists of a set of \( n \) workstations. Each workstation \( i \) (\( i=1,\ldots,n \)) has a single machine with both limited input and output buffers where the machine performs a set of operations in an exponential service time distribution. Since each product type requires its own service time at a workstation, the overall service time at each workstation may reasonably be represented by an exponential distribution. The capacities of input and output buffers are limited up to \( IB_i \) and \( OB_i \) respectively. All the workstations are linked to a central storage by a single AGV. The capacity of the central storage is unlimited, and external arrivals at the central storage follow a Poisson process with rate \( \lambda \).

Specifically, the AGV delivers the input parts from the central storage (AS/RS) to each input buffer of workstations, and carries the finished parts away from each output buffer of workstations to the central storage with exponential service time distribution. Also, it distributes all parts from central storage to the workstations according to a routing probability \( r_i \), which can be interpreted as the proportion of part dispatching from the central storage to workstation \( i \). The FCFS (first come first served) discipline is adopted here for the services of AGV and machines.

Any part can be blocked on arrival at an input buffer which is already full with earlier-arrived parts. Such a blocked part will be recirculated instead of occupying the AGV and waiting in front of the workstation. Any finished part can also be blocked on arrival at an output buffer which is already full with earlier-finished parts. Such a blocked part will occupy the machine to remain blocked until a part departure occurs from the output buffer. During such a blocking time, the machine cannot render service to any other part that might be waiting at its input buffer.

An approximation procedure is developed by decomposing the queueing networks into input and output level queue with revised arrival and service processes and analyzing each individual queue in isolation. The output-level queue will be analyzed by using the reversibility without solving the balance equation directly, upon which the clearance service time accommodating all the possible delays can be found. Then, the input-level queues will be analyzed.

2.2 Analysis of the Model

The output-level queue is composed of the central storage, output buffers and a single AGV. The capacity of each buffer is augmented by 1 in order to accommodate any
blocking unit. Let arrivals in each buffer occur in a Poisson process with rate $\lambda^j$ and the states of the system be denoted by $(k_0, k_1, \ldots, k_n)$ and (idle), where $k_j (j = 0, \ldots, n)$ denotes the number of parts at buffer $j$ and the state (idle) represents the situation that there is no part within the system. Since all inter-event-occurrence times are exponentially distributed, the system process is a Markov process with state space $\{ (\text{idle}), (k_2, \ldots, k_n) \mid 0 \leq k_j \leq \infty, 0 \leq k_i \leq \text{UB}_i + 1 \text{ for } i = 1, \ldots, n \}$.

In spite of the complexity nature, it is fortunate enough to show that the steady state probabilities can be easily obtained using reversibility without trying to solve any balance equations directly. The procedure based on the theory of reversibility is similar to that of Sung and Kwon [4], and the procedure begins with the instance of infinite capacity buffer.

**Lemma 1.** In the output-level queue, if each output buffer has infinite capacity, then the steady-state probability is derived as

$$P(k_0, \ldots, k_n) = (1 - \rho)^{k_0} \frac{\delta_1 \delta_2 \cdots \delta_n}{k_0! k_1! k_2! \cdots k_n!} \cdot (1 - \rho^2), \quad P(\text{idle}) = 1 - \rho,$$

where $\rho = \sum k_j \lambda^j / \lambda$ and $\delta_j = \lambda^j / \sum \lambda^j$.

Moreover, it can be shown that the associated Markov process with states $(k_0, k_1, \ldots, k_n)$ and (idle) is reversible.

**Lemma 2.** In the output-level queue, if each output buffer has infinite capacity, then the Markov process is reversible.

The result of Lemma 2 provides the theoretical basis for treating the finite capacity buffer as a truncated one of infinite capacity buffer. It is because a reversible Markov process has the property that the truncation of its state space is equivalent to the truncation of its probability distribution. Thus, the steady-state probability of the output-level queue is simply derived as in Theorem 3 by using the Lemma 2 and reversibility (Kelly [1], Corollary I.10).

**Theorem 3.** In the output-level queue, if each output buffer has finite capacity, then the steady state probability is derived as

$$\Pi(k_0, \ldots, k_n) = \frac{P(k_0, \ldots, k_n)}{G} \text{ and } \Pi(\text{idle}) = \frac{P(\text{idle})}{G},$$

where $G = \sum_{k_0, \ldots, k_n} P(k_0, \ldots, k_n) + P(\text{idle})$, $A = \{(k_0, \ldots, k_n) | 0 \leq k_j \leq \infty \text{ and } 0 \leq k_i \leq \text{UB}_i + 1 \text{ for } i = 1, \ldots, n \}$.

**Proof.** The system has state space $\Psi = \{(\text{idle}), (k_2, \ldots, k_n) \in A \}$ which is truncated from the state space of infinite capacity buffer. And it can be seen that

$$G = \sum_{k_0, \ldots, k_n} P(k_0, \ldots, k_n) + P(\text{idle}) = \sum_{k_0=0}^{\infty} \sum_{k_1=0}^{\infty} \cdots \sum_{k_n=0}^{\infty} (1 - \rho) \cdot \frac{\delta_1 \delta_2 \cdots \delta_n}{k_0! k_1! k_2! \cdots k_n!} + (1 - \rho).$$
This completes the proof.

The input-level queue is composed of finite capacity buffer and a machine, and the service of each machine accommodates all the possible blocking delays that a unit might undergo due to the phenomenon of blocking. That is, the clearance time \( T_i \) at machine \( i \) is a convolution of a random number of exponentials.

Any blocked unit does not enter the output buffer until a unit departure occurs from the output buffer, which incurs a blocking time. This depends on sequences of arrived units at the output-level queue due to the FCFS service discipline of the AGV. Therefore, it is important to consider the probability of state \((I_1, \ldots, I_n)\) which is the sequence of blocked units where \( I_j \) denotes the index of \( j \)-th blocked machine, and derived from \( \Pi(k_2, \ldots, k_n) \).

**Lemma 4.** The probability of state \((I_1, \ldots, I_n)\) is derived as

\[
P(I_1, \ldots, I_n) = \frac{b_i}{\Pi(k_2, \ldots, k_n)}
\]

Let \( b_i \) be the probability that machine \( i \) is blocked by output-level queue with one unit occupying the machine, and \( \alpha_i(k) \) be the probability that there are \( OB_i \) units remaining at output buffer \( i \) and the first unit among them has been in the \( k \)-th waiting order for service by AGV), then they can be simply obtained as follows.

\[
b_i = \sum_{k=0}^{\infty} \sum_{k_2=0}^{\infty} \cdots \sum_{k_n=0}^{\infty} \Pi(k_2, \ldots, k_n) \cdot \alpha_i(k) \cdot \Pi(k_3, \ldots, k_n)
\]

where \( \mathcal{B} = \{(I_1, \ldots, I_n) | \ I_1 = OB, \ I_n = l, \ I_{n-j} = 1 \ for \ j = 1, \ldots, k-1 \} \).

And, let \( \beta_i(k) \) be the conditional probability that, upon service completion for a unit at machine \( i \), the unit sees \( OB_i \) units remaining at output buffer \( i \), among which the first unit has been in the \( k \)-th waiting order for service by AGV, then it can be computed as follows.

\[
\beta_i(k) = \frac{\alpha_i(k)}{1 - b_i}
\]

The expected clearance time of machine \( i \) can be computed from the value of \( \beta_i(k) \)

\[
E[T_i] = E[S_i] + \sum_{k=1}^{\infty} k \cdot \beta_i(k) \cdot E[S_i]
\]

The first term in equation (6) represents clearly the expected actual service time at machine \( i \), and the second term is the expected delay due to blocking. Then, the input-
level queue can be analyzed by this expected clearance time in the approach of the M/M/1/K queueing model.

The iterative procedure to obtain the true value of the effective input rate at each input-level queue is as follows (i = 1, ..., n):

Step 0. (set up) give the initial values of \( \lambda_i, \mu_i, \mu_g, IB_i, UB_i, \eta_i \)

Step 1. select starting values for \( \frac{\lambda_i}{\mu_i} \)

Step 2. find values of \( \lambda_i^* \) using the equation \( \lambda_i^* = \frac{\lambda_i}{\mu_i} \cdot (2 - \beta_i) \)

Step 3. compute \( \Pi_k(k) \) and \( \Pi(tidle) \) by using (2)

Step 4. compute of \( \phi_k(\eta_i), \beta_i(\eta_i) \) by using (3), (4), (5)

Step 5. compute \( \mathbb{E}[R_i] \) by using (6)

Step 6. compute \( \frac{\beta_i^*}{\phi_k(\eta_i)} \), \( 0 \leq k \leq IB_i + 1 \)

Step 7. calculate \( \frac{\lambda_i^*}{\mu_i} = \lambda_i \cdot [1 - \frac{\beta_i^*}{\phi_k(\eta_i)}] \), if \( \left| \frac{\lambda_i^*}{\mu_i} - \lambda_i \right| < \epsilon \) then stop, otherwise set \( \frac{\lambda_i^*}{\mu_i} = \frac{\lambda_i^*}{\mu_i} \), and go back to Step 2.

2.3 Computational Results

In order to test the accuracy of the approximation method, the system performance measures are computed for a variety of different systems which are described in parameter sets including parameter set 1 (\( \lambda = 2, \mu = 10, \mu_g = 3, IB_i = 3, \eta_i = 0.5 \)) and set 2 (\( \lambda = 2, \mu = 10, \mu_g = 3, \eta_i = 0.5 \)). The computational results are shown in Table 1 and 2 where TH, \( L_q \), UT, \( R_i \), \( \eta_i \) denote system throughput, mean queue length, steady state probability of the output buffer and input buffer respectively, and each table gives both approximation and simulation results.

<table>
<thead>
<tr>
<th>Problem description</th>
<th>( \lambda = 2, \mu = 10, \mu_g = 3, IB_i = 3, \eta_i = 0.5 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Size of output buffer</td>
<td>( OB = 1 )</td>
</tr>
<tr>
<td>( i = 1, 2 )</td>
<td>Appr.</td>
</tr>
<tr>
<td>( R_i(0) )</td>
<td>.8883</td>
</tr>
<tr>
<td>( R_i(1) )</td>
<td>.0767</td>
</tr>
<tr>
<td>( R_i(2) )</td>
<td>.0261</td>
</tr>
<tr>
<td>( R_i(3) )</td>
<td>.0089</td>
</tr>
<tr>
<td>( L_q )</td>
<td>.1555</td>
</tr>
<tr>
<td>( UT )</td>
<td>.3372</td>
</tr>
<tr>
<td>( R_i(0) )</td>
<td>.9445</td>
</tr>
<tr>
<td>( R_i(1) )</td>
<td>.0555</td>
</tr>
<tr>
<td>( R_i(2) )</td>
<td>.0078</td>
</tr>
<tr>
<td>( L_i )</td>
<td>.0555</td>
</tr>
<tr>
<td>( TH )</td>
<td>1.982</td>
</tr>
</tbody>
</table>

The accuracy of the approximation procedure, in general, depends on the magnitude of the blocking probability. If the probability that a unit will get blocked upon completion of merging queue is low, then the approximate results have negligible errors. As this blocking probability increases, deviation between the
approximate results and the exact or simulation data are observed. However, the solution algorithm is evaluated to be effective at an acceptable error rate. And, when a system with definite buffer size considered such as Automatic Pallet Changer system, increasing the system throughput enforces to extend the size of the input buffer more than that of the output buffer.

Table 2. Performance measure with parameter set 2 (\(s = 0.02\))

<table>
<thead>
<tr>
<th>Problem description</th>
<th>(\lambda = 2)</th>
<th>(\mu = 10)</th>
<th>(\gamma = 2), (\eta = 0.2)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Size of output buffer</td>
<td>IB, = 2, OB, = 1</td>
<td>IB, = 1, OB, = 2</td>
<td></td>
</tr>
<tr>
<td>(i = 1, 2)</td>
<td>Appr.</td>
<td>Simu.</td>
<td>error</td>
</tr>
<tr>
<td>(R_0(0))</td>
<td>.8964</td>
<td>.8958</td>
<td>.0006</td>
</tr>
<tr>
<td>(R_1(0))</td>
<td>.0773</td>
<td>.0779</td>
<td>.0006</td>
</tr>
<tr>
<td>(R_2(0))</td>
<td>.263</td>
<td>.263</td>
<td>.0000</td>
</tr>
<tr>
<td>(L_q)</td>
<td>.1299</td>
<td>.1310</td>
<td>.0011</td>
</tr>
<tr>
<td>UT</td>
<td>.3130</td>
<td>.3340</td>
<td>.0030</td>
</tr>
<tr>
<td>(R_0(1))</td>
<td>.9466</td>
<td>.9473</td>
<td>.0007</td>
</tr>
<tr>
<td>(R_1(1))</td>
<td>.0534</td>
<td>.0527</td>
<td>.0007</td>
</tr>
<tr>
<td>(R_2(1))</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(L_q)</td>
<td>.0534</td>
<td>.0580</td>
<td>.0046</td>
</tr>
<tr>
<td>TH</td>
<td>1.9350</td>
<td>1.9778</td>
<td>.0428</td>
</tr>
</tbody>
</table>

3 Concluding Remarks

In this paper, a queueing network model is considered for the performance analysis of an FMS composed of several workstations, each with both limited input and output buffers, where a single AGV is used for handling input and output materials. An efficient approximation algorithm is developed for obtaining the performance measures such as system throughput, machine utilization and blocking probability. In particular, since the steady state probability of the output-level queue is derived by using the reversibility rather than solving the associated conventional balance equations, the approximation method is very fast and the algorithm complexity is better than that of any other method. And, it is evaluated to be effective at acceptable error rate.

The proposed solution procedure can also be used to determine some optimal design configurations for computer systems, telecommunications, and manufacturing systems.

References