

# Analysis of Flexible Manufacturing Systems with Distinct Repeated Visits: DrQ

JEONGSEOB KIM

*School of Management, State University of New York at Binghamton, Binghamton, New York 13902*

PAUL J. SCHWEITZER

*William E. Simon Graduate School of Business Administration, University of Rochester, Rochester, New York 14627*

ABRAHAM SEIDMANN

*William E. Simon Graduate School of Business Administration, University of Rochester, Rochester, New York 14627*

**Abstract.** This article examines the performance effects caused by repeated part visits at the workstations of a flexible manufacturing system (FMS). Such repeated part visits to the same workstations are commonly associated with fixture changes for machining complex parts, reclamping, and remounting or reorienting them. Since each of the repeated visits to a workstation may require different processing requirements, the resulting queueing network does not have a product form solution. We therefore develop an approximate mean value analysis model for performance evaluation of an FMS that may produce multiple part types with distinct repeated visits.

We provide numerical examples and validate the accuracy of our solution algorithm against simulation. These examples show that the proposed model produces accurate throughput and utilization predictions with minimal computational efforts. These examples reveal that increasing the total pallet population may result in a reduction of the aggregate throughput, and that the FMS's performance could be more sensitive to the mix of pallets and part routes than to the total number of pallets. Our model will be of use, in particular, when managers wish to control individual operations (e.g., to adjust individual operation times to achieve economic savings in tool wear and breakage costs) or to investigate the performance implications of route changes due to alternate assignments of particular manufacturing tasks to certain workstations.

**Key Words:** queueing systems, performance evaluation, mean value analysis, capacity planning.

## 1. Introduction

The concept of flexible manufacturing systems (FMSs) emerged in the 1960s as decentralized computer control combined with the possible use of machine tools around the clock (Luggen, 1991). The improvement and increased availability of computers and machine tool technologies during the last two decades have accelerated the development of FMSs. This rapid evolution will continue as technology evolves, global competition intensifies, and, most importantly, the concept of flexible manufacturing gains wider use and acceptance.

There are many complex design and planning problems associated with implementing and managing an FMS. Since an FMS is a large and complex system with few human operators and mutually interacting components, with many parts competing for heavily utilized resources, efficient planning and management tools are required. Consider the following

scenario, which is abstracted from a real FMS machining high-precision fuel injection parts in a major automobile company we have studied:

The FMS supplies a family of part types to major automobile assembly plants in a Just-In-Time manner with long-term contracts. The supply contracts usually state that the assembly plants can ask the FMS to ship any number of each part type within a predetermined range, during the next period. Most of the part types that this FMS produces have complex geometries. The parts require reorientation, and they visit certain machines several times with different processing requirements at each visit.

An efficient planning method for producing part types with a varying production mix at minimum cost is critical for this FMS. In particular, it requires an efficient performance evaluation tool that takes into consideration these part types' distinct repeated visits to assess quickly the feasibility of a given production plan or evaluate the impact of changes in the part mix or job routes.

Closed queueing network (CQN) models are an efficient tool for the performance evaluation of FMSs.<sup>1</sup> These models assume that a fixed number of pallets circulate through the system according to prescribed routing requirements, and that new parts are introduced only as old ones depart, either because of physical constraints or as a matter of management policy. Solberg (1977) developed the first software package of a CQN model of FMSs, CAN-Q. The FMS is modeled as a product-form CQN, solved by an extension of Buzen's (1973) recursive algorithm. CAN-Q assumes that the processing times of all operations at each workstation are exponentially distributed with the same mean, which is required for the product-form solution algorithm. This even applies to the material handling system; all part movements must have the same distribution. If a workstation is visited several times by a part or by several part types with distinct processing times at each visit, then a weighted average processing time is taken over all visits. Thus, CAN-Q handles, in principle, only a single part type. If the FMS produces multiple part types, the *production ratios* must be specified so that a single average processing time can be computed at each workstation (Stecke, 1981). Suri and Hildebrant's (1984) MVAQ extends the CAN-Q model to explicitly consider multiple part types with different exponential processing times; it was implemented using an approximate mean value analysis scheme (Schweitzer, 1979; Bard, 1979), which requires very little computer memory. Solot and Bastos (1988) present a product-form multiclass CQN model, MULTIQ, extending CAN-Q to model multiple types of pallets. By comparing MULTIQ and CAN-Q through a numerical example and a case study, they show that the aggregation of different pallet types may cause an inaccuracy in predicting the system performance. Priority Mean Value Analysis (PMVA), developed by Shalev-Oren et al. (1985), can model the head-of-line priority discipline as well as the first-come-first-served (FCFS) and ample-server (AS) disciplines. Zhuang and Hindi (1991) analyze blocking in the material handling system of FMSs, which is caused by limited buffer capacities at workstations. Seidmann et al. (1987) provide a comparative evaluation of some computerized CQN models of FMSs. Buzacott and Yao (1986) outline several studies using queueing network models; they focus on identifying the major features of the models as they relate to the operational characteristics of FMSs and also discuss prescriptive methodologies for their design and operation. For comprehensive reviews see Askin and Standbridge (1993),

Buzacott and Shanthikumar (1993), Gray et al. (1993), and Viswanadham and Narahari (1992). All of the models described above ignore the effects of distinct repeated visits.

In many FMSs, a part makes repeated visits to a workstation for different operations, and each operation has its own processing time and may employ distinct tools. This is common in metal-cutting operations, where parts following different routes may have very different processing times each time they visit a particular workstation. Repeated visits may also result from rework tasks, part reclamping and remounting (after reorientation at the load/unload (L/UL) station), or from fixture changes. Modeling such repeated visit patterns is also essential for performance studies of those facilities in which the same pallets and fixtures are shared among several product types. The queueing network models of these systems lack a product form solution, because the FCFS workstations do not have the same mean processing time for all parts. Schweitzer and Seidmann (1989) and Schweitzer et al. (1991) propose an approximate-mean-value analysis model for a CQN having a *single* part type with distinct repeated visits. Schweitzer (1990) and Schweitzer and Akyildiz (1992) study a special CQN where customers change classes probabilistically as they move from server to server. The latter develop a set of fixed-point equations and a Newton-method-based algorithm to solve them; their method uses finite differences to estimate the Jacobian matrix. Although our systems can be formally modeled using this approach (treating each repeated visit as a new class of customers), we develop a new intuitively clear model that is capable of directly computing all the typical performance measures of FMSs. We call this model DrQ (distinct repeated-part-visit queueing network). Our model will be of use, in particular, when managers wish to control individual operations, e.g., to adjust individual operation times to achieve savings in tool wear and breakage costs or to investigate the operational implications of changes in the assignment of particular manufacturing tasks to certain workstations. We present several numerical examples indicating that our estimates of average throughput rates for different part types and of workstation utilizations are generally accurate when compared with simulation results. We find that conventional successive substitution often fails to solve our queueing network model; we provide a new solution algorithm that exploits some of the structural properties of our model. This algorithm does not require derivative information, and it is computationally very efficient. The computational efficiency of our solution algorithm and its ability to provide a broad spectrum of detailed performance measures make it an attractive building block for various system design or management studies (Kim et al., 1994).

Section 2 of this article develops an approximate-mean-value analysis of a CQN model of the particular FMSs under consideration. Section 3 provides both a numerical example and validation against simulation. Section 4 offers a summary of our work and suggests further developments and applications of our model.

## **2. A multiclass closed queueing network model**

This section first describes the features and operating characteristics of our FMSs. Section 2.2 develops an approximate-mean-value-analysis CQN model of them. Section 2.3 proposes an iterative heuristic algorithm to solve the model.

### 2.1. Assumptions and notations

We employ the usual assumptions about the configuration and operation of an FMS.<sup>2</sup>

- Each new part enters the FMS at the L/UL station, visits the various workstations, and when finished returns to the L/UL station for unloading and replacement by another new part of the same type.
- All workstations and transporters are reliable.
- Each workstation either has one server<sup>3</sup> with an FCFS service discipline (with unlimited queueing space), or is an AS (enough parallel servers so that a queue never develops).<sup>4</sup>
- For each part type, the first task at the L/UL station is the actual part load/unload; additional visits at that station are for the reorientation of the parts.
- The number of parts (or pallets) for each part type, which will also be referred to as the work-in-process level of the part type, is always a constant, so a closed queueing network model is required.
- Sufficient (central and distributed) buffer space is available to store all parts while they wait for the next workstation on their routes.
- The material handling system is treated as a central server. However, we allow distinct transport times for each movement between operations. The material handling system can be either an ample server or an FCFS server.

Model inputs and intermediary notations:

- $M$  = the number of workstations ( $M \geq 1$ ).
- $i \in \{1, 2, \dots, M\}$ : index for the workstations.
- FCFS = the set of FCFS workstations.
- AS = the set of ample servers (AS) workstations.
- $R$  = the number of part types ( $R \geq 1$ ).
- $r \in \{1, 2, \dots, R\}$ : index for the part types.
- $K_r$  = the number of pallets for part type  $r$  ( $K_r \geq 1, \forall r$ ).
- LUL = the L/UL station.
- $n(r, i)$  = the number of distinct operations on part type  $r$  performed by workstation  $i$ .  
 $n(r, i) = 0$  implies that part type  $r$  does not visit workstation  $i$ .
- $j \in \{1, 2, \dots, n(r, i)\}$ : index for the operation type, for each part type  $r$  at each workstation  $i$ .
- $(r, i, j)$  = triad index to denote the  $j$ th type of operation of part type  $r$  at workstation  $i$ . In particular,  $(r, LUL, 1)$  denotes the actual part load/unload.
- $s_{rij} \equiv$  mean processing time for workstation  $i$  to perform the  $j$ th type of operation of part type  $r$ ,  $1 \leq r \leq R$ ,  $1 \leq i \leq M$ ,  $1 \leq j \leq n(r, i)$ .  $s_{r,LUL,1}$  is the sum of load and unload time. (Ignore  $s_{rij}$  if  $n(r, i) = 0$ .)
- $v_{rij} \equiv$  expected number of times that a new type  $r$  part will visit workstation  $i$  for the  $j$ th type of operation. In particular,  $v_{r,LUL,1} = 1$ , since each part has only one actual load/unload operation.
- $v_{ri} \equiv \sum_{j=1}^{n(r,i)} v_{rij}$   
 = expected total number of times that a type  $r$  part will visit workstation  $i$  ( $= 0$  if type  $r$  parts do not visit workstation  $i$ ).

$R(i) \equiv \{r | n(r, i) > 0\}$   
 = the set of part types visiting workstation  $i$ . We assume that  $R(i)$  is nonempty for each  $i$ .

To see how to model distinct repeated visits using these notations, suppose a part of type 1 visits workstation 2 a total of 4 times: first for operation 1, then for operation 2, then for another operation 1, and finally for operation 3. This would be represented as

$n(1, 2) = 3$  (part type 1 has 3 types of operations at workstation 2)  
 $v_{121} = 2$  (part type 1 visits workstation 2 twice for operation 1)  
 $v_{122} = 1$  (part type 1 visits workstation 2 once for operation 2)  
 $v_{123} = 1$  (part type 1 visits workstation 2 once for operation 3)  
 $v_{12} = 4$  (part type 1 visits workstation 2 four times per part)

Note that this notation counts the *number* of visits but ignores the *sequence*, because the sequence does not affect the overall workload on the system.

The principal output parameters of the model are:

$\lambda \equiv (\lambda_1, \lambda_2, \dots, \lambda_R)$ , where  $\lambda_r$  is the throughput of part type  $r$ , measured at the L/UL station.  
 $\lambda_{rij}$  = throughput at workstation  $i$ , measured as the number of type  $j$  operations of type  $r$  parts per unit time. In particular,  $\lambda_r \equiv \lambda_{r,LUL,1}$  = the system throughput of part type  $r$ .  
 $W_{rij}$  = mean sojourn time of parts of type  $r$  at workstation  $i$ , either in queue or in service, for one type  $j$  operation.  
 $N_{rij}$  = mean number of parts of type  $r$  at workstation  $i$ , either in queue or in service, for a type  $j$  operation.  
 $Q_{ri}$  = mean queueing time of a part of type  $r$  at workstation  $i \in FCFS$ .

## 2.2. An approximate-mean-value CQN model: DrQ

The approximate-mean-value-analysis CQN model of the FMSs under consideration is given by:

$$\lambda_{rij} = v_{rij} \lambda_r, \quad 1 \leq i \leq M, r \in R(i), 1 \leq j \leq n(r, i) \quad (1)$$

$$\lambda_r = \frac{K_r}{\sum_{i=1}^M \sum_{j=1}^{n(r,i)} v_{rij} W_{rij}}, \quad 1 \leq r \leq R \quad (2)$$

$$W_{rij} = s_{rij}, \quad \forall i \in AS, r \in R(i), 1 \leq j \leq n(r, i) \quad (3)$$

$$W_{rij} = s_{rij} + Q_{ri}, \quad \forall i \in FCFS, r \in R(i), 1 \leq j \leq n(r, i) \quad (4)$$

$$Q_{ri} = \sum_{p \in R(i)} \sum_{j=1}^{n(p,i)} N_{pij} s_{pij} \left[ 1 - \frac{\delta_{pr}}{K_r} \right], \quad \forall i \in FCFS, r \in R(i) \quad (5)$$

$$N_{rij} = \lambda_{rij} W_{rij}, \quad 1 \leq i \leq M, r \in R(i), 1 \leq j \leq n(r, i) \quad (6)$$

(We ignore equations where  $n(r, i) = 0$  and treat the sum over an empty set as zero; for example, if  $n(r, i) = 0$ , then  $\sum_{j=1}^{n(r,i)} = 0$ . The notation  $\delta_{pr}$  in (5) is the Kronecker delta.) Equations (1)–(6) comprise a fixed-point problem and are guaranteed to have at least one nonnegative solution by the Brouwer Fixed-Point Mapping Theorem, provided that every  $K_r \geq 1$ .

Equation (1) expresses the number of type  $j$  operations of part type  $r$  per unit time at workstation  $i$  as equal to the throughput  $\lambda_r$  ( $\equiv \lambda_{r,LUL,1}$ ) multiplied by the visit ratio  $v_{rij}$  (recall that  $v_{r,LUL,1} = 1$ ). By Little's law, (2) expresses the throughput  $\lambda_r$  of part type  $r$  as the part population  $K_r$  divided by the mean flow time. From (1) and (2) follow the population count for each part type  $r$ :

$$\sum_{i=1}^M \sum_{j=1}^{n(r,i)} \lambda_{rij} W_{rij} = K_r, \quad 1 \leq r \leq R. \quad (7)$$

In (7), the term  $\lambda_{rij} W_{rij}$  is, by Little's law, the mean number of type  $r$  parts at workstation  $i$  for the  $j$ th type of visit,  $N_{rij}$ , which is expressed by (6).

Both (1) and (2) are exact. Equation (3) equates sojourn times with service times for ample servers, and it is also exact. In (4), the mean sojourn time  $W_{rij}$  at an FCFS server is expressed as the sum of the mean service time  $s_{rij}$  and the mean queueing time  $Q_{ri}$ . The latter is estimated in (5) as the ergodic work backlog  $\sum_{p \in R(i)} \sum_{j=1}^{n(p,i)} N_{pij} s_{pij}$  with a  $(K_r - \delta_{pr})/K_r$  correction to preclude queueing for itself. This is the usual mean-value-analysis approximation for FCFS servers, which is invariant under operation splitting (Schweitzer, 1979; Schweitzer et al. 1986). It replaces the expected residual processing time with the expected full process time, which is exact for exponentially distributed processing times.

We can transform the above formulation (1)–(6) into a simpler form. First, using equations (1), (3), and (4), we replace (2) with

$$\begin{aligned} K_r &= \lambda_r \left[ \sum_{i=1}^M \sum_{j=1}^{n(r,i)} v_{rij} W_{rij} \right] \\ &= \lambda_r \left[ \sum_{i \in AS} \sum_{j=1}^{n(r,i)} v_{rij} s_{rij} + \sum_{i \in FCFS} \sum_{j=1}^{n(r,i)} v_{rij} (s_{rij} + Q_{ri}) \right] \end{aligned}$$

$$\begin{aligned}
&= \lambda_r \left[ \sum_{i=1}^M \sum_{j=1}^{n(r,i)} v_{rij} s_{rij} + \sum_{i \in FCFS} v_{ri} Q_{ri} \right] \\
&= \lambda_r \left[ E_r + \sum_{i \in FCFS} v_{ri} Q_{ri} \right], \quad 1 \leq r \leq R, \tag{8}
\end{aligned}$$

where the constant  $E_r$  is

$$E_r \equiv \sum_{i=1}^M \sum_{j=1}^{n(r,i)} v_{rij} s_{rij}, \quad 1 \leq r \leq R. \tag{9}$$

Once the ergodic waiting time  $Q_{ri}$  is expressed as a function of  $\lambda$ , say  $Q_{ri} \equiv Q_{ri}(\lambda)$ ,  $W_{rij}$  in equation (4) and in turn  $N_{rij}$  in equation (6) can be expressed as functions of  $\lambda$ . Thus, once the functions  $Q_{ri}(\lambda)$  are available, solving (1)–(6) for  $\{Q_{ri}, W_{rij}, \lambda_{rij}, N_{rij}\}$  is equivalent to solving the system of nonlinear equations (8) for  $\lambda$ . We next develop the desired functions  $Q_{ri}(\lambda)$  using the following lemma.<sup>5</sup>

**Lemma 1.** *The ergodic work backlog  $Q_{ri}$  of the formulation (1)–(6) can be expressed as a function of the throughput vector  $\lambda$ :*

$$Q_{ri}(\lambda) = \frac{1}{1 + a_{ri} \lambda_r / K_r} \left[ \frac{\sum_{p \in R(i)} b_{pi} \lambda_p / (1 + a_{pi} \lambda_p / K_p)}{1 - \sum_{p \in R(i)} a_{pi} \lambda_p / (1 + a_{pi} \lambda_p / K_p)} - \frac{b_{ri} \lambda_r}{K_r} \right], \tag{10}$$

where

$$a_{ri} \equiv \sum_{j=1}^{n(r,i)} v_{rij} s_{rij}, \tag{11}$$

$$b_{ri} \equiv \sum_{j=1}^{n(r,i)} v_{rij} (s_{rij})^2. \tag{12}$$

Note that all variables  $\lambda_{rij}$ ,  $W_{rij}$ , and  $N_{rij}$  have disappeared in (10), so the  $Q$ 's depend only upon  $\lambda$ .

### 2.3. Structural properties and a solution algorithm

If we express the throughput rate of part type  $r$  from (8) as

$$\lambda_r = \frac{K_r}{E_r + \sum_{i \in FCFS} v_{ri} Q_{ri}(\lambda)}, \quad 1 \leq r \leq R, \tag{13}$$

we can see that this is a fixed-point equation of  $\lambda$ , and the solvability of (1)–(6) assumes it has at least one nonnegative solution, provided that each  $K_r \geq 1$ . However, successive substitutions on (13) will not converge if  $1 - \sum_{r \in R(i)} \lambda_r a_{ri} / (1 + a_{ri} \lambda_r / K_r) > 0$  is violated for any  $i \in FCFS$ , since  $Q_{ri}$  may become negative and therefore produce meaningless results. Instead, we develop a multidimensional search algorithm to find the throughput  $\lambda$ . Our method involves univariate relaxation on the most violated equation, with precautions to keep every  $Q_{ri} \geq 0$ .

We denote the right-hand side of (8) by  $\hat{K}_r(\lambda)$ . Solving (1)–(6) for  $\lambda$  is equivalent to solving the following system of nonlinear equations:

$$K_r = \hat{K}_r(\lambda), \quad 1 \leq r \leq R. \tag{14}$$

We suppose that (14) has a unique solution. Our algorithm starts with an initial guess and iterates until it results in a sufficiently small relative error for every part type. The relative error is defined by:

$$\Delta_r(\lambda) \equiv \frac{\hat{K}_r(\lambda)}{K_r} - 1 = \frac{\lambda_r}{K_r} \left[ E_r + \sum_{i \in FCFS} v_{ri} Q_{ri}(\lambda) \right] - 1, \quad 1 \leq r \leq R. \tag{15}$$

The algorithm starts with a nonnegative initial  $\lambda$ , which is feasible in the sense that it satisfies  $1 - \sum_{r \in R(i)} \lambda_r a_{ri} / (1 + a_{ri} \lambda_r / K_r) > 0, \forall i \in FCFS$ . To obtain such a  $\lambda$ , we use the fact that  $\lambda_r = K_r / E_r$  is the theoretical upper bound on the throughput of part type  $r$ , which is binding if every workstation is an ample server. If not all workstations are ample servers, we start with every  $\lambda_r = K_r / E_r$  and decrease  $\lambda$  until the condition,  $1 - \sum_{r \in R(i)} \lambda_r a_{ri} / (1 + a_{ri} \lambda_r / K_r) > 0, \forall i \in FCFS$ , is met.

At each iteration, with the current solution  $\lambda$ , the algorithm picks a part type  $r$  such that  $r = \operatorname{argmax}_{1 \leq p \leq R} \{|\Delta_p(\lambda)|\}$  and solves the single equation  $\delta_r(\lambda_r) = 0$ , where  $\delta_r(\lambda_r)$  is defined by

$$\delta_r(\lambda_r) \equiv \Delta_r(\lambda), \quad \text{with all } \lambda_p, p = 1, 2, \dots, R, p \neq r, \text{ fixed.} \tag{16}$$

The solution is then used to update the  $r$ th component of  $\lambda$ . We developed an efficient numerical algorithm to solve  $\delta_r(\lambda_r) = 0$  using the following two theorems:<sup>6</sup>

**Theorem 1.** *At any given  $\lambda$  such that  $1 - \sum_{p \in R(i)} \lambda_p a_{pi} / (1 + a_{pi} \lambda_p / K_p) > 0$ , the following holds:*

$$\lambda_r \leq \bar{\lambda}_r \equiv \min_{1 \leq i \leq M, r \in R(i)} \frac{C_2 - \sqrt{C_2^2 - 4C_1 K_r}}{2C_1}, \quad 1 \leq r \leq R, \tag{17}$$

where

$$C_1 \equiv E_r (1 - 1/K_r) a_{ri},$$

$$C_2 \equiv E_r + (K_r - 1) a_{ri} + v_{ri} \left[ \sum_{p \in R(i)} b_{pi} \lambda_p - \frac{b_{ri} \lambda_r}{K_r} \right].$$

Table 1. Algorithm DrQ.

---

Step 0.	{Initialization}
	<ul style="list-style-type: none"> <li>• Compute <math>a_{ri}, b_{ri}, v_{ri}, E_r, \forall r, i</math>;</li> <li>• Find a feasible <math>\lambda</math>;</li> <li>• <math>\text{iterCount} := 0</math>;</li> </ul>
Step 1.	{Loop Step}
	<ul style="list-style-type: none"> <li>• <math>r := \text{argmax}_{1 \leq p \leq R} \{ \Delta_p(\lambda) \}</math>;</li> <li>• <math>\text{maxError} :=  \Delta_r(\lambda) </math>;</li> <li>• if <math>(\text{maxError} &lt; \text{stopTol})^\dagger</math> or <math>(\text{iterCount} &gt; \text{maxIter})^\ddagger</math> go to Step 3;</li> </ul>
Step 2.	{Iteration}
	<ul style="list-style-type: none"> <li>• <math>\text{iterCount} := \text{iterCount} + 1</math>;</li> <li>• Compute <math>\bar{\lambda}_r</math>;</li> <li>• Solve <math>\delta_r(\bar{\lambda}_r) = 0</math>;</li> <li>• go to Step 1;</li> </ul>
Step 3.	{End}
	<ul style="list-style-type: none"> <li>• Print performance measures;</li> <li>• Print fail if <math>(\text{maxError} &gt; \text{stopTol})</math> or <math>(\text{iterCount} &gt; \text{maxIter})</math>.</li> </ul>

---

$^\dagger$ stopTol is a small tolerance limit, for example, 0.001.

$^\ddagger$ maxIter is a limit on the number of iterations, for example, 100.

**Theorem 2.**  $\hat{K}_r(\lambda)$  is convex increasing in  $\lambda_r$ .

Theorem 2 implies that  $\delta_r(\lambda_r)$  is a convex increasing function of  $\lambda_r$  with  $\delta_r(0) = -1$ . From (15) it is clear that  $\delta_r(\lambda_r)$  is positive at  $\lambda_r = K_r/E_r$  (assuming that it does not violate the condition,  $1 - \sum_{p \in R(i)} \lambda_p a_{pi} / (1 + a_{pi} \lambda_p / K_p) > 0$ ). This monotonicity property and the expected change in the sign of  $\delta_r(\lambda_r)$  over a finite range suggest an efficient numerical method.

We use Theorem 1 to get an upper bound  $\bar{\lambda}_r$  at which  $\delta(\cdot) \geq 0$ . We begin with every  $\lambda_r = K_r/E_r$ ; if this violates the condition that  $1 - \sum_{p \in R(i)} \lambda_p a_{pi} / (1 + a_{pi} \lambda_p / K_p) > 0$ ,  $\forall i \in FCFS$ , we decrease it by small amounts until the condition is satisfied. We then use (17) to compute  $\bar{\lambda}_r$ . If this  $\bar{\lambda}_r$  does not guarantee that  $\delta_r(\bar{\lambda}_r) > 0$ , we set  $\bar{\lambda}_r$  to the feasible  $\lambda_r$  just found. A formal description of the algorithm is given in table 1. Our algorithm is not theoretically guaranteed to converge, because driving  $\delta_r(\lambda_r)$  to zero may increase some other residual errors. In practice, it was robust and always worked. The maximal error residual was strictly monotone down to zero. The software code implementing this algorithm is available from the authors upon request.

### 3. Numerical examples

**Data Set 1: Three part types and six workstations.** Consider an FMS that produces three part types. It consists of one load/unload station {L/UL}, four machining centers {MC1, MC2, MC3, MC4}, an inspection station {INS}, and an AGV. Initially, ten pallets are allocated to each part type. Figure 1 is a logical diagram of this FMS; it also shows the routes of the three part types. For each part type, an operation at a workstation is indicated

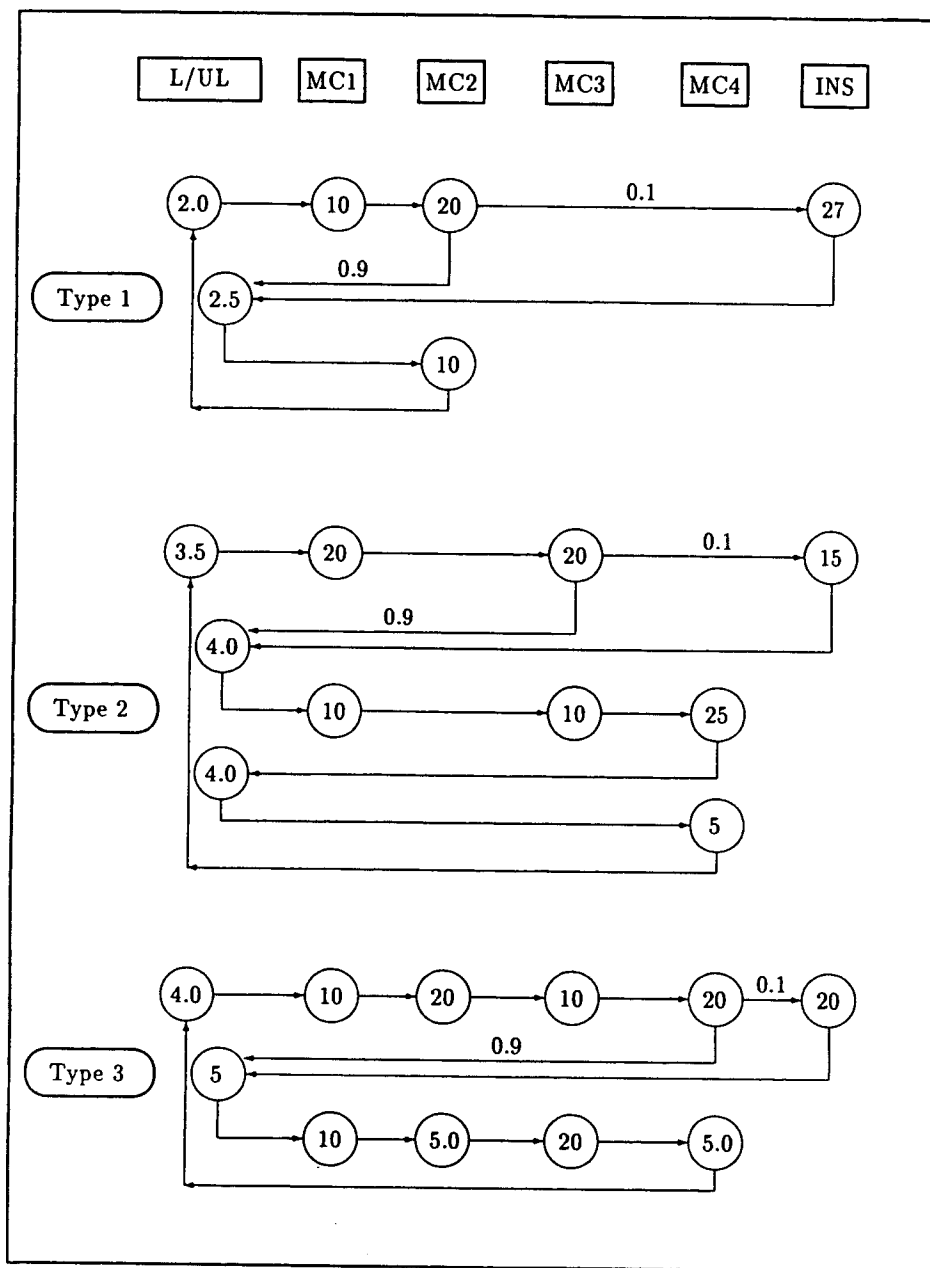


Figure 1. Part routes in the sample FMS: Data Set 1.

by a circle. The number inside the circle gives the mean processing time in minutes. The arrows show the movements of the parts between workstations. The fractional numbers on the lines show the probability that a part is moved to the next station linked by the line; all movements linked by lines without a probability are assumed to be deterministic. The mean AGV travel time is 30 seconds per transfer. For example, part type 1 is moved by the AGV from L/UL to MC1. It needs ten minutes of processing there before going to

MC2 for 20 minutes of processing; the part may then be sent to the inspection station 10% of the time, and so forth. Part type 1 visits MC2 twice, first for 20 minutes and then (following reorientation at L/UL) for ten more minutes.

The case with  $K = (10, 10, 10)$  was solved in 22 iterations in 0.11 seconds on an IBM-PC compatible computer with a 66 MHz Intel 486 DX2 CPU. The accuracy of the algorithm was validated by ten replications of a simulation of an operation lasting 6,000 minutes (6,000 minutes is equivalent to a five-day week with a 20-hour day). We performed several experiments with different pallet mixes. Tables 2 and 3 summarize the results in terms of throughputs and utilizations, respectively. Table 4 compares the approximation and simulation with respect to mean queue length when  $K = (10, 10, 10)$ . In these tables, "DrQ" and "SIM" indicate the corresponding values obtained by DrQ and the simulation, respectively. "% Error" is the percentage relative error, defined by

$$\% \text{ Error} \equiv 100 \times \frac{\text{DrQ} - \text{SIM}}{\text{SIM}} (\%). \quad (18)$$

The approximations for throughputs and utilizations are fairly good: the largest error was close to 13%, but most estimates were within  $\pm 5\%$  of the simulation's results. These results also indicate that an FMS's performance may be more sensitive to the mix of pallets than to the total number of pallets. For instance, more than doubling the pallet population

Table 2. Comparison of DrQ and the simulation for throughput: Data Set 1.

Pallet Mix	Part Type 1			Part Type 2			Part Type 3		
	DrQ	SIM	% Error	DrQ	SIM	% Error	DrQ	SIM	% Error
(7, 7, 7)	1.319	1.361	-3.1	0.946	0.857	10.4	0.637	0.618	3.0
(10, 10, 10)	1.331	1.386	-4.0	0.996	0.974	2.3	0.667	0.658	1.3
(15, 15, 15)	1.335	1.415	-5.6	1.030	1.021	0.9	0.696	0.702	-0.9
(10, 7, 15)	1.146	1.220	-6.1	0.787	0.732	7.6	0.941	0.929	1.3
(7, 15, 10)	1.167	1.172	-0.4	1.115	1.112	0.3	0.632	0.628	0.5
(15, 10, 7)	1.570	1.617	-3.0	1.084	1.061	2.1	0.435	0.435	0.1

Table 3. Comparison of DrQ and the simulation for utilization: Data Set 1.

Workstation	Pallet Mix								
	(7, 7, 7)			(10, 10, 10)			(15, 15, 15)		
	DrQ	SIM	% Error	DrQ	SIM	% Error	DrQ	SIM	% Error
L/UL	0.376	0.357	5.1	0.391	0.393	-0.5	0.402	0.416	-3.4
MC1	0.905	0.867	4.3	0.942	0.927	1.7	0.970	0.963	0.7
MC2	0.925	0.960	-3.7	0.943	0.978	-3.5	0.958	0.983	-2.6
MC3	0.791	0.768	3.1	0.832	0.830	0.2	0.863	0.867	-0.5
MC4	0.738	0.705	4.7	0.776	0.754	3.0	0.805	0.815	-1.2
INS	0.104	0.092	12.8	0.107	0.112	-4.3	0.109	0.121	-10.1
AGV	0.181	0.175	3.7	0.188	0.187	0.5	0.194	0.196	-1.3

Table 4. Comparison of DrQ and the simulation for mean queue length: Data Set 1.

Workstation	Pallet Mix								
	(7, 7, 7)			(10, 10, 10)			(15, 15, 15)		
	DrQ	SIM	% Error	DrQ	SIM	% Error	DrQ	SIM	% Error
L/UL	0.226	0.269	-16.2	0.255	0.319	-20.0	0.280	0.438	-36.0
MC1	5.954	5.870	1.4	10.078	9.566	5.4	18.339	16.569	10.7
MC2	6.079	5.835	4.2	8.923	8.737	2.1	13.225	13.264	-0.3
MC3	2.412	2.337	3.2	3.445	3.609	-4.5	4.796	4.877	-1.7
MC4	2.257	2.712	-16.8	3.062	3.527	-13.2	3.990	5.416	-26.3
INS	0.012	0.011	14.0	0.013	0.016	-16.2	0.014	0.021	-34.4
AGV	0.038	0.040	-6.4	0.042	0.046	-9.8	0.045	0.053	-15.6

from (7, 7, 7) to (15, 15, 15) increased the respective throughput of part types 1, 2, and 3 by only 1.2%, 8.8%, and by 9.2%, respectively. The smallest throughput increment was observed for part type 1. This is no surprise, because it mainly uses workstations MC1 and MC2, with both of them running at above 90% utilization. On the other hand, changing the mix from (10, 7, 15) to (7, 15, 10) while keeping the total number of pallets constant had a more pronounced effect on the FMS's performance. The throughput of part types 1, 2, and 3 changed by 1.8%, 41.6%, and -32.8%, respectively. In addition, it is clear that the aggregate performance is *not* monotone in the total number of pallets. Increasing the total pallet population from (7, 7, 7), or  $\sum_{r=1}^3 K_r = 21$ , to (10, 7, 15), or  $\sum_{r=1}^3 K_r = 32$ , results in a *reduction* in the aggregate throughput  $\sum_{r=1}^3 \lambda_r$  from 2.902 parts/hour to 2.874 parts/hour.<sup>7</sup> It is interesting to note the pronounced and somewhat counterintuitive interaction effects across part types. Changing the pallet population from (7, 7, 7) to (10, 7, 15) reduced throughput of part type 1 by 13.1%, despite a 43% increase in the number of pallets allocated to that part type.

**Data Set 2: Two part types and four workstations.** In this data set, we consider an FMS in which a part may visit a workstation more than once for rework due to quality problems. The routes of the two part types and their processing times are given in figure 2. The mean AGV travel time is 30 seconds per transfer. Every type 1 part is inspected, and 20% of them are reworked at MILL and then sent to the L/UL. Thus, on average, a type 1 part visits MILL 1.2 times before it leaves the system. Tables 5, 6, and 7 summarize the results of the experiments.

We observe results similar to those for data set 1. DrQ is very accurate in predicting throughputs and utilization, with less than 2% error in all six pairs of pallet mixes except in one case, namely, the utilization of INS when the pallet mix is (15, 15). The aggregate throughput with pallet mix (15, 7), 11.689 parts/hour, is higher than the one with mix (10, 20), 11.516 parts/hour, even though the latter has a larger total number of pallets.

From extensive experiments which we have done, DrQ seems to provide good approximations for the throughputs and the workstation utilization. DrQ is generally accurate in predicting mean queue length when the queues are relatively long, but it does have trouble when the queues are short, as seen in tables 4 and 7. These prediction errors are not, however,

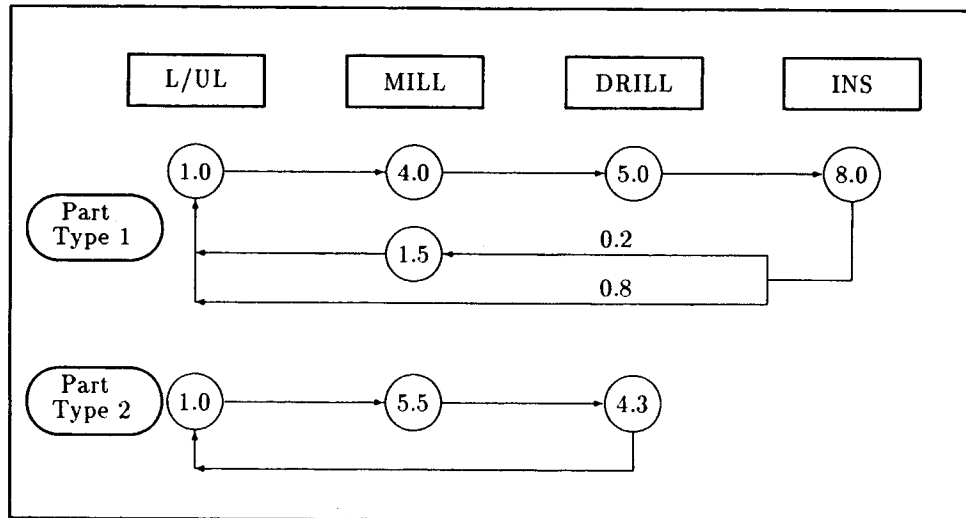


Figure 2. Part routes in the sample FMS: Data Set 2.

Table 5. Comparison of DrQ and the simulation for throughput: Data Set 2.

Pallet Mix	Part Type 1			Part Type 2		
	DrQ	SIM	% Error	DrQ	SIM	% Error
(7, 7)	4.688	4.713	-0.5	6.604	6.597	0.1
(10, 10)	4.920	4.931	-0.2	6.690	6.751	-0.9
(15, 15)	5.115	5.180	-1.3	6.720	6.785	-1.0
(20, 20)	5.212	5.292	-1.5	6.719	6.783	-0.9
(15, 7)	6.584	6.626	-0.6	5.105	5.092	0.3
(7, 20)	2.475	2.505	-1.2	8.865	8.921	-0.6
(10, 20)	3.274	3.315	-1.2	8.242	8.291	-0.6

Table 6. Comparison of DrQ and the simulation for utilization: Data Set 2.

Workstation	Pallet Mix								
	(7, 7)			(10, 10)			(15, 15)		
	DrQ	SIM	% Error	DrQ	SIM	% Error	DrQ	SIM	% Error
L/UL	0.188	0.188	-0.1	0.194	0.196	-1.4	0.197	0.200	-1.4
MILL	0.941	0.954	-1.3	0.966	0.971	-0.5	0.983	0.986	-0.4
DRILL	0.864	0.871	-0.8	0.890	0.898	-0.9	0.908	0.912	-0.5
INS	0.625	0.626	-0.2	0.656	0.661	-0.7	0.682	0.700	-2.5
AGV	0.329	0.331	-0.4	0.340	0.342	-0.8	0.347	0.352	-1.4

a significant flaw, since workstations with short queues (and thus short queueing delays) have relatively little effect on total job flow times. We did not find a significant correlation between the number of pallets in use and the magnitude of the estimation error.

Table 7. Comparison of DrQ and the simulation for mean queue length: Data Set 2.

Workstation	Pallet Mix								
	(7, 7)			(10, 10)			(15, 15)		
	DrQ	SIM	% Error	DrQ	SIM	% Error	DrQ	SIM	% Error
L/UL	0.040	0.043	-8.5	0.044	0.051	-14.8	0.046	0.055	-14.9
MILL	6.629	6.369	4.1	10.946	10.115	8.2	18.946	17.654	7.3
DRILL	3.512	3.817	-8.0	4.867	5.625	-13.5	6.535	7.788	-16.1
INS	0.721	0.639	12.9	0.946	0.956	-1.1	1.194	1.156	3.3
AGV	0.145	0.162	-10.5	0.162	0.184	-12.2	0.175	0.197	-10.9

#### 4. Conclusions

This article describes a newly developed approximate Mean Value analysis of an FMS with multiple part types. Our work extends many of the previous studies in that we analyze the dynamic effects caused by *repeated* part visits for *distinct* operations at the various workstations. This feature is of particular significance for flexible manufacturing systems, as the current practice is to assign a large number of production operations and rework tasks to a single machine in order to minimize transportation and queueing delays (Kusiak, 1986). We note that successive substitution algorithms can fail to solve the fixed-point model and present an alternative solution framework. Our new framework, which exploits the special structure of the stochastic parts flow model, leads to efficient and robust computational properties. The enhanced level of modeling details and the computational efficiency of our solution algorithm enable users to explore the major performance implications associated with a large variety of managerial decision options. For example, users can assess the impact of changes in the mix of pallets, in part routes, or in the allocation of tasks to machines, on the contention for primary system resources. Our performance evaluation framework may also be incorporated with a nonlinear optimization scheme that calculates the minimum cost processing rates, subject to some throughput constraints. Finally, several numerical examples are included to illustrate the accuracy of our model. We show that our model exhibits small relative errors for the throughput of all part types, and for the utilization of the workstations. These relative errors are not correlated with the number of pallets being used. Our numerical examples also show that the aggregate throughput is not monotone with the total number of pallets. In addition, we show why in practice an FMS's performance tends to be more sensitive to changes in the pallet mix than to the total number of pallets.

One possible direction of future research would be to explicitly characterize the output process of the individual workstations. With a better estimate of the departure process, we should be able to approximate more closely the moments of the queue length in front of the system's bottlenecks. A second direction of future research would be to approximate the higher moments of the residual service times at load-dependent workstations. Other extensions would be to production systems with parallel, nonidentical machines and priority service disciplines. Also in progress is research on selecting processing times to minimize operating costs, subject to throughput constraints.

**Appendix: Proofs of lemma and theorems***A.1. Proof of Lemma 1*

Using equations (1), (4), and (6), for  $i \in FCFS$ ,  $r \in R(i)$ , we rewrite (5) as follows

$$\begin{aligned} Q_{ri} &= \sum_{p \in R(i)} \sum_{j=1}^{n(p,i)} N_{pij} s_{pij} \left[ 1 - \frac{\delta_{pr}}{K_r} \right] \\ &= \sum_{p \in R(i)} \lambda_p \sum_{j=1}^{n(p,i)} v_{pij} (s_{pij} + Q_{pi}) s_{pij} \left[ 1 - \frac{\delta_{pr}}{K_r} \right]. \end{aligned} \quad (\text{A.1})$$

We rewrite (A.1) as

$$Q_{ri} = \xi_{ri} + \eta_i - \lambda_r a_{ri} Q_{ri} / K_r, \quad (\text{A.2})$$

where

$$\xi_{ri} \equiv \zeta_i - \frac{b_{ri} \lambda_r}{K_r}, \quad (\text{A.3})$$

$$\eta_i \equiv \sum_{p \in R(i)} a_{pi} \lambda_p Q_{pi}, \quad (\text{A.4})$$

and

$$\zeta_i \equiv \sum_{p \in R(i)} b_{pi} \lambda_p. \quad (\text{A.5})$$

By rearranging (A.2), we get

$$Q_{ri} = \frac{\xi_{ri} + \eta_i}{\tau_{ri}}, \quad (\text{A.6})$$

where

$$\tau_{ri} \equiv 1 + \frac{a_{ri}}{K_r} \lambda_r. \quad (\text{A.7})$$

We insert (A.6) into (A.4) to get an explicit expression for  $\eta_i$ :

$$\begin{aligned} \eta_i &= \sum_{p \in R(i)} a_{pi} \lambda_p (\xi_{pi} + \eta_i) / \tau_{pi} \\ &= \frac{\sum_{p \in R(i)} a_{pi} \lambda_p \xi_{pi} / \tau_{pi}}{1 - \sum_{p \in R(i)} a_{pi} \lambda_p / \tau_{pi}}. \end{aligned} \quad (\text{A.8})$$

We let  $D_i$  denote the denominator of the right side of (A.8) and insert (A.3) into (A.8) to simplify (A.8) as follows

$$\begin{aligned}
\eta_i &= \frac{1}{D_i} \sum_{p \in R(i)} \frac{a_{pi} \lambda_p}{\tau_{pi}} \left( \zeta_i - \frac{b_{pi} \lambda_p}{K_p} \right) \\
&= \frac{1}{D_i} \left[ \zeta_i \sum_{p \in R(i)} \frac{a_{pi} \lambda_p}{\tau_{pi}} - \sum_{p \in R(i)} \frac{a_{pi} \lambda_p}{K_p} \frac{b_{pi} \lambda_p}{\tau_{pi}} \right] \\
&= \frac{1}{D_i} \left[ \zeta_i (1 - D_i) - \sum_{p \in R(i)} (\tau_{pi} - 1) \frac{b_{pi} \lambda_p}{\tau_{pi}} \right] \\
&= \frac{1}{D_i} \left[ \zeta_i (1 - D_i) - \sum_{p \in R(i)} b_{pi} \lambda_p + \sum_{p \in R(i)} \frac{b_{pi} \lambda_p}{\tau_{pi}} \right] \\
&= -\zeta_i + \frac{1}{D_i} \sum_{p \in R(i)} \frac{b_{pi} \lambda_p}{\tau_{pi}}. \tag{A.9}
\end{aligned}$$

We then insert equations (A.3), (A.7), and (A.9) into (A.6) to get the desired form (10).  $\square$

## A.2. Proof of Theorem 1

Before proving Theorem 1, we introduce the following lemma.

**Lemma A.1.** *At a given point  $\lambda$  such that  $1 - \sum_{p \in R(i)} \lambda_p a_{pi} / \tau_{pi} > 0$ , the following inequality holds:*

$$\lambda_r \leq \frac{K_r}{E_r + \frac{\nu_{ri} \xi_{ri}}{1 - (1 - 1/K_r) a_{ri} \lambda_r}}, \quad r = 1, 2, \dots, R, r \in R(i), i \in FCFS. \tag{A.10}$$

*Proof:* From (A.8) we get

$$\begin{aligned}
\eta_i &= \frac{\sum_{p \in R(i)} a_{pi} \lambda_p \xi_{pi} / \tau_{pi}}{1 - \sum_{p \in R(i)} a_{pi} \lambda_p / \tau_{pi}} \\
&\geq \frac{a_{ri} \lambda_r \xi_{ri} / \tau_{ri}}{1 - a_{ri} \lambda_r / \tau_{ri}} \\
&= \frac{a_{ri} \lambda_r \xi_{ri}}{\tau_{ri} - a_{ri} \lambda_r}. \tag{A.11}
\end{aligned}$$

Inserting the relation (A.11) into (A.6) gives

$$Q_{ri} \geq \frac{1}{\tau_{ri}} \left[ \xi_{ri} + \frac{a_{ri}\lambda_r\xi_{ri}}{\tau_{ri} - a_{ri}\lambda_r} \right] \quad (\text{A.12})$$

$$= \frac{\xi_{ri}}{1 - (1 - 1/K_r)a_{ri}\lambda_r}. \quad (\text{A.13})$$

Inserting the relation (A.13) into (8) and rearranging it gives:

$$\lambda_r \leq \frac{K_r}{E_r + \sum_{i \in FCFS} \frac{v_{ri}\xi_{ri}}{1 - (1 - 1/K_r)a_{ri}\lambda_r}}, \quad (\text{A.14})$$

which implies (A.10).  $\square$

We can now prove Theorem 1. First, we rewrite (A.10) as a quadratic inequality as follows:

$$C_1\lambda_r^2 - C_2\lambda_r + K_r \geq 0. \quad (\text{A.15})$$

Since  $C_1 > 0$ , the solution to (A.15) is either

$$\lambda_r \leq \frac{C_2 - \sqrt{C_2^2 - 4C_1K_r}}{2C_1} \quad (\text{A.16})$$

or

$$\lambda_r \geq \frac{C_2 + \sqrt{C_2^2 - 4C_1K_r}}{2C_1} \quad (\text{A.17})$$

We next show that (A.17) is infeasible. First, from the condition of the lemma we obtain

$$1 > \sum_{p \in R(i)} \lambda_p a_{pi} / \tau_{pi} \geq \lambda_r a_{ri} / \tau_{ri}, \quad r \in R(i). \quad (\text{A.18})$$

From (A.18) follows

$$\lambda_r < \frac{1}{(1 - 1/K_r)a_{ri}}, \quad r \in R(i). \quad (\text{A.19})$$

By some algebra we can show that

$$C_2^2 - 4C_1K_r > [E_r - (K_r - 1)a_{ri}]^2. \quad (\text{A.20})$$

Using (A.20) reveals that

$$C_2 + \sqrt{C_2^2 - 4C_1K_r} > 2E_r + v_{ri}\xi_{ri}. \quad (\text{A.21})$$

We insert (A.21) into (A.17) to get

$$\lambda_r \geq \frac{C_2 + \sqrt{C_2^2 - 4C_1K_r}}{2C_1} > \frac{1}{(1 - 1/K_r)a_{ri}} \frac{2E_r + v_{ri}\xi_{ri}}{2E_r} > \frac{1}{(1 - 1/K_r)a_{ri}}. \quad (\text{A.22})$$

But (A.22) contradicts (A.19).  $\square$

### A.3. Proof of Theorem 2

Since

$$\frac{\partial \hat{K}_r(\lambda)}{\partial \lambda_r} = E_r + \sum_{i \in FCFS} v_{ri} Q_{ri}(\lambda) + \sum_{i \in FCFS} v_{ri} \frac{\partial Q_{ri}(\lambda)}{\partial \lambda_r}, \quad (\text{A.23})$$

it suffices to show that

$$\frac{\partial Q_{ri}(\lambda)}{\partial \lambda_r} > 0 \quad \text{and} \quad \frac{\partial^2 Q_{ri}(\lambda)}{\partial \lambda_r^2} > 0, \quad (\text{A.24})$$

which can be shown by some algebra.

### Notes

1. Various industrial applications of queueing network models were reported by Suri and Hildebrandt (1984), Brown (1988), Chen et al. (1988), and Johnson (1989).
2. See Solberg (1977), Suri and Hildebrandt (1984), Shalev-Oren et al. (1985), Yao and Buzacott (1985), and Schweitzer and Seidmann (1991) for further motivation for these assumptions.
3. See Seidmann et al. (1987) for a way to model workstations with parallel servers.
4. The ample server case is useful for modeling transporters such as conveyor belts or pick-and-place robots, and it also permits a "best case" analysis (no queueing delays) for selected workstations.
5. See appendix for its proof.
6. See appendix for the proofs of Theorems 1 and 2.
7. Buzacott and Shanthikumar (1993, pp. 392–393) report a related phenomenon in a special case which involves an FMS modeled with multiple part types but with identical exponential processing times at each workstation independent of the part types.

### References

- Askin, R.G. and Standbridge, C.R., *Modeling and Analysis of Manufacturing Systems*, John Wiley & Sons, New York, NY (1993).

- Bard, Y., "Some Extensions to Multiclass Queueing Network Analysis," in *Performance of Computer Systems: Proceedings of the 4th International Symposium on Modeling and Performance Evaluation on Computer Systems*, M. Arató, A. Butrimenko, and E. Gelenbe (Eds.), North-Holland, Amsterdam, pp. 51-62 (1979).
- Brown, E., "IBM Combines Rapid Modeling Technique and Simulation to Design PCB Factory of the Future," *Industrial Engineering*, Vol. 20, No. 6, pp. 23-26 (1988).
- Buzacott, J.A. and Shanthikumar, J.G., *Stochastic Models of Manufacturing Systems*, Prentice-Hall, Englewood Cliffs, NJ (1993).
- Buzacott, J.A. and Yao, D.D., "On Queueing Network Models of Flexible Manufacturing Systems," *Queueing Systems*, Vol. 1, No. 1, pp. 5-27 (1986).
- Buzen, J.P., "Computational Algorithm for Closed Queueing Networks with Exponential Servers," *Communications of the ACM*, Vol. 16, No. 9, pp. 527-531 (1973).
- Chen, H., Harrison, J.M., Mandelbaum, A., Van Ackere, A., and Wein, L.M., "Empirical Validation of a Queueing Network Model for Semiconductor Wafer Fabrication," *Operations Research*, Vol. 36, No. 2, pp. 202-215 (1988).
- Gray, A.E., Seidmann, A., and Stecke, K.E., "A Synthesis of Decision Models to Tool Management in Automated Manufacturing," *Management Science*, Vol. 39, No. 5, pp. 549-567 (1993).
- Johnson, E., "Manufacturing System Analysis Using Queueing Network Software," Working Paper No. 28AQ, Hewlett-Packard Corporation, Palo Alto, CA (1989).
- Kim, J., Schweitzer, P.J., and Seidmann, A., "Processing Time Optimization for Flexible Manufacturing Systems with Multiple Part Types and Distinct Repeated Visits," Unpublished working paper, William E. Simon Graduate School of Business Administration, University of Rochester, Rochester, NY (1994).
- Kusiak, A., "Design of Flexible Manufacturing Systems," in *Proceedings of the CASA/SME Conference on Flexible Manufacturing Systems*, Society of Manufacturing Engineers, Dearborn, MI, pp. 6-1-6-22 (1986).
- Luggen, W.W., *Flexible Manufacturing Cells and Systems*, Prentice-Hall, Englewood Cliffs, NJ (1991).
- Schweitzer, P.J. "Approximate Analysis of Multiclass Closed Networks of Queues," in *Proceedings of International Conference on Stochastic Control and Optimization*, Free University, Amsterdam, Netherlands (April 5-6, 1979).
- Schweitzer, P.J., "A Survey of Mean Value Analysis, Its Generalization, and Applications for Networks of Queues," Unpublished working paper, William E. Simon Graduate School of Business Administration, University of Rochester, Rochester, NY (1990).
- Schweitzer, P.J. and Akyildiz, I., "Approximate Mean Value Analysis for Non-Product Form Networks of Queues with Job Class Changes," in *Computer Performance Evaluation*, G. Balbo and G. Serazzi (Eds.), Elsevier Science Publishers B.V., Amsterdam, pp. 61-72 (1992).
- Schweitzer, P.J. and Seidmann, A., "Performance Optimization and Capacity Range Analysis for FMS's with Distinct Multiple Job Visits to Work Centers," Working Paper QM89-10, William E. Simon Graduate School of Business Administration, University of Rochester, Rochester, NY (1989).
- Schweitzer, P.J. and Seidmann, A., "Optimizing Processing Rates for Flexible Manufacturing Systems," *Management Science*, Vol. 37, No. 4, pp. 454-466 (1991).
- Schweitzer, P.J., Seidmann, A., and Goes, P.B. "Performance Management in Flexible Manufacturing Systems," *International Journal of Flexible Manufacturing Systems*, Vol. 4, No. 1, pp. 17-50 (1991).
- Schweitzer, P.J., Seidmann, A., and Shalev-Oren, S., "The Correction Terms in Approximate Mean Value Analysis," *Operations Research Letters*, Vol. 4, No. 5, pp. 197-200 (1986).
- Seidmann, A., Schweitzer, P.J., and Shalev-Oren, S., "Computerized Closed Queueing Network Models of Flexible Manufacturing Systems: A Comparative Evaluation," *Large Scale Systems*, Vol. 12, pp. 91-107 (1987).
- Shalev-Oren, S., Seidmann, A., and Schweitzer, P.J., "Analysis of Flexible Manufacturing Systems with Priority Scheduling: PMVA," *Annals of Operations Research*, Vol. 3, pp. 115-139 (1985).
- Solberg, J.J. "A Mathematical Model of Computerized Manufacturing Systems," in *Proceedings of the 4th International Conference on Production Research*, Tokyo, Japan, pp. 1265-1275 (1977).
- Solot, P. and Bastos, J.M., "MULTIQ: A Queueing Model for FMSs with Several Pallet Types," *Journal of the Operational Research Society*, Vol. 39, No. 9, pp. 811-821 (1988).
- Stecke, K.E., *Production Planning Problems for Flexible Manufacturing Systems*, Ph.D. dissertation, School of Industrial Engineering, Purdue University, W. Lafayette, IN (1981).
- Suri, R. and Hildebrant, R.R., "Modelling Flexible Manufacturing Systems Using Mean Value Analysis," *Journal of Manufacturing Systems*, Vol. 1, No. 3, pp. 27-38 (1984).

- Viswanadham, N. and Narahari, Y., *Performance Modeling of Automated Manufacturing Systems*, Prentice-Hall, Englewood Cliffs, NJ (1992).
- Yao, D.D. and Buzacott, J.A., "Modeling a Class of State-Dependent Routing in Flexible Manufacturing Systems," *Annals of Operations Research*, Vol. 3, pp. 153-167 (1985).
- Zhuang, L. and Hindi, K.S., "Convolution Algorithm for Closed Queueing Network Models of Flexible Manufacturing Systems with Limited Buffers," *Information and Decision Technologies*, Vol. 17, No. 1, pp. 83-90 (1991).