DECEMBER 1986

LIDS-P-1554
(Revised)

DYNAMIC ROUTING IN REENTRANT FMS

Oded Z. Maimon

Yong F. Choong

Laboratory for Information and Decision Systems
Massachusetts Institute of Technology
Cambridge, MA 02139

ABSTRACT

Consider a Flexible Manufacturing System (FMS), with several parallel similar production lines. Each line is statistically balanced. Due to process time and yield variations, during the FMS operation some workstations may be temporarily starved of parts, while others may have too many parts. The purpose of the dynamic routing algorithm described here is to achieve real-time load balancing in a stochastic processing environment and thus to increase the performance of the system, in throughput, workload balance and reduced work in-process queues. We formulate the problem and develop an optimal stationary policy (for two lines that have a material handling transport between them) based on the input buffer state of each station.

1 Introduction

We consider a dynamic flow type manufacturing environment with uncertainties such as failures and unavailability or quality variance of raw materials. Such manufacturing systems are common in the electronic assembly industry, where parallel lines are used to assemble a family of related products (with fast switchover time). The work presented here was motivated by designing a particular system of this nature (cf. Maimon [6]). One way to maximize the utilization of resources, and hence, to improve productivity, is to route, in real-time, the intermediary or Work-In-Process (WIP) to the appropriate stations, based on the current state of the manufacturing system. We call this practice Dynamic Routing of WIP. The computer controlled Flexible Manufacturing Systems (FMS) typically has flexible material handling system and has accessible information system that enable dynamic routing. (Dupont- Gatelmand [3] gives a general survey of FMS.)

The Dynamic Routing Problem (DRP) is a part of the operational control of FMS. Maimon [5] presents a real-time operational control system for FMS. The control is organized in a hierarchical structure according to the various decisions that take place at the different time scales at which each level operates, from a few days to milliseconds. At each level down the hierarchy, the details of the decisions and the communication rate increase. The real-time status of the system is taken into account at each level. The highest level accepts short-term management production requirements and the lowest level issues commands to the direct machine controllers. The FMS control system is comprised of three levels: from determining the changing product mix and input flow rate (see Gershwin et al. [4]); through flow of parts inside the system; to material handling moves and resources allocation. The DRP is part of the second level. Each level issues target commands to the level below it and receives performance feedback. The overall control aims to optimize FMS performance while meeting the production requirements. Buzacott [2] discusses planning and operational problems that occur in FMS, such as pre-release planning and input or release control. In that paper he mainly considers inventory level adjustment for transfer lines, and parts' input planning and control. He concludes that improved control policies for FMS operation should be developed for the various operational control issues.

Dynamic routing is difficult to achieve because of the complexity of the manufacturing system.

An information theoretic approach for the dynamic routing problem was developed by Yao [10] and Vinod [9]. They address the problem of material and information flow in FMSs, as pertaining to the DRP. They develop a concept of routing entropy to measure routing flexibility, and use the principle of "least reduction in entropy"- to determine part routing in FMS. However, these papers stop short of showing any numerical results of improvement in performance to justify their method for FMS, and the theory does not directly address production optimization criteria. Maimon and Gershwin [7] couple the real-time scheduling and routing decisions. They consider the effect of multiple route definition in the parts' process plans coupled with flexible machines. The scheduler specifies the flow rates per route. The policy developed is a feedback law that takes into account the production and machine status. This paper addresses a lower level, by considering more detailed stochastic production effects, whereby the result of applying the model presented in [7], define the load rate and distribution to each of the parallel lines.

In this paper we present a mathematical model and a method for dynamic routing of WIPs across pods. A pod is an independent entity; it consists of groups of workstations/cells which together can process raw materials into finished products. Dividing the factory into pods is practical in the environments where the production volume is high and the demands grow as the products mature. Other reasons are the factory floor layout and material handling system design. The workload of each pod is designed to be statistically balanced (i.e., considering the expected operation times). However, periodic unbalances occur due to random events.

The random events of concern during the production run include yield fluctuations, station failures and non-arrival of raw materials. Station failure could be caused by failure of mechanism or temporal absence of operator. There are several sources of yield fluctuation; it could be the quality of the raw materials, the skill of the operators, the changing quality of machinery output, or process time variation at the upstream stations. The gross effect of all these randomness is that the arrivals of WIP to a station are random.

Therefore, over a relatively short period, the workloads in a pod may

not be balanced due to the random arrival of WIP to the station. Hence some stations will be starved of WIP, while at other pods, similar WIPs have to wait to be processed at similar stations. The goal of dynamic routing is to distribute these temporary work load unbalances across the pods at all times, and thus improve manufacturing throughput and production smoothness.

The paper is organized as follows: In section 2, we present a model of WIP flow through a pod and show the existence of a stationary policy for dynamic routing. In section 3, we apply the policy to a simulation model of a real-life system and evaluate the performance. Discussion of future research effort and conclusions follow in section 4.

2 Modeling and Formulating Dynamic Routing

In this section, we present an analytical model of the pod for the purpose of dynamic routing. We demonstrate the existence of an optimal stationary policy for dynamic routing under normal operating conditions of a flow line, considering a material handling system connecting two lines at an arbitrary level.

2.1 Model

Figure 1 depicts the flow of WIP into and out of station i in a pod. Each station consists of several identical machines. They are flexible in processing different part types at a given set-up configuration; in other words, there is no set-up time when changing from one part type to another (among a family of part types).

In this model, reentrancy is allowed (i.e., parts visit the same station several times). The effects of the random events will propagate down to and back-up to station i resulting in the fluctuation of the input WIP buffer. For the purpose of workload balancing, this fluctuation of input WIP can be treated as random.

The idea of dynamic routing is to distribute the input WIP to similar stations at other pods, according to some policy. A WIP is transferred

across pods via the transportation mechanism. Note that with improper dynamic routing policy there is the risk of a station running down its own input WIP at the next cycle. Hence redistribution of the input WIP has to be done properly to realize the gain of workload balancing.

The input WIP is measured in terms of the processing time required at the station. Similarly the production at each station is measured in terms of time units. This time based measurement unit is applicable as long as there is no set up time when changing from processing one part type to another as one would find among a group of products assigned daily to an FMS. In this manner, a multi-dimensional problem is reduced to single continuous variable problem. The workload flow is modeled in the continuous domain.

The production run time is discretized into periods. Each period should be long enough to have few pieces produced and for WIP to arrive from other pods. However, the period should be as short as possible so that it approximated the continuous time better and the randomness is independent from one period to another.

2.2 System Equations

Let us define some nomenclature to be used later.
Let i denote a workstation, and n denote a production period, n = 1,...,N.
The state of the system is defined by

I_n^i – input WIP state (input buffer size),
at the beginning of period n.
P_n^i – required production,
measured in time units (thus incorporating mix of parts).
$\tilde{z}_n^i$ – random yield factor, i.i.d.r.v. (independent identically distributed random variables) for the various n's. (from upstream stations), $\tilde{z}_n^i \epsilon (0,1)$.
d_n^i – the decision,
where d is positive for request of WIP from other pods
and negative for sending WIP to other pods,
$\tilde{w}_n^i$ – unrealized decision (a random variable).

At any period n, the decision d_n^i is the WIP units requested from, or

desired to be shipped to other pods. This decision may not be realized fully due to the inventory status at the other pods. The unrealized portion of the decision is the random variable $\tilde{w}_n^i$. Hence the realized decision, denoted by $\tilde{u}_n^i$, is given as
$\tilde{u}_n^i = d_n^i$ - $\tilde{w}_n^i$,

where

$$\tilde{w}_n \in \begin{cases} [0,\ d_n^i\ ] & \text{for } d_n^i > 0, \\ [d_n^i\ ,0] & \text{for } d_n^i < 0. \end{cases}$$

The system dynamics can be stated as

$$I_{n+1}^i = max(0, (I_n^i + \tilde{u}_n^i + \tilde{z}_n^{i-1} P_n^{i-1} - P_n^i)).$$

Here the required production not met during a period will not be backlogged.

The term $(\tilde{z}_n^{i-1} P_n^{i-1})$ is the arrival of WIP from the upstream station $(i-1)$. The expected cost to be minimized is given by

$$\sum_{n=0,\ldots,N-1} E_{\tilde{z}_n^{i-1},\tilde{w}_n^i}(C(\tilde{u}_n^i) + p\, max(0, P_n^i - I_n^i - (d_n^i - \tilde{w}_n^i) - \tilde{z}_n^{i-1} P_n^{i-1}) + h\, I_{n+1}^i)) \tag{1}$$

The first term reflects the cost of transporting WIP across pods. It is given by

$C(\tilde{u}) = K\delta(\tilde{u}) + c\tilde{u}$; where K reflects the fixed transportation cost

($\delta(\tilde{u}) = 1\ if\ \tilde{u} \neq 0\ and\ \delta(\tilde{u}) = 0\ otherwise$), and c is the coefficient for reward(penalty) for each WIP sent (delivered).

The second term is the penalty for not meeting the required production for the given period (p is the penalty per unit).

The last term is the cost for holding extra WIP at the end of the period (h is the cost per unit).

This formulation is different from inventory control problem with no backlogging [8] by the following features :
(1) the 'demand' is $(P_n^i - \tilde{z}_n^{i-1} P_n^{i-1})$ which is unrestricted, (as opposed to positive demand);
(2) the decision, d_n^i is unrestricted (as opposed to positive order); and
(3) the decision might not be realized fully; the realized portion is random.

2.3 Optimal Strategy

Here we use dynamic programming approach to obtain the optimal decision to be carried out by each workstation.

For convenience, we assume that the inventory at the end of the production has no value. (Actually, the cost function at the end of the production run need only be convex). Hence the dynamic programming equations are

$$J_N(I_N^i) = 0$$

$$J_n(I_n^i) = \min_{d_n^i} E_{\tilde{z}_n^{i-1}, \tilde{w}_n^i}(C(\tilde{u}_n^i) + p\ max(0, P_n^i - \tilde{z}_n^{i-1} P_n^{i-1} - I_n^i - (d_n^i - \tilde{w}_n^i)) + h I_{n+1}^i + J_{n+1}(I_{n+1}^i)) \tag{2}$$

for n=0,...,N-1.

We define

$$H_n(y) = cy + E_{\tilde{z}_n^{i-1}, \tilde{w}_n^i}(-c\tilde{w}_n^i + p\ max(0, P_n^i - \tilde{z}_n^{i-1} P_n^{i-1} - y + c\tilde{w}_n^i) + h\ max(0, y - c\tilde{w}_n^i - P_n^i + \tilde{z}_n^{i-1} P_n^{i-1}) + J_{n+1}(I_{n+1}^i)), \tag{3}$$

where $y = I_n^i + d_n^i$.
Now equation (2) can be rewritten as

$$J_n(I_n^i) = -cI_n^i + \min_{d_N^i} E_{\tilde{w}_n^i}(K\delta(d_n^i - \tilde{w}_n^i)) + H_n(I_n^i + d_n^i),$$

for n=0,...,N-1.

Next let us show that$H_n(y)$ is K - convex (see Figure 2), and continuous and $H_n(y) \to \infty$ as $|y| \to \infty$.

Definition: A function H is K-convex, if it has the following property:

$K + H(z+y) \geq H(y) + (z/h)(H(y) - H(y-h))$ for every $z \geq 0,\ h, y > 0$.

Rewrite $H_n(y) = -cE_{\tilde{w}_n^i}(\tilde{w}_n^i) + G_n(y)$

where

$$G_n(y) = cy + E_{\tilde{z}_n^{i-1}, \tilde{w}_n^i}(p\ max(0, P_n^i - \tilde{z}_n^{i-1} P_n^{i-1} - y) + h\ max(0, y - P_n^i + \tilde{z}_n^{i-1} P_n^{i-1}) + J_{n+1}(I_{n+1}^i)). \quad (4)$$

In Bertsekas [1, pages 105-106], it is proven that $G_n(y)$ is K-convex, continuous, and $G_n(y) \to \infty$ as $|y| \to \infty$. Since $E_{\tilde{w}_n^i}(\tilde{w}_n^i)$ is a constant, therefore $H_n(y)$ has the desired properties.

With these properties, it can be inferred that the optimal decisions are:

$$d_n^i = \begin{cases} b_n^i - I_n^i & \text{for } I_n^i \leq a_n^i \text{ or } I_n^i \geq c_n^i, \\ 0 & \text{otherwise;} \end{cases}$$

where $a_n^i \leq b_n^i \leq c_n^i$.

The reason is that if I is between a and b then because of the fixed cost incurred in moving then the total cost of getting to the WIP level that minimize $H()$ is larger then doing nothing (this is the meaning of a K-convex function for this case).

The parameter b_n^i is the y that minimizes $H_n(y)$. The parameters a_n^i and c_n^i are the smallest and largest, respectively, values of y such that $H_n(a_n^i) = H_n(c_n^i) = K + H_n(b_n^i)$ (see Figure 2).

Since the required productions at every period are the same (especially for a flow line), and the random events are independent, the parameters a_n^i, b_n^i *and* c_n^i are the same for all periods. Hence at any period, the policy is:

$$d^i = \begin{cases} b^i - I^i & \text{for } I^i \leq a^i \text{ or } I^i > c^i, \\ 0 & \text{otherwise;} \end{cases}$$

Such a policy make sense as it try to keep the WIP level between control bounds. A realization of the system's WIP level behavior (with no time delay) is shown in Figure 3.

3 Simulation

The analytical development in the previous section proved the existence of a stationary policy to achieve dynamic routing of WIP. In this section, we evaluate the performance of the policy via simulation of a model of a generic part of a manufacturing facility.

The model considered here is depicted in Figure 4. It consists of two identical pods, each with four work stations. The model captures the reentrancy aspect of the actual FMS. The arrows in the figure indicate the part flow. The upper work station (in the figure) reflects all the stations before the reentrancy occurs. That is, in the plant there are several upstream stations consolidated into stations 1 and 2. Parts flow, at each pod, from the upstream station to the middle station. Then the parts continue to the side station (station 7 or 8 in Figure 4) with yield $\tilde{z}_i$ $i = 1, 2$. From there the parts return to the middle station for further processing (with FIFO policy with respect to parts coming from upstream). The final process is done at the bottom station which, in this model, encompasses the part of the system downstream of the reentrant station. The operation times (in minutes) at each station are:

WORK STATION	:	1	2	3	4	5	6	7	8
OPERATION TIME	:	1.00	1.00	.606	.606	1.54	1.54	1.45	1.45

Each pod is statistically balanced (with $E\tilde{z}_i = .65$), but because of statistical fluctuations (e.g., in yield), short-term inbalances may occur. This may cause some stations in one cell to starve, while parts are waiting to be processed in a similar station in the other cell. The dynamic routing helps in balancing the situation. In the model of Figure 4, we are concerned with the queues in front of stations 7 and 8 only. A simulation program (using the Siman package) simulated the production of this system. Two versions were used. In one version the DRP was not incorporated. In the other version the policy developed in the previous section was implemented. According to the value of the input parameters a, b and c, stations 7 and 8 request or send parts to the other pod. For the policy developed here the following values were used in the simulation: a = no input WIP ; b = 1 piece worth of processing time; and c = 2.

Table 1 shows some results of a simulation run. The upper part (A) contains the data without dynamic routing, while the lower part (B) displays the dynamic routing effect. The first eight lines in the first block of data display the utilization of stations 1 through 8. The last two rows refer to the WIP buffer size. The second block of data shows the number of requests and responses (send) for the dynamic routing. (Zero reflects the version without the dynamic routing implementation.) The last two rows give the production output.

Two production cases were simulated. In one case each pod was statistically balanced with yield factor $\tilde{z}_i$ $(i = 1, 2)$ being a uniform r.v. with parameters (.4,.9). In the other case, $\tilde{z}_1 = .5$ and $\tilde{z}_2 = .8$ for the duration of the nsimulation run. Thus the system as a whole was balanced, but each pod separately was not. The latter reflects cases that occur in manufacturing, where, for a while, one line receives consistently better quality raw material.

Each case was repeated ten times (with different random seeds). For the

first case simulation results showed that the average throughput improvement was 2.1%, and the average queue length in front of stations 7 and 8 was reduced by 47%. (Table 2 presents more detailed results.) Simulation results for the second case showed an improvement of 15% in throughput, the average queue length in front of stations 7 and 8 was reduced by 83%, and utilization of workstations drastically improved.

Note that the simulation did not take into account a finite buffer size (in the real system the buffer could hold up to six parts) which partly explains the drastic improvement in average queue length (and in production smoothness), and the relative lower improvment in throughput (as, in a balanced pod, parts that accumulate can be treated later).

4 Conclusions and Future Work

We have given a mathematical formulation to a Dynamic Routing Problem in an FMS operation, and have developed a stationary policy for the type of system considered here. This policy is an extension of the (s,S) policy used in inventory control. Moreover, this policy has the virtue of easy implementation on actual manufacturing systems.

We have demonstrated that this policy can improve the FMS performance, in throughput, workload balance and reduction of work-in-process inventory.

However, further experience and simulation results are required in order to claim conclusive quantitative results. Future work will include methods for calculating the Dynamic Routing Policy parameters (e.g., a, b, and c), and extensive testing to evaluate the performance and robustness of the policy under different conditions and for various types of systems.

REFERENCES

1. Bertsekas, D.P., "Dynamic Programming and Stochastic Control", Academic Press, 1976.

2. Buzacott, J.A., "Optimal Operating Rules of Automated Manufacturing Systems", IEEE Transactions on Automatic Control, V. 27, No. 1, 1982, pp. 80-86.

3. Dupont-Gatelmand, C., "A Survey of Flexible Manufacturing Systems," Journal of Manufacturing Systems, V. 1, No. 1, 1982, pp. 1-16.

4. Gershwin, S.B., R. Akella, and Y.F. Choong, "Short-term Production Scheduling of an Automated Manufacturing Facility", IBM Journal of Research and Development, V. 29, No. 4, 1985, pp 392-400.

5. Maimon, O., "Real-Time Operational Control of Flexible Manufacturing Systems", Journal of Manufacturing Systems, V. 6, No. 1, 1987.

6. Maimon O., "Intelligent Material Handling Control Systems", Internal Report, Digital Equipment Co., March 1986.

7. Maimon O. and S. B. Gershwin, "Dynamic Scheduling and Routing for FMS that have Unreliable Machines", MIT, LIDS-P 1610, 1986.

8. Veinott, A.F. and H.M. Wagner, "Computing Optimal (s,S) Inventory Policy", Management Science, V. 11, No. 5, 1965, pp. 515-552.

9. Vinod, K., "On Measurement of Flexibility in Flexible Manufacturing Systems: Information Theoretic Approach", Proceedings of the 2nd ORSA/TIMS Conference on Flexible

Manufacturing Systems: Operations Research Models and Applications, pp. 132-143, 1986.

10. Yao, D.D., "Material and information Flows in Flexible Manufacturing Systems", Material Flow, V. 2, 1985, pp. 143-149.

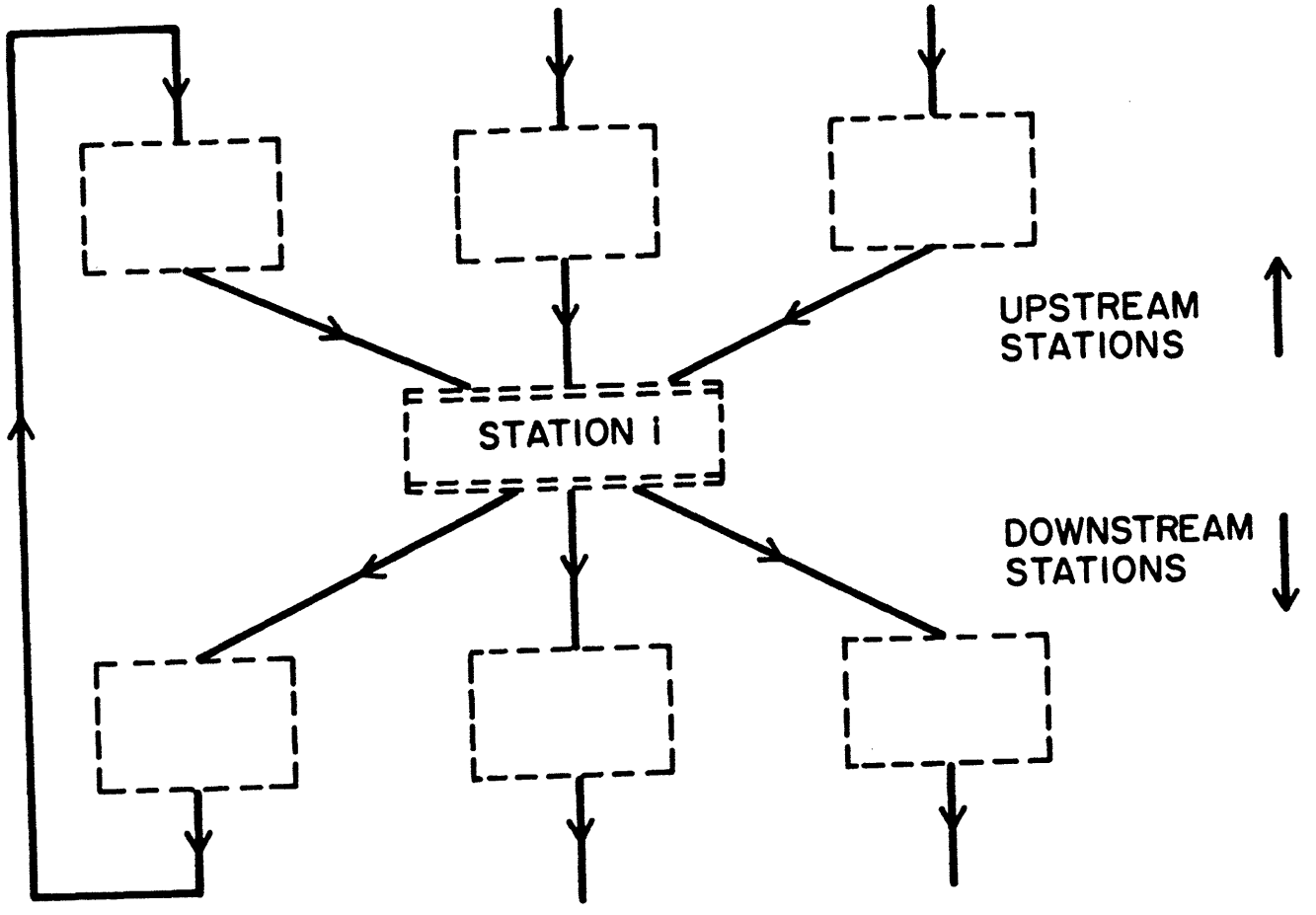

Figure 1: Reenrant FMS

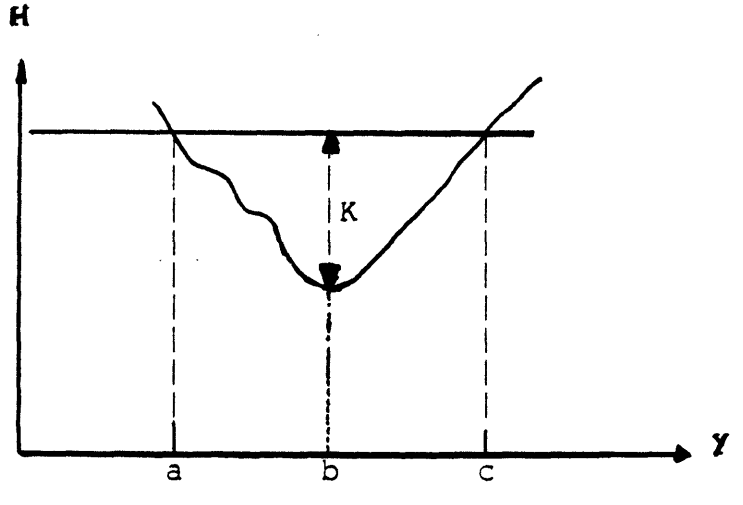

Figure 2: K - Convex Function

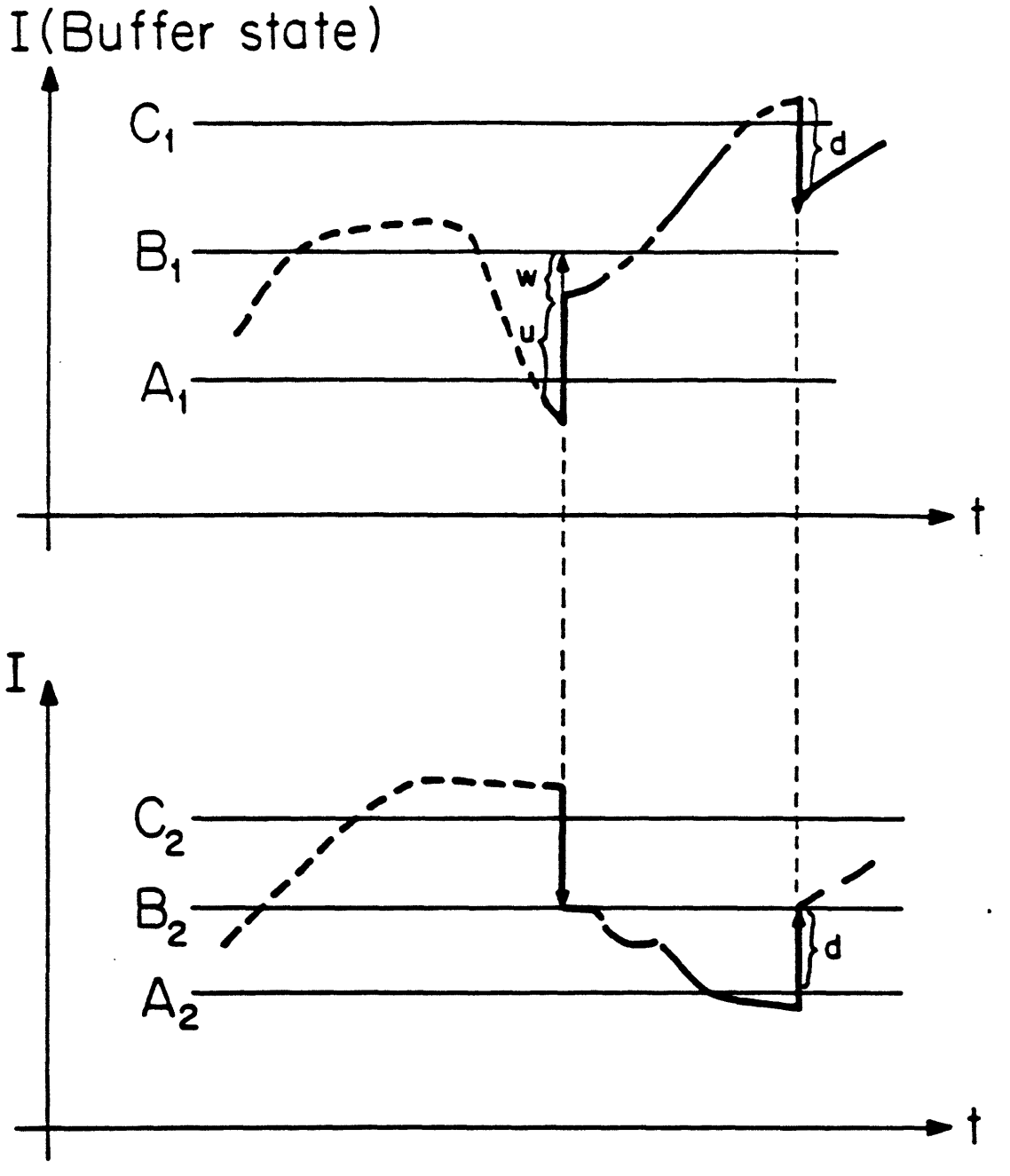

Figure 3: A REALIZATION OF TWO WORK STATIONS

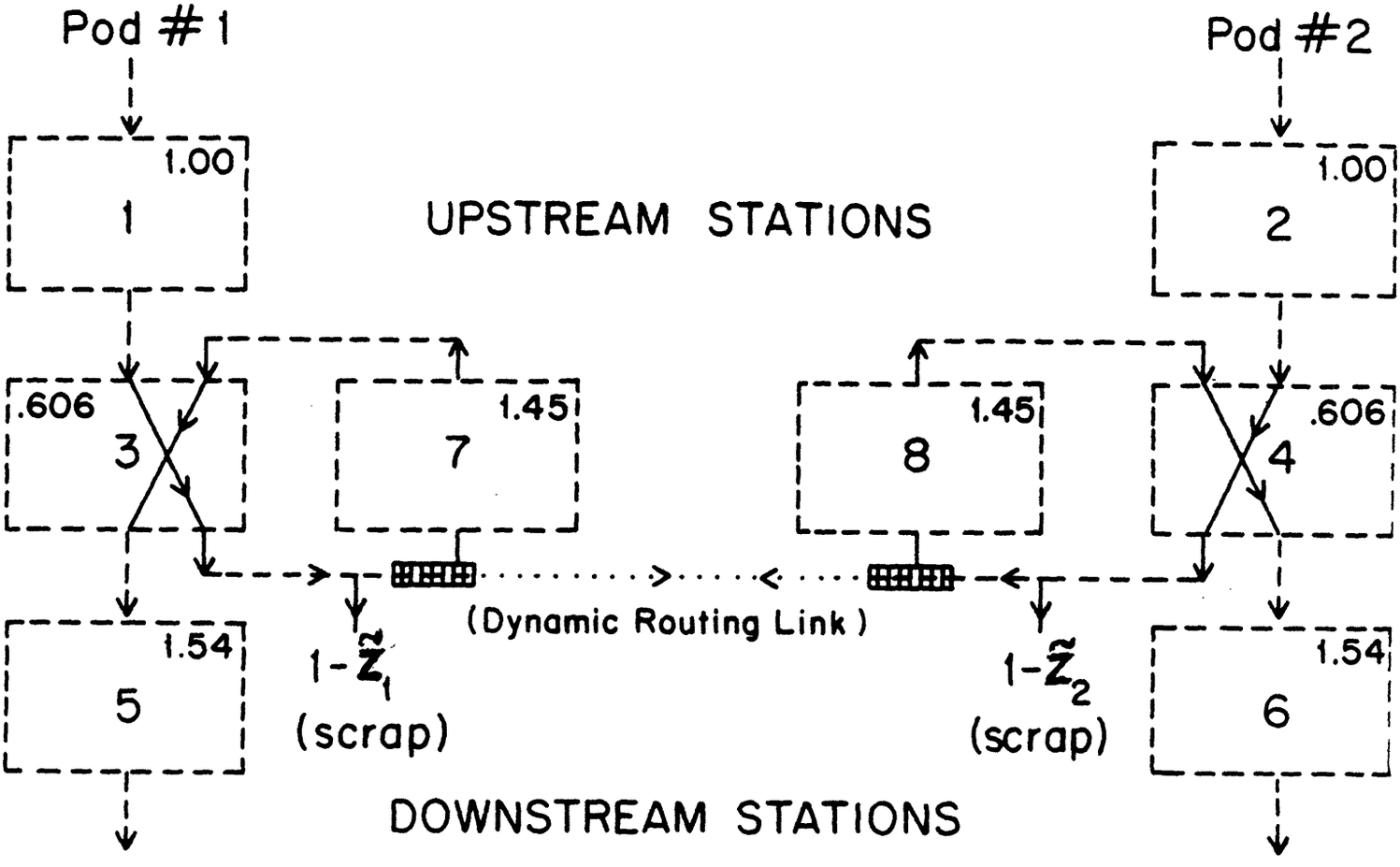

Figure 4: SCHEMATIC SYSTEM FOR SIMULATION

A.

NUMBER	IDENTIFIER	AVERAGE	STANDARD DEVIATION	MINIMUM VALUE	MAXIMUM VALUE	TIME PERIOD
1	UTIL. OF STAT#1	0.99979	0.01443	0.00000	1.00000	480
2	UTIL. OF STAT#2	0.99979	0.01443	0.00000	1.00000	480
3	UTIL. OF STAT#3	0.96610	0.18098	0.00000	1.00000	480
4	UTIL. OF STAT#4	0.90568	0.29228	0.00000	1.00000	480
5	UTIL. OF STAT#5	0.91535	0.27836	0.00000	1.00000	480
6	UTIL. OF STAT#6	0.76264	0.42546	0.00000	1.00000	480
7	UTIL. OF STAT#7	0.87198	0.33411	0.00000	1.00000	480
8	UTIL. OF STAT#8	0.72532	0.44635	0.00000	1.00000	480
9	WIP Q IN STAT#7	8.12641	9.52249	0.00000	31.00000	480
10	WIP Q IN STAT#8	2.21634	3.15622	0.00000	11.00000	480

COUNTERS

NUMBER	IDENTIFIER	COUNT	LIMIT
1	#REQ BY STAT#7	0	INFINITE
2	#REQ BY STAT#8	0	INFINITE
3	#SEND BY STAT#7	0	INFINITE
4	#SEND BY STAT#8	0	INFINITE
5	#OUT AT POD1	285	INFINITE
6	#OUT AT POD2	237	INFINITE

B.

NUMBER	IDENTIFIER	AVERAGE	STANDARD DEVIATION	MINIMUM VALUE	MAXIMUM VALUE	TIME PERIOD
1	UTIL. OF STAT#1	0.99979	0.01443	0.00000	1.00000	480
2	UTIL. OF STAT#2	0.99979	0.01443	0.00000	1.00000	480
3	UTIL. OF STAT#3	0.94947	0.21904	0.00000	1.00000	480
4	UTIL. OF STAT#4	0.95870	0.19898	0.00000	1.00000	480
5	UTIL. OF STAT#5	0.87427	0.33155	0.00000	1.00000	480
6	UTIL. OF STAT#6	0.89739	0.30345	0.00000	1.00000	480
7	UTIL. OF STAT#7	0.83115	0.37462	0.00000	1.00000	480
8	UTIL. OF STAT#8	0.85239	0.35472	0.00000	1.00000	480
9	WIP Q IN STAT#7	1.26359	1.44288	0.00000	7.00000	480
10	WIP Q IN STAT#8	1.03904	0.99991	0.00000	5.00000	480

COUNTERS

NUMBER	IDENTIFIER	COUNT	LIMIT
1	#REQ BY STAT#7	418	INFINITE
2	#REQ BY STAT#8	360	INFINITE
3	#SEND BY STAT#7	24	INFINITE
4	#SEND BY STAT#8	20	INFINITE
5	#OUT AT POD1	272	INFINITE
6	#OUT AT POD2	279	INFINITE

Table 1. Results of a simulation run

	THROUGHPUT		AVG QUEUE LENGTH		UTILIZATION	
	AVG	S.D.	AVG	S.D.	AVG	S.D.
NO ROUTING	572	23	13.4	7.8	.89	.04
DYN. ROUTING	580.5	28.5	6.0	5.4	.91	.06

Table 2. Average simulation results