A MODEL FOR ANALYZING CLOSED-LOOP CONVEYOR SYSTEMS WITH MULTIPLE WORK STATIONS

Lynn E. Bussey, Ph.D.
Kansas State University
Manhattan, Kansas

M. Palmer Terrell, Ph.D.
Oklahoma State University
Stillwater, Oklahoma

Abstract

This paper presents the modeling methodology incorporated in a GPSS/360 program to simulate and test the operation of a generalized recirculating conveyor-supplied system, consisting of loader, conveyor, and multiple work stations. Features of the generalized model include its ability to simulate a constant-speed closed-loop conveyor with discretely spaced random loads, variable spacing between work stations, variable supply and return distances, variable total length, zero or specifiably finite local storages at each work station, homogeneous service rates in the work stations, variable number of work stations, and choice of arrival (loading) distribution. Output statistics include many tabulated operating characteristics of the system.

Introduction

The design of recirculating conveyor-supplied multiserver systems constitutes a significant area of responsibility for engineers in manufacturing organizations. Since investments in conveyor systems are expensive, it is desirable that engineers have a reliable design methodology for such systems.

A search of pertinent conveyor theory literature indicates that a number of prior investigators apply multichannel queueing theory or theory-based simulations to the analysis of recirculating conveyor-supplied multiserver systems. Previous research efforts on the conveyor system typically have investigated one of two situations: (1) conveyors with randomly-spaced loads serving single or multiple servers (e.g., a powered belt conveyor), and (2) conveyors with discretely spaced loads (usually uniformly spaced) serving single or multiple servers (e.g., a powered line or overhead-type conveyor with "hooks").

The methods used to analyze the discretely spaced ("hook") conveyor are completely different from those used for the randomly spaced ("belt") type. The present model is concerned with the analysis of the recirculating "hook"-type conveyor, in which loads can be visualized as being spaced at integer multiples of the "hook" spacing on a closed-loop conveyor that moves at constant speed. While there is a fair amount of prior research concerned with non-recirculating conveyor-supplied systems, there appears to be only a meager amount of work concerned with the recirculating system in which the conveyor is a continuous loop moving at constant speed.

Kwo (2), 1958, was the first to conceive the recirculating conveyor in its role as a storage and delivery device as a part of a larger system composed of a loader, the conveyor and the service channels. He develops three intuitive principles governing recirculating conveyor operation, but presents no operating characteristics other than how to calculate limiting values of speed, capacity and uniformity of loading.

Pritsker (6), 1966, using simulation methods, investigates the effect of feedback (recirculation) for systems experiencing both deterministic and Markovian arrivals, and concludes that distance between channels and feedback delay (distance) do not affect the steady-state operating characteristics of the system. A careful reading of Pritsker's investigation will reveal, however, that he (1) precludes the formation of an arrival queue at the loading point, (2) permits the possibility for recirculated items and newly arrived items to arrive at the first server at the same virtual instant in real time, and (3) assumes zero time delay for items transiting between channels in search of an open channel, while simultaneously specifying a finite time delay on the return portion of the conveyor. These are the characteristics, not of a constant-speed system, but rather of a looped system, for example, composed of two belt-type conveyors (one, the supply conveyor to the servers, running very fast -- perhaps at "infinite" speed to provide the zero time delay between servers -- and the other, the return conveyor, running at some slower, finite speed), with both belts being randomly loaded (i.e., with loads randomly spaced).

The same comments may be directed to the simulations used by Phillips (4), 1969, and Phillips and Skeith (5), 1969, who extend Pritsker's simulation to include more system operating characteristics. The essential difference between Phillip's (4) model and Pritsker's (6) is that Phillips requires a queue to build at the head of the return conveyor if the return conveyor is occupied, whereas Pritsker's model does not contain this restriction.

These prior investigations apparently fail to model the constant-speed recirculating conveyor faithfully; that is, a recirculating conveyor in the form of a continuous loop that operates at constant speed. Consequently, the effects due to finite time delays occuring in the system are not isolated and analyzed. Thus, a remedy for these deficiencies became the impetus for the development of the model described herein.

An additional objective, adopted early in the conceptual phase, was to create a utility simulation model of the constant-speed recirculating system -- a model that would be highly flexible and capable of application over a wide range of operating conditions without changing the program code. The result is presented herein.

Description of the Physical System

The simulation program presented in this paper models the physical system depicted in Figure 1. This system consists of a supply point (or "loader"), the recirculating closed-loop conveyor, and the m service facilities or channels ("servers").

Functionally, the system shown in Figure 1 can be described as follows. Semi-processed items, presumably from a prior stage in a manufacturing process, arrive at the tail of the loader at random times, according to some inter-arrival time distribution with a mean rate λ. If a queue is present in the loader, an arriving item waits; otherwise, it is loaded directly on the first empty position on the passing conveyor. Queue discipline in the loader is FIFO. Each such on-loaded arriving item, or any recirculated item, is conveyed at constant conveyor speed to the first service channel for additional processing where the server is polled for availability. If the first server is idle, or if space in a local storage reserve is available, then the conveyor-supplied item is off-loaded and either enters the server or is added to the local storage of reserved (useables), as the case may be. If the first server or the first storage is full, the conveyor-supplied item remains on the conveyor and, a short time later, depending upon the constant conveyor speed, polls the second server for availability. If the second server is available (idle server or non-full storage reserve), the item is off-loaded; otherwise it continues to the third server, and so forth.

The polling sequence is repeated until one of two events occur: (a) entry is gained at one of the subsequent channels, or (b) entry is not gained at any channel, causing the conveyor-supplied item to recirculate at constant speed on the return portion of the conveyor, finally to reappear again at the first channel where the polling sequence begins again. With the proper choice of system parameters, in the form of arrivals, services, conveyor speed, and physical dimensions of the system, the arrivals at the loader constitute a "birth" process and the services in the servicing channels constitute a "death" process, so that the system can be operated at stochastic equilibrium.

The conceptual system shown in Figure 1 is based on the following assumptions:

1. Arrivals into the system (at the tail of the loader queue) are assumed to follow an interarrival time distribution of the investigator's choice (e.g., exponential, deterministic, etc.)

2. Local storage (reserve) may or may not be specific at any channel -- that is, the capacity of each server is restricted to the item receiving service, plus any value of local storage (including zero).

3. All service channels have the same mean service rate.

4. Outputs from the service channels are not returned to the recirculating conveyor.

5. The unloading of an item from the conveyor is determined solely on the basis of a free channel on local storage availability; that is, the unloading decision is made at the discrete point in time when the item reaches a particular channel (no "look-ahead"). Items poll the servers for availability in sequence in the direction of conveyor movement, and items failing to gain admission at the m^{th} (last) channel are recirculated.

6. The "hooks" on the conveyor are equally spaced and the conveyor moves at constant speed. Since the conveyor is a continuous loop, both service and return portions move at the same speed.

7. The physical dimensions of the system -- that is, loader to first channel distance, return conveyor length, and inter-channel spacing -- are finite ($0 < D_L, D_R, D_S < \infty$).

Model Description

Based on the foregoing assumptions and on the functioning of the physical system, a simulation model was encoded in the GPSS/360 simulation language (7).

The flow chart in Figure 2 displays the basic coding. The format of Figure 2 generally follows the physical system in Figure 1 to facilitate identification of the coded program steps with the functional portions of the system in Figure 1. Most of the GPSS statistical tabulation blocks (the system "instrumentation" by which output information is gathered) have been omitted from Figure 2 for clarity.

In describing the simulation model the following topics will be presented: composition, model input, model functioning, unique features, model output, and model execution. Each of these topics is discussed in the following paragraphs.

Model Composition

Referring to Figure 2, the model system program consists of eight functional portions: (1) data input cards (not illustrated), (2) a data initialization sequence (not illustrated), (3) a sequence of blocks representing the recirculating conveyor, (4) a sequence of blocks providing for system arrivals and representing the loader, (5) a generalized block sequence representing from 1 - 10 service channels, (6) a block sequence representing the unloading operation at the service channels, (7) a data computing sequence (not illustrated), and (8) a report generator (not illustrated).

By means of the REALLOCATE feature in the GPSS language best use is made of the necessary core, so that the program in its present version runs in 122K of a 128K partition on the IBM 360 series computers. Compile and run time together on a 360/65 varies from about 40 seconds to more than 2 minutes, depending upon the "length" of the conveyor (total number of transactions in the system) and on the number of service channels (proportional to the amount of internal computation required). The entire source deck including comment cards is about 300 cards in length.

Model Input

Two distinguishing features of a utility simulation program are flexibility of application and ease with which the model configuration can be changed. These objectives are accomplished in the model by writing the program in generalized form and INITIALizing input data. All input to the program is through ten INITIAL cards, one STORAGE card, and two FUNCTION definitions. By means of these inputs, the following system parameters are controlled:

1. Number of service channels (1 - 10),
2. Spacing between channels (in "hook" pitches),
3. Distance from loader to first channel (pitches),
4. Total length of conveyor (no. of hooks),
5. Conveyor "hook" pitch (in feet),
6. Conveyor speed (feet/min),
7. Mean interarrival time of arrivals at the loader queue tail (millimin),
8. Mean service time of a served item (millimin),
9. Standard deviation of interarrival time,
10. Standard deviation of service time,
11. Capacities of local storage ("useables") at each service channel (no. of items),
12. Statistical distribution of item interarrival time,
13. Statistical distribution of item service time.

These system parameters completely specify <u>any</u> constant-speed recirculating conveyor-supplied system of the form depicted in Figure 1, in which loads are spaced at random discrete intervals (by integral multiples of the "hook" pitch). By merely specifying the values of these parameters, one can use this model without change to simulate any particular system of the type mentioned which incorporates any arrival distribution and any service time distribution (given that they have finite variances), and which consists of 400 or less "hooks" and 10 or less service channels.

Model Functioning

After the necessary data are read in to describe the exact model configuration desired, the GPSS program first generates one transaction which performs some calculations on these data and stores the results in SAVEVALUEs for subsequent internal use. (These details are not reported here).

Subsequently, the next block sequence of interest is the manner in which the program "builds" the recirculating conveyor and simulates its operation (see Figure 2). The "hook" transactions representing the conveyor are generated by the "conveyor building generator":

```
GENERATE  X30,,2,X5,0,6,F
```

SAVEX 30 contains the mean creation time which the program has earlier computed from the specified "hook" pitch and the conveyor speed. No modifier is specified in Field B, so that the conveyor "hook" transactions are created at constant intervals of the mean time required by SAVEX 30. The first of the "hook" transactions is created at an offset of 2 time units (Field C), to permit input data initialization and calculation to take place via another (earlier) transaction originating in a preceding GENERATE block (not depicted in Figure 2). SAVEX 5 (Field D) contains the total number of transactions (number of "hooks") to be generated, each of which is generated with PRIORITY 0 (Field E), and also with 6 fullword parameters (Fields F and G). In this manner, the total length of the conveyor is established, as well as the discrete (and uniform) time spacing of each "hook" corresponding to the conveyor speed and pitch.

Since each "hook" transaction in the entire circuit is never destroyed and always experiences the same relative incremental delays in the circuit, each "hook" transaction retains its identity throughout the simulation and also its time-relative position to every other "hook" at all times, regardless of its physical location in the system logic. The functional effect is thus one of a specific sequential, time-ordered, finite numbered set of transactions which behave in the simulation as a conveyor chain on which load points ("hooks") are discretely spaced and which move at constant speed in a closed loop.

The six "hook" transaction parameters are utilized as follows:

P1: Contains the value 0 or 1, to indicate whether the "hook" is loaded (1) or empty (0).

P2: Contains a value (1, 2, ..., 10) to indicate the next service channel to be polled for entry. If entry is successful, this value becomes the channel identifying index.

P3: Contains a value for the variable time delay to be experienced by the "hook" transaction in "skip" transiting between a channel where a load has just been unloaded and the beginning point of the recirculation loop.

P4: Contains the number of recirculations experienced by a loaded "hook" before it is unloaded (reset to zero immediately after the "hook" is unloaded).

P5,P6: For use in future modifications of the model.

The next block sequence of interest is the one defining the operation of the "loading station" and its interaction with the conveyor "hook" transactions. The transactions representing incoming physical items to be conveyed and processed by the system are generated by the block

```
GENERATE  X7,FN1,X25,,1,2,F  .
```

SAVEX 7 contains the "item" mean interarrival time (from INITIAL data), which is modified by FUNCTION 1, the interarrival time distribution (any desired normalized statistical distribution with finite variance). The offset interval is specified in Field C, which uses the value contained in SAVEX 25 -- a value calculated internally in the data initialization sequence, sufficiently great timewise to permit the recirculating conveyor to be "built" before the first item to be processed "appears" at the loader. There is no creation limit (Field D), and each "item" transaction is created with PRIORITY 1 (Field E) and with 2 fullword parameters (Fields F and G).

The key blocks in the "loader" are the ENTER MAIN 1, GATE LS 1, and LEAVE MAIN 1 blocks. Storage "MAIN" is capacitated by a STORAGE definition card to a value of one. GATE LS 1 prevents the forward movement of an "item" transaction in "MAIN" until LOGIC Switch 1 is set ("on"). This combination of blocks forces the formation of a first-come, first-served queue ("LINE") of "items" behind the loader in the event the conveyor cannot load "items" as fast as they arrive.

Assuming now that the simulation has advanced to the point where there is an "item" transaction in "MAIN" (delayed by GATE LS 1), and where a queue ("LINE") exists waiting to enter "MAIN", the interaction of the conveyor with the loader can be described as follows. A recirculating "hook" transaction is first tested at

```
LOAD  TEST E     P1,0,BYPAS
```

to determine if the conveyor "hook" is empty (P1 = 0). If it is already loaded (P1 ≠ 0) the "hook" bypasses the loading sequence to the BYPAS ADVANCE X26 block. If the "hook" is empty (P1 = 0), it then tests for the presence of one or more "item" transactions in the loader:

```
TEST G     Q$LINE,0,BYPAS
```

If there is no item waiting (Q$LINE = 0), the "hook" bypasses to BYPAS ADVANCE X26. If there is an item waiting (Q$LINE ≠ 0), the "hook" transaction then sets LOGIC S 1 ("on") and immediately encounters the BUFFER block, with the following consequences:

(a) the current events chain scan is reinitiated,

(b) the "hook" transaction is delayed with zero relative time change (since it is Priority 0),

(c) the "item" transaction in "MAIN" is activated and moves (since it is Priority 1) as a result of Logic Switch 1 having been set ("on") by the "hook" transaction. The "item" transaction then continues to move, which resets Logic Switch 1 (resetting the "gate" so as to delay a subsequent "item" transaction), then on to TERMINATE 0 after passing through the DEPART LINE, TABULATE, an LEAVE MAIN 1 blocks.

This sequence destroys one "item" transaction, which has now served its purpose -- representing a load or "item" for the "hook" waiting in BUFFER. After the "item" transaction is TERMINATEd, the waiting "hook" transaction moves immediately (since it was delayed with zero relative time), and its Parameter 1 is changed from 0 to 1 (signifying now a <u>loaded</u> "hook") by the ASSIGN 1, X1 block. The "loaded hook" transaction then proceeds to the BYPAS ADVANCE X26 block, where it is delayed for the transit time (X26) analogous to the conveyor distance from the loader to the first service channel. (The time delay value in SAVEX 26 is internally calculated once from the initialized input data by an earlier one-time transaction, and all "hooks", loaded or unloaded, are delayed by the same increment of relative time).

The next block sequences of interest are the "channel test sequence" and the "channel indexing sequence" (Figure 2). These are nearly self-explanatory, except to point out that the first test (CNT2 TEST E P1,1,ADV31) is to determine if the "hook" transaction is loaded (P1 = 1). If not, the hook bypasses the channel polling sequence to the ADVANCE block (ADV31 ADVANCE X31) which provides the time delay necessary (stored in SAVEX 31) to transit the distance between the first and last service channels. Otherwise, if the "hook" is loaded (P1 = 1), the ASSIGN 2,K1 block assigns the value 1 to Parameter 2, which indicates that the polling sequence is to begin with Channel 1.

Channel 1 is tested for entry in the block

```
GATE1 GATE SNF    P2,NEXT1
```

(when the value of the channel index, P2 = 1), which allows the "loaded hook" transaction to enter the "unloading sequence" at Channel 1 if that channel (or its local storage) is not full; otherwise the transaction moves to the block

```
NEXT1 TEST L      P2,X1,OVER
```

which determines if the last channel has been tested. If not, the transaction moves through the indexing sequence in which the channel identifying number (P2) is incremented by one, a delay is encountered corresponding to the distance between two adjacent channels, and the next sequential channel is polled for entry. Loaded, unserviced "hook" transactions, after polling each channel and failing to unload, then begin the recirculation journey via the ADVANCE X28 block after passing through the statistical TABULATE RECIR block, which tabulates the recirculation frequency of loaded "hooks".

If a "loaded hook" finds an open service channel or available local storage in the polling sequence, it then enters the "unloading sequence" blocks (Figure 2). Here, the entering "hook" transaction is first increased in priority (PRIORITY 2 block), so that when a duplicate or "copy" transaction (representing an unloaded "item") is created in the SPLIT 1, CHAN block, the duplicate will have a higher priority in the current events chain. Immediately after the SPLIT block, the "hook" transaction encounters the PRIORITY 0, BUFFER block, which develops the following consequences:

(a) the priority of the parent "hook" transaction is reduced (PR = 0), while the priority of the duplicate transaction (representing the unloaded "item") remains higher (PR = 2);

(b) the current events chain scan is reactivated by the BUFFER command;

(c) the "copy" transaction moves first (due to PR = 2), ENTERing the storage whose identifying channel number is contained in Parameter 2 of the "copy" transaction, joining the appropriate QUEUE*2, and so forth;

(d) when the "copy" transaction ceases to move, due to QUEUE*2 or ADVANCE X8,FN2, then the parent "hook" transaction (now with PR = 0) moves again through the remainder of the unloading sequence blocks, which perform the housekeeping functions noted in Figure 2. It is noteworthy to add that the ASSIGN 1, K0 block changes the value of Parameter 1 in the "hook" transaction from 1 to 0, signifying that the "hook" has now been unloaded; thus, after the recirculation trip it is again available for loading again at the loader.

The final group of blocks of particular interest in Figure 2 is the "generalized service channel sequence", which models from one to ten service channels and their associated local storages. These blocks are fairly self-explanatory, although it should be noted that one of the outstanding utility features of the entire model is embodied in this sequence. To write the storage, queue and service facilities in general form, we use to advantage the fact that the channel, storage and queue identifying number is contained in Parameter 2 of the "copy" transaction created in the SPLIT block. Thus, one can use this identifying number in the form *2 to identify which ENTER, QUEUE, SEIZE, DEPART, RELEASE, and LEAVE blocks and statistics are to be affected by a particular "item" unloaded from the "hook". It should also be noted briefly that we use Parameters 3 and 4 of the "copy" transactions for the accumulation of statistical data concerning waiting and service times of an "item" being serviced, rather than for the purposes mentioned earlier for the parent "hook" transaction.

Unique Features

Some of the unique features of the model, described in detail above, can be summarized as follows:

(1) the GENERATion of a finite number of sequenced, undestroyed transactions, equally spaced in relative time, to represent a closed-loop conveyor with equally spaced load points ("hooks") travelling at constant speed throughout the simulation;

(2) the interfacing of random "item" arrivals with the continuously moving conveyor at the "loader", which realistically models the conditions (a) that an "item" is "loaded" only when an empty conveyor "hook" is available, and (b) that arriving "items" build a first-come-first-served queue if they arrive momentarily faster than the conveyor can "load" them;

(3) as a consequence of (2) "items" arrive at the first service channel at random integral multiples of the time required for the conveyor to advance one pitch length -- this is different from the completely random interarrival times assumed by prior investigators for belt-type conveyors;

(4) the "generalized service channel sequence" that permits representation of up to 10 service channels by one set of blocks;

(5) all input to the program, completely specifying the model configuration and operation, is through the use of the INITIAL, STORAGE and FUNCTION cards, which makes major model changes necessary in order to simulate different systems.

Model Output

The output from the simulation occurs via the REPORT editor, and consists of the following:

1. Report of all data inputs.
2. Report of internally calculated data:
   - (a) Conveyor length (feet)
   - (b) Distance between channels (feet)
   - (c) Distance, loader to first channel (feet)
   - (d) Distance, first to last channel (feet)
   - (e) Distance of return loop (feet)
   - (f) Conveyor circuit transit time (min)
   - (g) Transit time, loader to first channel (min)
   - (h) Transit time, first to last channel (min)
   - (i) Transit time, return loop (min)
   - (j) Transit time, between channels (min)
   - (k) Transit time, one pitch length (min)
   - (l) Conveyor service rate (hooks/min)
   - (m) Nominal input utilization (new

arrivals) on supply portion of conveyor (decimal)

(n) Mean input utilization (new arrivals) on supply portion of conveyor (decimal)

(o) Mean utilization (recirculated items) on return loop of conveyor (decimal)

(p) Mean total utilization (new arrivals plus recirculated items) on supply portion of conveyor (decimal)

(q) Mean frequency of recirculation, given that an item recirculates on return loop (decimal)

(r) Probability of recirculation (= relative frequency with which all items recirculate, including those that do not)

(s) Nominal overall system utilization (overall "black-box" utilization), %.

3. Utilization of all service channel facilities.
4. Utilization and other storage statistics for all storages, including local storages at each channel.
5. Queue statistics for each queue in the model.
6. The statistical distributions of:
   (a) Recirculation frequency of items that recirculate once or more;
   (b) Waiting times in each service channel queue;
   (c) Service times for each service channel;
   (d) Departure rate from each service channel;
   (e) Waiting time in the "loader" queue;
   (f) Loading rate (pick-up rate by the conveyor) at the "loader";
   (g) Loaded hook recirculation rate on the return loop;
   (h) Loaded hook arrival rate at the first service channel; and
   (i) Inter-item times corresponding to the rate distributions mentioned in (f), (g) and (h).

The richness of these selected performance statistics permits almost any post-simulation operational analysis to be made, as will be seen in the sample simulation results reported in subsequent paragraphs.

Model Execution

The program has been used so far to simulate about 80 different conveyor system configurations, ranging from 1 to 4 channels and with various loop-lengths, various distances between loader and first channel, various channel spacings, various speeds, "hook" pitches and other operating parameters. For a typical simulation, the following simulation control cards would be used:

```
*
*
 START          200,NP
 RESET
 START          1000,NP
 GENERATE       ,,,1,,0
 SAVEVALUE      9,V9
 SAVEVALUE      12,V12
      .
      .
      .
 SAVEVALUE      17,V17
 TERMINATE      1
 START          1
 REPORT
*
*
```

The START 200,NP command initially "loads" an empty and idle system which, in response to the RESET card, permits the actual simulation to commence with the START 1000,NP command. This tends to remove the transient start-up effect from the output statistics, which is desirable for this model. Realistically, a recirculating conveyor in a manufacturing plant, for example, would be used continuously or if it were stopped intermittently (say, overnight) then it would be restarted in the same loaded condition as when it was stopped. Thus, the steady-state operating condition is of primary interest, not the transient phase.

Following the START 1000,NP card which initiates the actual simulation, a GENERATE block is used to generate one final transaction whose function it is to calculate some statistical information from data developed in the simulation. The statistical information is stored in SAVEVALUES, via numerical calculations performed by VARIABLE statements analogous to the arguments of the SAVEVALUE blocks. Once this duty is performed the final transaction is destroyed in the TERMINATE 1 block, which ends the simulation. The REPORT card initiates the GPSS report editor, and output ensues.

A comment should be made about the transaction sample sizes (200 and 1000) used to "load" the system and perform the simulation. These are merely representative. In an actual simulation, one would need to replicate the experiment with a particular configuration several times with different random number seeds (using the RMULT feature), so as to provide an independent measure of the random error in the experiment. Then statistical tests for significance could be applied to the experimental results, and the experimenter could decide whether or not his simulation sample were large enough for steady-state to have been reached.

Typical Simulation Results

As a result of some prior work with this model, several interesting and unexpected phenomena have been observed in studying constant speed recirculating conveyors that are random discretely loaded and serviced.

For example, consider the following:

1. Based on an argument that finite delay times in the system give rise to opportunities

for system regeneration points to occur (which would not be possible if delay times were zero), it was hypothesized that system performance parameters in the finite delay system would be different from those given by earlier investigators who assumed zero time delays in the system.

The authors' earlier research substantiates this hypothesis (1). Two system performance parameters, the probability of recirculation, P_r, and the return conveyor utilization, ρ_R, were shown to be functions not only of the system loading, ρ_s, but also the input loading, ρ_i, the latter being related inversely to conveyor speed (see Figure 3). If these performance parameters were independent of finite delay times, then they would be invariate with changes in conveyor speed; however, such was not the case. Both of these performance parameters were shown to be non-simple functions of finite delay times in the system (Figure 3). This confirmed the hypothesis and also tended to indicate that an assumption of zero delay times within or between points of the system is not a realistic one for constant-speed finite-delay recirculating systems.

2. Another result is that constant-speed (finite delay) recirculating conveyor-supplied systems can display "blocking" on the service (supply) portion of the conveyor (see also Figure 3), a phenomenon not possible under prior research in which zero time delays were assumed. A constraint of all such finite delay recirculating systems is that they be operated so that the utilization of the service conveyor, ρ_c^*, never exceeds unity (see also Muth (3) for a different approach with the same conclusion for cyclically-loaded continuous conveyors). This constraint apparently does not exist and has not been reported for previously investigated recirculating conveyors.

3. It can also be inferred that the probability of recirculating, P_r, collapses into two theoretical conditions as "end points", when conveyor speed (or input loading, ρ_i) reaches limiting values (see Figure 4). For the case in which conveyor speed becomes "infinitely" large and the number of hooks great compared to a possible queue length, the system is analogous to a zero delay search condition, and P_r apparently collapses into theoretical values for P(m), the probability of all channels simultaneously busy, for a theoretical M/M/m:∞ queueing system. For the single server system case when the conveyor speed is slowed so that blocking occurs on the serving conveyor ($\rho^*_c = 1.0$), arrivals at the service channel become exactly deterministic, and P_r apparently collapses into theoretical values of P(1) for the theoretical D/M/1:0 queueing system. In between these limiting cases the probability of recirculation (analogous to probability of "overflow" for a non-feedback system) takes on intermediate values indicative of a complex function of the finite delay experienced between channels, the form of which is not

presently understood.

4. A final result is that constant-speed recirculating conveyors not only can be used as storage devices, but that maximum utilization of the return leg effectively occurs in the approximate range $0.5 \leq \rho_i \leq 0.7$ for all values of m (number of servers) and ρ_s (system loading) -- see Figure 3. In this range, the expected loader queue length is small (virtually all arrivals are "stored" on the conveyor). Thus, since the input loading, ρ_i, is under control of the designer, an opportunity exists for an economic trade-off of queue storage space at the loader for increased conveyor speed or decreased hook spacing, the principal controllable determinants of ρ_i.

Conclusions

The GPSS/360 simulation program described herein effectively models the operation of a constant speed recirculating conveyor-supplied multiserver system, in which random arrivals are accommodated and served, and in which the conveyor is loaded in a random discrete manner with load-points ("hooks") equally spaced on the conveyor loop.

The program is effective as a utility program for the investigation of different system configurations without the necessity of changing the model. To date, over 80 different configurations have been investigated, with the sole changes in the program being in the 10 INITIAL cards, 1 STORAGE card, and 2 FUNCTION definitions.

Compilation and execution time for a simulation is quite short considering the logic complexity of the system -- on the order of 40 seconds to 2.5 minutes on an IBM 360/65 computer.

The value of this utility program is demonstrated by the fact that preliminary usage has already provided some interesting and unexpected operating phenomena concerning discretely loaded recirculating constant-speed conveyors, not heretofore observed quantitatively; namely, the "blocking" phenomenon, the dependence of certain system performance parameters on loadings and conveyor speed, and the apparent collapse of the probability (relative frequencey) of recirculation into theoretically determined values under limiting conditions of system operation.

References

1. Bussey, Lynn E., and M. P. Terrell,"Characteristics of Constant-speed Recirculating Conveyors with Markovian Loadings and Services", presented at 39th National Meeting, ORSA, Dallas, May 3-9, 1971.
2. Kwo, T. T., "A Theory of Conveyors", Management Science, Volume 6, Number 1, Pages 51-71, 1958.
3. Muth, Eginhard, "System Engineering Applied to the Analysis of Continuous Flow Conveyors", Technical Report No. 64, Department of Industrial and Systems Engineering, University of Florida, Gainesville (1972).
4. Phillips, Donald T., "A Markovian Analysis of the Conveyor-Serviced Ordered Entry Queueing System with Multiple Servers and

Multiple Queues", (Unpublished Ph.D. dissertation, the University of Arkansas, Fayetteville, 1969).

5. Phillips, Donald T. and Ronald W. Skeith, "Ordered Entry Queueing Networks with Multiple Servers and Multiple Queues", *AIIE Transactions*, Volume 1, Number 4, Pages 333-342, December, 1969.

6. Pritsker, A. Alan B., "Application of Multichannel Queueing Results to the Analysis of Conveyor Systems", *Journal of Industrial Engineering*, Volume 17, Number 1, Pages 14-21, January 1966.

7. "General Purpose Simulation System / 360 -- User's Manual", Publication No. H20-0326-2, (International Business Machines Corp., White Plains, New York, Technical Publications Department, 1968).

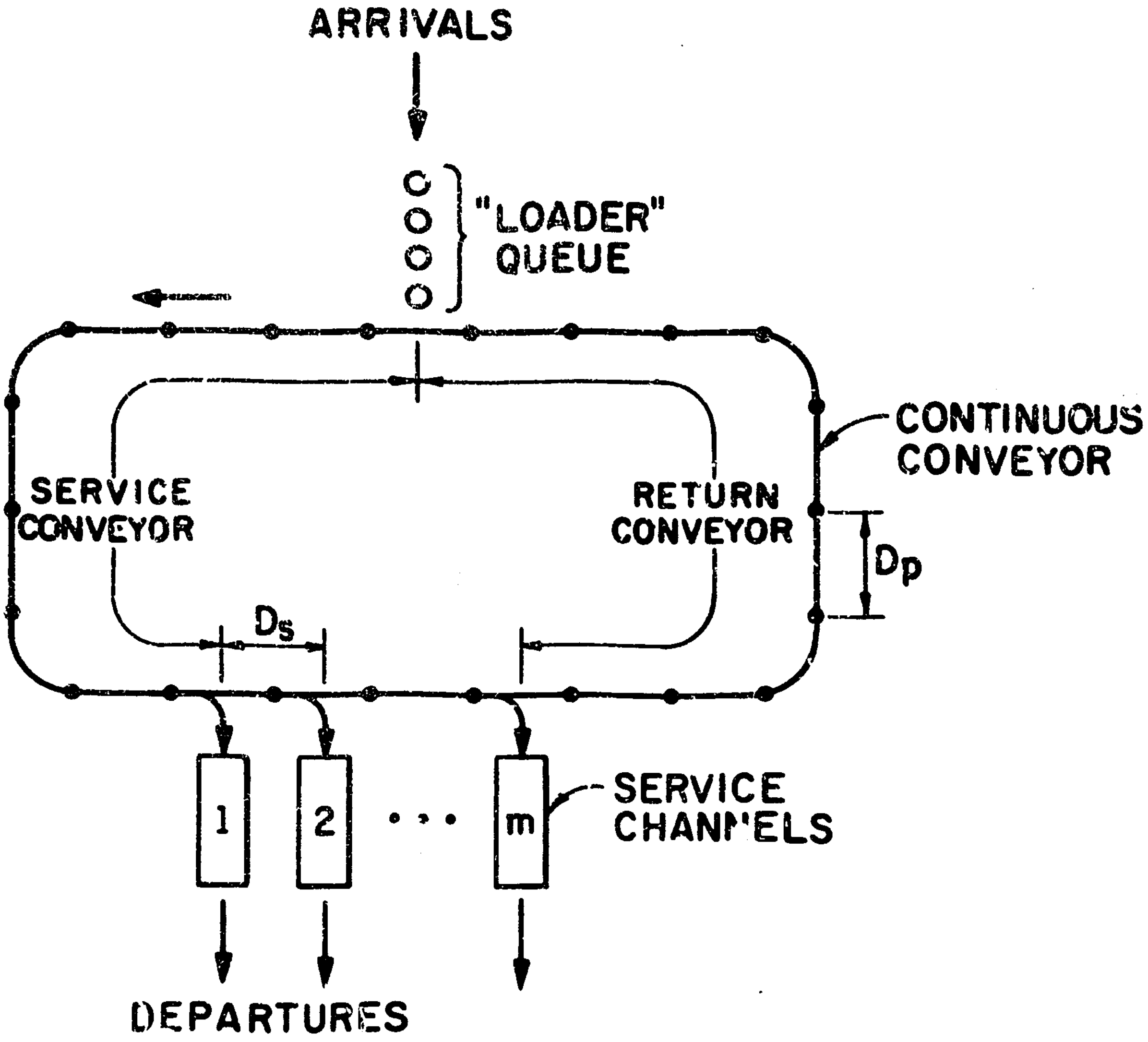

Figure 1 - Conveyor-Supplied Multiserver System.

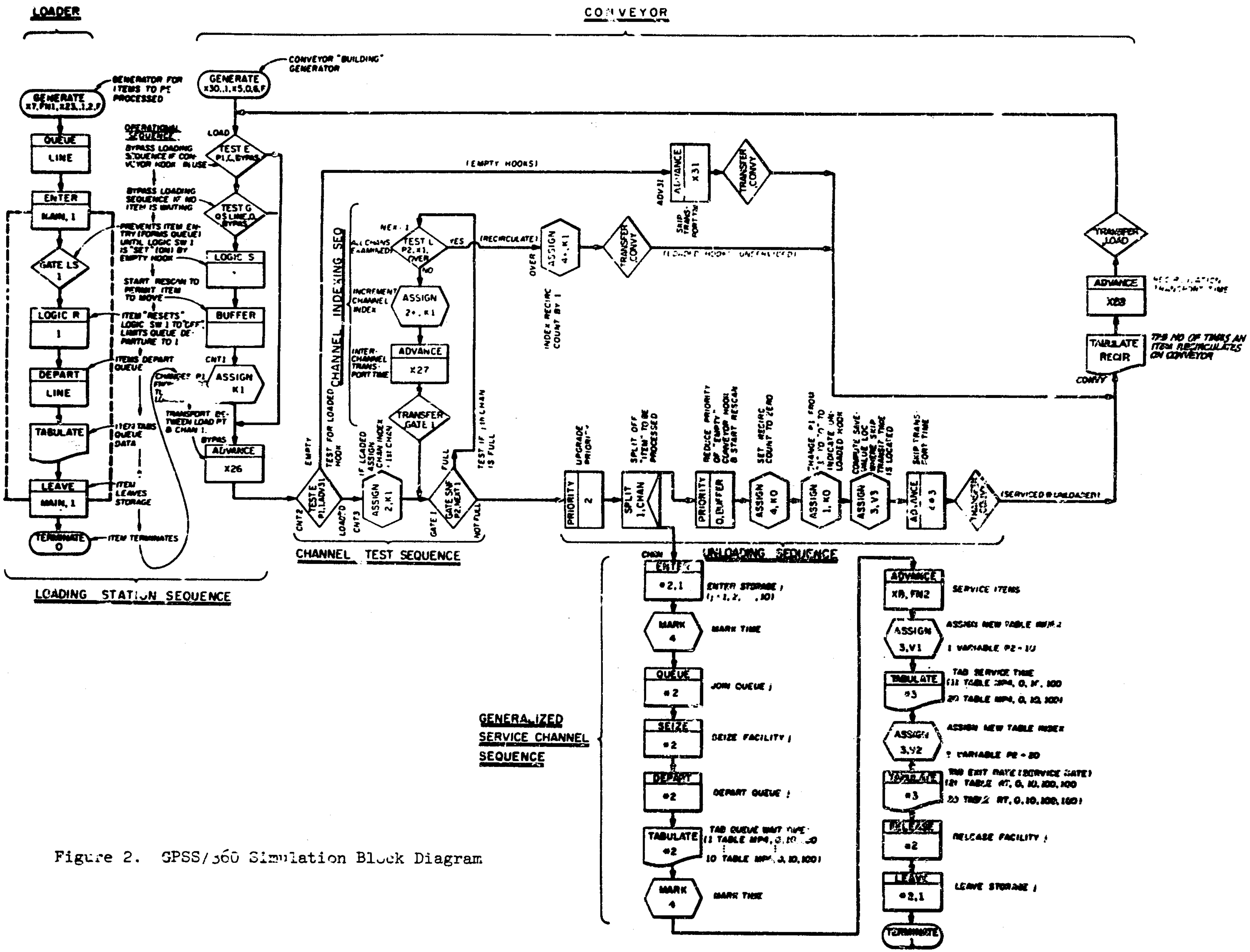

Figure 2. GPSS/360 Simulation Block Diagram

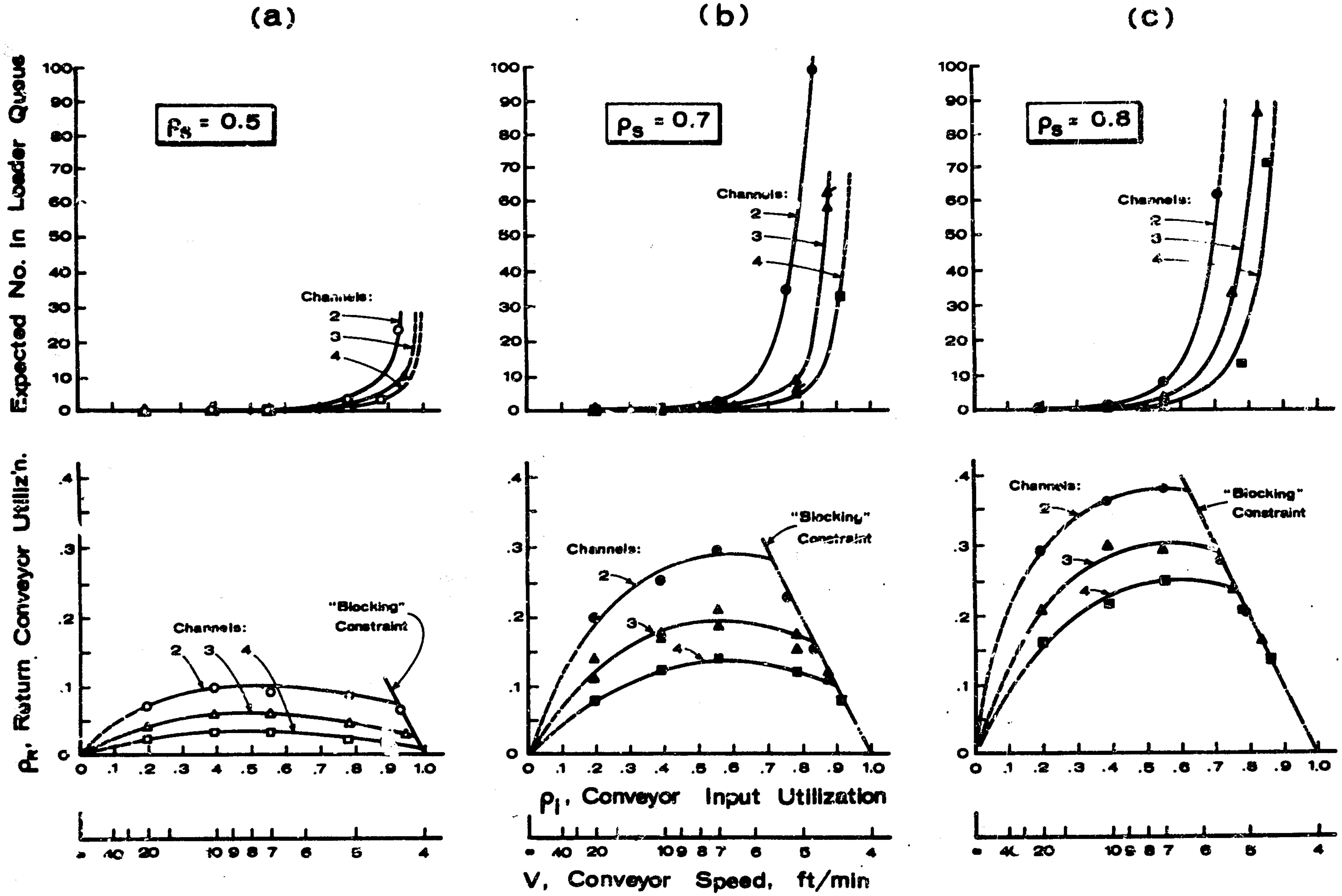

Figure 3. Effects of Conveyor Input Utilization, Conveyor Speed, and System Traffic Intensities on Basic Operational Characteristics.

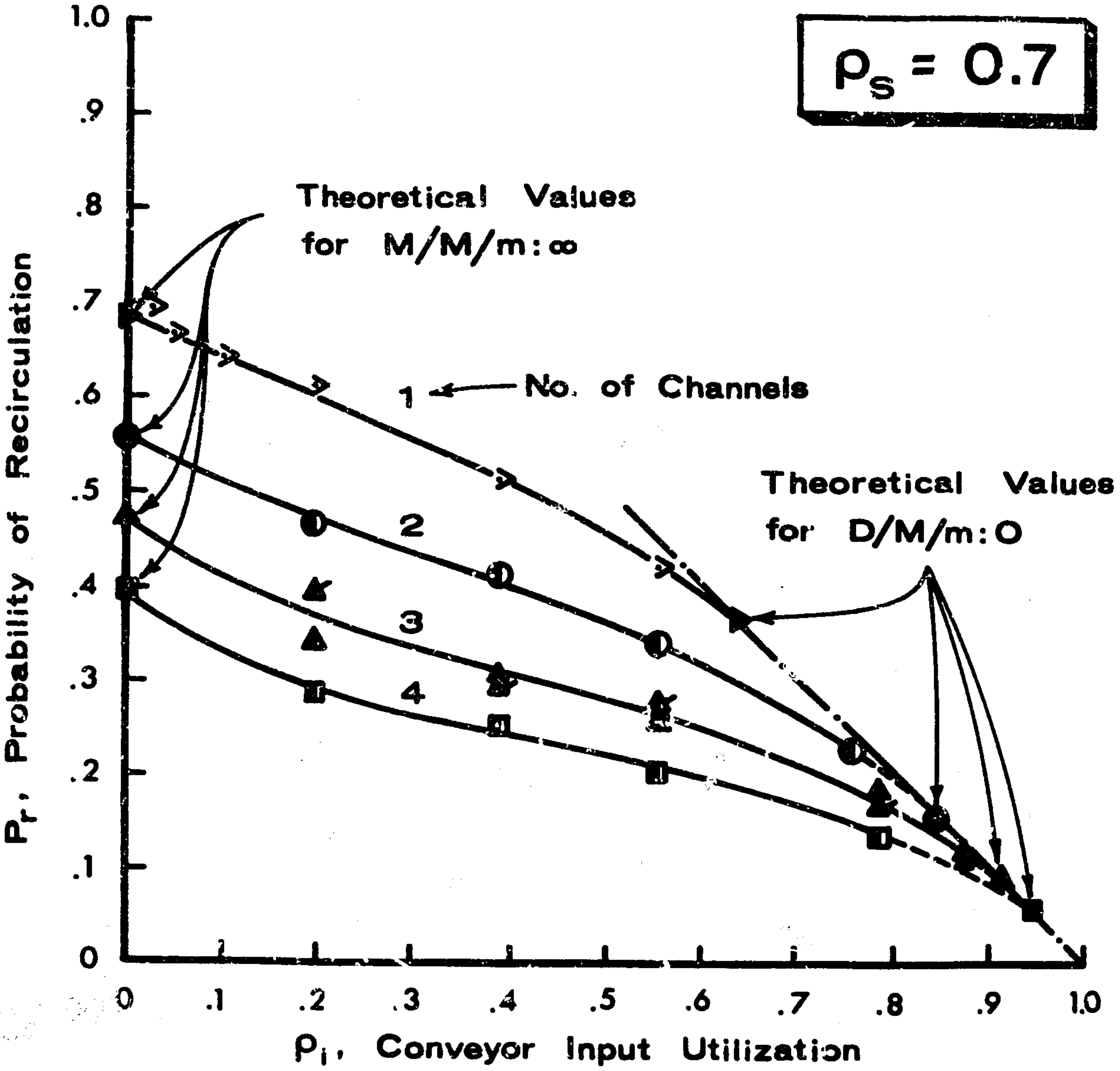

Figure 4. Effects of Conveyor Input Utilization and Number of Channels on the Probability of Recirculation.