An algorithm for carrier routing in a flexible material-handling system

by C. L. Haines

Flexible material-handling systems for manufacturing have the capability of moving articles or carriers between process stations in different sequences. The traditional method for controlling the routing of carriers is to determine, in advance, all of the useful paths within the system, and store the information in a central computer until needed. This article describes a routing algorithm that determines the correct turns a carrier should make while it is in motion. Making routing decisions does not require a global knowledge of the system's layout, because a method of numbering stations within the system which reflects its natural path of flow is employed. A brief survey of contemporary material-handling mechanisms is provided. The implementation of the algorithm using distributed controllers is discussed.

Introduction

In general, manufacturing consists of a series of well-defined processes that are performed on an article or articles to produce finished products. Some processes must be performed in a certain sequence; others may be completed in several sequences. A simple traffic pattern may be built around any single sequence of operations; the transport mechanisms and process stations are installed to implement the sequence serially. The traditional assembly line is of this type and is adequate so long as the line produces only one product or a group of products that require the same sequence of operations. The widespread use of robots and other general-purpose computer-controlled machines in manufacturing has resulted in facilities capable of producing a wide range of products each of which, in general, requires that a different series of operations be performed. To exploit the capabilities of these *flexible manufacturing* facilities, it is necessary to provide a *flexible material-handling system (FMHS)* that will allow the process stations to be visited in any order required [1].

The more stations a manufacturing facility has, the more possible routes there are between stations. The minimum number of paths between stations for a facility with n stations is described by the number of permutations of n things, taken two at a time [assuming that (1) there is only one path from a given station to each of the other stations, or (2) if multiple paths exist, only one is used]:

$$PATHS = \frac{n!}{(n-2)!} = n(n-1).$$

For large values of n, generating the paths between stations can become very time-consuming, particularly if done by hand. If there is more than one path between stations, the problem of generating all possible paths is further complicated (an example of such a system is presented in the section on the multiple-recirculating-loop pattern). In a

©**Copyright** 1985 by International Business Machines Corporation. Copying in printed form for private use is permitted without payment of royalty provided that (1) each reproduction is done without alteration and (2) the *Journal* reference and IBM copyright notice are included on the first page. The title and abstract, but no other portions, of this paper may be copied or distributed royalty free without further permission by computer-based and other information-service systems. Permission to *republish* any other portion of this paper must be obtained from the Editor.

typical FMHS, all possible paths between stations are stored in a central computer and applied as needed [2, 3]. Changes to the facility that add or delete stations may require that the paths be recomputed.

In this article we evolve an algorithm that solves the routing problem for loop systems in real time, as each article is moved. The algorithm is suitable for distributed control, allowing components of the FMHS to make all necessary routing decisions at intersections, without global knowledge of the system. A survey of the common types of material-transport mechanisms is provided for background. Several different plant-floor layouts are then examined to provide insight into the routing problem, culminating with our routing algorithm.

Material-transport mechanisms

This section contains a brief survey of the various types of material-transport mechanisms in use in automated material movement. More detailed discussions of some of these systems may be found in the articles by Pierson [4] and Zisk [5].

The function of a material-handling system is to move articles from point to point. For convenience we assume that all articles transported by material-handling systems are attached to a pallet or *carrier* for transport, though this is not true in general. For the purposes of this article, the various mechanisms have been divided into three categories: powered carriers, conveyors, and other material-handlers. All of the mechanisms can be configured to allow *buffering* at selected process stations, and those segments of a system so utilized are referred to generically as *buffers*. A buffer is used to provide a small queue of available work at a station to minimize the time that the station is idle waiting for work. Conversely, buffers provide a place for work in process (WIP) to accumulate while waiting for a station to process it.

• *Powered carriers*

In a powered carrier system, articles are transported from point to point on an individually powered cart of one form or another; for our purposes, the cart itself is the carrier. Automatically guided vehicles (AGVs), power-and-free conveyors, tow-carts, and rail-guided vehicles all fit in this category.

AGV systems An AGV is a wheeled, battery-powered cart (carrier) which moves directly on the factory floor. An onboard electronic control system allows the AGV to follow a path along the floor. The path may be either a fluorescent line painted on the floor or a guide-wire embedded in the floor. In the latter case, the wire emits a radio-frequency (rf) signal which is tracked, and may also be used for communications to and from the FMHS's central control computer(s). Different radio frequencies are used to distinguish between several paths at intersections. Commands (such as "slow down" or "sound horn") are signaled to the wire-guided AGVs by buried magnets or rf sources. Some AGVs are so sophisticated that their onboard computers contain a complete map of the guide path system and constantly monitor the positions of all other AGVs.

Power-and-free conveyors The carrier in a power-and-free conveyor system rides on a set of rails beneath the carrier or is suspended from overhead rails. The carrier is propelled by a constantly moving chain or rotating tube. Disconnecting the carrier from the power source allows the carrier to stop at process stations or in buffering areas. In some systems, the carrier is disconnected from the power source by a mechanical device mounted on the rails, whereas others use electronics mounted on the carrier to control the interface to the power source.

Tow-cart systems A tow-cart is a wheeled carrier that is pulled along the factory floor by a moving chain located in the floor or overhead. Mechanical switches are used to divert the carriers from one path to another. Their motion may be controlled by connecting and disconnecting them from the tow-chain or by controlling the tow-chain speed.

Rail-guided-vehicle systems Rail-guided vehicles are usually small carriers with an onboard electric motor. Power and guidance are received from the supporting rail. Switching of the carriers from path to path is accomplished mechanically; a typical control scheme uses sensors to read the destination or identification of the carrier as it approaches a switching mechanism and operate the switch accordingly.

• *Conveyors*

In a conveyor system, the carrier serves only to adapt the article being transported to the FMHS, and may not be required at all (in which case the term "carrier" refers to the article itself). Several types of conveyors are roller conveyors, belt conveyors, air-cushion conveyors, and modular conveyors.

Roller conveyors A roller conveyor uses powered rollers to propel carriers along. Carriers may travel end to end, subject to the control system's capability (or need) to track them. Allowing carriers to pile up in a segment of the conveyor is one method of buffering. The roller power is controlled so that the carriers press against one another without allowing the rollers to slip against the carriers. As a carrier is released from the front of the buffering segment, the carriers behind it move forward. A wide range of diverting mechanisms are available, allowing carriers to be routed off the main rollers to an alternate path. One such mechanism uses a set of narrow belts that pop up between the conveyor's rollers, allowing a carrier to be lifted from the rollers and moved sideways off the roller conveyor.

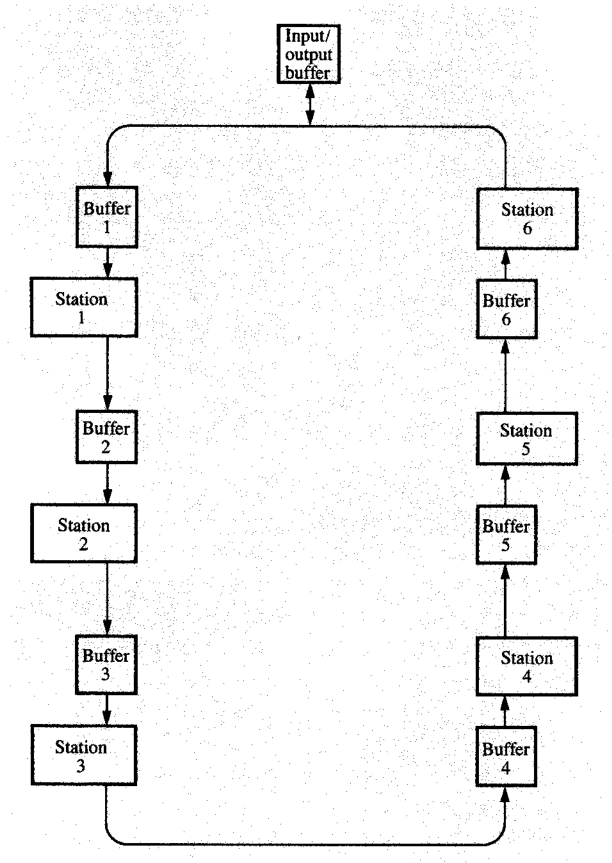

Figure 1

Serial-loop pattern.

Belt conveyors Continuous belts powered by motorized rollers or pulleys are used to implement systems of belt conveyors. Belt conveyors are usually restricted to transporting carriers in a straight line. Segments of a roller conveyor placed between two belt conveyors allow carriers to negotiate curves or turn at right angles. Mechanical diverters may be employed to push carriers off the conveyor belt and into process stations alongside the conveyor. Some systems allow carriers to travel in one direction by placing carriers on top of the conveyor, and in the opposite direction by setting the carrier under the top portion of the belt so that it rides on the returning portion of the belt.

Air-cushion conveyors Air-cushion conveyors use air flow to support and propel carriers along the track. The track is actually an air duct that has louvers cut in the top to direct the air flow, and side rails to keep the carriers on the track. Air jets may be used to divert carriers at forks in the track, allowing routing control. Air-cushion conveyors usually handle small, fragile items.

Modular conveyors Small, rectangular modules, resembling short sections of a belt conveyor, comprise a modular-conveyor system. Each module has an electric motor, two drive belts, and carrier sensors. The modules are arranged end to end so that carriers may travel off the end of one and onto the next. At intersections, adjacent pairs of modules are rotated for alignment, allowing carriers to make right-angle turns. Because the modules are individually controllable, modular-conveyor systems are extremely versatile.

• *Other material-handlers*

The material-handlers mentioned here do not fit into any of the previous categories, nor do their routing problems bear much similarity to those of the above systems. They are mentioned here only for completeness, the control of their routing being beyond the scope of this article.

Automatic storage/retrieval systems A typical automatic storage/retrieval system (AS/RS) has one or more vertical hoists, which move laterally in aisles between rows of shelves or cubbyholes. A picker mechanism mounted on the hoist pulls containers in and out of the cubbyholes and carries them to and from input/output stations at the ends of the aisles. When used as a material-transport system, the AS/RS transports the containers to special cubbyholes cut through to the outside; there, stations are set up that remove the containers from a cubbyhole, process them, and return them to the cubbyhole.

Robots Stationary robots may be used to transfer an item from one tool to another, or onto an adjacent material-handling system. They can be considered a flexible material-handling system because the paths between any two process stations within their reach are easily modified. Mobile, cart-mounted robots have also been employed to carry items from station to station.

Plant-floor traffic patterns

In this section we examine three traffic patterns of increasing complexity which serve the same basic layout of stations. The common features of the three patterns are identified.

• *Serial-loop pattern*

Figure 1 shows a serial-loop pattern implementing a single sequence of processes (carriers are assumed to be loaded into and removed from the system at the buffer marked input/output). It is possible to travel between any two stations by passing through every buffer and station in between. Routing is not a problem because a carrier simply follows the serial loop until the desired station is encountered. This pattern is not generally useful in an FMHS because carriers must wait at each station for all carriers in front of them to be processed before they can pass through the station. Also, it is possible that the loop may have to be traversed once for each process to be performed (i.e., visiting the six stations of Fig. 1 in the order 6, 5, 4, 3, 2, 1 requires six complete trips through the system).

• *Single-recirculating-loop pattern*

Many of the problems of the serial-loop pattern are solved by placing all stations and their dedicated buffers on spurs off a main loop, as shown in **Figure 2**. Because carriers are returned to the main loop after processing, each spur can be thought of as a subloop of the main loop. The routing problem is one of following the main loop until the subloop that contains the destination is encountered. At the entrance to each subloop a decision must be made whether to branch into the subloop or continue around the main loop. The main loop may also be used for buffering by allowing carriers to circulate around it until required, and is therefore called a *recirculating loop.*

The single-recirculating-loop pattern does not solve the potential problem of a carrier having to traverse the loop once for each station to be visited. As the number of stations increases, requiring a longer recirculating loop, the time it takes a carrier to traverse the loop increases. A longer transport time increases the time required to build an item, so more WIP inventory must be maintained if items are to be produced at a given rate. The additional WIP costs money, ultimately increasing the cost of the finished product and making it less competitive.

To minimize travel time (and thereby WIP), stations must be carefully arranged around the recirculating loop, such that their ordering is a best fit for various sequences of operations required by all of the products to be manufactured. This is only a compromise, however, because a new product, fully within the capabilities of the manufacturing equipment, may require such a different series of operations as to be inefficiently produced merely because of the transport time between stations. Many large tools are not easily moved to make room for new machines; therefore, adding new stations to provide increased capability may require costly changes in the system if the optimum order of stations is to be maintained.

• *Multiple-recirculating-loop pattern*

Figure 3 shows a traffic pattern that solves the problems of the previous two patterns. Crossover points have been added to the main recirculating loop to provide more direct paths between stations, resulting in a main recirculating loop that may be treated as the interconnection of multiple, smaller recirculating loops. Carriers can travel from a given station or buffer to any other station and move only the row and column distances between the points (plus twice the width of the main recirculating loop in some cases, such as a move from station 3 to station 2 in Fig. 3).

As with the previous system, each station and its dedicated buffer have a separate subloop, allowing carriers to bypass

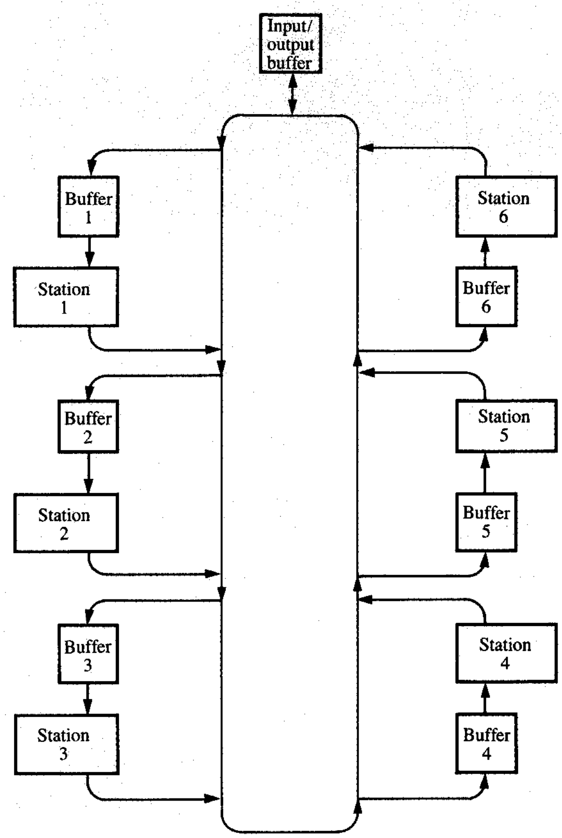

Figure 2

Single-recirculating-loop pattern.

the station. In addition, the station and buffer have been placed on spurs off the subloop (again, these spurs can be thought of as subordinate loops of the subloop). This allows carriers to be sent directly to the process station without passing through the buffer. Similarly, carriers can leave the dedicated buffer and subloop without passing through the process station.

The simplistic routing of the previous two systems has been sacrificed to gain additional flexibility and more direct paths between tools; multiple paths exist between all stations. Even the simple system of Fig. 3 would require a fairly complex program to find all the possible routes between stations (admittedly, not all of the paths are useful). It is still possible to control movement on the system by selecting a set of paths between stations and storing the set in a central computer until needed. This set of paths is, in effect, a map of the system. If control of the system were to be handled by more than one computer, each managing a segment of the FMHS, each would have to have access to the complete map to optimize the movement of carriers. In the case of a very large system, each map would occupy so

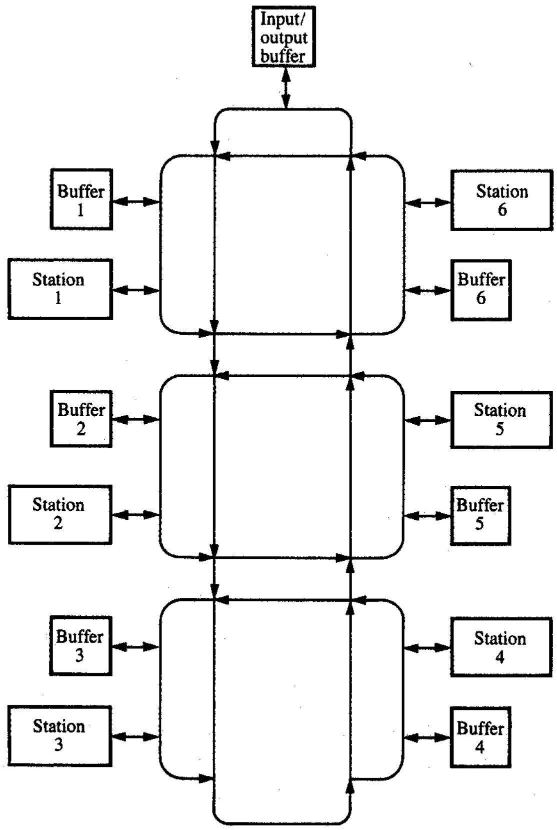

Figure 3

Multiple-recirculating-loop pattern.

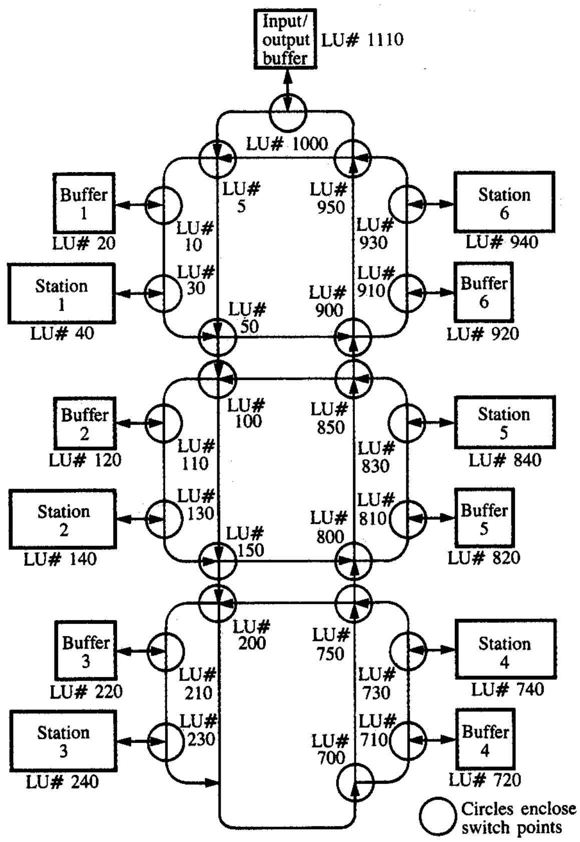

Figure 4

Possible LU numbering sequence.

much memory as to preclude the use of microprocessors for the distributed control.

There are three important features in the traffic patterns examined. First, all routing decisions consist of a choice between bearing right (clockwise) or left (counterclockwise). Second, in the latter two systems, if a carrier always bears right, it follows the path of the serial-loop system, providing a determinate path between any two points. Finally, a decision to bear left bypasses some loops of the system. We are now ready to develop a general routing algorithm to handle the types of systems discussed thus far.

The routing algorithm

The route of a carrier from one location within the FMHS to another can be described by a set of intersections that the carrier crosses, and the corresponding decisions to bear either right or left at each intersection. [This is true for the traffic patterns described above; however, in some complex systems (e.g., AGV systems) there may be more than two paths leaving an intersection. Our algorithm, as presented here, has two requirements: (1) there must exist a path through the system that passes exactly once through each intersection, buffer, and process station; and (2) there may be no more than two paths leaving any intersection. In some cases it may be possible to extend the algorithm to handle the additional paths from intersections, but we do not undertake to do so here.]

Our routing algorithm determines the correct action at each intersection, as the carrier arrives at the intersection. The algorithm may be applied in real time, as a carrier actually moves. Alternatively, a computer may be used to simulate the motion of carriers between all points in the system, storing the paths generated by the algorithm for later use.

Before we describe our routing algorithm, it is convenient to define several terms:

Switch point	Any intersection in the FMHS where a routing decision is required.
LU	Logical unit. Refers interchangeably to individual switch points, buffers, and process stations.

Candidate LU One of two LUs, adjacent to a given switch point, to which a carrier may travel directly from the switch point.

LU# Unique logical unit number.

DLU# The destination of a carrier is its destination logical unit number. The DLU# is set equal to the LU# of the logical unit toward which a carrier is traveling. If the carrier is at its destination, its DLU# is set equal to the LU# where it is located.

The key to our routing algorithm is the assignment of an ascending numeric sequence to all elements of the system, reflecting the longest path around the system (the path of the serial-loop pattern). Numbering begins at the first switch point on the serial loop (this is done for convenience, as the starting point is somewhat arbitrary). We then proceed around the system, always bearing right, and assigning an LU# to each LU as it is encountered. LU#s are assigned from an ascending sequence but need not have a regular interval. **Figure 4** shows a possible LU# assignment.

At each switch point, the routing problem is reduced to the choice between two paths. The first LU encountered on each of these paths is the candidate LU; therefore, the routing problem may also be phrased in terms of a choice between two adjacent candidate LUs at each successive switch point. Our routing algorithm, shown in flowchart form in **Figure 5**, uses two logical comparisons of a carrier's DLU# against the LU#s of candidate LUs at each switch point to determine what turns to make.

Branch 1T of the flowchart handles the situation in which the carrier is located at a point on the serial loop downstream from (numerically above) the carrier's destination. The carrier loops back, bypassing as much of the serial loop as possible, until it is inserted into the serial loop at or before its destination. For example (refer to Fig. 4), a carrier with a DLU# of 50, located at switch point 150, follows the path 150-800-850-900-950-5-50.

The case of a carrier whose DLU# falls between the LU#s of the candidate LUs is handled by branch 2T of the flowchart. In this situation, the carrier's destination lies on the subloop in the direction of the lesser-numbered candidate LU: i.e., a carrier located at the switch point with LU# 5, traveling to the buffer with LU# 20, moves to LU# 10 and then into the buffer. The final branch of the flowchart causes a carrier traveling from LU# 50 to LU# 200 to bypass the station 2 subloop, following the path 50-100-150-200.

- *Implementation using distributed controllers*

In this section we consider implementing the routing algorithm in control systems that use distributed controllers. Several advantages of using a distributed controller scheme are discussed.

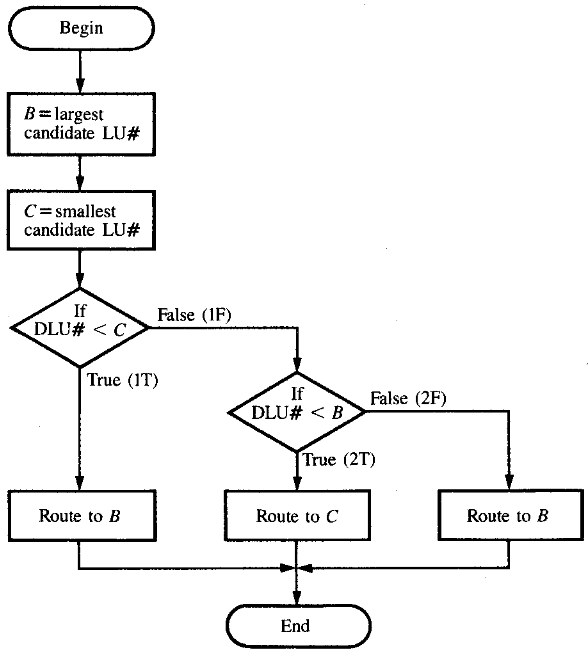

Figure 5

Flowchart of routing algorithm.

Distributed controllers require only local knowledge As discussed in the previous section, deciding which way a carrier should turn at a switch point requires that three numbers be known: the DLU# of the carrier and the LU#s of the two candidate LUs for the switch point. If there is a dedicated controller for each switch point, or alternatively, a dedicated controller for each carrier in the system, each controller requires only the three numbers, which are explicitly local in nature, to process routing decisions. Only three pieces of data are required by each controller, so that their interdependence is held to a minimum.

The system may be made arbitrarily large The only restriction on the number of LUs that a system can have is the amount of storage allocated in each distributed controller for storing the three numbers required by the routing algorithm. Thus, in a system where 24-bit binary integers are used (requiring a total of nine bytes of storage), over 16 million process stations, buffers, and switch points are possible. Adding new controllers as new switch points or carriers are added (depending on the type of system) automatically increases the system's control input/output capability. Because controllers require only local knowledge, the size of their memory requirement is independent of the number of LUs in the system; this cannot be said of a system that uses a central processor to control routing.

Changes in the system configuration have minimum impact
Because controllers that utilize our routing algorithm do not make use of a global knowledge of layout of the FMHS, radical alterations to the layout of the system can have little or no effect on a given controller. For example, adding four stations and buffers to the layout of Fig. 4, between stations 3 and 4, would affect only one candidate LU each for switch point LUs 200 and 230 (this is true as long as the LU#s assigned to the new LUs are in the range 231–699). Space was deliberately provided between the LU#s of the example system to allow for just such an eventuality.

Providing a distributed controller at each switch point is an option for most of the systems discussed in the section on material-transport mechanisms. In such systems, the controller must have some method of determining the DLU# of any carrier that arrives at the switch point. This might be accomplished by reading the DLU# from the carrier optically or electronically. Alternatively, adjacent LUs might be provided with the capability to communicate, allowing the DLU# of a carrier to be passed from one LU to the next as the carrier moves. The LU#s of the switch point's two candidate LUs are constants and could be programmed in a number of ways, including direct entry into the controller's memory via switches and loading the data from a central computer via a communications channel.

AGV systems (see the related section) lend themselves to a control scheme that places a distributed controller on each carrier. In this case, the carrier's DLU# is known by the controller, but it must acquire the LU#s of the candidate LUs for each switch point at which it arrives. Again, electrical or optical means could be used to read what is basically a signpost, providing the carrier with all the information required at a specific intersection.

Conclusions

The algorithm described here provides a simple method for making routing decisions for a certain class of material-handling systems. The algorithm is applicable to material-handling systems where two things are true: (1) there exists a path through the system that passes exactly once through each intersection, buffer, and process station; and (2) there are no more than two paths leaving any intersection. The algorithm is based on an ascending sequence of numbers assigned to the components of the system, according to their order of occurrence along the aforementioned path. Routing choices may then be made by two comparisons of the number of the carrier's destination against the numbers of the next logical unit that occurs on each of the two paths. The algorithm has the advantage that once the numbering of logical units is complete, global knowledge of the layout of the material-handling system is not needed to control routing. It is therefore ideal for use in systems where the control of routing is handled by distributed controllers functioning as peers (as opposed to systems where distributed controllers are organized in a hierarchical manner, with a central computer controlling the routing).

We hope that further investigation of this algorithm will lead to its incorporation in an actual material-handling system. The application of the numbering rules and routing algorithm to cover systems that violate the criteria stated above is another avenue for additional research.

References

1. C. Dupont-Gatelmand, "A Survey of Flexible Manufacturing Systems," *J. Manuf. Syst.* **1**, No. 1, 1–6 (1982).
2. Author anonymous, "Computer Integrated Manufacturing: Not Even Japan Has a System This Advanced," *Mod. Mater. Handling* **37**, No. 6, 60–65 (1982).
3. Author anonymous, "'We Achieved Flexible Flow to 250 Assembly Stations,'" *Mod. Mater. Handling* **39**, No. 14, 48–55 (1984).
4. R. A. Pierson, "Adapting Horizontal Material Handling Systems to Flexible Manufacturing Setups," *Indust. Eng.* **16**, No. 3, 62–71 (1984).
5. B. I. Zisk, "Flexibility Is Key to Automated Material Transport System for Manufacturing Cells," *Indust. Eng.* **15**, No. 11, 58–64 (1983).

Received January 30, 1985; revised March 1, 1985

Calvin L. Haines *IBM General Products Division, Tucson, Arizona 85744.* Mr. Haines has worked as a manufacturing engineer developing robotic tools and material-handling systems for the assembly of printed circuit cards since joining IBM in 1982. He graduated with honors from Montana State University, Bozeman, in 1982, receiving a B.S. degree in electrical engineering. Mr. Haines is a member of the IEEE Computer Society and Tau Beta Pi.