⊠ 응용논문

A Study of Buffer Management in Flexible Manufacturing Systems with an AGV System
-AGV시스템을 적용한 유연생산시스템에서의 버퍼관리 기법에 관한 연구-

Kim, Kyung Sup*
김 경 섭
Lee, Chong Ha**
이 종 하

요 지

버퍼는 생산시스템에서 시스템내의 기계고장, 프로세스시간의 변화, 그리고 부품들의 이동경로의 복잡성 등과 같은 요인들로 인해 발생 가능한 blocking현상과 starving현상을 감소시키는데 사용되고 있다. 유연 생산시스템(Flexible Manufacturing System)에서는 버퍼의 단위당 비용이 높아 시스템내의 총 버퍼 크기는 제한적이어서, 적절한 버퍼 관리를 통해 시스템 효율성을 향상시킬 수 있다. 본 연구는 이러한 버퍼의 특성을 분석하고 AGVS(Automated Guided Vehicle System)을 사용하는 다중셀방식의 FMS(multi-cell Flexible Manufacturing Systems) 환경에서의 버퍼관리와 관련된 기존 연구들을 검토하고, 새로운 개념의 가상시스템버퍼를 소개한다.

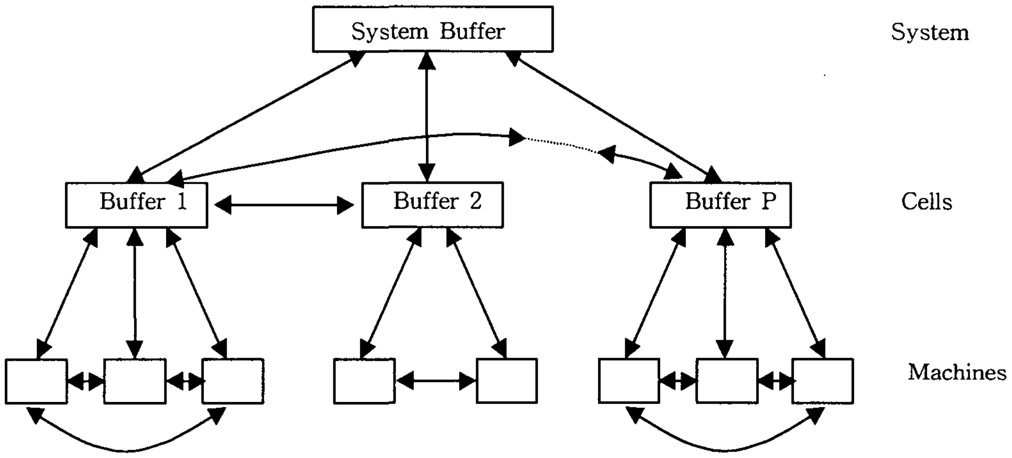
1. Introduction

A multi-cell flexible manufacturing system(FMS) consists of a group of programmable and semi-autonomous cells that are interconnected by a material handling system(MHS) and are capable of processing multiple part families with diverse processing needs. Machines within a cell in a multi-cell FMS are grouped in such a way that either the cell performs similar operations, or it performs operations required by a part family[12]. Activities of a multi-cell FMS are coordinated by an MHS. Two categories of material handling are performed in a multi-cell FMS: primary and secondary. A primary MHS is responsible for movement of parts between cells, and often an automated guided vehicle system(AGVS) is the desired handling system. A secondary MHS is responsible for movements of parts within a cell, and robots are generally among the handling devices used. The management of buffers in a multi-cell FMS with an AGVS is the subject of this paper.

---

* School of Mechanical and Electrical Engineering, Yonsei University
** Department of Management Information Systems, Yuhan College

Buffers are required to reduce blocking and starving caused by breakdowns, variability in process times, and diversity of part routing. Buffers in a multi-cell FMS can be broadly classified into three types: (1) machine-buffers, (2) cell-buffers, and (3) system-buffers. A machine-buffer is attached to a machine and is accessible only by parts that visit that machine. A cell-buffer is accessible by parts visiting that cell, while a system-buffer is accessible to any part in the system. Some multi-cell FMSs use all three buffer types while others use one or two buffer types. Figure 1 depicts the hierarchy of the three buffer types based on the location of the buffers. The arrows indicate the possible part movements between buffers.



<Figure 1> A hierarchy of three buffer types

Access to buffer spaces may be sequential or random. In a sequentially-accessible buffer, only the location at the ends may be accessed, whereas in a randomly-accessible buffer, any location in the buffer may be accessed. An example of a sequentially-accessible buffer is a chute where parts are loaded at one end and unloaded at the other end. On the other hand, an example of a randomly-accessible buffer is a rotary pallet shuttle mechanism that is used as a machine-buffer or a cell-buffer. The number of indexing stations identifies the capacity of the buffer. An example of a randomly-accessible system-buffer is an automatic storage and retrieval system(ASRS) where in-process parts are delivered and picked up automatically.

FMSs are capable of operating as an automated job shop, whereby a variety of parts follow different process steps. In an automated job shop it is very plausible that some parts require backtracking, that is, some parts must revisit a process that they have already visited. Furthermore, since parts have different processing steps, the flow of parts might not be unidirectional. These conflicts in the part's flow generally increase system congestion and affect the system throughput. A combination of backtracking, nonunidirectional flow, and limited buffer space promote deadlock conditions in an FMS.

A set of resources is in a state of deadlock when any part flow to and from these resources is inhibited. In automated manufacturing systems such as FMSs, deadlocks should be prevented or the system controller should be equipped with a deadlock resolution mechanism. Otherwise, deadlocks must be resolved by humans in real-time. Such an occurrence in the system is undesirable since it prohibits the operation of an FMS in an automated mode and affects the system productivity. For avoiding deadlock, an alternative approach is to control the number of parts allowed in the system[10, 24]. An upper limit on the number of parts in the system can be specified such that the system is deadlock free. This upper limit is a function of the part mixture and required processing steps in the system, as well as the buffer management strategy.

The total capacity of buffers in an FMS is very limited. This is due to the high per unit buffer cost that primarily consists of floor space and equipment costs. Therefore, in order to increase the system throughput and to reduce the deadlock possibility, a proper buffer management strategy must be considered before contemplating any increase in the buffer capacities. When devising a buffer management strategy for an FMS with AGVs, the unique characteristics of buffering in such a system must be considered. In this paper we discuss the problem of buffer management in an FMS with AGVs, and propose a buffer management strategy. This strategy increases the system throughput by increasing the number of parts allowed in the system without causing deadlock.

The rest of the paper is organized as follows. In section 2, review of relevant literature is presented. This is followed by a classification scheme of buffers in FMSs with AGVs in section 3. Buffer types in FMS and the management problems of each buffer type are also discussed in section 3. Finally, conclusions are presented in section 4.

2. Review of Literature

Buzacott and Shanthikumar[3] showed analytically that an FMS with only a system-buffer is superior to a system with only individual cell-buffers. However, it is not necessarily the case that an FMS with only a system-buffer achieves better throughput because the system-buffer can be occupied by parts waiting for a particular cell[25]. Kamoun and Kleinrock[9] proposed a scheme that controls the usage of a shared buffer(system-buffer) for a computer network node environment. In this scheme, a minimum number of spaces in the shared buffer are reserved for each node and also there is a limit on the maximum number of spaces that can be allocated to each node. Yamashita[25] proposed an approximation algorithm for the analysis of open queueing networks under this scheme. Sharing of cell-buffers between cells can be found in a paper by Tang et al.[20]. Tang et al.[20] used a mutil-cell FMS that has only shared common cell-buffers to test the performance of various scheduling rules. In their simulation test, a part in a cell-buffer is sent to the next closest cell-buffer whenever a machine in the cell is blocked because the cell-buffer is full.

A large number of parts in the system cause excessive congestion and traffic on the shop floor and consequently increase the possibility of deadlock[15, 26]. One approach to keep congestion and blocking at the desired level is to control the flow of unit loads into the system. Several researchers have been investigating the effects of the input control on the part flow in FMS environments[2, 5, 17, 13, 19]. A usual input control for both conventional systems and FMSs attempts to balance the assigned workload on each machine and/or to keep the workload under a specified level[19]. Garetti et. al.[8] carried out a simulation study for a static dedicated FMS by examining machine dispatching rules versus input control rules with utilization and mean flow time as the criteria. The results indicate that as the size of the system-buffer increases, the utilization value of the system rises, and the difference between the various input control rules diminishes. Similar results can be found in the paper by Sabuncuoglu and Hommertzheim[16]. They conclude that at a very low buffer capacity, scheduling of MHS and input control may become more critical than dispatching machines.

Sabuncuoglu and Hommertzheim[15] investigated the effects of the number of jobs allowed into an FMS with AGVs and examined different machine and AGV scheduling rules by comparing the mean flow time performance. The results suggest the use of a limit on the number of parts allowed in the system. Also, they pointed out that this limit is a function of scheduling rules and buffer capacities as well as the capacities of the machines and the AGVS. The results can be supported by the fact that if the number of parts is controlled to be less than a specific number, the system is free from deadlocks that are caused by part flow[10].

O'keefe and Kasirajan[14] tested 9 machine dispatching rules versus 4 route selection rules on a dedicated FMS with a system-buffer and common cell-buffers with a capacity of two parts. They concluded that route selection rules are more important, in the model they used, than the choice of dispatching rules. However, they also pointed that as the cell-buffer size increases, dispatching rules become more important because of the reduction in the probability of blocking and the increase in the choice of next parts in the cell-buffer.

An alternative for the system with limited buffer capacity is changing the buffer configuration during the operations. Given the total capacity of buffers in each cell, it is possible, in some cells, to reallocate machine-buffers in real time for different part mixes to maximize the system throughput via a reprogramming of the automated material handling device (eg. robot) within each cell[21]. Unlike reallocation of buffers within a cell, reallocation of buffer capacities to cells for different part mixes is difficult without physically relocating buffers, and consequently the reallocation of system level might not be justified.

In an FMS with an AGVS, one of the important problems is the system deadlock caused by the limited buffer and the lack of mechanism to prevent it[7]. The deadlock problem in FMS environments is addressed by several authors. The approaches taken in the FMS research to solve deadlock problems include 1) Petri-net[1, 4, 6, 22, 27], 2) Graph theory[23], 3) Queueing network[10], and 4) others[11]. Most of the above approaches are implemented in real-time scheduling and control. Complexity and performance of each model is problem specific.

An increasing number of papers concerned with FMS scheduling and control continues to appear in the literature. However, research efforts to investigate the effect of different buffer management policies on FMS performance are very scarce and limited [18]. In this paper, we will discuss the characteristics of buffers, and propose a buffer sharing strategy in FMSs with AGVs.

3. Buffer Management Strategies

High FMS buffer cost is an impetus for keeping the buffer capacity as small as possible while effectively managing buffers. An effective buffer management strategy should take advantage of the characteristics of different buffer types available in the system. As stated previously, in general, three buffer types could exist in a multi-cell FMS. They are machine, cell and system buffers.

At the machine and cell levels, buffers may be segregated into input and output buffers. A segregated input buffer at the machine level serves as the input queue for the machine, while a segregated output buffer serves as the output queue for the machine. The capacity of each segregated buffer is determined by the operational requirements of the production unit.

Segregated buffers can be accessed sequentially or randomly. A sequentially- accessible buffer is generally less expensive to build and easier to controll than a randomly-accessible buffer. The choice of priority rule for selecting a part from a sequentially-accessible buffer is limited to the first come first serve rule(FCFS) or the last come first serve rule(LCFS) which makes a sequentially-accessible buffer less flexible than a randomly-accessible buffer. However, a variety of priority rules can be used with a randomly-accessible buffer. Thus, by using a randomly-accessible buffer, the full flexibility potential of an FMS can be realized.

A non-segregated input and output buffer creates a common input/output(I/O) buffer for use by both incoming and outgoing parts. Preferably an I/O buffer should be randomly-accessible. Because managing a sequentially-accessible I/O buffer creates unavoidable machine starvation, and nullifies the advantages gained from nonsegregation of input and output buffers. The primary advantage of a randomly-accessible I/O buffer over segregated buffers is reduction of machine and part blockings.

A system-buffer can be accessed by any part in the system. When there is only one centralized system-buffer, by necessity, it must be randomly accessible. A centralized system-buffer is usually placed in the vicinity of cells with high transfer demand. However, because of the fluctuation in parts demands, transfer demands fluctuate and location of an existing centralized system-buffer may become undesirable. Therefore, decentralization of the system-buffer into several distributed system-buffers is an appropriate choice at the system level. In general, a centralized system-buffer can be less expensive to build and simpler to manage than distributed system-buffers. However, distributed system-buffers provide faster response to the requesting part and can improve system throughput.

In general, material handling systems have some degree of buffering function and the degree is determined by their management strategy. Since the primary MHS is often the largest single MHS in a multi-cell FMS, its buffering capacity can have a large impact on the throughput of the system. An AGVS is often the primary MHS of choice. However, the buffering function of the AGVS depends on its dispatching strategy. AGV dispatching strategies fall into two categories: (1) Reservation Dispatching(RD) and (2) None Reservation Dispatching(NRD). The RD strategy is to dispatch idle AGVs only to the parts that have reserved their destination. Therefore, under this strategy, the following sequence of events occurs: a part reserves its destination, an idle AGV is dispatched to the part, the idle AGV goes to the location of the part and picks the part up, then it delivers the part to its destination. Under this strategy, AGVs do not increase the buffer capacity because the parts on them have already reserved their destination resources.

Under the NRD strategy, an idle AGV is dispatched to a part that may or may not have reserved its destination at the time of the dispatching. The part can reserve its destination any time before it is unloaded at its destination. Therefore, there may be several loaded AGVs with parts that have no reservations at their destinations. A loaded AGV can be thought of as a temporary system-buffer until the part that it carries reserves a space at its destination. Under this strategy, an AGVS can be viewed as a distributed system-buffer. However, controlling AGV traffic is more complex under NRD strategy than under RD strategy. The reason is that loaded AGVs with none reserved destinations can create traffic blockage and/or system deadlock which reduces system throughput. A possible remedy is to send loaded AGVs with none reserved destinations to a system-buffer to unload the jobs. However, with this remedy, a job may visit a system-buffer several times before it goes to its next destination which generates more demand for AGVs.

In general, the movement of parts among machines and cells is controlled by either AGV or machine oriented scheduling policy such as the shortest remaining processing time or the closest distance to the AGV rule. These topics are well reported in the literature. However, papers on the movement of parts between cell-buffers and system-buffers are less common and then only briefly discussed. The rest of this section will discuss this topic.

To reduce deadlocks as well as blocking, some parts should be sent to the system-buffer from cell-buffers. Also to reduce starving, some parts in the system-buffer should be sent to corresponding cell-buffers. The decisions about the movements of parts between cell-buffers and system-buffers are when and which part should be sent to system-buffer from a cell-buffer and vice versa. In the remainder of this paper, for convenience, the cell output buffer for a cell with a common I/O buffer is defined as the places which hold outgoing parts, whereas the input buffer for the cell is defined as the rest of the common I/O buffer.

3.1 From a Cell-buffer to a System-buffer

The primary purpose of sending a part from a cell output buffer to a system-buffer is to reduce the blocking possibility of the machines in the cell. Blocking occurs when the output buffer is full. The cause of part build up in the output buffer might be the scarce output buffer capacity, slow AGVs, and/or the scarce input buffer spaces that the parts in the output buffer are heading. Especially when the AGVS is a bottleneck resource, the waiting period from a move request to the time that a part is actually removed from the output buffer and its variance may be large and not negligible. Therefore, in that case, the request for part movements from a cell output buffer to a system-buffer should be in advance of actual blocking.

There are a couple of major decisions that should be made when devising a buffer management strategy, in regards to the part movement from a cell-buffer to the system-buffer. They are (1) which part in an output buffer should be sent to the system-buffer and (2) when it should be sent.

When a part at an output buffer finds that its next machining cell is full, there are two choices: (1) send it to the system-buffer or (2) keep it at the current output buffer. Without any restriction on sending parts to the system-buffer, a large system-buffer may be required and unnecessary transportation might occur. For example, assume that every part is sent to the system-buffer as soon as its next machining cell becomes full. In this case, if some free spaces at the output buffer are available, transferring parts in the buffer to the system buffer is premature and unnecessary. Sending a part to the system-buffer and then from the system-buffer to the next machining cell requires two AGV dispatchings, whereas only one AGV dispatching is required if the part is directly sent to the machining cell. Also, a large capacity of the system-buffer is needed. To reduce the demand on an AGVS and the required system-buffer capacity, in general, more restricted conditions are preferable in sending a part from an output buffer to the system-buffer. Some alternatives are to send a part from an output buffer (1) when the current output buffer is full, (2) when both the input buffer and the output buffer of the current cell are full, or (3) when the current output buffer is full and all input buffers that the parts in the current output buffer are heading toward are full. Alternative (1) is less restricted than alternative (2) and alternative (3) is most restricted among them. The more restricted conditions under which parts are sent to the system-buffer, the fewer parts actually go to the system-buffer and so the higher the possibility that the system-buffer has large available places but output-buffers are congested, and consequently leaving more machines blocked.

Under RD dispatching strategy, a machining cell to which a loaded AGV is heading has free spaces and a space is reserved for the part on the AGV. Consequently, there is in general no need to send the AGV to the system-buffer. Therefore, under RD dispatching strategy, the decision of sending a part to the system-buffer should have been decided before dispatching an idle AGV to the part. On the contrary, under NRD dispatching strategy, a part in an output buffer has only the system-buffer as its immediate destination if AGVs are thought of as distributed system-buffers. However, the decision whether the AGV will go first to the ultimate destination cell or to the system-buffer will be decided later. The decision time of sending a loaded AGV to the system-buffer can vary from the moment when the part is loaded onto it to the moment when it arrives at its destination cell and finds that the input buffer is full.

When it is decided to send a part from a system-buffer in a system that has several distributed system-buffers, the decision should be made about which system-buffer the part will be sent. Some alternatives in the selection of a system-buffer are: (1) selecting the system buffer that is closest to the part's next machining cell, (2) selecting the system-buffer that gives the smallest travel distance to complete delivering the part to its next machining cell, and (3) selecting the system-buffer that has largest available space.

Finally, when it is decided to send a part from an output buffer to the system-buffer, the next decision is to select a part that will be sent. If a cell has a sequentially-accessible output buffer, then the choice of the part that will be sent to the system-buffer should be the part that is at the head of the buffer. If a cell has a randomly-accessible output buffer, then there are alternatives in selecting the part to be sent to the system-buffer, such as the longest waiting part at the buffer, or the part that has the largest work-in-queue at its immediate destination.

3.2 From a System-buffer to a Cell-buffer

The time of request for a part to move to its next immediate cell from a system-buffer can vary from the moment when the part is loaded onto an AGV for transferring to the system-buffer, to the moment when its immediate destination cell input buffer becomes empty. However, to be efficient, the system should prohibit dispatching an AGV to the parts in the system-buffer unless certain conditions are met. For example, immediately after an AGV has unloaded a part into the system-buffer under NRD strategy, it is looking for a part that requests a transportation. If there is no restriction on requesting transportation of a part in the system-buffer, the AGV may be dispatched again to the part just unloaded, though its ultimate destination cell input buffer is still full. Also, when a part arrives at the system-buffer under RD strategy, it is highly probable that the destination buffer of the part is still full. Then, the part has no choice but wait at least until its ultimate destination cell input buffer changes its status from full to available so the part can reserve its destination.

A simple strategy for part movement from the system-buffer to a cell-buffer is to dispatch an idle AGV to a part in the system-buffer only when the destination input buffer of the part has less than a certain number of parts in it or reserved spaces in it. In this case, at most a part in the system-buffer is sent to a cell every time a part in the input buffer of the cell is removed from the buffer. The threshold number is a function of the capacity of the input buffer, the capacity of the AGV, and the part arrival rate at the cell. Another alternative strategy is to send a fixed number of parts in the system-buffer to a cell input buffer if the number of parts in the buffer is less than a certain number. The main idea of this strategy is to send as many parts as possible to a cell whenever the cell is no longer a bottleneck.

Using the more restricted conditions (i.e. smaller threshold value) to receive parts from the system-buffer, there is a high possibility that a cell may be starved. There is less of a chance that the parts from the system-buffer are blocking other parts that are heading to the cell at other cell-buffers, and consequently fewer parts are forced to go to the system-buffer. Similar situations happen in the case of the number of parts. The higher the number of parts in the system-buffer is sent to a cell-buffer, the less the possibility of starving but the higher the possibility of forcing other parts to the system-buffer.

In many cases, the system controllers are designed with an assumption that the system-buffer is always available, though there is a limit on the number of parts that can use the buffer at a given moment. This assumption can lead a system to a deadlock. The deadlocks can be prevented by implementing a real-time deadlock resolution model. However, real-time deadlock resolution models require detailed system status information and may be too complex to be implemented in the system level. A simple solution of the deadlock problem caused by the capacity of the system-buffer is the addition of new system-buffers. However, space limitation and cost may prevent new system-buffers. In the next section we will propose a system and cell-buffer management scheme that uses random accessibility of cell I/O buffers to increase the maximum number of parts allowed in the system without causing deadlocks.

3.3 Buffer Sharing

Conventionally, only the system-buffer receives parts that are requested to leave their current cells while their ultimate destinations have no free spaces. However, if the system-buffer has no free spaces either, the parts have to wait at their current locations and will consequently block other parts. For the system with a small system-buffer, one alternative is to send those parts temporarily to other cell-buffers that have large available spaces and are close to the ultimate destinations of the parts. This sharing of a cell-buffer with other cells may cause problems if the cell as well the system has certain properties. The most difficult obstacle to implement this idea is that to be efficient, cell-buffers should be randomly-accessible and capable of loading and unloading parts to AGVs, and the system controller should have the information on the status of the each buffer. Indeed, it is difficult to find a system that uses this idea in conventional job shop environments. Technically, the concept of FMSs with AGVs can accommodate this scenario because in general FMSs with AGVs and common I/O cell-buffers do provide the randomly-accessible cell-buffers, such as a carousel, and also provide the integrated computer system. However, there has been no accepted control model to implement this idea with FMSs in combination with AGVs. Here, a virtual system-buffer is defined as the portion of a cell-buffer that is temporarily shared with other cells. In this section we will discuss management strategies of virtual system-buffers.

Decision variables in managing virtual system-buffers are: (1) what portion of each cell-buffer should be part of a virtual system-buffer, and (2) how and when virtual system-buffers will be used. The size of a cell's virtual system-buffer should be a function of the capacity and nature of the cell as well the total capacity of system-buffers in the system. For example, the size of a cell's virtual system-buffer when the cell is a bottleneck cell should be smaller than that when the cell is a nonbottleneck cell. In general, for each cell, the larger the portion of its cell-buffer is dedicated to the virtual system-buffer, the larger the increase in the system-buffer capacity. However, the possibility that the cell may be inefficient due to the lack of the buffer capacity dedicated to the cell also increases. To prevent such inefficiency, at least two places of cell-buffers in each cell should be reserved for the parts visiting the cell, one for the incoming parts and one for the outgoing parts. For example, suppose that a cell is located next to the bottleneck cells and all of its cell-buffers are shared with other cells. Then, there is a high possibility that its cell-buffers become full with parts that need to be processed by the bottleneck cells. In this case, even though machines in the cell are all free, no part visiting the cell can be delivered to or removed from the cell unless a part heading to a bottleneck cell leaves the buffer. Therefore, it is reasonable to reserve one place for the incoming parts and one place for the outgoing parts. Consequently, the portion of a virtual system-buffer in a cell should be two places less than the total capacity of cell-buffers in the cell.

Due to the similarity between a virtual system-buffer and a distributed system-buffer, the management strategy discussed in sections 3.1 and 3.2 can be applied to the virtual system-buffer management with a small modification. One situation that can be modified is when selection among system-buffers is required. The management schemes that give different priority to the virtual system-buffer over the system-buffer are worthy to be considered. For example, suppose that an objective of the system design is to reduce the required system-buffer capacity. One alternative is to give a higher priority to the virtual system-buffer over the system-buffer. By doing so, less parts are going to the system-buffer, and consequently a smaller capacity of the system-buffer is required. However, since the capacity of the system-buffer is already known at the operational level, there is no particular reason to use the virtual system-buffer over the system-buffer. For another example, since it is more likely that parts in the system-buffer do not block other parts, giving higher priority to the parts in virtual-buffers over the parts in the system-buffer is more reasonable in terms of selecting the next incoming part.

After a part arrives at a system-buffer, the part does not leave the system-buffer until it is pulled by its destination machining cell. However, after a part arrives at a virtual system-buffer, it can be pushed to another virtual system-buffer or to a system-buffer as well pulled by its next machining cell. For example, when a cell-buffer is almost full, its cell controller may send a part in its virtual system-buffer to other virtual system-buffer to make room for outgoing parts as well for incoming parts. Consequently, a part might take several trips before it goes to its next immediate cell and AGV demands may increase. Therefore, for the system with small cell-buffers and tight AGV capacity, virtual system-buffer schemes may not perform well.

AGVs are used as a distributed system-buffer under the NRD strategy. In most cases an AGVS is a bottleneck resource and the main function of the AGVS is transporting parts. To prevent all AGVs to be loaded with blocked parts and be deadlocked under the NRD strategy, some parts on the AGVS should be pushed to virtual system-buffer or the system-buffer by other parts waiting for transport so that some AGVs can be free later and transport other parts. It is possible that an AGV loaded with a part is continuously traveling the system looking for place (virtual system-buffer) to unload the part. An alternative is to mix the two AGV dispatching strategies, such that a system uses both AGV dispatching strategies simultaneously, with some AGVs controlled under RD strategy and the rest controlled under NRD strategy. When some conditions are met such as the above mentioned situation, an AGV that is controlled under NRD strategy may be controlled under RD strategy and vice versa. The number of AGVs under NRD determined the possible increase in the system-buffer capacity, whereas the number of AGVs under RD determines the minimum transportation capacity of the AGVS.

4. Conclusions

In this paper, the characteristics and the classification scheme of buffers in FMS environments with an AGV system are discussed. Also, the work presented in the literature on buffer management strategies has been reviewed. Most of techniques and methodologies discussed in the literature are concerned with FMS scheduling and control. Research efforts to investigate the effect of different buffer management policies on FMS performance are very scarce and limited. It is important to note that proper buffer management can provide high system efficiency. The idea of the virtual system buffer strategy is proposed in this paper, and it would be one such concept that is worth of a continuing research.

Reference

[1] Banaszak, Z. A., and Krogh, B. H., "Deadlock Avoidance in Flexible Manufacturing Systems with Concurrently Competing Process Flows," IEEE Transactions on Robotics and Automation, 6(6): 724-734, 1990.

[2] Bernardo, J. J., and Mohamed, Z., "The measurement and Use of Operational Flexibility in the Loading of Flexible Manufacturing Systems," European Journal of Operational Research, 60(2):166-182, 1992.

[3] Buzacott, J. A., and Shanthikumar, J.G., "Models for Understanding Flexible Manufacturing Systems," AIIE Transactions, 12(4): 339-350, 1980.

[4] Cecil, J. A., Srihari, K. and Emerson, C. R., "A Review of Petri-net Applications in Manufacturing," Int. J. Adv. Manufacturing Technology, 7: 168-177, 1992.

[5] Denzler, D. R., and Boe, W.J., "Experimental Investigation of Flexible Manufacturing System Scheduling Decision Rules," Int. J. Prod. Res., 25(7): 979 994, 1987.

[6] D'souza, K. A., and Khator, S. K., "A Survey of Petri Net Applications in Modeling Controls for Automated Manufacturing Systems," Computers in Industry, 24: 5-16, 1994.

[7] Egubelu, P. J., and Tanchoco, J. M. A., "Characterization of Automatic Guided Vehicle Dispatching Rules," Int. J. Prod Res., 22(3): 359-374, 1984.

[8] Garetti, M., Pozzetti, A. and Bareggi, A., "On-line Loading and Dispatching in Flexible Manufacturing Systems," Int. J. Prod. Res., 28(7): 1271-1292, 1990.

[9] Kamoun, F., and L. Kleinrock, L., "Analysis of Shared Finite Storage in a Computer Network Node Environment under General Traffic Conditions," IEEE Transactions in Communications, COM-28: 992-1003, 1980.

[10] Kundu, S. and Akyildiz, I. F., "Deadlock Free Buffer Allocation in Closed Queueing Networks," Queueing Systems, 4: 47-56, 1989.

[11] Leung, Y. T., and Sheen, G. J., "Resolving Deadlocks in Flexible Manufacturing Cells," Journal of Manufacturing Systems, 12(4): 291-304, 1993.

[12] Maccarthy, B. L., and Liu, J., "A New Classification Scheme for Flexible Manufacturing Systems," Int. J. Prod. res., 31(2): 299-309, 1993.

[13] Modi, B. K., and Shanker, K.,"Models and Solution Approaches for Part Movement Minimization and Load Balancing in FMS with Machine, Tool and Process Plan Flexibilities," Int. J. Prod. res., 33(7): 1791-1816, 1994.

[14] O'keefe, R. M. and Kasirajan, T., "Interaction between Dispatching and Next Station Selection Rules in a Dedicated Flexible Manufacturing System," Int. J. Prod. res., 30(8): 1753-1772, 1992.

[15] Sabuncuoclu, I., and Hommertzheim, D. L., "An Investigation of Machine and AGV Scheduling Rules in an FMS," in: K.E. Stecke, K.E. and Suri, R. (eds.), Flexible Manufacturing Systems: Operations Research Models and Applications, (Proc. 3rd ORSA/TIMS Conf.), Elsevier, Amsterdam, pp. 261-266, 1989.

[16] Sabuncuoclu, I., and Hommertzheim, D. L., "Experimental Investigation of FMS Machine and AGV Scheduling Rules against the Mean Flow-time Criterion," Int. J. Prod. Res., 30(7): 1617-1635, 1992.

[17] Shanker, Kipra, and Tzen, Y. J."A Loading and Dispatching Problem in a Random Flexible Manufacturing System," Int. J. Prod. Res., 23(3): 579-595, 1985.

[18] Solot, Philippe, and Vliet, M. V., "Analytical Models for FMS Design Optimization: A Survey, " The International Journal of Flexible Manufacturing Systems, 6: 209-233, 1994.

[19] Stecke, K. E., "Formulation and Solution of Nonlinear Integer Production Planning Problems for Flexible Manufacturing Systems," Management Science, 29(3): 273-288, 1983.

[20] Tang, L. L., Yih, Y. and Liu, C. Y., "A Study on Decision Rules of a Scheduling Model in an FMS," Computers in Industry, 22: 1-13, 1993.

[21] Tansel, B. C., and Kiran, A. S., "Optimum Central Storage Location in Flexible Manufacturing Cells," Journal of Manufacturing Systems, 7(2): 121-129, 1988.

[22] Viswanadham, N., Narahari, Y.and Johnson, T. L., "Deadlock Prevention and Deadlock Avoidance in Flexible Manufacturing Systems Using Petri Net Models," IEEE Transactions on Robotics and Automation, 6(6): 713-723, 1990.

[23] Wysk, R. A., Yang, N. S. and Joshi, S., "Detection of Deadlocks in Flexible Manufacturing Cells," IEEE Transactions on Robotics and Automation, 7(6): 853-859, 1991.

[24] Wysk, R. A., Yang, N. S. and Joshi, S."Resolution of Deadlocks in Flexible Manufacturing Systems: Avoidance and Recovery Approaches," Journal of Manufacturing Systems, 13(2): 128-138, 1994.

[25] Yamashita, H., "Approximate Analysis of Buffer Sharing Scheme in Arbitrary Configurations of Queueing Networks," IIE Transactions, 27: 332-338, 1995.

[26] Yim, D. S., and Linn, R. J., "Push and Pull Rules for Dispatching Auto-mated Guided Vehicles in a Flexible Manufacturing System," Int. J. Prod. Res, 31(1): 43-57, 1993.

[27] Zeng, L, Wang, H. P. (Ben) and Jin, S., "Conflict Detection of Automated Guided Vehicles: a Petri Net Approach," Int. J. Prod. Res., 29(5): 865-879, 1991.