

# Control Classification of Automated Guided Vehicle Systems

Mudit Sharma

**Abstract:** An automated guided vehicle or automatic guided vehicle (AGV) is a mobile robot that follows markers or wires in the floor, or uses vision or lasers. They are most often used in industrial applications to move materials around a manufacturing facility or a warehouse. Application of the automatic guided vehicle has broadened during the late 20th century and they are no longer restricted to industrial environments. Automated guided vehicle systems (AGVS) are widely used for transporting material in manufacturing and warehousing applications. These systems offer many advantages over other forms of material transport. However, the design of these systems is complex due to the interrelated decisions that must be made and the large number of system design alternatives that are available. In particular, the design of the AGVS control system can be quite challenging, and it can dramatically affect the system cost and performance. This paper presents a classification of automated guided vehicle systems developed from a control perspective. This classification is demonstrated on several example systems from the literature. This paper develops a classification scheme that provides a structured mechanism for organizing the relevant information about the design of the AGVS from a control perspective. It allows the system designer to determine how design decisions will impact the control complexity. It also provides the foundation for a design aid that will help the system designer determine the most appropriate AGVS design for a specific application.

**Index Terms:** AGV, AGVS

## I. INTRODUCTION

Automated guided vehicle systems (AGVS) are commonly used for transporting material within a manufacturing, warehousing, or distribution system. These systems provide for asynchronous movement of material through the system and are used in a wide variety of applications. They offer many advantages relative to other types of material handling systems, including reliable, automatic operation, flexibility to changes in the material handling requirements, improved positioning accuracy, reduced handling damage, easily expandable layout and system capacity, and automated interfaces with other systems (Miller, 1987). The design of AGVS, however, can be very complicated.

A number of interrelated decisions must be made including determining the guide path layout and characteristics, the number and type of vehicles, the location, type, and buffer capacities of pickup/deposit stations, the operating procedures (e.g., vehicle dispatching and routing), the type of communications, and the type and characteristics of the control system (e.g., centralized, decentralized, zone or distributed, etc.) (Bakkalbasi and McGinnis, 1988; Bohlander and Heider, 1988). There has been considerable research into these different aspects of AGVS design (e.g., see Maxwell and Muckstadt, 1982; Gaskins and Tanchoco, 1987; Kouvelis *et al.*, 1992; Sinriech and Tanchoco, 1991

and 1992; Egbelu, 1987; Tanchoco *et al.*, 1987; Kiran and Tansel, 1989; Goetz and Egbelu, 1990).

The first AGV was brought to market in the 1950s, by Barrett Electronics of Northbrook, Illinois, and at the time it was simply a tow truck that followed a wire in the floor instead of a rail. Over the years the technology has become more sophisticated and today automated vehicles are mainly Laser navigated e.g. LGV (Laser Guided Vehicle). In an automated process, LGVs are programmed to communicate (via an off board server) with other robots to ensure product is moved smoothly through the warehouse, whether it is being stored for future use or sent directly to shipping areas. Today, the AGV plays an important role in the design of new factories and warehouses, safely moving goods to their rightful destinations.

A classification scheme is needed to identify the relevant AGVS design alternatives from a *control* perspective. This paper provides such a classification. This classification shows the impact each of these decisions has on the controller design. That is, the classification identifies the controller *functionalities* required for a particular system design. The classification is useful to system designers in understanding the impacts the design decisions have on the control system so that the tradeoffs among the different design alternatives can be better evaluated. The next section provides the necessary background on AGVS control systems and their required functions. Then, the AGVS classification scheme is presented and described, and examples from the literature are categorized using the scheme. Finally, we discuss the use of the classification scheme as a first step toward the development of a useful AGVS design aid that helps a system designer understand the impact of design decisions on the control system so that performance versus complexity tradeoffs can be made to determine the most appropriate AGVS design for a particular application.

### 1.1 Agvs Controller Structure

In automated or semi-automated manufacturing systems, the AGVS controller is an integral part of the shop floor control system. The shop floor control system is responsible for routing products through the individual processing stations and interacting with the shop floor equipment and operators to affect production. The AGVS's role is to facilitate the transport of parts, tools, fixtures, etc., between individual processing centers as specified by the shop floor control system.

Joshi *et al.* (1990) and Smith *et al.* (1992) describe a hierarchical shop floor control system in which the control functions have been partitioned into planning, scheduling, and execution functions. According to this architecture, planning is responsible for determining which tasks the control system should perform. This responsibility includes decomposing tasks into smaller sub-tasks and selecting the most appropriate task when alternatives exist.

**Manuscript published on 30 October 2012.**

\* Correspondence Author (s)

**Mudit Sharma**, Department of Mechanical Engineering Mewar University, Chittorgarh Rajasthan.

© The Authors. Published by Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP). This is an [open access](http://creativecommons.org/licenses/by-nc-nd/4.0/) article under the CC-BY-NC-ND license <http://creativecommons.org/licenses/by-nc-nd/4.0/>.

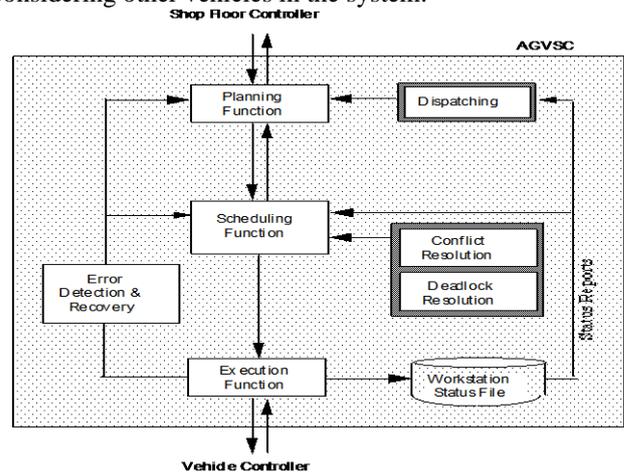
For example, in the case where multiple machines are available to process a given part, the planning function will select a specific machine. Scheduling then sequences and/or assigns start/end times to the planned tasks. Finally, execution interacts with the lower control levels or the physical equipment to perform each task. The rationale of this partitioning is that the execution function depends only on the configuration of the physical system, whereas the planning and scheduling functions also depend on the production requirements. In a flexible manufacturing system, the production requirements change much more frequently than the physical system configuration. Consequently, the planning and scheduling functions can change as required by the production requirements, and the execution functions will remain unaffected.

We view the AGV system as being made up of a supervisory controller (AGVSC) and subordinate vehicle controllers. Under this control paradigm, the vehicle controllers are responsible for the low-level drive system control (e.g., motors, transmission, etc.), and the supervisory controller is responsible for the higher-level system control functions including management of vehicle interactions. The control system functionality can be partitioned into planning, scheduling, and execution functions. The AGVSC is responsible for performing these functions. The overall structure of the AGVSC is shown in Figure 1. The individual functions of the AGVSC are described in the following paragraphs.

The AGVSC planning function is responsible for selecting an appropriate vehicle and determining the appropriate routing for that vehicle. Planning can be viewed as assigning tasks to individual vehicles, where the task identifies the path that the vehicle is to take. Planning is often referred to as *routing* and *dispatching* in the context of AGVS. Task assignment may be performed either dynamically, where the tasks are assigned to vehicles that are currently unassigned as requests for the service are received, or preplanned, where the tasks are assigned to vehicles without regard to their current assignment status (Co and Tanchoco, 1991). Egbelu and Tanchoco (1984) identify two broad categories of task assignment in the context of flexible manufacturing: workcenter-initiated, referring to vehicle selection from a set of competing idle vehicles; and vehicle-initiated, referring to the assignment of a workcenter to a vehicle from a set of competing workcenters. The study also demonstrates the effect of different assignment strategies on vehicle congestion and overall shop floor performance. Various assignment strategies have been evaluated for carrying a single load (Egbelu and Tanchoco, 1984; Russell and Tanchoco, 1984) and multiple loads (Ozden, 1988; Bartholdi and Platzman, 1989).

The scheduling function is responsible for combining all of the individual vehicles' routes into an overall sequence of vehicle segments. In other words, the planning function is responsible for breaking down the individual vehicle paths into smaller segments, and scheduling is responsible for sequencing the vehicles' access to each segment. The scheduling function is also responsible for resolving vehicle conflicts or deadlocks and generating/updating expected start and finish times for the selected routes. A number of alternate routes for a given origin/destination pair may have to be evaluated before identifying a feasible route. In this context, *feasibility* means that the selected route is not blocked and starting the vehicle along the route will not lead to unresolvable deadlock. Note that both blocking and deadlock

are dynamic problems and can only be handled by considering other vehicles in the system.



## 1.2 Detailed Schematic of AGVS Controller

Conflict in automated guided vehicle routing is said to occur when two or more vehicles are temporarily delayed if they are: (1) traveling along the same guidepath but at different speeds, or (2) arrive at the same intersection from different guidepath segments. The AGVSC must be able to resolve the conflict. Researchers have proposed several rules for resolving conflict at intersections in unidirectional, single lane/aisle, guidepath networks, such as: allowing departure of vehicles from an intersection on a first-come-first-served basis, restricting the first-come-first-served rule to vehicles transferring to the same path segment, prioritizing tasks and allowing departure based on these priorities, etc. (Egbelu and Tanchoco, 1982 and 1984). Also, Taghaboni and Tanchoco (1988) present a number of similar rules for unidirectional, two lanes/aisle guidepath networks.

Deadlock resolution is another important support function of the AGVSC, which aids the scheduling function in evaluating feasible routes. Deadlock is a situation where further movement of a set of vehicles is inhibited due to the current status of the AGV system. Researchers have identified three standard approaches to manage deadlock situations: prevention, avoidance, and detection and resolution. While the first two approaches ensure that deadlock will never occur, the last approach allows deadlock to occur and resolves it appropriately (Venkatesh *et al.*, 1994). It is the responsibility of the scheduling function to avoid or detect and resolve a deadlock in the AGVS to prevent eventual shop floor lockup.

We categorize two distinct deadlock situations that may arise in a general manufacturing system: (1) part routing deadlock; and (2) material handling deadlock. Part routing deadlock is a state when parts are assigned to various machines in such a way that any further progress of part movement is inhibited (Wysk *et al.*, 1991). Material handling deadlock is a state of deadlock when further movement of a material handling entity is inhibited due to the routing of the material handling entity (as against the routing of a part). This situation is equivalent to a traffic gridlock in a city road network. For the purposes of this paper, we are interested only in the latter form of deadlock.

Consider the situation of an unmanned manufacturing system as depicted in Figure 2(a).



The figure shows a manufacturing system with two machines ( $M_1$  and  $M_2$ ), an input/output buffer, and an AGV for material transfer between the machines and the input/output buffer. The figure depicts part 1 and part 2 currently being processed on machines  $M_1$  and  $M_2$ , respectively, with the consequent destinations for part 1 and part 2 being machine  $M_2$  and machine  $M_1$ , respectively. Clearly, any further movement is impossible without the intervention of an operator, and, hence, the system is in deadlock. Part routing deadlocks occur in unmanned systems with finite buffer capacities without proper planning and scheduling. However, several researchers have presented deadlock detection and resolution and deadlock avoidance schemes to handle part routing deadlock (Cho *et al.*, 1993; Wysk *et al.*, 1991).

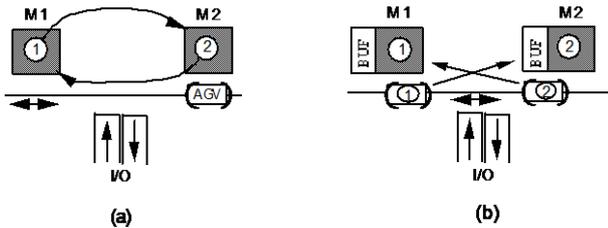


Figure 1. Deadlock Situations in Manufacturing Systems:  
(a) Part Routing Deadlock; (b) AGV Deadlock

Material handling deadlock can occur in material handling systems such as AGVS or bridge cranes due to the conflicting routes of the material transporters. Once again, consider the system shown in Figure 2, but now with two vehicles on the same guideway and adequate buffer space at each machine for the parts on the vehicles. It is not difficult to imagine a situation where the vehicles would need to move in opposing directions (Figure 2(b)). The ensuing deadlock is due to the intrinsic operation of the automated guided vehicle system and not the part routing.

Researchers have arrived at a variety of solutions to the material handling deadlock problem. These include design strategies such as building special spurs/sidings (Egbelu *et al.*, 1986; Ozden, 1988) and operational strategies such as scheduling around the deadlock (Huang *et al.*, 1989; Kim and Tanchoco, 1991; Vagnenas, 1991). Deadlock must be addressed either by preventing it at the design stage or building functionality into the controller to prevent or resolve it. The execution function is responsible for interfacing with the subordinate vehicle controllers, initiating start-up and shut-down procedures, issuing commands for assigned activities, and monitoring for error detection and recovery. As such, the execution function provides the interface between the physical system and the software control system.

### 1.3 AGVS Classification

The purpose of the classification scheme described in this paper is to identify design alternatives of the AGVS that impact the controller design. This classification shows the impact each of these decisions has on the controller design. Based on the controller structure described in the previous section, the classification identifies the controller functionalities required for a particular system design. However, these functionalities can be implemented using a variety of different control system structures.

The classification system has three basic levels as shown below.

1. Guidepath determination
  - a) Static path
    - i) Unidirectional
    - ii) Bidirectional
  - b) Dynamic path

2. Vehicle capacity
  - a) Single unit load
  - b) Multiple loads
3. Vehicle addressing mechanism
  - a) Direct address
  - b) Indirect address

Each of these levels and the choices available for each level are briefly discussed in the following sections.

## II. GUIDEPATH DETERMINATION

AGVS guidepaths are determined in one of two ways: static *a priori* determination or dynamic real-time determination. In a static guidepath system, the vehicles use a set of predetermined paths between possible origins and destinations. The vehicles can use a variety of guidance mechanisms, such as floor embedded guide wires, chemical/optical sensor stripes, dead reckoning and position updating using targets or beacons based on a software map of the paths, etc. Dynamic real-time systems use completely autonomous vehicles that are capable of determining a path using obstacle detection and avoidance systems. With dynamic paths, the vehicle is given the destination, which is a location that it knows about, perhaps specified relative to some world coordinate system. The vehicle then determines the path from its current position to the destination using its internal navigation scheme. Note that a *virtual guidepath* system (Taghaboni and Tanchoco, 1988; Gaskins *et al.*, 1989), in which there is no physical guidepath, but the supervisory controller determines the specific path from a set of fixed paths in a database, is considered a static guidepath system. In this case, the controller function is the same, although the vehicle navigation system changes significantly.

With a static guidepath system, there is a further distinction between *unidirectional* and *bidirectional* systems. In a unidirectional system, vehicles are only allowed to travel in a single direction in a given lane. An exception to this occurs when short bidirectional segments are used at load/unload stations. In this case, the vehicles enter and leave the segment from the same end and the segment is only large enough to allow one vehicle on it at any time. Therefore, the control system does not need the functionality of handling the bidirectional case. It does imply, however, that the vehicles are capable of bidirectional travel over short distances.

An aisle in the system can be divided into multiple lanes with each lane having its own direction of travel. Vehicle movement in one lane is independent of vehicle movement in another lane, e.g., the aisle is wide enough that two vehicles, one in each lane, can pass each other in the aisle. In this case, two lanes in an aisle may have opposing travel directions, but the system is still considered a unidirectional system, since each lane can be controlled independently. This restriction to unidirectional travel makes the system easier to control, since many of the deadlocking and collision avoidance problems are eliminated.

In a bidirectional system, vehicles are allowed to travel in both directions in the same lane. This functionality can be accomplished by providing turnaround points for vehicles or by using bidirectional vehicles, which are capable of moving forward and backward along the same guidepath.



Any system with at least one bidirectional segment is classified as a bidirectional system, since the controller must have the functionality to accommodate bidirectional travel. There are obvious advantages to bidirectional systems in terms of potential throughput efficiencies. Egbelu and Tanchoco (1986) have shown a marked improvement in productivity and a reduction in the number of vehicles required in bidirectional systems. However, the control of bidirectional systems is complex because of the contention of multiple vehicles for the shared guidepath segments. In a bidirectional system, the controller must be able to manage the movement of vehicles to avoid or recover from deadlock situations. Egbelu and Tanchoco (1986), Huang *et al.* (1989), and Kim and Tanchoco (1991) present methods for scheduling and deadlock resolution in bidirectional systems.

Dynamic real-time systems are not very prevalent in industrial applications. There has been, however, considerable research into the creation of autonomous vehicles and into the dynamic planning of their motion (Shiller and Gwo, 1991; Leonard *et al.*, 1992; Baumgartner and Skaar, 1994). Most of this research considers techniques for navigation of the autonomous vehicles, including using vision systems, terrain topology estimation, and ultrasonic obstacle avoidance. Development of the sensory systems required for dynamic motion planning is also an active area of research. The distinctive characteristic of these types of systems is that there is no *a priori* knowledge of paths or aiseways in the facility. The vehicle maps the region and determines the path dynamically as it executes a task. Therefore, the facility configuration can change and the AGVS control system is just updated with the new machines and the new locations relative to its coordinate system. The AGVS needs no knowledge of resources in the facility with which it does not pickup or dropoff loads. Therefore, these activities can change without affecting the AGVS control system. Although the potential exists for dynamic real-time AGVS, the authors are not aware of any reported applications.

### III. VEHICLE CAPACITY

AGVs can be classified as either single load or multiple load vehicles, depending on the number of loads that the vehicle can simultaneously carry. For our purposes, a *load* consists of a single "unit" carried by the vehicle from an origin to a destination. This unit may contain a number of distinct parts of the same or different types, e.g., assembly kits contained in a tote, but are considered a single load as long as all of the parts have the same origin and destination and the vehicle handles the tote as a unit. For AGVS with multiple vehicle types, the system will be considered a multiple load system if any of the vehicles are multiple load vehicles. The distinction for the control system lies primarily in the planning function.

In a single load system, an idle, empty vehicle is selected for a task (i.e., assigned a load to deliver). The vehicle then travels from its current position to the pickup station to obtain the load and then travels to the destination station to drop off the load. Once the vehicle is assigned the task, it is not interrupted with another task assignment (although this would be possible anytime before the vehicle has reached the pickup station). The planning function must dispatch the vehicle (assign it to a task) and determine the route for the vehicle from its current location to the task origin and then to the task destination.

In a multiple load system, the assignment of tasks to vehicles is more complicated. Partially loaded vehicles may be interrupted in their current tasks to pickup additional loads. Assigning a vehicle to a task affects not only the load for this task but all of the other loads the vehicle may currently be carrying. Therefore, the planning and scheduling functions of the controller must determine the best vehicle assignment considering all of the loads and then re plan and reschedule the vehicle's movement to integrate the new tasks into the previously assigned tasks.

### IV. VEHICLE ADDRESSING MECHANISM

AGVS can be classified as *direct* or *indirect* address systems, depending on the nature of the system operation. In an indirect address system, each vehicle visits the load/unload stations in a fixed sequence, similar to a city bus service. In this case, the routes for each vehicle are determined in advance as part of the system design and are, therefore, not part of the controller planning function. In addition, dispatching in an indirect address system is straightforward. Since the vehicle visits the stations in a prescribed order, it picks up and drops off loads as it comes to the appropriate station. The only complication is if the controller can have the vehicle "wait" at a particular station for a load to arrive. With indirect systems, the route for each vehicle may not include every station in the system. That is, the stations are partitioned such that a vehicle serves some subset of the stations. This situation introduces a different type of planning problem, namely, how to "route" a *load* through the system. It may occur that a load's origin is served by one vehicle and a load's destination is served by a different vehicle. In this case, the load must be transferred from one vehicle to another, much as people transfer between buses. Depending on the configuration of the system, the load may be handled by several vehicles before reaching its destination. In addition, there may be alternative "routes" for the load to take from its origin to its destination. The AGVS controller in an indirect address system must be able to plan this routing of loads through the system. Bartholdi and Platzman (1989) discuss the operation of a simple indirect address system, and Bozer and Srinivasan (1991) present the *tandem* AGV concept, which is an indirect address system with single-vehicle loops. The primary rationale for the single-vehicle loops is that the simplicity of the control makes the configuration attractive. Given that the tandem configuration is to be adopted, the design problems are to determine the number of loops required and to combine workcenters into individual loops. Bozer *et al.* (1989, 1991, and 1992) provide formulations of the design problem with the objectives of meeting throughput requirements for each individual loop, for partitioning the work centers into loops that meet the system throughput requirements, and for partitioning the work centers into loops to minimize the number of transfer loads for each loop. Direct address AGVS allow any vehicle to visit any station in the system. Each origin/destination pair of stations is directly served by the vehicles, much as in a city taxi service. In a direct address system, the planning function must route vehicles from their current location to their destination considering the current status of the system. Similarly, vehicles must be dispatched or assigned to tasks, since vehicles are not restricted to serving a subset of stations.

These functions are complicated since the location of the vehicle is not known in advance but changes as the status of the system changes. This creates a dynamic planning problem in which the current state of the system must be taken into account in both the dispatching and route planning functions. The intuitive advantage of direct addressing is that individual loads can be transported more quickly using a direct route rather than an indirect route in which multiple vehicles could be involved.

In controlling and designing AGV systems the problem of prevention of AGV collisions and deadlocks should be addressed. By attaching sensors on AGVs, physical collisions can be avoided.

An AGV should have the ability to avoid obstacles and the ability to return to its original path without any collisions. Xu et al. (2003) discuss concepts for this dynamic obstacle avoidance.

However, if AGVs moving in opposite directions are forced to stop in front of each other, blocking of vehicles occurs and no further transport is possible. Without manual intervention a deadlock situation is caused.

Deadlocks can also occur at buffer areas of pick-up and delivery points. If a load is available for transport at a pick-up and delivery point and a loaded AGV is the first in line before an empty AGV, then the loaded AGV cannot be unloaded and the new load cannot be transported. As a result, the performance of the system will decrease. In designing the system it is tried to avoid the occurrence of deadlock situations during operation. During operation, deadlocks and collisions are both detected and resolved by rerouting vehicles to buffer areas or deadlocks and collisions are

predicted and avoided by preplanning of routes. Detection and solving instead of avoidance of deadlocks results in a lower performance of the system. Therefore, methods have been developed to avoid collisions and deadlocks.

V. EXAMPLES

In this section, the AGVS control classification scheme described above is illustrated using examples taken from the published literature. Each AGVS design is categorized according to this scheme. The results are summarized in Table 1. In some cases, complete information is not available as a particular detail about the AGVS, which was perhaps unimportant to the published research, was omitted from the paper. Instances where a design class level was inferred from the paper are marked in the table.

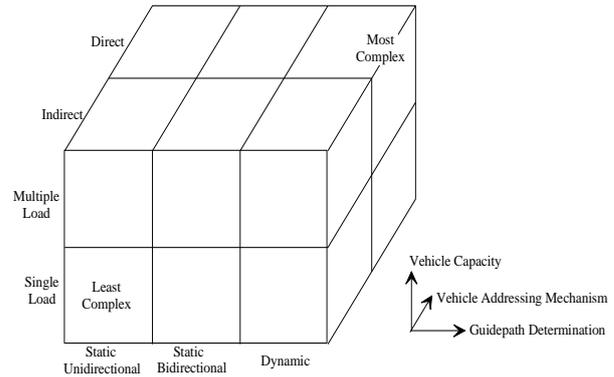


Figure 2. AGVS Classification Scheme.

Table 1. A Listing of Example AGVSC from Current Literature

AGV System Characteristics				
	Guidepath Determination (SU, SB, D)	Vehicle Capacity (S or M)	Addressing Mechanism (I or D)	Description
1.	SU	M	I	Bartholdi and Platzman (1989)—Study a single loop AGV system; develop FEFS as a simple, but effective, control rule.
2.	SB	S*	D	Huang et al. (1989)—Develop a time-window based method to route a single vehicle through a bidirectional network.
3.	SB	M	I	Ozden (1988)—Simulates an AGVS in an FMS to investigate the effect on performance of various factors such as number of vehicles, their carrying capacities, I/O buffer capacities, etc.
4.	SU	S	D	Taghaboni and Tanchoco (1988)—Develop a controller for dispatching, routing, and scheduling tasks in AGVS with virtual guidepaths and multiple travel lanes per aisle.
5.	SU	S	D	Egbelu and Tanchoco (1984)—Develop and test dispatching rules for vehicles in AGV system with and without a loop configuration.
6.	SU	S	I	Egbelu and Tanchoco (1984)—Develop and test dispatching rules for vehicles in AGV system with and without a loop configuration.
7.	SU	S*	I	Bozer et al. (1989, 1991, and 1992)—Describe a tandem system, which is a multi-loop system with a single vehicle in each loop.
8.	SU	S	I	Tanchoco and Sinriech (1992)—Develop a model to find the optimal single loop guide path for a given facility layout.
9.	SB	S	D	Kim and Tanchoco (1991)—Develop a procedure to determine the optimal vehicle route in a bidirectional AGV system.
10.	SB	M	D	Vagnenas (1991)—Describes a dispatching and control strategy for remote controlled automatic load-haul-dump (LHD) vehicles in a mining operation.
11.	SU	S	D	Miller (1987, p. 80)—Discusses an AGVS implemented at the University of Michigan Hospital for bulk supply delivery.



12.	SU	S	D	Miller (1987, p. 100)—Discusses an AGVS implemented in an Intel clean room facility for integrated circuit packaging.
13.	D	*	*	Shiller and Gwo (1991)—Discusses the motion planning of an autonomous outdoor terrain vehicle. This is not a manufacturing application of AGVS, hence the lack of information.
* Information not explicitly provided in the study, but assumed for example purposes.				

The classification scheme described above accommodates the wide variety of designs surveyed. It provides a structured mechanism for organizing the relevant information about the design of the AGVS from a control perspective. It allows the system designer to determine how design decisions will impact the control complexity.

VI. DISCUSSION

Obviously, the “best design” of an AGVS is system dependent. As illustrated by the wide variety of published results, there are a large number of models available for design analysis once a configuration has been selected. For example, given that a single-loop AGVS system is to be designed for a particular system, several of the models referenced above can identify a “good” loop configuration. However, there does not appear to be much information (other than intuitive arguments) on how to select a configuration for a given application.

This paper provides a classification of AGVS that organizes the information from a control perspective. It gives the user information about the controller complexity. As can be seen from Table 1, many systems are static, unidirectional systems with single load vehicles and direct addressing. This situation probably reflects a complexity versus performance tradeoff on the part of the system designers. For example, a bidirectional guideway provides shorter travel distances than a unidirectional guideway for the same system. These shorter distances decrease the time vehicles spend delivering loads. However, this reduced time comes at the expense of greatly increased complexity in the control system, since traffic management, collision avoidance, and deadlock detection and recovery are much more difficult in a bidirectional AGVS. Furthermore, if these control issues are not handled appropriately, the full performance benefits from the bidirectional system may not be realized.

Similarly, increased efficiencies may result from the use of vehicles capable of carrying multiple loads. However, the effort to plan and schedule these vehicles and coordinate the load movement is substantially greater than the effort required in a system with single load vehicles. The increased complexity in the control system translates into increased system cost. Therefore, the tradeoff becomes one of cost versus performance. The system designer must make these tradeoffs for each specific design situation.

The control system classification developed in this paper provides helpful information to a designer faced with a new AGVS application in making this complexity versus performance tradeoff. Future research efforts will concentrate on building a methodology that uses the classification scheme to make design decisions that explicitly consider this tradeoff between controller complexity, and hence system cost, and system performance. Such a methodology would provide the designer with tools to assist in making these important, yet complex, decisions.

VII. CONCLUSIONS

This paper presents a classification scheme for automated guided vehicle systems. This scheme is developed from a system control perspective. The paper provides a discussion of the functionalities required of a generic AGVS controller. The classification scheme is then developed based on the impact the AGVS design alternatives have on the control system. This classification scheme is illustrated with a number of examples from the literature. The scheme is useful as a structured method for understanding the impact of design decisions on the control system. It provides a mechanism for organizing the academic literature on AGVS and comparing the application domains of different techniques. It also provides helpful information to the system designer about the impact of design decisions on the required controller functionality and resulting complexity. The ultimate goal is to use the classification scheme as a design aid. Further research is needed to develop a procedure based on this classification scheme that will help a user choose the most appropriate design, from among the many possibilities, based on the requirements and characteristics of their particular application.

REFERENCES

1. Bakkalbasi, O. and McGinnis, L.F., 1988, “ABC’s of Preliminary In-House Planning and Analysis of AGVS Applications,” Proceedings of AGVS’88, MHI, Cincinnati, OH, September 27-28.
2. Bartholdi, J.J. and Platzman, L.K., 1989, “Decentralized Control of Automated Guided Vehicles on a Simple Loop,” IIE Transactions, vol. 21, no. 1, pp. 76-81.
3. Baumgartner, E.T. and Skaar, S.B., 1994, “An Autonomous Vision-based Mobile Robot,” IEEE Transactions on Automatic Control, vol. 39, pp. 493-502.
4. Biemans, F.P.M. and Vissers, C.A., 1989, “Reference Model for Manufacturing Planning and Control Systems,” Journal of Manufacturing Systems, vol. 8, no. 1, pp. 35-46.
5. Bohlander, R.A. and Heider, W., 1988, “Control Considerations When Planning AGVS Installations,” Proceedings of AGVS’88, MHI, Cincinnati, OH, September.
6. Bozer, Y.A., and Srinivasan, M.M., 1991, “Tandem Configurations for Automated Guided Vehicle Systems and the Analysis of Single-Vehicle Loops,” IIE Transactions, vol. 23, no. 1, pp. 72-82.
7. Bozer, Y.A., and Park, J.H., 1992, “New Partitioning Schemes for Tandem AGV Systems,” Proceedings of the 1992 International Material Handling Research Colloquium, Milwaukee, WI.
8. Bozer, Y.A. and Srinivasan, M.M., 1989, “Tandem Configurations for AGV Systems Offer Simplicity and Flexibility,” IE Magazine, February, pp. 23-27.
9. Christensen, J.H., Struger, O.J., Norrie, D. and Schaeffer, C., 1994, “Material Handling Requirements in Holonic Manufacturing Systems,” Proceedings of the 1994 International Material Handling Research Colloquium, Grand Rapids, Michigan, June.
10. Co, C.G. and Tanchoco, J.M.A., 1991, “A Review of Research on AGVS Vehicle Management,” Engineering Costs and Production Economics, vol. 21, pp. 35-42.



11. Egbelu, P.J., 1987, "The Use of Non-Simulation Approaches in Estimating Vehicle Requirements In An Automated Guided Vehicle Based Transport System," *Material Flow*, vol. 4, pp. 17-32.
12. Egbelu, P.J. and Tanchoco, J.M.A., 1982, "Operational Considerations for the Design of Automated Guided Vehicle Material Handling Systems," Technical Report 8201, Department of Industrial and Operations Research, Virginia Polytechnic Institute and State University, Blacksburg, VA.
13. Egbelu, P.J. and Tanchoco, J.M.A., 1984, "Characterization of Automatic Guided Vehicle Dispatching Rules," *International Journal of Production Research*, vol. 22, no.3, pp. 359-374.
14. Egbelu, P.J. and Tanchoco, J.M.A., 1986, "Potentials for Bidirectional Guidepath for Automatic Guided Vehicle Systems," *International Journal of Production Research*, vol. 24, pp. 1075-1099.
15. Gaskins, R.J. and Tanchoco, J.M.A., 1987, "Flow Path Design for Automated Guided Vehicle Systems," *International Journal of Production Research*, vol. 25, no. 5, pp. 667-676.