Robotics and Computer-Integrated Manufacturing ∎ (∎∎∎∎) ∎∎∎–∎∎∎

Contents lists available at ScienceDirect

Robotics and Computer-Integrated Manufacturing

journal homepage: www.elsevier.com/locate/rcim

Evaluation of automatic guided vehicle systems

Sigal Berman [a,b,*], Edna Schechtman [a], Yael Edan [a]

[a] *Department of Industrial Engineering and Management, Ben-Gurion University of the Negev, Beer-Sheva 84105, Israel*
[b] *Department of Industrial Engineering and Management, Judea and Samaria College, Ariel 44837, Israel*

ARTICLE INFO

Article history:
Received 30 June 2006
Received in revised form
8 December 2007
Accepted 29 February 2008

Keywords:
Performance evaluation
AGV
AGVS
Material handling
Multi robots

ABSTRACT

This paper presents a methodology for detailed evaluation of autonomous automated guided vehicles systems (AGVS) used for material handling. The methodology includes: stand-alone sub-module evaluation, including comprehensive simulations and statistical analysis of the system's sub-modules, along with hardware validation; quantitative system evaluation for integrated system performance investigation; and structured qualitative analyses for identifying strengths and weaknesses not readily apparent. The defined performance measures include aspects from both multi-robot and AGV fields. The developed methodology provides a systematic way to model, experiment with, analyze, and compare different AGVS control methods. To demonstrate the methodology, it was applied to evaluate a recently developed decentralized AGVS control method.

1. Introduction

An automated guided vehicles system (AGVS) enables flexible material routing and dispatching, and is especially suited for flexible manufacturing environments in which product mix and priorities may continuously vary [1]. The use of AGVS has increased considerably since their introduction in the 1950s, along with the number of AGVS application areas and types [2]. As the complexity and size of AGVS applications increase, evaluating performance becomes more difficult.

The last decade has also seen great progress in control methodologies of autonomous multi-robot systems [3,4]. Concepts and methodologies from the multi-robot field have migrated to AGVS control. The advent of free navigation capabilities is becoming increasingly apparent, especially when outdoor AGVS applications are considered. Performance measurement of such systems should augment concepts from both the AGVS and the multi-robot fields.

Evaluation of AGVS control methods is most often conducted using simulations (several examples can be found in [2]). AGVS performance is evaluated by measuring both parameters representing the AGVS operation and parameters representing the manufacturing system operation as a whole. Production system measures are coupled to other parameters influencing the manufacturing system, e.g., throughput is influenced by order load and plant layout. This makes the comparison between different applications problematic. Production measures used for AGVS evaluation include ([2,5]):

- Throughput.
- Manufacturing lead-time (MLT).
- The average waiting time of ready parts.
- Output queue length.
- Work in progress (WIP).
- The number of deadlock situations.
- Mean tardiness and rate of tardy parts (relative to the number of parts produced on-time).
- AGVS operation measures:
  - o AGV idle time.
  - o AGV utilization.
  - o AGV empty travel times.
  - o Number of AGVs required for meeting the production demands.

Research projects emphasizing hardware implementations of multi-robot systems are more common (several examples can be found in [6]). This is probably due to the rationale set forth by Brooks in the 1980s [7] that the real world is where systems are truly tested. Defining measures for multi-robot system is a topic of open research [3,8]. Parker [8] demonstrated the problematic quantification of architecture qualities by summarizing success indicators used in eight different implementations of ALLIANCE multi-robot architecture [9]. Performance evaluation of these

* Corresponding author at: Department of Industrial Engineering and Management, Ben-Gurion University of the Negev, Beer-Sheva 84105, Israel.
Tel.: +972 8 6479785; fax: +972 8 6472958.
E-mail address: sigalbe@bgu.ac.il (S. Berman).

0736-5845/$ - see front matter © 2008 Elsevier Ltd. All rights reserved.
doi:10.1016/j.rcim.2008.02.009

implementations warranted application-dependant measures, such as the number of objects moved, the distance traveled, etc. But these measures do not explicitly quantify the broader perspective architecture traits, such as robustness, fault tolerance, and adaptivity.

This paper presents a methodical evaluation of AGVS control methods, applying concepts from both AGVS and multi-robot fields. Accordingly, both simulation and hardware evaluation tests are conducted and measures from both fields are addressed. Considerations used in defining manufacturing scenarios for the evaluation are described. Both quantitative and qualitative analyses are presented. Evaluation concepts are illustrated using a case study. Section 2 presents the evaluation methodology, Section 3 presents the AGVS control method used as a case study, and Section 4 presents the implementation of the evaluation methodology.

2. Evaluation methodology

The evaluation process is divided into three parts (Fig. 1) and is iterative, since results obtained in one part may affect the evaluation of another:

Stand-alone sub-module evaluation—AGVS are very complex systems. As is the common engineering practice whenever possible, sub-models should be first evaluated in a stand-alone mode. AGVS functionality can be divided into three main sub-modules [10]: system management, navigation, and load transfer. Dividing the system functionality into these sub-modules reduces complexity considerably, thus facilitating detailed statistical analysis of each sub-module. Evaluation is conducted using simulation and hardware implementation wherever appropriate. The parameters used in the simulations should be based on actual values extracted from the hardware implementation. Such a choice increases the reliability of the simulations results. Additionally, hardware tests should be executed to validate the simulation results whenever possible. Simulations should be designed according to the conjectures tested. Typical simulation parameters include:

- System management—various manufacturing system sizes, AGV numbers, and production rates.
- Navigation—different factory layouts, routes, and AGV numbers.
- Load transfer—different load port entry angles.

Quantitative system evaluation—Integration-related issues are lost when each sub-module is evaluated separately. It is therefore important to evaluate sub-module operation when the system is operating as a whole. Due to the system complexity this stage is difficult to implement; yet its importance is immense. Hardware implementation of the integrated system is very costly, and deriving statistically significant results in this case may be impossible, yet it is valuable in proving the concept and in gaining insight of the system operation and correct appreciation of the interrelationships between the modules.

Qualitative system evaluation—Not all system attributes may be readily quantified. Therefore qualitative system evaluation is important to reveal underlining system qualities and characteristics. The debate regarding the relative value of quantitative and qualitative evaluation methodologies dates to back many years [11]. Yet, these two approaches are in fact complementary rather than competing, as they illuminate different aspects of the research. Hoepfl, [12] states

> *Where quantitative researchers seek casual determination, prediction, and generalization of findings, qualitative researchers seek instead illumination, understanding, and extrapolation to similar situation. Qualitative analysis results in a different type of knowledge than does quantitative inquiry.*

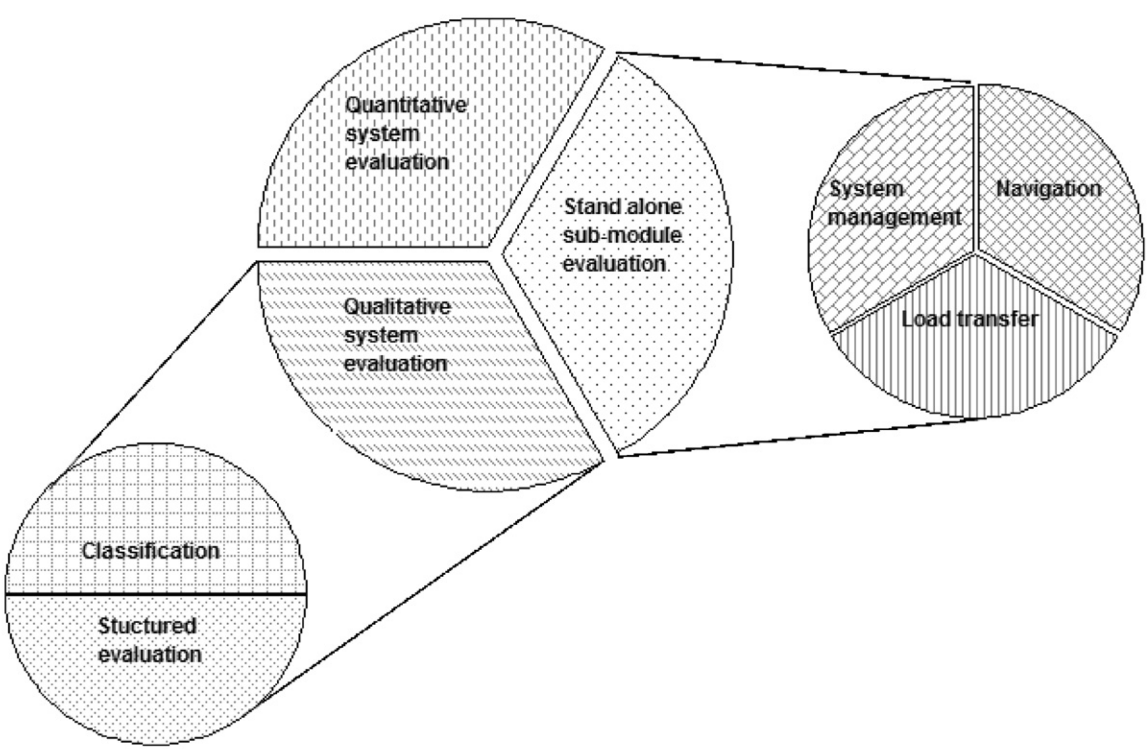

Fig. 1. AGVS evaluation process.

Structuring the qualitative evaluation process helps in maintaining focus and facilitates comparison between systems. The qualitative evaluation process is divided into two stages: classification of the system according to its attributes, followed by a qualitative assessment of the system using a formalized structure. Classification is important since it clearly represents the system complexity level, and provides a better perspective of the system's capabilities. Two classification schemes are suggested: one from the multi-robot field and one from the AGVS field. We believe that viewing the system from both perspectives helps placing its capabilities into context, and clarifies future development possibilities.

2.1. Performance measure

2.1.1. System management

In selecting the performance measures for evaluating the system management module, emphasis is made to encompass all aspects influenced by the module operation, and to decouple the measures as much as possible from other parameters influencing the manufacturing system. The selected measures are:

- The number of deadlock situations—Highlighting system-management-related problems that require operator intervention.
- Dispatching rate (the number of dispatches per hour)—Dispatching rate is used instead of the commonly used measure, throughput, to allow for products with a different number of production stages, and to account for deliveries conducted for products sent to scrap.
- To quantify the performance of the AGV allocation algorithm with respect to waiting parts two measures are used: the average waiting time of ready parts and the average orders queue length. These measures are also correlated to WIP and MLT.
- AGV utilization is difficult to quantify since establishing AGV capacity, namely the number of deliveries the AGV is capable of, is problematic. It is correlated to AGV empty travel rate (empty/loaded travel time) and AGV idle time; therefore these two measures are chosen.

2.1.2. Navigation

Three operation measures were defined for evaluation of the navigation module [13]: mission success (S), time factor (F_t), and path length factor (F_l). Measures of both conclusion time and path length are included, since an AGV may take a longer path but drive faster. The fact that path length and time are uncorrelated has also been experimentally proven [14]. All measures were normalized to facilitate comparison over different layouts and different AGV systems.

The optimal path length D_i^{st} of each run is computed according to the *a priori* shortest path found and the run-time rerouting points (points in which the AGV had to choose alternative paths)

$$D_i^{st} = \sum_{j=1}^{m} DS(j)$$

where i is the AGV identifier, s is the start point, t is the target point, m is the number of rerouting operations the AGV had to perform, DS is the shortest path computed *a priori* between the rerouting points.

Mission success (S) is defined as

$$S(i) = \begin{cases} 1 & \text{If AGV}_i \text{ reached its} \\ & \text{target without colliding} \\ 0 & \text{Otherwise} \end{cases}$$

Time factor (F_t) is defined by

$$F_t(i) = \frac{T_i^{st}}{D_i^{st}/V_{MAX}}$$

where T_i^{st} is the time it took AGV$_i$ to reach its target, V_{MAX} is the maximal velocity, path length factor (F_l) is defined by

$$F_l(i) = \frac{L_i^{st}}{D_i^{st}}$$

where L_i^{st} is the measured path length for AGV$_i$.

2.1.3. Load transfer

Measures for evaluating load transfer were not found in the literature. Due to the straightforward character of this operation, the following measures seem adequate:

- Success rate.
- Completion time.

2.2. AGVS classification

Peters [15] formalizes an AGVS control classification scheme. The classification clarifies the impact of design alternatives on the controller requirements. It is divided threefold:

(1) Guidepath determination:
  a. Static path—navigation along sets of predetermined paths typically using guidance systems:
    i. Unidirectional.
    ii. Bidirectional.
  b. Dynamic path—autonomous vehicle navigation.
(2) Vehicle capacity:
  a. Single unit load.
  b. Multiple loads.
(3) Vehicle-addressing mechanism:
  a. Indirect address—sequential dispatching. In case of segmented flow paths the load has to be transferred between vehicles.
  b. Direct address—any AGV may visit any station.

The classification helps reflect complexity versus performance tradeoffs. For example, for a given system, a bidirectional guidepath provides shorter travel distances than a unidirectional guidepath, thus reducing travel time. However, system control complexity is increased since a bidirectional controller must deal with traffic management. Similarly, multi-load vehicles may increase system efficiency, but task assignment in a multi-load system is far more complicated than in a single-load system.

2.3. Multi- robot system classification

Methodical categorization of multi-robot systems is still an open research topic [6]. Cao et al. [3] suggested a simple characterization that catches the important characteristics of multi-robot systems:

- Architecture
  - o Centralized.
  - o Decentralized.
- Differentiation of agents
  - o Homogeneous (structure and control system are similar).
  - o Heterogeneous (different).
- Communication structure

  - o Via the environment (for example by sensing a trail).
  - o Via sensing (observing other robot's actions).
  - o Via communication (intentional signaling).
- Models of other agents
  - o Intentions.
  - o Capabilities.
  - o States.
  - o Beliefs.

2.2.3. *Qualitative system analysis structure*

Arkin published one of the most comprehensive books on behavior-based robotics [6]. He suggests a list of desirable attributes of behavior-based robotic architectures, and uses this list to evaluate several behavior-based architectures. This list captures central attributes of the architecture's utility. The list includes the following criteria:

- Support for parallelism—how does the architecture support the inherent behavior-based methodology for parallelism?
- Hardware targetability—hardware targetability refers to two different things. How well does the architecture map to physical systems? What type of support is available for realizing the design using integrated circuits (IC) (since improved performance can be gained by implementation using ICs instead of software coding?)
- Niche targetability—how well can the application be tailored to the specific environment?
- Support for modularity—what methods are provided for behavior abstraction encapsulation? Abstractions provided over a wide range of behavior levels facilitate software reuse.
- Robustness—what mechanisms are provided for fault tolerance in the face of failing components (e.g., sensors, actuators)?
- Timeliness in development—what are the available development tools? Are there specific methods and tools for generating real systems?
- Run-time flexibility—how can the control be adjusted during runtime? How easily can learning and adaptation be introduced?
- Performance effectiveness—how well does the system perform its mission(s) and meet its run-time deadlines? For measurement of this notion, task-specific evaluation metrics can be applied, e.g., task completion, minimum travel.

3. AGVS control

This section presents the main concepts of the AGVS control method used as a case study to demonstrate the evaluation method. A decentralized autonomous AGVS control method was developed and implemented in a computer integrated manufacturing (CIM) system [16]. It augments concepts from both the AGVS field and the multi-robot field, and addresses all aspects of AGVS functionality: system management, navigation, and load transfer.

System management—System management is decentralized and dynamic. Each AGV gathers workstation status information directly from the workstations and dynamically decides which workstation to service. The AGVs are equipped with wireless Ethernet communication devices and communicate with the workstations via the factory intranet. AGV allocation is divided into two cases. In case the AGV is carrying a retrieved load, it will travel to the workstation the load is bound to. In case the AGV is not carrying a load it will choose a destination workstation using a weighted sum of dispatching rules.

Navigation—Navigation is based on combining behavior-based control with *a priori* shortest path calculation and way-point determination. Multi-robot coordination is introduced using coordination behavior. The behaviors are attenuated according to an *a priori* classification of the path segments, thus facilitating right of the way determination [13]. This combination provides robust and flexible real-time operation, while utilizing *a priori* knowledge to determine the shortest safe path.

Load transfer—In the vicinity of the load port the AGVs are controlled differently, utilizing improved sensory perception, since both a higher accuracy and different obstacle avoidance techniques are needed. In a multi-robot arena matters are complicated by the need to share the load port.

Hardware implementation of the AGVS is part of an ongoing project of creating an open and flexible CIM environment in the CIM laboratory at Ben-Gurion University of the Negev [17]. The CIM laboratory includes four robotic workstations: an automatic storage and retrieval station, a flexible machining station with two lathes and one mill serviced by an articulated robot on a linear slide base, an assembly station with two robots, and a quality control station with one robot and a vision system. The AGVS consists of two AGVs, 'Lachish' and 'Negev' (Fig. 2). They are Pioneer 2DX mobile platforms, 0.44 m long, 0.38 m wide, and 0.22 m tall, having a two-wheel drive along with a passive caster. Sensing capabilities include 16 front and rear sonars, bump sensors, encoders, a compass and a camera.

4. Implementation of the evaluation methodology

4.1. *Sub-module stand-alone evaluation*

Each sub-module was thoroughly evaluated in a stand-alone mode. Due to the inherent properties of each sub-module, system management was evaluated only in simulation, navigation was evaluated both in simulation and in hardware, and load transfer was evaluated only in hardware.

AGVS system management was evaluated using ARENA^{TM} simulation software [18]. Several dispatching policies (several single attribute algorithms and a weighted summation of multiple attributes) [26] were compared for various manufacturing system sizes, AGV numbers, and production rates. The data used for constructing the simulation were taken from the integrated

Fig. 2. 'Negev' (on the left) and 'Lachish', Pioneer 2DX mobile robots in the Computer Integrated Manufacturing laboratory, Ben-Gurion University of the Negev.

hardware implementation of the system. The complete analysis was conducted only in simulation due to the high complexity of such an experiment. Results of all measured parameters show that fuzzy dispatching performs as well or better than the other tested algorithms. The superiority of the fuzzy dispatching algorithm becomes significant in high-volume systems with high production rates.

AGVS navigation was evaluated using a multi-robot simulation software [13,19]. The simulation consisted of a 2-D environment populated by two distinct objects: stationary polygon objects and robots modeled after the actual robots. In the simulation, different navigation algorithms (the developed algorithm, an algorithm not utilizing *a priori* knowledge and an algorithm not utilizing multi-robot coordination mechanisms) were compared in several factory layouts, with different AGV numbers. In addition, a set of specially tailored hardware tests were conducted to validate the simulation results. The simulation and hardware runs were compared using paired *t*-tests. The success rate of the developed algorithm was significantly higher than the success rate of two other algorithms tested, with clearly non-overlapping 95% confidence intervals (Table 1).

Using one-way analysis of variance (ANOVA), we found that the difference between the path length factor and time factor for successful run of both the developed algorithm and the algorithm without multi-robot coordination behavior was not statistically significant. This may be attributed to the fact that for cases requiring multi-robot coordination the latter algorithm failed completely. On the other hand, the coordination behaviors did not cause any significant decrease in performance. Comparing the successful runs of the developed algorithm and the algorithm utilizing no *a priori* knowledge, we found that the developed algorithm has a significant advantage when path complexity is high.

In the simulation validation tests, all p-values for the time factor and path length factor were larger than 0.05, indicating a statistically non-significant difference between the simulated and the hardware runs. Yet the success rate was significantly different, with many more successful runs in the simulation (only 83% of the hardware runs were successfull compared to all of the comparable simulated runs). This indicates that the failure modeling in the simulation is faulty. It is a typically encountered result, e.g. [14], induced by the need to simplify the complex real-world environment to make the model tractable. This mismatch highlights the need for the hardware tests in the evaluation process.

Load transfer was evaluated in hardware due to the need for a very accurate model for obtaining meaningful results. The AGVs were situated in different angles at a set distance of 0.7 m in front of the load port (Fig. 3). They executed the load port entry maneuver successfully for entry angles of up to $\pm 30^\circ$.

4.2. Quantitative system evaluation

Two full system tests of the AGVS operating within the CIM production system were run. In one test the AGVS was comprised of only one AGV, and in the other the AGVS was comprised of two AGVs. The dispatching algorithm used in the full system test was based on multi-attribute rules augmented using weighted average. A random set of three production orders was chosen. Each order included three products. The orders were issued consecutively with a 20 min interval between them. This interval was set according to the part machining time in order not to overload the system. Analytical methods were used to verify that two AGVs are sufficient for servicing the manufacturing apparatus in the given conditions [20]. Each test took approximately 2 h, and both tests ended before final completion of all production rounds due to general failures not related to the AGVS operation (in the first test the university air compressor was turned off and in the second test there was a problem with Negev's chassis and Lachish's controlling laptop). Due to the complexity of the full system and since the hardware used is academic (not industry standard), a 2 h shift length is close to the consecutive operation time limit of the system. Additionally, mean time between failures-MTBF of the production system is much higher than expected MTBF in an industrial setting biasing results of the AGVS functionality tests.

Table 1
Navigation success rates (based on $n = 153$ runs)

Algorithm	Sample proportion	Lower limit of 95% confidence interval	Upper limit of 95% confidence interval
Developed	91.5	87.08	95.92
No coordination	53.59	45.69	61.5
No a priori knowledge	69.93	62.67	77.2

Fig. 3. Illustration of AGV load port entry angle (α).

System management—During the test with two AGVs, three products completed the production cycle, one completed a partial cycle and 18 successful dispatches were executed. During the test with one AGV four products completed the production cycle and 17 successful dispatches were executed. In both tests there were no lock situations and the AGVs fetched all waiting parts. Yet, the full system tests highlighted problems in the currently employed dispatching policy. The average waiting time of ready parts was relatively high with a wide distribution of waiting times (1 min–1 h with one AGV and 0–30 min with two AGVs). This suggests a sub-optimal dispatching policy. Furthermore, the empty travel time rate was relatively high. Therefore, it is expected to improve drastically after optimizing the dispatching policy. The dispatching rate is similar in both system tests. Since calculations had shown that two AGVs are sufficient for the system, we expected the dispatching rate for two AGVs to be significantly higher than the dispatching rate for one AGV. This was not the case probably for two reasons:

- The dispatching rate is influenced by the relatively short duration of the experiment. During the first batch, one AGV is enough to service the system as can be seen by the AGV idle time during this period.

- In the test with two AGVs, 'Negev' stopped functioning after 73 min (due to the sonar problem). So there was only one operating AGV when the system was most loaded.

Navigation—In the full system test with one AGV it ran according to the optimal path with 100% success rate. Two AGVs had a harder navigation problem, and success rate dropped to 89% due to one collision near the load port entry point when one AGV was trying to enter and the other trying to leave, and an AGV stumbling on an obstacle that was placed on the floor, which it could not see with its sensors. The collision with Lacish caused Negev's chassis to block its front right sonar. This made it believe that an obstacle was constantly at its side. Such an event highlights the need for improved self-diagnostic capabilities. Averages of both path length factor and time factor (exluding Negev's runs after the collision) were close to one, indicating close to optimal performance. This was also influenced by the fact that there were not many AGV–AGV encounters. The difference between Lachish's results from the test with one AGV and from the test with two AGVs is statistically insignificant both for the time factor and for path length factor. These results were verified twice by using the two sample *t*-test and the Wilcoxon rank-sum test.

Load transfer—Load port entry time depends on the initial point from which the AGV starts the entry maneuver. This point depends on the final point reached by the AGV using the navigation module. During the integrated tests the mean entry time was about 50 s and reached up to 70% of the total travel time (exit time excluded). This is rather long and it highlights a problematic issue in the system design. Such an issue can only be demonstrated in an integrated hardware test. Another issue that surfaced in the hardware test was that the entry success rate was only 83%. The deteriorated success rate as compared to the stand-alone evaluation is probably due to the fact that inaccuracies in the arrival to the final target point of the navigation module were related not only to angle but also to the distance from the load port.

4.3. Qualitative system evaluation

4.3.1. Classification

According to the AGVS control classification the presented AGVS control method is single load, dynamic, and direct, making it a fairly complex AGVS system. According to the multi-robot system characterization the system is decentralized and homogeneous. Communication via sensing is used for coordination of movement and communication via the environment (workstations) is used both for coordination (in the load port vicinity) and cooperation (sharing the work load). The robots do not maintain models of other group members. This makes the system a fairly simple multi-robot system. Applying these two classifications jointly highlights the fact that even a simple multi-robot system can make considerable contributions and implications when applied as an AGVS.

4.3.2. Structured qualitative evaluation

The process of evaluating the system using the structured evaluation process served to highlight the approaches' strengths and weaknesses. Accordingly, it served to identify future development directions.

The following strengths are identified:

- Support for modularity—comes from the hierarchical nature of the HFBB supports behavior abstraction. The decentralized control paradigm of the system supports both vehicle and station modularity.
- Run-time flexibility—the fuzzy logic algorithm used provides real-time flexibility and adaptation. Learning is currently not implemented but can be added in a straightforward manner. Fuzzy logic is amenable for integration with other soft computing methodologies such as neural networks and evolutionary algorithms [21]. De Oliveira [22] combined HFBB control with Kohonen neural nets employed for sensor fusion and situation assessment. The designated routes are currently calculated *a priori*. Due to the dynamic and uncertain nature of the environment, the shortest path may not be the optimal path. Periodic recalculation using a score parameter for each path segment updated in real-time as in Hu and Brady [23] could be easily added.
- Hardware targetability (second type)—mapping into a physical system is inherent in the structure of primitive behaviors.

The following attributes are neither strengths nor weaknesses:

- Niche targetability.
  - o Since primitive behaviors are strongly connected to hardware, it is simple to customize behaviors for different environments and robots.
  - o Software modularity enables separate adaptation of each component, e.g., load transfer module.
  - o The environment map for *a priori* planning adheres to the Saphira™ mobile robot environment [24] world generation format and is simple to edit though obtaining the measurements necessitated tedious work. One drawback of the map format is that circular objects representation is awkward.
  - o Integration of the AGVS within a CIM requires adaptation of AGV-station communication protocols.
- Robustness—fuzzy logic is good for handling real-world perturbations. But, fault detection, except low energy detection, is currently not implemented, though it is provided for by the software. Compensation mechanisms could be added using the composite behaviors.
- Timeliness in development—a simulation was implemented for expediting development. Behavior design is currently empiric and this slows development. Computerized behavior design, e.g., using genetic programming [25], may help expedite the process.

Finally, the following attributes are weaknesses:

- Support for parallelism—the hierarchical structure decreases behavior parallelism. Only behaviors sets at each single layer can run independently. The operation is arranged sequentially in a state diagram, which also decreases parallelism.
- Hardware targetability (first type)—fuzzy logic is not easily adapted to integrated circuits (IC), though there are fuzzy chips available. *A priori* calculation and generated map alleviate runtime calculations considerably and the generated routes can be implemented with ICs as a look-up-table (LUT).

5. Conclusions

A comprehensive evaluation methodology was developed for evaluating AGVS control. The methodology incorporates both quantitative analysis and structured qualitative analysis for fully exposing system qualities and limitations. It facilitates detailed analysis of both system components and their interrelationships, while augmenting concepts from the multi-robot and AGVS control.

Please cite this article as: Berman S, et al. Evaluation of automatic guided vehicle systems. Robot Comput Integr Manuf (2008), doi:10.1016/j.rcim.2008.02.009

The methodology highlights the need for both sub-module and full system evaluation and calls for using both simulation and hardware implementation for gaining full appreciation of AGVS control issues, e.g., the integrated evaluation clearly highlights the fact that the load transfer is a bottleneck of the test case system. Accordingly, efforts should be exerted to reduce the entry time and improve the entry angle calculation algorithm. Such improvements are expected to greatly improve system operations.

Simulation is an important tool expediting system development and optimization. Hardware implementation is valuable in truly proving concepts, in gaining full understanding of the correct emphasis of each aspect, and in maintaining simulation validation.

The methodology advocates the incorporation of a qualitative analysis for evaluating system qualities not readily evaluated by quantitative analysis. Structured analysis is proposed, so focus is maintained and comparison between projects is facilitated.

The full system tests were run prior to the thorough evaluation of the system management module since developing the simulation for the latter depended on data retrieved during the full system test. The dispatching algorithm used in the full system test was based on multi-attribute rules augmented using weighted average. Using simulation fuzzy dispatching proved superior to weighted average for rule augmentation. Following these findings the full system tests should be re-run using fuzzy dispatching.

Acknowledgments

This work was partially supported by the BGU Paul Ivanier Centre for Robotics and Production Management and by the Rabbi W. Gunther Plaut Chair in Manufacturing Engineering. Special thanks to: Doron Maresse, Ran Hessel, Victor Livshitz, Shmuel Epstein, Yossi Zahavi, Greg Piltz, Nissim Abuhazera, Ran Shneor, and the rest of the CIM lab team members for their contribution to this project.

References

[1] Reveliotis SA. Conflict resolution in AGV systems. IIE Trans 2000;32:647–59.
[2] IFA Vis. Survey of research in design and control of automated guided vehicle systems. Eur J Oper Res 2006;170:677–709.
[3] Cao YU, Fukunaga AS, Kahng AB. Cooperative mobile robotics: antecedents and directions. Autonomous Robots 1997;4:1–23.
[4] Arai T, Pagello E, Parker LE. Guest editorial, Special issue on advancement in multirobot systems. IEEE Trans RA 2002;18(5):655–62.
[5] Hoff EB, Sarkar BR. An overview of path design and dispatching methods for automated guided vehicles. Integrated Manuf Syst 1998;9(5):296–307.
[6] Arkin RC. Behaviour-based robotics. USA: MIT Press; 1998.
[7] Brooks RA. A robust layered control system for a mobile robot. IEEE J Robotics autom 1986;2(1):14–23.
[8] Parker LE. Evaluating success in autonomous multi-robot teams: experiences from ALLIANCE architecture implementations. J Theor Exp Artif Intell 2001;13:95–8.
[9] Parker LE. ALLIANCE: an architecture for fault tolerant multirobot cooperation. IEEE Trans Robotics Autom 1998;14(2):220–40.
[10] Koff GA. Automatic guided vehicle systems: applications, control and planning. Material Flow 1987;4:3–16.
[11] Patton M. Qualitative evaluation and research methods. 2nd ed. CA: Sage Publication Inc.; 1990.
[12] Hoepfl MC. Choosing qualitative research: a primer for technology education researchers. J Technol 1997;9:12–39.
[13] Berman S, Edan Y, Jamshidi M. Decentralized autonomous automatic guided vehicles in material handling. IEEE Trans Robotics Autom 2003;19(4):743–9.
[14] Gat E. Towards principled experimental study of autonomous mobile robots. Autonomous Robots 1995;2(3):1573–7527.
[15] Peters BA, Smith JS, Venkatesh S. A control classification of automated guided vehicle system. Int J Ind Eng 1996;3(1):29–39.
[16] Berman S, Edan Y. Decentralized autonomous AGV system for material handling. Int J Prod Res 2002;40(15):3995–4006.
[17] Berman S, Edan Y, Rabinowitz G. CIM NEGEV—a computer integrated manufacturing teaching environment. In: WFEO/ASEE e-Conf; 2003.
[18] Shneor R, Edan Y, Paz E, Naor N, Berman S. Fuzzy dispatching of automated guided vehicles. In: The Institute of Industrial Engineers Annual Conference IERC; 2006.
[19] Berman S. Decentralized autonomous automatic guided vehicle system for material handling. PhD dissertation, Ben-Gurion University of the Negev, 2002.
[20] Egbelu PJ. The use of non-simulation approaches in estimating vehicle requirements in an automated guided vehicle based transport system. Material Flow 1987;4:17–32.
[21] Akbarzadeh-T M, Jamshidi M, Tunstel E, Kumbla K. Soft computing for autonomous robotic systems. Intl J Comput Electr Eng 2000;26(1):5–32.
[22] De Oliveira MAA, Berman S, Tunstel E, Jamshidi M. Remote surface exploration with soft-computing based cooperative rovers. In: World Automation Congress (WAC); 2000.
[23] Hu H, Brady M. Dynamic path planning with uncertainty for mobile robots in manufacturing. IEEE Trans Robotics Autom 1997;13(5):760–7.
[24] Konlig K, Myers K. The saphira architecture for autonomous mobile robots. In: Kortenkamp D, Bonasso RP, Murphy R, editors. Artificial Intelligence and Mobile Robotics. USA: MIT press; 1998 [Chapter 9].
[25] Tunstel E, Jamshidi M. On genetic programming of fuzzy rule-based systems for intelligent control. Autosoft J 1996;2(3):215–22.
[26] Qiu L, Hsu WJ, Huang SY, Wang H. Scheduling and routing algorithms for AGVs: a survey. Int J Prod Res 2002;40(3):745–60.

Please cite this article as: Berman S, et al. Evaluation of automatic guided vehicle systems. Robot Comput Integr Manuf (2008), doi:10.1016/j.rcim.2008.02.009