Path Planning in Multi-AGVs Using a Modified A-star Algorithm

Munashe Zhoya[1], Xing Xu*[1]
[1]*School of Mechanical and Energy Engineering, Zhejiang University of Science and Technology, Hangzhou 310023, Zhejiang, China ;*
*Corresponding author: Professor Xing Xu
Email: xuxing3220@163.com

Abstract
The problem of path planning is a hot and exclusive research topic on multiple Automatic Guided Vehicles (multi-AGVs) systems. Many research results have been reported, but outrightly solving path planning problem from the perspective of reducing traffic congestion have faced obstacles. A collision-free path planning procedure based on a modified A-star Algorithm for multi-AGVs logistics sorting system is proposed in this paper. AGVs are now a poplar way to handle materials in latest smart warehouses. Many researches have been conducted and new technologies are still being developed. There is wide scale research on algorithms to help in scheduling, routing and path planning. Multi-AGVs are used to load goods automatically in a packaging factory. To ensure an effective and safe collision free path planning, this work investigates movement, scheduling and routing, speed manipulation and efficiency of machinery to target positions. The A-star algorithm with grid method to map out a typical warehouse scenario into multiple nodes was used. To have the shortest possible path, for obstacle avoidance, we employed the Braitenberg model. The waiting strategy is used for conflict resolution at intersections.

Key words: AGV, A-star algorithm, collision free, path planning, efficiency, throughput.

Acknowledgements: This work has been supported proudly by the Zhejiang Province Industrial Joint R&D Program Project (2019C54005).

1. Introduction

Automatic Guided Vehicles (AGVs) are effective machines used mainly in factories to run complex transportation of goods. AGVs operate in pathways defined as networks, thus they ensure quick transport of goods around specified confines of the factory. These guided vehicles were introduced in the 1950s for automatic material handling systems. Because they offer material handling, flexibility and efficiency, they have been adopted in many production lines (Gawrilow *et al.* 2008; Henesey *et al.* 2009). To employ their usage, defined routes have to be constructed in the warehouse or factory.

In a packaging factory multi-load AGVs are now increasingly being used so as to minimize human intervention and ensure quick reliable transportation of goods. The use of AGVs greatly reduces cost of production by cutting human labour usually hired for packing and moving of goods. So in line with this, path planning is a great concern, as the goods need to be moved without collision or ineffective use of time.

(Huls *et al.* 2014) described the general system of AGV components consisting of the hardware and software. Functionalities of an AGV system essentially involves task management, measuring, optimization and safety. An abstract interface for the managers and the factory workers which determines how the system should work and what connections there are, complete the system.

Multiple automated guided vehicles are characterized by many objectives and play a major role in the distribution logistics and effective material handling around work stations (Santos *et al*. 2016; Vivaldini *et al.* 2016). A determination of the number of vehicles plays a pivotal role in the management of an AGV system through path planning and constraints monitoring (Ebben, 2001). (Xia and Zheng, 2001) built a new model for an AGV system and studied the minimum number of vehicles approximated by an analytical method based on binary search. Another interesting study was carried out by (Koo *et al*. 2002) to determine AGV fleet size; the waiting time was estimated for various vehicle dispatching nodes to determine the proper AGV fleet size.

1.2 Algorithm Preview

AGVs are intricately integrated systems that require high degree of intelligence to remotely accomplish an entire control of the setup and there still exist various constraints in multi-AGV path planning. Many algorithmic strategies have been developed and implemented in a bid to solve and advance the problems associated path planning and safety in AGV systems. To note, a multi-AGV path planning with double-path constraints by using an improved genetic algorithm was developed by (Han *et al.* 2017) and the acquired results indicated that path distance and the longest single AGV path distance are shortened by using the improved genetic algorithm. (Luo *et al.* 2018) dealt with the problem of collision free path planning using the kinematic model of SIX-DOF serial manipulator constructed by using the Denavit-Hartenberg (D-H) method. The model of obstacles was defined by the axis-aligned boundary box, and the configuration space of harvesting robot was described by combining the obstacle and robot. Their developed approach showed it can effectively plan a collision-free path for the grape harvesting robot in a complex vineyard environment. With the problem of multiple overlapped and adjoining grape clusters, the path planning is still an issue that needs to be solved and requires further research.

(Yuan *et al*. 2016) developed an improved A star algorithm (A*) and yielded encouraging results. This novel method showed improved sorting efficiency of multi–AGVs and relived traffic jam. (Gochev *et al.* 2017) further developed a collision avoidance regime using the A-star algorithm and collision free avoidance method. Each AGV will have its path generated on the factory floor, to the end position the vehicle needs to reach. In the implementation of the A* algorithm, the collision avoidance and obstacles avoidance rules, collisions are restricted. (Yang *et al.* 2016) also used also A* algorithm in conjunction with the unidirectional graph method for path planning of AGV. It effectively solves the problem of conflict of AGV and is highly reliable. But with the use of the unidirectional graph method the operation efficiency of the system is compromised.

In this research work, we seek to investigate a collision free path planning, a very critical aspect in ensuring effective flow of material during production and packaging processes. Three fundamental aspects are involved, i.e., dispatching, scheduling and routing of tasks (simultaneously). We can relate the multi-AGV path planning problem to that of a travelling salesman. It has to locate the shortest time with excessively

large space search. (Smolic- Rocak *et al.* 2010) investigated multi-AGV systems using time windows in vector form to solve the shortest path problem and yielded good results.
However, there still exist various constraints in multi-AGV path planning, that is collision free constraints, time window constraints and time/distance constraints. The optimal path may not be the shortest. But because the A* algorithm is a fast path finder, which can navigate efficaciously in a planar environment, we herein use this algorithm for our investigations in this research paper. To deal effectively with the problem of collision, Braitenberg model and A-star Algorithm have been modelled so that path planning, efficiency and speed become more effective.

2. AGV Efficiency Rules

To plan the path of each AGV in a dynamic environment without properly designed algorithms, collisions, jamming and delays will happen due to the uncoordinated integration of the AGVs working at the same time in a practical operation. Therefore, when designing algorithms, storage environment in relation to its dynamic efficiency should be largely considered (Egbelu, 1984).

2.1 Scheduling

The scheduling of AGVs is to dispatch them to complete a batch of pickup/drop-off jobs to achieve best results (e.g. shortest completion time, minimum AGV idle time, etc.) under given constraints. For instance, one typical scenario is to successfully achieve all the pickup/drop-off jobs under the constraints of priority or deadline. Another typical scenario is to optimize the scheduling so that the total travel time of all vehicles is minimized, or the number of AGVs involved is minimized while the system throughput is as high as possible (Suárez *et al.* 2004). In actual fact, the decision of scheduling might involve selecting a vehicle among several idling vehicles, or selecting one load among all loads to be transported.

2.2 Routing

The aim of routing AGVs is for them to use the shortest possible time path and flexible route for every single job. All routing decisions are divided into three; firstly to detect whether there exists a route which could lead the vehicle from its origin to the destination, secondly the selected route selected must be congestion-free, conflict-free and deadlock-free, thirdly the route must minimize idling runs of vehicles (Weynes, 2005).

2.3 Efficiency

There are two general AGV control systems i.e. centralized and decentralized. The decentralized system gives the AGV system remote assignments which they should accomplish independently using their own communication systems among each other to prevent collisions. Adding more AGVs means overhead communication should also increase. On the other hand, the centralized system gives full control of how the AGV moves, which makes planning and management more easy (Nishi *et al.* 2006). To measure efficiency a lot of information has to be stored, e.g. how much work the AGVs does, in the form of throughput which is the AGV's working time against the time it is stationary. Throughput is the amount of assignments done by the vehicle in a specified time interval, tracks the metrics related to congestion, notice what causes congestion, frequency, measures how many times an AGV has to wait for another AGV or obstacle (as it is a downside) to the efficiency of the network.

2.4 Collision

Collision is a critical factor to consider when designing an algorithm for an AGV. Normally collisions come when there is no proper laying of routes in a system thereby many accidents maybe seen within a system. To deal with this firstly routes can be divided into zones, allowing one AGV to travel through a zone, which leads to deadlocks and ineffective use of resources. Another solution to prevent collision is to divide the road network into zones (Olmi, 2011). The zone control system allows only one vehicle in a zone at a time. Information is relayed to system before an AGV enters a zone. (Egemin, 2013) created sensors to prevent collision. The first thing is they make the AGV slow down, and if the obstacle hasn't moved the AGV will stop and wait to proceed when the obstacle moves.

3. The Modeling of Warehouse Environment Using Grid Method

The proposed A* algorithm works to find a path from the initial position to the target position in the environment with obstacles. Modelling of the environmental surrounding is the basis of collision-free path planning. In this work, an environmental alert model for AGV work space is established using grid method. Grid environment has the characteristics of good visibility and simple model construction, so the application of this process has reached maturity. The size and quantity of grids are determined by the size of AGV and work space. The grids are demarcated in rectangular coordinate system as shown in Table 1.

Grid map method is used for the A* Algorithm. A grid map is a way of environment mapping which is gotten by the discretization of the actual environment (Yang and Wuashan, 2016). In the cell interior, the path is the same, and the path between the adjacent grid and the grid is not continuous.

Table 1. Grid map model of a warehouse set up with 18 Nodes

N14					N5	
N13	N16	N12	N4	N7	N6	
N17			N8	N9	N15	
	N18		N1	N2	N3	
	N10	N11				

Basing on the environmental model, the AGV work process can be given as follows; firstly, the AGV will receive a tasks to carry goods to the allotted sorting area; secondly, goods which had been sorted out are carried to the original location and AGV is allocated another task. However, if there are no other tasks, the AGV will return to the pausing area.

3.1 A-star Algorithm Modification in Path Planning of multi-AGV System

3.1.1 Basic A* Algorithm

An A* algorithm has the capacity to maneuver and calculate the shortest path in real time proficiently and is extensively applicable in practical engineering. The main factors considered for the method to calculate the distance between the current point and the target point includes the actual cost, which is the cost of the path that AGV had taken, and the estimated cost, which is the cost of the path that the AGV will take (Atere and Lehtinen, 2013). The evaluation function form is given as follows:

$$f(n) = g(n) + h(n) \tag{1}$$

where $g(n)$ denotes the actual cost, $h(n)$ is the estimated cost (including empirical data).

The estimated cost is mostly used to explore search direction, which has serious influence on the ultimate search results and efficiency (Guruji *et al.* 2016). The nearer the projected cost is to the actual cost, the faster the convergences speed will be. When the projected cost is below the actual cost, the convergence speed will be slower but the optimal solution can be acquired, on the contrary the convergence speed will be faster but the optimal solution can probably not be obtained.

3.1.2 Modifying the A* Algorithm

The directions of movement for AGVs in a warehouse or factory are due north, due south, due east and due west. We used the Manhattan Distance (MD) to estimate the cost $h(n)$ on the sorting route. Estimating cost according to a given point (x_n, y_n) and the target point (x_{target}, y_{target}) is given as follows:

$$h(n) = | x_n - x_{target} | + | y_n - y_{target} | \quad (2)$$

If the Manhattan Distance is used to estimate cost, the path planning will be limited to a single static AGV (rather than in a dynamic environment), and it can result in traffic jam. To counter this and improve calculation efficiency, at the intersection, re-planning the paths of the AGVs becomes necessary, detecting all feasible paths and adding the penalty value of the paths that AGVs share. The penalty value is dependent on the distance from the AGV and is in inversely proportional to the distance. The estimated cost function as follows:

$$h(n) = | x_n - x_{target} | + | y_n - y_{target} | +$$

the other static objects in the environment such as walls, boxes, sensors and conveyor belts. The advantage of the Braitenberg approach is speed is not lost, so the motor speed is modified as in the following equation:

$$V_m = v_m + B_c + \qquad (5)$$

where is the motor speed, B_c is the Braitenberg weight coefficient, is the normalized data from the ultrasonic sensor. The normalization is the function of the minimum safety distance and the maximum detection radius.

3.3 Description of modified A* Algorithm

The A* algorithm is extended from all directions of the starting node. We select the node as *h(n)* value with respect to the second (target) nodes in all the 18 nodes, and then expand the sequence. The A* algorithm follows path planning with close similarity to reported work (Yang and Cheng, 2016; Yuan *et al*. 2016) which is essentially set up to two lists: one list is called OPEN; the other is closed down list the name being CLOSED. When the node is expanded, the existing obstacles and the existing node direction will not be extended, and stored in the closed list. A modified A* algorithm pseudo code program for computing shortest path and improved AGV throughput is displayed in Fig. 2 below.

Initialize

```
package astaralgorithm;
/**
 *
 * @author munashezhou
 */
import java.util.PriorityQueue;
import java.util.HashSet;
import java.util.Set;
import java.util.List;
import java.util.Comparator;
import java.util.ArrayList;
import java.util.Collections;
public class AStarAlgorithm {
        public static void main(String[] args){
                Node n1 = new Node("Aisle 1",366);
                Node n2 = new Node("Aisle 2",374);
                Node n3 = new Node("Aisle 3",380);
                Node n4 = new Node("Control Room",253);
                Node n5 = new Node("Empty Palletes",390);
                Node n6 = new Node("Store Room 1",193);
                Node n7 = new Node("Store Room 2",198);
                Node n8 = new Node("Aisle 4",340);
```

```
Node n9 = new Node("Aisle 5",320);
Node n10 = new Node("Stuff Offices",240);
Node n11 = new Node("Toilets",232);
Node n12 = new Node("Car Park",160);
Node n13 = new Node("Un/Loading Docks 1",180);
Node n14 = new Node("Dispatch", 200);
Node n15 = new Node("Supervisor Office",328);
Node n16 = new Node("Un/Loading Docks 2",185);
Node n17 = new Node("Lobby 1",170);
Node n18 = new Node("Lobby 2",175);
    //initialize the edges
n1.adjacencies = new Edge[]{
        new Edge(n2,75),
        new Edge(n9,140),
        new Edge(n8,118),
        new Edge(n11,115),
};
n2.adjacencies = new Edge[]{
        new Edge(n1,75),
        new Edge(n3,71),
        new Edge(n8,80),
        new Edge(n9,95),
        new Edge(n15,80),
};
```

Fig. 2. Modified A* Algorithm simulation for a typical smart warehouse

3.3.1 Simulation analysis

Table 2. Sorting efficiency between pristine and modified A* algorithm

Run Times (*k*)	Time (mins)	ASP/s of pristine A* algorithm	ASP/s of modified A* algorithm	% Change
1	60	5.44	5.92	8.82
2	120	5.38	5.76	7.06
3	180	5.32	5.72	7.52
4	240	5.25	5.64	7.43

In the simulation experiment (*k*=4) with the proposed improved A-star algorithm to perform path optimization, the throughput of multi-AGVs path planning is compared to that of the pristine A* algorithm, as shown in Table 2. The average sorting pieces per second (ASP/s) has been provided. The production scene was set with up to 10 AGVs in a predetermined warehouse area, divided into 1 × 1 grids on an attained after modification.

4. Conclusion

With smart technology (multi-AGVs) being implemented in factories and warehouses recently, advances in remote intelligence has become a necessary tool to compliment the gap. The traditional algorithm takes more time to fetch all nodes and to calculate the heuristic function values. We present a modified A* algorithmic that plans the AGV network with high safety, built using grid method. The proposed time proficient A* algorithm fetches all nodes but calculates the heuristic function values prior collision, reducing the processing time so that the AGV can perform its work quickly and safely. The paths of AGV are planned from the perspective of relieving traffic jam and avoiding collisions. The Braitenberg model has also been roped into discussion to ensure almost all limitations and hindrances are avoided in the operation of the AGVs in a dynamic environment. The three conflicts that usually arise in the multi-AGV system will be solved by the waiting strategy, the grid method and the Braitenberg model approach.

References

Atere, A. and Lehtinen, J. (2013) A Multiresolution A* Method for Robot Path Planning.
Applications of Artificial Intelligence in Engineering XII, 19, 132-137.

Braitenberg, V. (1984). Vehicles: Experiments in synthetic psychology. Cambridge, MA: MIT Press.

Ebben, M.J.R. (2001). Logistic Control in Automated Transportation Networks. Doctoral Thesis. Published by University of Twente, Enschede Gademan, A.J.R.M. van de Velde, S.L. (2000). Positioning automated guided vehicles in a loop layout. European Journal of Operational Research, 127, pp. 565-573. 7.

Egbelu, P. J, & Tanchoco, J. M. A. (1984). Characterization of Automatic Guided Vehicle; Dispatching Rules in facilities with Existing Layouts, International Journal of Production Research, 22, 3, pp. 359-374.

Egemin. Automation. (2013). Automated Guided Vehicle Safety. Retrieved from http://www.egeminusa.com/pages/agv_education/education_safety.html, visited 04-12-2019.

Gawrilow, E., Köhler, E., Möhring, R. H., & Stenzel, B. (2008). Dynamic routing of automated guided vehicles in real-time. In Mathematics– Key Technology for the Future, pp. 165-177, Springer Berlin Heidelberg.

Gochev, I., Nadzinski, G., Prof. DSc Mile Stankovski, (2017) Path Planning and Collision Avoidance Regime for a Multi-Agent System in Industrial Robotics. Faculty of Electrical Engineering and Information Technology – University of Ss Cyril and Methodius, Republic of Macedonia, Skopje.

Guruji, A.K., Agarwal, H. and Parsediya, D.K. (2016) Time-Efficient A* Algorithm for Robot Path Planning. Procedia Technology, 23, 144-149.

Han, Z., Wang, D., Liu, F., Zhao, Z. (2017). Multi-AGV path planning with double-path constraints by using an improved genetic algorithm, Plos One, 12(7), e0181747.

Henesey, L., Davidsson, P., & Persson, J. A. (2009). Evaluation of automated guided vehicle systems for container terminals using multi agent based simulation. In Multi-Agent-Based Simulation IX, pp. 85-96, Springer Berlin Heidelberg

Huls, C., Piggott, J., Windhouwer, D., Aksit, Prof. Dr. Ir. M. (2014), An Architecture for a Factory Automation System Master course: Design of Software Architecture, University of Twente, pp. 1-63.

Koo, P.H., and Jang, J.J. (2002). Vehicle Travelling Models for AGV Systems under Various Dispatching Rules. The International Journal of Flexible Manufacturing Systems, 14, pp. 249-261.

Luo, L., Wen, H., Lu, Q., Huang, H., Chen, H., Zou, X., and Wang, C. (2018) Research on the Collision-Free Path-Planning for Six-DOF Serial Harvesting Robot Based on Energy Optimal and Artificial Potential Field.

Nishi, T., Ando, M. and Konishi, M. (2006). Experimental Studies on a Local Rescheduling Procedure for Dynamic Routing Autonomous Decentralized AGV Systems. Robotics and Computer Integrate Manufacturing, 22, 154-65.

Olmi, R. (2011). Traffic Management of Automated Guided Vehicles in Flexible Manufacturing Systems. Doctoral Thesis, Università degli Studi di Ferrara.

Santos, J., Costa, P., Rocha, L., Vivaldini, K., Moreira, A. P., & Veiga, G. (2016). Validation of a Time Based Routing Algorithm Using a Realistic Automatic Warehouse Scenario. Robot 2015: Second Iberian Robotics Conference.

Smolic-Rocak, N., Bogdan, S., Kovacic, Z., Petrovic, T. (2010). Time windows based dynamic routing in Multi-AGVsystems. IEEE Transactions on Automation Science and Engineering. 7(1): pp. 151–155.

Suárez, J.I., B. M. Vinagre, F. Gutierrez, J. E. Naranjo, & Y. Q. Chen. (2004). Dynamic models of an AGV based on experimental results. IFAC Proceedings Volumes, 37(8).

Vivaldini, K., Rocha, L.F., Martarelli, N.J., Becker, M., Moreira A.P. (2016) Integrated tasks assignment and routing for the estimation of the optimal number of AGVs. The International Journal of Advanced Manufacturing Technology. 82(1): pp. 719–736.

Weyns, D., Shelfhout, K., & Holvoet, T. (2005). Architecture-centric development of an AGV transportation system, Lecture notes in Computer Science, volume 3690, pp. 640-644.

Weyns, D., Shelfhout, K., Holvoet, T., & Lefever, T. (2005). Decentralized control of EGV transportation system, Proceedings of the forth international joint conference on autonomous agents and multiagents system.

Weyns D., & Holvoet T. (2008). Architectural design of a situated multiagent system for controlling automatic guided vehicles, Int. J. Agent-Oriented Software Engineering, Vol.2, No. 1.

Xia, G.M., Zeng, J.C. (2007). A stochastic particle swarm optimization algorithm based on the genetic algorithm of roulette wheel selection, Computer Engineering and Science. 29(6), pp. 6–11.

Yang, X., and Wushan, C. (2016). AGV Path Planning Based On Smoothing A* Algorithm, College Of Mechanical Engineering, Shnghai University Of Enineering Science, China

Yuan, R.P., Dong, T.T and Li, J.T. (2016) Research on the Collision-Free Path Planning of Multi-AGVs System Based on A* Algorithm. American Journal of Operations Research, 6, 442-449.