Available online at: https://journals.researchsynergypress.com/index.php/lomr
Logistic and Operation Management Research (LOMR)
e-ISSN 2830-2680/p-ISSN 2830-3334
Volume 1 Number 1 (2022): 1-15

A Study of AGV Collaboration with IoT Concept to Avoid a Collision at Warehouse Intersection

Rizky Muftygendhis, Wei-Jung Shiang, Chia-Hung Hsu

Departement of Industrial and Systems Engineering, Taiwan

Abstract

Technology nowadays is always changing very fast. Automated tools are used to face the advanced technology in the warehouse industry, such as automated guided vehicles (AGV). It becomes a common daily vehicle when working in the warehouse, especially on the shop floor. It means collaboration between AGVs is needed and has become important. This paper suggests how to avoid collisions between AGVs and how to avoid obstacles is a practical problem if two or more AGVs interact with each other. This is a common need that should be addressed. As a result, a more effective system design should be created and implemented. This paper used a three-tier internet of things (IoT) structure to avoid collisions and reduce path deviation to design an AGV collaboration system. The IoT was used to authorize AGVs to communicate with each other by transmitting position messages. In this situation, AGV would be aware of other AGV positions and would decide what to do next. According to the findings, two AGVs can avoid colliding by knowing each other's positions in any path scenario by recognizing obstacles through Light Detection and Ranging (LIDAR) detection. This paper designed an AGV collaboration system using the Robot Operating System (ROS) and conducted an experiment to test its practicality. The experiment simulated a warehouse intersection in the real world. This study used the statistical ANOVA method to identify motion planning parameters concerning final AGV position error values.

Keywords: *Automated Guided Vehicle; Collision Avoidance; Internet of Things; Robot Operating System*

INTRODUCTION

Industry 4.0 has been adopted in many manufacturing and warehousing companies, and Cyber-Physical System (CPS) is a basic and essential technology to Industry 4.0 (Yan et al., 2020). CPS is defined as combining and managing physical assets and computational elements (Liu et al., 2019). The Internet of Things (IoT) is a subset of CPS. It is a rapidly expanding network of networked "things" with sensors that collect and share data over the internet without human intervention (Karale, 2021). The concept of CPS architecture can be utilized for the implementation of a smart Automated Guided Vehicles (AGV) system for the warehouse (Małopolski, 2018). The AGVs stand out among the most popular ways of automating operations, specifically the flow of materials (Tebaldi et al., 2021).

AGVs have been widely used in manufacturing and supply chain applications for material handling (Zhang et al., 2017). The utilization of AGV in autonomous warehouses for material handling has the potential to improve warehouse efficiency and company competitiveness (Quadrini, 2020). To provide safe and effective movement, AGVs use pattern mapping (Dundar, 2021). In the warehouse industry, AGVs have been implemented to enhance flexibility and cost savings (Małopolski, 2018).

Corresponding author
rizkymufty@gmail.com
https://doi.org/10.31098/lomr.v1i1.957

It is frequently utilized in various sectors of modern industry to increase efficiency and operation (Cheong & Lee, 2018).

Warehousing is an important part of almost every supply chain (Polten & Emde, 2021). In warehouse management systems, the IoT has been utilized to track product information in material handling (Wills & Marshall, 2016). The AGV was an important device that used IoT technology to reduce human resources and boost up the manufacturing process (Cheong & Lee, 2018). In order to guarantee the reliable functioning of AGV systems, the AGV control and management systems are getting extremely advanced, which unavoidably results in low system efficiency and reliability difficulties, especially when the number of AGVs in the system is huge (Yan et all., 2021). As a middleware framework, Robot Operating System (ROS) provides a layer between the hardware and application levels (Estefo et al., 2019). To build communication, the ROS system enables a message exchange using a publish-subscribe mechanism. (Scholl et al., 2013). Handling a large amount of data from the environment is a tough challenge, but it is crucial for enhancing the implementation of these vehicles (Alzubi et al., 2021). It aids in the coordination of AGV communication while on the job. As a result, the fundamental objective of this research is to create a communication system using ROS and to ensure optimal AGV responses through IoT architecture.

This paper aims to integrate the IoT system in the AGV motion control and add the LIDAR equipment to prevent collision at the intersection. At this phase, the AGV determines motion decisions based on the coordinates of two AGVs. AGVs expect to engage and collaborate in order to fulfill assigned tasks and enhance their movement agility. The objective of this study is to build a task-based collaboration system between two AGVs using an IoT wireless communication framework. In a warehouse, it is assumed that an AGV collaborates with another AGV to complete a task. To validate the presented design, the ROS system is combined with an IoT three-layer architecture to create a communications infrastructure.

LITERATURE REVIEW

This section describes the fundamental concepts of IOT and ROS. A common description of IoT is a self-configuring global network infrastructure based on standard and integrated communication protocols between physical and virtual 'Things' have identities, physical attributes, and virtual personalities, and use intelligent interfaces, and are seamlessly integrated into the information network (Yousuf et al., 2015). Three-tier architecture was widely used to visualize IoT applications based on the previous definition. The physical layer, which features sensors, actuators, and embedded electronics, is at the bottom layer. The network layer, which may contain Wi-Fi, Bluetooth, ZigBee, and other technologies, is the middle layer. The application layer is the top layer, which works as the host for any specific IoT applications. The physical layer is used for AGVs, the network layer is used for Wi-Fi network access, and the application layer is used for AGV motion activities in this study.

ROS is an open-source middleware platform to build advanced robotic systems and applications (Bipin, 2018). The Ubuntu and ROS is an excellent option for programming robots since a Linux-based operating system can provide a lot of flexibility in terms of interaction and customization for robot applications (Lentin, 2018). The ROS Master functions similarly to a DNS server, assigning unique names and IDs to active ROS components. The ROS Master is responsible for providing identification and registration services to the remainder of the ROS system's nodes. It keeps track of subject and service publishers and subscribers. The Master's job is to make it possible for individual ROS nodes to communicate with each other. These nodes communicate with each other peer-to-peer once they have found each other. To implement the designed of AGV collaboration system, one type of AGV is recommended, and NeuronBot is chosen and depicted in Fig. 1. The NeuronBot is an auto robotic development platform with an integrated processing unit, many sensors, and motion capability. It can be used for a variety of research, training, and instructional purposes (NeuronBot | ROS Opensource Solution | ADLINK, 2021). It was developed by ADLINK Technology, Inc. as a corporate entity and has the copyright on it. The NeuronBot using Neuron Software Development Kit, Ubuntu 18.04 as an environment, and ROS 1/ ROS 2 Intel® OpenVINO™ as a middleware.

Figure 1 Neuron Bot

METHODOLOGY

The proposed AGV collaboration system was presented and illustrated in Figure 2. The collaboration system architecture based on the IoT paradigm was made up of three layers:

1. Perception layer.
   It is a physical layer in the IoT architecture that receives the data from the AGV sensors and actuators, such as current position (coordinates), obstacle range, direction, etc. AGV performs collaborative tasks in the local environment in this layer and makes decisions based on information transmitted from the network layer.

2. Network layer.
   This layer is responsible for passing the data from the perception layer to the application layer. The ROS architecture is used to build up the communication between AGVs and the computer. The same ROS master is applied to the AGVs and the system computer for exchanging information.

3. Application layer.
   The data processing is performed in this layer. It will analyze the data to determine the distance, current position, and speed, etc. The server will generate several ROS nodes based on the task content to provide information about the AGV's current status through a network layer, and

users could set the destination of each AGV using the server. Each AGV node's data will be transmitted to the others. As a result, all AGV locations would be published to the entire system. Then, after the node was modified by an application layer, AGV would reply immediately.

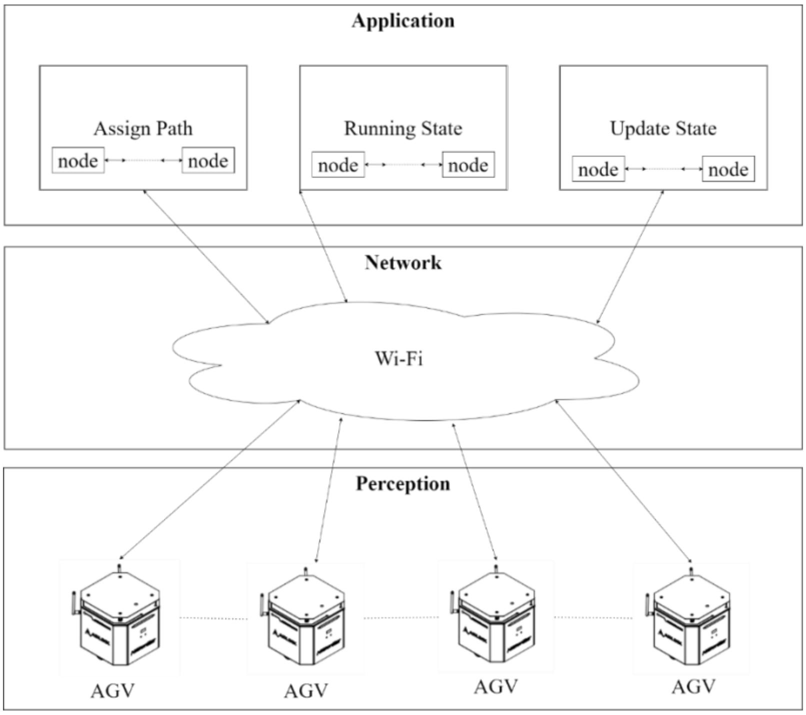

Figure 2 AGV Collaboration System Architecture

Two AGVs were used at this phase to verify the proposed collaboration system. While two ROS Masters are typically required as middleware, this study only used one ROS Master to complete the task, as illustrated in Fig. 3. By publishing and subscribing to these topics through a ROS Master, the AGV can be moved or monitored, as demonstrated in Figure 4. This study modified subject names with ROS Launch and used group1 and group2 to separate comparable topics. The main user interface topic in the ROS originally was cmd_vel.

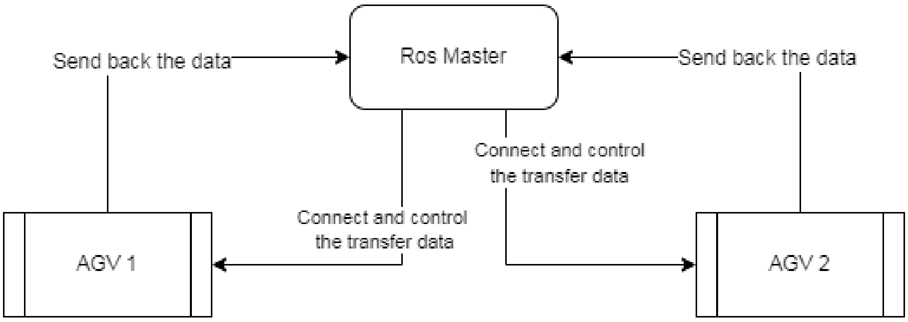

Figure 3 Wireless Communications with ROS (Hung Hsu,2021)

Group1/cmd vel would be the AGV 1 topic, and group2/cmd vel would be the AGV 2 topic because the ROS system could not use the same topic at one time. Creating different topics with the same function would resolve the issue of having duplicated subject names for the same ROS Master.

As a result, even two AGVs can be controlled or monitored with just one computer, as shown in Figure 4. Because the ROS Master is the same for all nodes, they can immediately publish or subscribe to the information in the same network. AGVs might be tracked with just one computer, regardless of how many are in the system.

The AGV collaboration system could have various functions. The design of an AGV collision-avoidance system for two AGVs provided by wheel encoders and LIDAR is explained in this work. In this research, several control rules and constraints for AGV actions are designed, and motion planning is used to help the AGV fulfill the moving task. The following flow charts (Figure 4) show the development of motion controls for the central computer and each AGV.

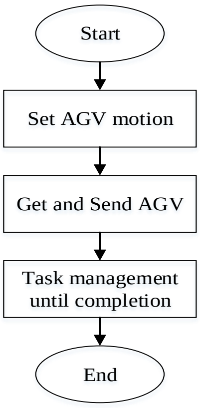

Figure 4 Control Flow Chart for Central Computer (Hung Hsu,2021)

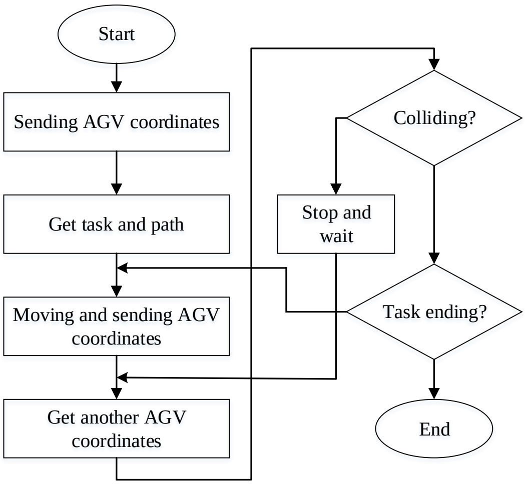

Figure 5 Basic Control Flow Chart for AGV (Hung Hsu,2021)

Figure 5 explains the basic concept of assigning a task for AGV. The central computer assigns moving tasks to each AGV and exchanges AGV coordinates and stops until the job is completed. Each AGV receives its assigned work and begins moving, deciding whether to stop or continue based on the location of other AGVs or the task status. For the avoiding obstacle, some constraints are also determined. Due to the varied tasks and pathways, the researcher determined two decision-making processes because they assumed each AGV had a different job. The design of the decision-making flowchart is shown in Figure 6 and Figure 7.

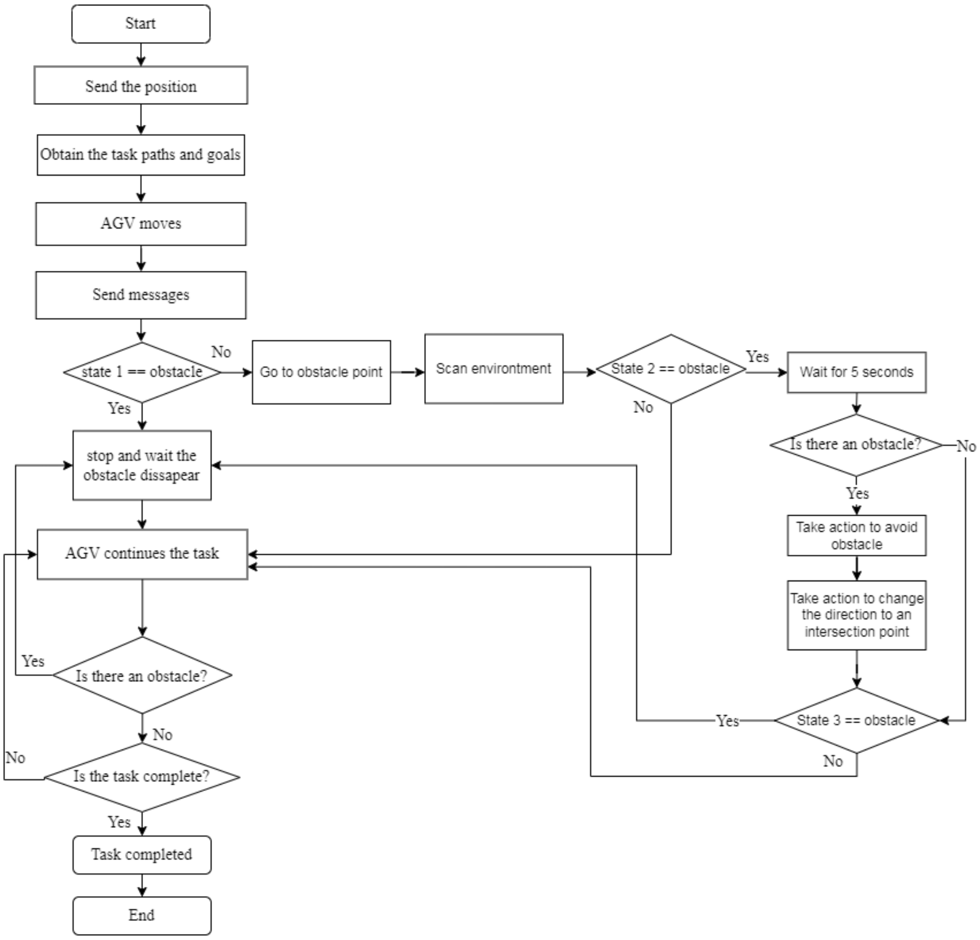

Figure 6 Obstacle Decision-Making 1

AGV 1 has three plan states to choose the movement. The decision-making process is depicted in detail in Figure 6. The aim of AGV 1 sending the current position to the server is to get the task pathways and goals from the server. AGV 1 sends the current position message to AGV 2 while on the task, and AGV 2 will begin moving when AGV 1 is close to the intersection. In Figure 7, decision-making for AGV 2 is explained.

The AGV 1 decision making divided into three states: the first state AGV 1 moves from the starting point to the obstacle point. AGV 1 moves from the obstacle point to the intersection point in the second state. Moreover, AGV 1 moves from the intersection point to the goal point in the last state to complete the task. The decision making divided into three states: the first state AGV 1 moves from the starting point to the obstacle point. AGV scans the environment. AGV 1 moves from the obstacle point to the intersection point in the second state. If the obstacle exists, it waits for 5 seconds and rescans. Then if the obstacle disappears, the AGV continues the task. AGV 1 moves from the intersection point to the goal point in the last state to complete the task, but if it meets with an obstacle, AGV 1 waits until the object is gone.

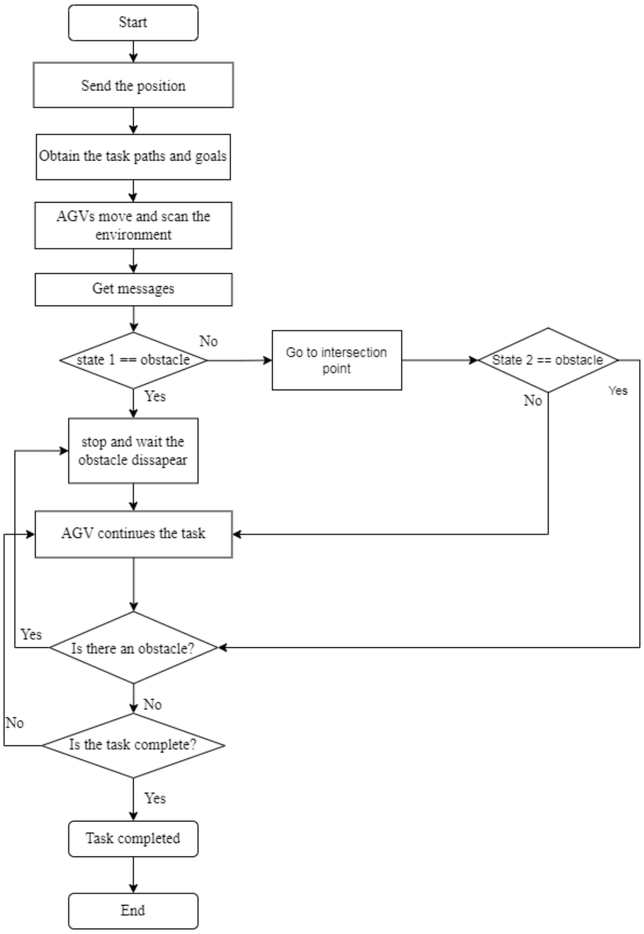

Figure 7 Obstacle Decision-Making 2

AGV 2 has a two-plan state to choose the movement. The decision-making process is depicted in detail in Figure 7. The aim of AGV 2 is the same as AGV 1. In Figure 7, decision-making for AGV 2 is explained. The AGV 2 decision making divided into two states: the first state AGV 2, moves from the starting point to the intersection point. When AGV 2 moves to complete the task, the AGV 2 scans the environment. If the obstacle exists, it waits until the object disappears, then the AGV continues the task. The second state of AGV 2 moves from the intersection point to the endpoint to finish the job. Experiments will be conducted to see if a designed collaboration system for collision avoidance is possible, and the results will be discussed in the next section.

FINDINGS AND DISCUSSION

Two AGVs were placed at two different starting positions and started moving. When one AGV reaches a certain point, it will work with other AGVs to safely pass through an intersection based on the coordinates of two AGVs. At the moment AGVs get to a certain point and meet with an obstacle, they will scan using LiDAR and take action. There were two types of an impediment, static obstacles and non-static obstacles. However, when wheel encoders are the only sensors that can calculate the movement distances of AGVs.

This paper used global and local path planning from each AGV's starting point to the intersection and ending points (goal tasks). The building floor plan for both AGVs is presented in Figure 8, while the unit is a meter. The intersection coordinate point is (0,0). AGV 1 will scan the environment when it reaches a half-meter away from the starting point, and it will decide to keep on moving and taking action or stop and wait based on what kind of obstacle is in front of the AGV.

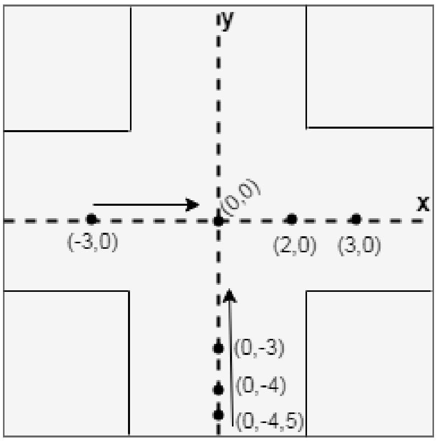

Figure 8 Global Map of the Intersection

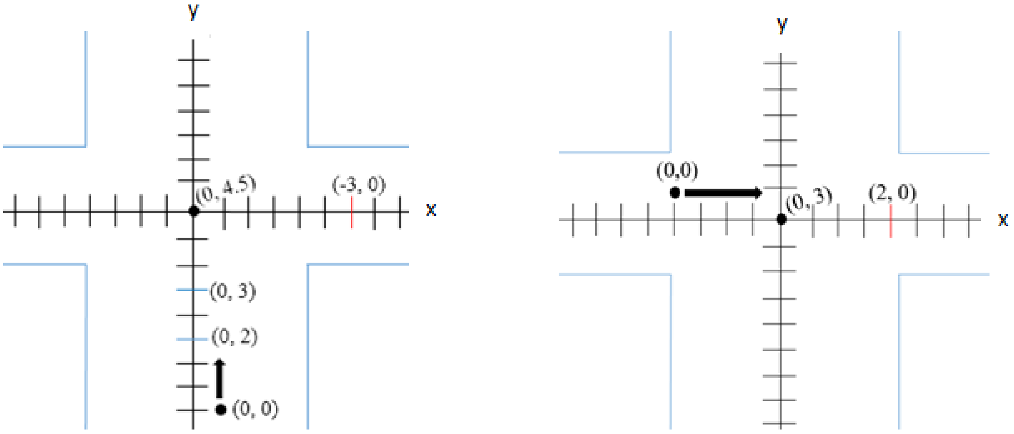

Figure 9 Local Maps and Path Planning of Two AGVs

There are three movement combinations scenarios, each of which was tested ten times to assure the system's reliability. Figure 9 shows how each AGV's path is planned. This experiment included 30 trials, all of which passed the test and distance errors to the end sites for statistical purposes. Only the findings were displayed, and it was demonstrated that when two AGVs both follow straight paths, the error is the smallest. The largest mistake may have happened when one AGV took action to avoid obstacles by turning right, turning left, turning left, and turning right.

To provide more detailed scenarios for these tests, each AGV has three motion options to avoid the obstacle:

1. AGV is working on a path and must avoid a non-static object.

2. AGV is working on a path and must avoid a static object.

3. AGV works on a path that is free of objects.

Some assumptions and limitations are mentioned below in order to carry out this experiment:

1. A fixed object has been defined as a static obstacle.
2. As a result, the AGV knows how to maneuver around obstacles.
3. The obstacle's capacity was estimated to be less than 1 meter^3.
4. It takes 5 seconds to identify an object and determine whether it is a static or non-static obstacle.
5. LIDAR has a detection range of less than one meter.
6. The AGV has specified the work plan implemented on the floor.

After the experiment was conducted, the data from AGV 1 was collected in each scenario from each trial to analyze the error coordinate differences when trying to avoid obstacles and collisions. The data result has 30 replication, and Table 1 shows the following experimental findings for the error X and Y based endpoint location coordinates of AGV 1:

Table 1 End Point Error Results (1)

Number	Scenario	X	Y
1	1	0.02	0.61
2	1	0.23	0.60
3	1	0.27	0.04
4	1	0.22	0.01
5	1	0.13	0.00
6	1	0.27	0.01
7	1	0.17	0.60
8	1	0.15	0.89
9	1	0.22	0.01
10	1	0.17	0.61
11	2	0.02	0.61
12	2	0.17	0.33
13	2	0.12	0.64
14	2	0.01	0.53
15	2	0.00	0.90
16	2	0.04	0.85
17	2	0.01	0.85
18	2	0.06	0.56
19	2	0.13	0.02
20	2	0.13	0.36

Table 2 End Point Error Results (2)

Number	Scenario	X	Y
21	3	0.23	0.02
22	3	0.23	0.02
23	3	0.23	0.04
24	3	0.23	0.04
25	3	0.20	0.02
26	3	0.18	0.62
27	3	0.25	0.01
28	3	0.17	0.29
29	3	0.14	0.61
30	3	0.27	0.01

The X and Y value in Table 1 is the data deviation or error value of coordinates from AGV 1 from each scenario in each experiment. The data were analyzed using the one-way ANOVA statistical method. It was performed to find if there is a significant difference in X or Y coordinate errors for each scenario. Table 2 and Table 3 show the results of one-way ANOVA.

Table 3 One-way ANOVA for X errors for Scenario

```
ANOVA: x versus Scenario

Factor    Type  Levels  Values

Scenario  fixed    3     1, 2, 3

Analysis of  Variance for x

Source    DF   SS          MS        F      P

Scenario   2  0.116587   0.058293  15.68  0.000

Error     27  0.100350   0.003717

Total     29  0.216936



S = 0.0609645   R-Sq = 53.74%   R-Sq(adj) = 50.32%
```

From Table 2 it can be seen that the p-value is 0.000, which is less than 0.05 and the differences between X error means in each scenario are statistically significant.

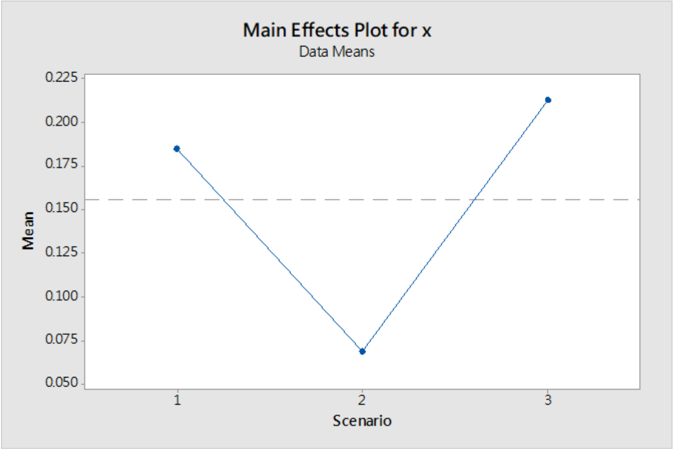

Figure 10 X Vs Scenario Chart

Figure 10 is the plot for the error means of the X coordinate, and each point represents the mean of one scenario. Scenario 1 and 3 may not has a different range too far, while the data in scenario 2 has differences, and it can conclude that the different scenarios may affect the error values of X.

Table 4 One-way ANOVA for Y errors for Scenario

```
ANOVA: y versus Scenario
Factor    Type   Levels  Values
Scenario  fixed     3    1, 2, 3

Analysis of Variance for y

Source    DF    SS       MS       F     P
Scenario   2   0.79346  0.39673  4.55  0.020
Error     27   2.35657  0.08728
Total     29   3.15003

S = 0.295433   R-Sq = 25.19%   R-Sq(adj) = 19.65%
```

From Table 3, it can be seen that the p-value is 0.020, which is less than 0.05 and the differences between Y error means in each scenario are statistically significant.

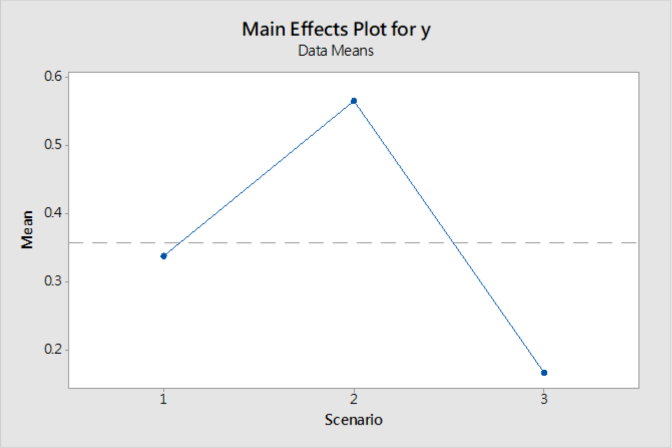

Figure 11 Y Vs Scenario Chart

Figure 11 is the plot for the error means of the Y coordinate, and each point represents the mean of one scenario. For the Y coordinates Vs. Scenario chart, it seems every scenario has significant differences, which means the different scenarios may affect the error values of Y. The data were analyzed with one-way ANOVA to know the relationship between the scenario and each coordinate. Then, the researcher used two-way ANOVA to analyze whether there is a significant relationship between errors for scenario and X,Y coordinates. The result is shown in Table 4 as a following :

Table 5 Two-way ANOVA for Errors for Scenario and X,Y

```
ANOVA: Error versus Scenario_1, XY

Factor       Type   Levels  Values

Scenario_1  fixed      3      1, 2, 3

XY          fixed      2      1, 2

Analysis of Variance for Error

Source       DF    SS        MS         F      P

Scenario_1    2  0.16082   0.08041    1.40  0.254

XY            1  0.60803   0.60803   10.62  0.002

Error        56  3.20614   0.05725

Total        59  3.97499

S = 0.239275   R-Sq = 19.34%   R-Sq(adj) = 15.02%
```

Table 4 shows the results of two-way ANOVA. The p-value for scenario errors is 0.254 and is larger than 0.05, and it means the difference between the error means of scenarios is statistically insignificant. However, the p-value for X and Y coordinate errors is 0.0020, which is less than 0.05, and the differences between X and Y error means are statistically significant.

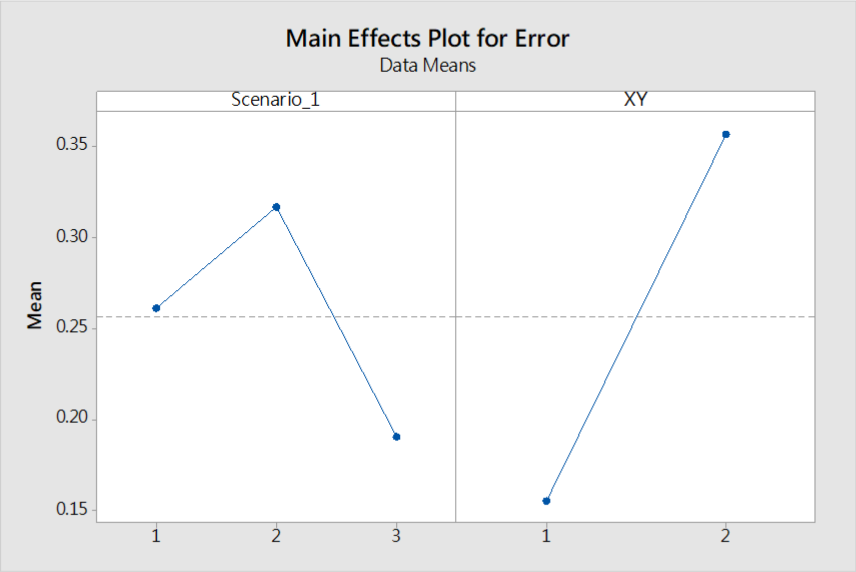

Figure 12 Error Vs Scenario and XY Chart

Figure 12 shows that the X and Y coordinate error data are analyzed together, and the error means have a larger variation between X and Y in the chart. It means that the different scenarios may not significantly affect the error values of X and Y. The control codes for the experiment need some modification to reduce the error differences between X and Y.

CONCLUSION

This study aims to design an AGV collaboration system that integrates IoT architecture with AGV actions to avoid a collision at the warehouse intersection. The collaboration system in the first phase was to build AGVs communication, and it was applied by the ROS architecture and IoT concept that were integrated to achieve real-time communication between two AGVs by sending messages. Two kinds of messages were sent by AGV 1 to AGV 2. It was a signal to start moving and stop when close to the intersection point. LIDAR is utilized in AGVs to recognize static and nonstatic obstacles. The AGV identifies the object's location and adjusts it to respond to a motion accordingly. AGVs expect to engage and collaborate in order to fulfill assigned tasks and improve their movement agility.

The AGV collaboration system was validated through an experiment. The AGV position was determined using data from the odometer sensor in this experiment. The location data were used to determine which AGV would pass first and which would stop and wait near the crossing site. The obstacle avoidance in this experiment used LIDAR to examine the surroundings and applied object detection codes to address the collision avoidance problem when AGV faced obstacles while performing work. Obstacles were divided into two categories: static and non-static obstacles. The position of a static obstacle was determined in advance; however, the position of a non-static obstacle was not fixed. Motion planning was used to develop AGV movement for certain jobs. Three scenarios with ten trial experiments each were executed to test the AGV collaboration systems when executing tasks. All trials completed the jobs without colliding, and the findings revealed that the error is the minimum when two AGVs both follow straight trajectories. Different scenarios may have different error values the X and Y coordinate individually, and this situation implies that more research into better motion control systems is required.

REFERENCES

AlZubi, A., Alarifi, A., Al-Maitah, M., & Alheyasat, O. (2021). Multi-sensor information fusion for Internet of Things assisted automated guided vehicles in smart city. Sustainable Cities And Society, 64, 102539. https://doi.org/10.1016/j.scs.2020.102539

Bipin, K. (2018). Robot Operating System cookbook. Packt Publishing.

Cheong, H., & Lee, H. (2018). Requirements of AGV (Automated Guided Vehicle) for SMEs (Small and Medium-sized Enterprises). Procedia Computer Science, 139, 91-94. https://doi.org/10.1016/j.procs.2018.10.222

Dundar, Y. (2021). Dynamic path finding method and obstacle avoidance for automated guided vehicle navigation in Industry 4.0. Procedia Computer Science, 192, 3945-3954. https://doi.org/10.1016/j.procs.2021.09.169

Estefo, P., Simmonds, J., Robbes, R., & Fabry, J. (2019). The Robot Operating System: Package reuse and community dynamics. Journal Of Systems And Software, 151, 226-242. https://doi.org/10.1016/j.jss.2019.02.024

Hung Hsu, C. (2021). A Study of AGV Collaboration for Collision Avoidance at Intersection. Chung Yuan Christian University.

Joseph, L. (2022). Robot operating system for absolute beginners. Apress.

Karale, A. (2021). The Challenges of IoT Addressing Security, Ethics, Privacy, and Laws. Internet Of Things, 15, 100420. https://doi.org/10.1016/j.iot.2021.100420

Liu, Y., Ji, S., Su, Z., & Guo, D. (2019). Multi-objective AGV scheduling in an automatic sorting system of an unmanned (intelligent) warehouse by using two adaptive genetic algorithms and a multi-adaptive genetic algorithm. PLOS ONE, 14(12), e0226161. https://doi.org/10.1371/journal.pone.0226161

Małopolski, W. (2018). A sustainable and conflict-free operation of AGVs in a square topology. Computers &Amp; Industrial Engineering, 126, 472-481. https://doi.org/10.1016/j.cie.2018.10.002

NeuronBot | ROS Opensource Solution | ADLINK. ADLINK Technology. (2021). Retrieved 17 December 2021, from https://www.adlinktech.com/Products/ROS2_Solution/ROS_Opensource_Solution/NeuronBot.

Polten, L., & Emde, S. (2021). Scheduling automated guided vehicles in very narrow aisle warehouses. Omega, 99, 102204. https://doi.org/10.1016/j.omega.2020.102204

Quadrini, W., Negri, E., & Fumagalli, L. (2020). Open Interfaces for Connecting Automated Guided Vehicles to a Fleet Management System. Procedia Manufacturing, 42, 406-413. https://doi.org/10.1016/j.promfg.2020.02.055

Scholl, P., Majoub, B., Santini, S., & Van Laerhoven, K. (2013). Connecting Wireless Sensor Networks to the Robot Operating System. Procedia Computer Science, 19, 1121-1128. https://doi.org/10.1016/j.procs.2013.06.158

Tebaldi, L., Maria, G., Volpi, A., Montanari, R., & Bottani, E. (2021). Economic evaluation of automated guided vehicles usage in a food company. Procedia Computer Science, 180, 1034-1041. https://doi.org/10.1016/j.procs.2021.01.352

Wills, R., & Marshall, A. (2016). Development of a cyber physical system for fire safety. Springer.

Yan, R., Dunnett, S., & Jackson, L. (2021). Model-Based Research for Aiding Decision-Making During the Design and Operation of Multi-Load Automated Guided Vehicle Systems. Reliability Engineering &Amp; System Safety, 219, 108264. https://doi.org/10.1016/j.ress.2021.108264

Yan, R., Jackson, L., & Dunnett, S. (2020). A study for further exploring the advantages of using multi-load automated guided vehicles. Journal Of Manufacturing Systems, 57, 19-30. https://doi.org/10.1016/j.jmsy.2020.08.005.

Yousuf, T., Mahmoud, R., Aloul, F., & Zualkernan, I. (2015). Internet of Things (IoT) Security: Current Status, Challenges and Countermeasures. International Journal For Information Security Research, 5(4), 608-616. https://doi.org/10.20533/ijisr.2042.4639.2015.0070

Zhang, Y., Zhu, Z., & Lv, J. (2018). CPS-Based Smart Control Model for Shopfloor Material Handling. IEEE Transactions On Industrial Informatics, 14(4), 1764-1775. https://doi.org/10.1109/tii.2017.2759319