Uluslararası Mühendislik Araştırma ve Geliştirme Dergisi
International Journal of Engineering Research and Development

UMAGD, (2021) 13(1), 55-65.

10.29137/umagd.686123

Cilt/Volume:13 Sayı/Issue:1 Ocak/January 2021

Araştırma Makalesi / Research Article

Implementation of Collaborative Multi-Robot System Carrying Cargos Autonomously

Emrah Budak[1], Fecir Duran[2], Meral Özarslan Yatak [3], Raif Bayır[4]

[1]UDEA Elektronik A.Ş., ODTÜ Teknokent Gümüş Bloklar K1-3, 06800, Ankara, TURKEY
[2]Gazi University, Technology Faculty, Department of Computer Engineering, 06560, Ankara, TURKEY
[3]Gazi University, Technology Faculty, Department of Electrical-Electronics Engineering, 06560, Ankara, TURKEY
[4]Karabuk University, Technology Faculty, Department of Mechatronics Engineering, , 78050, Karabuk, TURKEY

***Başvuru/Received**: 07/02/2020* ***Kabul / Accepted**: 10/10/2020* ***Çevrimiçi Basım / Published Online:** 18/01/2021*

***Son Versiyon/Final Version**: 18/01/2021*

Abstract
The paper presents implementation of collaborative multi-robot system for carrying cargo autonomously. Multi-robot systems are especially used to carry cargos to target place in the shortest way in the shortest duration by path planning. This system is composed of two robots called as Leader and Assistant. They sense the cargo with load cells on themselves and carry it to the target place. After determination of the cargo, if its weight is in the limits of the weights for Leader, it pushes the cargo by itself and Assistant waits on standby mode. If the cargo is higher than carrying capacity of Leader, Assistant is called and both push it to the target. Detecting cargos task is performed with a method similar to method of calculating fitness value. Carrying cargos task was performed by finding the shortest way with curve fitting algorithm. Carrying cargos with multi-robots by using curve fitting is the most practical solution. Consequently, reducing the route by 13.7% could be provided successfully by this algorithm instead of line following method and so energy saving was ensured. Task performance rate for carrying the cargo to the target place is achieved up to 90% for stand-alone and cooperative operation.

Keywords
"Multi-robot systems, path planning, curve fitting"

***Corresponding Author:** fduran@gazi.edu.tr*

1. Introduction

Goods and cargos are transported in factory with autonomous mobile robots, avoiding obstacles and coordinating with group mates. An autonomous robot is a very important part in industries as automotive, military, electrical, mine, chemical, etc. It is evident that using autonomous robots in industry is quite widespread especially for increasing work safety, decreasing manpower and ensuring material safety (Berglund et all. 2010, Graetz and Michaels 2018, Kermorgant 2018, Özarslan Yatak et all. 2018, Prabuwono et all. 2008, Thor et all 2015, Tominaga et all. 2017). Intelligent solutions in the field of intralogistics are required by the industrial production. Autonomous robots are adequate solution for a highly flexible, reliable and transparent material flow.

One of the ways of providing autonomy in robotics is the ability of the robot to correctly map an environment and to coordinate itself in that environment. This situation is called Simultaneous Localization and Mapping (SLAM). SLAM provides the ability to anticipate and represent its location in an environment. SLAM's performance depends on the accuracy of the environmental sensors and requires very high data processing and also communication between the robots. (Howard 2006, Koch et all. 2016, Marjovi and Marques 2011, Saeedi et all. 2011, Saeedi et all. 2015, Sun et all. 2008). SLAM is formed with some classes of algorithms such as applying Extended Kalman filter (EKF), parsing information matrices produced by generating a factor graph of observation interdependencies, camera tracking system, operating on image intensities. SLAM is employed especially for self-driving cars, planetary rovers, unmanned vehicles, and autonomous underwater vehicles.

Multi-robots are faster than a single one on complete mapping tasks and exploration. Multi-robots are designed to perform various tasks. They are often used for the places where human intervention is difficult, elimination, in large factories, storage centers, etc. (Adrian and Ribickis 2013, Hussein et all. 2014, Kim et all. 2016, Macwan et all. 2016, Madhu et all. 2017, Seçkin et all 2019). Multi-robots can realize action as synchronously with data sharing and task sharing on account of communication techniques.

Path planning lets robots find optimal path between two points. Various path planning methods for the multi-robot systems have been studied to implement the task in a short time with high accuracy (Abdi et all. 2011, Alotaibi and Al-Rawi 2016, Behzad 2019, Ganeshmurthy and Suresh 2015, Nagy 2014, Nazarahari et all. 2019, Kim et all. 2016, Thabit and Mohades 2019). Task sharing and path planning should be done when multi-robots are used in dynamic environments and distributed geographical tasks. Curve fitting method was used by Gong and the others on the B-spline curve which was used for fitting the nodes to derive a smooth feasible path such that the kinematic constraints of vehicle are taken into consideration (Gong et all. 2017).

In this study, a multi-robot system was implemented for carrying cargos and goods. The robots worked in cooperation with each other and made path planning with curve fitting method. The robots communicated each other and the center with a wireless sensor area network designed by using mesh topology. The robots used curve fitting method to select the shortest distance with minimum power expenditure for completing the task. The robots could find a collision free path beside optimality of the path with this method. After the robot scanned the area and found the coordinates of the cargos, it went to the nearest cargo. If the cargo was light as Leader could carry, it could carry the cargo to the target by itself. If the weight was too heavy for Leader, it asked for help from Assistant by using decision algorithm. The route was reduced by 13.7% with curve fitting algorithm and so energy saving was ensured. This work can be transformed into robots to be used to carry cargos under the industrial 4.0 in factories.

2. Design and Implementation of Robots

Multi-robots called as Leader and Assistant work together to carry cargos and goods. The robots were implemented similarly and consisted of a microcontroller, 6 contrast sensors, 3 distance sensors, a load cell, 2 DC motors with encoder, a motor driver, and Xbee module. Block diagram of Leader and Assistant is shown in Figure 1. The contrast sensors are for following the lines on the floor. 3 distance sensors are at right, left and in front of the robots to detect the cargo and each robot. There is a load cell with capable of measuring maximum weight of 5 kg. The robots communicate with each other on XBee Module. The motor driver can drive two motors and support PWM signals up to 100 kHz. Rotation speed of the motors is 125 Revolutions per Minute (RPM). Power Management System is for supplying power for the modules.

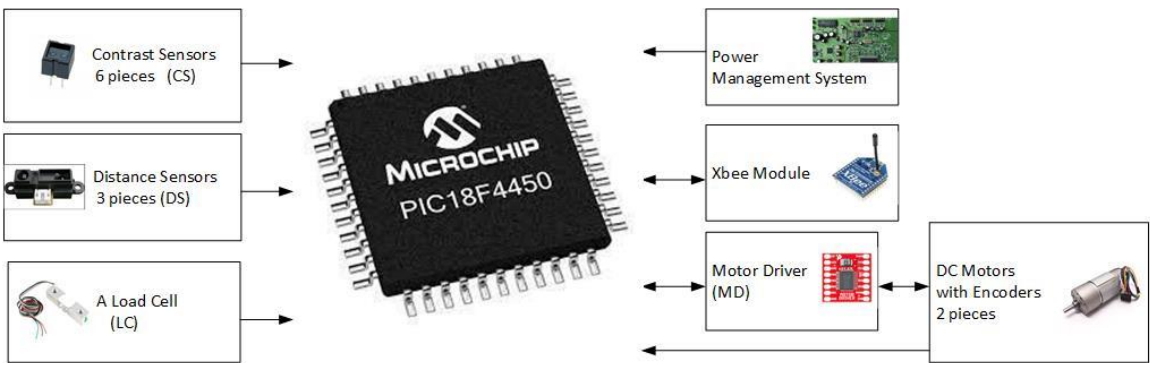

Figure 1. Block diagram of leader and assistant

ANKAKIT was designed and implemented for control of the robots. PIC18F4550 microcontroller was used on ANKAKIT. Motor driver (MD) card, serial communication module, regulator circuit, analogue reading card and pins of the contrast sensors were placed on ANKAKIT. The robots used six CNY70 contrast sensors (CS) shown in Figure 2(a). GP2D120 distance sensors (DS) were used as shown in Figure 2(b). The load cell (LC) was used for decision algorithm of calling Assistant. If the cargo was too heavy for Leader, it called Assistant. As the load cell placed in front of the robot was in an upright position according to the ground, it measured applied thrust to the cargo and amplifying the voltage regulation formed after pushing the cargo was necessary to measure this thrust data (Figure 2b). INA125 instrumentation amplifier was used for amplifying.

Robot motion was realized with two DC motors. Voltage of the motors is 6 Volts and the no-load current is 90 mA. The motors have built-in encoder structure with 90° phase-shifted two internal outputs. XBee module was used for communication and it was preferred because it is a platform that provides data transfer to all users connected to the network by creating an RF network.

The mechanical parts of the robots were designed from 2,5 mm aluminum and 3 mm Plexiglas material and they were cut with laser. Solid work drawing of the robots can be seen from Figure 2c.

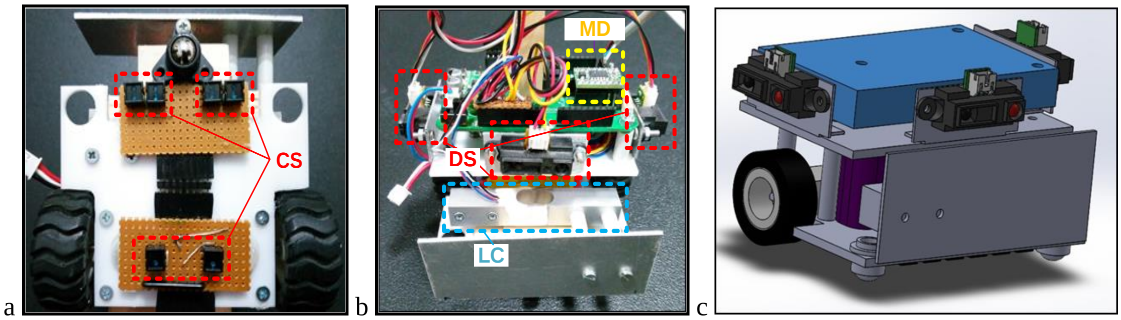

Figure 2. Implemented robot views. **(a)** Bottom view with the contrast sensors. **(b)** Top and front view with the load cell, the motor drive, and the distance sensors. **(c)** Solidworks drawing of the robot

3. Software Algorithms

Leader and Assistant Robots are desired to carry the cargo to the target in a short way with a high accuracy. The software of these two robots was coded differently. Line following, automatic angled rotations, line counting algorithms were the same, but the main algorithms were different in terms of tasks.

The task of carrying cargo is primarily Leader's responsibility. If the cargo isn't too heavy for Leader, Leader pushes it by itself, but if it is heavy, Leader calls Assistant. Firstly, Leader is starting to scan the area with data from computer. While the area is scanning, the coordinates of the cargos are determined and these coordinates are kept in memory. After the field scanning is finished, Leader turns 180° and goes to the calculated coordinates and tries to carry the cargo to the target by itself. Assistant waits outside of the area until help is asked by Leader. If the cargo is too heavy to carry, Assistant is called. Leader sends also data about the coordinate for coming of Assistant with the code for calling it. Assistant arrives at the coordinate of the cargo. The algorithm developed for Leader is shown in Figure 3.

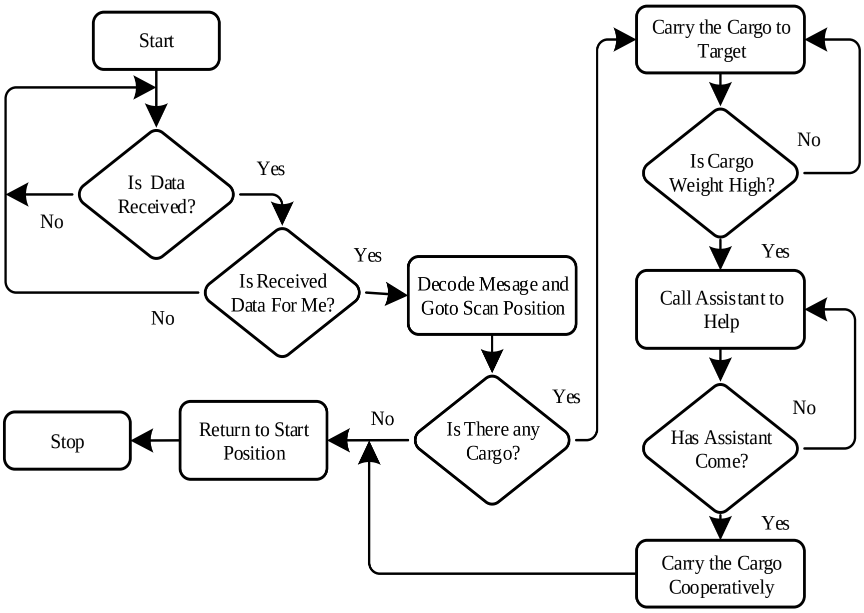

Figure 3. Flow chart of Leader's software

Assistant reaches to the coordinate it has been called to, after it receives the call for help and sends this information to Leader. Assistant and Leader carry the cargo together until the robots receive the data that the cargo is carried to the target. After the cargo is transported to the target, the robots return to the starting point in turn. The algorithm of Assistant is shown in Figure 4.

PID controller was used for the functions of line following and path trip. PID coefficients and sampling time were different from each other. The line data from the contrast sensors were used for line following and calculation of total distance covered was used for path trip by counting the step with encoder. PID controller block diagram of the robots is as shown in Figure 5. There is a switch function to determine PID inputs (Inx and Refx) and constants (Px, Ix and Dx). If the robot is switched to line following function, PID inputs become contrast sensors data and constants are updated to P1, I1, and D1. Otherwise PID inputs become encoders data and constants are updated to P2, I2, and D2.

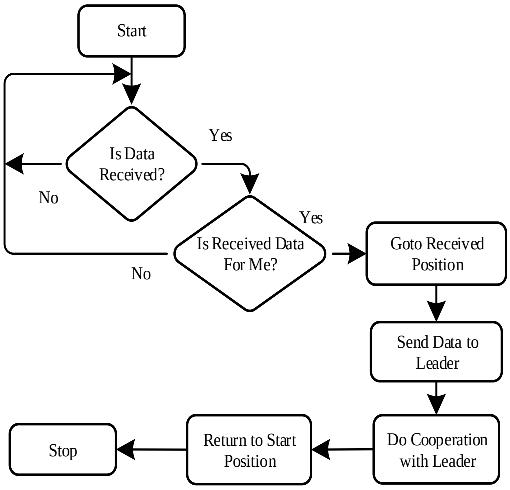

Figure. 4. Flow chart of Assistant's software

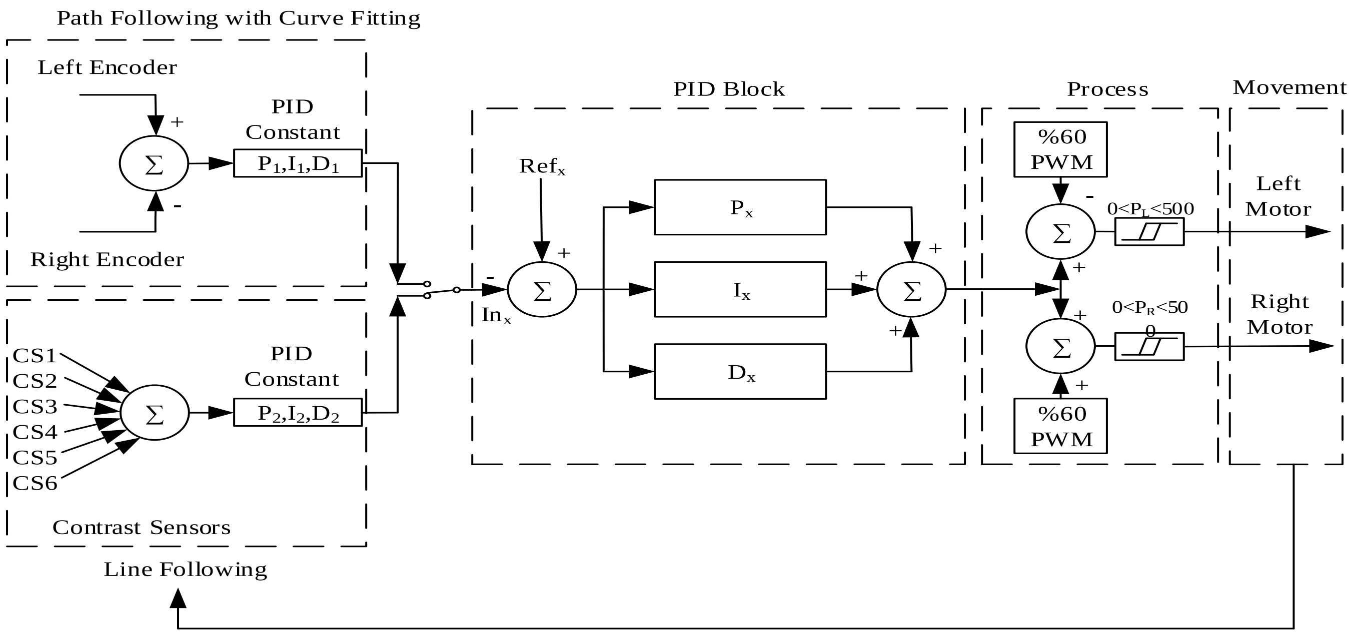

Figure 5. Robots PID block diagram for line following and path trip

Transmitter LED of the distance sensor has 30º optical faring angles and so, reflections of wide area come to the receiver of the sensor while the robot is moving. ADC of the microcontroller is set to 10 bit so as to minimize the errors. For calculation of the cargo coordinates, data from the distance sensors are looked at where they are collected on the platform.

An experimental setup was prepared for detecting the cargo place without error as shown in Figure 6. In this experiment, the sensor output values for the cargos placed at different points were recorded while the robot follows the line from -4 to +4 points on the y axis.

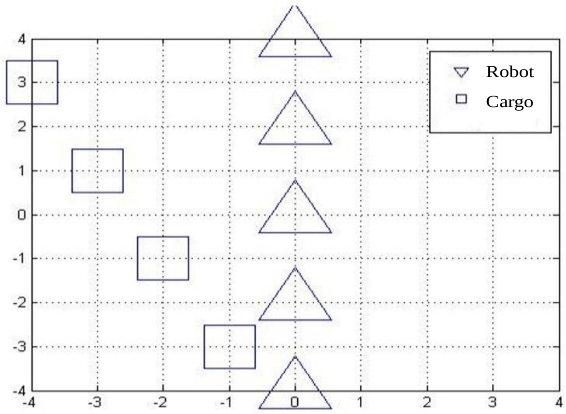

Figure 6. Cargo detection experimental setup

The obtained sensor data is as shown in Fig 7. The density of nearby values gives the knowledge that the cargo is there. As a result of 20 tests performed, the place of the cargo is correctly identified.

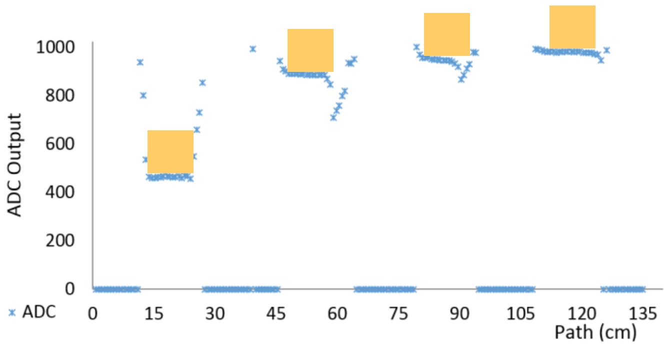

Figure 7. Outputs of distance sensors for sensing cargos

For power management system, lithium polymer battery which could provide high currents with a small size was used for the robots. A 2-cell, 7.4 Volt 900 mAh battery formed by serial connection of Li-Po cells with each cell 3.7V voltage was preferred. The robots measured the voltage periodically and calculated how much of the total power was consumed by comparing the battery with the values in the discharge curve.

The robots estimate the power consumption required for the task and produce data on whether performing the task with this power or not by comparing the total power remaining in the battery to the estimation before performing the task from the center or the robot members. If it does not have power to perform the task, it sends information that it cannot perform the task.

4. Path Planning with Curve Fitting

Curve fitting is an analysis method for using to measure the relationship between two or more variables. If analysis is performed using a single variable, this is called univariate regression. If multiple variables are used, it is called multivariate regression analysis. The least squares method is used to write the mathematical relation between two physical quantities, which depend on each other, as a true equation as much as possible. In other words, this method helps to find a function curve that will pass "as close as possible" to the data points resulting from the measurement.

In this study, linear sampling method which is a kind of least squares method was used with two variables. The linear samples were expressed as $y = mx + n$. m and n values in the equation can be defined as the most accurate values to draw the line. To do this, sum of squares of distances between the $(x_i, \overline{y_i})$ points on $y = mx + n$ line and the (x_i, y_i) scatter points are calculated. When m and n data are found, the regression line is found.

$$S = \sum_{i=0}^{k} (mx_i + n - y_i)^2 \tag{1}$$

The prerequisite for finding m and n values are that the distance of the straight line to the data values entering the equilibrium must be the smallest of the squares. To calculate m and n values which make S minimum, the partial derivative of S by m (2) and the partial derivative of S by n (3) should equal to zero.

$$\frac{\partial s}{\partial m} = \sum_{i=1}^{k} (mx_i + n - y_i)x_i = 0 \tag{2}$$

$$\frac{\partial S}{\partial n} = \sum_{i=1}^{k}(mx_i + n - y_i) = 0 \tag{3}$$

This equation system is reorganized as Eq. 4-10 and the m and n values are calculated.

$$m \sum_{i=1}^{k} {x_i}^2 + n\sum_{i=1}^{k} x_i - \sum_{i=1}^{k} x_i y_i = 0 \tag{4}$$

$$m\sum_{i=1}^{k} x_i + n\sum_{i=1}^{k} 1 - \sum_{i=1}^{k} y_i = 0 \tag{5}$$

$$\begin{pmatrix} \sum_{i=1}^{k} {x_i}^2 & \sum_{i=1}^{k} x_i \\ \sum_{i=1}^{k} x_i & \mathrm{k} \end{pmatrix}\begin{pmatrix} m \\ n \end{pmatrix} = \begin{pmatrix} \sum_{i=1}^{k} x_i y_i \\ \sum_{i=1}^{k} y_i \end{pmatrix} \tag{6}$$

$$m = \frac{\begin{vmatrix} \sum_{i=1}^{k} x_i y_i & \sum_{i=1}^{k} x_i \\ \sum_{i=1}^{k} y_i & \mathrm{k} \end{vmatrix}}{\begin{vmatrix} \sum_{i=1}^{k} {x_i}^2 & \sum_{i=1}^{k} x_i \\ \sum_{i=1}^{k} x_i & \mathrm{k} \end{vmatrix}} \tag{7}$$

$$n = \frac{\begin{vmatrix} \sum_{i=1}^{k} {x_i}^2 & \sum_{i=1}^{k} x_i y_i \\ \sum_{i=1}^{k} x_i & \mathrm{k} \end{vmatrix}}{\begin{vmatrix} \sum_{i=1}^{k} {x_i}^2 & \sum_{i=1}^{k} y_i \\ \sum_{i=1}^{k} x_i & \mathrm{k} \end{vmatrix}} \tag{8}$$

$$m = \frac{\mathrm{k}\sum_{i=1}^{k} x_i y_i - \sum_{i=1}^{k} x_i \sum_{i=1}^{k} y_i}{\mathrm{k}\sum_{i=1}^{k} {x_i}^2 - (\sum_{i=1}^{k} x_i)^2} \tag{9}$$

$$n = \frac{\mathrm{k}\sum_{i=1}^{k} y_i \sum_{i=1}^{k} {x_i}^2 - \sum_{i=1}^{k} x_i \sum_{i=1}^{k} x_i y_i}{\mathrm{k}\sum_{i=1}^{k} {x_i}^2 - (\sum_{i=1}^{k} x_i)^2} \tag{10}$$

The cargo coordinates are entered in the mathematical equations and a straight line is drawn between the robot and the cargo. After acquiring the necessary angle data, the robot rotates to the calculated angle with the information it receives from the encoders and moves towards the cargo (Figure 8).

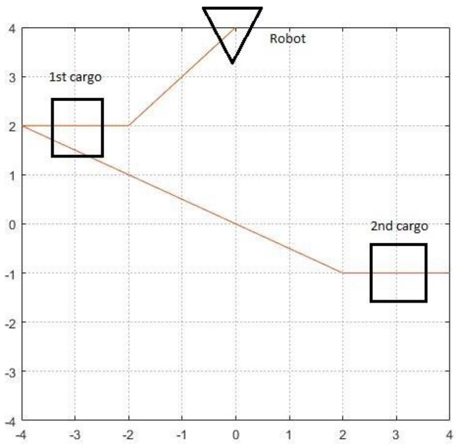

Figure 8. The straight line drawn between the robot and the cargo with curve fitting method

5. Experimental Setup and Communication Protocol

5.1. Experimental Setup

The platform where the system tests are performed is shown in Fig 9. This platform consists of 9 vertical and horizontal lines spaced at 150 mm intervals with a white band of 18 mm thickness on a black area of one and a half square meters.

Leader and Assistant get in platform “B” from entrance "A" .Leader finds the coordinates of the cargos and carries them to the target "C" area.

Leader moves on from A until vertical lines run out, detects and records the cargos on the right and left sides. After the scanning process is finished, it returns to the coordinates where the cargos are located and carries the cargos to the C area except for too heavy weights. After sensing the heavy load by the load cell, Leader asks Assistant for help by sending the coordinates and waits for help. When Leader receives the data on coming help through the communication, it pushes the cargo with Assistant to the C area.

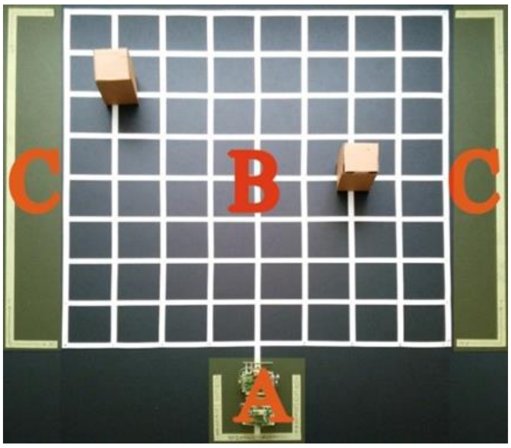

Figure 9. Test platform

5.2. Communication Protocol

A protocol has been developed to provide synchronization between the robots. The length of this protocol is 6 bytes. The described protocol contains sender and receiver information and coordinate points. This protocol is given in Figure 10.

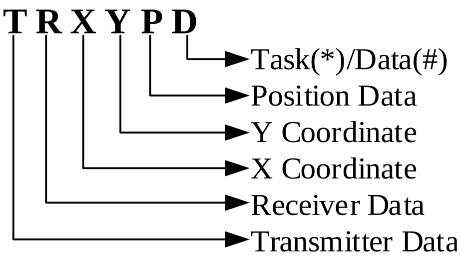

Figure 10. Communication Protocol

The first byte of the protocol carries the sender data. In this study, the letters "M" for the central computer, "O" for Leader and "D" for Assistant are used. The second byte carries the receiver data. As the first byte of the protocol, the letters M, O, and D are also valid for this byte. The third and fourth bytes carry the X and Y axis information. Although the coordinate plane is between -4 and +4 in this study, the coordinate information is sent by summing with 5 for the simplicity of protocol. This prevents the negative numbers in the protocol. The receiver robot adapts the data in the protocol to the coordinate plane by subtracting 5 from the robot coordinates. The fifth byte contains the position information. Position information is organized as 0 for +Y, 1 for -Y, 2 for -X, 3 for + X. If the sixth byte is "*", the robot is asked to go to the coordinates and position values. If the sixth byte is "#", the coordinate information of the robot is requested regardless of the coordinate information sent. In Figure 11, the protocol data that Leader asked help from Assistant is shown.

Figure 11. Stand-alone work and an example of protocol

When Assistant received help message from Leader, moves to received coordinates. After that they carry the cargo cooperatively to target (Fig 12).

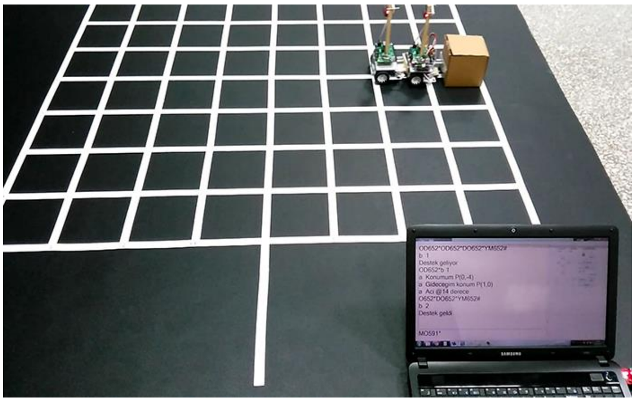

Figure 12. Cooperative work (https://youtu.be/sfQggAvIxNE)

6. Experimental Results

Experimental studies were carried out in two main categories. In the first step, the robots were tested on path planning algorithms. The robots performed path planning with line following and curve fitting algorithms. Figure 13 and Figure 14 show the paths that are followed according to line following method and curve fitting respectively. In both methods, firstly, the task of determining the coordinates of the cargo was applied (object detection task). When the cargo was on coordinates (-3,1), the travelled distance with line following method was 120 cm. For the same task, the covered distance was 99 cm with curve fitting method (cargo carrying task).

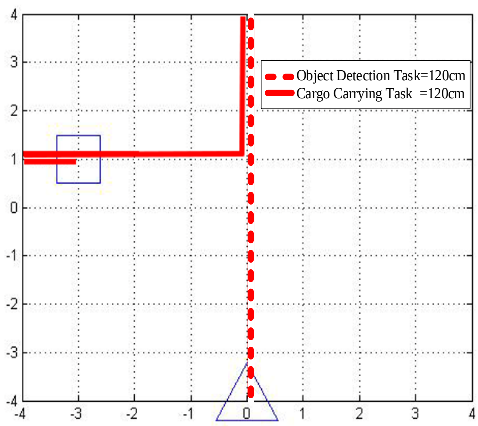

Figure 13. Path Planning with line following

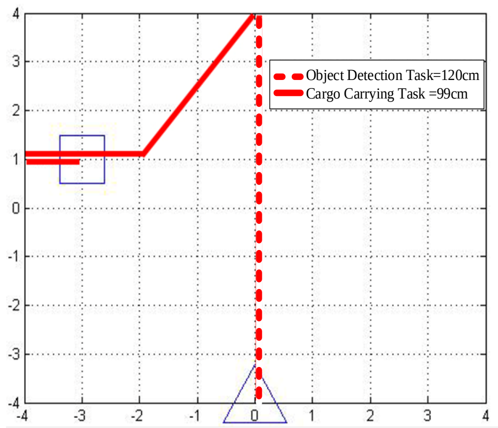

Figure 14. Path Planning with curve fitting

Cargo was placed at the coordinates (-3, Y) in the test setup and the travelled distance for each cargo carrying task was measured. The distances for each method are shown in Table 1. The travelled distance was reduced up to 17.5% with curve fitting method.

Table 1. Results on path planning algorithms

Coordinates of Cargo	Line Following Distance	Curve Fitting Distance	Saved Distance	%Saved Distance
-3,4	75	75	0	0.0%
-3,3	90	79	11	12.2%
-3,2	105	87	18	17.1%
-3,1	120	99	21	17.5%
-3,0	135	112	23	17.0%
-3,-1	150	126	24	16.0%
-3,-2	165	139	26	15.8%
-3,-3	180	154	26	14.4%
-3,-4	195	169	26	13.3%
% Average of saved distance				13.7%

In the second step, the robots were tested on completing object detection, cargo handling and updating coordinates tasks. The experiment was realized for four different scenarios and repeated ten times. These scenarios and experiment results are given in Table 2. Consequently, Leader detected cargos with 93% average accuracy. The cause of 7% error was the propagation angle of the distance sensor and the line deflection angle during line following (Figure 15(a)). For the same reasons, the updating coordinate's task performance was determined as 80%. The average cargo carrying task performance rate was reduced to 78%. The cargo approach deflection angle (Figure 15(b)) caused falling average rate of success. The tests showed that cooperative work is more successful for further cargos.

Table 2. Success rates of the tasks

Working principle	Task\Cargo position	Object detection	Updating coordinates	Carrying cargo
Stand-alone	Nearest Cargo	100%	100%	90%
	Further Cargo	90%	60%	60%
Cooperative	Nearest Cargo	90%	90%	90%
	Further Cargo	90%	70%	70%
% Average performance rate		93%	78%	78%

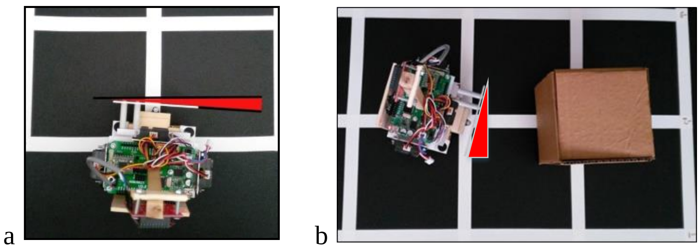

Figure 15. Main causes of errors. **(a)** Line deflection angle **(b)** Approach deflection angle

7. Conclusion

In this study, collaborative multi-robot system of consisting two robots for carrying cargos was designed and implemented via curve fitting and line following algorithm. Line following and curve fitting methods were compared on path planning. The travelled distance was measured for each algorithm. Curve fitting method for path planning reduced the average distance 13.7% according to line following method.

Tests on accomplishment of task completion for near and further carrying cargo were done. The tests were implemented as stand-alone and cooperatively. Cooperative work was more effective than stand-alone for especially further cargos.
For further work, the robots can be extended to be used to carry cargos under the industrial 4.0 in factories. Robot orientation and object detection errors can be reduced by increasing sensors.

References

Abdi H., Black T. & Nahavandi S. (2011). An adjustable force field for multiple robot mission and path planning. In Proceedings of the 2011 IEEE International Conference on Systems, Man, and Cybernetics, Anchorage, USA, 1950–1955. doi:10.1109/ICSMC.2011.6083957

Adrian L. R. & Ribickis L. (2013). Fuzzy logic analysis of photovoltaic data for obstacle avoidance or mapping robot. Elektronika Ir Elektrotechnika, 19(1), 1392-1215. doi:10.5755/j01.eee.19.1.3243

Alotaibi E. T. S. & Al-Rawi H. (2016). Push and spin: a complete multi-robot path planning algorithm. In Proceedings of the 14th International Conference on Control, Automation, Robotics and Vision, Phukhet, Thailand, 13-15. doi:10.1109/ICARCV.2016.7838836

Berglund T., Brodnik A., Jonsson H., Staffanson M. Soderkvist I. (2010). Planning smooth and obstacle-avoiding B-spline paths for autonomous mining vehicles. IEEE Transactions on Automation Science and Engineering,7(1), 167-172. doi:10.1109/TASE.2009.2015886

Ganeshmurthy M. S. &vSuresh.G. R. (2015). Path planning algorithm for autonomous mobile robot in dynamic environment, In Proceedings of the 3rd International Conference on Signal Processing, Communication and Networking, Chennai, India. doi:10.1109/ICSCN.2015.7219901

Gong H., Yin C., Zhang, F., Hou Z., Zhang R. (2017). Path planning algorithm for unmanned vehicles based on target-oriented rapidly-exploring random tree. In Proceedings of the 11th Asian Control Conference (ASCC), Gold Coast, Australia, 760-765. doi:10.1109/ASCC.2017.8287266

Graetz G. & Michaels G. (2018). Robots at work. Review of Economics and Statistics, 100(5), 753-768. doi:10.1162/rest_a_00754

Howard A. (2006). Multi-robot simultaneous localization and mapping using particle filters. International Journal of Robotics Research, 25(12), 1243–1256. doi:10.1177/0278364906072250

Hussein A., Adel M., Bakr M., Shehata O. M. & Khamis A. (2014). Multi-robot task allocation for search and rescue missions, Journal of Physics: Conference Series, 570: 1–10. doi:10.1088/1742-6596/570/5/052006

Kermorgant O. (2018). A magnetic climbing robot to perform autonomous welding in the shipbuilding industry. Robotics and Computer-Integrated Manufacturing, 53, 178-186. doi:10.1016/j.rcim.2018.04.008

Kim H., Kim D., Kim H., Shin J.U. & Myung H. (2016). An extended any-angle path planning algorithm for maintaining formation of multi-agent jellyfish elimination robot system. International Journal of Control, Automation, and Systems, 14(2), 598–607. doi:10.1007/s12555-014-0349-0.

Koch P., May S., Schmidpeter M., Kuhn M., Pfitzner C., Merkl C., Koch R., Fees M., Martin J., Ammon D. & Nuchter A. (2016). Multi-robot localization and mapping based on signed distance functions. Journal of Intelligent & Robotic Systems, 83, 409–428. doi:10.1007/s10846-016-0375-7

Macwan A., Vilela J., Nejat G.& Benhabib B. (2016). Multi-robot deployment for wilderness search and rescue. International Journal of Robotics and Automation, 31(1), 45-51. doi:10.2316/Journal.206.2016.1.206-4366

Madhu B., Sakkaravarthi R., Singh G. M.; Diya R. &. Jha D. K. (2017). Modeling, simulation and mechatronics design of a wireless automatic fire fighting surveillance robot. Defence Science Journal, 67(5), 572–580. doi:10.14429/dsj.67.10237

Marjovi A. & Marques L. (2011). Multi-robot olfactory search in structured environments. Robotics and Autonomous Systems, 59,11867–11881. doi:10.1016/j.robot.2011.07.010

Moradi B. (2019). Multi-objective mobile robot path planning problem through learnable evolution model. Journal of Experimental & Theoretical Artificial Intelligence, 31(2), 325-348. doi: 10.1080/0952813X.2018.1549107

Nagy I. (2014). From exploring to optimal path planning: considering error of navigation in multi-agent mobile robot domain. Acta Polytechnica Hungarica, 11(6), 39-55. doi:10.12700/aph.11.06.2014.06.3

Nazarahari M., Khanmirza E., Doostie S. (2019). Multi-objective multi-robot path planning in continuous environment using an enhanced genetic algorithm. Expert Systems with Applications, 115, 106-120. doi:10.1016/j.eswa.2018.08.008

Özarslan Yatak M., Göktaş B. & Duran F. (2018). Design and implementation of Android-based autonomous human tracking vehicle. International Journal of Informatics Technologies. 11(2), 157-162. doi:10.17671/gazibtd.340566

Prabuwono A. S., Burhanuddin M. A. & Said S. M. (2008). Autonomous contour tracking using staircase method for industrial robot. In Proceedings of the 10th International Conference on Control Automation Robotics and Vision, Hanoi, Vietnam, 2272–2276. doi:10.1109/ICARCV.2008.4795886

Saeedi S., Paull L., Trentini M & Li H. (2011). Neural network-based multiple robot simultaneous localization and mapping. IEEE Transactions on Neural Networks, 22(12), 2376–2387. doi:10.1109/TNN.2011.2176541

Saeedi S., Paull L., Trentini M, & Li H. (2015). Occupancy grid map merging for multiple robot simultaneous localization and mapping. International Journal of Robotics and Automation, 149-157. doi:10.2316/Journal.206.2015.2.206-4028

Seçkin A.Ç., Özek A. & Karpuz C. Çoklu robotlarda işbirlikli davranışların karşılaştırılması ve bulanık mantık yaklaşımı. Politeknik, Erken Görünüm. doi:10.2339/politeknik.481177

Sun D., Kleiner A. & Wendt T.M. (2008). Multi-robot range-only SLAM by active sensor nodes for urban search and rescue. In Proceedings of the 12th annual RoboCup International Symposium, Suzhou, China 318–330. doi:10.1007/978-3-642-02921-9_28

Thabit S., Mohades A. (2019). Multi-robot path planning based on multi-objective particle swarm optimization. IEEE Access,7, 2138-2147. doi:10.1109/ACCESS.2018.2886245

Thor J., Schultz U. P. & Kuhrmann M. (2015). On the use of safety certification practices in autonomous field robot software development: A systematic mapping study. In Proceedings of the 16th International Conference on Product-Focused Software Process Improvement, Bolzano, Italy, 335–352. doi:10.1007/978-3-319-26844-6_25

Tominaga A.; Hayashi E. & Sasao T. (2017). Localization method of autonomous moving robot for forest industry. In Proceedings of the International Conference on Artificial Life and Robotics, Miyazaki, 376-379.