Remote Wi-Fi Control of a Holonomic Mobile Robot

Danail V. Slavov and Vladimir D. Hristov

***Abstract* — This paper describes a mobile robotic system remotely controlled by a smartphone using Android app. The mobile robot platform has omnidirectional Mecanum wheels which enables movement in any direction and increases its agility. Data communication is accomplished using Wi-Fi connection between the smartphone and the robot. Amica NodeMCU integrating ESP8266 is used to set up the wireless server, and Arduino Mega 2560 to receive user's commands and apply motion control. Ten locomotion types are successfully performed, both rectilinear and rotational. Safety anti-collision function is implemented using ultrasonic distance sensor. It reliably stops the robot just before reaching an obstacle even at highest speed. Smoother motor rotation is achieved using different speed reduction coefficients, to overcome the unequal losses in the motors. The realized communication procedure allows fast transfer of larger data which can help improving the robot control and functionality. The mobile platform has potential for extending its sensory system and achieve a more intelligent behavior.**

***Index Terms*—Mecanum wheels, Arduino, omni-directional robot, ESP8266**

I. INTRODUCTION

Mobile robots, vehicles, and robot arms are now widely used in various fields, such as autonomous transportation in factories and warehouses. Furthermore, mobile robots that can carry people have also been attracting attention. To increase efficiency for transporting products, vehicles, such as conveyance carriers and mobile robots, are used in factories and warehouses. The conventional vehicles combine going forward/backward and steering to realize various motions. However, the vehicles need switchbacks and turnabouts to move in different directions (right, left, or diagonally) and achieve holonomic mobility. This in turn increases their efficient motion in environments with narrow passages or many obstacles. Thus, omnidirectional vehicles that can easily move in the crosswise and diagonal directions, in addition to front-back direction and steering, are required. To meet this requirement, various mechanisms for omnidirectional movement have been proposed and studied [1]

Holonomic mobile robots can also be equipped with various effectors like for example the Mobile Collaborative robotic Assistant (MOCA), described in [2]. It is composed of a manipulator arm, an underactuated hand, and a mobile platform driven by four omni-directional wheels enabling mobility in the workspace. These robotic systems can accomplish different tasks in both industrial and domestic settings and also are highly adaptive. They can assist human operators, reducing their physical risks, with agile mobility and advanced interaction and manipulation. Another example of omni-wheeled mobile robot for local and indoor navigation purposes, including automatic warehousing in industry or healthcare environment, is described in [3]. The angular speed and position are read by internal rotary encoders mounted in all DC motors which can also be combined with inertial measurement unit (IMU).

With the fast growth of mobile internet environment, robot applications utilizing smartphones have become interesting and feasible research topic. Remote control methods of mobile robots have been proposed in many studies, such as [4], which uses Windows Mobile OS, Microsoft's .Net Compact Framework and Bluetooth data communication. In [5] a Wi-Fi based remote control system is presented, which performs complex operations in dangerous environment and improves efficiency and accuracy of industrial production. The controlled four-wheel-drive mobile robot communicates with a PC terminal through wireless network.

This paper presents a Mecanum-wheeled mobile robotic platform which can perform omni-directional locomotion and is controlled remotely by a smartphone through Wi-Fi communication. It also has a safety anti-collision function and improved motion of its four independently driven motors.

II. MOBILE PLATFORM CONFIGURATION

Described robotic platform consists of solid metal chassis with four wheels which are independently driven by four DC motors. It also has a front-facing ultrasound (US) sensor for anti-collision safety function (Fig. 1).

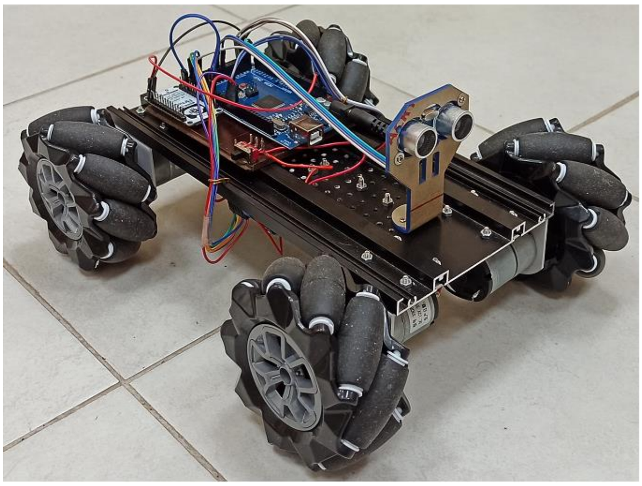

Fig. 1. General view of the mobile robot

D. V. Slavov is with the Department of Electrical Motion Automation Systems, FA, Technical University of Sofia, 1000 Sofia, Bulgaria (e-mail: d.slavov@tu-sofia.bg).

V. D. Hristov is with the Department of Electrical Motion Automation Systems, FA, Technical University of Sofia, 1000 Sofia, Bulgaria (e-mail: vdhristov@tu-sofia.bg).

https://doi.org/10.47978/TUS.2021.71.03.005

To achieve omni-directional movement, three types of wheels can be used: universal wheels, omni wheels and Mecanum wheels. The universal wheel can rotate in any direction but its weighing ability is poor, and the probability of sliding and accidents is high. The omni wheel can move flexibly in all directions but the structure is more complicated, the lateral direction cannot be fixed, and the load-bearing capacity is relatively poor. It requires precise angles of the rollers (90° for holonomic drive) and if these are not perfectly achieved in the manufacturing, the movement can have adverse driving characteristics. The Mecanum wheel (also called Swedish wheel or Ilon wheel) moves flexibly in all directions, and the load-bearing capacity is better. Compared to omni wheels, the Mecanum wheels are not easy to slip and their installation method can generate force in any direction. [1]

For the present implementation, Mecanum wheels are chosen which are tireless and have rubberized external rollers obliquely attached to the whole circumference of the rim. Each roller has an axis of rotation at 45° to the wheel plane and at 45° to the axle line. [6] The Mecanum wheel is an independent non-steering drive wheel with its own powertrain, and when spinning, it generates a propelling force perpendicular to the roller axle, which can be vectored into a longitudinal and a transverse component in relation to the vehicle. [7] Fig. 2 shows the platform movement depending on the corresponding wheels spinning direction (at constant speeds).

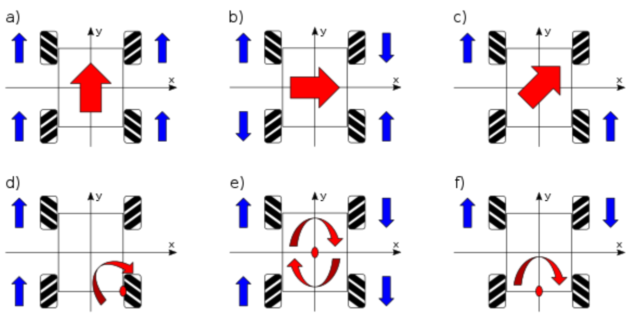

Fig. 2. Wheel rotation requirements for 2D holonomic locomotion of the mobile platform [8]. Blue: wheel drive direction; red: vehicle moving direction. a) Straight ahead, b) sideways, c) diagonally, d) around a bend, e) rotation, f) around the axle center

III. Communication

For data communication with the robot, Wi-Fi connection is used. The following devices partake in the communication process: Android smartphone, Amica NodeMCU, Arduino Mega 2560 (Fig. 3). Both Arduino Mega and NodeMCU development boards are mounted on the robot chasis. The NodeMCU has an integrated ESP8266 microchip with Wi-Fi functionality supporting full TCP/IP protocol stack. It acts as a wireless access point (AP) creating a network with specific SSID and password. The smartphone connects to the Wi-Fi network through its standard Android functionality. Then, using an Android app (named ESP8266 Controller plus ULTRA), the user can send commands to a specific IP address. By default, it is 192.168.4.1 but this can be changed. The commands are read by the NodeMCU and retransmitted to the Arduino via a wired serial communication.

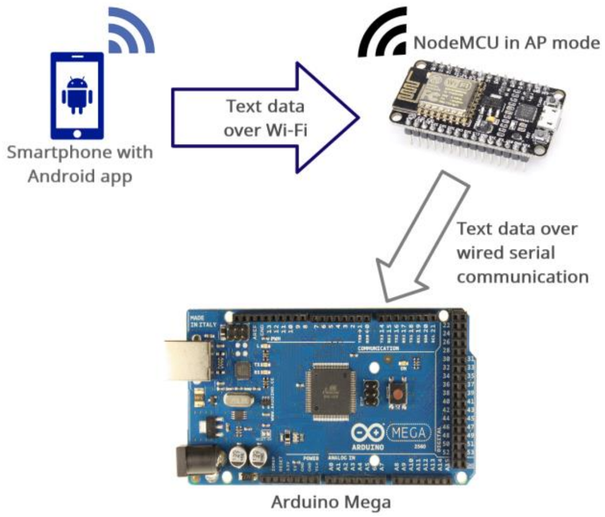

Fig. 3. Communication scheme

This way, the NodeMCU acts as a server and the phone as its client. ESP8266 can also operate as Wi-Fi station. In such mode, it can connect to a wireless LAN (e.g. emitted by a router), communicate with other devices on the network, and if the local network is connected to the Internet, the ESP8266 can also communicate with any device on the web. [9]

Several UI items of the Andoid app (Fig. 4) are used to generate signals for wheel speed, sensor operation, and platform locomotion. Each interaction with the UI sends a single character to the server. This character is in turn read by the Arduino, which maps it to a required command and applies control upon the motors. Thus, the corresponding robot behaviour is achieved depending on the command.

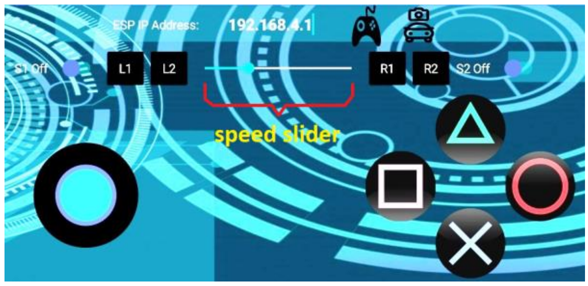

Fig. 4. User interface of the ESP8266 Controller plus ULTRA app

IV. Remote Control

When the user touches a soft button or a slider in the smartphone app, a character value is sent to the Wi-Fi server. The Arduino controller receives and maps the symbol to a corresponding command. Table I presents the character mappings to specific control functions.

The speed slider has nine levels, i.e. it sends an integer number from 1 to 9 (as character data type) depending on the user's choice. After releasing one of the direction buttons (☐ , △ , ○ or ✕), the app sends the character “s”. It is mapped to a halt command, so the mobile platform moves only while a direction button is been touched and hold, otherwise the platform stops. In horizontal-vertical mode, direction buttons indicate forward, backward, left or right movement, while in diagonal mode, corresponding diagonal movements.

TABLE I
CHARACTER-TO-COMMAND MAPPING

UI item	Character	Command
S1 On	A	Distance sensors activation
S1 Off	B	Distance sensors deactivation
L1	C	Currently not used
L2	D	Currently not used
R1	E	Clockwise rotation
R2	F	Counterclockwise rotation
S2 On	G	Diagonal locomotion mode, D
S2 Off	H	Horizontal-vertical mode, H-V
Speed slider	1 – 9	Set motor speed level
☐ (square)	M	Left (H-V) or backward-left (D)
△ (triangle)	N	Forward (H-V) or forward-left (D)
◯ (circle)	O	Right (H-V) or forward-right (D)
╳ (cross mark)	P	Backward (H-V) or backward-right (D)
Direction button release	s	Stop

The mobile robot uses an US sensor to avoid collisions with front-located obstacles. When the distance becomes smaller than a certain value, the robot stops and the front locomotion command is forbidden until the distance to obstacle becomes greater than the limit. This safety function can be deactivated, e.g. for precise navigation at slow speed around closely spaced objects. The control algorithm is presented on Fig. 5.

V. IMPLEMENTATION AND TESTING

The software program on the NodeMCU side includes Wi-Fi server initialization, access point set-up, remote client administration, as well as data obtainment and transmission to the Arduino Mega controller. Connection management between the Wi-Fi server and its clients is organized using the ESP8266WiFi Arduino library. The access point is configured with the help of the softAP() function.

The program on the Arduino side defines all necessary pin configurations, speed levels and reduction coefficients for the four motors, as well as the allowable obstacle distances. It implements the mobile robot control algorithm taking into account the user input and sensor readings. The program includes software functions for ten locomotion types: four horizontal-vertical, four diagonal and two rotational movements (see Table I). They correspond to certain combinations of motor rotation directions and speeds – as described in Section II. The speed control uses pulse-width modulation (PWM) realized by two L298N dual H-Bridge motor drivers (Fig. 6). Each of them can control two DC motors, so a four-wheel independent drive is achieved for the robot.

The pulse widths are programmatically set as byte numbers in the range of 0 to 255. They correspond to certain voltage levels (0 through 5 V) applied to the driver "Enable" pin. Experiments showed that, in order to overcome losses in the motors used, PWM values should be above 80, otherwise the motors cannot rotate and carry the platform. For this reason, all PWM wave values are programmatically bounded in the range

$$PWM\ wave \in [-MAX, -MIN] \cup [MIN, MAX],$$

$MIN = 80$, $MAX = 255$. The absolute value determines the speed and the sign determines the direction of rotation.

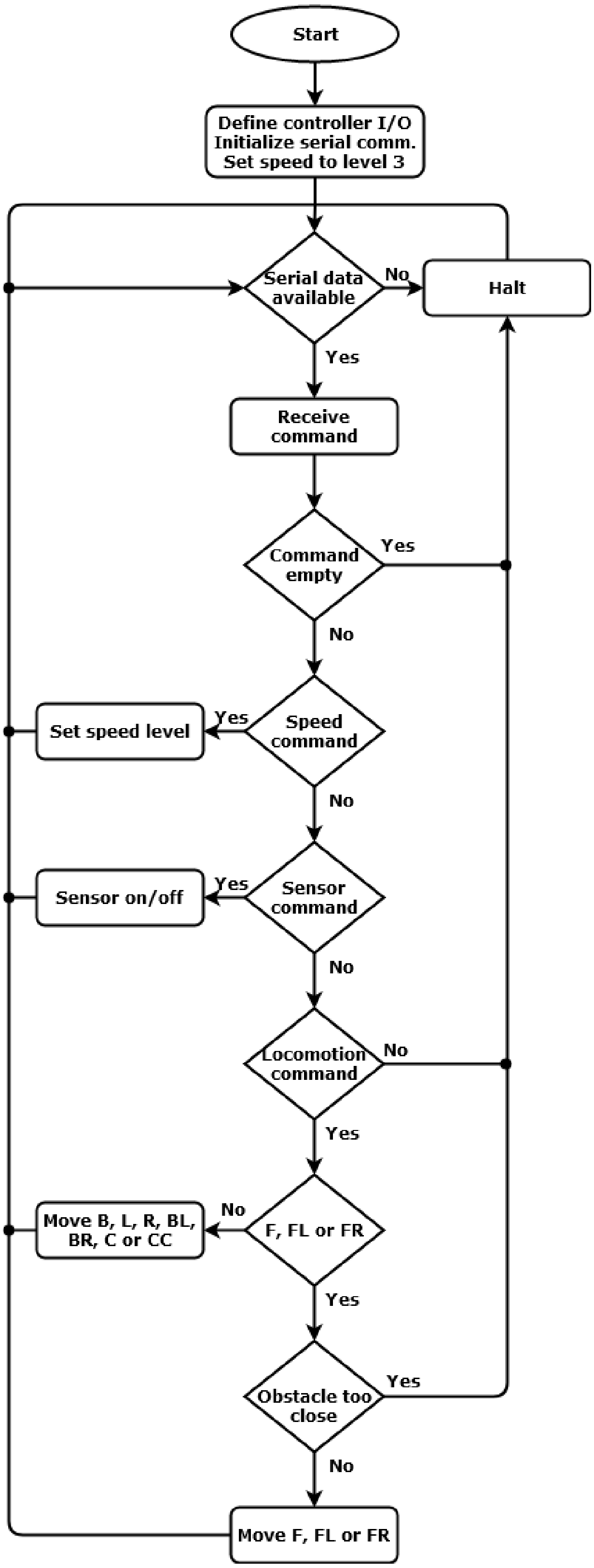

Fig. 5. Block diagram of the control algorithm

Another problem which appeared during the test phase was the unequal speeds of the four motors, even when the PWM target applied to them was the same. As a result, the mobile platform could not maintain straight movement in a certain direction. This imposed the use of reduction coefficients for each motor whose values were additionally determined through experiments.

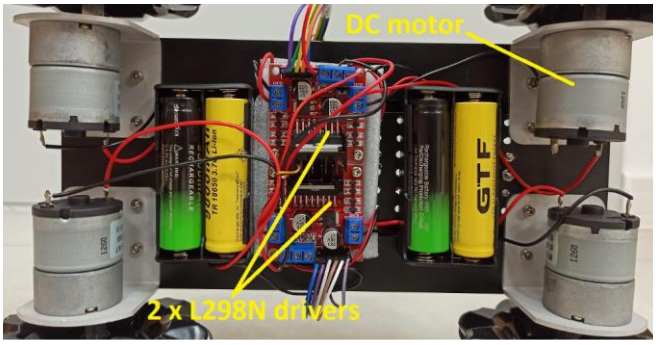

Fig. 6. Bottom view of the mobile platform

The obstacle distance at which the robot should stop depends on the speed – when the movement is fast, the distance is larger so that the robot can have enough time to decelerate. Thus, a reliable braking at several centimeters from the object is achieved regardless of speed.

Another aid implemented for a more rapid deceleration is movement in the opposite direction for a short duration (around 50 – 200 ms), which is initiated during the “halt” command issue. This duration proportionally depends on the speed. The current locomotion direction (forward, backward, etc.) is kept into the memory in order for the system to know which opposite movement should apply if such abrupt stop is needed.

VI. Conclusion

During this research work, a mobile robotic system was assembled with a structure capable of omnidirectional movement. The necessary software functions have been developed and tested for several locomotion types, speed setting and collision prevention. Finally, they all performed as planned, although after completing the necessary corrections, adjustments and improvements. The data communication proved fast and reliable enough for the desired remote control approach. The following remarks and favorable opportunities for future research and development are worth noting:

Described implementation uses communication where each command is represented by a single character. However, the system is capable of transferring larger portions of data, e.g. strings transformed to character arrays and submitted by the NodeMCU serially to the Arduino microcontroller. This allows wider set of instructions towards the robot and more variegated remote control. The system could also be further developed for two-way communication, in which the mobile robot detects and measures environment parameters and sends them to the user or even to an automatic decision making subsystem.

Presented function against collision is currently implemented only for the front part of the mobile platform but in a similar way can be easily extended to cover the side and back parts, thus ensuring a more complete and precise safety capabilities.

In order to achieve even smoother locomotion, the motor reduction coefficients can be changed from fixed values to variables depending on the target speed. As this dependency would probably appear non-linear, they can be trained by machine learning techniques, for which appropriate measurement and registration equipment needs to be ensured.

References

[1] W. Binghui, Q. Dezhao, C. Yuhan, C. TianQi and W. Maoliang, "Structure design of an omni-directional wheeled handling robot," Journal of Physics, 2021.

[2] W. Kim, M. Lorenzini, P. Balatti, Y. Wu and A. Ajoudani, "Towards Ergonomic Control of Collaborative Effort in Multi-human Mobile-robot Teams," in IEEE International Conference on Intelligent Robots and Systems, 2019, https://doi.org/10.1109/IROS40897.2019.8967628

[3] A. Sofwan, H. Mulyana, H. Afrisal and A. Goni, "Development of omni-wheeled mobile robot based-on inverse kinematics and odometry," in 6th International Conference on Information Technology, Computer and Electrical Engineering, ICITACEE 2019, 2019, https://doi.org/10.1109/ICITACEE.2019.8904418

[4] Y. Seo, S. Kwak and T. Yang, "Mobile Robot Control Using Smart Phone and Its Performance Evaluation," Advanced Communication and Networking. Communications in Computer and Information Science, vol. 199, 2011, https://doi.org/10.1007/978-3-642-23312-8_47.

[5] Y. Z. Yang and e. al., "A Study of Robot Platform Based on WiFi Remote Control," Applied Mechanics and Materials, vol. 418, p. 20–24, 2013, https://doi.org/10.4028/www.scientific.net/AMM.418.20

[6] M. Blackwell, "The URANUS Mobile Robot," Carnegie Mellon University, Pittsburgh, 1990, https://doi.org/10.21236/ADA236593.

[7] H. Leon-Rodriguez, "Mobile mecanum robot," Modulable Robotics & NDT Automation, 2020. [Online]. Available: http://robotics.umng.edu.co/mobile.mecanum.html. [Accessed 20.06.2021].

[8] "Mecanum wheel," 2021. [Online]. Available: https://en.wikipedia.org/wiki/Mecanum_wheel. [Accessed 20 06 2021]

[9] "Establishing a Wi-Fi connection," 2020. [Online]. Available: https://tttapa.github.io/ESP8266/Chap07%20-%20Wi-Fi%20Connections.html. [Accessed 18 06 2021].