*Vietnam Journal of Science and Technology* ***59*** *(1) (2021) 110-119*

doi:10.15625/2525-2518/59/1/15301

# IMPROVEMENT OF CONTROL ALGORITHM FOR MOBILE ROBOT USING MULTI-LAYER SENSOR FUSION

**Xuan-Ha Nguyen[1, *], Van-Huy Nguyen[2], Thanh-Tung Ngo[1], Duy-Anh Nguyen[1]**

[1]*Department of Machine Design and Robotics, Hanoi University of Science and Technology, No. 1, Dai Co Viet, Ha Noi, Viet Nam*

[2]*CMC Institute of Science and Technology, No. 11, Duy Tan, Ha Noi, Viet Nam*

[*]Email: *ha.nguyenxuan@hust.edu.vn*

Received: 21 July 2020; Accepted for publication: 27 October 2020

**Abstract.** Mobile robots have received much of attention in the last three decades due to their very high potential of applications in smart logistics, exploration, and autonomous-intelligent services. One of the important functions of mobile robots is the navigation in which robot must know their location, the map of the environment and perform path planning with obstacle avoidance. In this paper, an improvement of a control algorithm for a mobile robot using multi-layer sensor fusion toward the target of an efficient obstacle avoidance is introduced. Based on our method, three layers of sensors arranging in three height-different planes of robot's housing for sensor fusion were used. A control algorithm, which is extended from the so-called bubble rebound algorithm and combined with a SLAM algorithm, was proposed. Experimental implementations on a mobile robot, named EAI, show that our algorithm can control the robot to navigate and avoid obstacles much efficiently, in which obstacles in forms of different shapes and heights can also be avoided. High repeatability and stability of the algorithm are obtained.

*Keywords:* mobile robot, sensor fusion, obstacle avoidance, navigation, robot control.

*Classification numbers:* 5.3.3, 5.3.6, 5.3.7.

## 1. INTRODUCTION

Mobile robots offer many potential applications both in daily-life services and industry. For all applications, robots must be able to move in a working environment with an optimized path, accompanying with an obstacle avoidance ability. For this task, robots must know the map of environments and their positions in the map simultaneously, which are obtained through a so-called SLAM algorithm (Simultaneous Localization And Mapping). In a regular operation of robots, any changes of environments must be updated to the map, including both static and moving objects. Therefore, robots should be equipped with a reliable sensor system to fuse and detect any changes of environments. The use of multi-type sensors arranging in height-different planes of robots housing would be a good solution for any obstacle detection.

Obstacle avoidance is a key problem for navigating mobile robots in dynamic changing environments. While moving from the current position to a goal, robots must detect and avoid obstacles without any external influences [1]. There is a variety of solutions to solve this issue including both theoretical and practical ones. The so-called Virtual Force Field (VFF) method [2, 3] and the Vector Field Histogram (VFH) one [4, 5] are two of the earliest approaches.

Several approaches exploiting the LIDAR sensor to detect obstacles were also introduced [1, 6, 7]. With these approaches, the contour of obstacles can be approximately estimated. The obtained information is then used in obstacle avoidance algorithms. Infrared sensors or ultrasonic sensors or a combination of them are also used. For instance, the so-called "bubble rebound algorithm" [8], a simple infrared-based obstacle avoidance algorithm, was suggested and used in [9, 10]. A combined system of 6 ultrasonic sensors and an infrared sensor, establishing a 180-degree scrolling window in front of a robot, was introduced in [11] to overcome disadvantages of each type of sensors. In this method, the authors suggested 3 response layers. The limitation of both ultrasonic sensors and infrared sensors is that they cannot be able to estimate the form of obstacles effectively. However, the amount of data to be processed is quite small, which allows for real-time algorithms with a low cost of computation.

In another approach, a remarkable method, called model-based Dynamic Window Approach (μDWA), was introduced [12]. The method combines the data from multiple types of sensor (infrared, sonar and LIDAR) and finds a suitable moving plan to avoid obstacles depending on the map's information. However, this method requires a complex algorithm with a very high computational requirement.

It is known from the above-mentioned works that these approaches just tried to use data from multi-sensor fusion to develop algorithms for obstacle detection and avoidance. As a result, control algorithms to navigate robots to a goal were developed. However, the map of environments was not established. That means the SLAM algorithm was not performed. This would lead to failure when robots operate in an unknown environment.

The contribution of this work is the development of a more general control algorithm for mobile robots, which combines the SLAM algorithm and an obstacle detection/avoidance algorithm. A robot, named EAI Dashgo D1 [13], is chosen for the experimental platform. An enhancement for the ability of EAI Dashgo D1 robot to reliably avoid obstacles is targeted. An additional infrared sensor layer is installed on the highest plane of the robot's housing collaborating with the readily pre-installed ultrasonic sensor layer and the LIDAR sensor. The information from LIDAR sensor and odometry system was used for the SLAM algorithm. A new algorithm based on the bubble rebound algorithm and information of ultrasonic and infrared sensors to detect obstacles was developed and integrated with the SLAM into a control algorithm. As a result, the robot can localize its positions, estimate the map, detect the height-different type of obstacles, and thus navigate in working environments more efficiently. The efficiency of the method is experimentally evaluated. Furthermore, the introduced algorithm needs a very low computational cost. Thus, no need for a high-performance computer is required.

## 2. SYSTEM DESCRIPTION

### 2.1. Robot platform

The robot platform consists of 3 main parts: i) the robot base for moving, ii) the sensor system and ii) control system. The EAI Dashgo D1 platform is exploited to serve as a mobile platform. As shown in Figure 1a, the platform is equipped with 4 ultrasonic sensors arranging around the front side at the lowest layer of the robot's housing. A LIDAR sensor is setup at the center of the middle layer.  At the highest layer, a system of 10 infrared sensors was additionally installed. Two encoders are attached to the two shafts of driving wheels to transmit positions of the mobile platform through an odometry. Figure 1b shows the architecture of the control system. An embedded computer, named Raspberry Pi, is used as a core controller, which is installed with a Linux OS and a Robot Operating System (ROS). The LIDAR is directly connected to Raspberry Pi. Data from the infrared sensor system is processed and sent to the Rasberry Pi via a microcontroller, named Arduino. The Raspberry Pi is also communicated with the base controller. The task of the base controller is to read data from ultrasonic sensors and encoders/odometry then send them to the embedded computer. Additionally, the base controller simultaneously controls the two motors through a motor-driven circuit.

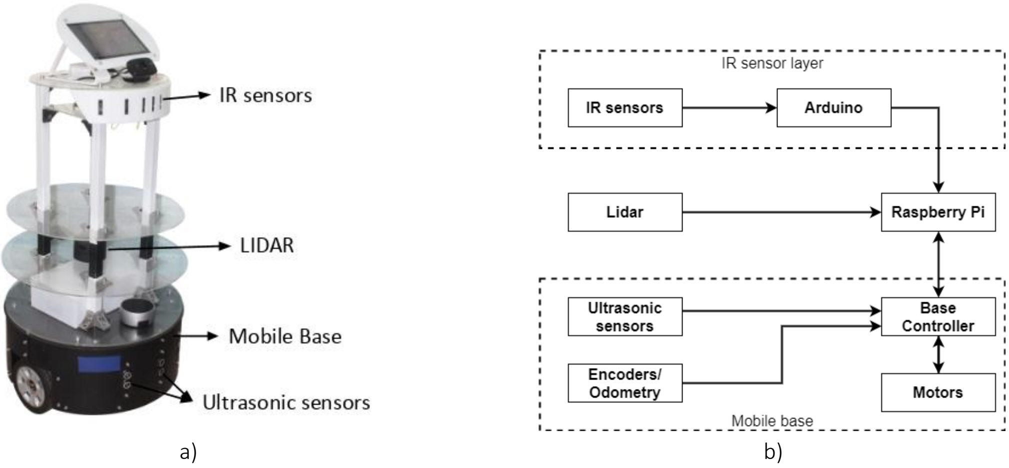


*Figure* 1. System description of the investigated robot,
a: Robot platform, b: Main connected modules.

## 2.2. Localization, mapping and navigation

Figure 2 illustrates a relationship between components for navigation tasks. There are three main groups: the localization, the mapping and the navigation. From the information of the odometry and the LIDAR sensor, a SLAM algorithm was performed to create the map of working environments and localize positions of the robot. In this work, the so-called Adaptive Monte Carlo Localization (AMCL) and Gmapping algorithm were applied. Based on the information of the obtained map and positions of the robot, a local/global cost-map and a local/global planner are generated, which are used to navigate the mobile platform to a goal. This is called navigation or path planning task. When using only the LIDAR sensor, the robot can only detect obstacles located in a range of the scanning plane of the LIDAR. This leads to a reality that many obstacles which have complex forms or irregular shapes, would not be detected and thus collision can appear. To solve this issue, information from ultrasonic and infrared sensors was exploited. A new algorithm, an extension of the bubble rebound algorithm, was built in such a way that obstacles in the working space of the robot can be detected more efficiently. The information of detected obstacles was then updated to both local and global map for navigation task with an obstacle avoidance ability. Since three layers of sensors were arranged in height-different planes of the robot’s housing, a ‘2.5D SLAM’ is achieved.  The contribution of this work is thus two folds: i) a combined SLAM algorithm allowing to detect and avoid obstacles more efficiently in comparison to purely LIDAR-based systems; ii) An advantage of computational efficiency due to the fact that the processing cost for ultrasonic and infrared sensors is much smaller than 3D-camera-based solutions. Without any need of 3D-camera, a ‘virtual’ 3D SLAM is obtained.

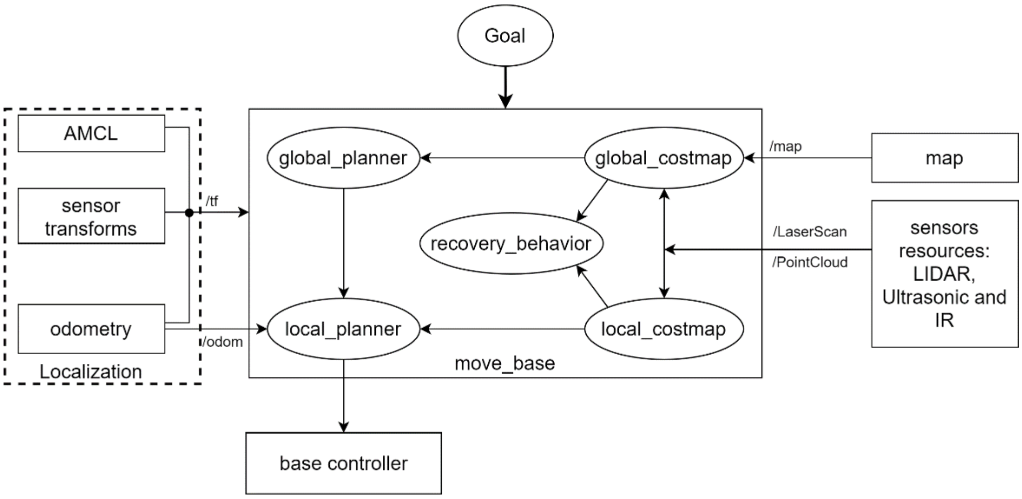


*Figure* 2. The relationship between navigation components.

### 2.3. Infrared sensor system

In this work, seven infrared sensors on the front side and three sensors at the back side of the robot’s housing were installed. The arrangement of these sensors is shown in Figure 3a. The Arduino circuit reads raw data from these sensors and sends them to the Raspberry Pi. The embedded computer has a ROS node which can convert the data to distance via formula (2.1) [14]. After that, an average filter was applied by (2.2).

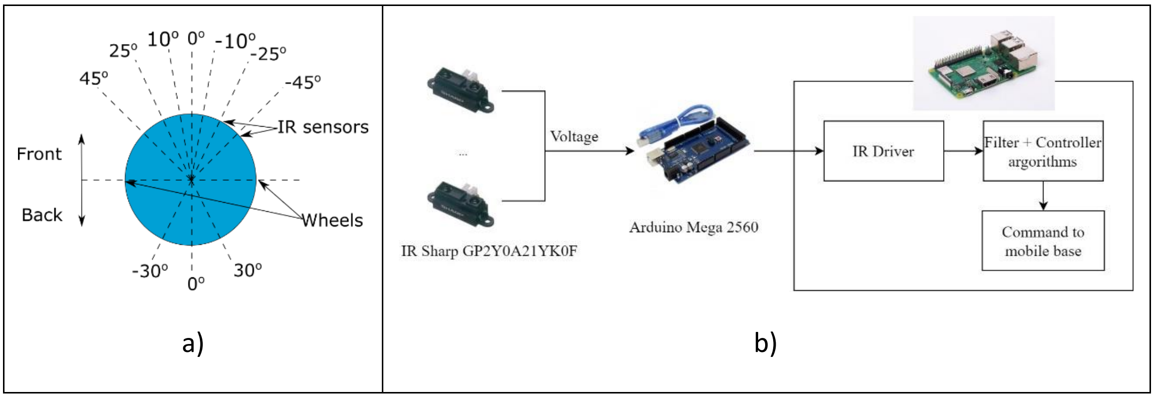


*Figure* 3. The IR sensor system, a: Arrangement of the system, b: Connection diagram.

$$d = 29.988 * \left(\frac{ADC * 5.0}{1023.0}\right)^{-1.173} \tag{2.1}$$

$$d_f = \frac{\sum_{i=1}^{N} d_i}{N} \tag{2.2}$$

where: $\boldsymbol{d}$ is the measured distance, which is converted from ADC voltage of the IR sensors; $\boldsymbol{d_f}$ is the average of 100 measurements ($\boldsymbol{N = 100}$).

### 2.4. Control algorithm

As illustrated in Fig. 4, two obstacle detection ranges, corresponding to two types of the robot reaction including the emergency range U and the bubble rebound range B were set. These ranges are described as follows:

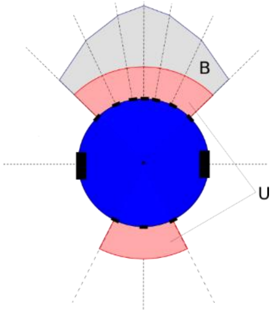


*Figure* 4. Two IR obstacle detection levels.

**The emergency range U**: This range is defined by a circle which center is the center of the robot's housing. The circle has a radius of $R_U$. When the robot detects obstacles in this range, it will move in the opposite direction far from obstacles and play a sound indicating an emergency until there are no obstacles in the range anymore. The detailed algorithm is shown in Fig. 5.

**The bubble rebound B:** A contour, whose shape and size are changed depending on the movement of the robot, was defined. The size of the bubble at each position of the infrared sensor is calculated by the formula: $bb[i] = K_i.V_t.\Delta_t$ [8]. $K_i$ is the factor corresponding on the $i^{th}$ sensor; $V_t$ is the velocity of the robot at an instance t and $\Delta_t$ is the time between two updates. This bubble ensures that the robot can move in the $\Delta_t$ period without collisions. Figure 6 illustrated the bubble rebound algorithm applied in the robot.

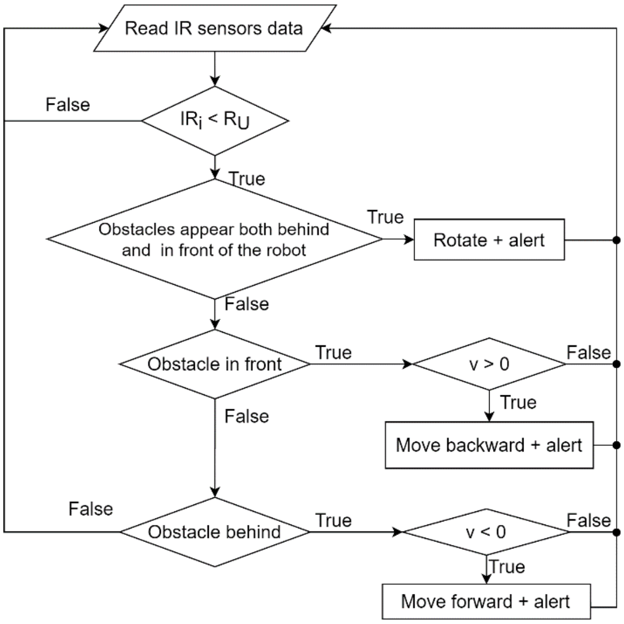


*Figure* 5. The algorithm of the emergency range U.

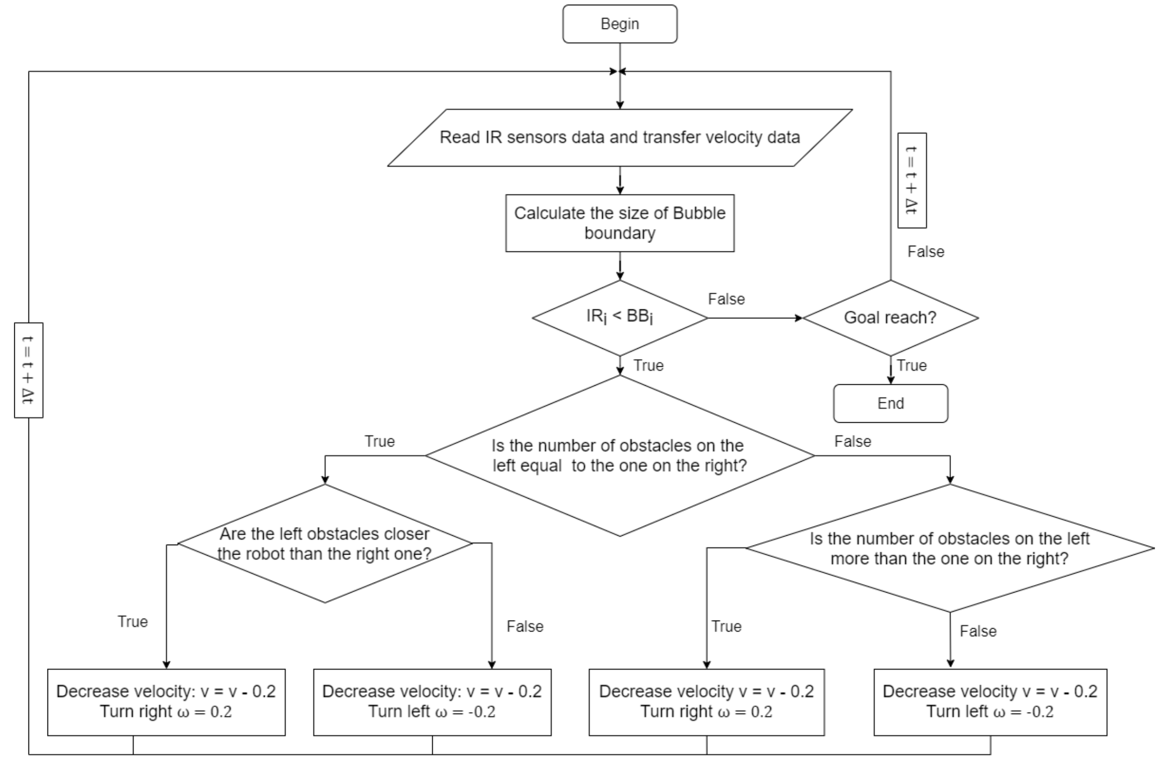


*Figure* 6. The extended Bubble Boundary algorithm.

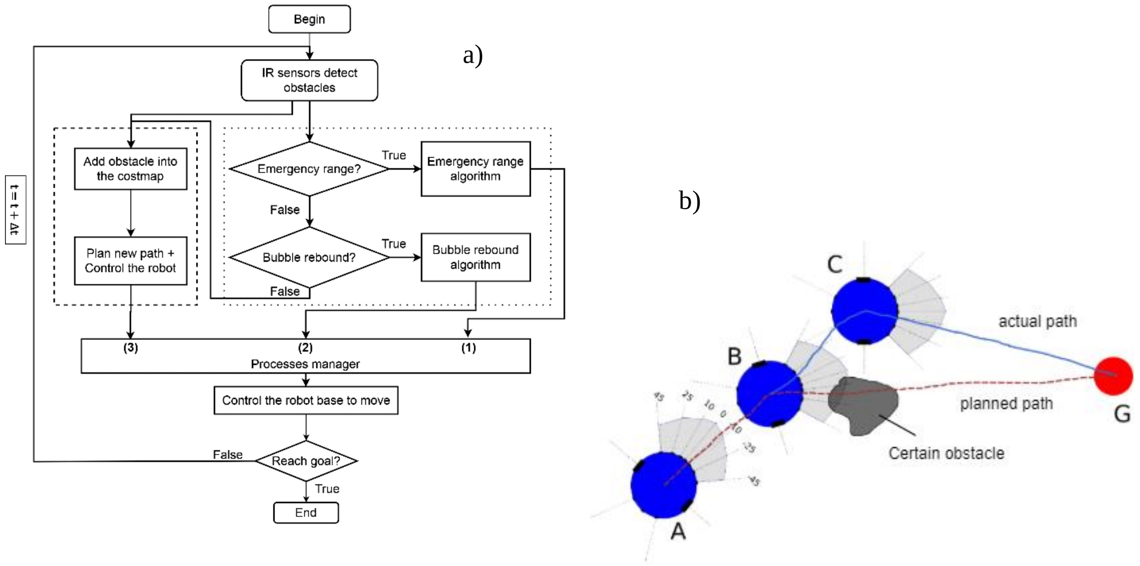


*Figure* 7. Hierarchical control system: a) General algorithm; b) Illustration of avoiding an obstacle.

**Hierarchical control system for the navigation and obstacles avoidance:** Generally, there are multiple processes that simultaneously control the robot at an instance. A priority for these processes must be determined according to a given rule. In our case, there are three processes corresponding to the three algorithms: i) the navigation; ii) the emergency range U; iii) the bubble rebound B, as shown in Fig. 7a. A priority for these three processes is implemented via a so-called “Processes manager”. It can be seen the number in the processes manager that the smaller the number is, the higher the priority is. That means, in our general algorithm, the process for the emergency range is set as the highest level of priority because this state has the highest probability of a collision. In normal operation of the robot without any obstacle, the process for the navigation is enabled. When an obstacle is detected, the navigation process is disabled and the two left processes have the right to control the robot according to the priority rule.

An example of the operation of the processes manager is shown in Fig. 7b. While the robot moves from the position A to the position B, the priority is set for the process of the navigation since no obstacle is detected. When arriving in the position B, an obstacle appears. The process for the emergency range and the bubble rebound range have the priority to control the robot to avoid the obstacle. Additionally, the information of the obstacle is updated to the map. At position C, the navigation is set active again because there is not any obstacle more. It is noticed that while the navigation process operates, the LIDAR sensor can also detect obstacles. However, as commented in the introduction section, many types of obstacles may not be detected without using ultrasonic and infrared sensors.

## 3. RESULTS AND DISCUSSION

We successfully integrated the infrared sensor system into a robot platform to improve the ability to avoid the obstacle of the robot. To evaluate the performance of our method, we set up a testbed with the corresponding results as follows:

### 3.1. The reliability of sensors

For evaluating the reliability of sensors, experiments on each sensor were implemented. Each measured data is an average of 100 measurements. The results of our experiments are showed in Tab. 1. The error rate of the sensors is below 5% with distance from 0.1 m to 0.7 m. It is an acceptable number for mobile robot applications.

*Table* 1. Evaluating the reliability of infrared sensors.

| Targeted (m) | 0.1 | 0.2 | 0.3 | 0.4 | 0.5 | 0.6 | 0.7 | 0.8 |
|---|---|---|---|---|---|---|---|---|
| Avg. Measured (m) | 0.1051 | 0.2009 | 0.31 | 0.4068 | 0.5157 | 0.6086 | 0.7146 | 0.7229 |
| Mean error (m) | 0.0051 | 0.0009 | 0.01 | 0.0068 | 0.0157 | 0.0086 | 0.0146 | -0.077 |
| Error in (%) | 5.14 % | 0.44 % | 3.34 % | 1.69 % | 3.14 % | 1.43 % | 2.08 % | 9.63 % |

### 3.2. Obstacle avoidance

In this experiment, we put an obstacle, which LIDAR sensor cannot detect, in front of the robot. A good example of this type of object is a spinning chair as shown in Fig. 8. The LIDAR can only detect a small circle at the center of the chair. Figure 9 shows the map of the environment and the path plan of the robot in two situations. The first situation is when the IR sensors are turned off (Fig. 9a), the second one is when the infrared sensors are turned on (Fig. 9b). When turning off the infrared sensors, the LIDAR sensor detects only a small circle in front of the robot. Meanwhile, the robot can detect the obstacle with two infrared sensors and change the local path to avoid the obstacle while turning on the infrared sensors.

Figure 10 shows the reaction of the robot when detecting obstacles by the infrared sensors system. As shown in Fig. 10a, the robot finds an obstacle detected by infrared sensors while moving. In this case, the obstacle is inside the bubble rebound range, so the robot decreases the velocity, turn right. At the same time, a grey region is marked on the local cost-map and the local path is changed as in Fig. 10b.

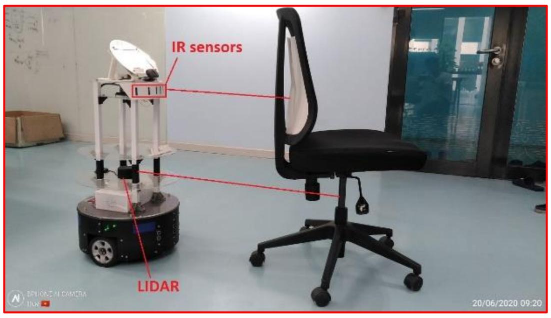
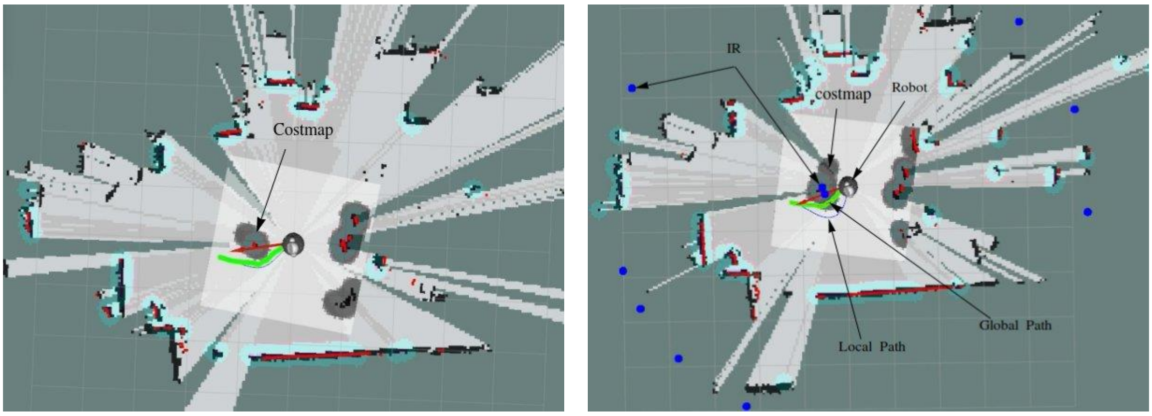
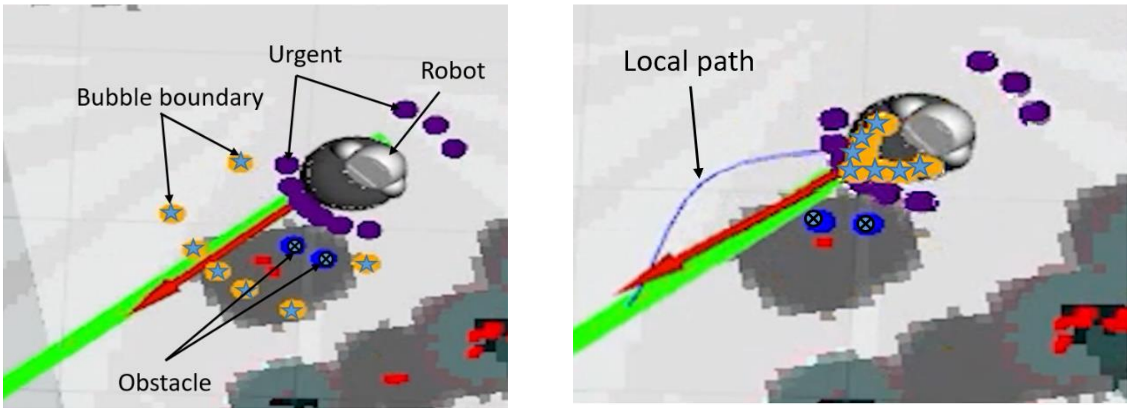


*Figure* 8. Experimental setup: the robot is opposite to a spinning chair.



a) b)

*Figure* 9. Comparison between turning on and turning off the IR sensors.



a) b)

*Figure* 10. The reaction of the robot when meeting an obstacle, a) Map when meeting an obstacle; b) the change of the robot's path to avoid the obstacle.

We perform two cases of testing: a chair in static status as a barrier on the moving path of the robot and a chair moving across the robot’s path. The result is that the robot can avoid the chair in both tests. However, we have not had a test about the limitation of the relative velocity between the robot and the obstacle for the robot to perform the avoidance. Generally, our method when activating the infrared sensors work more effectively than when turning them off. To validate the repeatability accuracy, we let the robot move back and forward 20 times through an obstacle at the infrared sensor layer and it can avoid the obstacle at all time successfully.

## 4. CONCLUSIONS

We integrated a system of sensors into a robot platform to collect data for the obstacle avoidance problem. Based on these data, we improved and applied our method to bypass the obstacle at the infrared sensor layer and cooperate with two other layers on controlling the robot. The experimental results have shown that our control algorithm works effectively. As a result, the robot can operate more reliable and avoid several kinds of obstacles different in form (shape and height). In comparison to previously-published works, our algorithm – a combination of multi-layer sensor fusion, the extension of bubble rebound algorithm and the SLAM, has improved significantly. The speed of the robot when reacting the obstacle is suitable for real-time operation. Furthermore, the computational requirement for the proposed algorithms is low, which allows for many applications with a cost-reasonable computer and a mean of energy savings.

Limitations have remained. The infrared sensor system cannot estimate the contour of obstacles, which causes confusion when meeting a big obstacle or a narrow slot between two obstacles. Although the reaction time of the robot to avoid obstacle is quite quick, we have not experimentally evaluated thoroughly. In the future we will develop an algorithm to estimate the contour of obstacles reliably. A combination of ultrasonic sensors and infrared sensors into a sensor layer is thinkable. Evaluation of reaction time will be performed.

***Acknowledgements.*** This research is funded by Hanoi University of Science and Technology (HUST) under project number TC2018-PC-022.

***CRediT authorship contribution statement.*** Author 1: Propose the idea and methodology, analyze the results, partly write the paper and funding acquisition. Author 2: Implement the ideas and perform experiments, partly write the paper. Author 3: Analyze the results and partly write and correct the final paper. Author 4: Partly perform the experiments and correct the writing.

***Declaration of competing interest.*** We confirm that we have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

## REFERENCES

1. Wu P., Xie S., Liu H., Luo J., Li Q. - A novel algorithm of autonomous obstacle-avoidance for mobile robot based on LIDAR data, Proceedings of IEEE Int. Conf. Robot. Biomimetics, IEEE-ROBIO 2015 **1** (2015) 2377-2382.
2. Borenstein J., Koren Y. - Real-time obstacle avoidance for fast mobile robots, IEEE Trans. Syst. Man, Cybern. **19** (5) (1989) 1179-1187.
3. Borenstein J., Koren Y. - High-speed obstacle avoidance for mobile robots, Proceedings IEEE Int. Sym. on Intel. Con., Arlington, VA, USA, 1988, pp. 382-384.

4. Borenstein J., Koren Y. - The Vector Field Histogram—Fast Obstacle Avoidance for Mobile Robots, IEEE Trans. Robot. Autom. **7** (3) (1991) 278-288.
5. Ulrich I., Borenstein J. - VFH+: reliable obstacle avoidance for fast mobile robots, Proceedings of EEE Int. Conf. on Robotics and Auto, Vol. 2, 1998, pp. 1572-1577.
6. Gao M., Tang J., Yang Y., He Z., Zeng Y., - An Obstacle Detection and Avoidance System for Mobile Robot with a Laser Radar," Proceedings of IEEE 16th International Conference on Networking, Sensing and Control (ICNSC), (2019) 63-68.
7. Baras N., Nantzios G., Ziouzios D., Dasygenis M., - Autonomous Obstacle Avoidance Vehicle Using LIDAR and an Embedded System, Proceedings of 8th International Conference on Modern Circuits and Systems Technologies (MOCAST), (2019) 1-4.
8. Susnea I., Mînzu V., Vasiliu G., - Simple, real-time obstacle avoidance algorithm for mobile robots, Proceedings of 8th WSEAS Int. conf. on Compu. Intel. Man-Machine Sys. and Cyber., (2009) 24-29.
9. Susnea I., Filipescu A., Vasiliu G., Coman G., Radaschin A., - The bubble rebound obstacle avoidance algorithm for mobile robots," Proceeding of IEEE Int. Conf. on Contr. and autom. (ICCA 2010), (2010) 540-545.
10. Quinlan S., Khatib O., - Elastic bands: connecting path planning and control, Proceedings IEEE International Conference on Robotics and Automation, **2** (1993) 802-807.
11. Qu D., Hu Y., Zhang Y., - The Investigation of the Obstacle Avoidance for Mobile Robot Based on the Multi Sensor Information Fusion technology, Int. J. Mater. Mech. Manuf. **1** (4) (2013) 366-370.
12. Fox D., Burgard W., Thrun S., Cremers A. B., - A hybrid collision avoidance method for mobile robots, Proceedings of IEEE International Conference on Robotics and Automation (Cat. No.98CH36146), **2** (C) (1998) 1238-1243.
13. "EAIBOT D1." [Online]. Available: http://www.eaibot.com/product/D1.
14. "IR Sharp library on Github." [Online]. Available: https://github.com/guillaume-rico/SharpIR.