J. Inst. Eng. India Ser. B (April 2022) 103(2):449–460
https://doi.org/10.1007/s40031-021-00659-6

ORIGINAL CONTRIBUTION

Hardware Schemes for Autonomous Navigation of Cooperative-type Multi-robot in Indoor Environment

G. Divya Vani[1,2] · Srinivasa Rao Karumuri[1] · M. C. Chinnaiah[2]

Received: 16 September 2020 / Accepted: 28 July 2021 / Published online: 24 August 2021

Abstract This study proposes about the autonomous navigation of a multi-robot for transportation in indoor environments. This transportation is integrated with three foldings of VLSI architectures; they are 1) shortest optimal path planning 2) behavioral control between multi-robot with leadership swapping methods as per dynamic conditions and 3) obstacle avoidance by multi-robot. The hardware schemes have been designed for navigation of multi-robot with shortest path planning, based on an extended Dijkstra algorithm along with the Delaunay triangulation method. The behavioral control mechanism between the multi-robot is another challenge at the time of navigation and obstacle avoidance at both static and dynamic conditions in real-time scenario. The leader and follower approaches are deployed for cooperation between multi-robot to accomplish the task. The VLSI architectures are proposed for multi-robot navigation in the warehouse-type indoor environment. It is developed using Verilog HDL, simulated and synthesized with Xilinx Vivado 17.1. The Zynq-7000 SoC ZC702 FPGA is used as the target device.

Keywords FPGA · Multi-robot · Behavioral control · Dijkstra algorithm · Obstacle avoidance

Introduction

An extensive study has been conducted on autonomous service-based robot navigation in indoor environments. The indoor-based service robots are differentiated as per environment such as industry, home applications, hospital and cafeteria/hotel. In COVID-19 pandemic, the importance of service robot has been recognized for social services. The warehouses and industries are also integrated with robots for carrying the products.

The challenges of individual robots at the warehouse are carrying products with outbound due to bigger sizes than the basement of the robot, where it creates a chance for collision and cross docking at the environment. In this regard, individual robots can carry products when the product size is less than the robot basement size. Cranes can be used for carrying huge products, but they cannot be opted for the medium sized products due to space issues. It is one of the major challenges in warehouses and industrial environment. This paper provides solutions using cooperative-type multi-robot for carrying medium-sized products. In this context, the collaboration of multi-robot with the cooperative-type formation plays a vital role. This cooperative mechanism impacts the multi-robot to traverse with the behavioral control among them to defined destination nodes.

The research towards shortest path planning has been protracted in the robotic field from few decades, which is essential to accomplish the robotic navigation. Most of the navigational algorithms have been developed based on grid map and graph theory. The Dijkstra algorithm is one of the familiar shortest path planning algorithms, and it is widely used for 2D mobile robot navigation. Edsger Wybe Dijkstra is a Dutch scientist, expert in the computer field; he proposed shortest path planning as a Dijkstra's algorithm.

✉ Srinivasa Rao Karumuri
srinivasakarumuri@gmail.com

[1] Department of Electronics & Communication Engineering, K L University, Guntur, Andhra Pradesh, India

[2] Department of Electronics & Communication Engineering, B V Raju Institute of Technology, Narsapur, Medak, Telangana, India

In the recent years, the algorithm is modified with various aspects as per real-time scenario; the authors K. Wei et al. [1] represented about how a maximum load path problem can be dissolved with modified Dijkstra's algorithm. Similarly, the same algorithm has been used for optimal path and other applications like parking of robots [2].

The authors Dong Guo et al. [3] has mentioned about emissions effect while driving and also about fuel consumption by using the shortest path method. The automatic guided vehicle transmission in an environment with the grid method using Dijkstra algorithm was represented by Zheng Zhang et al. [4]. The team of authors Deepak Gautam et al. [5] discussed regarding Dijkstra-based shortest path for quad rotor helicopter movement. Sai Shao et al. [6] mentioned about the importance of shortest path in light-sport aircraft transmit. The authors M. Luo et al. [7, 8] have discussed regarding the extended version of the Dijkstra algorithm for optimal path planning. Thus, path planning is highly indeed in an industrial environment for material transportation.

In the existing system, at warehouse and industries the products are carried out by individual autonomous robot and they are not capable of carrying the medium-sized products. In this aspect, the multi-robot-based transportation methods are essential to carry medium-sized products using collaboration methods.

The collaboration of motion robots like PUMA 560 arms for carrying payload was presented by H. Bai et.al [9]. In similar lines, bio-inspired methods like coalition formation of multi-robot to execute gaming theory of robotics is mentioned by authors X. Liang et al. [10]. One of the robot formations with queue structure was developed by authors C. Fua et al. [11]. The authors Y. Kim et al. [12] investigated about formation of multi-robot with localization strategy.

The behavioral control between robots is another aspect of multi-robot formation. The cooperative control methods for vehicle formation have been developed by W.Ren et al. [13]. The other researchers G. Antonelli et al. [14] mentioned about control algorithms of null-space-based behavioral (NSB). The researchers have contributed control mechanism with fuzzy methods by authors J. Huang et al. [15], and multi-robot control with adaptive methods was discussed by J. Fan et al. [16]. The leader follower approaches are one of the best methods in the cooperative formation of multi-robot. The leader follower method for mobile vehicle was described by Shao et al. [17]. V. Kumar et al. [18] have mentioned regarding obstacle avoidance using Gaussian algorithms with a heuristic approach for leader follower formation control.

The obstacle avoidance can be performed by a multi-robot with two approaches: 1. distributive formation behavior control and 2. centralization formation behavior control. Among these, centralized approach is an appropriate method used in the flock group for transportation of products. The single-robot obstacle avoidance is mentioned with various control methods such as bug, vbug, bug2, voronoi diagram and graph theory. The key aspect of centralized multi-robot approach is leadership, and it plays a vital role in the execution of the multi-robot transportation with obstacle avoidance.

In this context, at dynamic situations, leadership swapping between multi-robot is essential. The affection-based concept between robots has been represented by F. Li et al. [19]. The practical leader–follower concept with the tracking control of multi-robots has mentioned by Y. Wang et al. [20]. The obstacle avoidance between multi-robot has been specified by X. Wu et al. [21]. The vision-based methodologies have arbitrated into robotics object avoidance by Y. Wang et al. [22] to evaluate the shape and size of the object in the process of obstacle avoidance execution. The authors D. Sakai et al. [23] have represented about the obstacle avoidance with preserving connectivity and without data transmission.

To accomplish the real-time robotic algorithms like navigation, obstacle avoidance and behavioral control is depended on computational devices and sensory data acquisition. The processing of these robotic control algorithms is required for parallel computing devices such as CPU/laptops. In this perspective, power consumption is another challenge while executing mobile robot; it impacts power management of battery mentioned by Yongguo Mei [24]. Field-programmable gate array (FPGA) will provide optimized power consumption with high computation which are appropriate for mobile robot. In this view, few researchers have deployed their respective navigation and obstacle avoidance algorithms on FPGA devices [25–28].

The proposed research work consists of novel hardware schemes for integration of multi-robot navigation using an extended version of the Dijkstra shortest path planning along with the bug2-based obstacle avoidance with the Delaunay triangulation approach. The other novelty of research work is formation of mechanism with behavioral control between multi-robot at both navigation and obstacle avoidance scenario. The static and dynamic leadership as part of leader and follower approaches is one of the novel approaches with hardware schemes. These three novel modelings are integrated into a single unit to perform parallel processing. The proposed VLSI architectures are versatile with parallel processing which is a novel approach.

This paper contribution is organized as follows: the proposed transportation based on multi-robot, shortest path navigation with static and dynamic obstacle avoidance and also along with behavioral control between multi-robot is discussed in section II. The VLSI architectures for proposed multi-robot services at warehouses and industrial applications are discussed in section III. In section IV the proposed work is illustrated with simulation results and synthesis results of the multi-robot navigation, behavioral control and leadership swapping methods. Section V deals with the conclusion and future research works of the paper.

Autonomous-based Multi-robots for Intelligent Transportation

The autonomous-based multi-robot systems are used for intelligent transportation in indoor environment such as warehouse and industrial environment. The extended version of the traditional Dijkstra algorithm is used for navigation of the multi-robot. The behavioral control approaches between the multi-robot have been developed with affection-based method, and obstacle avoidance is performed with integration of bug2 algorithm and Delaunay triangulation method.

The autonomous multi-robot system (MRS) with formation is represented in Fig. 1. The small MRS flock group is considered with four robots, each edge of the product is carried out by one of the MRS team. And MRS team will maintain equidistance with other teammates. The position of MRS team is represented by the robotic names as R1….R4, and it is represented in matrix formation as follows.

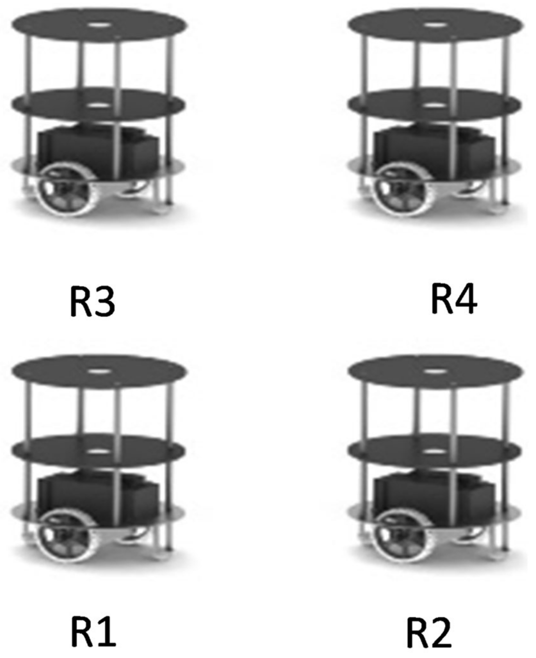

Fig. 1 Illustration of multi-robot formation with equidistance

Pseudo-Code of Hardware-Based Shortest Path Planning of MRS

The extended version of the Dijkstra algorithm for shortest path planning is framed using nodes (N) and route weights (W) as digraph D = (N, W). The distance between two nodes is calculated using the Euclidean distance method. Graphically represented in Fig. 2, the essence of the algorithm is to find shortest path when the initialization node and the destination node are given. Let us consider the MRS team starts from node 'A' to targeted node 'I'. The number of paths is available with different route weights; among them, the shortest route is A-D-F-I as per route weights as 30. The extended version is Dijkstra with Delaunay triangulation for MRS.

Inputs: The MRS team navigation is sensitized with ultrasonic sensors to traverse in the environment. This sensor array is placed on robot as follows: S_x = {S$_{F\_x}$, S$_{L\_x}$, S$_{R\_x}$, S$_{B\_x}$}, and digital compass is essential to calculate the angular deviation of mobile robot at node points, and it is noted as DC_R_x.

Step 1: Localization and destination nodes are initialized with explicit communication to MRS team's G_L. As the Dijkstra algorithm has the information on all possible paths and distances for various routes, among all the paths the shortest path is confined as per the inputs.

Step 2: Let us consider the navigation as mentioned, example: 'A' as a start node and 'I' as a destination node, shortest route: A-D-F-I. The MRS team will be registered with the shortest route, while MRS G_L will initiate their navigation with behavioral control mechanism.

Step 3: MRS team's G_L with S_x sensory information, E_D is initialized with route weights as E_D$_1$ for A to D nodes, D_En = 0, I_N = A and D_N = I, E$_{DT}$ = $\sum ED1\ldots.EDn$.

```
If (S_x = {0, x, x, x}) && BC_MRS = 1.
If (D_En ≠ E_D$_1$) && (R_w ≠ E$_{DT}$).
{
```

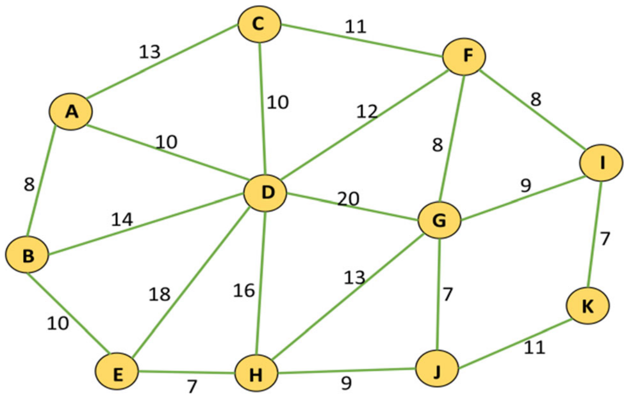

Fig. 2 Illustration of Dijkstra algorithm with route map

```
G_L ($M_L$ = 1, $M_R$ = 1).
}
*Else if* (*R_w* $\neq$ $E_{DT}$).
{
G_L ($M_L$ = 0, $M_R$ = 0).
E_D = E_$D_{x+1}$ (it means route weight from D to F).
}
Else
{
G_L ($M_L$ = 0, $M_R$ = 0).
P_N = D_N, MRS reached the destination node.
}
End
*Else if* (S_x = {1, x, x, x}) && BC_MRS = 0.
{wait until BC_MRS = 1}.
*Else if* (S_x = {1, x, x, x}) && BC_MRS = 1.
{
MRS team executes the obstacle avoidance algorithm.
Then retrieve back to navigation algorithm.
}
Else
{
G_L ($M_L$ = 0, $M_R$ = 0).
P_N = D_N, MRS team reaches the destination node
and waits for next instructions.
}
```

The above pseudo-code represents the shortest path planning with behavioral control. In the example, initial and destination nodes are ‘A’ and ‘I’, respectively. When the front sensor of MRS team’s G_L = 0 (leader) which indicates no obstacle, then after immediately, it checks for BC_MRS = 1, where data from BC_MRS give the presence of follower robot. MRS G_L moves forward until it completes route weights (R_w) by checking the distance from node to node (‘A’ to ‘D’, ‘D’ to ‘F’ and ‘F’ to ‘I’). The MRS team executes the obstacle avoidance followed by the navigation, if the front sensor of G_L = 1. G_L always checks for BC_MRS = 1 to maintain the synchronization among them.

Remark: In the navigation, MRS team will change global leader, according to the dynamic conditions, and navigation is continued from the same place.

Pseudo-Code of Hardware-based Behavioral Control of MRS

The behavioral control between MRS team formation is constructed with kinematic momentums as 1. initial stage, 2. locomotion stage and 3. terminus stage.

Figure 3 illustrates the MRS team’s behavioral control mechanism. Among MRS team, global leader will be decided with respect to flock group direction movement. Towards the north direction, R3 is considered as a leader. In similar lines for east, south and west directions, the R4, R2 and R1 will act as global leaders respectively.

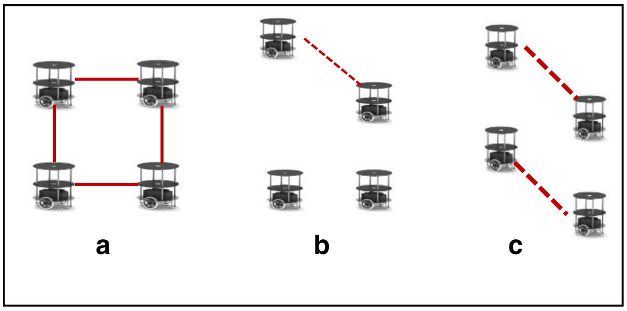

Fig. 3 Behavioral control between MRS

At the initial stage, the G_L robot takes the first movement as illustrated in Fig. 3b, the next step will be performed by I_L, and then R1 will be the leader as presented in Fig. 3c. The R2 and R4 react to the actions of the G_L and I_L.

The locomotion stage will be executed after the initial stage and robot’s synchronization momentum as illustrated in Fig. 3a. In terminus stage G_L, I_L and F_F sequentially stops the navigation and maintain the equidistance.

Inputs: S_x = {S_{F_x}, S_{L_x}, S_{R_x}, S_{B_x}}, IR_x = {IR_R_x, IR_B_x, IR_L_x, IR_F_x} and digital compass on robot x (DC_R_x).

Step 1: Initialization: DC_R_x = 0^0, represents robots are facing towards north direction. In such conditions R3 acts as G_L, R1 as I_L and R2 & R4 as F_F. Initial robot’s status is represented as shown in Fig. 1.

Step 2: If ((DC_R_GL = 0^0).
If (S_{F_3} = 0),
{
R3 robot takes the movement with the velocity of y units
} Perform step 3.
Else (S_{F_4} = 0) && (S_{L_4} @ $\vartheta_{45}{}^0$ = 1)
{
R3 is blocked with obstacle due to this, leadership swaps to R4, it acts as G_L, R3 as I_L, R4 takes $\vartheta_{45}{}^0$ right movement, followed by R3, R1 and R2 as mentioned in Fig. 3. R4 operated with a velocity of y units.
},
Else If ((DC_R_x = 90^0),
∴ Leadership swap to R4.
Similarly, R4 acts as GL, R3 as I_L, R2 & R1 act as F_F as represented in Fig. 4b
Else If ((DC_R_x = 180^0),
∴ Leadership swap to R2.
Similarly, R2 acts as GL, R4 I_L, R3 & R1 act as F_F as represented in Fig. 4d
Else ((DC_R_x = 270^0),
∴ Leadership swap to R1.

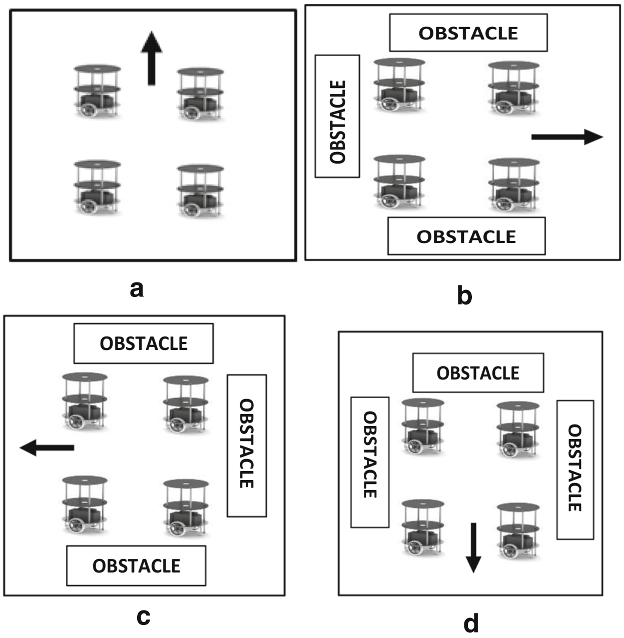

Fig. 4 Static position of robots as per indoor environment

Similarly, R1 acts as GL, R2 as I_L, R3 & R4 act as F_F as represented in Fig. 4c.

Step 3: The G_L robot moves with y velocity by maintaining the distance as illustrated in Fig. 3, $\therefore$ I_L robot acts as the follower moves with $(y + Rs)$ times velocity. The F_F moves with $2y$ times velocity. This velocity is preferred until each robot reaches DR_R distance values. Next fetch to step 4.

Step 4: Once initial speed synchronization completes with step 3, robots will move with equal y units velocity and maintain the DR_R distance.

The above pseudo-code represents the behavioral control among multi-robot and swapping of leadership in dynamic scenarios. The speed velocity of individual robot differs from each other in order to maintain the synchronization. Initially in Fig. 4a, robot R3 acts G_L as it is free from obstacles and it continues its leadership mechanism until it gets obstructed by all surroundings. In Fig. 4b, robot R4 acts as a global leader as all the other robots are surrounded by obstacles. Similarly, in Fig. 4c–d the leadership is owned by robot R2 and robot R1 respectively.

Pseudo-Code for Hardware-based Behavioral Control of MRS Obstacle Avoidance

```
Inputs: S_x = {$S_{F\_x}$, $S_{L\_x}$, $S_{R\_x}$, $S_{B\_x\}}$ & $D_{SF}$_x = {$D_{SF\_x}$, $D_{SL\_x}$, $D_{SR\_x}$, $D_{SB\ \_x}$} and digital compass on robot x (DC_R_x).
    Step 1: If ((DC_R_x = $0^0$).
    If ($S_{F\_3}$ = 1) && If ($S_{F\_4}$ = 1)
    {R3 & R4 robots detect an obstacle} as illustrated in Fig. 5a
    Go to step 2 Else {R3 & R4 Robots will move forward}.
    Step 2: If ($S_{F\_3}$ @ $\vartheta_{45}{}^{0} = 0$, $S_{F\_4}$ @ $\vartheta_{45}{}^{0} = 0$),
    R3 & R4 will take $\vartheta_{45}$ rotation towards the right, illustrated in Fig. 5b.
    R1 & R2 will delay until R3 &R4 accomplishment of the job.
    Go to step 3.
    Else ($S_{F\_3}$ @ $\vartheta_{315}{}^{0}$=0, $S_{F\_4}$ @ $\vartheta_{315}{}^{0}$=0)
    R3 & R4 will take $\vartheta_{45}$ rotation toward left,
    R1 & R2 will delay until R3 & R4 accomplishment of tasks.
    Go to step 4.
```

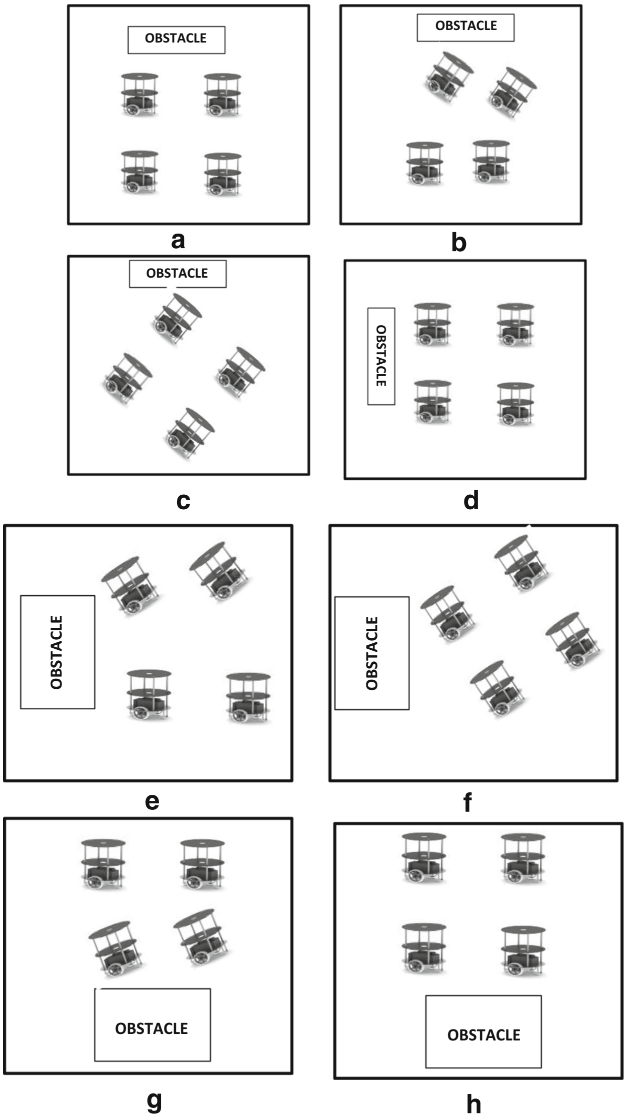

Fig. 5 **a-d** MRS obstacle avoidance with behavioral control. **e–h** MRS obstacle avoidance with behavioral control

Step 3: R3 & R4, R1 & R2 are rotated toward $\vartheta_{45}{}^{0}$. Move forward with the same angle as illustrated in Fig. 5c.

The leader robot will calculate the distance with internal odometer methodology.

a. The robots take rotation with ϑ_{45} towards the left. R3& R4 and R1 & R2 move forward until the next edge as illustrated in Fig. 5d.
b. The leader robots will take ϑ_{45} towards the left and followers will wait until leaders complete their action as illustrated in Fig. 5e.
c. The follower robots will rotate to ϑ_{45} left turn. The leaders and follower will move forward until decrement of odometer values as shown in Fig. 5f.
d. The leaders will rotate to right ϑ_{45} and follower in Fig. 5g.

The follower also completes the similar task as leaders as shown in Fig. 5h.

Step 4: Similar to step 3 movements, the robots will move, act as anti-direction of the step 3.

The above pseudo-code describes obstacle avoidance of MRS by maintaining behavioral control. The leader robots move forward by sensing the obstacle and waits for the delay time until it gets synchronized with follower robots. The odometer values are stored by the robots in order to retrieve the line of the path.

VLSI Architectures of Multi-Robot System for Intelligent Transportation

The MRS team is deployed with VLSI architecture for an intelligent transportation, which plays a vital role to accomplish successful transportation in industrial environment. The overall VLSI architecture of the MRS team for intelligent transportation is illustrated in Fig. 6; it consists of sensor module, control unit, navigation module, behavioral control module, obstacle avoidance module, processing and execution modules.

The input sensor module resided with infrared sensors (IR_1…IR_4), ultrasonic sensor module (US_1…US_4) with a pulse width distance converter and digital compass to evaluate the robot direction. The robots are integrated with infrared sensors on each side of bottom corner of the robots, it is placed to evaluate the beside robot movement. The ultrasonic sensors are placed in the middle of each side on the top of the robot. Digital compass is placed on the front side of the robot. The execution module is interfaced with stepper motors and wheels (M_L, M_R).

Control Unit of Robot for Intelligent Transportation

The control unit (CU) is the heart of the system, and the sensors are interfaced to CU by using protocols like UART. The digital compass has interfaced with UART protocol. Ultrasonic sensors are triggered and received its echo signal in the format of distance using PWDC. The output of PWDC is applied to navigation, obstacle avoidance and behavioral control module. The sensors have different frequencies such as ultrasonic sensor operates at 40 KHz and stepper motors operates at 10 KHz. An FPGA device (Xilinx Zynq-7000 SoC ZC702) is used as CU and it operates at 100 MHz frequency.

All these frequencies are synchronized by CU and operated in parallel. The CU takes care of the system until it executes unit. As part of synchronization, the data driven from the various modules are different in size and it is synchronized with appropriate action as per the condition.

The processing module (PM) is under control of CU, as per algorithm. The PM consists of processing elements (PEs), and each PE is interfaced with other modules such as navigation module, obstacle avoidance module and behavioral control module. PE of the navigation module in PMs will process the conditions of shortest path planning and it drives the execution module (EM). EM is interfaced with motors such as stepper and servo motors. The servo motors are placed on top of the robot and ultrasonic sensors are placed on servo motor; for example, servo motor is used by F_F robot to estimate the G_L robot movement as part of behavioral control. The stepper motors are used to drive the robot, and it is integrated with driver circuits and wheels. Total MRS team is coded in Verilog HDL and synthesized by using the Vivado 17.1 software.

Navigation Module of MRS

The VLSI architecture for shortest path navigation used MRS team for intelligent transport. The CU enables the navigation module; it sensitizes the sensory information of ultrasonic sensor and compass to localize the node. It receives the destination node as a task to all the robots with explicit communication. The navigation VLSI architecture is illustrated in Fig. 7, and this architecture is deployed in all robots. The G_L robot takes lead as a leader as per the robot's direction.

The shortest path module is an internal part of the architecture. In the known environment, learning is conducted earlier before the validation of navigation. In the learning process all different probabilities of short path as per Dijkstra and Delaunay triangulation are designed and registered in the look-up tables (LUT) of the FPGA. The sample inputs are as follows: initial localization of the robot at node 'A' and destination node are applied as ' I '.

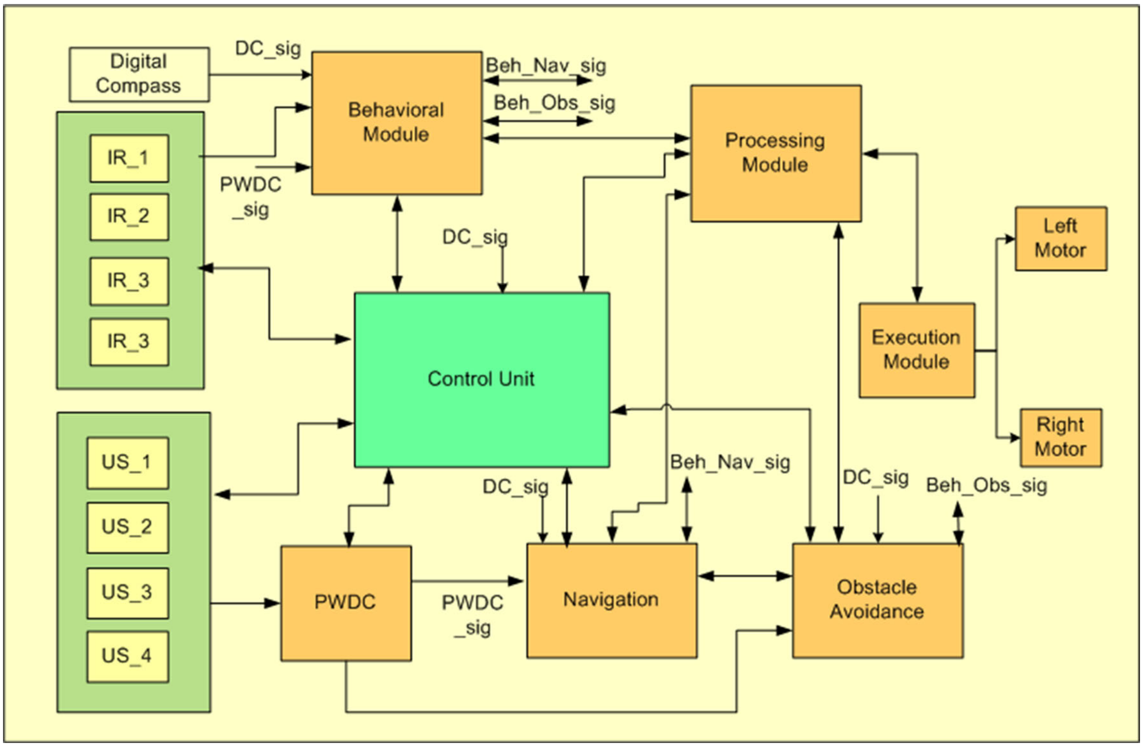

Fig. 6 Overall hardware scheme for MRS intelligent transportation

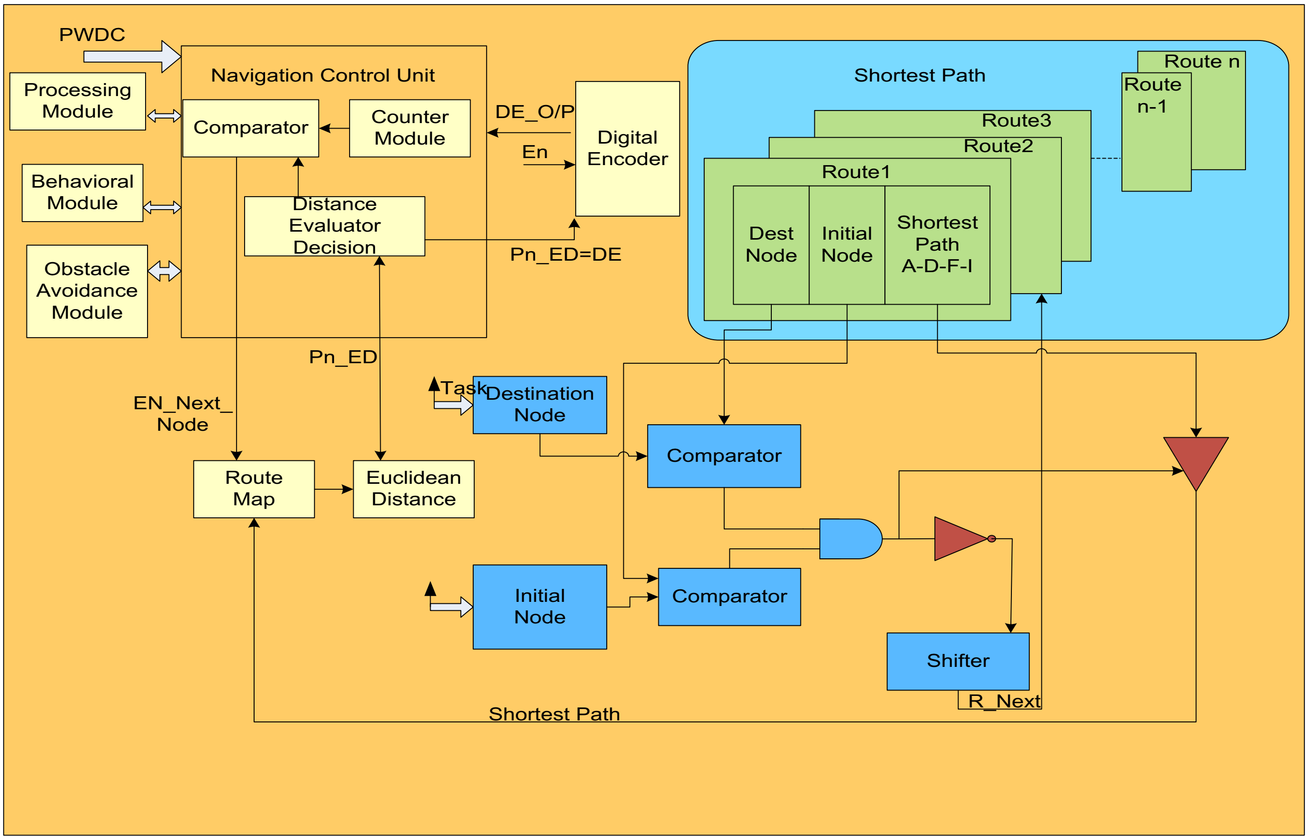

Fig. 7 VLSI architecture of MRS navigation module

The shortest path module routes reference values as compared with applied inputs. If both comparators output with high ’1’, we get results to select the shortest path, i.e., LUT slice. This path is stored in the route map module (Table 1).

The route map splits the route into smaller destinations and applies same to Euclidean distance. As per example, the shortest path is registered as A-D-F-I, A-D route is weighted as 10 units of the environment, and these distance values are mapped in the Euclidean distance module. The soft digital encoder is designed, and it calculates the distance of the robot traversal in the environment. The navigation control unit operates on two levels as synchronizing the total navigation module and another level as comparing between the Euclidean distance of the shortest path destinations and robot real-time distance registered at soft encoder. At the same time, navigation module will communicate in parallel with the behavioral module, processing module and obstacle avoidance. The computational challenges are at obstacle avoidance conditions. The distance error is calculated by navigation control unit and redefining with the Delaunay triangulation method.

Behavioral Control Module of MRS

The behavioral control module (BCM) consists of four internal modules such as leader module, follower module, swapping module and behavioral control unit. The BCM is enabled with the MRS team control unit, and BCM is sensitized with PWDC, digital compass UART output and interfaced with CU, navigation module, obstacle avoidance and feeding the response to the processing module. The behavioral control unit maintains the BCM and instructs the MRS team as per the algorithm.

The leader module is activated as per the behavioral control algorithm; G_L robot plays a behavioral role taking care of the environment conditions and follower robot movements. The novelty of the behavioral control has designed without data transmission. IR sensor of the robot estimates movement of the robot and ultrasonic sensors information is estimated for the follower distance from its point in dynamic conditions using Euclidean distance approach and velocity.

The follower module classifies with multiplexers; either it is in I_L or F_F mode. The I_L and F_F strategies are different in movements and distance conditions. Movement of MRS team formation shapes is changed from square to rhombus. After achieving synchronization of all robot movements, it enters into locomotion movements. In locomotion condition, G_L, I_L and F_F maintain equal distance using ultrasonic sensors and IR sensors.

The swap module is used to switch robot either in leader or follower mode. It is differentiated as per the direction and obstacle avoidance at deadlock situations.

Obstacle Avoidance of MRS

The MRS of an obstacle avoidance is enabled as per the conditions from both navigation module and BCM. The obstacle avoidance is designed based on the bug2 (illustrated in Fig. 5a–h) and Delaunay triangulation methods.

Table 1 The notations used in this paper

S_{F_X},S_{L_X}, S_{R_X}, S_{B_X}	Ultrasonic sensors at front, left, right & back side of the robots, where x = 1 it indicates as R_1 robot
DC_R_x	A digital compass for angular calculation
ϑ_r	Angle of reference
ϑ_c	Angle from digital compass
D_{SF}, D_{SL}, D_{SR} & D_{SB}	Distance value of the front side sensor (D_{SF}), left, right and back sensor distance value
R_s	Robotic frame size
D_{R_R}	Distance between robots
I_N	Initialization node
D_N	Destination ode
D_En	Digital encoder
E_D	Euclidean distance
M_L, M_R	Left motor, right motor
BC_MRS	Behavioral control MRS
R_w	Route weight
G_L	Global leader between robots
I_L	Intermediate leader
F_F	Global & cooperative follower
IR_F_x, IR_R_x, IR_L_x, IR_B_x	Infrared sensors at front, left, right & back side, when x = 1 stands as R_1 robot

PWDC is used for evaluating the type of object; either it is static or dynamic. In dynamic conditions, the velocity of the object is evaluated using a Euclidean distance method with the support of PWDC hardware module. The BCM is used at the time of bug2 algorithm execution. The triangulation with Euclidean distance is used to compute the size of the object and to perform as shown in Fig. 5g, h.

Results

MRS Synthesis report

Table 2 is about the device utilization of behavioral control of MRS team obstacle avoidance. As mentioned in Fig. 4, the hardware schemes of obstacle avoidance of MRS team are coded in Verilog HDL and synthesized by using the Xilinx Vivado 17.1 software. The area occupied by each submodule of the algorithm is stated as the number of look-up tables (LUT) occupied in the Zynq board. All the submodules of the algorithm used 14,992 slices of LUT. Overall device utilization is about 28.18% which is achieved with the proposed work using Zynq-7000 SoC ZC702-type FPGA. The static power consumed by the proposed research work is 6.2 watts, as per Xilinx power analysis.

Simulation Results

Figure 8 represents the control unit actions of overall simulation results. It represents the robot movements, and it checks the status of all control signals. It is representing the leadership of robot and swapping toward as per robot conditions as shown in Fig. 3a. The timing analysis is decreased to achieve the results in simulation.

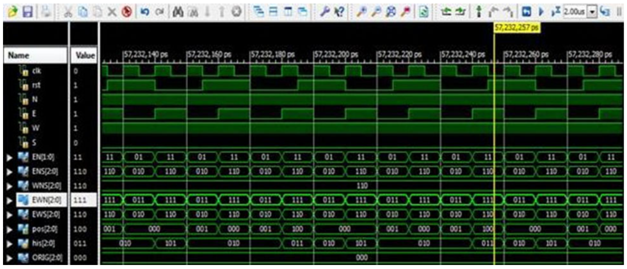

Fig. 8 Simulation results of overall control unit module

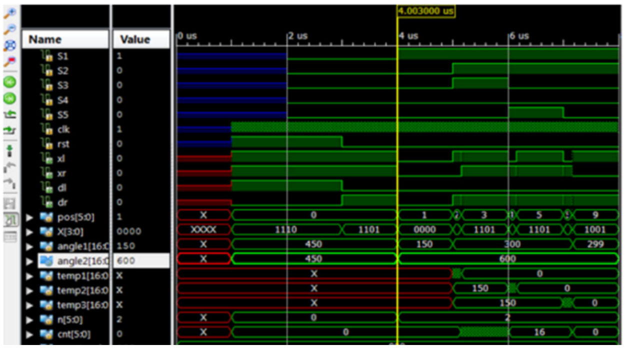

Fig. 9 Simulation results of obstacle avoidance of robot R3

Figure 9 represents about the obstacle avoidance of the R3 robot. The SF, SB and SL are facing obstacles, before SR no obstacle as mentioned in Fig. 3b that robot takes action to the right turn. The temp1, temp2 and temp3 register is used to store the data of distance traveled by the robot in overcoming the obstacle. These position changes are mentioned at POS [2:0].

In Fig. 10, simulation shows the bug2 obstacle avoidance algorithm from initial point to destination for the robot R4. The R4 moves with different velocities of speed to maintain the synchronization with R2 with respect to the digital compass input and sensory information. The

TABLE 2 Device utilization of MRS

Name of module	No. of slices LUT occupied in (Zynq board)
Navigation module	7980
Leader behavior module	212
Swap module	148
Follower behavior module	244
Digital compass interface using UART	268
Control unit module	2660
Sensor module	168
Motor control	120
Obstacle avoidance	3192
Total	14,992
Overall device utilization	28.18%

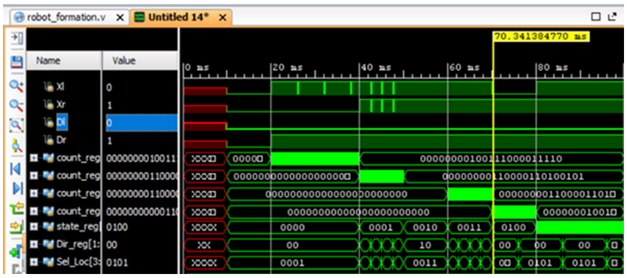

Fig. 10 Simulation results of R4 from the initial point to destination point

variations of directions and its self-localization are shown in Fig. 10.

In Fig. 11, simulations show the behavior of synchronization between robot R4 and robot R2. The front sensor data are continuously checked by R2 to get the information about the presence of R4. The motor signals are shown in the simulation.

In the Fig. 12, the task is given to the robot as the initial node as A = 4’b0001 and destination node as I = 4’b1001. For a given task, mode is switched to 4b0001 where all the possible paths from ‘A’ to all destination nodes are available, and for a given input, shortest path is A_D_F_I. The robot navigates through A = 4’b0001, D = 4’b0100, F = 4’b0110 and I = 4’b1001 indicating the register node_reg. Respective motor signals are executed through xl_next, dl_next, xr_next and xr_next.

Comparison

The proposed research work is integrated with multi-robot navigation, behavioral control and obstacle avoidance by multi-robot methods. K. Wei et al. [1] have discussed an extended version of Dijkstra algorithm, but it is represented for individual robot. F. Li et al. [19] have presented about affection-based leader selection method among MRS. The proposed work is on similar lines of the Dijkstra algorithm for multi-robot system (MRS) navigation along with behavioral control which is one of the successful accomplishments and MRS performing static obstacle avoidance with the flock group is another achievement. P. R. Kumar et al. [29] have presented hardware schemes for a single robot. The proposed hardware schemes are developed with optimized coding techniques; with this it consumes less look-up tables and slices of FPGA around 28% of Zynq Zed board FPGA device. The power consumed for robot navigation of 40 nodes (and 52 edges) is 6.2 watts, and it is analyzed by the Xilinx Power Estimator (XPS) tool (Fig. 12).

Conclusion

The proposed research work has been developed for intelligent transportation by autonomous multi-robot system (MRS). The novel contributions of research work deployed for intelligent transportation are shortest path planning of robotic navigation using Dijkstra algorithm, BCM with leadership swapping method and obstacle avoidance in real-time scenario. The VLSI architecture has been developed for successful accomplishment of algorithms. The state of the art is the integration of three algorithms, robot navigation, and its equivalence hardware scheme has been developed. The behavioral control played a vital role in communicating MRS team without data transmission which is one of the novel approaches. This method decreases the chip area of the FPGA. Another novel approach is accomplished of obstacle avoidance in lines of bug2 for intelligent transportation in real time with BCM. This hardware scheme is simulated and synthesized using the Xilinx Vivado 17.1 software. The device utilization represents the hardware schemes occupancy of the look-up table (LUT). The 28.18% of LUT are used as part of Zynq-7000 SoC ZC702-type FPGA. As per the Xilinx

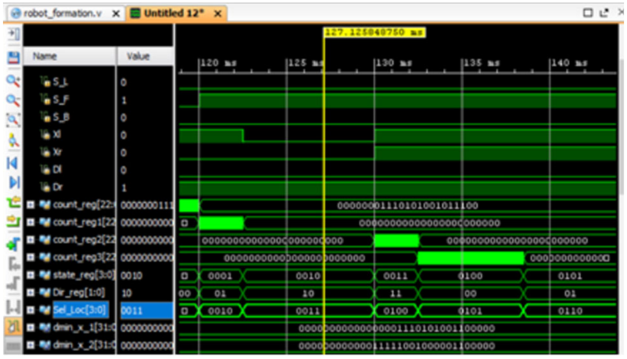

Fig. 11 Simulation results of R2 with synchronization of R4

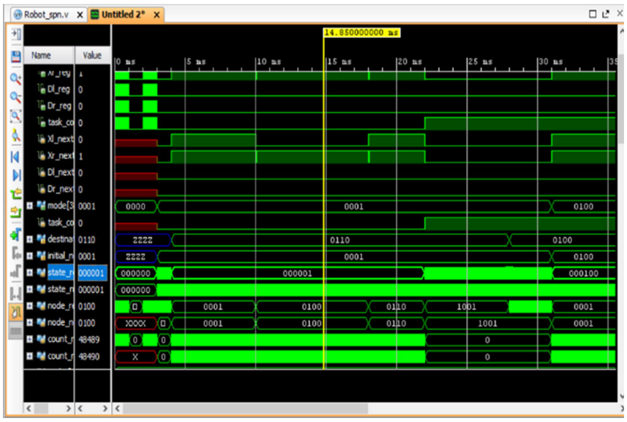

Fig. 12 Shortest path navigation

power analysis, it consumed 6.2 watts of static power for proposed research work.

Funding The authors thank to Science and Engineering Research Board, Department of Science and Technology, New Delhi, India, for funding this work as part of a project entitled "A Companion type Assistive System for Elderly People using VLSI based Service Robot" under the Science and Engineering Research Board (SERB-ECR).

Declarations

Conflict of interest The authors declare that they have no conflict of interest.

References

1. K. Wei, Y. Gao, W. Zhang, S. Lin, A modified Dijkstra's Algorithm for solving the problem of finding the maximum load path. *2019 IEEE 2nd International Conference on Information and Computer Technologies (ICICT)*, 2019, pp. 10–13. https://doi.org/10.1109/INFOCT.2019.8711024
2. A. Alyasin, E.I. Abbas, S.D. Hasan, An efficient optimal path finding for mobile robot based on Dijkstra Method. *2019 4th Scientific International Conference Najaf (SICN)*, 2019, pp. 11–14. https://doi.org/10.1109/SICN47020.2019.9019345
3. D. Guo, J. Wang, J.B. Zhao, F. Sun, S. Gao, C.D. Li, M.H. Li, C.C. Li, A vehicle path planning method based on a dynamic traffic network that considers fuel consumption and emissions. Sc. Total Environ. **663**, 935–943 (2019). https://doi.org/10.1016/j.scitotenv.2019.01.222
4. Z. Zhang, Q. Guo, J. Chen, P. Yuan, Collision-Free Route Planning for Multiple AGVs in an Automated Warehouse Based on Collision Classification. IEEE Access **6**, 26022–26035 (2018). https://doi.org/10.1109/ACCESS.2018.2819199
5. D. Gautam, C. Ha, Control of a quadrotor using a smart self-tuning fuzzy PID controller. Int. J. Adv. Robotic Syst. **10**, 1–9 (2013). https://doi.org/10.5772/56911
6. W.C. Lu, M.T. Lee, M.W. Wang, Route planning for light-sport aircraft in constrained airspace. Procedia Eng. **67**, 140–146 (2013). https://doi.org/10.1016/j.proeng.2013.12.013
7. M. Luo, X. Hou, S.X. Yang, A multi-scale map method based on bioinspired neural network algorithm for robot path planning. IEEE Access **7**, 142682–142691 (2019). https://doi.org/10.1109/ACCESS.2019.2943009
8. M. Luo, X. Hou, J. Yang, Surface optimal path planning using an extended Dijkstra algorithm. IEEE Access **8**, 147827–147838 (2020). https://doi.org/10.1109/ACCESS.2020.3015976
9. H. Bai, J.T. Wen, Cooperative load transport: a formation-control perspective. IEEE Trans. Rob. **26**(4), 742–750 (2010). https://doi.org/10.1109/TRO.2010.2052169
10. X. Liang, Y. Xiao, Studying bio-inspired coalition formation of robots for detecting intrusions using game theory. IEEE Trans. Syst. Man cybern. Part B (Cybern.) **40**(3), 683–693 (2010). https://doi.org/10.1109/TSMCB.2009.2034976
11. C. Fua, S.S. Ge, K.D. Do, K. Lim, Multirobot formations based on the queue-formation scheme with limited communication. IEEE Trans. Rob. **23**(6), 1160–1169 (2007). https://doi.org/10.1109/TRO.2007.911198
12. Y. Kim, J. Kwak, D. Hong, J. Ahn, S. Wee, J. An, Localization strategy based on multi-robot collaboration for indoor service robot applications. *2013 10th International Conference on Ubiquitous Robots and Ambient Intelligence (URAI)*, 2013. pp. 225–226. https://doi.org/10.1109/URAI.2013.6677348
13. W. Ren, Consensus strategies for cooperative control of vehicle formations. IET Control Theory Appl. **1**(2), 505–512 (2007). https://doi.org/10.1049/iet-cta:20050401
14. G. Antonelli, F. Arrichiello, S. Chiaverini, Experiments of formation control with multirobot systems using the null-space-based behavioral control. IEEE Trans. Control Syst. Technol. **17**(5), 1173–1182 (2009). https://doi.org/10.1109/TCST.2008.2004447
15. J. Huang, N. Zhou, M. Cao, Adaptive fuzzy behavioral control of second-order autonomous agents with prioritized missions: theory and experiments. IEEE Trans. Industr. Electron. **66**(12), 9612–9622 (2019). https://doi.org/10.1109/TIE.2019.2892669
16. J. Fan, Y. Liao, Y. Li, Q. Jiang, L. Wang, W. Jiang, formation control of multiple unmanned surface vehicles using the adaptive null-space-based behavioral method. IEEE Access **7**, 87647–87657 (2019). https://doi.org/10.1109/ACCESS.2019.2925466
17. J. Shao, G. Xie, L. Wang, Leader-following formation control of multiple mobile vehicles. IET Control Theory Appl. **1**(2), 545–552 (2007). https://doi.org/10.1049/iet-cta:20050371
18. D. Panagou, V. Kumar, Cooperative visibility maintenance for leader-follower formations in obstacle environments. IEEE Trans. Rob. **30**(4), 831–844 (2014). https://doi.org/10.1109/TRO.2014.2304774
19. F. Li, Y. Ding, M. Zhou, K. Hao, L. Chen, An affection-based dynamic leader selection model for formation control in multi-robot systems. IEEE Trans. Syst. Man Cybern. Syst. **47**(7), 1217–1228 (2017). https://doi.org/10.1109/TSMC.2016.2564931
20. Y. Wang, D. Wang, S. Yang, M. Shan, A practical leader-follower tracking control scheme for multiple Nonholonomic mobile robots in unknown obstacle environments. IEEE Trans. Control Syst. Technol. **27**(4), 1685–1693 (2019). https://doi.org/10.1109/TCST.2018.2825943
21. X. Wu, S. Wang, M. Xing, Observer-based leader-following formation control for multi-robot with obstacle avoidance. IEEE Access **7**, 14791–14798 (2019). https://doi.org/10.1109/ACCESS.2018.2889504
22. Y. Wang, M. Shan, Y. Yue, D. Wang, Vision-based flexible leader-follower formation tracking of multiple nonholonomic mobile robots in unknown obstacle environments. IEEE Trans. Control Syst. Technol. **28**(3), 1025–1033 (2020). https://doi.org/10.1109/TCST.2019.2892031
23. D. Sakai, H. Fukushima, F. Matsuno, Leader–follower navigation in obstacle environments while preserving connectivity without data transmission. IEEE Trans. Control Syst. Technol. **26**(4), 1233–1248 (2018). https://doi.org/10.1109/TCST.2017.2705121
24. Y. Mei, Lu. Yung-Hsiang, Y.C. Hu, C.S.G. Lee, Deployment of mobile robots with energy and timing constraints. IEEE Trans. Robot. **22**(3), 507–522 (2006). https://doi.org/10.1109/TRO.2006.875494
25. M.C. Chinnaiah, S. Dubey, K. Anusha, J. Vani, G. Divya Vani, A versatile path planning algorithm with behavioural control using FPGA based robots. *2016 10th International Conference on Intelligent Systems and Control (ISCO)*, 2016, pp. 1–6. https://doi.org/10.1109/ISCO.2016.7727076
26. M.C. Chinnaiah , T.S. Savitri, P.R. Kumar, A novel approach in navigation of FPGA robots in robust indoor environment. *2015 International Conference on Advanced Robotics and Intelligent Systems (ARIS)*, 2015, pp. 1–6. https://doi.org/10.1109/ARIS.2015.7158382
27. H.B. Barua, K.C. Mondal, Approximate computing: a survey of recent trends—bringing greenness to computing and communication. J. Inst. Eng. India Ser. B **100**, 619–626 (2019). https://doi.org/10.1007/s40031-019-00418-8

28. P.R. Kumar, K. Sridharan, VLSI-efficient scheme and FPGA realization for robotic mapping in a dynamic environment. IEEE Trans. Very Large Scale Integr. (VLSI) Syst. **15**(1), 118–123 (2007). https://doi.org/10.1109/TVLSI.2007.891100
29. T.K. Priya, P. Rajesh Kumar, K. Sridharan, A hardware-efficient scheme and FPGA realization for computation of single pair shortest path for a mobile automaton. Microprocess. Microsyst. **30**(7), 413–424 (2006)

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.