Mobile Robot Motion Planning Based on a Concept of Attractive and Repulsive Forces and Variable Target and Robot Perception Circles

Nedim Osmic, Jasmin Velagic and Adnan Tahirovic

Abstract— **This paper proposes a mobile robot motion control and planning system for trajectory tracking and obstacle avoidance in a prior unknown robot environment. The proposed system has two-level control and planning architecture: the higher is used to generate a path, while the lower provides the control actions that drive the robot. The planning level represents a reactive planer which determines on-line way-points during the robot's movement towards the target and allowing the robot to move autonomously through an environment without colliding with obstacles. The main objective of this algorithm is to reduce the number of obstacles that are taken into consideration when determining the intermediate target point (way-points) in the movement towards the target location. This proposed algorithm is based on the concept of calculating the intersection of the variable target circle and the robot perception circle (VTPC), as well as attractive and repulsive forces. The lower level includes a fuzzy logic controller that drives the robot along generated online trajectory. It compares the current position of the mobile robot with the desired position, generating the appropriate linear speeds for the robot's wheels to reach the target point in the shortest possible time. A series of simulations demonstrate its effectiveness in generating and executing the paths in various unknown robot environments.**

I. INTRODUCTION

One of the crucial requirements for autonomous mobile robots is their ability to navigate in unknown or partially known environments [1], [2]. In the case of autonomous task execution, it is important that the robot can reach a target point in an unknown environment or avoid obstacles at a safe distance during its motion [3]. Trajectory tracking is a common task in mobile robotics and it can be defined as a robot following a predefined path while minimizing deviation errors at all points through which the mobile robot passes. There are many methods that address this problem. In [4], [5], [6], the model predictive controllers are employed, while an approach based on an adaptive control is proposed in [7]. The problem of trajectory tracking is also solved using fuzzy logic [8], [9], as well as neural networks [10], [11], [12]. Therefore, sliding mode controllers are also widely used in solving this problem [13], [14]. Furthermore, several techniques to address the aforementioned problem include approaches based on the potential field method [15], tracking a virtual trajectory [16] and error propagation [17]. It should be noted that the approaches used in these studies address only a problem of tracking the desired trajectory of the mobile robot.

*This work was not supported by Aerostream Horizon project

Nedim Osmic, Jasmin Velagic and Adnan Tahirovic are with Faculty of Electrical Engineering, University of Sarajevo, 71000 Sarajevo, Bosnia and Herzegovina (nosmic, jvelagic, atahirovic)@etf.unsa.ba

Another type of problem occurs when the mobile robot moves towards the target location in the environment populated by obstacles. There are two main approaches to solve this problem [18]: a global approach that generates a path off-line within a known environments and a local approach that produces the path on-line in a partially or completely unknown environment. The most used global motion planners are: A^*, D^* and Dijkstra algorithms for graph search, random tree search and probabilistic road-map methods [19]. The main shortcoming of these methods is that they require prior knowledge of the robot environment. In the case of an unknown static or dynamic environments, the mobile robot may collide with obstacles. In order to navigate the mobile robot through such an environment, the local planners are required. Local planners based on the potential field methods belong to this class of planners. In these methods, the target asserts attractive force on the robot, and obstacles assert repulsive forces [20]. The robot is considered as a point in an artificially created potential field and moves in the direction of the field gradient towards the target point. Our paper considers a slightly modified approach based on the potential field method. Furthermore, a local-level planning is combined with sensor readings, which can be ultrasonic, infrared, or laser. In this way, the mobile robot perceives the environment and using acquired sensors readings makes plan how to move through an environment [21]. It should be noted that the mobile robot operates in a prior unknown environment with static and dynamic obstacles.

In our paper, we proposed a two-level control and path planning system which is capable to navigate in both static and dynamic environment while moving towards the target point. The higher level represents a reactive action approach that assigns repulsive forces to obstacles and an attractive force to the target point. The computation of a new intermediate target point is based on the vector summation of attractive and repulsive forces, which generates a new target point that the robot needs to reach. In this way, the mobile robot avoids detected obstacles, and if there are no obstacles, the mobile robot moves straight towards the final target point. To speed up the calculation of the intermediate target point, we reduce the number of obstacles that affect the robot motion to only those located in the cross section of the target circle and the robot perception circle (VTPC). After reducing the number of relevant obstacles, the reactive planner generates the intermediate target point that the mobile robot needs to reach. This point is considered as a referent point in the lower level control module that implements the fuzzy logic controller. This ensures that the mobile robot reaches an intermediate target point, as well as the final target point, while avoiding obstacles using only local information.

The paper is organized as follows. The overall two-level control and planning robot system is described in Section II. Section III discusses the design of the fuzzy controller. A description of the proposed reactive planner for moving the mobile robot, through unknown obstacles populated environment, toward a target point is the subject of Section IV. The simulation results for different scenarios are shown in Section V. Conclusions and recommendations for future work are presented in Section VI.

II. CONTROL SYSTEM DESCRIPTION

The used a two-wheeled differential drive mobile robot is depicted in Fig. 1 [22]. The robot has two drive wheels located on the same axis and a free wheel at the back of the mobile platform for stability. The position of the mobile robot in the global coordinate system *X, Y* can be described by the position of point A, which is located in the center of the wheel axle, or by the position of point C, which is located at the center of mass of the mobile platform, and the orientation between the local frame of the mobile robot x_m, y_m and the global frame.

Due to the non-slip and rotating wheels, the mobile robot can only move in the direction perpendicular to the axis of the drive wheels (not sideways). The nonholonomic constraint of the mobile robot can be expressed as:

$$\dot{y}cos(\theta) - \dot{x}sin - d\dot{\theta} = 0 \tag{1}$$

where d is the distance from the center of gravity of the mobile platform to the center of the axle. The kinematic equations of the mobile robot are as follows:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} cos(\theta) & 0 \\ sin(\theta) & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix} \tag{2}$$

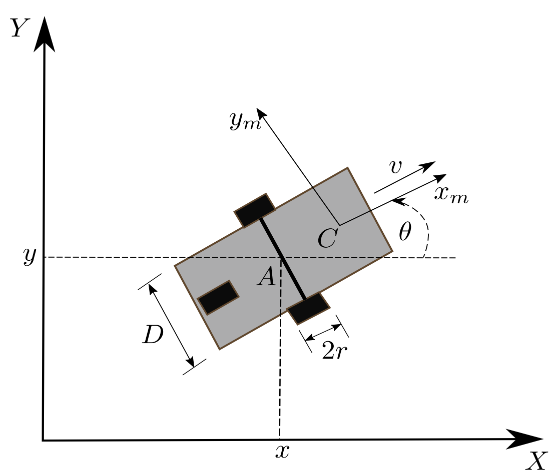

Fig. 1. Nonholonomic mobile robot with differential drive

$$\begin{bmatrix} v \\ \omega \end{bmatrix} = \begin{bmatrix} \frac{1}{2} & \frac{1}{2} \\ \frac{1}{2D} & -\frac{1}{2D} \end{bmatrix} \begin{bmatrix} v_R \\ v_L \end{bmatrix} \tag{3}$$

where x and y are the coordinates of the mobile robot's center, θ is the angle for the orientation of the mobile robot relative to the positive X-axis, v and ω are the linear and angular velocities of the mobile robot, v_R and v_L are the linear velocities of the right and left wheels, r is the wheel radius, and D is the distance between the left and right wheels. Combining (2) and (3) yields:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \frac{1}{2}cos(\theta) & \frac{1}{2}cos(\theta) \\ \frac{1}{2}sin(\theta) & \frac{1}{2}sin(\theta) \\ \frac{1}{2D} & -\frac{1}{2D} \end{bmatrix} \begin{bmatrix} v_R \\ v_L \end{bmatrix} \tag{4}$$

The integration of these equations, assuming that there is no slip and perfect odometry, allows us to determine the current position of the mobile robot (x_t, y_t) and the orientation (θ_t) at any time. Knowing the current position and orientation, by determining the linear velocities for the right and left wheels (v_R, v_L), the robot can be guided to the desired position and orientation. If we choose the control variables as the linear velocities of the right and left wheels $(u_1 = v_R,\ u_2 = v_L)$, then by suitable combinations of these control variables (u_1, u_2) the mobile robot can be efficiently controlled.

As mentioned above, the proposed system for motion planning and control has two levels, as shown in Fig. 2. The inputs to the reactive planner are: the given target point (x_t, y_t), the current position (x, y) and the orientation of the mobile robot (θ) along with information about the presence/absence of obstacles, obtained from sensors on the mobile robot. Based on this information, the reactive planner generates a new intermediate target point that the mobile robot needs to reach. For this new intermediate target point, the deviation in position of the mobile robot $R = \sqrt{(x-x_t)^2 + (y-y_t)^2}$ and the desired orientation $(\Delta\theta = \theta_t - \theta)$ are calculated to reach the target point, assuming that the robot moves at the maximum allowed speed. The position

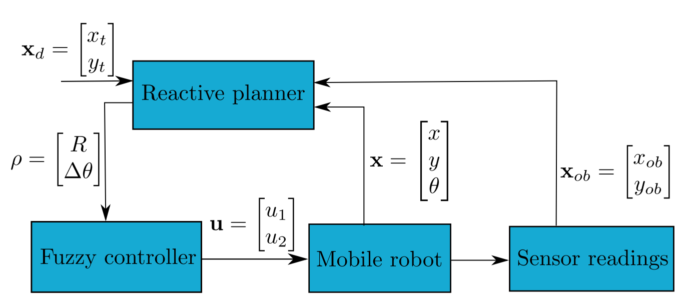

Fig. 2. Proposed system for the motion planning and control based on reactive planner and fuzzy controller

and orientation error information $\rho = [R; \triangle\theta]$ is forwarded to the fuzzy controller, which produces the control outputs (u_1, u_2). These control variables are applied to the right and left wheels of the mobile robot, with the aim of reducing errors in position and orientation.

III. SYNTHESIS OF THE FUZZY CONTROLLER

Consider the problem of driving the mobile robot to a target point in an obstacle-free environment. Suppose the initial position of the mobile robot in the global coordinate frame is (x, y). At this position the robot is headed in a direction making an angle θ with the X axis. The desired target point is located in (x_t, y_t) and the line that passes through points (x, y) and (x_t, y_t) creates the angle θ_t with the X axis, as shown in Fig. 3.

Since the task of the mobile robot is to reach the desired target point, the shortest path would be a straight line. Therefore, the desired orientation of the mobile robot is calculated so that it can move straight towards the target point at the highest possible velocity. At any time, the difference between the desired and current orientation is:

$$\Delta\theta = \theta_t - \theta \tag{5}$$

Based on the magnitude and sign of the angle $\Delta\theta$, the following actions can be taken: 1) If $\Delta\theta$ is greater than some critical value in absolute terms, it would be necessary to rotate the mobile robot in place until the difference between the desired and current orientation falls below the critical value. 2) If the angular error is less than the critical value, but greater than a threshold considered to be zero error, then different velocities are forwarded to the left and right wheels so that correction to the desired angle occurs during movement. 3) If $\Delta\theta$ is less than the absolute zero error threshold, then the maximum allowed velocities are transmitted to the left and right wheels. This procedure is repeated until the difference between the current and target points falls below 0.01 m. Since this procedure includes the IF-THEN propositions, it can be easily implemented using fuzzy logic. The fuzzy controller proposed to reach the target point has two inputs and two outputs. The input variables are as follows:

- Distance (diameter) between the current and target points - R
- Difference between the desired and current orientation - $\Delta\theta$.

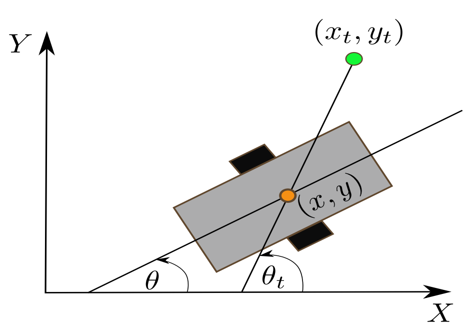

Fig. 3. Current and desired orientation of the mobile robot

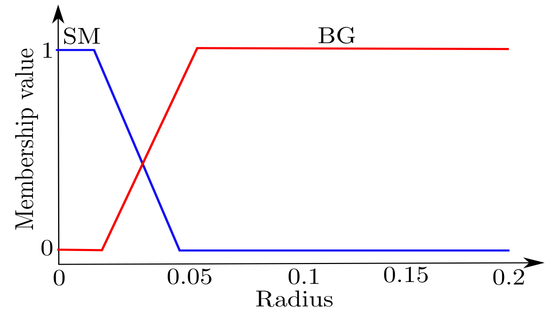

Fig. 4. Membership functions for the input variable diameter (SM-small, BG-big)

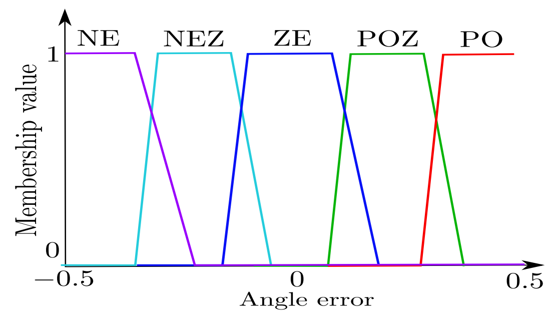

Fig. 5. Membership functions for the input variable angle error (NE-negative, NEZ-negative zero, ZE-zero, POZ-positive zero, PO-positive)

The output of the fuzzy controller are the control variables u_1 and u_2, which represent the linear speeds of the right and left wheels, respectively. The proposed fuzzy controller is of Sugeno type.

The values of the input variable "diameter" are divided into two subdomains, and the shapes of its membership functions are depicted in Fig. 4. The another input variable "angle error" is divided into five subdomains, and the shapes of the membership functions for this variable are shown in Fig. 5. The output variables u_1 and u_2 are divided into three subdomains, as shown in Fig. 6.

Based on the defined input variables, the output is determined as shown in Tables I i II. The obstacle avoidance mechanism is described in the next section.

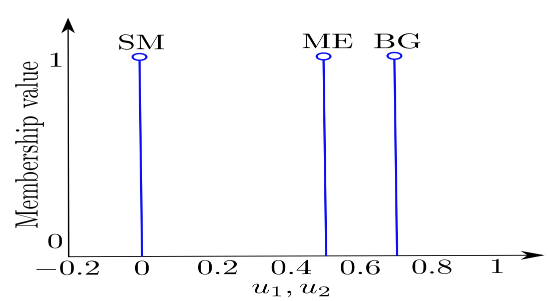

Fig. 6. Membership functions for the control output variable (SM-small, ME-medium, BG-big)

TABLE I

CONTROL OUTPUT FOR VARIABLE u_1

R/$\Delta\theta$	NE	NEZ	ZE	POZ	PO
SM	SM	SM	SM	SM	SM
BG	ME	BG	BG	ME	SM

TABLE II

CONTROL OUTPUT FOR VARIABLE u_2

R/$\Delta\theta$	NE	NEZ	ZE	POZ	PO
SM	SM	SM	SM	SM	SM
BG	SM	ME	BG	BG	ME

IV. REACTIVE MOBILE ROBOT MOTION PLANNER BASED ON VTCP

The reactive planner exploits an approach of avoiding obstacles and guiding the robot towards the target point based on the vector summation of attractive and repulsive forces [23]. The target point can be considered as a point that generates an attractive force towards the robot, which is expressed by a vector. It starts at the current position of the mobile robot and ends at a point along the line between the center of the mobile robot and the target point. The intensity of the attractive force is:

$$R_a = \sqrt{\Delta x_a^2 + \Delta y_a^2} = \sqrt{(x - x_t)^2 + (y - y_t)^2} \tag{6}$$

It decreases as the robot approaches the target point. In the case when the distance between the mobile robot and the obstacle is less than 1 meter, the intensity of the repulsive forces can become significantly higher than the intensity of the attractive force. Then the mobile robot would not be able to reach the target point. For this reason, the calculation of the intensity of the attractive force is carried out as follows:

$$F_a = \begin{cases} k\,|R_a|, & za \quad |R_a| < 1m \\ |R_a|, & za \quad |R_a| > 1m \end{cases} \tag{7}$$

where $k = 5$. This amount was determined experimentally. In experiments, obstacles were placed at various distances from the target point. For lower values of the parameter k, the mobile robot was unable to reach the target point, while for higher values, collisions with obstacles occurred. This choice of parameter k ensures that the mobile robot reaches the target point. In the proposed approach, obstacles are considered as repulsive forces. The intensity of the repulsive force depends on how far the obstacle is from the robot and is calculated by the expression:

$$F_r = \frac{1}{R_{ob}^2} \tag{8}$$

R_{ob} represents the distance between the robot and the obstacle. R_{ob} is calculated by the expression:

$$R_{ob} = \sqrt{\Delta x_{ob}^2 + \Delta y_{ob}^2} = \sqrt{(x - x_{ob})^2 + (y - y_{ob})^2} \tag{9}$$

(x, y) and (x_{ob}, y_{ob}) are the coordinates of the center of the mobile robot and the obstacle, respectively. While moving, the mobile robot scans its environment. If more than one

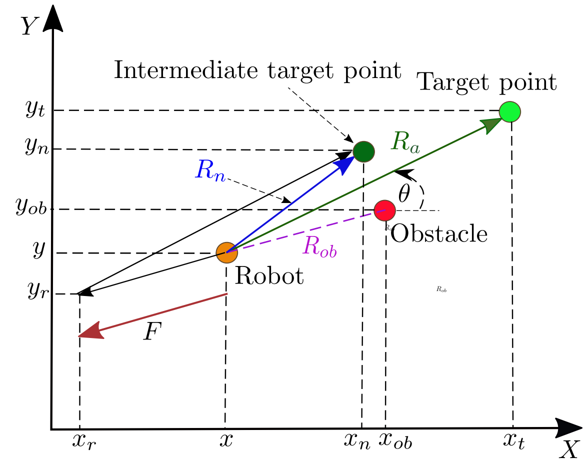

Fig. 7. Determining the new intermediate target point in the presence of one obstacle and the final target point

obstacle is detected, the currently closest relevant obstacle is considered. The algorithm vectorially sums the attractive forces (caused by the target point) and the repulsive forces (arising from obstacles) to generate a new intermediate target point. Thus, when the mobile robot detects the closest obstacle, with coordinates (x_{ob}, y_{ob}), the algorithm assigns a repulsive force to this obstacle that acts to push the robot away from it. The described procedure for calculating the new intermediate target point is shown in Fig. 7.

While the mobile robot is moving it scans its environment and can detect obstacles in its vicinity. However, even when obstacles are detected, not all of them may affect the movement of the mobile robot towards the target point. Which obstacles will be taken into account for calculating repulsive forces is determined by the proposed VTPC approach. This path planning approach, based on variable target and robot perception circles, is represented in Fig. 8. The robot's field of view (robot perception circle marked in red) has the constant radius R_f, whereas the circles between the target point and the robot (variable target circle marked in blue) have variable radii equal to half their mutual distance. The field of view of a mobile robot depends on the sensors used and their accuracy. In our simulation study, the sensors have a range of 4 meters, which means that any obstacle at a distance greater than 4 meters is invisible to the robot. Only obstacles detected within the shaded area of the two mentioned circles, marked with green lines, are taken into account when calculating the repulsive force vector and the new intermediate target point. This means that these obstacles will affect the robot's further movement. After determining a new intermediate target point, the robot moves from its current location to that point employing the fuzzy controller described in the previous section. Then, in relation to the new location of the robot, the relevant obstacles that have an influence on the robot's movements are determined using the concept of the variable target and robot perception circles. For each of these obstacles, the repulsive force vector is calculated and than the total repulsive vector is calculated

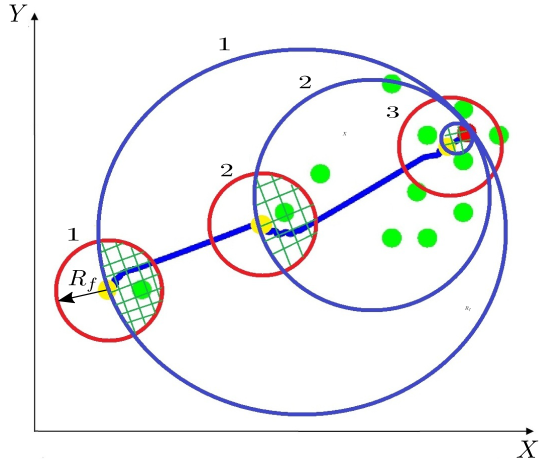

Fig. 8. Variable target (blue) and robot perception (red) circles

and added to the attractive force vector between the final target point and the current location of the robot, thus determining a new intermediate target point. The procedure is repeated until the robot reaches the final target point.

The pseudocode describing the determination of relevant obstacles for calculating a new intermediate target point using the VTPC planner which exploits the concept of attractive and repulsive forces is illustrated in Algorithm 1.

V. SIMULATION RESULTS

To evaluate the effectiveness of the proposed motion control and path planning system, comprehensive simulation studies are conducted under realistic scenarios in MATLAB/Simulink. The results are reported for two different robot environments: unknown static environment and dynamic environment with moving obstacles of the unknown dynamics. The used kinematic model of the Pioneer 3-DX mobile robot is taken from [22], where the robot is represented as a single point in 2D space, while all sensors are located in this point with different orientations. In simulations, the robot's paths will be plotted as the curved green line, where the start and target positions are represented by the orange and the green circles, respectively. Static obstacles and walls are shown as black colored rectangles and lines, while moving obstacle trajectories are represented by curved red lines. For both type of environments, locations of the obstacles are unknown and ranges between the robot and obstacles are measured by simulated on-board sensors (sonars). Only given information are the start and target position and the maximal velocity of dynamic obstacles.

A. Unknown Static Environment

The results obtained for partially unknown environments are presented in Figs. 9 and 10. These figures show the results for narrow environment and cluttered environment, respectively. As the robot moves through the environment, it detects obstacles using ultrasonic sensors. The robot uses a circuit of 16 sensors and each of them can detects one point on the obstacle, where the sensor reading range is

Algorithm 1 VTPC-based planner

Input: Robot starting position (x_s, y_s), Target point (x_t, y_t)
Output: Robot trajectory
Current robot position $(x, y) = (x_s, y_s)$
for $(x, y) \neq (x_t, y_t)$ **do**
  Draw a robot perception circle around the robot with radius R_f
  Draw a target circle with diameter R_a equal to distance between the target point and current robot position
  Determine the area of intersection between the circles
  Perform sensors readings and determine distances between the current robot location to the obstacles within the intersection area (R_{ob_i})
  Calculate repulsive forces for obstacles within the cross-section of circles and their sum F_r
  Calculate attractive force F_a
  Sum the attractive and repulsive forces $F = F_a + F_r$
  Calculate intermediate target point (x_n, y_n) based on vector summation as a new robot way-point
  if Robot did not reach a new way-point (x_n, y_n) **then**
    Drive the robot to a new way point using the proposed fuzzy controller
  else
    Robot reached a new way-point (x_n, y_n)
  end if
  $(x, y) = (x_n, y_n)$
end for
return Robot trajectory

limited to 4 [m]. The detected obstacle points are used in the VTPC motion planning algorithm described earlier to determine an intermediate target point or a final target point. In both scenarios the robot successfully reached the target configuration and avoids obstacles on the path to it.

B. Partially Unknown Dynamic Environment

Further tests were conducted to verify the effectiveness of the proposed approach for avoiding both stationary and moving obstacles and target reaching within partially unknown environment. The scenario with two dynamic obstacles is given in Fig. 11. It is important to note that the starting positions of moving obstacles as well as their trajectories are randomly generated. The robot successfully moves from the start to the target position with simultaneous avoidance of the moving obstacle with unknown dynamics. Results obtained for a more complex scenario with three moving obstacles with unknown motion is shown in Fig. 12.

In the next scenario two robots move toward their targets whereas they paths may cross each other, where a motion of one robot is unknown for another. The movement of the another mobile robot is presented by a path in blue color. Both robots successfully move to their targets while avoiding each other, which is evident in Fig. 13.

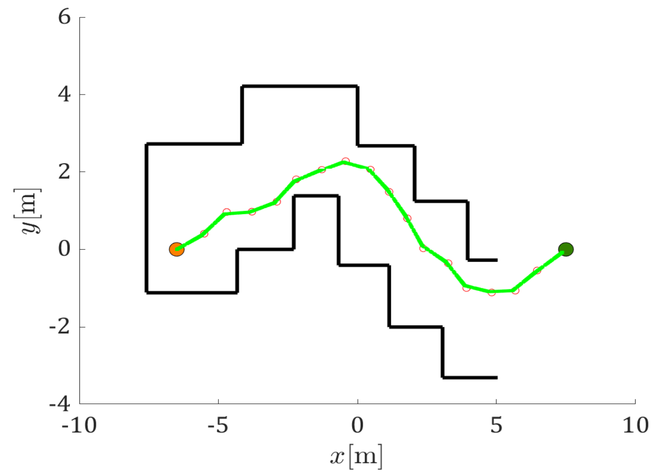

Fig. 9. Path planning and path execution for the narrow environment

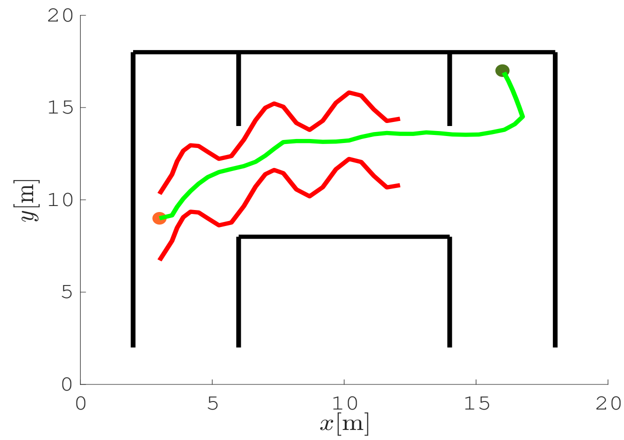

Fig. 11. Path planning and collision avoidance with static and two dynamic obstacles with unknown dynamics

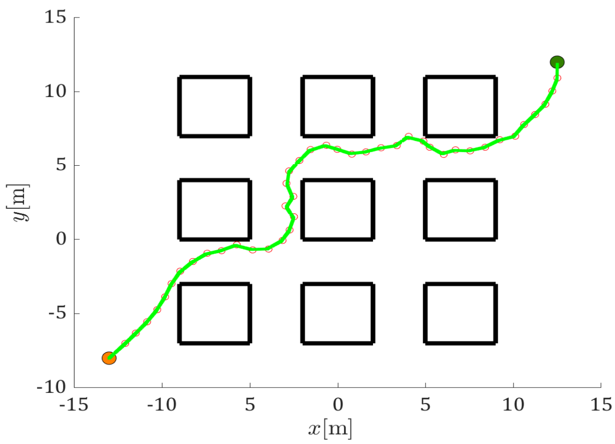

Fig. 10. Path planning and path execution for the cluttered environment

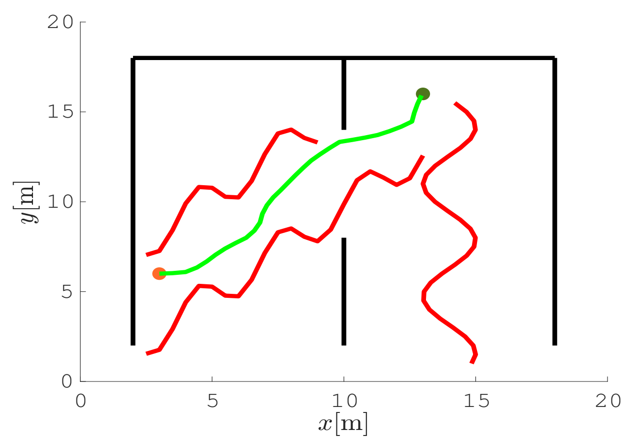

Fig. 12. Variable target (blue) and robot perception (red) circles

TABLE III
REAL-TIME CONSTRAINTS ANALYSIS

Figure	Path length [m]	Computational time [s]	Pulse time [ms]
9	21.971	0.317	8.241
10	32.435	0.4354	12.293
11	16.1	0.317	10.571
12	11.2	0.215	8.716

C. Real-Time Constraints Analysis

In order to demonstrate that the proposed approach satisfies real-time constraints, the following parameters will be considered for Figs. 9-12 the path length, the computational time and the pulse time. The path length represents a total length of the robot's path from the start to the target position. The computational time is the time needed for sonars data fusion, grid map building and path planning, while the pulse time refers to average computational time per iteration. The obtained values for these parameters are listed in Table III.

Furthermore, the efficiency of the proposed approach compared to other competing methods, such as D* [24], Dynamic Window Approach (DWA) [25] and Particle Swarm

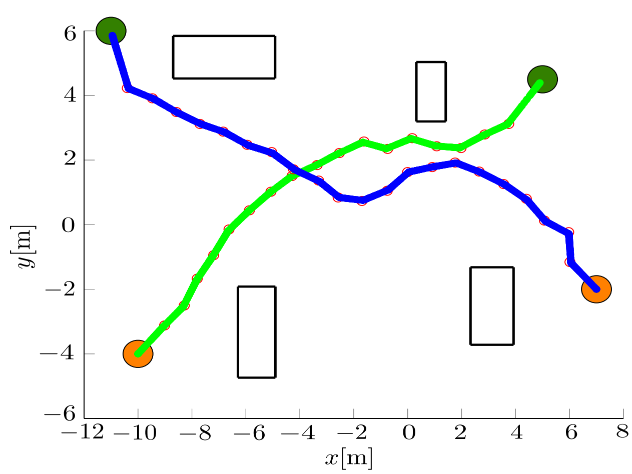

Fig. 13. Mobile robots motions with paths crossing

TABLE IV

COMPARISON RESULTS WITH OTHER APPROACHES

Method	Path length [m]	CT [s]	PT [ms]
VTPC	22.1782	0.4832	14.3792
PSO based	22.4176	0.4719	14.4960
D*	23.3864	0.4680	15.1724
DWA	23.2650	0.4498	14.2307

Optimization (PSO) [26] will be assessed for the mobile robot path planning in a dynamic environment. In this scenario two robots move towards different targets and intersect each other's paths (Fig. 13). The results obtained indicate that the proposed approach generates the shortest path with satisfactory computation time, while the smallest computation time is produced by DWA method (Table IV).

Notably, simulation results confirmed that the proposed method is cost efficient for the real-time applications in terms of the computational time and generated path length. This is particularly evident in the case of more complex environments with unknown static and dynamic obstacles, where the movement of dynamic obstacles are also unknown. The main reason for this is to speed up the motion planning process by reducing the number of obstacles using proposed VTPC algorithm that are taken into account for calculating the vectors of repulsive forces.

VI. CONCLUSIONS

This paper presents a dual-level control concept for mobile robots that solves the problem of trajectory tracking and reaching a target point in an unknown environments. For obstacle avoidance and movement towards the target point, an approach based on the vector summation of attractive and repulsive forces generated by the target point and obstacles was used. To speed up the this process we reduce the number of obstacles that affect the robot motion to only those located in the cross section of the target circle and the robot perception circle (VTPC). The designed low-level position controller was based on the fuzzy logic, which generates the control inputs to drive the mobile robot. The effectiveness and performance of the proposed system were verified through simulations in both unknown static and dynamic environments.

ACKNOWLEDGMENT

This work has been supported in part by the scientific project "Strengthening Research and Innovation Excellence in Autonomous Aerial Systems - AeroSTREAM," supported by the European Commission HORIZON WIDERA-2021-ACCESS-05 Programme through the project under G.A. number 101071270.

REFERENCES

[1] O. Arslan and D.E. Koditschek, Sensor-based reactive navigation in unknown convex sphere worlds, International Journal of Robotics Research, vol. 38, pp. 196-223, 2019.

[2] K. Bektas and H.I Bozma, Apf-rl: safe mapless navigation in unknown environments, In Proc. International Conference on Robotics and Automation (ICRA), 2022, pp. 7299-7305.

[3] S. Petty and T. Fraichard, Safe motion planning in dynamic environments, In Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, 2005, pp. 3726-3731.

[4] D. Cui and H. Li, Model predictive control of nonholonomic mobile robots with backward motion, IFAC PapersOnLine, vol. 52-24, pp. 195-200, 2019.

[5] D. Wang, W. Wei, Y. Yeboah, Y. Li and Y. Gao, A robust model predictive control strategy for trajectory tracking of omni-directional mobile robots, J. Intell. Robot. Syst., vol. 98, pp. 439-453, 2020.

[6] M. Rosenfelder, H. Ebel and P. Eberhard, Cooperative distributed nonlinear model predictive control of a formation of differentially-driven mobile robots, Robot. Auton. Syst., vol 150, 2022.

[7] M. Cui, H. Liu, X. Wang and W. Liu, Adaptive control for simultaneous tracking and stabilization of wheeled mobile robot with uncertainties, Jour. of Intel.and Rob. Systems, vol. 108, paper 46, 2023.

[8] H.P. Liu, D.S. Wang and Y.X. Wang, Research of path planning for mobile robots based on fuzzy artificial potential field method, Control Engineering of China, vol. 29, pp. 33-38, 2022.

[9] B. Lacevic and J. Velagic, Evolutionary design of fuzzy logic based position controller for mobile robot, Journal of Intelligent and Robotic Systems, vol. 63, pp. 595-614, 2011.

[10] J. Velagic, N. Osmic and B. Lacevic, Neural network controller for mobile robot motion control, Proc. Wor. Acad. Sci. E., vol. 47, pp. 193-198, 2008.

[11] Z. Chen, Y. Liu, W. He, H. Qiao and H. Ji, Adaptive-neural-network-based trajectory tracking control for a nonholonomic wheeled mobile robot with velocity constraints, IEEE Transactions on Industrial Electronics, vol. 68, pp. 5057-5067, 2021.

[12] N. Hassan and A. Saleem, Neural network-based adaptive controller for trajectory tracking of wheeled mobile robots, IEEE Access, vol. 10, pp. 13582-13597, 2022.

[13] H. Rios, M. Mera and A. Polyakov, Perturbed unicycle mobile robots: a second-order sliding-mode trajectory tracking control, IEEE Transactions on Industrial Electronics, vol. 71, pp. 1-9, 2023.

[14] M. Mera, H. R?os and E.A. Mart?nez, A sliding-mode based controller for trajectory tracking of perturbed unicycle mobile robots, Control Engineering Practice, vol. 102, 2020.

[15] Liu, Zhen-yu, et al., Trajectory tracking control of wheeled mobile robots based on the artificial potential field, In Proc. 4th International Conference on Natural Computation. Vol. 7, 2008.

[16] Q. Wu and X. Wang, A virtual reference trajectory scheme for tracking control of wheeled mobile robots with slip disturbances, Transactions of the Institute of Measurement and Control, vol. 47, pp. 75-83, 2024.

[17] E.J. Hwang, H.S. Kang, C.H. Hyun, C.H. Hyun and M. Park, Robust backstepping control based on a Lyapunov redesign for skid-steered wheeled mobile robots, International Journal of Advanced Robotic Systems, vol. 10, 2013.

[18] J.C. Latombe, Robot motion planning, Springer Science and Business Media, vol. 124, 2012.

[19] J. Yu, T. Nguyen and T.Dao, Global path planning algorithm for mobile robots: a review, Journal of Network Intelligence, vol. 9, pp. 643-657, 2023.

[20] O. Khatib, Real-time obstacle avoidance for manipulators and mobile robots, In Proc. IEEE International Conference on Robotics and Automation, 1985, pp. 500-505.

[21] J. Borenstein and Y. Koren, The vector field histogram-fast obstacle avoidance for mobile robots, IEEE transactions on robotics and automation, vol. 3, pp. 278-288, 1991.

[22] D. Osmankovic and J. Velagic, Gradient based adaptive trajectory tracking control for mobile robots, In Proc. 24th International Conference on Information, Communication and Automation Technologies, Sarajevo, 2013, pp. 1-6.

[23] J. Velagic, B. Lacevic and N. Osmic, Efficient path planning algorithm for mobile robot navigation with a local minima problem solving, In Proc. IEEE Inter. Conf. on Ind. Technology, 2006, pp. 2325-2330.

[24] M. Likhachev, D. Ferguson, G. Gordon, A. Stentz and S. Thrun, Anytime dynamic A*: An anytime, replanning algorithm, In Proc. International Conference on Aut. Planning and Scheduling, 2005.

[25] D. Fox, W. Burgard and S. Thrun, The dynamic window approach to collision avoidance, IEEE Robotics and Automation Magazine, vol. 4, pp. 23-33, 2002.

[26] I. Jabandzic and J. Velagic, Particle swarm optimization-based method for navigation of mobile robot in unstructured static and time-varying environments, In Proc. 3rd Conference on Control and Fault-Tolerant Systems, 2016, pp. 59-66.