Autonomous mobile robot navigation coupling fuzzy logic and reactive DVZ 3D obstacle avoidance control

Emna Baklouti, Mohamed Jallouli, Nader Ben Amor, Sondes Titi, Ahlem Nafti
École Nationale d'Ingénieur de Sfax ENIS
Université de Sfax, Tunisie
Computer & Embedded Systems Laboratory (CES)
Email: baklouti.emna@gmail.com

***Abstract*—This paper deals with the reactive control of an autonomous mobile robot which should move safely in a unknown three-dimensional(3D) environment to reach a desired goal. In order to proceed with the problem, we propose a hybrid approach based on fuzzy logic controller (FLC) for goal seeking behavior and Deformable Virtual Zone (DVZ) reactive algorithm to address 3D obstacles avoidance. We suggest a practical solution to combine both controllers. The proposed approach has been successfully tested in different configuration on simulation**

***Keywords*—Autonomous robot, Reactive control, FLC, Obstacle avoidance, DVZ, 3D obstacle.**

I. Introduction

Robot navigation, coupled with obstacle avoidance control, remains one of the major issues in the field of mobile robotics [1]. While Reliable and efficient obstacle detection is a fundamental requirement to attain an autonomous navigation, this paper addresses the problem of the autonomous navigation with reactive behaviors of wheeled robots evolving in 3D dynamic and unknown environment.

A successful way of structuring the navigation task when the environment of the robot is obstacle free is within behavior based navigation approaches. Many works have used fuzzy logic systems for mobile robot navigation[2], [3]. But as the environment becomes complex, motion planning needs much more treatments to allow the robot to move between its current position and final destination without any collision within the surrounding environment. Thus, Many researches have turned their attention to the obstacle avoidance problem. They developed interesting real-time methods and algorithms for mobile robot navigation in unknown environments. Existing algorithms that deal with 3D workspace are inherited from algorithms for 2D workspace [4], [5], but the traditional 2D autonomous navigation techniques have often proven insufficient for safe navigation through complex 3D environments. Notably, they lack an ability to drive under obstacles if there is enough height, or refusing to drive under obstacles that are too low to the ground.

Focusing in obstacle avoidance techniques in 2D workspaces, a wide variety of algorithms have been proposed [6]. More precisely, references [7], [8], [9] present a system for automated path finding in static environments. The system uses an off-line compilation process to derive a cognitive model of the environment which is suitable for efficient navigation. Yet, the system does not deal with dynamic obstacles in real-time.

Other approaches directly calculate obstacle avoidance commands from the information available on the environment, called reactive approach. Reactive approaches calculate the instantaneous control to be applied to the actuators in each time step, based on environmental information provided by the sensors of the system. The Vector Field Histogram (VFH) approach, first suggested by Borenstein, generates a polar histogram of the space occupancy in the vicinity of a robot [10]. Besides, using the Bugs algorithm [11], the robot moves directly toward the goal unless faced by an obstacle, in which case the obstacle would be contoured until motion to the goal is again possible. In the Potential Field method introduced by Khatib, each obstacle is modeled with a built-in repulsive force that prevents the robot from bumping into the obstacle [12]. These algorithms are set up to build a cognitive model for path finding from sensory information without prior acquaintance with the environment. But they are not always sure to yield a navigable path without undesired local minima.

Another approach, based on a reflex behavior reaction [13], consists in using the DVZ concept. The main idea is to set the robot/environment interaction as a DVZ surrounding the robot. The deformation of this zone is due to the intrusion of proximity information. The system reaction is made in order to reform the risk zone to its nominal shape. The main characteristic of this approach is that there is no geometrical modeling of the obstacles, but rather a vector representation of the interaction between the robot and its environment [14]. On the other hand, experiments carried out on a real robot with sensors have shown that there is never a real symmetry in the deformation of the DVZ such that it couldnt fall into local minima[15].Also the DVZ principle can be easily reformulated for 3D detection [15]. In the present paper we develop a navigation system that is able to drive a mobile robot from point to point without human intervention. For goal reaching we develop a FLC taking into account robot kinematic constraints. For obstacle detection and avoidance we are interested in having a 3D reactive term (deformation of the risk zone) that addresses the problem of 3D obstacles by reformulating the DVZ algorithm. Finally we utilise this

978-1-4673-7751-5/15/$31.00 ©2015 IEEE

reactive term to combine both controllers (DVZ and FLC).

The paper is organized as follows: sections 2 and 3 review the Fuzzy-reformulated DVZ algorithms we will use for navigation and 3D obstacle avoidance. Section 4 outlines a practical solution to combine both controllers while section 5 reports on the results we have obtained. The last section comes in the form of a general conclusion .

II. Fuzzy logic navigation controller

This section reviews a solution for the problem of mobile robot navigation to arrive at a desired target in an environment without obstacles. We use the most common strategies for the mobile robot navigation which are FLCs [16]. FLC generate velocities to be applied on the mobile base wheels during the transition from the initial position toward another desired position, taking into account robot kinematic constraints. The kinematic model is given by (1):

$$\begin{cases} \dot{x}_R &= \frac{V_R+V_L}{2}\cos\alpha_R \\ \dot{y}_R &= \frac{V_R+V_L}{2}\sin\alpha_R \\ \dot{\alpha}_R &= \frac{V_R-V_L}{L} \end{cases} \quad (1)$$

Where V_R and V_L are respectively the robot's wheels right and left velocities, $dot\alpha_R$ is the robot's angular velocity, L is the distance between the two wheels and α_R is the angle between the robot's direction and the X-axis as shown in Fig. 1.
By approximating the derivatives in (1) using Euler's method, it becomes (2) , where T is the sampling time

$$\begin{cases} x_R^{new} = x_R^{old} + T\frac{V_R^{old}+V_L^{old}}{2}\cos\alpha_R^{old} \\ y_R^{new} = y_R^{old} + T\frac{V_R^{old}+V_L^{old}}{2}\sin\alpha_R^{old} \\ \alpha_R^{new} = \alpha_R^{old} + T\frac{V_R^{old}-V_L^{old}}{L} \end{cases} \quad (2)$$

In the current work, the fuzzy controller has two inputs: the distance d and the angle ϕ. It has also two outputs: the right velocity V_R and the left velocity V_L. In Fig. 1 d is the distance between the center of the robot and its target; and ϕ is the difference between the angle of the robot's direction and the angle $\theta_T$of the connecting line between the center of the robot and its target. The distance d and the angle ϕ are expressed according to (3) and (4)

$$d = \sqrt{(x_T - x_R)^2 - (y_T - y_R)^2} \quad (3)$$

$$\varphi = \theta_T - \alpha_R \quad (4)$$

With

$$\theta_T = \tan^{-1}\frac{y_T - y_R}{x_T - x_R} \quad (5)$$

Five fuzzy subsets are assigned to the variable distance d: VS: Very Small; S: Small; M: Medium; L: Large; and VL: Very Large. Membership functions for distance (d) are outlined in Fig. 2.

Seven fuzzy subsets have been associated to the angle ϕ: NL: Negative Large; NM: Negative Medium; NS: Negative Small; Z: Zero; PS: Positive Small; PM: Positive Medium;

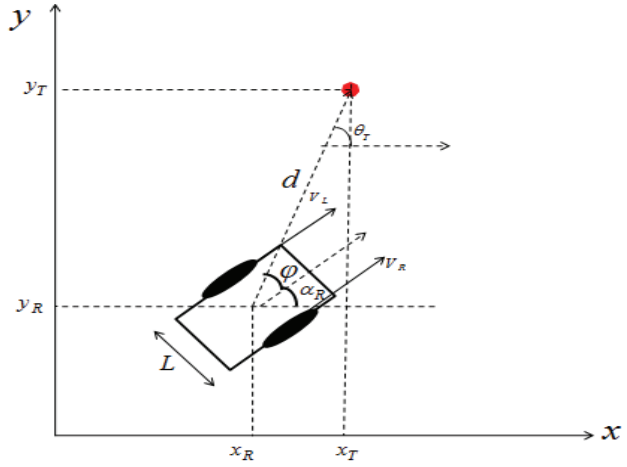

Fig. 1: Robot configuration according to the objective

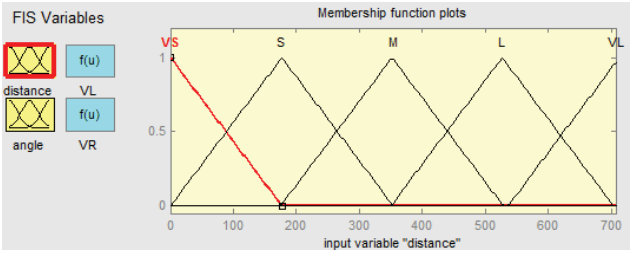

Fig. 2: Membership functions for the distance d

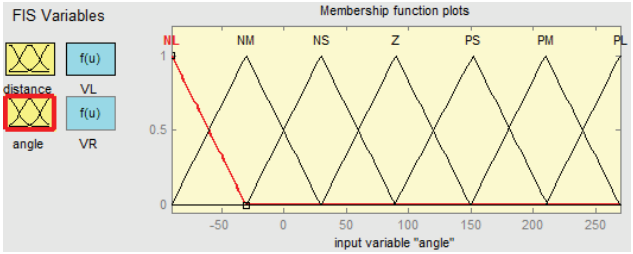

Fig. 3: Membership functions for the angle ϕ

and PL: Positive Large. Membership functions for the angle ϕ are reported in Fig. 3. With each variable input value combination, an action of the output variables is associated. Fuzzy rules (situation/action) are proposed in Table I. These rules are manually built following several simulations. With the notations: V_L: Left Velocity; V_R: Right Velocity; Z: Zero; S: Small; M: medium; B: Big and VB: Very Big.

TABLE I: Linguistic inference table

V_g & V_d		NL		NM		NS		Z		PS		PM		PL	
		V_g	V_d	V_g	V_d	V_g	V_d	V_g	V_d	V_g	V_d	V_g	V_d	V_g	V_d
d	VS	B	S	M	S	S	Z	Z	Z	Z	S	S	M	S	B
	S	VB	M	B	S	M	S	S	S	S	M	S	B	M	VB
	M	VB	B	VB	M	B	S	M	M	S	B	M	VB	B	VB
	L	VB	B	VB	B	VB	M	B	B	M	VB	B	B	B	VB
	VL	VB	B	VB	B	VB	B	VB	VB	B	VB	B	VB	B	VB

III. DVZ REFORMULATION FOR 3D OBSTACLE AVOIDANCE

In current DVZ literature, the risk zone is based on the definition of an ellipse surrounding the robot. In the case of 3D detection, this concept can be applied by defining an ellipsoid as a virtual envelope that protects the robot.

A. 3D undeformed DVZ $(d_h(\theta,\phi)$:

In this section, we suggest an analytical expression for the spherical signature of the 3D undeformed DVZ. Mathematically, an ellipsoid shape is defined by three axes $c_x, c_y and c_z$. We assume that the proper reference frame of the DVZ is translated from the center of the robot frame (R) by a vector $[a_x a_y a_z]$.Thus, the Cartesian equation of the ellipsoid is (6):

$$\frac{(x-a_x)^2}{c_x^2}+\frac{(y-a_y)^2}{c_y^2}+\frac{(z-a_z)^2}{c_z^2}=1 \tag{6}$$

The 3D undeformed DVZ moves with the robot. Therefore, the coefficients $a_x, a_y, a_z, c_x, c_y and c_z$ are chosen in a way that they depend on the translational velocity of the robot . Thus, we have chosen:

$$\begin{cases} c_x = \lambda_{cx}V^2 + c_{xmin} \\ c_y = \frac{\sqrt{5}}{3}c_x \\ c_z = c_y \\ a_x = -\frac{2}{3}c_x \\ a_y = a_z = 0 \end{cases} \tag{7}$$

Further-more, Considering that the DVZ is rigidly attached to the robot, oriented in the main direction of the vehicle movement, the resulting DVZ need to be rotated around the z-axis with this angle of the robot.

$$\begin{cases} x = xcos\alpha_R + ysin\alpha_R \\ y = ycos\alpha_R - xsin\alpha_R \\ z = z \end{cases} \tag{8}$$

Considering the spherical coordinates $(d_h(\theta,\phi),\theta,\phi)$ of a point P, the Cartesian coordinate system (x,y,z), are defined by (9)

$$\begin{cases} x = d_h(\theta,\phi)cos\theta cos\phi \\ y = d_h(\theta,\phi)sin\theta cos\phi \\ z = d_h(\theta,\phi)sin\phi \end{cases} \tag{9}$$

with $d_h(\theta,\phi)$ is the distance between the center of the robot and the undeformed DVZ, θ refers to the longitude direction in the DVZ and ϕ refers to the latitude direction as shown in Fig. 4. Substituting equation (8) and equation (9) in equation (6) leads to the quadratic equation:

$$Ad_h(\theta,\phi)^2 + Bd_h(\theta,\phi) + C = 0 \tag{10}$$

with:

$$\begin{cases} A = c_y^2c_z^2cos\alpha_R^2cos\theta^2cos\phi^2 + 2*cos\alpha_R sin\alpha_R cos\phi^2 cos\theta \\ \quad sin\theta*(c_y^2c_z^2 - c_x^2c_z^2) + c_y^2c_z^2sin\alpha_R^2cos\phi^2sin\theta^2 + c_x^2c_z^2 \\ \quad cos\alpha_R^2sin\theta^2cos\phi^2 + c_x^2c_z^2sin\alpha_R^2cos\theta^2cos\phi^2 + c_x^2c_y^2 \\ \quad sin\phi^2 \\ B = -2c_y^2c_z^2a_xcos\alpha_R cos\theta cos\phi - 2c_y^2c_z^2a_x sin\alpha_R sin\theta cos\phi \\ C = a_x^2c_y^2c_z^2 - c_x^2c_y^2c_z^2 \end{cases} \tag{11}$$

the solution of equation (10) is (12)

$$d_h(\theta,\phi) = \frac{-B+\sqrt{B^2-4AC}}{2A} \tag{12}$$

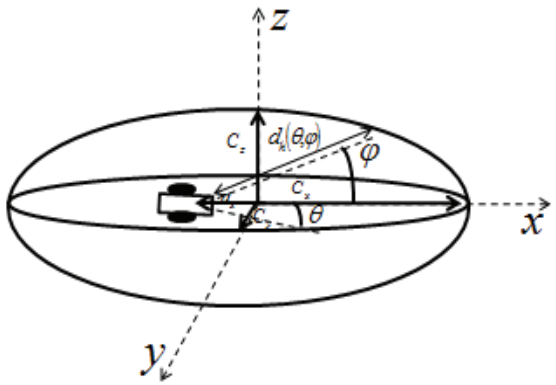

Fig. 4: Undeformed DVZ parametres in 3D

B. 3D deformed DVZ $d(\theta,\phi)$:

The 3D deformed DVZ is acquired from the sensor information. But the meaningful information is restricted to that inside the 3D undeformed DVZ. So, a preliminary test on the sensor information is necessary. If we consider that $c(\theta,\phi)$ is the 3D sensor information and that $d(\theta,\phi)$ is the distance between the sensor and the deformed DVZ (see Fig. 5), then $d(\theta,\phi)$ is obtained by saturation of $c(\theta,\phi)$ according to (13):

$$d(\theta,\phi) = \begin{cases} c(\theta,\phi) & if c(\theta,\phi) \prec d_h(\theta,\phi) \\ d_h(\theta,\phi) & elsewhere \end{cases} \tag{13}$$

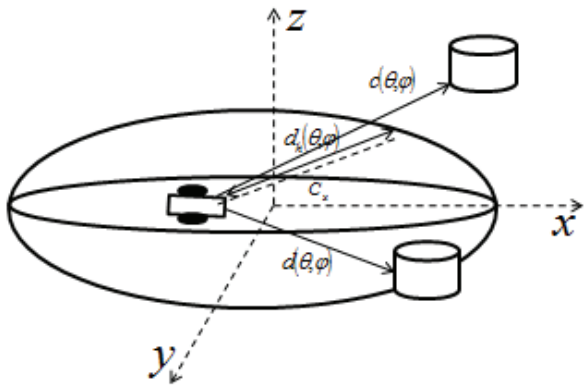

Fig. 5: 3D Deformed DVZ

C. Reactive term (Intrusion):

The choice of the intrusion information expression is instrumental in the control of the robot reactivity. The intrusion ratio is introduced to quantify the DVZ deformation. we introduce the intrusion ratio in (14):

$$I = \int_{\theta=0}^{\pi}\int_{\phi=0}^{\pi}\frac{d_h(\theta,\phi)-d(\theta,\phi)}{d(\theta,\phi)}d\theta d\phi \tag{14}$$

The above equation quantifies the global amount of intrusion, normalized by the factor $d(\theta,\phi)$. Note that a null distance robot/obstacle yields an infinite value of I. Thus, a control strategy that guarantees a bounded intrusion ratio at any time insures that the system avoids any obstacle. The intrusion ratio I is a function of V (through the coefficients $a_x, a_y, a_z, c_x, c_y and c_z$), the orientation of the robot α_R and the absolute position of the detected obstacle $[x_O y_O z_O]$. We can write:

$$I = f(V, \alpha_R, x_o, y_o, z_o) \tag{15}$$

D. Control design for 3D static and/or dynamic environments:

We design this controller in order to minimize the DVZ deformation. Thus, to conceive the speed applied to the wheels of the mobile robot, we choose to satisfy the Lyapunov theorem of stability according to (16).

$$V_I = \frac{I^2}{2} \tag{16}$$

The derivation generates:

$$\dot{V}_I = I(J_I^V \dot{V} + J_I^{\alpha_R}\omega + F^{Rob.vel}V + F^{Obs.vel}) \tag{17}$$

with

$$\begin{cases} J_I^V = \int_{\theta=0}^{\pi}\int_{\phi=0}^{\pi} \frac{1}{d(\theta,\phi)}(\frac{1}{2A}[-J_V^B + \frac{BJ_V^B - 2CJ_V^A - 2AJ_V^C}{\sqrt{(B^2-4AC)}}] \\ \quad -J_V^A\frac{-B+\sqrt{(B^2-4AC)}}{2A^2})d\theta d\phi \\ J_I^{\alpha_R} = \int_{\theta=0}^{\pi}\int_{\phi=0}^{\pi} -\frac{1}{d(\theta,\phi=)}(\frac{1}{2A}[-J_{\alpha_R}^B + \frac{BJ_{\alpha_R}^B - 2CJ_{\alpha_R}^A}{\sqrt{(B^2-4AC)}}] \\ \quad -J_{\alpha_R}^A\frac{-B+\sqrt{(B^2-4AC)}}{2A^2})d\theta d\phi \\ F^{Rob.vel} = \int_{\theta=0}^{\pi}\int_{\phi=0}^{\pi}\frac{d_h(\theta,\phi)}{d^2(\theta,\phi)}\cos(\theta)\cos(\phi)d\theta d\phi \\ F^{Obs.vel} = \int_{\theta=0}^{\pi}\int_{\beta=0}^{\pi}\frac{d_h(\theta,\phi)}{d^2(\theta,\phi)}[\dot{X}\cos(\theta+\alpha_R)\cos(\phi) \\ \quad +\dot{Y}\sin(\theta+\alpha_R)\cos(\phi) + \dot{Z}\sin(\phi)]d\theta d\phi \end{cases} \tag{18}$$

Where

$$\begin{cases} J_V^A = 2*(c_y\frac{dc_y}{dV}c_z^2 + c_z\frac{dc_z}{dV}c_y^2)cos\alpha_R^2 cos\theta^2 cos\phi^2 + \\ \quad 4*cos\alpha_R sin\alpha_R cos\phi^2 cos\theta sin\theta * (c_y\frac{dc_y}{dV}c_z^2 + \\ \quad c_z\frac{dc_z}{dV}c_y^2 - c_x\frac{dc_x}{dV}c_z^2 - c_z\frac{dc_z}{dV}c_x^2) + \\ \quad 2*(c_y\frac{dc_y}{dV}c_z^2 + c_z\frac{dc_z}{dV}c_y^2)sin\alpha_R^2 cos\phi^2 sin\theta^2 + \\ \quad 2*(c_x\frac{dc_x}{dV}c_z^2 + c_z\frac{dc_z}{dV}c_x^2)cos\alpha_R^2 sin\theta^2 cos\phi^2 + \\ \quad 2*(c_x\frac{dc_x}{dV}c_z^2 + c_z\frac{dc_z}{dV}c_x^2)sin\alpha_R^2 cos\theta^2 cos\phi^2 + \\ \quad 2*(c_x\frac{dc_x}{dV}c_y^2 + c_y\frac{dc_y}{dV}c_x^2)sin\phi^2 \\ J_{\alpha_R}^A = -2*c_y^2c_z^2 cos\alpha_R sin\alpha_R cos\theta^2 cos\phi^2 + \\ \quad 2*cos\phi^2 cos\theta sin\theta*(c_y^2c_z^2 - c_x^2c_z^2)* \\ \quad ({cos\alpha_R}^2 - {sin\alpha_R}^2) + \\ \quad 2*c_y^2c_z^2 cos\alpha_R sin\alpha_R sin\theta^2 cos\phi^2 - \\ \quad 2*c_x^2c_z^2 cos\alpha_R sin\alpha_R sin\theta^2 cos\phi^2 + \\ \quad 2*c_x^2c_z^2 cos\alpha_R sin\alpha_R cos\theta^2 cos\phi^2 \end{cases} \tag{19}$$

$$\begin{cases} J_V^B = -2(4*c_y^3\frac{dc_y}{dV}a_x + c_y^4 a_x\frac{da_x}{dV})(cos\alpha_R cos\theta cos\phi + \\ \quad sin\alpha_R sin\theta cos\phi) \\ J_{\alpha_R}^B = -2*c_y^2c_z^2a_x(cos\alpha_R sin\theta cos\phi - sin\alpha_R sin\theta cos\phi) \end{cases} \tag{20}$$

$$\begin{cases} J_V^C = 2*a_x\frac{da_x}{dV}c_y^4 + 4*a_x^2c_y^3\frac{dc_y}{dV} - 2*c_x\frac{dc_x}{dV}c_y^4 - 4*c_x^2 \\ \quad c_y^3\frac{dc_y}{dV} \end{cases} \tag{21}$$

and $\dot{X}, \dot{Y}, \dot{Z}$ define the obstacle's absolute velocity. Hence, we come out with the robot control system which is expressed through the following equations systems (22) and (23), where $K_V and K_{r_1}$ are arbitrary positive gains yielding $\dot{V}_I \leq 0 \ \forall\ t$. In the case of static obstacles:

$$\begin{cases} \dot{V} = -K_V J_I^V I - \frac{F^{Rob.vel}}{J_I^V}V \\ \omega = -K_{r_1} J_I^{\alpha_R} I \end{cases} \tag{22}$$

In the case of dynamic obstacles:

$$\begin{cases} \dot{V} = -K_V J_I^V I - \frac{F^{Rob.vel}}{J_I^V}V - \frac{F^{Obs.vel}}{J_I^V} \\ \omega = -K_{r_1} J_I^{\alpha_R} I \end{cases} \tag{23}$$

IV. Combining FLC and DVZ

The generic control system of a mobile robot must be able to: i) determine the intentions of the user to reach the target ii) generate an optimal trajectory without risk, and, when necessary, (i.e. in the presence of obstacles) iii) adjust the control signals in order to reach the destination safely. Therefore,the coupling of the two algorithms (FLC and DVZ) must meet two requirements. The first consists in attempting to reach the desired target using the fuzzy controller, the second occurs when risk zone is deformed , in which case the DVZ controller must steer the robot away from the obstacle.

We propose a hysteresis solution that satisfies both these conditions. It is the amount of the DVZ reactive term that allows the system to make decision to keep FLC control or switch to DVZ control. This solution comes in the form of applying threshold values $I_{FLCtoDVZ}$ and $I_{DVZtoFLC}$ for allowed intrusion. As a result two cases could occur as pointed out in Fig. 6 below:

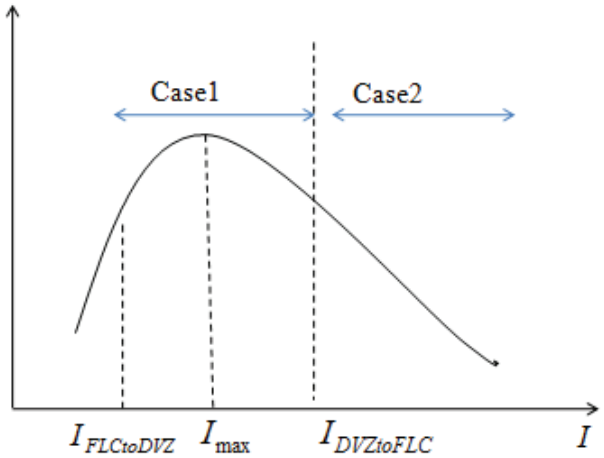

Fig. 6: evolution of intrusion

In the first case (case 1 in Fig. 6), the robot is intended to reach a target while the distance to an obstacle decreases, i.e. the intrusion increase from $I_{FLCtoDVZ}$ to I_{max}. Consequently, the controller switches to DVZ algorithm and persists in this situation as $I \succ I_{DVZtoFLC}$.

In the second case (case 2 in Fig. 6), the robot is trying to avoid an obstacle; thus, the deformation decreases from I_{max} to $I_{DVZtoFLC}$. In this case, the controller switches to fuzzy navigation only when $I \prec I_{DVZtoFLC}$ in order to eliminate the transition between the two controllers (overlap zone).

V. Results

To test the robustness of our control approach we simulate mobile robot khepera II behavior in a virtual environment . It is a circular robot of 55 mm in diameter. It has two driving wheels, which can be independently controlled. The distance between the two wheels (L) is 53 mm. We start with obstacle free environment. Table II provides several configuration in which we diversify the robot start position, target position and the robot angle. Fig. 7 shows the path which has been made by the robot in these different case with FLC. To combine

TABLE II: Configurations for FLC test

configuration	Start(mm)	Target(mm)	Robot angle()
Case1(a)	(-200,0,0)	(200,0,0)	0
Case2(b)	(-200,0,0)	(200,0,0)	-180
Case3(c)	(150,100,0)	(-200,0,0)	-90
Case4(d)	(50,220,0)	(0,-200,0)	45

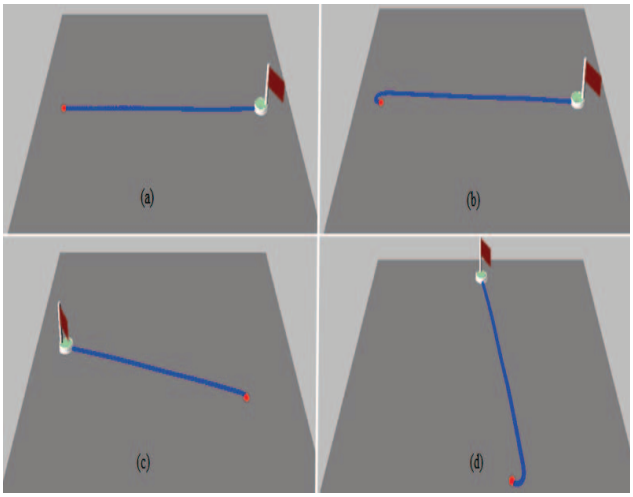

Fig. 7: Robot's path with FLC

the FLC with DVZ, the control parameters already defined in section 3 for DVZ 3D obstacle avoidance are chosen according to the Table III. We modeling a virtual environment with

TABLE III: DVZ Control Parameters

$\lambda_{cx} = 0.0096$	$c_{xmin} = 100$	$V = 25$

several 3D obstacle. Fig. 8 shows the path which has been made by the mobile robot without (entirely blue) and with obstacle (red shape) avoidance (blue and green) to reach the desired target (red flag).

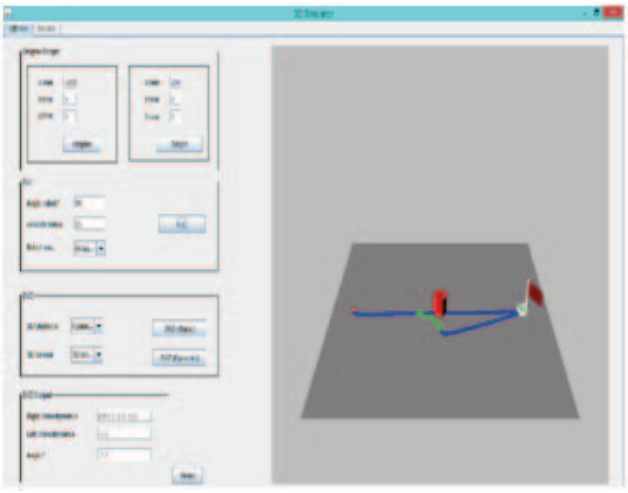

Fig. 8: Coupling FLC (blue) and DVZ (Green) control

Fig. 9 and Fig. 10 indicates other situations, in which we see the controlled robot path in the presence of many obstacles or in a hallway situation. The DVZ controller allows

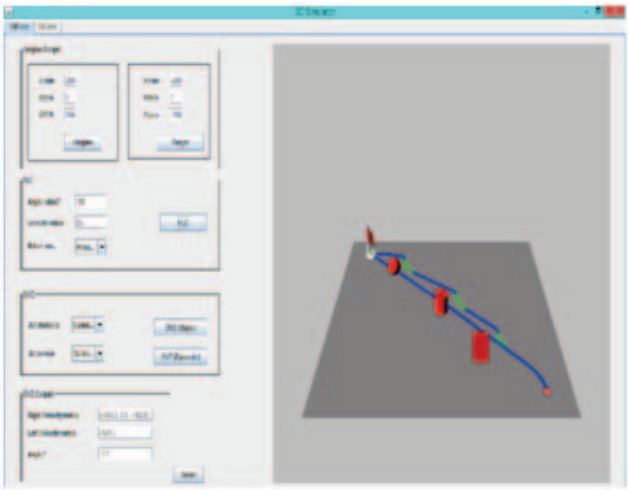

Fig. 9: DVZ obstacle avoidance

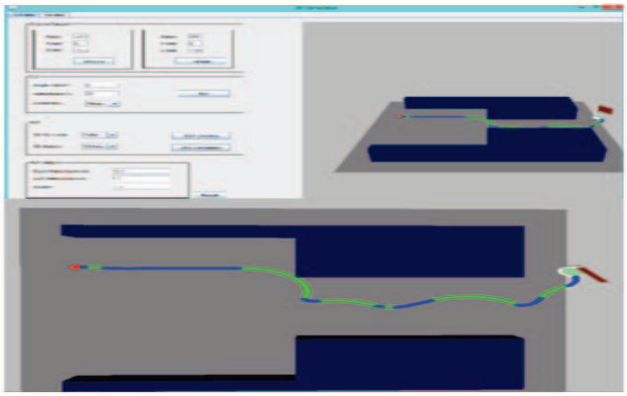

Fig. 10: DVZ configuration for hallway situation

dynamic obstacle avoidance across the term $F^{Obs.vel}$. The reactive system for unpredictable dynamic obstacles is drawn in Fig. 11.

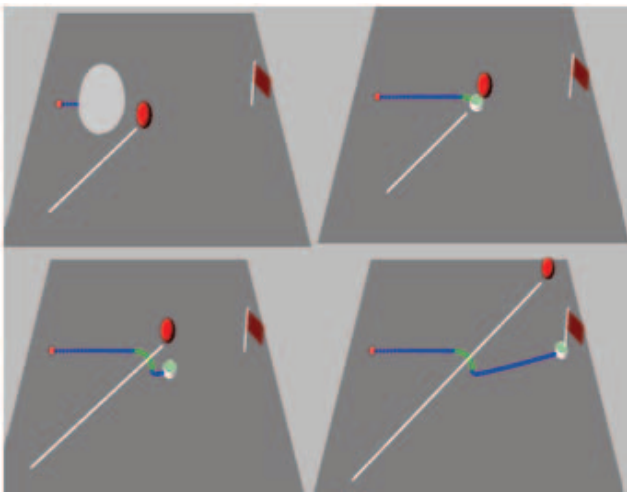

Fig. 11: DVZ configuration for dynamic obstacles

For obstacles with complex shape, the effectiveness of our algorithm against complex 3D obstacles can only be proven if the third dimension Z is considered. Fig. 12 shows the simulation of such situations.In Fig. 12.a the robot could navigate under case when $c(\theta, \phi) \succ d_h(\theta, \phi)$. In Fig. 12.b the table is not sufficiently high to allow the robot to navigate under it case when $c(\theta, \phi) \prec d_h(\theta, \phi)$.

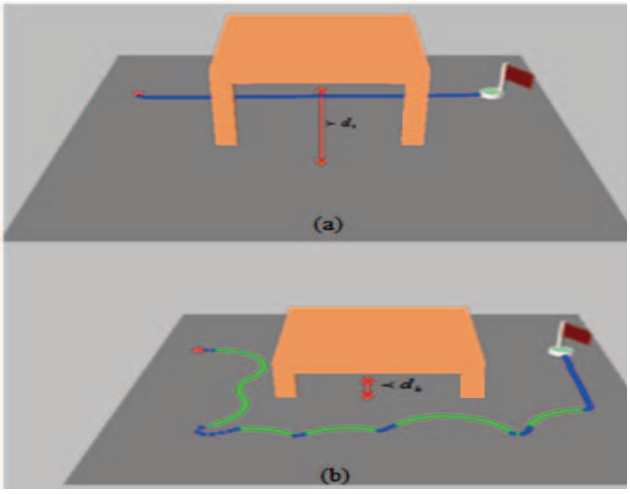

Fig. 12: (a) DVZ configuration for higher table. (b) DVZ configuration for lower table

VI. Conclusion

We have designed the control of mobile robot navigation and obstacle avoidance in 3D virtual environments, based on the extension of the DVZ concept coupled with an FLC. This new formulation allows the robot soft navigation even with complex shape or dynamic obstacles. As you can see, collision avoidance algorithms are based at least on a measure of the distances between robot and obstacles. There is a wide variety of obstacle perception sensors. The future work will be aimed at exploring 3D sensors that accurate 3D measurements with a low-cost.

References

[1] R. S. Ortigoza, C. M. Sanchez, F. C. Corral, V. M. Hernandez-Guzman, J. R. Garca-Snchez, H. Taud, M. Marciano- Melchor, and J. A. Alvarez-Cedillo, *Obstacle Avoidance Task for a Wheeled Mobile Robot A Matlab-Simulink-Based Didactic Application*, MATLAB Applications for the Practical Engineer, Mr Kelly Bennett (Ed.), ISBN: 978-953-51-1719-3, InTech,2014.

[2] V. Yerubandi , Y.M. Reddy and M.V. Kumar Reddy, *Navigation system for an autonomous robot using fuzzy logic*, International Journal of Scientific and Research Publications, Volume 5, Issue 2, February 2015.

[3] A. Jasmine Xavier and R. Shantha Selvakumari, *Behavior architecture controller for an autonomous robot navigation in an unknown environment to perform a given task*, International Journal of Physical Sciences 10, no. 5, pp. 182-191, 2015.

[4] D. Vikerimark, J. Minguez, *Reactive Obstacle Avoidance for Mobile Robots that Operate in Conned 3D Workspaces*, Electrotechnical Conference, IEEE Mediterranean, ISBN: 1-4244-0087-2, Pages 1246 - 1251, 2006.

[5] A. Aalbers, *Obstacle avoidance using limit cycles*, Master thesis, Faculty of Mechanical, Maritime and Materials Engineering, Department Delft Center for Systems and Control, 2013-09-24

[6] M. Khansari, and A. Billard, *A Dynamical System Approach to Realtime Obstacle Avoidance*, Autonomous Robots DOI 10.1007/s10514-012-9287-y Received: 1 May 2011 / Accepted: 8 March 2012.

[7] J.M.P. van Waveren, and L.J.M. Rothkrantz, *Automated path and route finding through arbitrary complex 3D polygonal worlds*, Elsevier, Robotics and Autonomous Systems, February 2006.

[8] R. Axelrod, *Navigation Graph Generation*, AI Game Programming Wisdom 4, Charles River Media, 2008.

[9] J.W. Ratcliff, *Automatic Path Node Generation for Arbitrary 3D Environments*, AI Game Programming Wisdom 4, Charles River Media, 2008.

[10] A.Babineca, M.Dekana, F. Duchoa, A.Vitkoa, *Modifications of VFH navigation methods for mobile robots*, Elsevier, Modelling of Mechanical and Mechatronics Systems, Procedia Engineering, Volume 48, Pages 10 14, 2012.

[11] M.Zohaib, S.M.Pasha, N.Javaid, J.Iqbal, *Intelligent Bug Algorithm (IBA): A Novel Strategy to Navigate Mobile Robots Autonomously*, arXiv ID: 1312.4552, December 2013.

[12] H.Seki, Y.Kamiya, and M.Hikizu, *Real-Time Obstacle Avoidance Using Potential Field for a Nonholonomic Vehicle*, Factory Automation, Javier Silvestre-Blanes (Ed.), ISBN: 978-953-307-024- 7, InTech, 2010.

[13] A. Cacitti, and R. Zapata, *Reactive Behaviours of Mobile Manipulators Based on the DVZ Approach*, IEEE International Conference on Robotics and Automation, Seoul, Korea, pp. 680685, May 21-26, 2001.

[14] L. Lapierre, R. Zapata, and P. Lepinay, *Simultaneous Path Following and Obstacle Avoidance Control of a Unicycle-type Robot*, The International Journal of Robotics Research, vol. 26(361), pp. 261375, 2008.

[15] R. Zapata, P. Lepinay, *Flying among obstacles*, IEEE, ISBN: 0-7803-5672-1, 1999

[16] T.S. Hong, D. Nakhaeinia, and B. Karasfi, *Application of Fuzzy Logic in Mobile Robot Navigation*, Fuzzy Logic - Controls, Concepts, Theories and Applications, Prof. Elmer Dadios (Ed.), ISBN: 978-953-51-0396-7, InTech, 2012.