The International Journal of Engineering and Science (IJES)
|| Volume || 7 || Issue || 5 Ver. III || Pages || PP 11-24 || 2018 ||
ISSN (e): 2319 – 1813 ISSN (p): 23-19 – 1805

An Unicycle Mobile Robot Employing DuckStrategy For Navigation

Bahri Radwen[1], Boucetta Rahma[2], Bel Hadj Ali Saloua[3]
[1]*Research Laboratory: Modeling, Analysis and Control of Systems(MACS) LR16ES22,*
National Engineering School of Gabes,Street of Medenine, 6029 Gabes, University of Gabes, (Tunisia)
Email: bahriradwen16@gmail.com
[2] *Sciences Faculty of Sciences, University of Sfax, Soukra Avenue, 3000 Sfax,(Tunisia)*
Email: boucetta.rahma@gmail.com
[3] *Preparatory Institute for Engineering Studies, El Manar University, El Manar ,Tunis, (Tunisia)*
Email: Saloua.BelHadjAli@enit.rnu.tn
Corresponding Author' Bahri Radwen

---***ABSTRACT***---
This paper deals with design and real time implementation of a novel navigation strategy applied to a unicycle mobile robot. The principle of navigation is inspired from the natural duck walk. A single sensor (gyroscope) is used in closed-loop to determine the posture of the robot. A global presentation of the robot with its different components are given firstly. An experimental identification for the useful parameters are then exhibited. Design of the neural network and practical validation are given afterwards. Finally, real time experiments are accomplished to valid simulation results.

KEYWORDS*-Robot Navigation, Duck walk, Gyroscope, Rotational movement*

Date of Submission: 12-05-2018 Date of acceptance: 28-05-2018

I INTRODUCTION

The emergence of mobile robots dated of 1970s used generally for some simple tasks. Since then, the evolution of computing and electronics enables robots to perform more complex interventions in a large variety of fields, with increasing autonomy and improved rapidity. Without human intervention in the field of navigation, mobile robots must have high-performance strategies to accomplish their task, but most of these strategies are based on classical position a velocity sensors. Nowadays, a great variety of position sensors have been developed. Nevertheless, measurement methods are limited to odometry in order to determine the position and the speed. In this regard, the simplest way to measure displacement and speed of a mobile robot is to place an encoder in the drive motors in the active wheels [J. Borenstein, (1994)], [T. Sauer, (2001)], [A.Martinelli, (2002)], [J. Palacn, (2004)] or even in additional passive wheels [J. borenstein (1994)]. However, measurements given by the encoder suffer from many problems such as accurate wheel diameter estimation, wheel wear and slippage and uncertainty in the determination of the contact point with the floor. Actually, there are various techniques for measuring the speed rather than odometry method and we are interested here in measuring the true velocity of the vehicle. Doppler navigation systems use the Doppler Effect to determine the speed of vehicle. The principle of the process is based on the Doppler shift in frequency observed when radiated energy reflects a surface that is moving with respect to the emitter. Those systems cause errors in detecting true ground, speed and temperature. They also affect the velocity of sound and oscillation frequency of a crystal oscillator which impact the velocity given by the sensor [OKAMOTO Kenji (2001] and require the existence of a reflection surface. Furthermore, the global positioning systems GPS suffer from signal masking in areas such as densely treed streets and tunnels [Lin Zhao (2003)] as well as sampling rate [L. Zhao, (2003)].

In order to achieve their goals, to find food or to migrate, animals must effectively navigate their environments. In this regard, animal locomotion can be defined as an harmonic act by applying forces to the environment [A. Cherubini (2007)]. Therefore, analyzing the locomotion of biological animals are fundamental in developing a novel architecture of robot to be able to adapt in many fields and satisfy certain needs. Many navigation strategies were inspired from animal's navigation which provide a good performance in achieving their goals and avoiding obstacles but most of them suffer from erroneous measurements due to the previously mentioned sensors.

One of the simplest navigation strategies is that of the duck, in fact it is easy to model, even at the implementation level it can be implemented on different types of mobile robots (walkers or wheeled). All of these proprieties encourage our research team to develop a new unicycle-type mobile robot's architecture to study this type of navigation on the theoretical level and the practical plan. A novel navigation strategy is proposed to overcome these problems.

This paper is arranged as follows. Section 2 presents the structural design, principle of the strategy of navigation of the robot. Section 3 presents the mathematical model, simulation results, practical validations and specifications for the novel strategy. Finally, the paper is ended with a conclusion.

II THE ROBOT DESIGN AND ARCHITECTURE

II.1 The Robot Design

The robot used in this study is a unicycle robot as shown in the next figure 1.

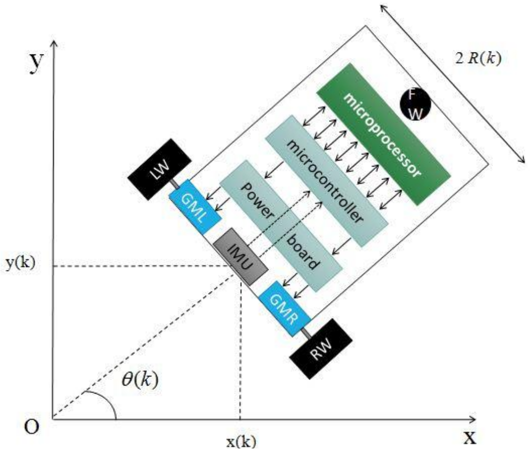


Figure 1: Robot architecture.

II.2 The Robot architecture

The Robot comprises a microprocessor to implement a neural network for the path planning which requires high calculation capacity and a microcontroller for interface with IMU and the power board managing the energy required for the operation of the actuators, the electronic part will be described later with the possibility of changing the geometry of the robot, exactly R(k) which presents the distance between wheels and the wheels center.

III NAVIGATION STRATEGY

This section describes the idea origin from the duck walk to develop a functional analysis and to show a mathematical description of the robot evolution in the plan.

III.1 FUNCTIONAL ANALYSIS

An example of a duck walk in a straight line parallel to the axis Y is given in the figure 2.

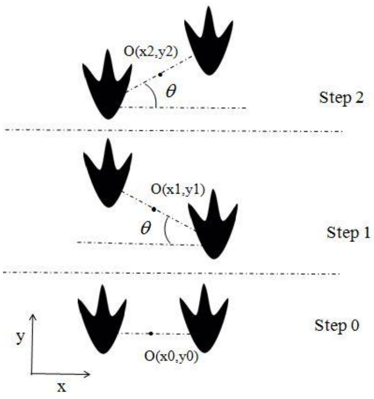


Figure 2: The duck walk

where O is the center of gravity and (x,y) are the Cartesian coordinates in the plane. It is assumed that the evolution medium of duck is a non-inclined plane, the direction is constant and we are interested in the evolution of the Y-ordinate, so we obtain:

$$y(k+1) = y(k) + \Delta y(1)$$

where

$$\Delta y = l.\sin(\Delta\theta)(2)$$

and

$$\theta(k+1) = \theta(k) + \Delta\theta(3)$$

III.2 Path Planing

It is assumed that the evolution medium of the duck is a non-inclined plane, $x_f > x_0$ and $y_f > y_0$. The new strategy relies on the change of the angular position of the driving wheels, one of which plays the role of the center of rotation and the other wheel moves. The robot posture in base frame is given by $q(k) = [x(k)\ \ y(k)\ \ \theta(k)]^T$. We assume that the initial pose of the robot is $q_i = [x_i y_i \theta_i]^T$ and the final pose is $q_f = [x_f y_f \theta_f]^T$. The first step is to determine the optimal distance D :

$$D = \sqrt{(x_f - x_i)^2 + (y_f - y_i)^2}(4)$$

The first step is to maximise the distance between the wheel and the center $L(k) = L_{max}$. Then a first one is a rotational movement to set the direction of navigation.

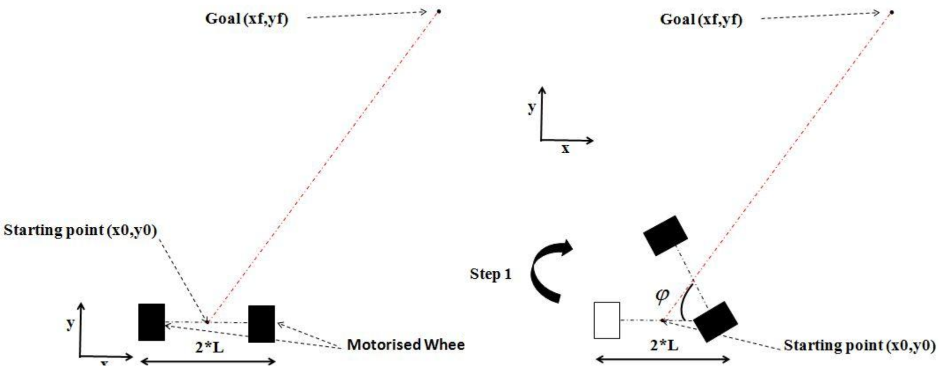


Figure 3.0 step 0 **Figure 3.1 step 1**

Figure 3: Rotational movement in a first step
with

$$\varphi = \pi - 2\theta(5)$$

and

$$\theta = \widehat{GOX}(6)$$

The second step is to compensate the first step by turning in the opposite direction with the same angle φ

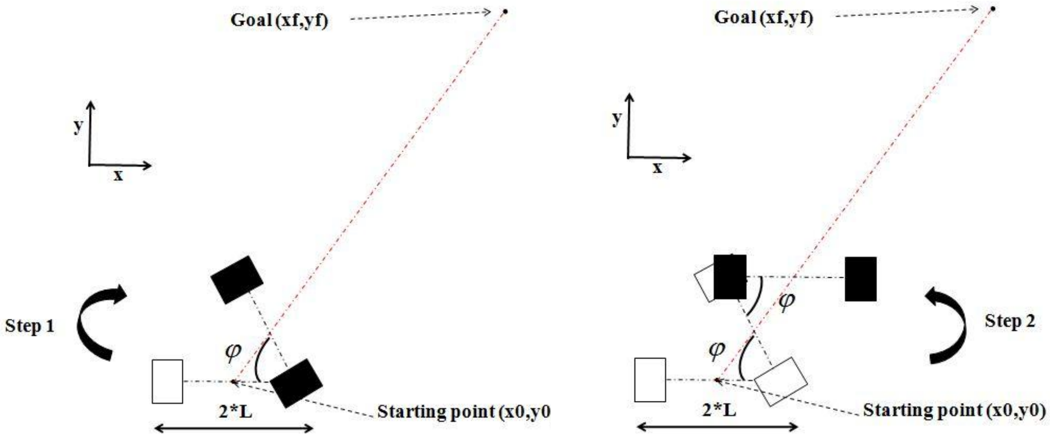


Fig 4.0step 1Fig 4.1step 2

Figure 4 : Rotational movement in a second step

Now, we focus on the calculation of the distance traveled and coordinates of the center of the robot after the first transition.

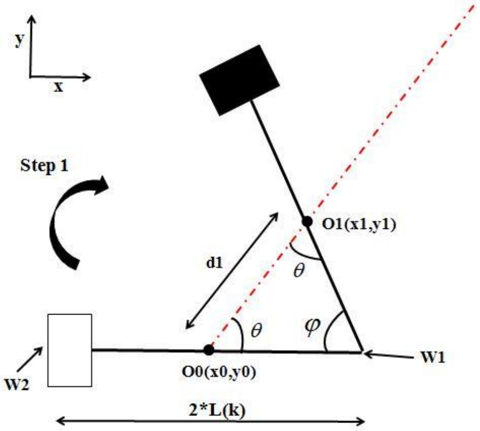


Figure 5: First step

In the isosceles triangle $W_1O_0O_1$ the opposite side of the angle φ presents the distance traveled by the robot in this step.

$$d1 = L(k)\frac{sin\varphi}{sin\theta}(7)$$

$$xo1 = d1cos\theta(8)$$

$$yo1 = d1sin\theta(9)$$

Also in the following step

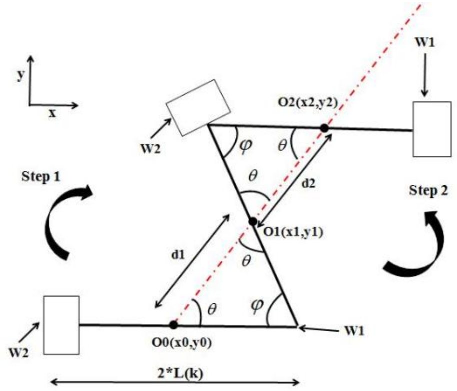


Figure 6: Second step

The same model

$$d2 = L(k)\frac{sin\varphi}{sin\theta}(10)$$

$$xo2 = (d2 + d1)cos\theta(11)$$

$$yo2 = (d2 + d1)sin\theta(12)$$

The total movement is given by the following figure

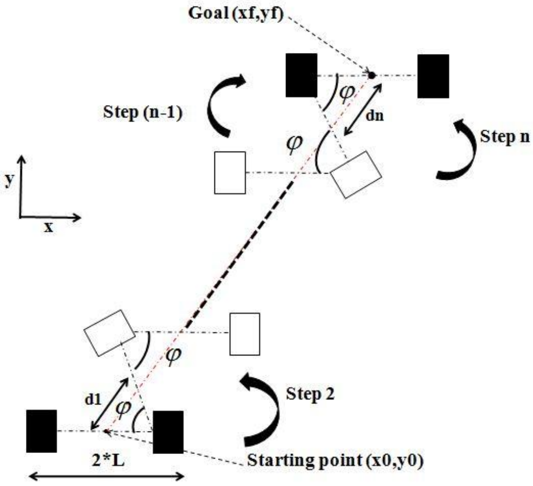


Figure 7: The entire cycle of movement

With

$$d_n = L(k)\frac{sin\varphi}{sin\theta}(13)$$

$$xon = (\textstyle\sum_{i=1}^{n} d_i)cos\theta(14)$$

$$yon = (\textstyle\sum_{i=1}^{n} d_i)sin\theta(15)$$

To ensure the convergence of the robot to the desired posture it is necessary to choose $L(k)$ in terms of the initial and the final posture of the robot.

$$L(k) = f(x_0 y_0 x_f y_f)(16)$$

III.3 Calculation of L(k)

The procedure of calculus of the optimal distance $L(k)$ is presented in this section. The relation between $L(k)$ and n is given by the following equation

$$D(n, L) = 2 * n * L(k)\frac{sin\varphi}{sin\theta}(17)$$

where n is the number of steps and $L(k)$ isthe radius of the robot.

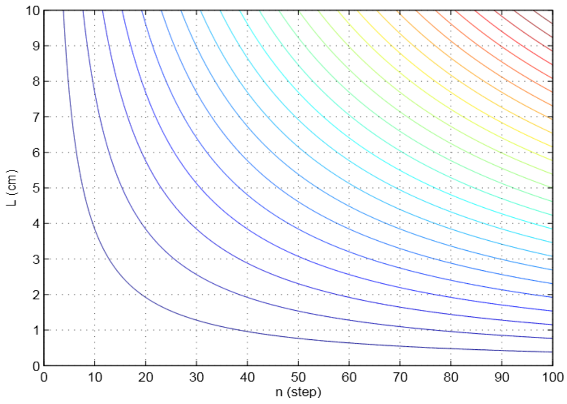


Figure 8: Variation of D in terms of (L,n)

The proposed approach to determine $L(k)$ is presented in the following steps

- Indicate the intitial pose of the robot $q_i = [x_i y_i \theta_i]^T$
- Indicate the final pose of the robot $q_f = [x_f y_f \theta_f]^T$
- Calculate the potimal distance $D = \sqrt{(x_f - x_i)^2 + (y_f - y_i)^2}$
- Calculate $\theta = \arcsin(\frac{y_f}{D})$
- Calculate $\varphi = \pi - 2\theta$
- Calculate $d_i = L_i \frac{sin\varphi}{sin\theta}$

- Calculate $m = Dmod(d_i)$
- Calculate $n = (\frac{D}{d_i}) - (\frac{m}{d_i})$
- Calculate $\Delta L = \frac{m}{n}.\frac{sin\theta}{sin\varphi}$
- Update $L_i = L_i + \Delta L$

The general algorithm is given as follow

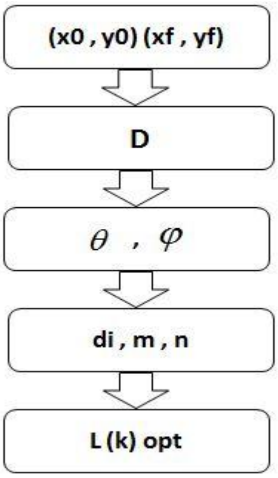


Figure 9: General Algorithm

III.4 Controller Design

To estimate the robot position in the desired trajectory, the angular encoders will be get rid using a novel navigation strategy based on a single sensor (gyroscope) which gives us the angular position. The calculation of the position which the robot must reached necessitates a somewhat special regulator by proposing the following regulators shown in fig. 10.

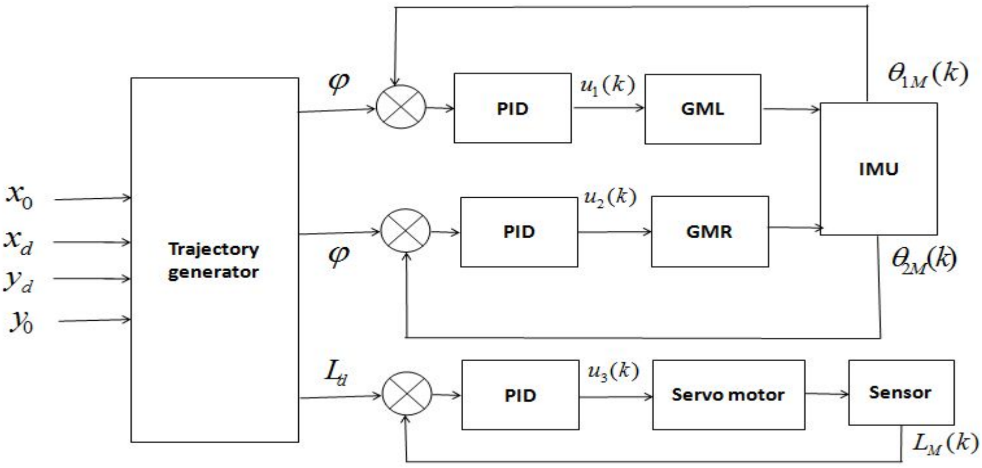


Figure 10: Controller Design

III.5 Development of the robot

The robot was developed in our laboratory, fig. 11. It is composed of 2 motorized wheels ensuring mobility (1,2). An inertial measurement unit IMU (mpu9150 (7)) is introduced to determine the latitude of the robot (roll, pitch, and yaw) in the field of evolution, in our work, only the Yaw angle is considered. There is also a raspberry pi 3 board that serves as a microcomputer (3) with high performance (CPU 4 * ARM Cortex-A53, 1.2GHz with 1GB of RAM) to generate trajectory to the robot by yielding the desired angle θ. The robot embedded a microcontroler (5) who treats data from sensor and generates control signals to the electronic board (the power board is based on the integrated circuit L298N designed for the control of speed and direction of the DC motors) (6) which controlled the actuators. A LiPo battery 2s is used for power alimentation.

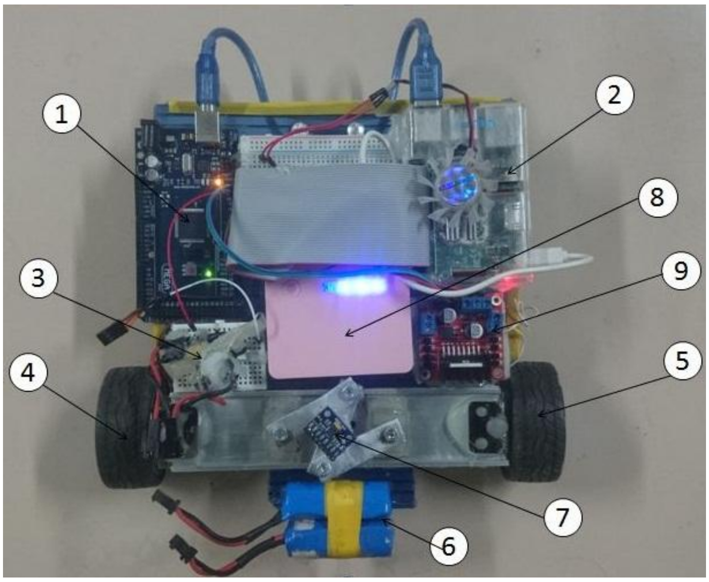


Figure 11: The developed prototype robot

The mass of the robot is 500g with the dimensions $200 \times 210 \times 75mm$ and the possibility of wireless communication with the PC. The only difference between the new architecture of the robot developed and the classic unicycle robot is the capacity to modify the term L(k) which presents the distance between the center of the robot and its driving wheels.

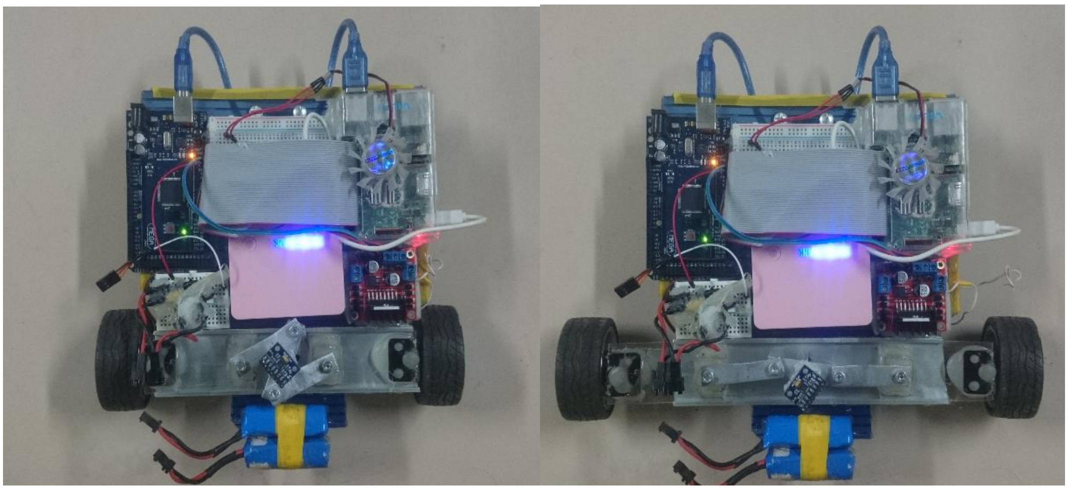


Figure 12.0 L(k) minimalFigure 12.1 L(k) maximal

Figure 12: The novel architecture

IV IDENTIFICATION AND SIMULATION

IV.1 Experimental identification

In order to control the robot, a mathematical model is required for the actuators. Experimental identification is used to determine the angular speed in terms of voltage, a pseudorandom binary sequence (PRBS) has been applied to the actuators shown in fig. 36.

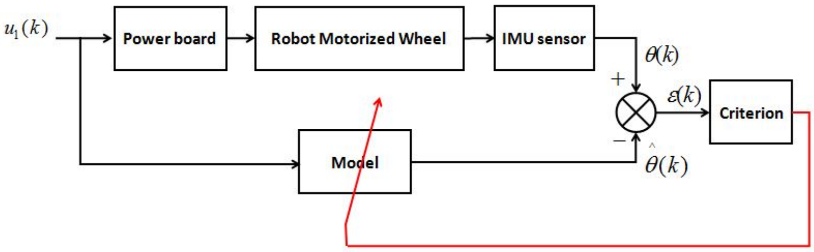


Figure 13: Identification principle

Real time identification test is presented in the following figure

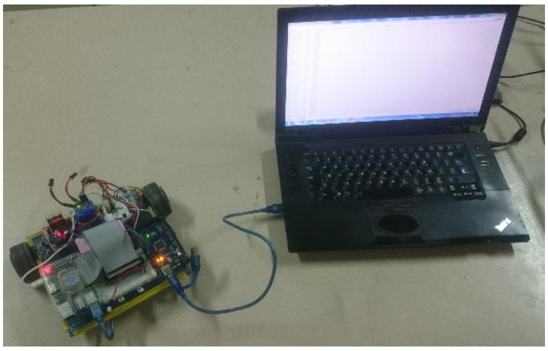
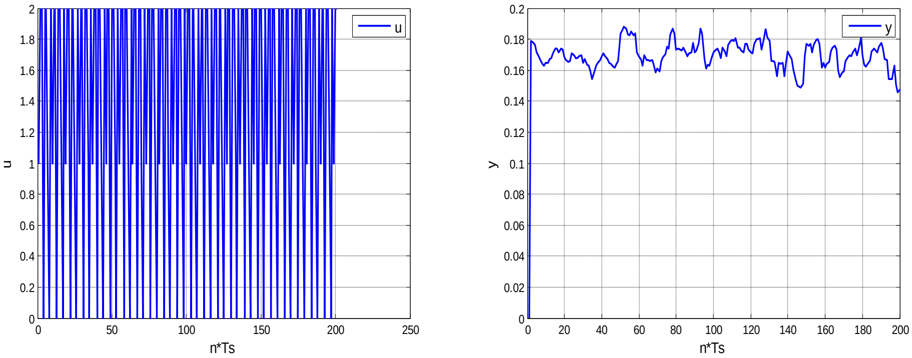


Figure 14: Real time identification test

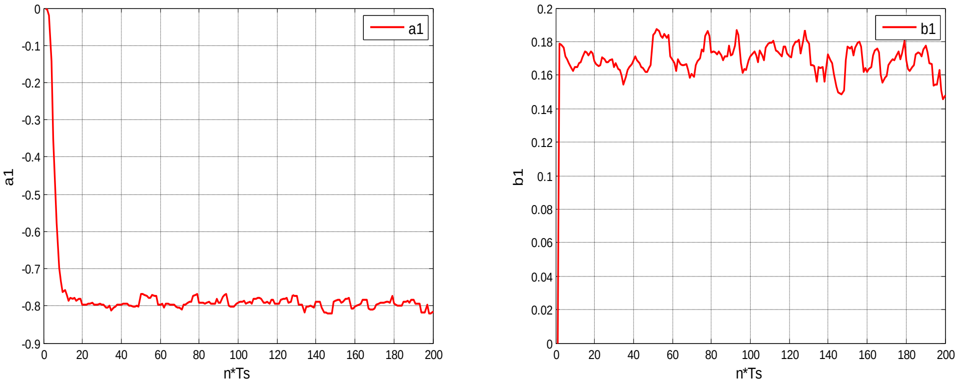
The mathematical model of the angular velocity is given by

$$H(z) = \frac{b1}{z+a1} \quad (18)$$

The measurement (Input/Output) is given by the following figure



(a) Input signal PRBS (b) System response for PRBS
Figure 15: Input Output signals



(a) Estimated a1 **(b) Estimated a1**

Figure 16: Estimated parameters

The step response signal is measured within real time experience, and afterwards plotted by a soft computing simulation.

$$\begin{cases} b1 = 0.17 \\ a1 = -0.79 \end{cases}$$

After obtaining the mathematical model of the angular velocity, the model of the angular position can be obtained by simple integration.

$$H(p) = \frac{k}{(1+\tau p)p} \tag{19}$$

with

$$\begin{cases} k = 0.81 \\ \tau = 0.22s \end{cases}$$

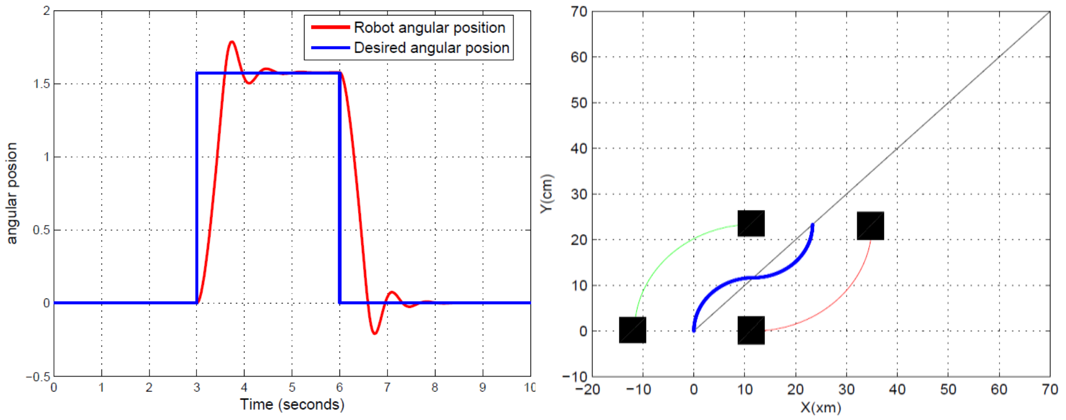
As a controller, a PID controller is chosen with three parameters (k_p,T_i,T_d).

IV.2 Numerical Simulation

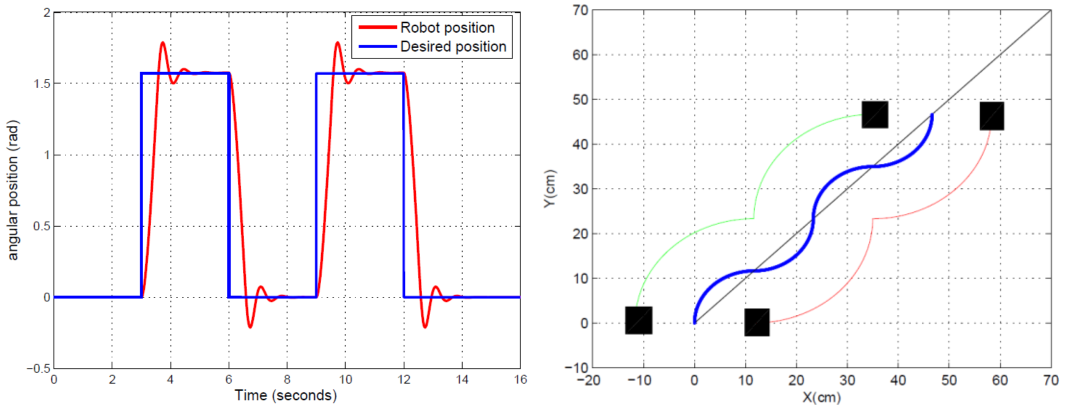
For the numerical simulation, the starting point of the initial robot posture in base frame is given by $q_i = [0\ \ 0\ \ 0]^T$ and the final pose is $q_f = [0.7\ \ 0.7\ \ 0]^T$. The different parameters are chosen as

- The potimal distance $D = \sqrt{(x_f - x_i)^2 + (y_f - y_i)^2} = 0.9899m$
- $\theta = \arcsin(\frac{y_f}{D}) = 0.7854rad$
- $\varphi = \pi - 2\theta = \pi/2rad$
- $d_i = L_i \frac{sin\varphi}{sin\theta} = 0.2828m$
- $m = Dmod(d_i) = 0.2262m$
- $n = (\frac{D}{d_i}) - (\frac{m}{d_i}) = 3$
- $\Delta L = \frac{m}{n} * \frac{sin\theta}{sin\varphi} = 0.0266m$
- $L_i = L_i + \Delta L = 0.1166m$

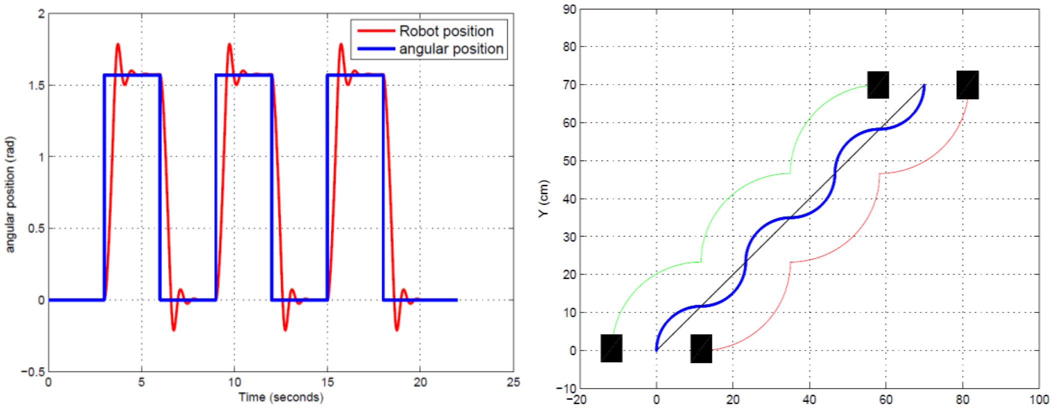
The simulation of is given by the Fig. 17.



(a) angular position(b)Robot trajectory
Figure 17: Simulation of the trajectory (n=1)



(a) angular position(b)Robot trajectory
Figure 18<: Simulation of the trajectory (n=2)



(a) angular position (b)Robot trajectory
Figure 19: Simulation of the trajectory (n=3)

V.6 PID Implementation

To implement the PID controller in the real closed-loop system, the transfer function of the command signal must be discretized to deduce the digital control law as:

$$u(k) = u(k-1) + r0.\varepsilon(k) + r1.\varepsilon(k-1) + r2.\varepsilon(k-2) \tag{20}$$

where

$$\begin{cases} r0 = K_p + k_i + K_d \\ r1 = -(K_p + 2 * K_d) \\ r2 = K_d \\ Ts = 0.05s \end{cases}$$

To program the control law, the steps of the algorithm are:

- Initialise $u_0 = 0$, $\varepsilon_1 = 0$ and $\varepsilon_2 = 0$
- Read the measurement y
- Calculate the error $\varepsilon = y_c - y$
- Calculate and send the control law $u(k) = u_0 + r0.\varepsilon + r1.\varepsilon_1 + r2.\varepsilon_2$
- Save $u_0 = u(k)$, $\varepsilon_1 = \varepsilon(k)$ and $\varepsilon_2 = \varepsilon(k-1)$
- Wait until the end of the sampling period
- Go to the first step b.

In real time test, the following track was selected as the robot's evolution medium in order to validate the simulation as shown in Fig. 20

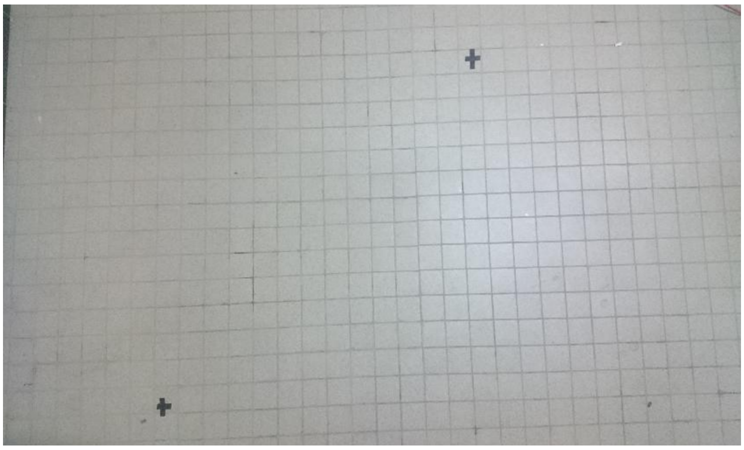


Fig 20. The evolution medium

The variation of L(k) in real time test are given in the figures bellow

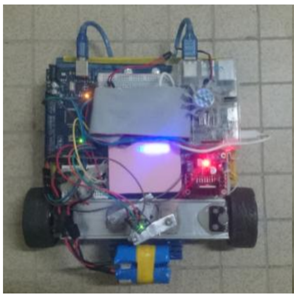


Figure 36: Initial L(K)

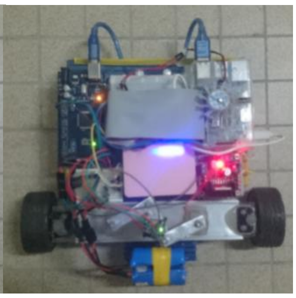
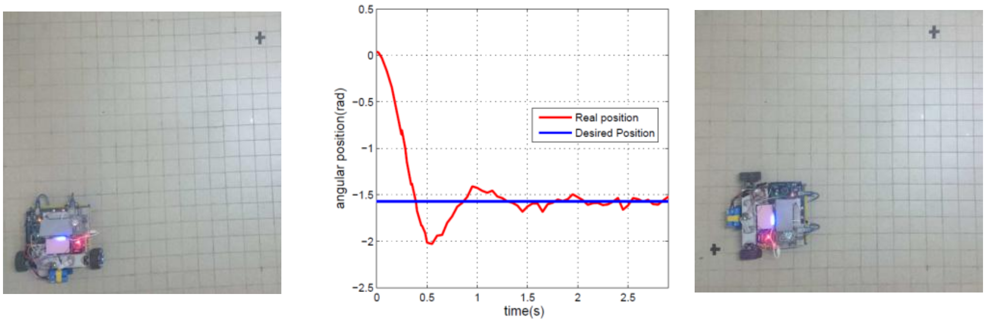
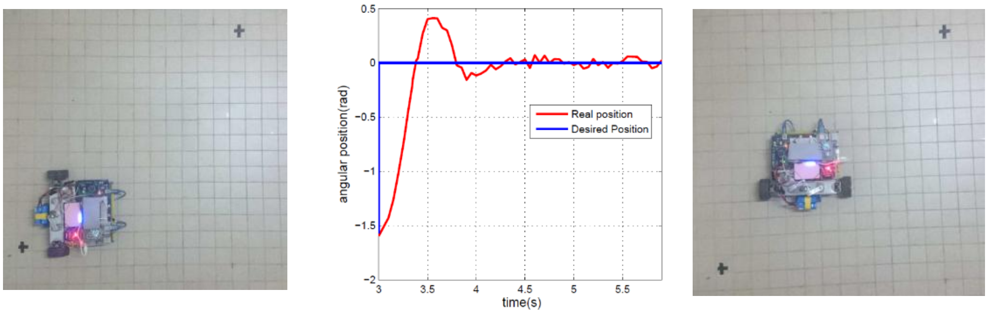


Figure 36: final L(K)

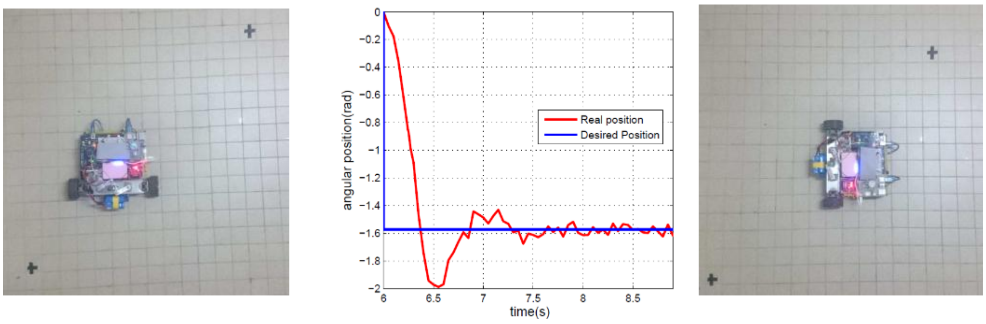
The evolution step of the robot in real time test are given in the figures bellow



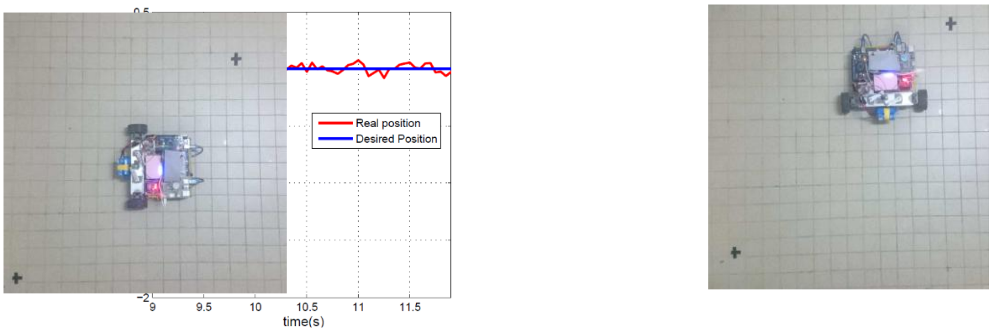
a) Real position b) Real positionc) Real position
Figure 23: STEP 0



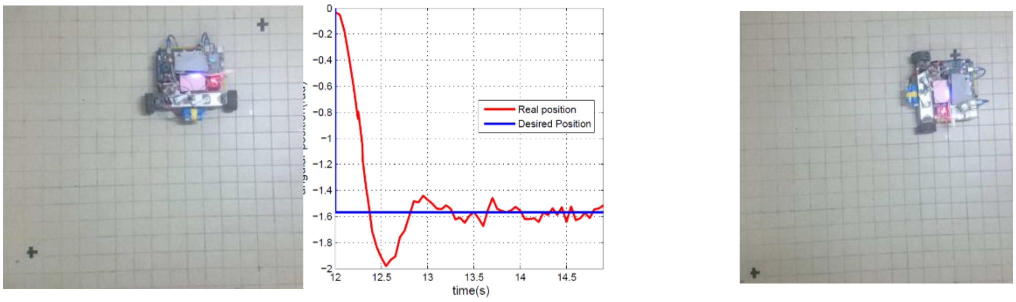
a) Real position b) Real positionc) Real position
Figure 23: STEP 1



a) Real position b) Real positionc) Real position
Figure 23: STEP 2

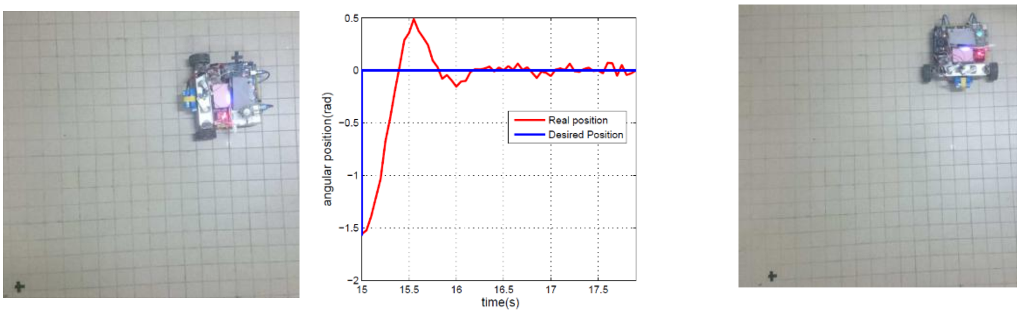


a) Realposition b) Real position c) Real position
Figure 23: STEP 3



a) Real position **b) Real position** **c) Real position**

Figure 23: STEP 4



a) Real position **b) Real position** **c) Real position**

Figure 23: STEP 4

The strategy characterization are given in the table bellow

Table 1 Strategy characterization

Characteristics	advantages or disadvantages
precision	++
complexity	++
Speed	–
Power consumption	++

V CONCLUSION

In this work, a novel navigation strategy for an unicycle mobile robot is developed. It presents a good performance with high accuracy. The strong point in this strategy is that the rotary encoder is no longer used which has a major perceptual problem in robotic systems.

ACKNOWLEDGEMENTS

This work was supported by the Ministry of the Higher Education and Scientific Research in Tunisia.

REFERENCES

[1]. [T. Sauer, (2001)] T. Sauer, H.Brugger, P. Hofer,(2001) Odometry error correction by sensor fusion for autonomous mobile robot navigation., robotics system , 152(2),1654–1658.

[2]. [A.Martinelli, (2002)] A.Martinelli,(2001) *The odometry error of a mobile robot with a synchronous drive system.*, IEEE Transactions on Robotics and Automation, 18(3):399–405.

[3]. [J. Palacn, (2004)] J. Palacn, J. Salse, Ignasi Valgan.,(2004) *Building a mobile robot for a floor-cleaning operation in domestic environments.*, IEEE Transactions on Instrumentation and Measurement , 1418-1424.

[4]. [J. Borenstein, (1994)] Johann Borenstein.,(1994) *A dual-drive mobile robot with internal correction of dead-reckoning errors.*, IEEE Transactions on Instrumentation and Measurement , 3085–3090.

[5]. [O. Kenji, (2001)] O. Kenji,(1994) *Ultrasonic doppler speed sensor with an offset parabolic reflector.*, JSAM, 63(2):53–59.

[6]. [L. Zhao, (2003)] Lin Zhao, Y. Ochieng, Mohammed Quddus, and Robert Noland.(2003) *An extended kalman filter algorithm for integrating gps and low cost dead reckoning system data for vehicle performance and emissions monitoring.*,The journal of Navigation, 56(02):257–275.

[7]. [J. borenstein (1994)] Johann Borenstein. *A dual-drive mobile robot with internal correction of dead-reckoning errors.* IEEE Transactions on Instrumentation and Measurement 3085–3090, 1994.

[8]. [OKAMOTO Kenji (2001] OKAMOTO Kenji. *Ultrasonic doppler speed sensor with an offset parabolic reflector.* .11em plus .33em minus .07em 63(2):53–59, 2001.

[9]. [Lin Zhao (2003)] Lin Zhao, Washington Y Ochieng, Mohammed A Quddus, and Robert B Noland. *An extended kalman filter algorithm for integrating gps and low cost dead reckoning system data for vehicle performance and emissions monitoring.* The journal of Navigation, 56(02):257–275, 2003.

[10]. [A. Cherubini (2007)] A. Cherubini, G. Oriolo, F. Macri *Development of a Multimode Navigation System for an Assistive Robotics Project.*IEEE International Conference on Robotics and Automation, 2007..

Bahri Radwen.”An Unicycle Mobile Robot Employing Duck Strategy For Navigation." The International Journal of Engineering and Science (IJES) 7.5 (2018): 11-24