Modeling, Control, Simulation and Prototyping of Tracked Mobile Robot

Eugenio Flores C.[1] and Francisco J. Triveno Vargas[2]

Abstract— Currently, many of the problems studied around mobile robotics aim to solve the provision of greater autonomy. This paper extends the field of mobile robotics with sliding wheels and differential traction. The main contributions of this work are: the use of specialized tools (software) to simulate the various stages of modeling and for the creation of prototype. Specifically, the MatLab Simulink® software is used for the simulation of the model exported from SolidWorks to the Simulink environment integrated with the controller design. Also, as mentioned before, SolidWorks for the project and simulation of the robot's mechanical components (Simscape application for multibody mechanical systems). Control laws, and trajectory tracking are based on kinematic and dynamic models. Desired trajectories are generated through parametric functions, continuous lines from specific points through interpolation techniques and using desired speeds. The results of simulation for three cases are presented and analyzed. Finally, the prototype and the experimental results are presented.

I. INTRODUCTION

Mobile vehicles have a wide range of applications where they can operate autonomously in various environments. It has been shown that they can be used to perform a variety of tasks in industrial, medical, agricultural, exploration, defense, security, rescue and military applications [1]. As current examples, it is possible to mention hypersonic missiles that combine the ability to carry out a prolonged flight at speeds greater than Mach 5, i.e., five times the sound speed [2]. Unmanned aerial systems should also be highlighted, which today are not only used for intelligence, reconnaissance, surveillance and destruction of a target, but also as loitering munitions that search for their target, attack and explode on it [3]. In the same way, unmanned ground vehicles are widely used in many industries where repetitive tasks or high-risk missions are carried out, in an inaccessible environment or within hostile terrain are required [4].

The mentioned systems are the result of years of research between industry and academy in the countries that hold this knowledge. To reduce the gap due to this exponential technological growth, the countries around the world that do not have it must establish development objectives in long term so as not to be left behind, for this reason several works and studies of aerial and ground robots are being development around the world [5], [6] [7], [8].

Bolivia is one of the countries that intends to establish robotics development plans. In this sense, the country began with educational robotics from the year 2000, at the same time two master's courses were executed in the area, one between 2000 and 2002 and the last one between 2016 and 2020. This article corresponds to the master's degree work presented in 2020 in the middle of the pandemic. The work deals with ground mobile robot with sliding wheels and differential traction (caterpillar type robot), includes mathematical modeling using CAD tools, the controllers design, dynamic simulation and finally the construction and implementation of the robot.

The article considers the basic problem of a mobile robot navigation, i.e., the movement from an origin point to another destination point along a trajectory and with a required orientation, the problem becomes more complicated if this distance increases. The other problem considered is the control of the linear and angular speeds that must be applied so that the robot can follow the path and reach the desired objective [9].

In the development two alternatives were considered to automate the movement of a robot: the first consists of the use of sensors and actuators, controlled by a micro controller programmed in an understandable language for the user, this method is known as the empirical method without scientific basis. The second method is automation through control laws based on mathematical models, known as the science-based method. This work adopts and use the second method.

The main contributions of this work are: the use of specialized tools (software) to simulate the various stages of modeling and for the creation of prototype. Specifically, the MatLab-Simulink® software is used for the simulation of the model exported from SolidWorks to the Simulink environment integrated with the controller design. Also, Mathematica™ software is used to solve complex differential equations [10] and, as mentioned before, SolidWorks for the project and simulation of the robot's mechanical components (Simscape application for multi-body mechanical systems).

Control laws, and trajectory tracking are based on kinematic and dynamic models. Desired trajectories are generated through parametric functions, continuous lines from specific points through interpolation techniques and using desired speeds. The results of simulation for three cases are presented and analyzed. Finally, the prototype and the experimental results are presented.

[1] Eugenio is with INESCO S.A. Ingeniería y Diseño, Av. Alemana, No. 2030, Santa Cruz, Bolivia eugenio.florecol@gmail.com
[2] Francisco is with Universidade de Araraquara, Rua Maria Antônia de Camargo, 170, CEP: 14800-370, Araraquara, SP, Brasil trivenoj@yahoo.com.br

II. MATHEMATICAL MODELING

The mathematical model of any plant is a representation using mathematical equations, functions, or formulas, in the case of tracked mobile robot is very well introduced in the literature (see Figure 1) [11], [12].

A. Tracked vehicle kinematics

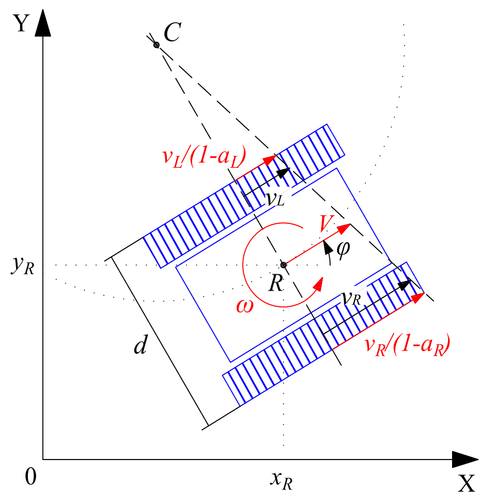

Fig. 1. Geometry of the robot with differential traction.

The kinematics of the robot depends on a fixed geometric structure in a generalized coordinate system, it can adopt a Cartesian, polar, cylindrical, spherical or articulated structure. The mobile robot with two, three or more wheels adopts one considering restrictions or not in its movement [13]. The presentation of the kinematic equations is a prerequisite for the dynamic study.

The position and orientation of the robot in general Cartesian coordinates are given by:

$$P = [x_r, y_r, \varphi_r]^T \tag{1}$$

Considering Figure 1, the speed in generalized coordinates at point R for any robot [14], is given by:

$$\begin{cases} \dot{x}_r = V\cos\varphi \\ \dot{y}_r = V\sin\varphi \\ \dot{\varphi}_r = \omega \end{cases} \tag{2}$$

For robots with differential traction, where wheel slip is considered, the tangential speed of each wheel is [5]:

$$v_L \frac{1}{1-a_L} = r\dot{\theta}_L \quad ; \quad v_R \frac{1}{1-a_R} = r\dot{\theta}_R \tag{3}$$

Where, v_L and v_R are the left and right wheel tangential speed. r is the equivalent radius of the drive wheel. $\dot{\theta}_L$ and $\dot{\theta}_R$ are the left and right wheel angular velocities.

a_L and a_R are the longitudinal slip ratio of the left and right wheels and are given by:

$$\begin{aligned} a_L &= \frac{(r\omega_L - v_L)}{r\omega_L}, \qquad 0 \le a_L < 1 \\ a_R &= \frac{(r\omega_R - v_R)}{r\omega_R}, \qquad 0 \le a_R < 1 \end{aligned} \tag{4}$$

The two slider parameters are written as P_L and P_R, as follows:

$$P_L = \frac{1}{1-a_L}, \quad P_R = \frac{1}{1-a_R} \tag{5}$$

Therefore, the Equation (3) is rewritten as:

$$v_L = \frac{r}{P_L}\dot{\theta}_L \quad ; \quad v_R = \frac{r}{P_R}\dot{\theta}_R \tag{6}$$

The linear and angular speeds of the right and left wheels with respect to the generalized coordinates, are given by:

$$\begin{aligned} v_R &= V + \frac{d}{2}\omega \\ v_L &= V - \frac{d}{2}\omega \end{aligned} \tag{7}$$

or,

$$\begin{aligned} \dot{\theta}_R &= \frac{P_R}{r}V + \frac{dP_R}{2r}\omega \\ \dot{\theta}_L &= \frac{P_L}{r}V - \frac{dP_L}{2r}\omega \end{aligned} \tag{8}$$

Adding v_R and v_L in the Equation (7), the linear velocity V of the robot is given by:

$$V = \frac{v_R + v_L}{2} \tag{9}$$

Subtracting v_R and v_L in the Equation (7), the angular velocity ω of the robot is:

$$\omega = \frac{v_R - v_L}{d} \tag{10}$$

where, d is the distance between the left and right wheels of the robot.

From the equations presented early, direct and inverse kinematic equations are obtained.

Substituting the Equations (6), (9) and (10) into the Equation (2), the direct differential kinematics is:

$$\dot{q} = J\dot{p}$$

$$\begin{bmatrix} \dot{x}_r \\ \dot{y}_r \\ \dot{\varphi} \end{bmatrix} = \begin{bmatrix} \frac{r}{2P_R}\cos\varphi & \frac{r}{2P_L}\cos\varphi \\ \frac{r}{2P_R}\sin\varphi & \frac{r}{2P_L}\sin\varphi \\ \frac{r}{dP_R} & -\frac{r}{dP_L} \end{bmatrix} \cdot \begin{bmatrix} \dot{\theta}_R \\ \dot{\theta}_L \end{bmatrix} \tag{11}$$

To obtain the inverse kinematics, V is calculated from its components:

$$V = \dot{x}cos\varphi + \dot{y}sin\varphi \tag{12}$$

Manuscript 459 submitted to 2023 9th International Conference on Control, Decision and Information Technologies (CoDIT). Received March 14, 2023.

Replacing the Equation (12) in the Equation (8), the following equations are obtained:

$$\dot{\theta}_R = \frac{P_R}{r}(\dot{x}cos\varphi + \dot{y}sin\varphi) + \frac{dP_R}{2r}\omega$$
$$\dot{\theta}_L = \frac{P_L}{r}(\dot{x}cos\varphi + \dot{y}sin\varphi) - \frac{dP_L}{2r}\omega \tag{13}$$

From Equation (13) the inverse differential kinematics

$$\dot{p} = J^{-1}\dot{q}$$

is:

$$\begin{bmatrix} \dot{\theta}_R \\ \dot{\theta}_L \end{bmatrix} = \begin{bmatrix} \frac{P_R}{r}\cos\varphi & \frac{P_R}{r}\sin\varphi & \frac{dP_R}{2r} \\ \frac{P_L}{r}\cos\varphi & \frac{P_L}{r}\sin\varphi & -\frac{dP_L}{2r} \end{bmatrix} \cdot \begin{bmatrix} \dot{x}_r \\ \dot{y}_r \\ \dot{\varphi} \end{bmatrix} \tag{14}$$

Observe that if the absolute speed is equal to the theoretical speed $a_L = 0$ and $a_R = 0$, then $P_L = P_R = 1$, therefore, the kinematics of mobile robots with conventional wheels analyzed in the center of mass is obtained.

1) Non-holonomic constraints: The speed at point R is in the direction of the longitudinal axis of symmetry, that is, the robot moves in the direction in which the traction wheels are located and the movement is due to the movement of the sliding wheels.

From the Equation (2), the following is obtained:

$$V = \frac{\dot{x}}{\cos\varphi} = \frac{\dot{y}}{\sin\varphi}$$

Doing operations, the restriction is obtained as:

$$-\dot{x}\sin(\varphi) + \dot{y}\cos(\varphi) = 0 \tag{15}$$

or in matrix form,

$$M(q) = \begin{bmatrix} -\sin(\varphi) & \cos(\varphi) & 0 \end{bmatrix}, \quad \dot{q} = \begin{bmatrix} \dot{x}_r \\ \dot{y}_r \\ \dot{\varphi}_r \end{bmatrix}$$

The constraint $M(q)\dot{q} = 0$ with three columns and one row. From the equation $B(q)M(q) = 0$, the matrix $B(q)$ is obtained.

$$B(q) = \begin{bmatrix} \cos(\varphi) & \cos(\varphi) \\ \sin(\varphi) & \sin(\varphi) \\ d/2 & -d/2 \end{bmatrix}$$

B. Tracked vehicle dynamics

The dynamic model of a mobile robot is obtained using different mechanical laws based on three physical elements: inertia, elasticity and friction present in any system of type. However, most of the robots used in practice use conventional wheels subject to non-holonomic constraints, for this reason, they require special treatment [13], [10]. In this sense, the Lagrange dynamic model of a non-holonomic robot (fixed or mobile) has the following form:

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{q}}\right) - \frac{\partial L}{\partial q} + M^T(q)\lambda = E\tau \tag{16}$$

Assuming that the robot is rigid and that the velocity is measured at the center of mass, the kinematic energy is given by:

$$K = \frac{1}{2}m(\dot{x}^2 + \dot{y}^2) + \frac{1}{2}J\dot{\varphi}^2 \tag{17}$$

The Langrangean is given by: $L = K - P$, since the robot moves in the XY plane, the potential energy $P = 0$. Therefore, $L = K$. Rewriting the Equation (17), is obtained:

$$L = \frac{1}{2}m(\dot{x}^2 + \dot{y}^2) + \frac{1}{2}J\dot{\varphi}^2 \tag{18}$$

Partial derivatives of L with respect to the variables $\dot{x}$, $\dot{y}$ and $\dot{\varphi}$ to solve Equation (16), are given by:

$$\frac{\partial L}{\partial \dot{x}} = m\dot{x}, \qquad \frac{\partial L}{\partial \dot{y}} = m\dot{y}, \qquad \frac{\partial L}{\partial \dot{\varphi}} = J\dot{\varphi} \tag{19}$$

Total derivative related to time of Equation (19), allow to obtain:

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{x}}\right) = m\ddot{x}, \quad \frac{d}{dt}\left(\frac{\partial L}{\partial \dot{y}}\right) = m\ddot{y},$$
$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\varphi}}\right) = J\ddot{\varphi} \tag{20}$$

substituting the Equations (19) and (20) in (16), the obtained matrix is:

$$D(q) = \begin{bmatrix} m & 0 & 0 \\ 0 & m & 0 \\ 0 & 0 & J \end{bmatrix}$$

The special case of mobile robot with sliding tracks and differential traction, the Equation (16) can be written as:

$$\overline{D}\ddot{p} = \overline{B}(q)\tau \tag{21}$$

where, $\ddot{p} = [\ddot{\theta}_R, \ddot{\theta}_L]^T$ are the left and right wheels accelerations, $\tau = [\tau_R, \tau_L]^T$ is the input vector taht represens the generalized forces of left and right wheels, $\overline{D} = J^T(q)DJ(q)$ and $\overline{B}(q) = J^T(q)B(q)$. J are the Jacobians of the direct differential kinematic model of the robot given by the Equation (11).

III. TRACKING CONTROL SYSTEM DESIGN

A. Control design based on the kinematic model

The objective of a control law is to find an auxiliary input $u_d = (V_d, \omega_d)^T$, such that:

$$\lim_{t\to\infty}(q_d - q_r) = 0$$

where the position of robot is $q_r = (x_r, y_r, \varphi_r)^T$ given by the Equation (2) and $q_d = (x_d, y_d, \varphi_d)^T$ is the desired reference, given by:

$$\dot{q}_d = M(q_d)u_d$$

that is,

$$\begin{bmatrix} \dot{x}_d \\ \dot{y}_d \\ \dot{\varphi}_d \end{bmatrix} = \begin{bmatrix} \cos\varphi_d & 0 \\ \sin\varphi_d & 0 \\ 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} V_d \\ \omega_d \end{bmatrix} \tag{22}$$

then, the following errors, in local coordinates of q_d are defined as:

$$\begin{bmatrix} e_x \\ e_y \\ e_\varphi \end{bmatrix} = \begin{bmatrix} \cos\varphi & \sin\varphi & 0 \\ -\sin\varphi & \cos\varphi & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x_d - x_r \\ y_d - y_r \\ \varphi_d - \varphi_r \end{bmatrix} \tag{23}$$

Deriving the Equation (23) related to time. Therefore, replacing the Equations (2) and (22) in the derivative of (23), the non-linear model is given by [15]:

$$\begin{cases} \dot{e}_x = \omega e_y + V_d \cos(e_\varphi) - V \\ \dot{e}_y = -\omega e_x + V_d \sin(e_\varphi) \\ \dot{e}_\varphi = \omega_d - \omega \end{cases} \tag{24}$$

Using the Lyapunov stabilization method technique to obtain the nonlinear controller, the candidate function is chosen as [6]:

$$V(e) = \frac{1}{2}e_x^2 + \frac{1}{2}(e_y + e_\varphi)^2 + (1 - \cos(e_\varphi)) \tag{25}$$

this Lyapunov function must satisfy the following proprieties:

1) $V(e)$, is a continuous function and it has continuous derivatives,
2) $V(0) = 0$, evaluated to zero, the function is zero,
3) $V(e) > 0$, for all $e \neq 0$.
4) $\dot{V}(e) \leq 0$, the derivative must be negative defined.

To satisfy these four properties, the conditions of $\dot{V}(e)$ must be verified. Deriving $V(e)$ related to time allows to obtain:

$$\dot{V}(e) = (V_d \cos(e_\varphi) + V)e_x + (V_d e_y + \omega_d - \omega)\sin(e_\varphi) + (\omega_d - \omega)e_\varphi$$

To make $\dot{V}(e) \leq 0$ the inputs V and ω are selected such that:

$$\begin{cases} V_C = V_d \cos e_\varphi - K_3 e_\varphi \omega + K_1 e_x \\ \\ \omega_C = \omega_d + \dfrac{V_d}{2}\left[K_2(e_y + K_3 e_\varphi) + \dfrac{1}{K_3}\sin e_\varphi\right] \end{cases} \tag{26}$$

where, for $K_1 > 0$, $K_2 > 0$ and $K_3 > 0$ is verified $\dot{V}(e) \leq 0$. Therefore, the controller V_C and ω_C guarantee full asymptotic tracking for the desired trajectory.

B. Control design based on dynamic model

The control law presented convert the desired speed input into a force control that takes into account the system dynamics. Such application has already been proposed in [16], this work implement the same experimentally.

Exposing τ from the Equation (21) of dynamic model, furthermore, assuming that u is an auxiliary control input, applying the following input force, it remains:

$$\tau = \overline{B}(q)^{-1}\overline{D}u \tag{27}$$

It is obtained the shape of Backstepping.

$$\dot{q} = M(q)\xi \tag{28}$$

$$\dot{\xi} = u \tag{29}$$

To design a control law using the backstepping technique, it is necessary to specify a input velocity ξ. This input ξ will generate the solution for kinematic tracking problem. Subsequently, the solution of the kinematic tracking problem will be denoted as a desired input velocity $\xi_d = (\omega_{Rd}, \omega_{Ld})^T$.

To obtain the input u that guarantees that the speed ξ, applied to the system, follows the desired speed ξ_d, the speed error is defined as:

$$e_d = \begin{bmatrix} e_{\omega_R} \\ e_{\omega_L} \end{bmatrix} = \xi - \xi_d \tag{30}$$

Therefore, the control law based on the dynamic model is given by the auxiliary input u, which guarantees that e_d converges to zero, it is given by the following equation:

$$u = \dot{\xi}_d - \begin{bmatrix} K_4 & 0 \\ 0 & K_5 \end{bmatrix}(\xi - \xi_d) \tag{31}$$

with K_4 and K_5 positive constants.

IV. TRAJECTORY GENERATION

When path-following control is performed on mobile robots, the paths are generally known and specified by mathematical functions, however, it is also desired that the robot be able to follow any other path defined by a series of points in the Cartesian plane (of otherwise it is necessary to use a GPS system). Two methods or means are known for the trajectories generation:

- Point-to-point planning.
- Continuous route generation.

The first method applies to position and orientation control, it does not require mathematical analysis. The second method consists in generating continuous routes from points or functions in the Cartesian plane, this work uses the following methods: polynomial interpolation, parametric functions and speed reference.

A. Polynomial interpolation

For the generation of a continuous path from specific points, interpolation methods are used, third and fifth order polynomials as spline functions. For a defined and continuous trajectory, a function can be found that comes as close as possible to the desired trajectory, one of the best known functions is the algebraic polynomial that has the following form [17], [18]:

$$P_m(\lambda) = a_0 + a_1\lambda + ... + a_{n-1}\lambda^{n-1} + a_n\lambda^n \tag{32}$$

where, n is a positive integer and $a_0, a_1, \ldots a_n$ are real constants.

Figure 2 illustrates the curves obtained from the reference points in Table I. It is necessary to observe that with the three order polynomial it is not possible to cover the desired points, with the fifth order polynomial the function is oscillating. Finally, with spline interpolation, the curve approximates the desired points as expected.

Manuscript 459 submitted to 2023 9th International Conference on Control, Decision and Information Technologies (CoDIT). Received March 14, 2023.

TABLE I

Reference points on the Cartesian plane

No	1	2	3	4	5	6	7	8	9	10	11
x	0.9	1.1	1.6	3.5	5.3	7	8.1	9	9.6	11	12.6
y	0	2	3.8	4.5	3.8	3.2	3.3	4	7	8	8.1

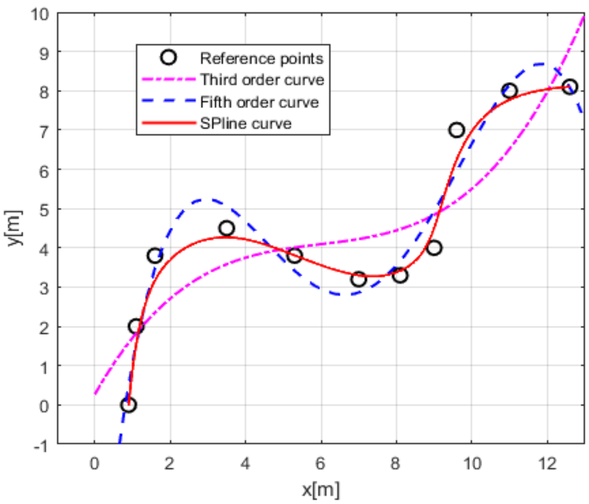

Fig. 2. Trajectory in the Cartesian space

B. Trajectory using speeds reference

The trajectory through speeds reference can be given in time intervals with a combination between the linear speed and the desired angular speed. Below is the trajectory used by [6].

$$
\begin{array}{rcl}
00s \leq t \leq 05s & V_d = 0.25(1-\cos(\pi t/5)) & \omega_d = 0 \\
05s \leq t \leq 20s & V_d = 0.5 & \omega_d = 0 \\
20s \leq t \leq 25s & V_d = 0.25(1+\cos(\pi t/5)) & \omega_d = 0 \\
25s \leq t \leq 30s & V_d = 0.15\pi(1-\cos(2\pi t/5)) & \omega_d = -V_d/1.5 \\
30s \leq t \leq 35s & V_d = 0.15\pi(1-\cos(2\pi t/5)) & \omega_d = V_d/1.5 \\
35s \leq t \leq 40s & V_d = 0.25(1-\cos(\pi t/5)) & \omega_d = 0 \\
40s \leq t & V_d = 0.5 & \omega_d = 0
\end{array}
$$

V. MOBILE ROBOT CAD MODEL

The development of the robot (prototype) begins with the modeling of parts or pieces in a CAD program; then these pieces are assembled in the same program. The created model is exported to Simscape (Matlab-Simulink®), where its movement is evaluated and its physical properties are defined. The Simscape model is brought into the simulation environment where the control laws, trajectory tracking, kinematics, and dynamic behavior of the robot are virtually evaluated. Finally, all the development is implemented (embedded)in a genuine prototype. Figure 3 illustrated all the steps followed up to the prototype.

The steps to take a CAD model to the simulation environment are the following: Export the XML file in Solidworks through the Simscape link; import the XML file into simulink, check and move tests in a Simulink block. Table II presents the data obtained from the mechanical design of the robot.

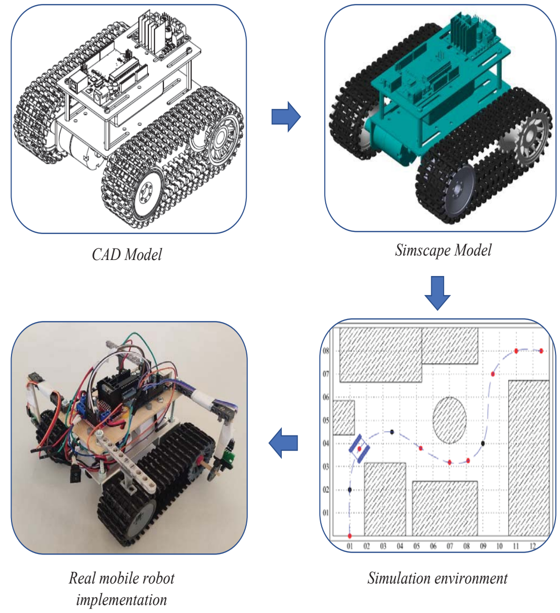

Fig. 3. Modeling and mechanical design of 3D mobile robot

TABLE II

Physical specifications of the robot simulated and real.

DETAIL	SYMB.	DATA
Wheelbase R/L	d	$0.1285[m]$
Wheel radius	r	$0.0257[m]$
Robot Weight	m	$0.5[Kg]$
Moment of inertia	J	$0.0013[Kg\cdot m^2]$
Motor/wheel torque	$\tau_R,\ \tau_L$	$0.0063[Kg\cdot m]$
Sliding wheels length	l	$0.1098[m]$
Longitudinal friction coeff.	u_x	0.800
Lateral friction coeff.	u_y	0.905

VI. PROTOTYPE AND EXPERIMENTAL RESULTS

Implementation is done using the scheme illustrated in Figure 4.

The control module and the physical components are the components of the implemented structure. The controller is the brain, since it houses all the algorithms and controllers fed back as such. This module receives the external reference (desired trajectory), the angular speed of the left and right wheels through the speed sensors and sends the modulated voltage (PWM) for speed control from the power module. This module can be embedded in Rapberry or Arduino. The physical components of the structure are the power module (H-bridge), the speed sensors coupled to the drive wheel (encoder), the drive wheels coupled to the motor, and the batteries.

Manuscript 459 submitted to 2023 9th International Conference on Control, Decision and Information Technologies (CoDIT). Received March 14, 2023.

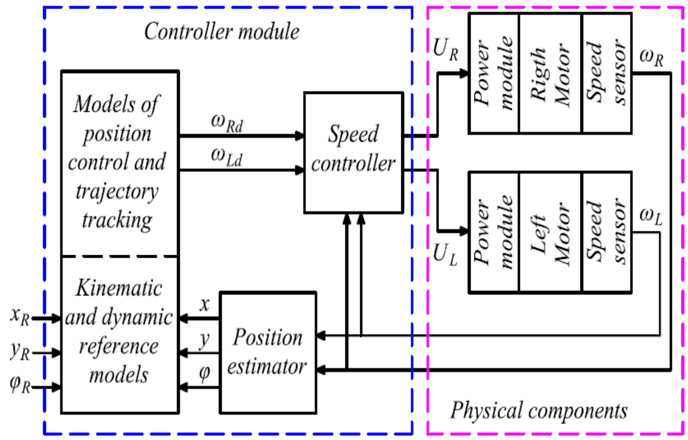

Fig. 4. Proposed scheme for implementation.

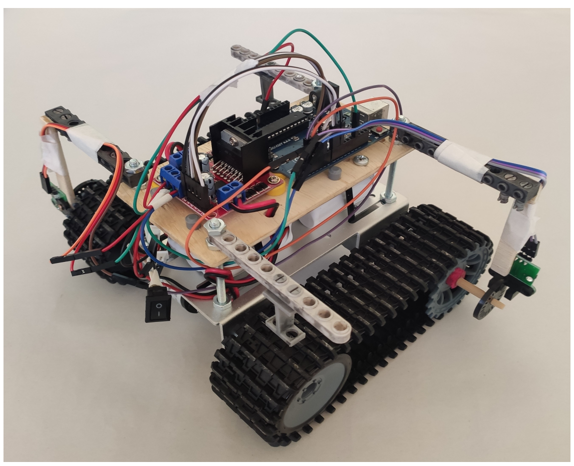

Fig. 5. Robot implemented.

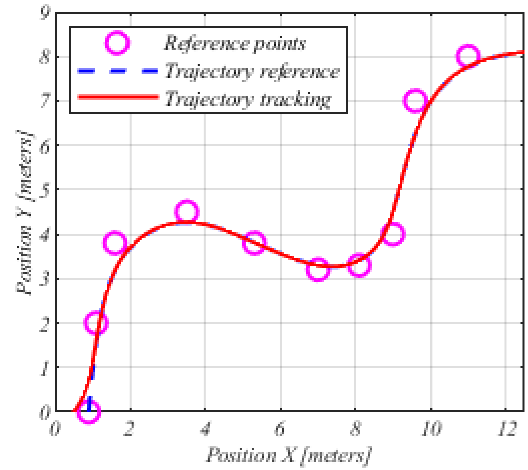

Fig. 6. Referenced and real trajectory of MR, first test

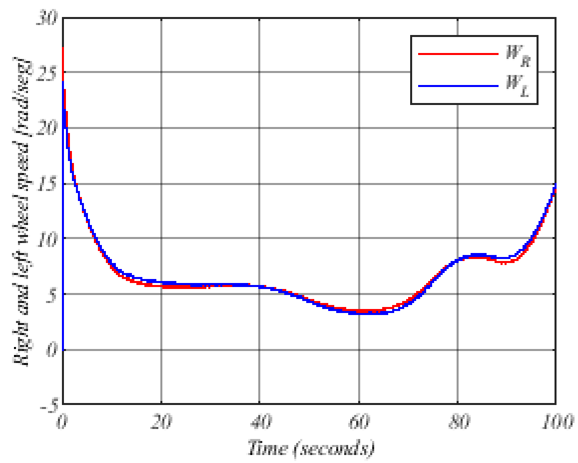

Fig. 7. Wheel speeds control, first test

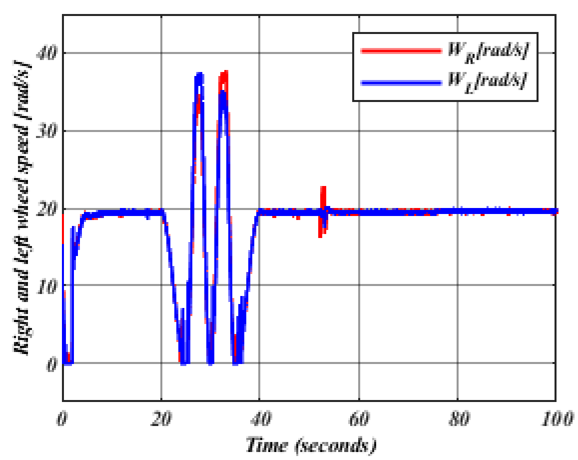

Fig. 8. The experimental results: wheel speed control, third test.

Figure 5 illustrates the built prototype, where all the algorithms generated in the simulation stage were embedded, generating the following results.

A. The first experimental result

The Figures 6 and 7, are the results obtained from the physical implementation of the first simulated test, note that the result obtained is similar, the speeds of the wheels were appropriate to the prototype, so the time was increased to $t = 100\ s$.

B. The third experimental result

The Figures 8 and 9, are the results obtained from the physical implementation of the third simulated test, the results are similar as expected.

For converting torque to angular velocity (of the wheels), the following relationship is included:

$$\omega_{R,L} = \int \frac{\tau_{R,L}}{I_{R,L}}, dt$$

where,
ω: Angular velocity of the wheel.
τ: Dynamic control torque.
I: Moment of inertia of the motor/wheel.

The kinematic and dynamic control is optimal, which

Manuscript 459 submitted to 2023 9th International Conference on Control, Decision and Information Technologies (CoDIT). Received March 14, 2023.

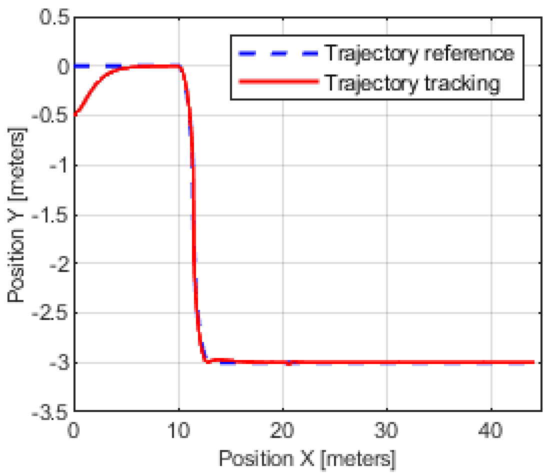

Fig. 9. The experimental results: required and real trajectory of MR, third test.

does not happen when the control is solely kinematic, as demonstrated in the first and second experiments.

VII. CONCLUSIONS

In this article, the kinematic/dynamic model of the mobile robot with sliding tracks and differential traction of small size was implemented for didactic purposes. It was used to design the feedback control algorithms for continuous path tracking based on previous work. Three methods were studied and presented to generate continuous trajectories that can be represented by a mathematical function.

The modeling of parts, pre-assembly and assembly of robot, was done using the Solidworks V2021. In the part assembly stage it is important to define the mathematical relations, reference axes and joints necessary to project and simulate the robot. After that, the robot model is exported. Simscape is an application developed for Matlab-Simulink® that is used to model multi-body mechanical systems: bodies, joints, motion stresses, force elements, kinematic and dynamic properties after export the mobile robot.

The simulation is done using the Matlab-Simulink® version R2021a and as commented Simscape application was used for the simulation of the robot, in this work: the rotational movement of the wheels, the translational movement and the rotational movement of the robot. The results obtained in the simulation are as expected, they can be verified in the different plots presented in this work.

When implementing the prototype, the same algorithms generated for the simulation were embedded with some adjustments and scaling in the control variable (speed). In the first and second tests, the results are similar to those obtained with the simulation, considering the controller based on kinematic model. In the third test, the results are similar to the simulation, with a good response to disturbances, the control implemented is based on the kinematic and dynamic model as presented.

Finally, control laws, and trajectory tracking based on kinematic and dynamic models were implemented successfully. Experimental results presented allows to verify that the objectives of the work were reached. It should be noted that this work marks the beginning of scientific robotics research in Bolivia.

References

[1] A. Al-Jarrah and M. Salah, "Trajectory tracking control of tracked vehicles considering nonlinearities due to slipping while skid-steering," SYSTEMS SCIENCE and CONTROL ENGINEERING: AN OPEN ACCESS JOURNAL, vol. 10, no. 1, pp. 887–898, 2022.

[2] K. Brockmann and D. Stefanovich, Hypersonic Boost-Glide Systems and Hypersonic Cruise Missiles. Stockholm International Peace Research Institute, 2022.

[3] V. P. Emanuele Bussagli, Francesco Giannelli and C. Viaggio, Drones and Land Forces. European Army Interoprability Centre, 2022.

[4] M. Fiede and J. Bałchanowski, "A mobile robot with omnidirectional tracks—design and experimental research," Applied Sciences, vol. 11, no. 11768, pp. 2–23, 2021.

[5] J. Iossaqui, J. Camino, and D. Zampieri, "Adaptive tracking control of tracked mobile robots with unknown slip parameter," in XVIII Congresso Brasileiro de Automatica 12 a 16 Setembro, 2010.

[6] J. G. Iosaqui, J. F. Camino, and D. E. Zampieri, "A nonlinear control design for tracked robots with longitudinal slip," IFAC Proceedings Volumes, vol. 44, no. 1, pp. 5932–5937, 2011.

[7] M. M. Emina PETROVIc, Sasa PAVLOVIc and B. RAJKOVIc, "Kinematic model and control of mobile robot for trajectory tracking," in ANNALS of Faculty Engineering Hunedoara–International Journal of Engineering, 2016.

[8] L. X. En Lu and Z. Tang, "Adaptive backstepping control of tracked robot running trajectory based on real-time slip parameter estimation," International Journal Agriculture and Biologic Engineering, vol. 13, no. 4, pp. 178–187, 2020.

[9] A. A. Kabanov, "Optimal control of mobile robot's trajectory movement," WSEAS Transactions on Systems and Control, vol. 9, no. 41, pp. 398–404, 2014.

[10] F. Triveño Vargas, "Modelado, simulación y control de sistemas mecatrónicos y aeroespaciales con matlab-simulink y mathematica," 2014.

[11] A. Kabanov, "Feedback linearized trajectory-tracking control of a mobile robot," in MATEC Web of Conferences, 2017.

[12] Z. A. K. Mohammad Hakim Ahmad Sidi, Khisbullah Hudha and N. H. Amer, "Modeling and path tracking control of a tracked mobile robot," in IEEE 14th International Colloquium on Signal Processing and its Applications, 2018.

[13] S. G. Tzafestas, Introduction to mobile robot control. Elsevier, 2013.

[14] H. A. Secchi, "Una introducción a los robots móviles," Instituto de Automática–INAUT. Universidade Nacional de San Juan–UNSJ–Argentina. Edição: Agosto de, 2008.

[15] B. Siciliano, L. Sciavicco, L. Villani, and G. Oriolo, Robotics: modelling, planning and control. Springer Science & Business Media, 2010.

[16] R. Fierro and F. L. Lewis, "Control of a nonholomic mobile robot: Backstepping kinematics into dynamics," Journal of robotic systems, vol. 14, no. 3, pp. 149–163, 1997.

[17] J. J. Craig, Introduction to robotics: mechanics and control, 3/E. Pearson Education India, 2009.

[18] A. O. Baturone, Robótica: manipuladores y robots móviles. Marcombo, 2005.

Manuscript 459 submitted to 2023 9th International Conference on Control, Decision and Information Technologies (CoDIT). Received March 14, 2023.