Genetic Programming Based Position Estimator and Control Model for Tracking Wheeled Robot Curvatures

Abeer M. Mahmoud

Computer Science dept., Faculty of Computer and Information Sciences,
Ain Shams University, Cairo, Egypt
abeer_f13@yahoo.com

Abstract

Wheeled mobile robots (WMRs) have a numerous applications and planetary of exploration tasks, all of which require building the suitable dynamic model of the robot. Also, the optimal control model that is intended to guide the robot to accomplish its mission is of a significant need. In addition, it is a certain that achieving a success in the robot controller mission requires an accurate estimation of the robot trajectory for setting the right control parameters and predicting the robot behavior in different situations. The overall WMR dynamics subject to skidding, wheel slip, regulation control and turning control are formulated and simulated in this paper for acommercial four wheel robot. In addition, the paper proposes a new genetic programming (GP) based control model for tracking curvature trajectory of a four wheels robot. The proposed model(GPCE) achieved a promising refinement in the robot of curvature estimation and simulation hence provided accurate control parameters for trajectory tracking.

Keywords: *Wheel robot, Simulation, Estimation, Genetic Programming*

1. Introduction

Wheeled mobile robots (WMRs) have a numerous applications and planetary of exploration tasks, all of which require building the suitable dynamic model. In addition, the optimal control that is intended to guide the robot to accomplish its main target is of a significant need. Certainly, achieving a success in the robot controller mission requires an accurate estimation of the robot trajectory for setting the right control parameters [1]. Usually, for simplicity purpose, the majority of studies developed a wheeled dynamic models under the assumption of nonslip and nonskid conditions. However, recently, the wheel slip becomes a significant factor to be included in the robot dynamic model especially when the smooth rolling be violated due to numerous circumstances such as rapid increase or decrease in acceleration or even rolling in unstructured environment. Instead, ignoring such factor certainly leads to facing some key problems as diverting from the desired curvatures or lack of stability during turning or spinning . Therefore, and in almost the last decade, the researchers in the domain have made a considerable contributions related to the control of WMRs with wheel slip [1-5].

Lin, et.al., [6], developed an adaptive critic anti-slip control design using dual heuristic programming and multi-layer perceptron neural networks. Where The road friction was considered as un-modeled dynamics. The critic structure enabled neural network learning by satisfying the Bellman equation so that the inclination of the action performance can be assessed to improve the control parameters. The paper results showed that the performance is significantly enhanced. Stonier, et.al,[7], presented an investigation of an omni-directional WMR model with slip. The motion equations of the nonlinear dynamics of traction considering wheel slip are derived using Euler-Lagrange formulation. The paper went through an explorations toward a comprehensive analysis of the dynamics for omniwheel.Ploeg, J., et.al, [8], proved that the slip effects caused by the tires can play an important role. The paper introduced both longitudinal and lateral tractions, which actually, have been approximated to be linearly dependent on longitudinal and lateral slip. In addition, a position controller based on feedback linearization were also presented using the so-called multicycle approach. Michalek, et.al, [9], considered a skid-slip effect for a WMR as disturbance on kinematic model, and proposed their estimation using Kalman filter with a compensation of a position tracking control. The proposed model assumed the skid slip effects solely on the kinematics level avoiding the need of modeling a complicated phenomenon of the wheels ground interaction. The results of the paper reveals substantial tracking quality improvement. Tian, and Sarkar, [10] proposed a new dynamics near-optimal autonomous pursuit evasion for non- holonomic wheeled mobile robot subject to wheel slip.

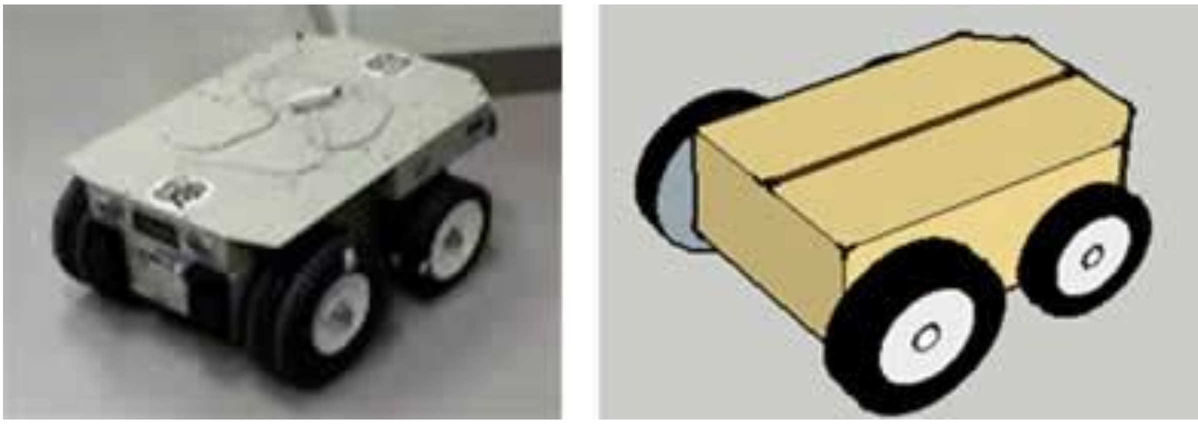

Fig. 1. GAIA-1arobot and simulated model

The author first showed how a feedback linearization controller can achieve capture with wheel slip, then improved the capture time by designing a new controller that maximizes lateral traction force to effect a sharper but stable turn. The simulation results show the efficacy of the proposed control approach.

On the other hand, the field of modeling and simulation has ripen recently due to much savage in time, effort as well as economic cost. But, one must put into consideration that, the simulation of robotics motion during rolling or turning has a complicated requirements especially when considering skidding or slippage [11-13]. The wheel skidding or slippage may happen due to soil or rough terrain interaction which in turn make it difficult to predict the exact motion of the vehicle on the basis of track velocities. The overall WMR dynamics subject to skidding, wheel slip, regulation control and turning control are simulated in this paper, where the dynamic model of skid steering mobile robot (SMRs) having decreased driving wheels either intentionally or unexpectedly. Based on the pervious formulation, The main contribution of this paper is proposing a new GP based position estimator and control model for improving curvature tracking of a wheeled robot. The proposed model (GPCC) achieved a promising refinement in curvature estimation and hence provided accurate control parameters for trajectory tracking. The rest of the paper is organized as follows. Section 2 introduces GAIA-Ia kinematic specification. Section 3 details the Synthesis of Kinematics Subject to Skidding & Slipping. Genetic Programming technique is briefed in Section 4. The simulation environments and computational results are in Sections 5. Section 6 concludes the paper.

2. *GAIA-1a*Mobile Robot& Preliminaries

This section introduces the necessary preliminaries for simulating the robot kinematics. In addition, it outlines the model of the commercial robot that have been used in this paper for validating the proposed approach estimation and simulation results. the necessary preliminaries are introduced in the following :

(a,b)	X and Y components of the robot center on the planes of $\lambda_o$and λ. (a_e, b_e) shows a small linear displacement
d_i, L	Translational wheel displacement, i.e., $\overline{P_i\, P_i'}$, Body size (body length; L_h, body width; L_w), respectively
Λ_ι	Expected rolling distance of the wheel within Δt
A_ι	Wheel tread area (width; t_{ai}, length; t_{bi})
R_ι	Wheel radius
$P_{i0,}$ P_i'	Initial wheel position (x_{i0}, y_{i0}, z_{i0}) on ζ_0 and Actual wheel position (x_i', y_i', z_i' i) on ζ after a small rotation
Ω_ι	Load at the wheel contact on ζ_0 (i= 1−5,7, t(total load))
P_{si}	Expected position (x_{si}, y_{si}, z_{si}) on ζ when wheel rolls small displacement
P_c , P_g	Geometric and mass center of the robot, respectively
A, η	Angular shift of a wheel from an initial direction, & virtual body axis extending straight forward
Φ	Twist angle of the front body part around η
$\theta_{r,}$ θ_s	Wheel rotation angle (small increment, Δθr), and Inclination angle of the ground surface
μ_i, ν_i	Friction coefficient (static; μ_{si}, dynamic; μ_{di}), Linear speed of a wheel movement
Ω_i,, λ_v	Angular velocity of i-th wheel (left; ω_l, right; ω_r), and Ratio of ω_l to ω_r (= νr/νl)

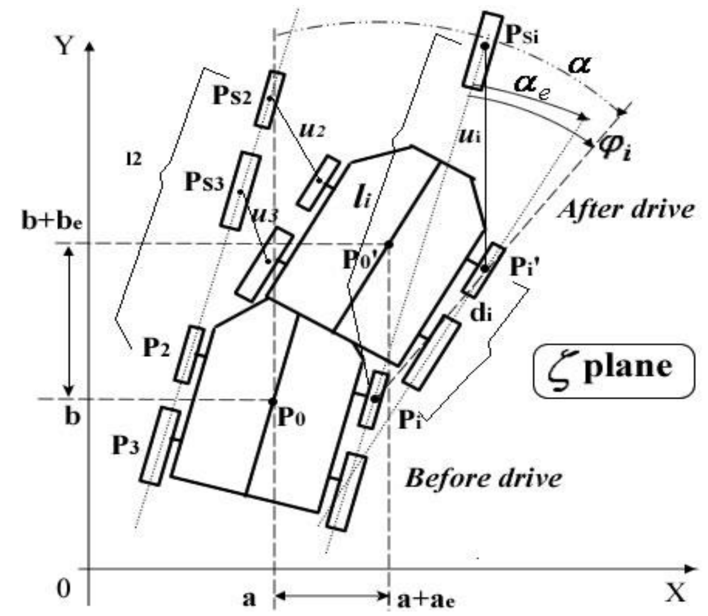

Fig. 2. Illustration of virtual distance U_i connecting P_{si} and P

The GAIA-1a mobile robot (see Figure 1), is commercially small four-wheeled mobile robots. The two front wheels are smaller than its two rear wheels. Its dimensions and specification are as follow

Wheel	:	front(solid); radius = 8.7cm, rear(nylon); radius = 10.8cm
Tread	:	front; ta = 7.5cm, tb = 3.0cm, rear; ta = 8.0cm, tb = 4.5cm
μ_i	:	p − tile; 0.44(front), 0.51(rear). wood; 0.56(front), 0.79(rear)
Driver	:	Vista board, Battery(24Ah, 12V), 100ppr, Max. speed; 2.8km/h
Sensor	:	Encoder; 100 ppr, Marker position; 32.5cm(Pmf), − 28cm(Pmb)
Lh	:	53cm(overall), 48cm(between axes)
Lw	:	36cm(overall), 31cm(between wheels)

3. Synthesis of Kinematics Subject to Skidding & Slipping

Since the robot trajectory cannot be expressed using a closed algebraic expression, an incremental change in the robot position caused by small change in the driving wheel angles could estimate the continuous path by continuous iteration. Now assume that the robot is on plane ζ_0. Also, assume each wheel rolls a distance $l_i$which is equal to $R_i\omega_i\Delta t$. Then, $P_{si}(x_{si}, y_{si}, z_{si})$ in the system Σ is combined with $P_{i0}(x_{i0}, y_{i0} + l_i, z_{i0})$ in the system Σ_0 (Fig.2). In fact, $l_i$is zero when the robot locks a wheel *i*from rolling. These relations are as given below:

$$\begin{bmatrix} x_{si} \\ y_{si} \\ z_{si} \\ 1 \end{bmatrix} = \text{Rot}(x, \theta_s)\text{Trans}(ab,0)\text{Rot}(z,-\alpha)\begin{bmatrix} x_{i0} \\ y_{i0} + l_i \\ z_{i0} \\ 1 \end{bmatrix} \tag{1}$$

Similarly, after each wheel rolls independently on the plane ζ, $P_i^{'}(x_i^{'}, y_i^{'}, z_i^{'})$ is combined with $P_{i0}(x_{i0}, y_{i0}, z_{i0})$ using the variables (a_e,b_e) and α_e. Moreover, $z_{i0}^{'} = 0$ holds for the same plane ζ_0. Accordingly,

$$x_i^{'} = x_{i0}\cos(\alpha+\alpha_e) + (y_{i0}+l_i)\sin(\alpha+\alpha_e) + a + a_e \tag{2}$$

$$y_i^{'} = \{-x_{i0}\sin(\alpha+\alpha_e) + (y_{i0}+l_i)\cos(\alpha+\alpha_e) + b + b_e\}\cos\theta_s - z_{i0}\sin\theta_s \tag{3}$$

$$z_i^{'} = \{-x_{i0}\sin(\alpha+\alpha_e) + (y_{i0}+l_i)\cos(\alpha+\alpha_e) + b + b_e\}\sin\theta_s - z_{i0}\cos\theta_s \tag{4}$$

Suppose one wheel is driven independently. Then, skidding and slipping occur while the robot moves. Therefore the wheel movement is composed both positional and angular displacements, which are mapped to *linear sliding* and *twist sliding*, respectively. In addition, drive power is needed to accomplish sliding, which is a combination of skidding and slipping and off course energy is necessary when the robot rolls up a slope. When rolling down a slope, the robot moves downward by consuming its gravitational potential energy. Therefore, the virtual work for sliding appears in the form of energy for twist sliding, linear sliding and any change in gravitational potential energy [14]. Let us suppose that E_1, E_2 and E_3 denote each of the three energy costs, respectively. In addition suppose that each wheel in contact with the ground with a rectangular tread, and that the pressure in the area is unique, irrespective of the contact position. That is, the weight distribution on wheel i→$W_i = t_{ai}t_{bi}\,\tau$, where τ is a pressure in units of pressure. Using α_e, we obtain two expressions:

$$E_{1i} = \int_0^{\alpha_i} \int_{-t_b/2}^{t_b/2} \int_{-ta/2}^{ta/2} \mu_{di} \tau (x^2 + y^2) dxdyd\alpha$$

$$= \mu_{di} |\alpha_e| w_i \cos\theta_s (t_{ai}^2 + t_{bi}^2)/12 \quad (5)$$

$$E_{2i} = \mu_{di} U_i c |\upsilon_i| W_i \cos\theta_s \quad (6)$$

It follows that

$$U_i c |\upsilon| = U_i c \left| U_i / \Delta t \right| = U_i^2 \, c / \Delta t \equiv K U_i^2 \quad (7)$$

Where

$$U_i^2 = (x_i^{'} - x_{si})^2 + (y_i^{'} - y_{si})^2 + (z_i^{'} - z_{si})^2$$

$$= \begin{matrix} a_e^2 + C_{1i} a_e \cos\alpha_e + C_{2i} a_e \sin\alpha_e + C_{3i} a_e + b_e^2 + C_{4i} b_e \cos\alpha_e + \\ C_{5i} b_e \sin\alpha_e + C_{6i} b_e + C_{7i} \cos\alpha_e + C_{8i} \sin\alpha_e + C_{9i} \end{matrix} \quad (8)$$

The constants $C_{1i} - C_{9i}$ are defined as

$C_{1i} = 2(x_{i0} \cos \alpha + y_{i0} \sin \alpha)$ $C_{2i} = -2(x_{i0} \sin \alpha - y_{i0} \cos \alpha)$

$C_{3i} = -2 \{ x_{i0} \cos \alpha + (y_{i0} + \_i) \sin \alpha \}$ $C_{4i} = C_{2i}$ $C_{5i} = -C_{1i}$

$C_{6i} = -C_{2i} - 2 l_i \cos \alpha$ $C_{7i} = (C_{1i} C_{3i} + C_{2i} C_{6i}) / 2$

$C_{8i} = 2 x_{i0} l_i$ $C_{9i} = 2(x_{i0}^2 + y_{i0}^2) + l_i(2 y_{i0} + l i)$

Using another constant K simplifies expression (10) as follows:

$$E_{2i} = W_i \mu_{di} K U_i^2 \cos \theta_s \quad (9)$$

So far in our analysis, U_i^2 is assumed to be zero when the *i*-th wheel has no power. This free wheel is supposed to roll a distance $l_j (j \neq i)$ that is generated by a neighboring powered wheel since the robot maintains its body frame nearly rigid except while twisting. However, there is some force feedback caused by the rotational friction around each wheel axis in an actual machine. Therefore, we assume that U_i^2 has a certain value proportional to the friction force, and express U_i^2 as a small value by defining a coefficient δi that is nearly equal to zero. For a powered wheel, $\delta i$is equal to 1. This consideration generalizes equation (13) as follows:

$$E_{2i} = \delta_i W_i \mu_{di} K U_i^2 \cos \theta_s \quad (10)$$

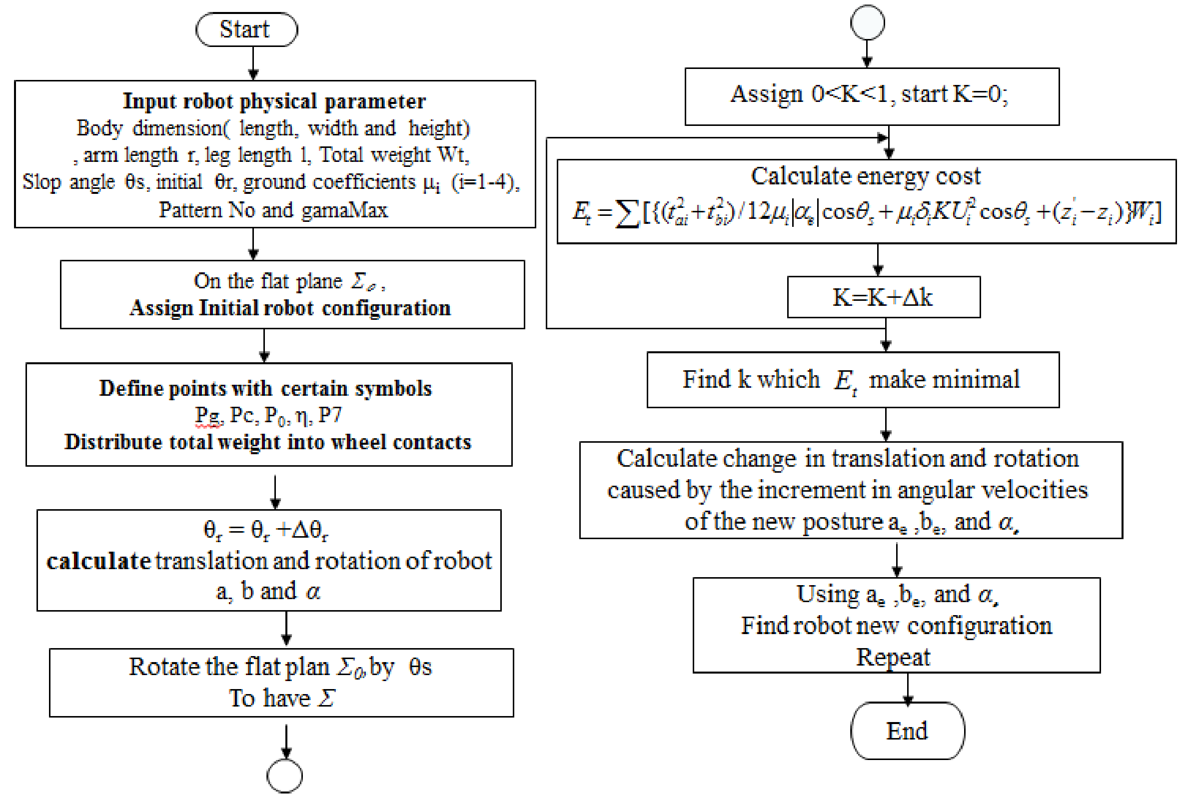

Fig.3. A Flow chart of building robot dynamic model subject to skidding &slipping

The exact value of μ_{di} is difficult to predict because it is subject to constant change in an actual environment. However, assuming there is a certain constant value, say e, then μ_{di} in equations (5), (6), (9) and (10) can be expressed as in [12], such that

$$\mu_{di} = (1 - e\, v_i)\, \mu_{si} \qquad (11)$$

In particular, v_i as well as d_i are both small quantities in our analysis. Thus, μ_{di} is assumed to be a smaller constant than that of μ_{si}. This implies that a faster motion will be estimated using the small value of μ_i. Henceforth we use μ_i in place of μ_{di} without the risk of underestimating motion. Concerning the gravitational potential energy, it is evaluated by the following expression:

$$E_{3i} = Wi(z'_i - \zeta_i) = W_i \sin\theta_s (C_{2i}\cos\alpha_e - C_{1i}\sin\alpha_e + 2b_e - C_{2i})/2 \qquad (12)$$

Finally, we can sum each energy cost to yield an expression for the total cost.

$$E_t = \sum_{i=1}^{4} \{E_{1i} + E_{2i} + E_{3i}\} = \sum[\{(t_{ai}^2 + t_{bi}^2)/12\mu_i|\alpha_e|\cos\theta_s + \mu_i\delta_i K U_i^2 \cos\theta_s + (z'_i - z_i)\}W_i] \qquad (13)$$

The robot movement can be predicted by solving for the unknown parameters in the previous equation to reach translation and rotation of the robot postures (a_e, b_e) and α_e. Thus, we differentiate (13) by each of these parameters since, based on the principle of virtual work, (13) is minimized when there is no energy loss except for skidding and slipping. The next robot position is introduced repeatedly by replacing the old parameters (a, b) and α with the new ones ($a+a_e$,$b+b_e$) and $\alpha+\alpha_e$ to connect position histories in order to form a continuous trajectory.Fig. 3 shows a flowchart for building robot dynamic model subject to skidding &slipping where The analysis of total energy consumption with respect to robot instance positions and next position proposed a good estimation for motion control parameter on the coordinate system for the simulator.

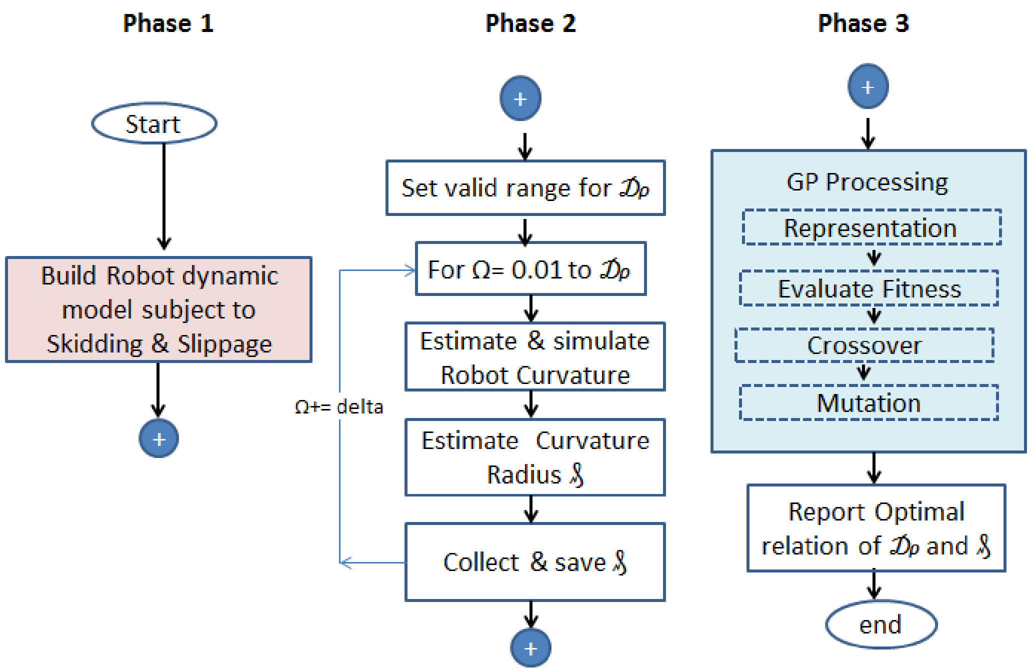

Fig. 4. The mainPhases of the proposed wheeled curvature estimator GPCE

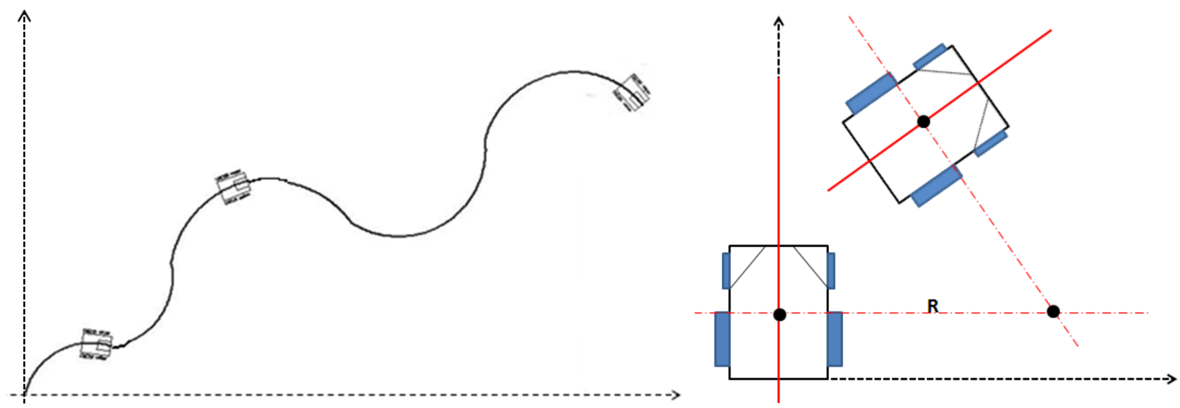

Fig. 5. (a) The simulation of robot curvature (b) methodology for calculating curvature radius between any two transitional robot curvature postures

4. The Proposed (GPCE)

When the chromosome encodes an entire program or function itself this is called genetic programming (GP). GP is ametaheuristic domain-independent method that genetically breeds a population of computer programs to solve a problem. It iteratively transforms a population of computer programs into a new generation of programs by applying analogs of naturally occurring genetic operations (crossover, mutation, reproduction, …). In addition, implementing the GP for a selected problem domain requires specifying the set of primitive functions, a fitness measure for each individuals, setting certain parameters for controlling the run, and the termination constrains. Figure 4 illustrates the relationship between the genetic programming and the proposed curvature estimator. Actually, the proposed model go through three main phases as depicted in the figure. Phase 1, the dynamic model is developed subject to skidding and slippagesee Fig.5 (b). In phase 2, the model generates curvature data relative to the relation between the right and left robot velocity difference $\mathcal{D}_p$ and the corresponding curvature radius $\mathcal{S}$see Fig.5 (b) . In phase 3, the GP receive all the generated data from the simulator regarding curvature position estimation, where it start to decode the genotype considering random sample of the generated data from phase 2. In the following, the proposed GPCE operator implementation and the setting of operator parameters.

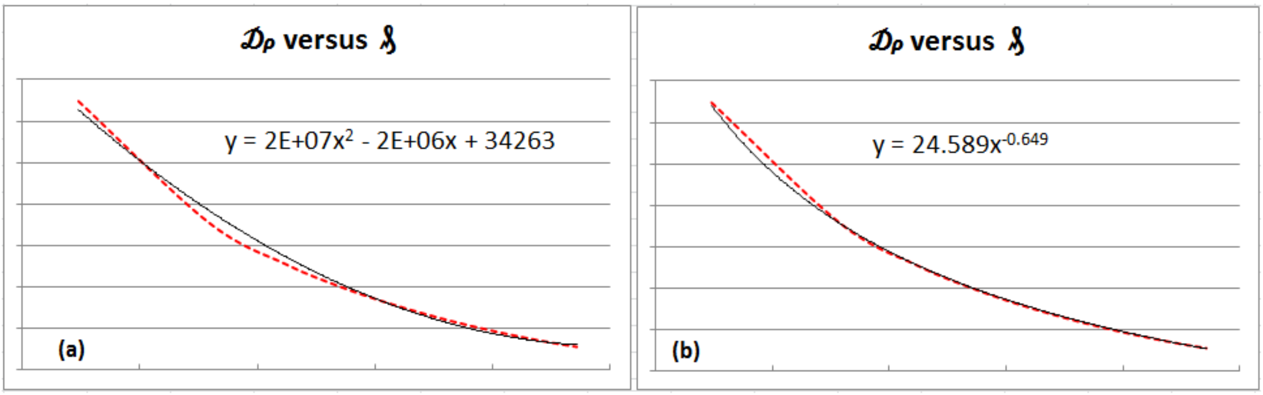

Fig. 6. Estimation trend line of $\mathcal{D}_p$ versus $\mathcal{S}$ using (a) polynomial function of order =2 (b) power function

4.1 GP Operators implementation

The Chromosome encode consists of the estimated relation of $\mathcal{D}_p$ versus $\mathcal{S}$ **,** where this relation is appropriately concluded aspower function of the form $y = ax^b$ after investigating some other trend-lines eg. polynomial function of order =2 (see Fig. 6). Based on that the initial population of GP is constructed by random sampling of the generated data from dynamic model of the robot subject to skidding and slippage using power function in the form $\mathcal{S} = a\,\mathcal{D}_p{}^{b}$.

The fitness function is the primary mechanism that is responsible for determining which individual is selected for reproduction . In the proposed GPCE, the fitness function is implemented as the variance between the original data and estimated one. The evaluation for each chromosome or solution is based on the root mean square error which has been calculated for all real data set.

The roulette wheel selection method was implemented for chromosomes reproduction. Crossover and Mutation are settled by the previous selection method, and then combined applying a single-cross-point. These produce two offspring chromosomes with different equation parameters set values. Additionally, an adjustable percentage of each generation is mutated randomly by adding or subtracting random numbers to the value of equation parameters.

4.2 GP Parameter Setup

The most important control parameter is the population size. In practice, the user may choose a population size that will produce a reasonably large number of generations in the amount of computer time we are willing to devote to a problem . Other control parameters include the probabilities of performing the genetic operations, the maximum size for programs, and other details of the run. The Elitism strategy is used to ensure that the current best individual survives to a next generation. The number of GP generation is set between 20-30, Number of individuals= 50, Crossover probability0.5-0.7, Mutation probability = 0.3.

index	cetaR	cetaL	Vr	Vl	D_p	S
0	14.6912	16.1603	1.44444	2.2	0.0191986	319.868
1	14.6912	16.3073	1.44444	2.22	0.0193732	318.063
2	14.6912	16.4542	1.44444	2.24	0.0195477	316.279
3	14.6912	16.6011	1.44444	2.26	0.0197222	314.517
4	14.6912	16.748	1.44444	2.28	0.0198968	312.777
5	14.6912	16.8949	1.44444	2.3	0.0200713	311.06
6	14.6912	17.0418	1.44444	2.32	0.0202458	309.362
7	14.6912	17.1887	1.44444	2.34	0.0204204	307.684
8	14.6912	17.3356	1.44444	2.36	0.0205949	306.028
9	14.6912	17.4826	1.44444	2.38	0.0207694	304.391
10	14.6912	17.6295	1.44444	2.4	0.0209439	302.774

Fig. 7.Sample of the collected data of the relation between D_p and S

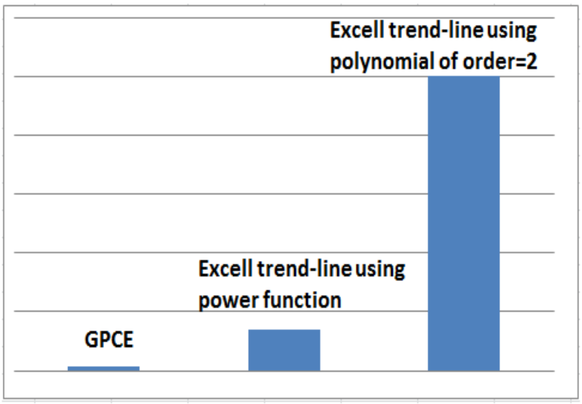

Fig. 4. Generated error

Table 1: sample of GPCE Initial population

Chrom No	Equation	error	No	Equation	error
1	$\mathcal{S} = 1.1459\ \mathcal{D}\rho^{-2.09931}$	3.55894e-006	6	$\mathcal{S} = 1.13767\ \mathcal{D}\rho^{-2.13121}$	2.7914e-006
2	$\mathcal{S} = 1.13037\ \mathcal{D}\rho^{-2.16235}$	2.20188e-006	7	$\mathcal{S} = 1.13767\ \mathcal{D}\rho^{-2.13121}$	2.7914e-006
3	$\mathcal{S} = 1.1459\ \mathcal{D}\rho^{-2.09931}$	3.55894e-006	8	$\mathcal{S} = 1.12385\ \mathcal{D}\rho^{-2.19283}$	1.74666e-006
4	$\mathcal{S} = 1.12385\ \mathcal{D}\rho^{-2.19283}$	1.74666e-006	9	$\mathcal{S} = 1.12385\ \mathcal{D}\rho^{-2.19283}$	1.74666e-006
5	$\mathcal{S} = 1.13037\mathcal{D}\rho^{-2.16235}$	2.20188e-006	10	$\mathcal{S} = 1.13037\ \mathcal{D}\rho^{-2.16235}$	2.20188e-006

5. Computational Results

The proposed model, generated the robot data for the GP to estimate the relation between $\mathcal{D}\rho$ and $\mathcal{S}$, considering slippage and skidding previously presented. Let us suppose that the robot turns with $\mathcal{D}\rho$= 0.1 and increased by delta $\mathcal{D}\rho$= 0.01. while $\mathcal{D}\rho$ is in the safe range of difference in velocities between the two left and right wheels sides of the robot, the robot moves to turn in a curvature. Then the $\mathcal{S}$ is calculated and recorded.Now, we have curvature data recorded as the sample shown in Fig. 7. The collected data of robot curvature was estimated using two excel trend-line functions. These are (1) polynomial function of order=2 and a power function, see Eq. (14) and (15) respectively. Obviously the power function is more appropriate for estimating data.

$$\mathcal{S} = 2\text{E} + 07\mathcal{D}\rho^2 - 2\text{E} + 06\mathcal{D}\rho + 34263 \tag{14}$$

$$\mathcal{S} = 24.589\ \mathcal{D}\rho^{-0.649} \tag{15}$$

Let suppose that the data is randomly sampled and every sub data set is separately estimated using a power function estimation similar to Eq.(15), see table 1. For a sample of the encoded initial genetic programming population where a, b are two constants for determining accurate relationbetween $\mathcal{D}\rho$ and $\mathcal{S}$. The genetic programmingstarts by taking the initial population and applying evaluation,crossover and mutation for number of generation until noenhancement for the calculated error for two consequent generations. The proposed GP concludes the final relation as in Eq.16.Figure 8 shows a comparison of computational results in terms ofthe generated error of equations (14-66). Obviously the estimated equation (126) recoded less errors and hence achieved best results for the robot curvature control.

$$\mathcal{S} = 1.13037\ \mathcal{D}\rho^{-2.16235} \tag{16}$$

6. Conclusion

Wheeled mobile robots (WMRs) have a numerous applications and planetary of exploration tasks, all of which require building the suitable dynamic model of the robot. The optimal control model that is intended to guide the robot to accomplish its mission is of a significant need. In addition, it is a certain that achieving a success in the robot controller mission requires an accurate estimation of the robot trajectory for setting the right control parameters and predicting the robot behavior in differentsituations. The overall WMR dynamics subject to skidding, wheel slip, regulation control and turning control are formulated and simulated in this paper for a commercial four wheel robot. The paper presented a trajectory estimation model as an extension of our previously published trajectory estimation model to apply for robots facing skidding or slippage circumstances. In addition, The paper, proposed a new genetic programming (GP) based control model for tracking curvature trajectory of a four wheels robot. The proposed model (GPCE) achieved a promising refinement in the simulation of robot curvatures and hence provided accurate control parameters for trajectory tracking.

References

[1]. L.Caracciolo, A.D. Luca and S.Iannitti, "Trajectory Tracking Control of a Four-Wheel Differentially Driven Mobile Robot", Proc. of IEEE Int.Conf. on Robotics and Automation, pp.2632-2638, 1999.

[2]. A.Shinichi, K.Hiroshi and Sh.Masatake, "One-Hand Drive-Type Power-Assisted Wheelchair with a Direction Control Device using Pneumatic Pressure", J. of Advanced Robotics", vol.16, no 8, pp. 773-784, 2002.

[3]. Kanayama, Y and et.al., “A Stable Tracking Control Method for an Autonomous Mobile Robot", Proc. of IEEE Int. Conf. on Robotics and Automation, vol. 1, pp. 384-389, 1990.

[4]. M.Lauri, Y.Piquet, and R.Siegwart, “OCTOPUS: An Autonomous Wheeled Climbing Robot", Proc. 5th Int. Conf. on CLAWAR, pp.315-322, 2002.

[5]. R.Morales, and et.al, “Kinematic Model of a New Staircase Climbing Wheelchair and its Experimental Validation”, J. of Robotics Research, vol. 25, no.9, pp. 825-841, 2006.

[6]. Lin, W.-S., Chang, L.-H., Yang, P.-C.: Adaptive critic anti-slip control of wheeled autonomous robot. Control Theory Appl. IET. vol.1, no .1, pp.51–57, 2007.

[7]. Stonier, D., Se, H.C., Sung-Lok, C., Kuppuswamy, N.S., Jong-Hwan, K.: Nonlinear slip dynamics for anomni-wheel mobile robot platform. In: IEEE Int. Conf. on Robotics and Automation, pp. 2367–2372 , 2007.

[8]. Ploeg, J., Schouten, H.E., Nijmeijer, H.: Control design for a mobile robot including tire behavior. In: IEEE Intelligent Vehicles Symposium, pp. 240–245, 2008.

[9]. Michalek, M.M., Dutkiewicz, P., Kielczewski, M., Pazderski, D.:Vector-field-orientation tracking control for a mobile vehicle disturbed by the skid-slip phenomena. J. Intell. Robot. Syst.vol. 59, no.304, 341–365, 2010.

[10]. Tian, Y., and Sarkar, N., ” Near-optimal autonomous pursuit evasion for nonholonomic wheeled mobile robot subjectto wheel slip” . In: IEEE International Conference on Robotics and Automation, pp. 4946–4951,2010.

[11]. Wang, D., Low, C.B.: Modeling and analysis of skidding and slipping in wheeledmobile robots: Control design perspective. J. IEEE Trans. on Automatic Control, vol.24, no.3, 676–687, 2008.

[12]. Low, C.B., Wang, D.: Integrated estimation for wheeled mobile robot posture, velocities,and wheel skidding perturbations. In: IROS 2008 Workshop on Modeling,Estimation, Path Planning and Control of All Terrain Mobile Robots, pp. 20–30, 2008.

[13]. E. Drumwright and D. Shell, “An evaluation of methods for modeling contact in multibody simulation,” in IEEE Int. Conf. on Robotics and Automation, pp. 1695–1701.2011.

[14]. Hadwich, V., Pfeiffer, F.: The principle of virtual work in mechanical nad electromechanical systems. Archive of Applied Mechanics 65, 390–400 , 1995.