Journal of Intelligent & Robotic Systems (2019) 95:999–1019
https://doi.org/10.1007/s10846-018-0932-3

A New Contact Angle Detection Method for Dynamics Estimation of a UGV Subject to Slipping in Rough-Terrain

Saeed Ebrahimi[1] · **Arman Mardani**[1]

Received: 24 January 2018 / Accepted: 6 September 2018 / Published online: 13 September 201 8

Abstract

In this paper, a new single-output resistive sensor is proposed to enhance the dynamics estimation, slip elimination, stability extraction and surface scanning of a ground UGV (wheeled-robot) in 2D and 3D spaces. The new sensor is based on the total resistance of a circuit which contains a continuous resistive belt with high accuracy. The Lagrange method is implemented to derive the stick-slip dynamics in both 2D and 3D spaces. Furthermore, the kinematics is solved using the Newton-Raphson approach. The slipping characteristics of the robot with and without the new sensor are firstly shown in 2D space. To demonstrate the abilities of the sensor in the real applications, the dynamic simulation is further extended to 3D space. A real time torque optimization aided by the new sensor is applied to the dynamics of the robot to eliminate slip during locomotion. Moreover, a stability measure is introduced and the real time stability margins are extracted during missions. The sensor empowers the robot to scan the surface and consequently, extract the main properties of the surface. The simulation results obtained for different case studies prove the ability of the new sensor in performing the above mentioned tasks.

Keywords Field robots · Torque optimization · Resistive belt sensor · Real time stability · Surface scanning

1 Introduction

The concept of field robotics has found great attention in a wide range of engineering aspects such as rescuing [1], exploring [2, 3] mining harvesting and adjustable exploration platforms [4]. Real applications involve missions on uneven surfaces including obstacles and hazards [5, 6], holes and ramps and danger of slip. Indeed, finding the surface characteristics becomes essential for these robots to overcome severe conditions. By implementing ordinary sensors, some researches focus mainly on the control and localization process on the rough surfaces [7]. The stereo vision techniques were also used for the wheel contact angle [8]. These control algorithms not only are implemented in mobile robots, but also control the manipulation processes [9]. On the other hand, force sensors improve the performance of the interactive missions by implementing the force effects [10]. Using force sensors empowers the systems to act in the presence of minimum error of the dynamic states estimation. Unfortunately, these sensors significantly increase the computation cost. In addition, these sensors are expensive to be implemented everywhere. The main issue is that the force sensors are usually point detectors which are unable to detect a regional force distribution. In such situations, usually an array or matrix of sensors are needed. It means that numerous expensive force sensors have to be implemented. When dynamics of a system can be known in a control process only by using kinetic sensors such as accelerometers and vision sensors, the force sensors can be eliminated and consequently, the total cost is reduced. In a recent research, a new shield sensor embedded in the robotic wheels are implemented to detect the exact margin of stability and to extract the complete kinematics of the wheel-surface interaction [11].

A main issue of the rough surface locomotion is to eliminate slips during motion. This problem encourages researchers to find the required torque for slip reduction of the wheeled mobile robots. Some instruments and algorithms are required to overcome slipping and other environmental issues. The conducted research in [12] highlighted the necessity of modeling and control of mechanical contacts which can also be considered in

✉ Saeed Ebrahimi
ebrahimi@yazd.ac.ir

Arman Mardani
Yovas1369@gmail.com

[1] Department of Mechanical Engineering, Yazd University, Yazd, Iran

applications such as mobile robots moving on rough surfaces. For this purpose, all parameters required to solve kinematics and dynamics such as contact points have to be detected. The analysis of mobile robot motion based on the slip dynamics and the wheel-ground interaction require estimation of the slip margins [13] and friction forces [14]. The torque of motors can be optimized to minimize the undesired wheel slip [15]. One of the essential parameters in slip reduction is the angle of contact. The work in [16] presents a method for wheel-ground contact angle measurement and a traction control strategy for minimizing slip in rough terrain. However, in addition to the contact angle, a coherent view of the kinematics and dynamics of slip is required to reduce slip. The authors of [17, 18] considered kinematic and dynamic modeling method for wheeled mobile robots with slip based on physical principles in order to reduce slips. Slip dynamics includes longitudinal and lateral slips. The structural issues such as number of wheels or angle of the steering axis can cause unrequired lateral slip which can be eliminated by some mechanical modifications [19]. The control algorithms can also be useful to tackle the modeling inadequacy that arises when slip is neglected by including both longitudinal and lateral slip dynamics into the overall dynamics of the wheeled mobile robots [20]. The effect of slip reduction (and especially the lateral slip) can be of great importance in the stability analysis of mobile wheeled robots. The authors of [21] showed that the slip reduction can be useful not only in the slipping issue but also in the stability and overturning. On the other hand, some researches showed that the slip compensation can increase the performance of the trajectory tracking [22, 23].

The efforts in reducing slip need to sense the environment of the contact completely. Without sensing the contact point or contact force distribution, the torque optimization can destroy the natural stability or increase the slip moments. Some of the resent researches considered sensing interaction between the mobile robot and the trail [24]. The usage of FSR sensors or touch sensors has to be widely extended to the exploration rovers. A sensitive wheel can make an alert about the risk of moving on some surfaces [25]. The solution is to use a Built-in Force Sensor Array (BFSA) wheel in order to determine the normal stress on a wheel [26]. The BFSA advanced usage yields development of an in-wheel sensor system for accurate measurement of the wheel terrain interaction characteristics [27]. The accuracy of the contact detection in the BFSA systems depends on the number of segments implemented in the sensor array. On the other hand, increasing the number of segments yields high-cost wheels. The segmented array cannot detect the exact position of the wheel contact position which is the most essential parameter to find dynamical parameters. The main question arises is how the cost of contact detection can be reduced. How a new sensor can be innovated in order to simultaneously reduce the cost and keep the result accurate?

Besides the ordinary sensors implementation in unknown environments [28], the contact sensors can solve a significant part of stability, lateral and longitudinal slip issues. Innovation of new sensors can considerably simplify achievement of this goal. Displacement sensors, the distance detectors such as ultrasonic and laser sensors [29, 30] and accelerometers are examples of the mobile robot sensors. As another type of sensors, tactile sensors detect the robot-environment contact [31, 32]. In hard skins method (bumper-based method), skins consist of sensors (three or four sensors per panel) covered by a hard shell in the shape of the robot body. This type is usually used in industrial manipulators to avoid hard contacts [33]. In some researches, Force Sensing Resistors (FSR) are utilized under a flexible skin [34]. ”Robovie” series of humanoid robots have used such sensors [35]. Multi-modal tactile sensors embedded in silicone make the Huggable robot interactive. Some applications of FSR rely on a technology which uses variable resistance to measure pressure applied to different points of foot. This technology is very reliable and can be incorporated into thin and flexible applications [36, 37]. The extra number of input terminals of these sensors requires a powerful data acquisition (DAQ) card and a fast computer. Touch screens consist of a group of resistors or capacitors. The idea of a sensitive wheel aided by touch sensors which can sense each obstacle and contact point at the same time is a promising and novel concept which has not been paid enough attention before. However, there are some limitations in using cheap sensors which still can be strong and easily to be connected to DAQ cards. The resistive touch path is one of the common types of touch screens which consists of a group of resistors assembled in a network.

Some important researches considering flat terrains found out that the rovers with uncertain systems can be perfectly controlled and robustly stabilized by control algorithms without a new sensor [38, 39]. Accordingly, the finite-time tracking control problem has been discussed for extended nonholonomic systems with parametric uncertainty, unmolded nonlinear dynamics and external uncertain time-varying disturbances implementing ordinary sensors and data [40]. In these researches, the level of model uncertainty and the amplitude of external disturbances are restricted. On the other hand, on the rough terrains, using new control algorithms can significantly improve the efficiency of locomotion as it is effective on flat surfaces [38–40]. Using the new sensors empowers detection methods to improve the efficiency of these control processes and

other methods on rough terrains by forming more reliable dynamic model of the UGV. The present research emphasizes that the complete kinematics of a mobile robot facing rough surfaces can extract the complete dynamic states without force sensors. This paper combines the concept of resistive touch paths with a sensitive wheel to make the robotic wheels sensitive by means of a novel resistive belt sensor. This sensor is composed of a resistive belt, a conductive metal belt and a direct voltage source. All these parts will be grouped together as a simple resistive single-output circuit. This new sensor can be embedded in some mobile robots whose duty is to simultaneously sense the location and map the environment (SLAM) [41]. The novelty and major contribution of the present study is to embed a single-output resistive belt in the UGV wheels which is totally different from the existent concepts such as the vision-based methods, BFSAs, flexi forces and contact switches. This idea has some main advantages including reduction in the number of sensor outputs, continuous contact angle detection as compared with BFSA arrays, efficient computation and detection of the contact angle for kinematics and dynamics evaluation of the UGVs, and low cost of production. The next part of this paper focuses on expanding and defining the new idea. The mathematic base of this idea is explored in the following sections. The remainder of the paper explains the abilities of the new single-output sensor in detecting multipoint contact, surface scanning, dynamic estimation, slip elimination and extracting the stability margins.

2 Definition of the New Idea

The advanced rovers and ground UGVs are equipped by BFSA sensors in order to detect wheel-terrain interaction [10, 27]. This detection method includes an array of FSR

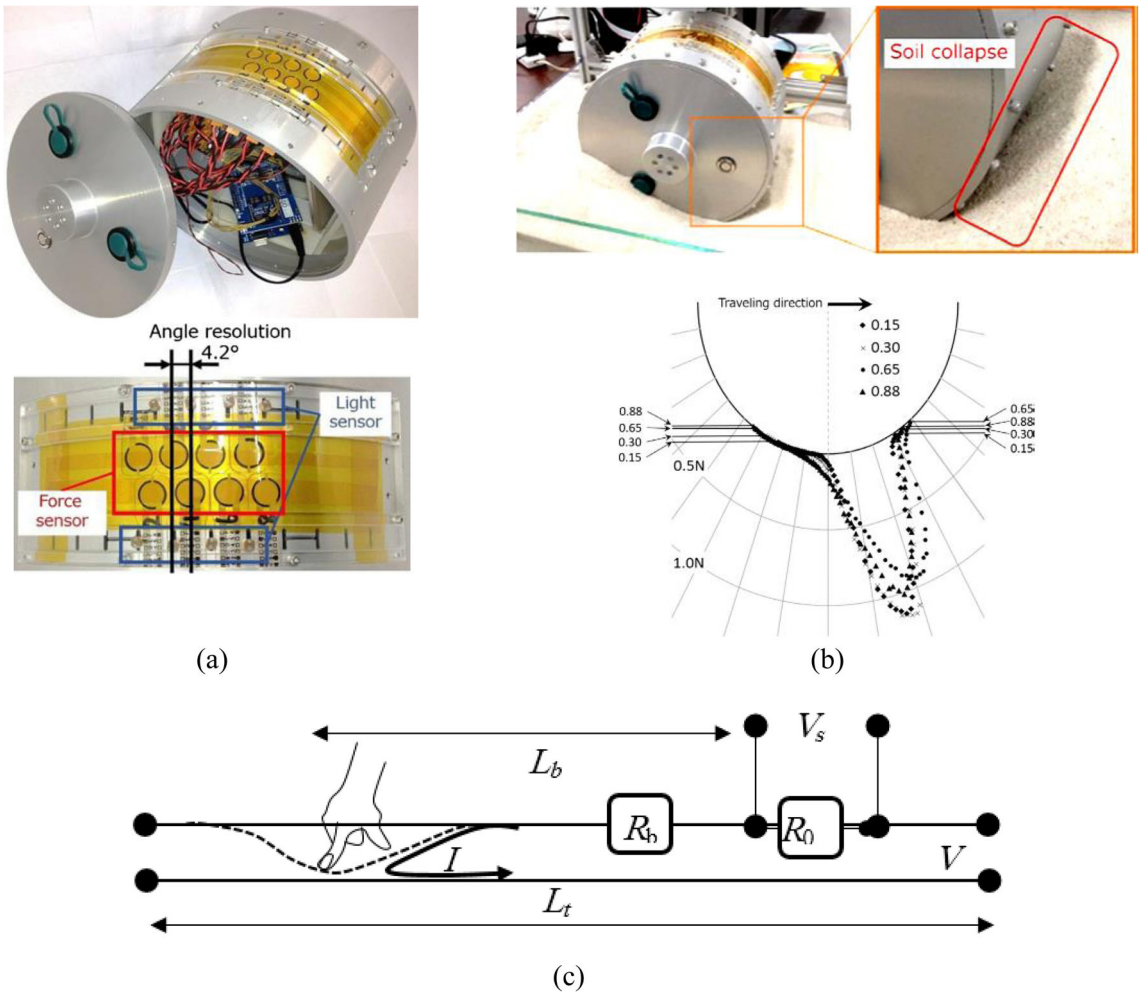


Fig. 1 Continuous resistive belt and FBSA sensor. **a** The FBSA sensor [27], **b** the force distribution beneath the wheel [27], **c** The concept of continuous resistive belt

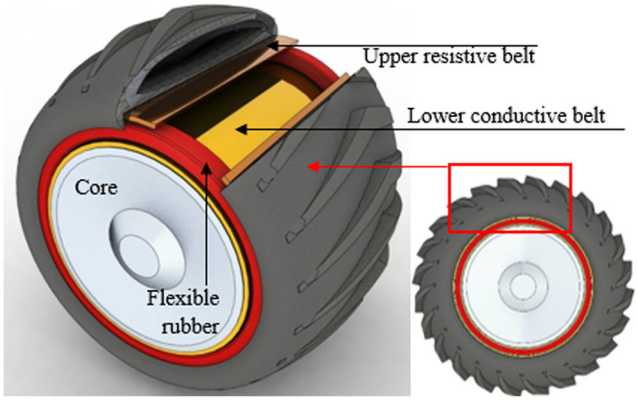


Fig. 2 Continuous resistive belt on a wheel

sensors embedded in the wheels (Fig. 1a). The discrete result of the force distribution beneath the wheel in Fig. 1b implies the accuracy of this force estimation method [27].

As it was pointed out in the introduction, the high cost of computation and detection of this method for a wide area and the discrete output yield the current research idea. A continuous resistive belt which yields a resistive belt detecting only one-point or two-point contact situations is the main idea of this paper. It can be embedded under the exterior rubber of the wheel, see Fig. 2. High accuracy can be achieved by using a new continuous resistive belt. Figure 1c shows the promoted continuous resistive belt in which a person pushes a point of the resistive belt. The contact of the upper resistive belt with the lower conductive belt closes the circuit. Parameter R_b is the resistance of that part of the upper resistive belt which conveys current. Therefore, this resistance is directly calculated from the length L_b of this part of the upper resistive belt. By considering the total resistance R_t and total length L_t, the resistive belt produces a resistance R_b as $R_b = R_t L_t^{-1} L_b$. The voltage V_s is then obtained from $V_s = V(R_t L_t^{-1} L_b + R_0)^{-1} R_0$.

3 Kinematics

There are some approaches for modeling the interaction between wheel and terrain based on the multibody dynamics formulations [42]. The contact points of the wheel and surface have to be detected during a virtual locomotion. The iterative Newton-Raphson method is implemented to solve kinematics. Figure 3 illustrates a planar model of a four-wheeled mobile robot equipped by an accelerometer to detect the coordinates of the CG of the robot. The objective of the kinematic solver is to find the position of contact points as well as some essential parameters including the relative angles of the front and rear wheels with respect to the robot body (ϑ_f and ϑ_r), and the parameters y_b and θ_b. The parameters $L_b$and R_w are respectively the half of the length of the robot and the wheel radius. The parameters θ_i and β_i $(i = r, f)$ are respectively the encoder angle and the angle detected by the continuous resistive belt sensor of the wheel i. Furthermore, the parameters $\mathbf{N}_i$ and $\mathbf{T}_i$ $(i = r, f)$ denote the normal and tangent vectors of the wheel i, respectively. The coordinate system $X_j Y_j$ of accelerometer is fixed to the CG of the robot body. The accelerometer is aligned with X_j direction. Therefore, $\mathbf{a}_j$ shows the absolute acceleration vector of the robot body expressed in the body-fixed coordinate frame $X_b Y_b$. To solve the kinematics, the

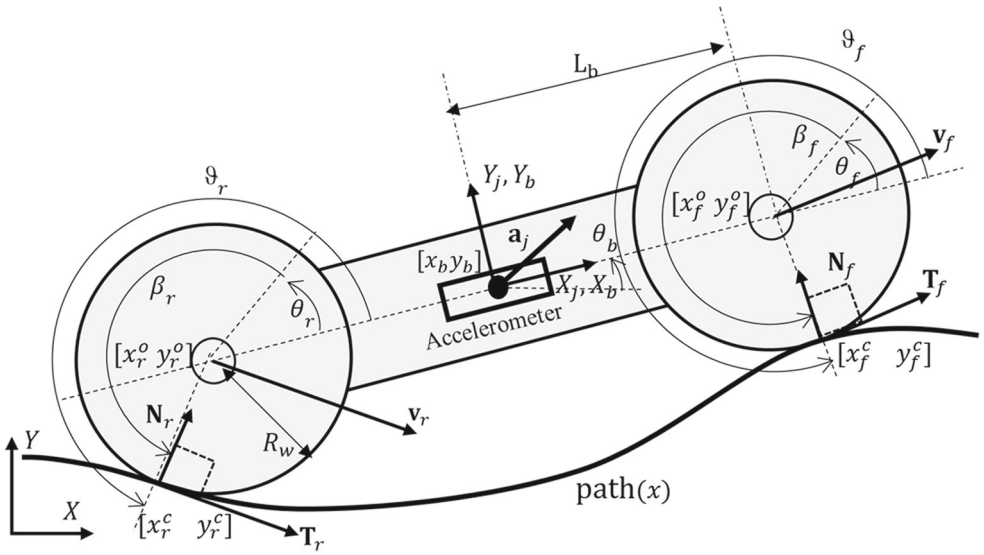


Fig. 3 Kinematics of a planar model of a four-wheeled mobile robot

first step is to obtain the unknown parameters (ϑ_f, ϑ_r, y_b and θ_b) from the accelerometer and gyroscope outputs ($\mathbf{a}_j$, θ_b).

In order to express the velocity vector $\mathbf{V}_b^o$ of the robot body in the inertial coordinate frame *XY*, the acceleration $\mathbf{a}_j$ is integrated to find the velocity vector $\mathbf{V}_j$ in the body-fixed coordinate frame yielding

$$\mathbf{V}_b^o = \mathbf{R}_{\theta_b}\mathbf{V}_j, \qquad \mathbf{V}_j = \int_0^t \mathbf{a}_j dt, \qquad \mathbf{V}_j = [v_j^x \;\; v_j^y]^{\mathrm{T}} \quad (1)$$

where $\mathbf{R}_{\theta_b}$ denotes the rotation matrix of the body-fixed coordinate frame. The acceleration of the robot body in the inertial coordinate frame can be found by differentiating (2) as

$$\mathbf{a}_b^o = \frac{d}{dt}(\mathbf{R}_{\theta_b})\mathbf{V}_j + \mathbf{R}_{\theta_b}\mathbf{a}_j, \quad \mathbf{a}_j = [a_j^x \;\; a_j^y]^{\mathrm{T}} \quad (2)$$

which yields the following expanded form

$$\mathbf{a}_b^o = \begin{bmatrix} -\sin(\theta_b)\, v_j^x\dot{\theta}_b - \cos(\theta_b)\, v_j^y\dot{\theta}_b + \cos(\theta_b)\, a_j^x - \sin(\theta_b)\, a_j^y \\ \cos(\theta_b)\, v_j^x\dot{\theta}_b - \sin(\theta_b)\, v_j^y\dot{\theta}_b + \sin(\theta_b)\, a_j^x + \cos(\theta_b)\, a_j^y \end{bmatrix} \quad (3)$$

The vectors $\mathbf{V}_b^o$ and $\mathbf{X}_b^o$ can then be obtained from

$$\mathbf{V}_b^o = [\dot{x}_b \;\; \dot{y}_b]^{\mathrm{T}} = \int_0^t \mathbf{a}_b^o dt \;, \quad \mathbf{X}_b^o = [x_b \;\; y_b]^{\mathrm{T}} = \int_0^t \mathbf{V}_b^o dt \quad (4)$$

If we consider time as the independent variable, x_b can be directly defined as a prescribed function of time. The unknown variables can then be obtained based on the current value of x_b using the iterative Newton-Raphson algorithm, which in turns requires definition of the mechanical constraint equations. In this example, there are four constraints: two constraints for contact point positions between surface and wheels, and two constraints for imposing the tangency constraints. Equation 5 expresses the estimated contact points for the rear and front wheels in terms of the unknown parameters as

$$\mathbf{p}_r^c = \begin{bmatrix} x_r^c \\ y_r^c \end{bmatrix} = \begin{bmatrix} x_b - L_b\cos(\theta_b) - R_w\cos(\theta_b+\vartheta_r) \\ y_b - L_b\sin(\theta_b) - R_w\sin(\theta_b+\vartheta_r) \end{bmatrix},$$
$$\mathbf{p}_f^c = \begin{bmatrix} x_f^c \\ y_f^c \end{bmatrix} = \begin{bmatrix} x_b + L_b\cos(\theta_b) + R_w\cos(\theta_b+\vartheta_f) \\ y_b + L_b\sin(\theta_b) + R_w\sin(\theta_b+\vartheta_f) \end{bmatrix} \quad (5)$$

where x_i^c and y_i^c $(i = r, f)$ specify the coordinates of the contact points in the inertial frame. The first two constraints for contact points can be obtained by inserting the x_i^c coordinate of the contact points in the path function and subtracting from the y_i^c coordinate of the contact points yielding

$$\begin{aligned} \mathrm{C}_1 &= y_b - L_b\sin(\theta_b) - R_w\sin(\theta_b+\vartheta_r) - \mathrm{path}(x_b \\ &\quad - L_b\cos(\theta_b) - R_w\cos(\theta_b+\vartheta_r)) \\ \mathrm{C}_2 &= y_b + L_b\sin(\theta_b) + R_w\sin(\theta_b+\vartheta_f) - \mathrm{path}(x_b \\ &\quad + L_b\cos(\theta_b) + R_w\cos(\theta_b+\vartheta_f)) \end{aligned} \quad (6)$$

The function $y = \mathrm{path}(x)$ defines the specified path function of the surface. The tangent vector of the surface at contact points can be obtained from $\mathbf{T}_i = \left[dx/dx \;\; d\,\mathrm{path}(x)/dx\right]^{\mathrm{T}}_{x=x_i^c}$ for any infinitesimal displacement. The vector from the contact point towards the wheel center can be written as $\mathbf{N}_i = \left[x_i^o - x_i^c \;\; y_i^o - y_i^c\right]^{\mathrm{T}}$. The parameters x_i^o and y_i^o $(i = r, f)$ denote the coordinates of the wheel center in the inertial frame. To obtain a feasible contact between each wheel and surface, one has to impose two tangency constraints as $C_3 = \mathbf{N}_r^{\mathrm{T}}\mathbf{T}_r$, $C_4 = \mathbf{N}_f^{\mathrm{T}}\mathbf{T}_f$. The final form of the constraints vector can be written as $\mathbf{C} = [\mathrm{C}_1 \; \mathrm{C}_2 \; \mathrm{C}_3 \; \mathrm{C}_4]^{\mathrm{T}}$. The constraint Jacobian matrix $\mathbf{C_q}$ is driven by taking the partial derivatives of $\mathbf{C}$ with respect to the unknown parameters, $\mathbf{U} = [\vartheta_r \;\; \vartheta_f \;\; \theta_b \;\; y_b]^{\mathrm{T}}$ according to the equation $\left[\mathbf{C_q}\right]_{(j,k)} = \partial\mathbf{C}_j/\partial\mathbf{U}_k$. The iterative Newton-Raphson algorithm is shown in Fig. 4 for approximating the unknown parameters vector $\mathbf{U}$ based on the current value of x_b coming from dynamic results and by imposing the above constraints.

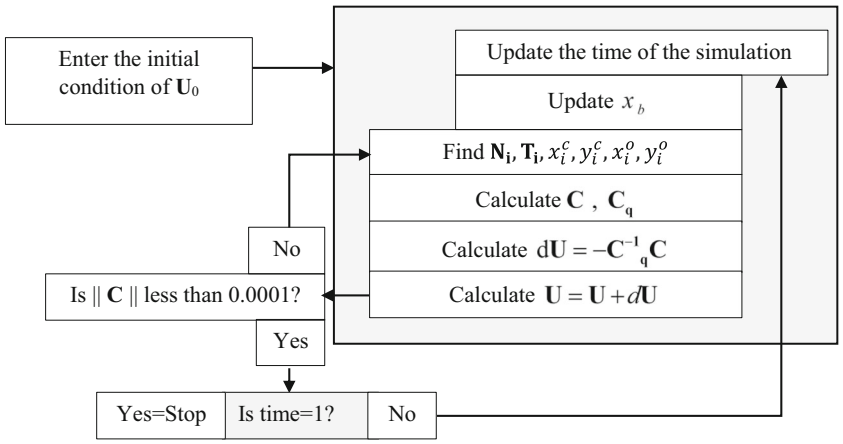
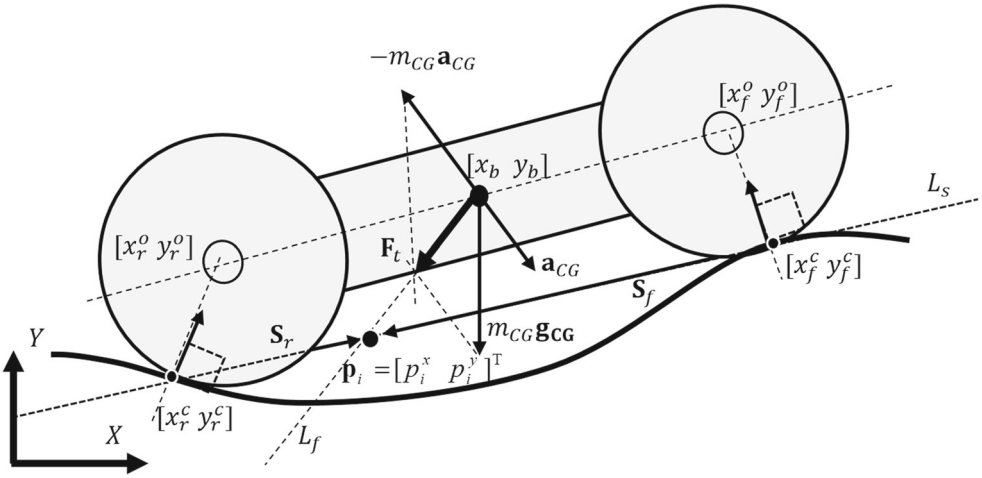


Fig. 4 The iterative Newton-Raphson approach

Fig. 5 The stability diagram



The total angle of each wheel θ_i^t $(i = r, f)$ can be considered as the sum of the encoder angle θ_i of the wheel and the robot body angle θ_b. As a result, the total angular velocity of the wheel can be written as $\dot{\theta}_i^t = \dot{\theta}_i + \dot{\theta}_b$. The velocity of the wheel center can be considered as $v_i = R_w\dot{\theta}_i^t$ The encoder angle can then be obtained by integrating the encoder angular velocity as. Furthermore, according to Fig. 3, the angle of the continuous resistive belt sensor is given by $\beta_i = \vartheta_i - \theta_i$ which can be substituted into the following relation for the effective length of the belt of both wheels as $L_{b_i} = 0.5\pi^{-1}\beta_i L_{t_i}$. The voltages of both wheels are detected in order to estimate β_i, $(i = r, f)$.

4 Real Time Stability Analysis

The stability and balancing of robots without exact estimation of contact angle have been already considered in some researches [43]. A mobile robot equipped by an accelerometer and a gyroscope can only detect its position and orientation. In this case, the stability margins during motion are not usually known, as they require the exact position of the contact points during locomotion on unknown surfaces. By implementing the new sensor, this issue can be properly treated by detecting the contact points and obtaining the required parameters. Figure 5 shows the stability parameters. According to the D'Alembert's principle, the inertial force $m_{CG}\mathbf{a}_{CG}$ of the CG of the robot can be exerted in its opposite direction as an external force. Therefore, according to Fig. 5, the total external force applied to the robot body at its CG can be shown by $\mathbf{F}_t = [F_t^x\ \ F_t^y]^T$. Direction of this force is defined as line L_f. In addition, by connecting two contact points another line denoted by L_s in Fig. 16 is considered. The intersection of these two lines is defined by point $\mathbf{p}_i = [p_i^x\ \ p_i^y]^T$. The vectors $\mathbf{S}_r$=$[S_r^x\ \ S_r^y]^T$ and $\mathbf{S}_f$=$[S_f^x\ \ S_f^y]^T$ pointing respectively from the contact points of the rear and front wheels to the intersection point define the stability margins.

If S_r^x is negative, the normal force applied to the front wheel at the contact point is zero meaning that the front wheel loses its contact with surface and the current position is out of the stability range. This situation happens for the front wheel when S_f^x becomes positive. On the other hand, the robot is stable when the components of S_r^x and $-S_f^x$ became positive. The stability measure ψ can be defined based on the minimum of $-S_f^x$ and S_r^x as $\psi = \min(S_r^x,\ -S_f^x)$ which represents that if ψ (or equivalently one of $-S_f^x$ and S_r^x) approaches zero, one of the wheel starts to leave the surface. For the positive values of ψ, both wheels lie on the surface. One of the wheels leaves the surface and robot overturns when ψ is negative. The components S_r^x and S_f^x can be written as

$$\begin{aligned} S_r^x &= \frac{(x_r^c y_f^c - x_f^c y_r^c)F_t^x x_b + (x_r^c - x_f^c)(F_t^x y_b - F_t^y x_b)}{F_t^y(x_r^c - x_f^c) - (y_r^c - y_f^c)F_t^x} - x_r^c \\ S_f^x &= \frac{(x_r^c y_f^c - x_f^c y_r^c)F_t^x x_b + (x_r^c - x_f^c)(F_t^x y_b - F_t^y x_b)}{F_t^y(x_r^c - x_f^c) - (y_r^c - y_f^c)F_t^x} - x_f^c \end{aligned} \tag{7}$$

One may assume that the inertial forces are negligible and the total external force only includes the gravity effect. If F_t^x is assigned zero value, the equation of stability is simplified and is written as the following equation

$$\psi = \min(x_b - x_r^c,\ x_f^c - x_b) \tag{8}$$

Defining the stability margins, Eq. 8 requires calculating angles β_r and β_f. These angles can be measured by using the resistive sensor proposed here. Therefore, the stability margins can be completely calculated during motion. It is important to notice that the aim of the real time stability analysis presented above is to evaluate the accuracy of the stability margins by using the new sensor data. The results of the stability analysis can then be implemented in a self-balancing process to adjust the positions and torque of mechanisms and robotic actuators.

5 Real Time Dynamics Analysis

The Lagrange method is implemented to derive the dynamic equations. The position and orientation vectors of the main body and two wheels are written as

$$\mathbf{P}_b^t = \begin{bmatrix} x_b \\ y_b \\ \theta_b \end{bmatrix}, \quad \mathbf{P}_{wr}^t = \begin{bmatrix} x_b - L_b\cos(\theta_b) \\ y_b - L_b\sin(\theta_b) \\ \theta_b + \theta_r \end{bmatrix},$$
$$\mathbf{P}_{wf}^t = \begin{bmatrix} x_b + L_b\cos(\theta_b) \\ y_b + L_b\sin(\theta_b) \\ \theta_b + \theta_f \end{bmatrix} \tag{9}$$

The parameters $(.)_{wf}$, $(.)_{wf}$ and $(.)_b$ respectively denote the front wheel, the rear wheel and the body. The first step to derive dynamic equations is to find the Jacobian matrix of the positions and orientations in terms of the generalized coordinates vector $\mathbf{v}_d = [x_b\ y_b\ \theta_b\ \theta_f\ \theta_r]^{\mathrm{T}}$ in the following manner

$$\left[\mathbf{J}_k^t\right]_{ij} = \partial\left[\mathbf{P}_k^t\right]_i / \partial\left[\mathbf{v}_d\right]_j, \ \ (k = wr, wf, b) \tag{10}$$

The closed-form expression of these Jacobian matrices can be represented as

$$\mathbf{J}_b^t = \left[\mathbf{I}_{3\times3}\ \mathbf{O}_{3\times2}\right], \ \ \mathbf{J}_{wf}^t = \begin{bmatrix} 1 & 0 & -0.5\,\mathrm{L}\sin(\theta_b) & 0 & 0 \\ 0 & 1 & 0.5\,\mathrm{L}\cos(\theta_b) & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 \end{bmatrix},$$
$$\mathbf{J}_{wr}^t = \begin{bmatrix} 1 & 0 & 0.5\,\mathrm{L}\sin(\theta_b) & 0 & 0 \\ 0 & 1 & -0.5\,\mathrm{L}\cos(\theta_b) & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 \end{bmatrix} \tag{11}$$

The inertia matrix $\mathbf{D}$ and the Christoffel symbol $\mathbf{C}_{chr}$ (Centrifugal and Coriolis forces) of the dynamics of the robot are calculated as

$$\mathbf{D}_{5\times5} = \sum_{i=(b,\,wf,\,wr)} \mathbf{J}_i^{v\mathrm{T}} m_i \mathbf{J}_i^v + \mathbf{J}_i^{\omega\mathrm{T}} I_i \mathbf{J}_i^\omega,$$
$$\mathbf{J}_i^v = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \mathbf{J}_i^t, \ \ \mathbf{J}_i^\omega = \left[\begin{array}{ccc} 0 & 0 & 1 \end{array}\right] \mathbf{J}_i^t$$
$$[\mathbf{C}_{chr}]_{\mathrm{k}} = \frac{1}{2}\sum_{i=1}^{5}\sum_{j=1}^{5}\left(\frac{\partial\,[\mathbf{D}]_{kj}}{\partial\,[\mathbf{v}_d]_i} + \frac{\partial\,[\mathbf{D}]_{ki}}{\partial\,[\mathbf{v}_d]_j} - \frac{\partial\,[\mathbf{D}]_{ij}}{\partial\,[\mathbf{v}_d]_k}\right)[\dot{\mathbf{v}}_d]_i\,[\dot{\mathbf{v}}_d]_j \tag{12}$$

which can be written in the expanded form as

$$\mathbf{D}_{5\times5} = \begin{bmatrix} m_b + 2\,m_w & 0 & 0 & 0 & 0 \\ 0 & m_b + 2\,m_w & 0 & 0 & 0 \\ 0 & 0 & I_b + 2\,I_w + 0.5L^2 m_w & I_w & I_w \\ 0 & 0 & I_w & I_w & 0 \\ 0 & 0 & I_w & 0 & I_w \end{bmatrix},$$
$$\mathbf{C}_{chr} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{13}$$

The parameters $m_{(.)}$ and $I_{(.)}$ respectively denote the mass and moment of inertia, and the parameters $(.)_w$ and $(.)_b$ respectively define the wheel and the body. The gravity effects associated with the components of $\mathbf{v}_d$ is given as the vector $\mathbf{G}$=[0 $g\,(m_b + 2m_w)$ 0 0 0]$^{\mathrm{T}}$. The constant g denotes the gravity acceleration. The contact forces F_i^X and F_i^Y in the absolute coordinate system XY are externally applied to the rear and front wheels. These forces can be represented in terms of the friction and normal contact forces (F_i^T and F_i^N) in the form of

$$\begin{bmatrix} F_i^X \\ F_i^Y \end{bmatrix} = \begin{bmatrix} \cos(\lambda_i) & -\sin(\lambda_i) \\ \sin(\lambda_i) & \cos(\lambda_i) \end{bmatrix} \begin{bmatrix} F_i^T \\ F_i^N \end{bmatrix} = \mathbf{R}_i^\lambda \begin{bmatrix} F_i^T \\ F_i^N \end{bmatrix},$$
$$\lambda_i = \theta_b + \theta_i + \beta_i + 0.5\,\pi \tag{14}$$

which after pre-multiplication by the transposed Jacobian matrix of the contact points position results the generalized external forces. The position of the contact points is provided in Eq. 7. The associated Jacobian matrix of these points are given by

$$\left[\mathbf{J}_r^c\right]_{ij} = \partial\left[\mathbf{P}_r^c\right]_i / \partial\left[\mathbf{v}_d\right]_j, \ \ \left[\mathbf{J}_f^c\right]_{ij} = \partial\left[\mathbf{P}_f^c\right]_i / \partial\left[\mathbf{v}_d\right]_j \tag{15}$$

which finally yields the generalized external forces as

$$\mathbf{F}_i = \mathbf{J}_i^{c\mathrm{T}}\left[\begin{array}{cc} F_i^X & F_i^Y \end{array}\right]^{\mathrm{T}} = \mathbf{J}_i^{c\mathrm{T}}\mathbf{R}_i^\lambda\left[\begin{array}{cc} F_i^T & F_i^N \end{array}\right]^{\mathrm{T}}, \quad (i = r,\, f) \tag{16}$$

The dynamic equations of motion of the robot can be described by the following equation

$$\mathbf{D}\,\ddot{\mathbf{v}}_d + \mathbf{C}_{chr} + \mathbf{G} = \mathbf{F}_r + \mathbf{F}_f + \boldsymbol{\tau} \tag{17}$$

which has to be solved for obtaining the contact forces and the generalized accelerations in terms of the generalized torques vector $\boldsymbol{\tau} = \left[\begin{array}{ccccc} 0 & 0 & -\left(\tau_f + \tau_r\right) & \tau_f & \tau_r \end{array}\right]^{\mathrm{T}}$. The unknown parameters of the above equation can be collected in a vector as

$$\boldsymbol{\chi}_D\,\boldsymbol{\Theta} = \boldsymbol{\chi}_\tau, \ \ \boldsymbol{\Theta} = \left[(\ddot{\mathbf{v}}_d)^{\mathrm{T}} \ \ F_f^T \ \ F_f^N \ \ F_r^T \ \ F_r^N\right]_{9\times1}^{\mathrm{T}}$$
$$\boldsymbol{\chi}_D = \left[\begin{array}{ccc} \mathbf{D} & -\mathbf{J}_f^{c\mathrm{T}}\mathbf{R}_f^\lambda & -\mathbf{J}_r^{c\mathrm{T}}\mathbf{R}_r^\lambda \end{array}\right]_{5\times9}, \ \ \boldsymbol{\chi}_\tau = [\,\boldsymbol{\tau} - (\mathbf{C}_{chr} + \mathbf{G})]_{5\times1} \tag{18}$$

Equation 18 contains five relations and nine unknown parameters, which means that four more equations are required to calculate all unknowns. In the sticking condition, these required equations can be derived from the accelerations of the wheels center. On the other hand, for each wheel undergoing slip condition, one of the required equations can be derived from the sliding friction. The velocity vector of the wheel center point in the absolute coordinate system can be represented in terms of its tangent and normal components as

$$\begin{bmatrix} \dot{x}_i^o \\ \dot{y}_i^o \end{bmatrix} = \mathbf{J}_i^v\dot{\mathbf{v}}_d = \begin{bmatrix} \cos(\lambda_i) & -\sin(\lambda_i) \\ \sin(\lambda_i) & \cos(\lambda_i) \end{bmatrix} \begin{bmatrix} v_i^T \\ v_i^N \end{bmatrix} \tag{19}$$

which can also be rewritten in the following form

$$\begin{bmatrix} v_i^T \\ v_i^N \end{bmatrix} = \begin{bmatrix} \cos(\lambda_i) & \sin(\lambda_i) \\ -\sin(\lambda_i) & \cos(\lambda_i) \end{bmatrix} \mathbf{J}_i^v \dot{\mathbf{v}}_d = \left[\mathbf{R}_i^\lambda\right]^{\mathrm{T}} \mathbf{J}_i^v \dot{\mathbf{v}}_d, \quad (i = r, f) \tag{20}$$

On the other hand, the parameters v_i^T and v_i^N can also be geometrically obtained by considering the robot kinematics as shown in Fig. 3.

$$\begin{bmatrix} v_f^T \\ v_f^N \end{bmatrix} = \begin{bmatrix} -r\left(\dot{\theta}_b + \dot{\theta}_f\right) \\ 0 \end{bmatrix} = \begin{bmatrix} 0 & 0 & -r & -r & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \dot{\mathbf{v}}_d = \mathbf{W}_f \dot{\mathbf{v}}_d$$
$$\begin{bmatrix} v_r^T \\ v_r^N \end{bmatrix} = \begin{bmatrix} -r\left(\dot{\theta}_b + \dot{\theta}_r\right) \\ 0 \end{bmatrix} = \begin{bmatrix} 0 & 0 & -r & 0 & -r \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \dot{\mathbf{v}}_d = \mathbf{W}_r \dot{\mathbf{v}}_d \tag{21}$$

which upon equating its components from Eqs. 20 and 21 yields $\left(\mathbf{W}_i - \left[\mathbf{R}_i^\lambda\right]^{\mathrm{T}} \mathbf{J}_i^v\right) \dot{\mathbf{v}}_d = 0$ whose differentiation with respect to time yields

$$\left(\mathbf{W}_i - \left[\mathbf{R}_i^\lambda\right]^{\mathrm{T}} \mathbf{J}_i^v\right) \ddot{\mathbf{v}}_d + \frac{d\left(\mathbf{W}_i - \left[\mathbf{R}_i^\lambda\right]^{\mathrm{T}} \mathbf{J}_i^v\right)}{dt} \dot{\mathbf{v}}_d = 0 \tag{22}$$

Equation (22) can be modified to include all unknowns in the following manner

$$\left[\mathbf{W}_i - \left[\mathbf{R}_i^\lambda\right]^{\mathrm{T}} \mathbf{J}_i^v \quad \mathbf{0}_{2\times 2} \quad \mathbf{0}_{2\times 2}\right]_{2\times 9} \left[(\ddot{\mathbf{v}}_d)^{\mathrm{T}} \quad F_f^T \quad F_f^N \quad F_r^T \quad F_r^N\right]_{9\times 1}^{\mathrm{T}} = -\frac{d\left(\mathbf{W}_i - \left[\mathbf{R}_i^\lambda\right]^{\mathrm{T}} \mathbf{J}_i^v\right)}{dt} \dot{\mathbf{v}}_d \tag{23}$$

This equation provides the required relations of each wheel in the sticking condition. On the other hand, in the slipping condition, the first line of (23) should be replaced by the sliding friction relation as $F_i^T = \mu_k F_i^N$. The parameter μ_k denotes the dynamic coefficient of friction. To express more clearly, Eq. 23 can be written in an expanded form for both the sticking and slipping cases separately. It yields

$$\begin{aligned} &\boldsymbol{\chi}_i^1 \boldsymbol{\Theta} = \boldsymbol{\chi}_i^2, \quad \boldsymbol{\chi}_i^1 = \left[\mathbf{W}_i - \left[\mathbf{R}_i^\lambda\right]^{\mathrm{T}} \mathbf{J}_i^v \quad \mathbf{0}_{2\times 2} \quad \mathbf{0}_{2\times 2}\right]_{2\times 9}, \quad \boldsymbol{\chi}_i^2 \\ &\qquad = -\frac{d\left(\mathbf{W}_i - \left[\mathbf{R}_i^\lambda\right]^{\mathrm{T}} \mathbf{J}_i^v\right)}{dt} \dot{\mathbf{v}}_d \end{aligned} \tag{24}$$

for sticking condition, and

$$\begin{aligned} &\boldsymbol{\chi}_i^3 \boldsymbol{\Theta} = \boldsymbol{\chi}_i^4 \\ &\boldsymbol{\chi}_i^3 = \begin{bmatrix} \mathbf{S}_i \\ \begin{bmatrix} 0 \\ 1 \end{bmatrix}^{\mathrm{T}} \left[\mathbf{W}_i - \left[\mathbf{R}_i^\lambda\right]^{\mathrm{T}} \mathbf{J}_i^v \quad \mathbf{0}_{2\times 2} \quad \mathbf{0}_{2\times 2}\right] \end{bmatrix}, \quad \boldsymbol{\chi}_i^4 \\ &\qquad = \begin{bmatrix} 0 \\ -\begin{bmatrix} 0 \\ 1 \end{bmatrix}^{\mathrm{T}} \frac{d\left(\mathbf{W}_i - \left[\mathbf{R}_i^\lambda\right]^{\mathrm{T}} \mathbf{J}_i^v\right)}{dt} \dot{\mathbf{v}}_d \end{bmatrix} \\ &\mathbf{S}_f = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 1 & -\mu_k & 0 & 0 \end{bmatrix}, \quad \mathbf{S}_r \\ &\qquad = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -\mu_k \end{bmatrix} \end{aligned} \tag{25}$$

for slipping condition. The final form of the conditional stick-slip dynamic equations is obtained by combining (18), (24) and (25) in Eq. 26. The coefficient matrices on the left hand sides of each equation are square matrices of the size 9×9. Therefore, they can be inverted to obtain the unknown parameters of vector $\boldsymbol{\Theta}$. The stick condition happens when $F_i^T \leq \mu_s F_i^N$ and it switches to the slipping condition when Eq. 26 violates. The computational scheme for solving conditional dynamic equations is shown in Fig. 6.

$$\begin{cases} \left[\left[\boldsymbol{\chi}_D\right]^{\mathrm{T}} \left[\boldsymbol{\chi}_f^1\right]^{\mathrm{T}} \left[\boldsymbol{\chi}_r^1\right]^{\mathrm{T}}\right] \boldsymbol{\Theta} = \left[\left[\boldsymbol{\chi}_\tau\right]^{\mathrm{T}} \left[\boldsymbol{\chi}_f^2\right]^{\mathrm{T}} \left[\boldsymbol{\chi}_f^2\right]^{\mathrm{T}}\right]^{\mathrm{T}} \rightarrow \text{Front sticking, Rear sticking (k=1)} \\ \left[\left[\boldsymbol{\chi}_D\right]^{\mathrm{T}} \left[\boldsymbol{\chi}_f^1\right]^{\mathrm{T}} \left[\boldsymbol{\chi}_r^3\right]^{\mathrm{T}}\right] \boldsymbol{\Theta} = \left[\left[\boldsymbol{\chi}_\tau\right]^{\mathrm{T}} \left[\boldsymbol{\chi}_f^2\right]^{\mathrm{T}} \left[\boldsymbol{\chi}_f^4\right]^{\mathrm{T}}\right]^{\mathrm{T}} \rightarrow \text{Front sticking, Rear sticking (k=2)} \\ \left[\left[\boldsymbol{\chi}_D\right]^{\mathrm{T}} \left[\boldsymbol{\chi}_f^3\right]^{\mathrm{T}} \left[\boldsymbol{\chi}_r^1\right]^{\mathrm{T}}\right] \boldsymbol{\Theta} = \left[\left[\boldsymbol{\chi}_\tau\right]^{\mathrm{T}} \left[\boldsymbol{\chi}_f^4\right]^{\mathrm{T}} \left[\boldsymbol{\chi}_f^2\right]^{\mathrm{T}}\right]^{\mathrm{T}} \rightarrow \text{Front sticking, Rear sticking (k=3)} \\ \left[\left[\boldsymbol{\chi}_D\right]^{\mathrm{T}} \left[\boldsymbol{\chi}_f^3\right]^{\mathrm{T}} \left[\boldsymbol{\chi}_r^3\right]^{\mathrm{T}}\right] \boldsymbol{\Theta} = \left[\left[\boldsymbol{\chi}_\tau\right]^{\mathrm{T}} \left[\boldsymbol{\chi}_f^4\right]^{\mathrm{T}} \left[\boldsymbol{\chi}_f^4\right]^{\mathrm{T}}\right]^{\mathrm{T}} \rightarrow \text{Front sticking, Rear sticking (k=4)} \end{cases} \tag{26}$$

6 3D Dynamics Analysis

2D simulation only considers proper definition of the sensor application. The motion of robot cannot be simplified as a 2D motion in most of contact situations. Some environment obstacles compel the robot to rotate about roll axis which yields 3D motion. The performance of this sensor in 2D space was previously investigated. Also, to complete the investigation, the 3D motion such as rotating about roll axis has to be considered. These situations can be seen when facing large non-symmetric obstacles in which the left and right wheels height become different and rotation about roll axis causes 3D motion. The 3D slip dynamics can be considered to achieve the semi-real virtual locomotion process. Figure 7 shows the 3D view of a three wheeled mobile robot and the related parameters. According to this figure, the independent dynamic parameters are the position and orientation of the CG, $\mathbf{P}_b^t = \left[x_b \ y_b \ z_b \ \theta_x \ \theta_y \ \theta_z^L\right]^{\mathrm{T}}$

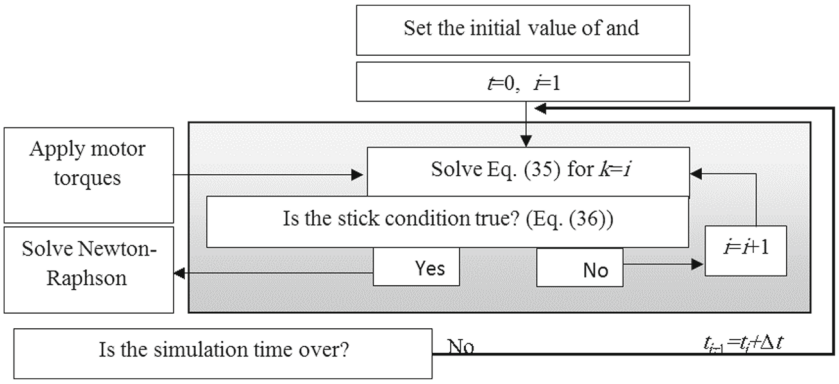


Fig. 6 Dynamics solver flowchart

and the relative orientations of three wheels $\left[\begin{array}{ccc}\theta^e_{fl} & \theta^e_{fr} & \theta^e_r\end{array}\right]$ measured from the encoder data. In addition to the dynamic parameters and similar to the 2D model, the new sensor outputs $\left[\begin{array}{ccc}\beta_{fl} & \beta_{fr} & \beta_r\end{array}\right]$ are essential to geometrically set the robot on the surface using the Newton-Raphson method.

By Imposing tangency constraints and using the longitudinal and lateral tangential vectors ($\mathbf{T}^A_i$ and $\mathbf{T}^L_i$ $(i = r, fl, fr)$) of each wheel at the contact point and the normal vector from the contact point to the wheel center ($\mathbf{N}_i$), feasible contact can be assured. The contact points, the center of each wheel and the CG are the most essential points which are illustrated in Fig. 7 and denoted by $\mathbf{p}^c_i$, $\mathbf{p}^t_{wi}(i = r, f)$ and $\mathbf{p}^t_b$ respectively. The Jacobian matrices of these points in terms of the dynamic parameters (system generalized coordinates) can be calculated from their corresponding relations (11) and (15). The mass matrix and Christoffel index can be calculated using Eq. 12.

The effect of the gravity can be obtained in the following manner as

$$\mathbf{G} = \frac{\partial g \left(\mathbf{p}^t_b + \sum_{i=(b,\, wf,\, wr)} m_i \mathbf{p}^t_{wi}\right)\left[\begin{array}{ccc}0 & 1 & 0\end{array}\right]}{\partial v_d} \tag{27}$$

where v_d collects the dynamic parameters. The dynamic equations can be derived similar to Eq. 17 by some modifications which can be written as

$$\mathbf{D}\ \ddot{\boldsymbol{v}}_d + \mathbf{C}_{chr} + \mathbf{G} = \mathbf{F}_{fl} + \mathbf{F}_{fr} + \mathbf{F}_r + \boldsymbol{\tau} \tag{28}$$

Additional kinematic equations have to be added to the dynamic equations to solve the dynamic problem. By setting the velocity of contact points to zero for non-slipping condition and by differentiating with respect to time, one obtains

$$\mathbf{J}^c_r \ddot{\mathbf{v}}_d + \dot{\mathbf{J}}^c_r \dot{\mathbf{v}}_d = 0 \Rightarrow \mathbf{J}^c_r \ddot{\mathbf{v}}_d = -\dot{\mathbf{J}}^c_r \dot{\mathbf{v}}_d \tag{29}$$

For the slipping case, additional kinematic equations have to be considered (similar to Eq. 26 obtained for the 2D robot). Since the general form of the equations for the 3D mobile robot is similar to the 2D case, they are not rewritten here for the sake of briefness. In the 3D wheeled mobile robot, turning without slipping is impossible. When the axis of the front wheels can be steered, the slip-less turning during motion becomes possible. However, in this paper, this axis is fixed and the robot can slip laterally during turning. Therefore, the slip elimination process considers only the longitudinal slips. Figure 8 shows the slip direction during turning.

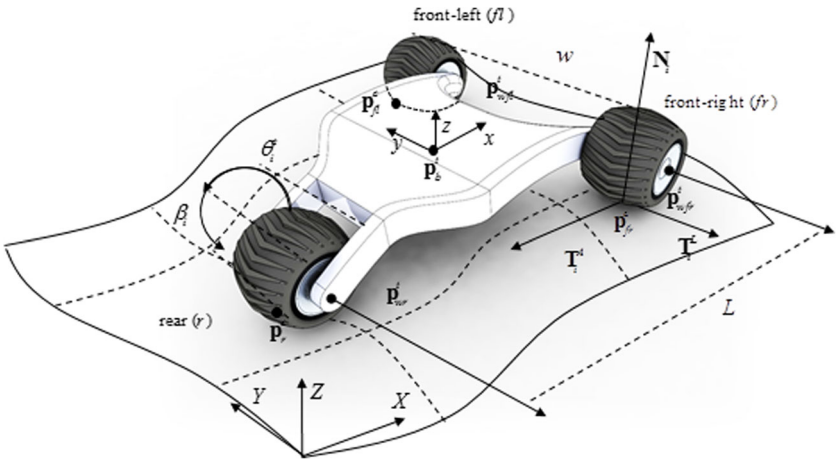


Fig. 7 Parameters of the 3D dynamic model

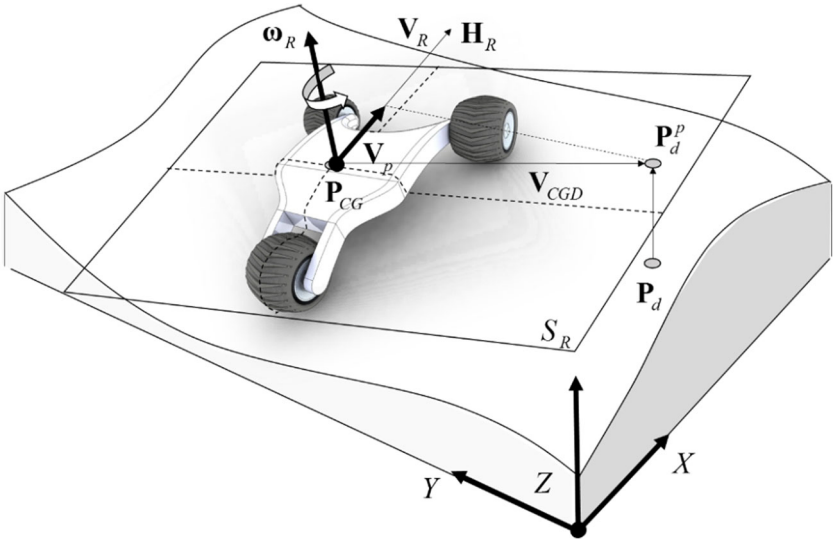


Fig. 8 Control parameters

7 Torque Optimization and Control

In this section, a control strategy is considered for proper navigation of the non-holonomic UGVs and rovers base on the model-based methods or uncertain-dynamics control models [39, 40]. Non-holonomic UGVs moving on flat surfaces do not require the new contact detection sensor because their terrains are completely defined. For these rovers, the ordinary sensors can be efficiently used in stabilizing, navigation, state estimation and slip reduction. Indeed, the contact angles are equal to $0.5\,\pi$ and consequently, the contact point position can be easily calculated without this sensor. Therefore, this paper focuses mainly on the case studies wherein the contact angles have to be detected by the new sensor. This method can be further extended to non-holonomic UGVs moving on rough terrains. Using this sensor for the holonomic rovers and UGVs depends on the platform of the rover. Indeed, when the wheel structure is non-holonomic but the platform of UGV laterally moves by steering all wheels, the new sensor can be used to properly detect the contact points. This sensor cannot be exploited for the rovers with holonomic wheels such as holonomic omni-direction models.

The first step of torque optimization is to propose a control method to produce the desired angular and linear velocity vectors of the rover. Figure 8 illustrates the control parameters of a simple vector-based control method.

The desired point coordinates $\mathbf{P}_d$ are projected onto the rover plane S_R to obtain the projected desired point $\mathbf{P}_d^p$. The vector $\mathbf{V}_{CGD}$ pointing from CG to $\mathbf{P}_d^p$ is projected onto the head vector $\mathbf{H}_R$ to yield $\mathbf{V}_p$. The required linear velocity vector $\mathbf{V}_R$ and angular velocity vector $\boldsymbol{\omega}_R$ are respectively equal to $K_v\mathbf{V}_p$ and $K_\omega\left(\mathbf{V}_{CGD}\,\|\mathbf{V}_{CGD}\|^{-1}\right)\times(\mathbf{H}_R)$. The constants K_v and K_ω are the controller coefficients. These velocity vectors result the desired angular velocity of the wheels. Furthermore, the angular velocity of the rear wheel is equal to $\omega_d^r = R_w^{-1}\,\|\mathbf{V}_R\|\,(\mathbf{V}_R.\mathbf{H}_R)\,\|\mathbf{V}_R.\mathbf{H}_R\|^{-1}$. Finally, the desired angular velocity of the front wheels ω_d^{fl} and ω_d^{fr} can now be defined as $R_w^{-1}\,(\mathbf{V}_R.\mathbf{H}_R)\,\|\mathbf{V}_R.\mathbf{H}_R\|^{-1}$ -0.5w $(\boldsymbol{\omega}_R.\mathbf{z})\,\|\boldsymbol{\omega}_R.\mathbf{z}\|^{-1}$ and $R_w^{-1}\,(\mathbf{V}_R.\mathbf{H}_R)\,\|\mathbf{V}_R.\mathbf{H}_R\|^{-1}$ +0.5w $(\boldsymbol{\omega}_R.\mathbf{z})\,\|\boldsymbol{\omega}_R.\mathbf{z}\|^{-1}$, respectively. In order to accurately solve the dynamic equations,

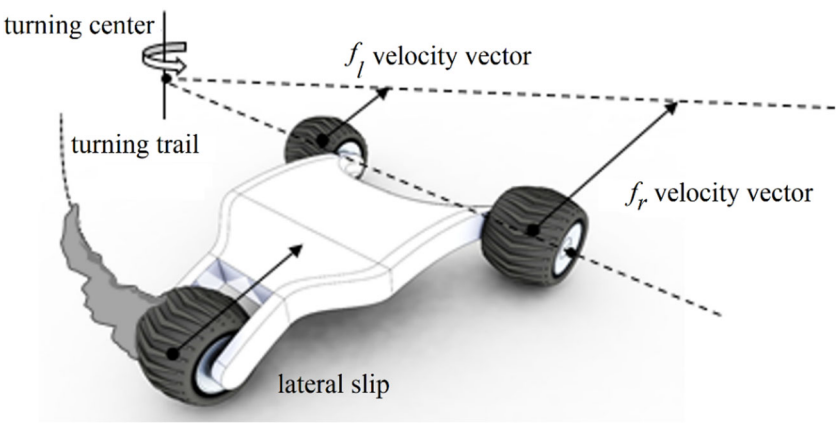


Fig. 9 Slips during turning

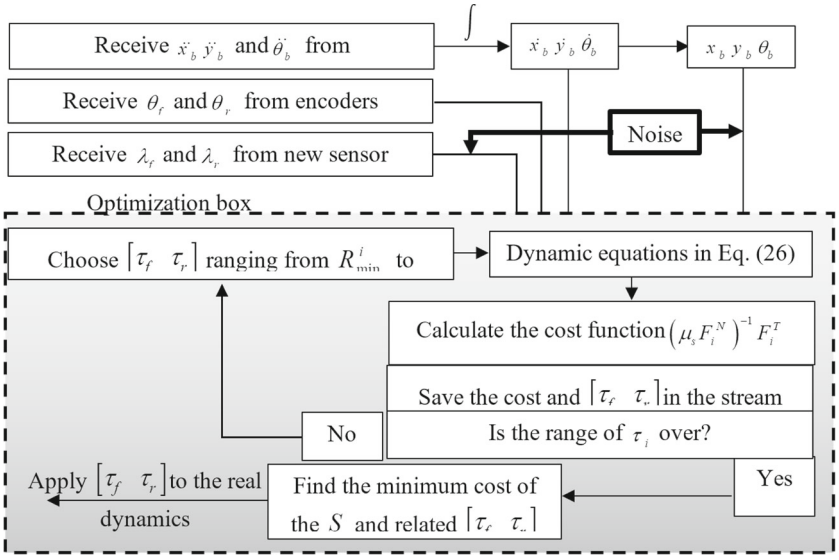


Fig. 10 Optimization algorithm

the new sensor data including the angles λ_i and β_i are required. For slip elimination, the motor torques can be obtained and applied during the real time dynamic analysis. For this purpose, a direct search method illustrated in Fig. 10 is implemented to find the optimal torques whereby the robot moves without slip. The angle $\lambda_i = \theta_b + \theta_i + \beta_i + 0.5\pi$ is considered for the robot equipped with the new sensor, and it is set on $\lambda_i = \theta_b + \theta_i + 1.5\pi + 0.5\pi$ for the case of without sensor. The notice is that in the 3D space the criterion of the torque selection is limited to the longitudinal slips. Lateral slips are out of this investigation (see Fig. 9).

The algorithm of Fig. 10 has to be modified in order to optimize the torques in the 3D model of slip-stick dynamics. At the beginning, the algorithm receives $(\ddot{x}_b, \ddot{y}_b, \ddot{z}_b, \ddot{\theta}_x, \ddot{\theta}_y, \ddot{\theta}_z^L, \theta_{fl}^e, \theta_{fr}^e, \theta_r^e)$ from the accelerometer and encoders and $(\beta_{fl}, \beta_{fr}, \beta_r)$ from the new sensors. The main states of the body including $(x_b, y_b, z_b, \theta_x, \theta_y, \theta_z^L)$ are obtained from time integration of $(\ddot{x}_b, \ddot{y}_b, \ddot{z}_b, \ddot{\theta}_x, \ddot{\theta}_y, \ddot{\theta}_z^L)$. All received parameters undergo noises in order to simulate more real detection process. All updated noisy parameters are implemented in the stick-slip dynamic equations. A similar method of the torque selection according to Fig. 10 is applied to select $(\tau_{fl}, \tau_{fr}, \tau_r)$ Parameters $R^i_{\min}$ and $R^i_{\max}$ in optimization algorithm define the range of possible torques which are respectively equal to $\min\left(K_\tau\left(\omega_d^i - \dot{\theta}_i^e\right), 0\right)$ and $\max\left(K_\tau\left(\omega_d^i - \dot{\theta}_i^e\right), 0\right)$. The constant K_τ defines the torque coefficient parameter of the control and optimization method.

8 Numerical Simulations

Case Study 1: Finding the Contact Point Position by the New Sensor The mass and moment of inertia of the body are respectively 6 kg and 1.1kg m^2. The mass and mass moment of inertia of the wheel are respectively 0.5 kg and

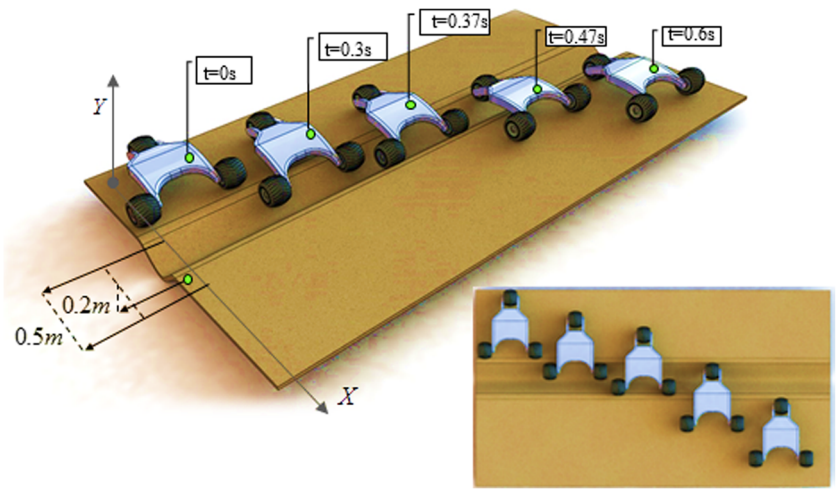


Fig. 11 The hole crossing process

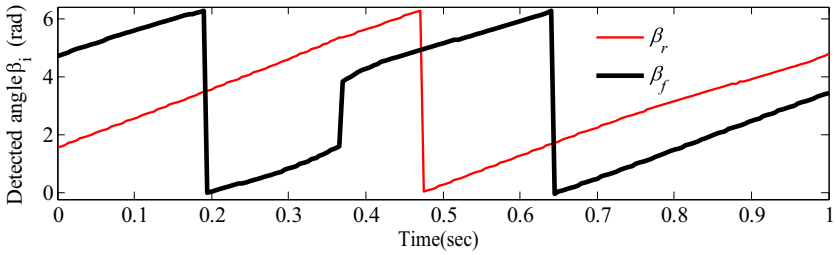


Fig. 12 Variation of the angles β_r and β_f

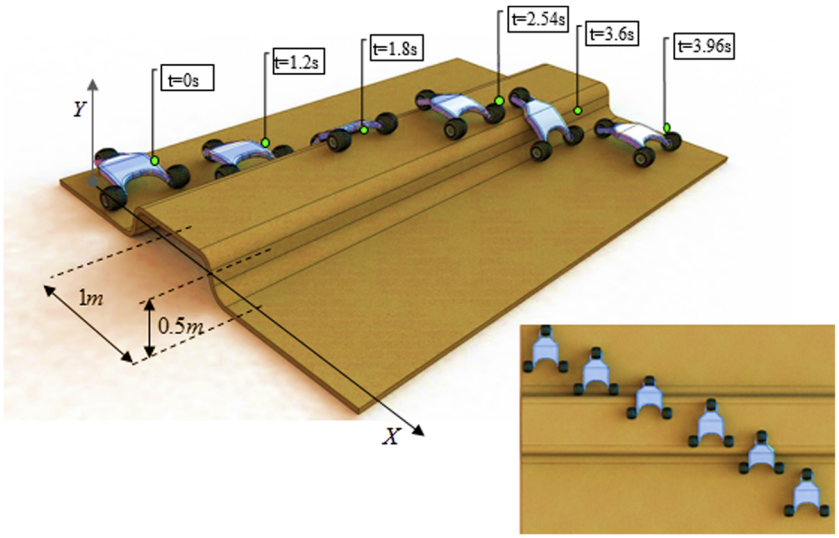


Fig. 13 The obstacle climbing up process

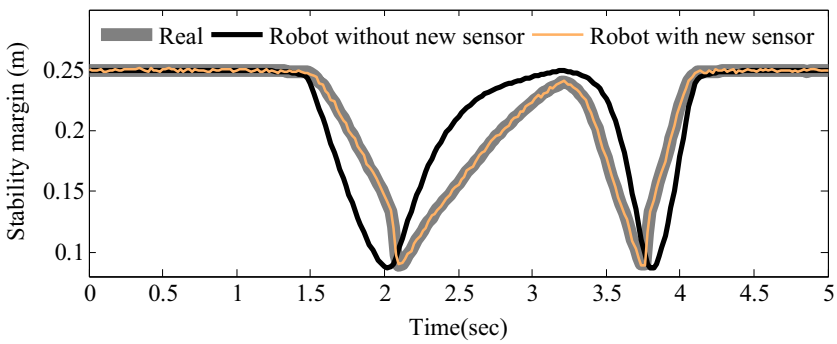


Fig. 14 The stability measure

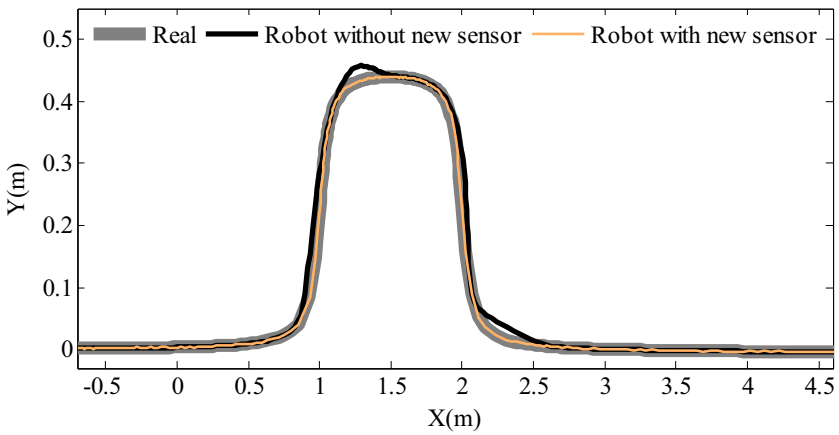


Fig. 15 The scanning result by stick-slip dynamics

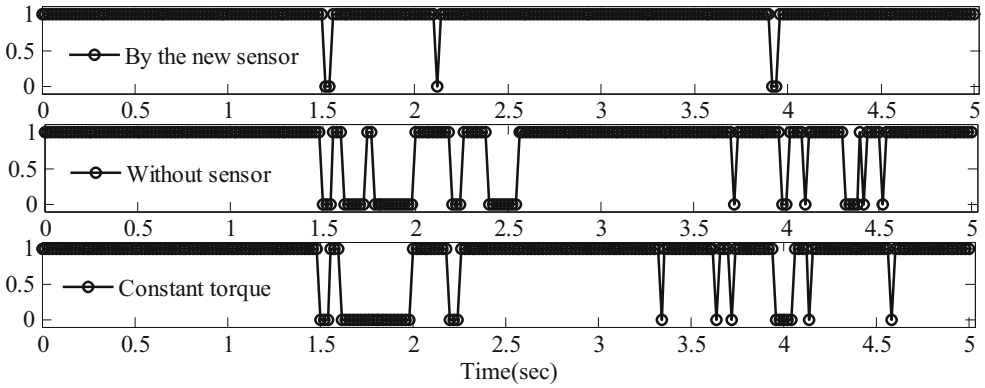


Fig. 16 The slip and stick moments by stick-slip dynamics

0.04kg m^2. The length of the body and the radius of the wheel are 0.5 m and 0.1 m. In the first simulation, a resistive belt is embedded in a 2D mobile platform with two wheels undergoing a motion toward a hole on an uneven surface as shown in Fig. 11 and written in (30) as

$$y = \text{path}(x) = -0.1e^{-40(x^2)} - 0.12e^{-120(x^2)} + 0.3 \qquad (30)$$

Variation of the angles β_r and β_f detected by the resistive belts of both wheels are shown in Fig. 12. Considering the front wheel, a drastic decrease at t =0.2 sec can be seen from angle 6.28 rad to 0 rad. It means that the end point B (see Fig. 1) of the resistive belt of the wheel had contact with surface before this time. After an infinitesimal rotation, the first point A comes into contact. The robot wheel faces the deepest point of the hole at t =0.38 s. At this moment, the wheel contact point drastically moves from a side of the wheel to it's another side which clearly implies that the sensor detects drastic change in the contact point position.

Case Study 2: Simultaneously Scanning, Stability Margin Extraction and Torque Optimization Facing Obstacle The main simulation considers simultaneously torque optimization, slip elimination, surface scanning and stability margin extraction. The robot faces an obstacle illustrated in Fig. 13 and is given by

$$y = \text{path}(x) = 0.15\,(\text{atan}\,(20\,(x-1)) - \text{atan}\,(20\,(x-2))) \qquad (31)$$

The robot moves in three different cases. According to the first case, the robot uses the new sensor. In the second case, the robot only uses common sensors containing accelerometer, encoder and gyroscope to optimize the torques, and in the third case, the robot moves by constant torque without optimization. Figure 13 shows the obstacle climbing up process. The stability measure in both cases is shown in Fig. 14. It can be obviously seen that the accuracy of the robot with the new sensor in stability estimation is more than the robot without the new sensor. The surface scanning result is provided in Fig. 15 which clearly shows the higher accuracy of the

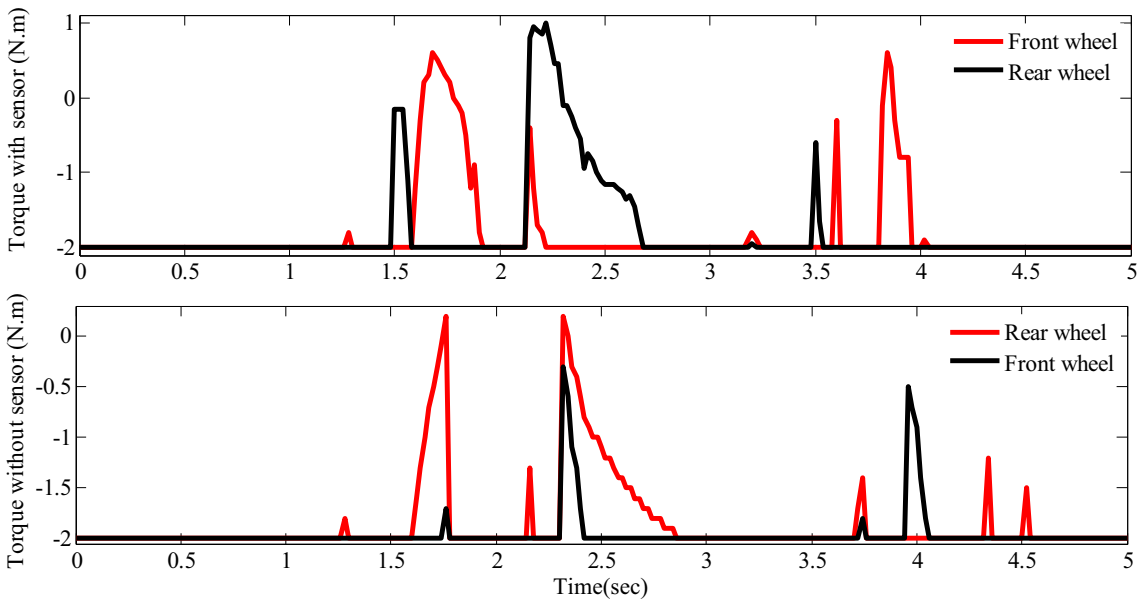


Fig. 17 Torque result

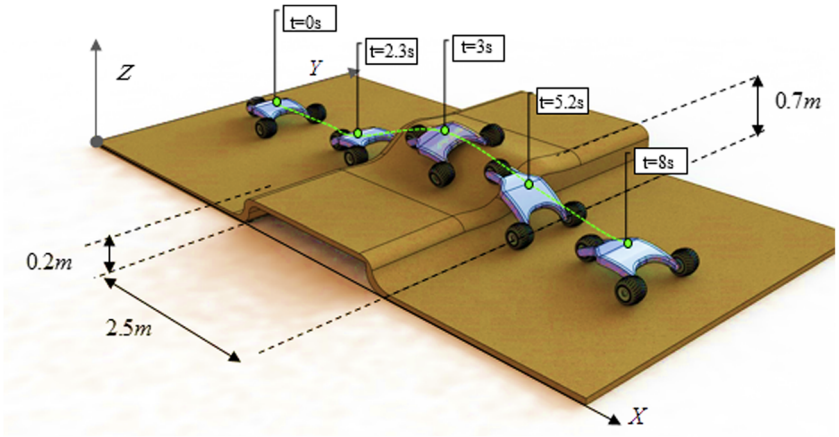


Fig. 18 The simulation process by stick-slip dynamics

robot with the new sensor in compared to the other cases. Figure 16 shows the slip-stick process for three cases. The first case considers torque optimization of the robot with the new sensor. The second case is the torque optimization results for the robot without the new sensor and the third case considers the case of torque exertion with constant value -2 Nm. Value 1 represents the stick moments and value 0 denotes the slip moments. It can obviously be seen that the torque optimization based on the data detected from the new sensor can approximately eliminate all slips during locomotion. Other cases have a lot of slips in the climbing up and down process.

Figure 17 illustrates the resulting optimized torque for the first and the second cases. The torques are limited between -2 Nm and 2 Nm.

Case Study 3: Extension to the Three Dimension Space To conduct more actual process, the sensor efficiency has to be investigated in a 3D space. The robot moves on a 3D surface with a non-symmetric obstacle generated as

$$\begin{aligned} z &= ((\text{atan}(4(-x+6))+0.5\pi)/\pi)(\text{step}_1+\text{step}_2) \\ \text{step}_1 &= 0.2\,((\text{atan}(5\,(x-3))+0.5\pi)/\pi)\,((15(y))+0.5\pi)/\pi) \\ \text{step}_2 &= 0.5\,((\text{atan}(5\,(x-3))+0.5\pi)/\pi)\,((15(y+2))+0.5\pi)/\pi) \end{aligned} \quad (32)$$

The 3D surface and the trail of the robot wheels are shown in Fig. 18. The result of the locomotion shows that the robot passes the obstacle. Figure 19 shows that the robot equipped with the new sensor eliminates successfully the slipping moments, while the robot without this sensor fails to reach this goal. The resulting optimized and non-optimized torques (with and without the new sensor, respectively) are shown in Fig. 20 for both the rear and front wheels.

As it was pointed out previously, the result of this case study illustrates the obvious influence of the new sensor output on the slip reduction. Implementing the sensor output β_i in the dynamic equation, a more realistic estimation of the results investigated through the case studies can be achieved. On the other hand, an appropriate variation of motor torques can change the contact forces in such a way that yields the slip reduction. In other words, the possible slip-less movement is obtained for the next step of simulation by using the optimized torque. Accordingly, opting user-defined motor torque according to the torque optimization algorithms (Fig. 10) empowers this sensor-based estimation to reduce the slip moments (Fig. 20). The results in Figs. 16 and 19 show this influence of the new sensor.

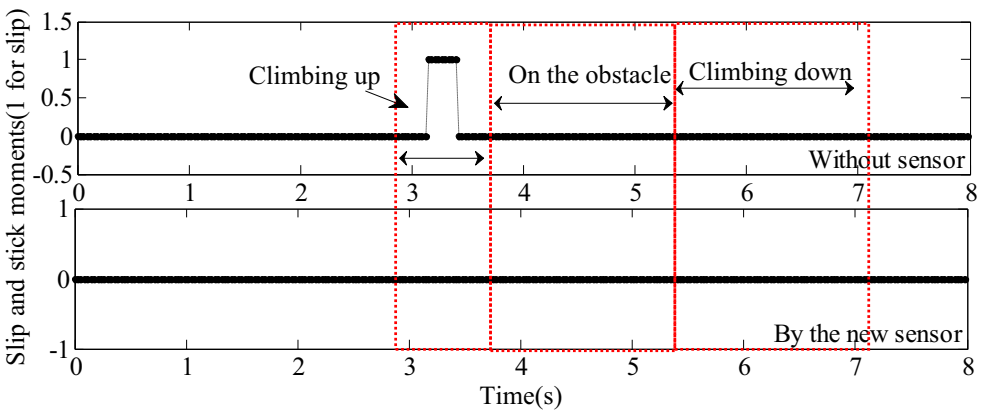


Fig. 19 The slip moments record; Top: without sensor, Below: equipped with the new sensor

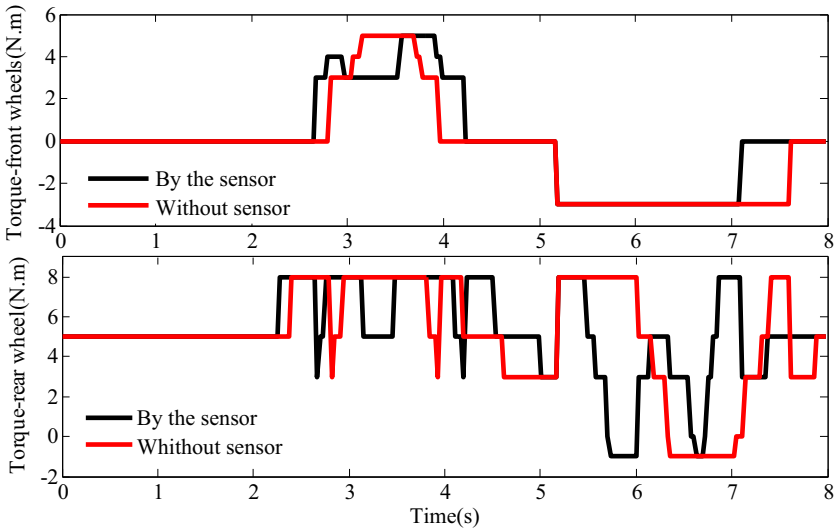


Fig. 20 The motor torques

On the other hand, when using ordinary wheels in slip reduction algorithms, a user-defined contact angle, β_i, instead of the sensor output has to be assumed which consequently yields errors in the next step of the dynamics estimation. According to Fig. 19, the slip in the first plot shows the less performance of the ordinary wheels as compared to the wheels empowered by the new sensor. Moreover, the proper performance of the new sensor facing 3D non-symmetric obstacle in Fig. 18 has implied that the sensor implementation can also reduce the slip when the robot rotates about roll axis. As it has been mentioned previously, in some contact situations, the non-slip movement is impossible when turning or climbing over large obstacles which yields slip reduction and accordingly incompletes slip elimination.

Case Study 4: Long-Term Simulation In a long-term simulation, the rover moves on a wide and complex surface including small and large obstacles, surface fluctuations, and hills ranging from 0 to 100 m. The surface equation is considered as:

$$
\begin{aligned}
z_1 &= 9e^{-0.01(x-30)^2}e^{-0.01((y-3)^2} \\
z_2 &= 21e^{-0.001(x-40)^2}e^{-0.001(y-1)^2}\text{-}15e^{-0.04(x-40)^2}e^{-0.04(y-2)^2} \\
z_3 &= 30e^{-0.04(x-50)^2}e^{-0.04(y-3)^2}\text{+}2\sin(0.25x)\cos(0.33y) \\
z_4 &= (\text{atan}(0.3(x\text{-}5))\text{+}(\pi/2))/\pi)\ (\text{atan}(0.3y)\text{+}(\pi/2)/\pi) \\
z_5 &= z_4\ (z_1\text{+}z_2 + z_3) \\
z &= 0.3z_5\text{+}0.05\sin(3(x+y))
\end{aligned}
\tag{33}
$$

This more-realistic unknown environment is composed of three type of surfaces. The first type contains small-size and distributed obstacles and stones which can be represented in Eq. 33 as $0.05\sin(3(x+y))$. The second type simulates the large hills which can be seen in Eq. 33 as $Ae^{-b(x-x_0)^2}e^{-b((y-y_0)^2}$ wherein multipliers

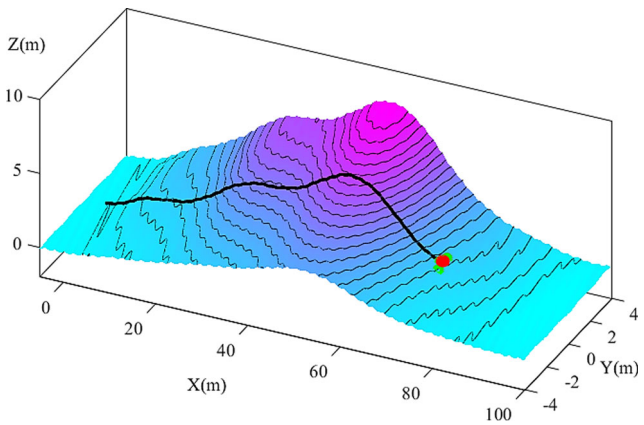


Fig. 21 Long-term simulation

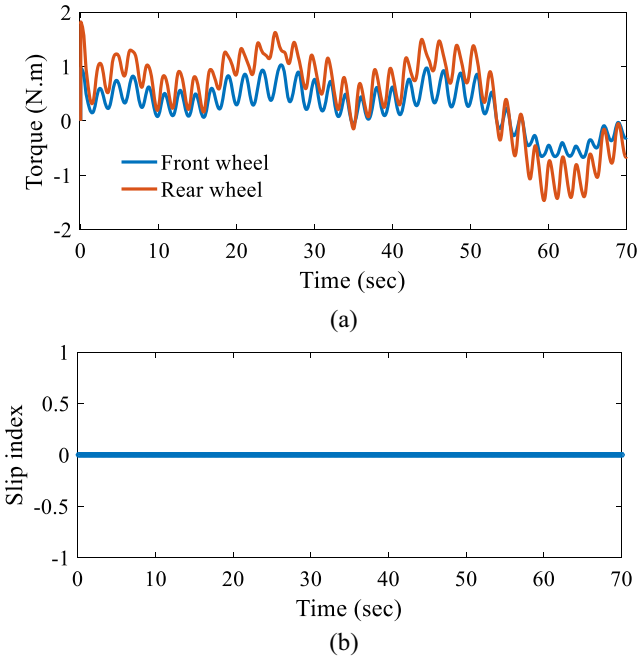


Fig. 22 **a** Torques of the motors, **b** the slip index

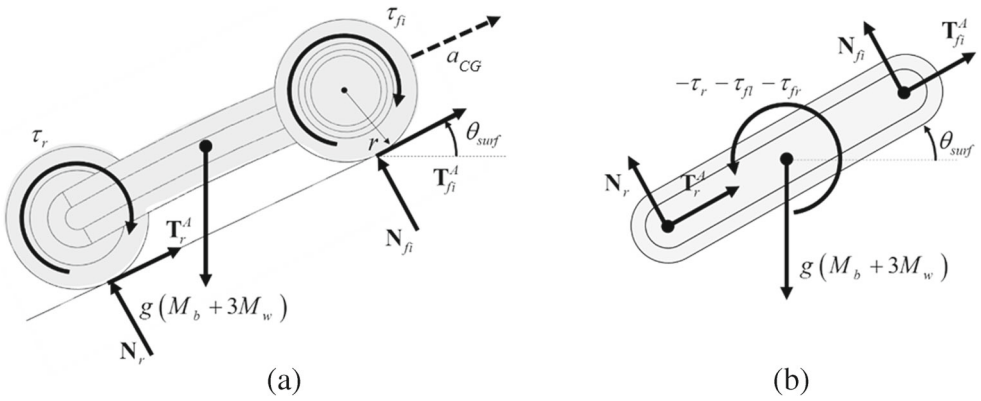


Fig. 23 2D ramp climbing: **a** motor torques and contact forces, **b** free body diagram of the main body

A, b and parameters $[x_0\ y_0]$ define height, sharpness and center of hills. The third type defines the level of the terrain and simulates gentle cliffs formulated as $(\text{atan}(0.3(x\text{-}5))+(\pi/2))/\pi)\ (\text{atan}(0.3y)+(\pi/2)/\pi)$. Other sharp-rising obstacles and surfaces were independently simulated in the previous case studies. The trajectory of the rover is shown in Fig. 21 as a black line. As it can be seen, the UGV has passed three mentioned types of surfaces including hill, distributed small-size obstacles and gentle cliffs. This simulation implies that the rover can overcome more-realistic unknown surfaces. The result of motor torques and slip index are respectively illustrated in Fig. 22a and b. As it can obviously be seen, the slip index remains zero during the simulation. It means that the detected data from the new sensor exploited in the dynamic formulations enhance the ability of the slip less motion, and approves the efficiency of the new sensor.

Figure 22a illustrates the motor torques. Between the seconds 0 and 60, that rover moves from $x = 0$m to $x = 40$ m, the torques are positive due to positive ramp slope. During the remainder of simulation, the rover moves on a negative-slope ramp. It logically has to yield negative torques to avoid undesired accelerating on the ramp. The results show that the slip elimination algorithm aided by the new sensor data keeps the torques negative to simultaneously avoid acceleration and to reduce slip. The fluctuation in the resulting torques is caused of the distributed small-size obstacles simulated as $0.05\sin(3(\text{x+y}))$ and $0.1\sin(0.25x)\cos(0.33y)$. The effect of these obstacles obviously appears in the resulting torque.

9 Validation Strategy of the Stick-Slip Dynamics and Sensor Performance

In this section, the proposed formulation and the efficiency of the sensor is validated. Ramp climbing is the first case investigated here. Figure 23a shows a simplified model of the rover during ramp climbing. The friction force $\mathbf{T}_i^A$ applied to each wheel is equal to $\tau_i r^{-1}$ when the stick condition is assumed at the first step. Neglecting the inertia and mass of the wheels and adding the mass of the wheels to the CG of the rover body, the forces and torques of the main body are simplified according

Table 1 Ramp climbing validation

θ_{surf}	Input $[\tau_{fr},\ \tau_{fl},\ \tau_r]$	Output (stick-slip dynamics approach)	Output (validation approach)
0	[0, 0, 0] Nm	$\begin{bmatrix}\ddot{x}_b & \ddot{y}_b & \ddot{z}_b\end{bmatrix}=\begin{bmatrix}0 & 0 & 0\end{bmatrix}(\text{m s}^{-2})$ $\begin{bmatrix}\ddot{\theta}_x & \ddot{\theta}_y & \ddot{\theta}_z^L\end{bmatrix}=\begin{bmatrix}0 & 0 & 0\end{bmatrix}(\text{s}^{-2})$ $\begin{bmatrix}\mathbf{N}_{fl} & \mathbf{N}_{fr} & \mathbf{N}_r\end{bmatrix}=\begin{bmatrix}18.75 & 18.75 & 37.5\end{bmatrix}(\text{N})$	$\begin{bmatrix}\ddot{x}_b & \ddot{y}_b\end{bmatrix}=\begin{bmatrix}0 & 0\end{bmatrix}(\text{m s}^{-2})$ $\ddot{\theta}_y=0(\text{m s})$ $\begin{bmatrix}\mathbf{N}_{fl} & \mathbf{N}_{fr} & \mathbf{N}_r\end{bmatrix}=\begin{bmatrix}18.75 & 18.75 & 37.5\end{bmatrix}(\text{N})$
0	[1, 1, 1] Nmfor 1 sec	$a_{CG}=4.03(\text{m s}^{-2}),\ \ dx_b=1.53\,(\text{m})$ stick condition $a_{CG}=4.564(\text{m s}^{-2})$	$a_{CG}=4(\text{m s}^{-2}),\ \ dx_b=1.52\,(\text{m})$ stick condition $a_{CG}=4.57(\text{m s}^{-2})$
0	[2, 2, 2] Nmfor 1 sec	$\begin{bmatrix}\mathbf{N}_r & \mathbf{N}_{fi}\end{bmatrix}=[49.43\quad 12.81]\,(\text{N})$ slip condition $a_{CG}=-0.798(\text{m s}^{-2})$	$\begin{bmatrix}\mathbf{N}_r & \mathbf{N}_{fi}\end{bmatrix}=[49.5\quad 12.75]\,(\text{N})$ slip condition $a_{CG}=-0.804(\text{m s}^{-2})$
asin(0.5)	[2, 2, 2] Nmfor 1 sec	$\begin{bmatrix}\mathbf{N}_r & \mathbf{N}_{fi}\end{bmatrix}=[44.481\ \ 10.228]\,(\text{N})$ slip condition	$\begin{bmatrix}\mathbf{N}_r & \mathbf{N}_{fi}\end{bmatrix}=[44.476\ \ 10.238]\,(\text{N})$ slip condition

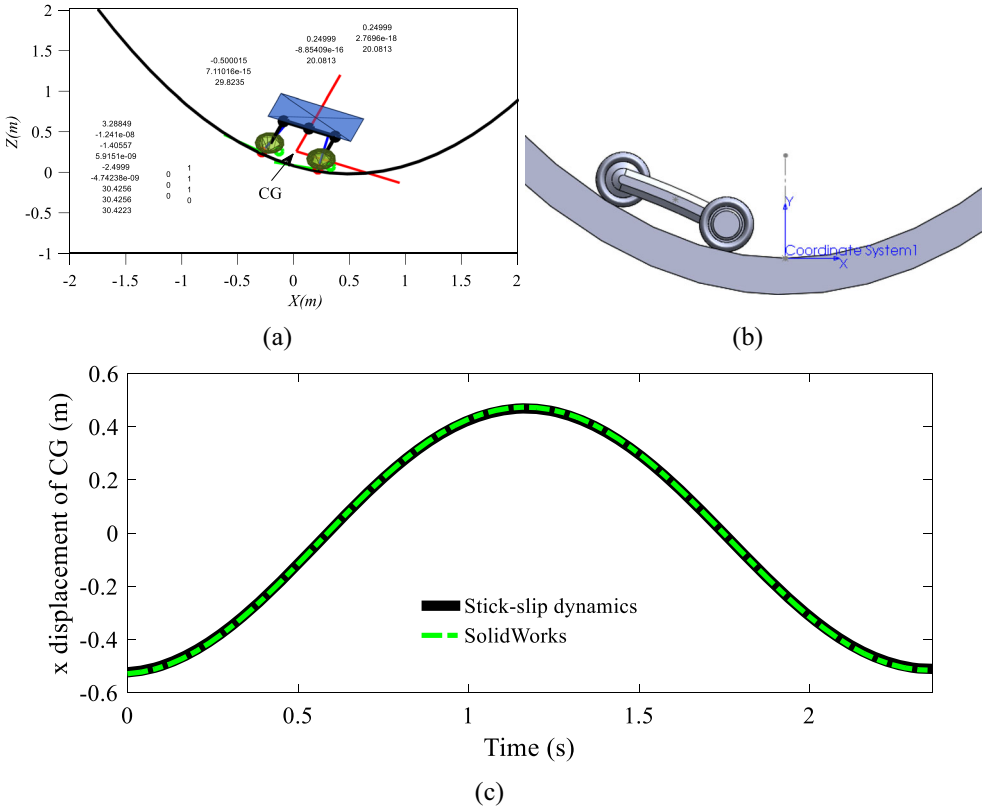


Fig. 24 Dynamics validation: **a** Stick-slip dynamics model, **b** SolidWorks model, **c** Horizontal displacement of CG

to Fig. 23b. The resultant reaction torque of the motors applied to the body is equal to $-\tau_r - \tau_{fl} - \tau_{fr}$. Furthermore, the normal force of the rear wheel can be obtained by writing the Euler equation with respect to the center of the front wheel as $\mathbf{N}_r = 0.5g\,(M_b + 3M_w)\cos(\theta_{surf}) + \left(\tau_r + \tau_{fl} + \tau_{fr}\right) L^{-1}$ and also, the normal force of each front wheel can be subsequently obtained as $\mathbf{N}_{fi} = 0.5\left(g\,(M_b + 3M_w)\cos(\theta_{surf})\text{-}\mathbf{N}_r\right)$, $(i = r, l)$. The stick assumption may only be violated when the resultant friction force $\mathbf{T}_i^A$ exceeds the maximum allowable static friction $\mu_s \mathbf{N}_i$ which means the slip occurs and $\mathbf{T}_i^A = \mu_k \mathbf{N}_i$. The acceleration is equal to $(M_b + 3M_w)^{-1}\left(\sum \mathbf{T}_i^A\text{-}g\,(M_b + 3M_w)\sin(\theta_{surf})\right)$. Table 1 compares the results of the ramp climbing for various situations obtained from the stick-slip dynamics formulations presented in Sections 5 and 6, and the above simple relations (validation approach). A good agreement between the results of both approaches can be seen.

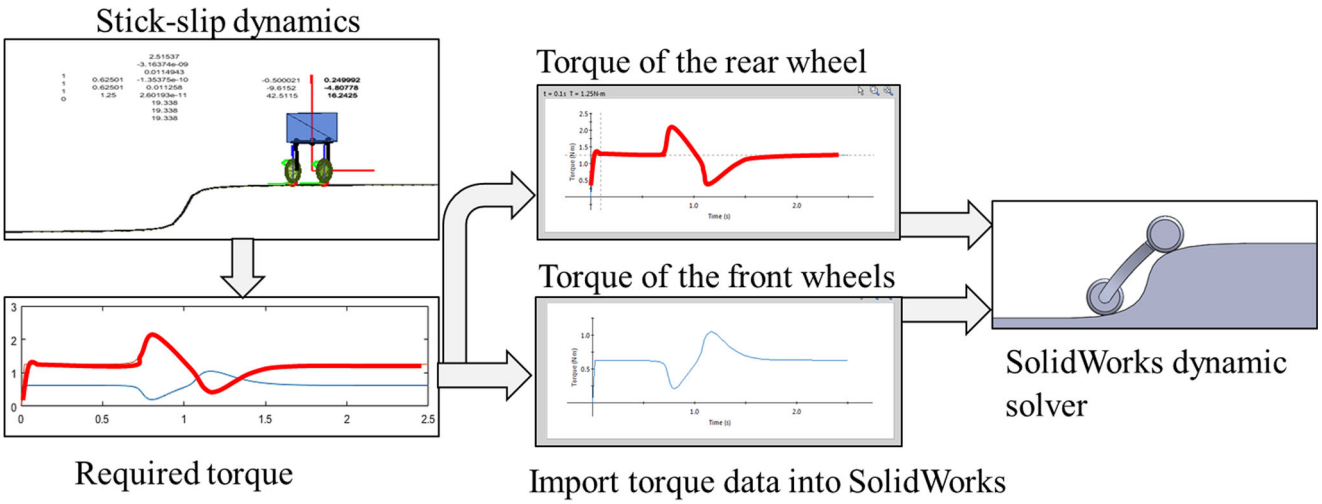


Fig. 25 Validation process

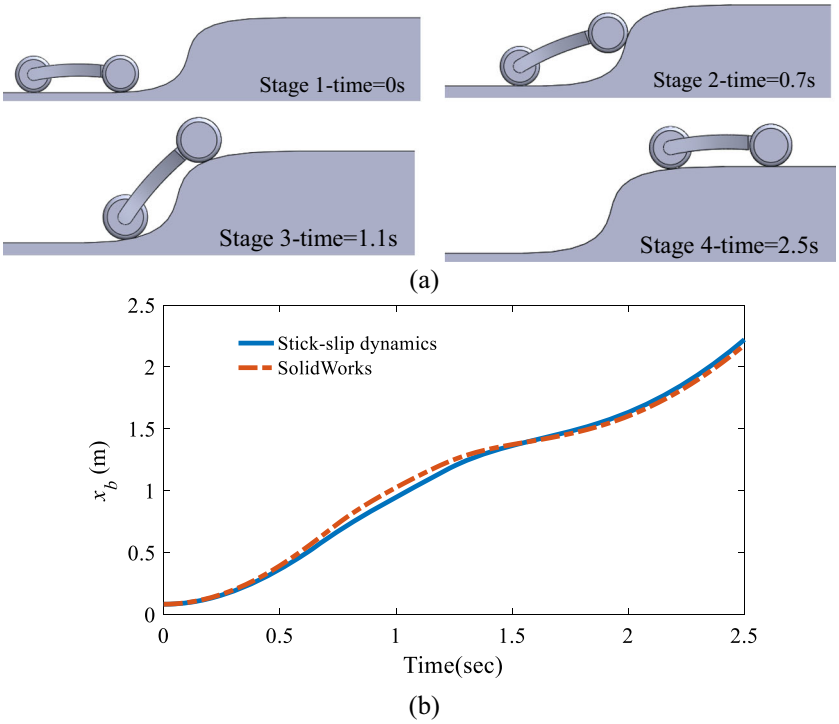


Fig. 26 Variation trend of *xb* during simulation

The validations tabulated in Table 1 start from the simplest case study (zero torque and flat surface). In the second and the third case studies, the torques increase to show typical stick and slip conditions on the flat surface. Finally, the slope of the surface increases to simulate the actuated rover on the ramp. Parameters a_{CG} and dx_b in Table 1 and Fig 23a are respectively the acceleration and the total horizontal displacement of the rover.

In another case study the rover is moved freely on a curve without applying any torque. For validation of the stick-slip dynamics formulations for this case, the motion studio part of SolidWorks software (motion and contact analysis part) is used to simulate the rover on the surface z=0.4$(x-0.5)^2$. The code-based dynamic model of the rover (Fig. 24a) is compared with its model in SolidWorks (Fig. 24b) while the motor torques are set to zero. The horizontal displacement of the CG of the rover is illustrated in Fig. 24c, which clearly shows that both approaches lead to a similar oscillating movement of the rover.

In the last validation study it is intended to simulate the rover when climbing an obstacle formulated as $z = 0.4\,(\text{atan}(10(x\text{-}0.7))+(\pi/2))/\pi)$ without slip by exploiting the torques obtained from the stick-slip dynamics and using the new sensor. For this purpose, the stick-slip dynamic

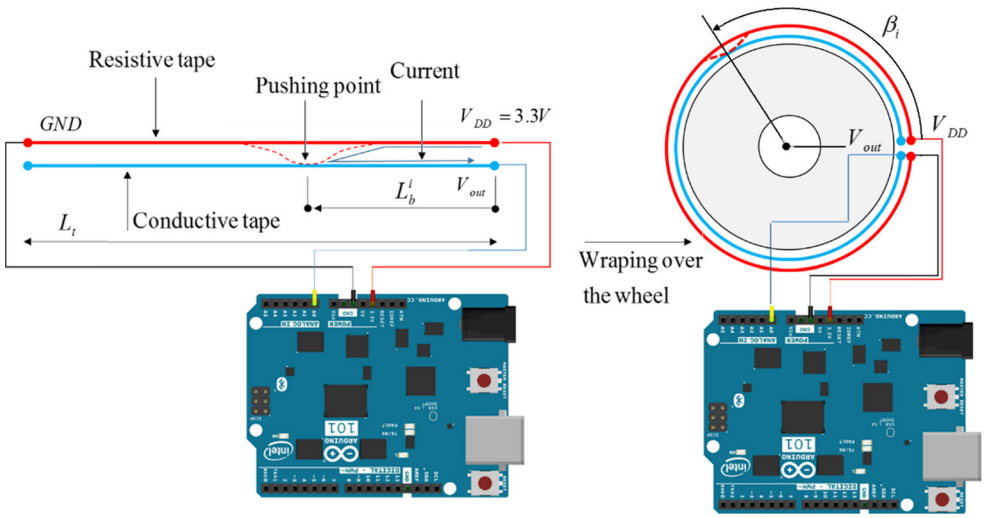


Fig. 27 Connecting the sensor to an ARDUINO UNO

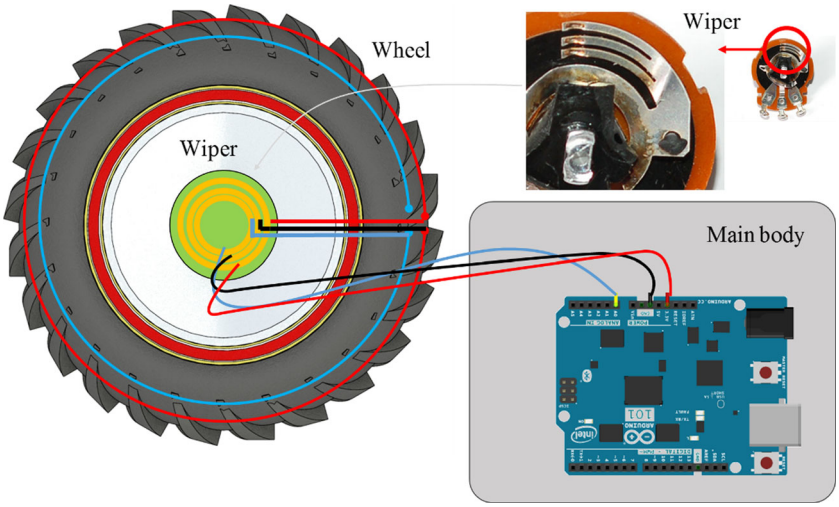


Fig. 28 Wiper usage

simulation of the rover based on our approach is carried out. Then, the motor torques obtained from this analysis are used as the inputs to the same model of the rover built in SolidWorks. It is maybe important to notice that the amounts of motor torques are relied on the new sensor data as well as the stick-slip dynamics formulation presented before. Finally, the obstacle climbing of the rover is simulated in SolidWorks and the variation of x_b is obtained as the output of this analysis. Figure 25 shows the required steps to conduct this validation. Furthermore, Fig. 26a shows schematically the rover in some certain times of the simulation in SolidWorks. Figure 26b compares the variation trend of x_b during simulation from both approaches, which clearly shows that they coincide well for most of the simulation times. This important result confirms the correctness and reliability of the proposed formulation as well as the efficiency of the new sensor.

10 Hardware Implementation of the New Sensor

In this section, the hardware implementation of the new sensor is presented briefly to prove the new concept of using this sensor in real applications. The equation obtained in Section 2 for V_s can be adequately used during simulation. However, in real applications the resistor R_0 is embedded in DAQ cards (such as ARDUINO cards) to directly detect the voltages from the analog input terminals. Therefore, it makes it possible to utilize a linear potentiometer (a voltage divider) instead of a circuit for this sensor. Figure 27 shows how the sensor is connected to an ARDUION UNO. The accuracy and delay of the sensor is equal to the UNO properties ("http://memoir.okno.be/physcomptutorial/ribbon4/ribbon4.htm"). The output voltage V_{out} of the voltage divider can be calculated as $V_{\text{out}} = V_{DD} R_t R_L^{-1}$ wherein the parameters R_t and R_L are respectively the resistance of the pushed part of the circuit and the resistance of the resistive tape. Accordingly, the output voltage is equal to $V_{\text{out}} = V_{DD} L_b^i L_t^{-1} = 0.5\, V_{DD} \pi^{-1} \beta_i$. The sensitivity of the output voltage S_v with respect to the variation of the contact angle β_i is equal to $\partial V_{\text{out}} / \partial \beta_i = 0.5\, V_{DD} \pi^{-1}$ which has a linear relationship with the supply voltage.

Another important issue in hardware implementation of the sensor and wrapping over the wheel is related to the connection part for the wires. Indeed, the sensor data is detected from the wheel surface and then transmitted to the main body of the rover for further process. It means that the sensor wires cross over the wheel axis and go to the main body which can yield twisting of the wires during the wheel rotation. To solve this problem, a wiper can be implemented in the same way as it is always utilized in the circular potentiometers. Figure 28 shows how a wiper can be used in a rover to transmit data from the wheel to the main body.

11 Conclusion

In this paper, a new simple and cheap single-output sensor was proposed to detect the contact point of a typical robotic wheel. The derived formulations and the results showed that adding this sensor to a mobile robot completes the required data to extract the real time stability margins and all required dynamic terms. The results of the first case study proved that the robot detects the drastic contact changes. Also, the robot was empowered to scan the surface without camera or laser. The results showed that the new sensor scanned the surface properly. On the other hand, the error of the stability margin extraction with the new sensor case was lower than other cases. By monitoring the stability margin results, the danger of overturning was reduced. This margin guarantees

the stable locomotion which is the base of complete contact of the wheels and consequently, establishes the essential condition of the slip reduction. As the most important result of this paper, the optimized torques of the wheels obtained from the new sensor data were able to significantly eliminate the slips as compared to other cases in both 2D and 3D dynamic simulations. A long-term simulation showed the ability of the sensor to overcome complex motions. Finally, validation data of the proposed stick-slip dynamics based on various situations implied the reliability of the equations.

References

1. Bogue, R.: Robots to aid the disabled and the elderly. Ind. Robot: Int. J. **40**, 519–524 (2013)
2. Bogue, R.: Robots for space exploration. Ind. Robot: Int. J. **39**, 323–328 (2012)
3. Weisbin, C.R., Lavery, D., Rodriguez, G.: Robots In space into the 21st century. Ind. Robot: Int. J. **24**, 169–181 (1997)
4. Mardani, A., Ebrahimi, S.: A Novel multimode mobile robot with adaptable wheel geometry for maneuverability improvement. Int. J. Robot. **4**, 1–15 (2016)
5. Ebrahimi, S., Mardani, A.: Dynamic Modeling And Construction of a New Two-Wheeled Mobile Manipulator: Self-balancing and Climbing. Int. J. Robot. Theory Appl. **4**, 22–34 (2015)
6. Dogramadzi, S., Giannaccini, M.E., Harper, C., Sobhani, M., Woodman, R., Choung, J.: Environmental Hazard analysis-a variant of preliminary hazard analysis for autonomous mobile robots. J. Intell. Robot. Syst. **76**(1), 73–117 (2014)
7. Wu, C.J., Tsai, C.C.: Localization Of an autonomous mobile robot based on ultrasonic sensory information. J. Intell. Robot. Syst. **30**(3), 267–277 (2001)
8. Xu, H., Liu, X., Fu, H., Putra, B.B., He, L.: Visual Contact angle estimation and traction control for mobile robot in rough-terrain. J. Intell. Robot. Syst. **74**(3-4), 985 (2014)
9. Cervantes-Sánchez, J.J., Rico-martínez, J.M., Pé, r.ez.-M.uñoz., V. H., Orozco-Muñiz, J.D.: A closed-form solution to the forward displacement analysis of a Schönflies parallel manipulator. J. Braz. Soc. Mech. Sci. Eng. **39**, 553–563 (2017)
10. Nagatani, K., Ikeda, A., Sato, K., Yoshida, K.: Accurate Estimation of drawbar pull of wheeled mobile robots traversing sandy terrain using built-in force sensor array wheel. In: IEEE/RSJ International Conference Intelligent Robots and Systems, 2009. IROS 2009, pp. 2373–2378 (2009)
11. Mardani, A., Ebrahimi, S.: Simultaneous Surface scanning and stability analysis of wheeled mobile robots using a new spatial sensitive shield sensor. Robot. Auton. Syst. **98**, 1–14 (2017)
12. Zadarnowska, K., Ratajczak, A.: Task-Priority motion planning of wheeled mobile robots subject to slipping. Robot Motion and Control **422**, 75–85 (2012)
13. Zielinska, T., Chmielniak, A.: Synthesis Of control law considering wheel–ground interaction and contact stability of autonomous mobile robot. Robotica **29**, 981–990 (2011)
14. Conceicao, A.G., Correia, M.D., Martinez, L.: Modeling And friction estimation for wheeled omnidirectional mobile robots. Robotica **34**, 2140–2150 (2016)
15. Siravuru, A., Shah, S.V., Krishna, K.M.: An optimal wheel-torque control on a compliant modular robot for wheel-slip minimization. Robotica **35**, 463–82 (2017)
16. Lamon, P., Siegwart, R.: Wheel Torque Control in Rough Terrain - Modeling and Simulation. In: IEEE International Conference on Robotics and Automation, ICRA, pp. 18–22. Barcelona (2005)
17. Gracia, L., Tornero, J.: Kinematic modeling of wheeled mobile robots with slip. Adv Robot **21**, 1253–1279 (2007)
18. Li, Y.P., Ang, M.H., Lin, W.: Slip Modelling, Detection and Control for Redundantly Actuated Wheeled Mobile Robots. In: IEEE/ASME International Conference on Advanced Intelligent Mechatronics, pp. 967-972. Xi'an (2008)
19. Tharakeshwar, A., Ghosal, A.: A Three-wheeled mobile robot for traversing uneven terrain without slip: simulation and experiments. Mech. Based Des. Struct. Mach. **41**, 60–78 (2013)
20. Sidek, N., Sarkar, N.: Exploiting wheel slips of mobile robots to improve navigation performance. Adv. Robot. **27**, 627–639 (2013)
21. Mohammadpour, E., Naraghi, M.: Robust adaptive stabilization of skid steer wheeled mobile robots considering slipping effects. Adv. Robot. **205-227**, 25 (2011)
22. Sidharthan, R.K., Kannan, R., Srinivasan, S., Balas, V.E.: Stochastic wheel-slip compensation based robot localization and mapping. Adv. Electr. Comput. Eng. **16**, 25–32 (2016)
23. Bayar, G., Bergerman, M., Koku, A.B.: Improving the trajectory tracking performance of autonomous orchard vehicles using wheel slip compensation. Biosyst. Eng. **146**, 149–164 (2016)
24. Fondahl, K., Kuehn, D., Beinersdorf, F., Bernhard, F., Grimminger, F., Schilling, M., Kirchner, F.: An Adaptive Sensor Foot for a Bipedal and Quadrupedal Robot. In: Biomedical Robotics and Biomechatronics (Biorob), IEEE RAS & EMBS International Conference, pp. 270–275 (2012)
25. Natalie, G., Dolores, B.: Nasa's opportunity rover rolls free on mars (2006)
26. Nagatani, K., Ikeda, A., Sato, K., Yoshida, K.: Accurate Estimation of Drawbar Pull of Wheeled Mobile Robots Traversing Sandy Terrain Using Built-In Force Sensor Array Wheel. In: Intelligent Robots and Systems, IROS 2009. IEEE/RSJ International Conference, pp. 2373–2378 (2009)
27. Shirai, T., Ishigami, G.: Development Of in-wheel sensor system for accurate measurement of wheel terrain interaction characteristics. J. Terrramech. **51-61**, 62 (2015)
28. Masehian, E., Katebi, Y.: Sensor-Based motion planning of wheeled mobile robots in unknown dynamic environments. J. Intell. Robot. Syst. **74**(3-4), 893 (2014)
29. Núñez, P., Vazquez-Martin, R., Bandera, A., Sandoval, F.: Fast Laser scan matching approach based on adaptive curvature estimation for mobile robots. Robotica **27**, 469–479 (2009)
30. Tungadi, F., Kleeman, L.: Autonomous Loop exploration and SLAM with fusion of advanced sonar and laser polar scan matching. Robotica **30**, 91–105 (2012)
31. Lauber, A., Sandell, B., Holmbom, P., Pedersen, O.: Tactile Sensors for industrial robots, vol. 8 (1988)
32. Kinoshita, G.I.: Representation and tactile sensing of 3-D objects by a gripper finger. Robotica **1**, 217–222 (1983)
33. Frigola, M., Casals, A., Amat, J.: Human-Robot Interaction Based on a Sensitive Bumper Skin. In: Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference, pp. 283–287 (2006)
34. Iwata, H., Sugano, S.: Human-Robot-Contact-State Identification Based on Tactile Recognition. In: IEEE Transactions on Industrial Electronics, vol. 52, pp. 1468–1477 (2005)
35. Ishiguro, H., Ono, T., Imai, M., Maeda, T., Kanda, T., Nakatsu, R.: Robovie: An interactive humanoid robot. Ind. Robot: Int. J. **28**(6), 498–504 (2001)
36. Pan, Z., Zhu, Z.: Flexible Full-body tactile sensor of low cost and minimal output connections for service robot. Ind. Robot: Int. J. **32**, 485–491 (2005)

37. Lowe, M., King, A., Lovett, E., Papakostas, T.: Flexible Tactile sensor technology: bringing haptics to life. Sensor Rev. **24**, 33–36 (2004)
38. Chen, H.: Robust stabilization for a class of dynamic feedback uncertain nonholonomic mobile robots with input saturation[J]. Int. J. Control Autom. Syst. **12**(6), 1216–1224 (2014)
39. Chen, H., Ding, S., Chen, X., et al.: Global finite-time stabilization for nonholonomic mobile robots based on visual servoing[J]. Int. J. Adv. Robot. Syst. **11**(11), 180 (2014)
40. Chen, H., Zhang, B., Zhao, T., et al.: Finite-time tracking control for extended nonholonomic chained-form systems with parametric uncertainty and external disturbance[J]. J. Vib. Control. **24**(1), 100–109 (2018)
41. Xiong, H., Xiong, H., Chen, Y., Chen, Y., Li, X., Li, X., Zhang, J.: A Scan matching simultaneous localization and mapping algorithm based on particle filter. Ind. Robot: Int. J. **43**, 607–616 (2016)
42. Ebrahimi, S., Eberhard, P.: Contact Of Planar Flexible Multibody Systems Using a Linear Complementarity Formulation. PAMM **5**(1), 197–198 (2005)
43. Tayefi, M., Geng, Z.: Self-Balancing controlled Lagrangian and geometric control of unmanned mobile robots. Journal of Intelligent &, Robotic Systems **90**, 1–13 (2017)

Saeed Ebrahimi is currently an Associate Professor at the Department of Mechanical Engineering, Yazd University, Iran. He received his PhD in Mechanical Engineering from University of Stuttgart, Germany 2006. Then he completed his graduation as a postdoctoral fellow at the Center for Intelligent Machines (CIM) of Mechanical Engineering Department, McGill University, Canada 2007. His current research interests include Multibody System Dynamics, Robotics, Vibration analysis, and Mechanisms Design.

Arman Mardani is currently a PhD student at the Department of Mechanical Engineering, Yazd University, Iran. He received his BSc in Mechanical Engineering in 2012 from Yazd University and his MSc in Mechatronic Engineering in 2014 from Shahrood University. His research interests include robotics, structural design of robots and multibody simulations related to multimode and mobile robots.