Off-Road Robot Modeling with Dextrous Manipulation Kinematics

Joseph Auchter and Carl Moore

Abstract—We present a novel way of modeling wheeled vehicles on outdoor terrains. Adapting concepts from dextrous manipulation, we precisely model the way that three dimensional wheels roll over uneven ground. Our method is easily adaptable to other vehicle designs of arbitrary complexity. Our modeling method is used to validate a new concept for design of off-road vehicle wheel suspensions, called Passive Variable Camber (PVC). Simulation results of a three-wheeled vehicle with PVC demonstrate that the vehicle can negotiate an extreme terrain without kinematic slip and with relatively small changes in camber angle (less than 5°).

I. INTRODUCTION

In recent years there has been increased interest in robots operating outdoors in unstructured environments [13],[6]. Despite this, the methods used to model mobile robots have not changed much. Traditional wheeled mobile robot (WMR) kinematic modeling (for example, [1]) is inadequate because of the complex nature of the an outdoor robot/ground system. Specifically, assumptions about planar motion and two-dimensional wheels are invalid.

Modeling of WMRs is complex because often there are non-holonomic rolling constraints at the wheel/ground contacts. On uneven terrain the contact point can vary along the surface of the wheel in both lateral and longitudinal directions. Therefore assumptions that the wheel can be modeled as a thin disk and that the linear velocity of the wheel center can be determined by \( v = \omega R \) become untenable.

A. Kinematic Slip

Accurate modeling of outdoor WMRs is important because unique issues emerge as a result of the unstructured environment. Among them is increased slip between the wheels and the ground. In addition to dynamic slippage due to terrain deformation or insufficient friction, a WMR is affected by kinematic slip [3],[13]. Kinematic slip occurs when there is no instantaneous axis of rotation compatible with all of the robot’s wheels. This is the general case on uneven terrain because the wheel/ground contact points vary along the surface of the wheel depending on the terrain shape and robot configuration. Ackermann steering geometry, designed to avoid such slip, works properly only on flat ground.

Wheel slip causes several problems. First, power is wasted [13],[3]. Second, wheel slip reduces the ability of the robot to self-localize because position estimates from wheel encoder data accumulate unbounded error over time [5]. Accurate kinematic models are needed to test robot designs which will potentially reduce this costly kinematic slip.

Sreenivasan and Nanua [12] used screw theory to explore the phenomenon of kinematic slip in wheeled vehicle systems moving on uneven terrain. Modeling two wheels joined by a rigid axle, their analysis showed that kinematic slip can be avoided if the distance between the wheel/ground contact points is allowed to vary. The authors of that work suggest the use of a Variable Length Axle (VLA) with a prismatic joint to achieve the necessary motion. The VLA is difficult to implement because it requires a complex wheel axle design.

As a more practical alternative to the VLA, Chakraborty and Ghosal [3] introduced the idea of adding an extra degree of freedom (DOF) at the wheel/axle joint, allowing the wheel to tilt laterally relative to the axle. This new capability, herein named Passive Variable Camber (PVC), permits the distance between the wheel/ground contact points to change without any prismatic joints. Figure 1 shows an example of an axle and two wheels equipped with PVC.

B. Contribution of this work

Traditional methods are not suitable for kinematic modeling of outdoor WMRs due to the complex nature of the terrain/robot system. This document introduces a kinematic simulation of a 3-wheeled mobile robot equipped with PVC and operating on uneven terrain. In order to precisely model the way that three dimensional wheels roll over uneven ground, we adapted concepts developed for modeling dextrous robot manipulators. To our knowledge, the union of the worlds of WMR modeling and dextrous manipulator modeling is novel and does not suffer from many of the assumptions inherent in other modeling techniques. Also,
our method is easily adaptable to other vehicle designs of arbitrary complexity.

The purpose of the simulation is to verify that a WMR equipped with PVC can traverse uneven terrain without kinematic slip. Simulation results show that this can be accomplished with small changes in the camber angle of the PVC joints.

II. ANALOGY BETWEEN WMRs AND DEXTROUS MANIPULATORS

In this work a kinematic model of the WMR/ground system is developed using techniques from the field of dextrous robot hands. The kinematics of dextrous manipulation provide an ideal description of the way wheels roll over uneven terrain.

A WMR in contact with uneven ground is analogous to a multi-fingered robotic “hand” (the WMR) grasping an “object” (the ground). Therefore the theories relating to manipulator contact and grasping are well-suited to modeling of outdoor vehicles. Table I summarizes the analogies between robotic hands and WMRs.

III. WHEELED MOBILE ROBOT SYSTEM WITH PVC

The modeling techniques described in this paper can be applied to any wheeled vehicle, indeed to any system which involves rigid bodies rolling against each other. Here, we model a three-wheeled mobile robot (one front and two rear wheels). The front wheel is steerable, and the two rear wheels have PVC joints. The wheels are torus-shaped, which is more realistic than the typical thin-disk model [12].

Figures 2 and 3 show the coordinate frames which will be used to develop the kinematic equations. Frame \( \{ G \} \) is the ground reference frame. Frame \( \{ contG_i \} \) is the ground contact frame for wheel \( i \). The z-axis of \( \{ contG_i \} \) is the outward normal to the ground surface at the contact point. Frame \( \{ P \} \) is the robot platform reference frame. \( \{ A_i \} \) is the frame at the point of attachment of the wheel \( i \) to the platform. \( \{ W_i \} \) is the reference frame of wheel \( i \). \( \{ contW_i \} \) is the contact frame relative to wheel \( i \). Its z-axis is the outward pointing normal from the torus-shaped wheel, which is collinear with the z-axis of \( \{ contG_i \} \). \( \psi_i \) is the angle between the x-axes of frames \( \{ contG_i \} \) and \( \{ contW_i \} \).

A. Robot Configuration Variables

In this section we introduce the position and velocity variables which describe the state of the robot. We borrow notation from [4] and [10]. The vector of joint velocities is:

\[
\dot{\theta} = \begin{bmatrix} \dot{\phi}_1 & \dot{\gamma}_1 & \dot{\phi}_2 & \dot{\gamma}_2 & \dot{\phi}_3 & \dot{\gamma}_3 \end{bmatrix}^T
\]

where \( \dot{\phi}_i \) is the driving rate of wheel \( i \), \( \dot{\gamma}_i \) is the steering rate of wheel 1, \( \dot{\gamma}_i \) is the rate of tilt of the wheel about the PVC joint of wheel \( i \) (for \( i = 2, 3 \)).

The surface \( S_w \) of a wheel is parameterized relative to its frame \( \{ W_i \} \) by the right-handed orthogonal coordinate chart:

\[
f(u, v) : U \subset \mathbb{R}^2 \rightarrow S_w \subset \mathbb{R}^3
\]

In other words, specifying two parameters \( u_i \) and \( v_i \) will locate a unique point on the surface of wheel \( i \). This point in the cartesian coordinates of \( \{ W_i \} \) is \( f(u_i, v_i) \). Similarly, the ground surface is parameterized relative to its frame \( \{ G \} \) by the chart \( g(x, y) \), meaning that any parameters \( x \) and \( y \) will locate a unique point \( (x, y, g(x, y)) = (x, y, z) \) on the ground surface.

The contact parameters for wheel \( i \) are:

\[
\eta_i = [u_i, v_i, x_i, y_i, \psi_i]^T, \quad i = 1, 2, 3
\]

They are grouped for all three wheels as: \( \eta = [\eta_1^T \eta_2^T \eta_3^T]^T \). Also important are the velocities of the wheel relative to the ground:

\[
contW V_{GW} = V_c = [v_x \ v_y \ \omega_x \ \omega_y \ \omega_z]^T
\]

The leading superscript indicates that the vector is resolved in the \( \{ contW \} \) frame.

<table>
<thead>
<tr>
<th>Manipulators</th>
<th>Mobile Robots</th>
</tr>
</thead>
<tbody>
<tr>
<td>Multi-fingered hand</td>
<td>Wheeled mobile robot</td>
</tr>
<tr>
<td>Grasped object</td>
<td>Ground</td>
</tr>
<tr>
<td>Fingers</td>
<td>Wheels</td>
</tr>
<tr>
<td>Palm</td>
<td>Robot platform</td>
</tr>
</tbody>
</table>
The vectors of configuration and velocity variables which define the state of the robot/ground system are:

\[
q = \begin{bmatrix} \theta \\ \eta \\ P_{PG} \end{bmatrix}, \quad \dot{q} = \begin{bmatrix} \dot{\theta} \\ \dot{\eta} \\ \dot{V}_{PG} \end{bmatrix},
\]

where \(P_{PG}\) and \(\dot{P}_{c}\) are the position equivalents of \(V_{PG}\) and \(\dot{V}_{c}\), respectively.

**B. Wheel/Ground Contact Model**

The way the wheel/ground contact is modeled determines what types of relative motion are allowed between the two surfaces. We model the wheel/ground interaction as a point contact with friction [10], meaning the \(\{\text{cont}W\}\) and \(\{\text{cont}G\}\) frames do not translate relative to each other. Relative rolling is permitted. Mathematically, these conditions are expressed as:

\[
V_{c} = \tilde{B} \dot{V}_{c}
\]

where

\[
\tilde{B} = \begin{bmatrix} 0_{3\times3} \\ I_{3\times3} \end{bmatrix}
\]

for each wheel/ground contact. \(\dot{V}_{c}\), a subset of \(V_{c}\) (from (1)), are called the allowable contact velocities. For point contact with friction \(\dot{V}_{c} = [\omega_x \omega_y \omega_z]^T\).

Montana [9] developed kinematic equations which describe how two arbitrarily-shaped smooth surfaces roll/slide against each other. In our case the two surfaces are the wheel and ground. Metric \((M)\), curvature \((K)\), and torsion \((T)\) forms are used to describe the ground and wheel surfaces. The equations for rolling contact are:

\[
\begin{align*}
(u, \dot{v})^T &= M_w^{-1} (K_w + K^*)^{-1} (-\omega_y, \omega_x)^T \\
(x, \dot{y})^T &= M_y^{-1} R_y (K_w + K^*)^{-1} (-\omega_y, \omega_x)^T \\
\dot{\psi} &= \omega_x + T_w M_w (u, \dot{v})^T + T_y M_y (x, \dot{y})^T.
\end{align*}
\]

The inputs to these equations are the allowable contact velocities \(\dot{V}_{c}\), and the outputs are \(\dot{\eta}\), so we abbreviate them by:

\[
\dot{\eta} = [CK] \dot{V}_{c},
\]

where \([CK]\) stands for “Contact Kinematics”. These are the non-holonomic constraints of the robot/ground system.

**IV. KINEMATIC MODELING METHOD**

In this section, our method for kinematic modeling of the three-wheeled mobile robot is introduced. Table II shows the desired inputs and outputs for the forward kinematics.

**TABLE II**

<table>
<thead>
<tr>
<th>Inputs</th>
<th>Outputs</th>
</tr>
</thead>
<tbody>
<tr>
<td>Desired wheel joint velocities (\dot{\theta})</td>
<td>Platform velocities (V_{PG})</td>
</tr>
</tbody>
</table>

**A. Choice of Inputs \(\dot{\theta}\)**

Following [4], we group the platform/ground relative velocities \(V_{PG}\) and contact velocities \(\dot{V}_{c}\) together in \(V_{GC} = [V_{PG} \quad \dot{V}_{c}^T]^T\). Jacobian matrices can be formed such that:

\[
J_{GC} V_{GC} = J_R \dot{\theta}.
\]

Equations (6) are constraints which relate the joint velocities \(\dot{\theta}\) to the relative ground/platform and contact velocities \(V_{GC}\). In the general case neither \(J_{GC}\) nor \(J_R\) are square and thus are not invertible.

Because of the constraints (6) we cannot freely choose our inputs \(\dot{\theta}\). However, we can calculate inputs consistent with (6) which are as close as possible (in the least-squares sense) to a vector of desired inputs. This is done as follows. Let \(c\) be the number of columns of \(J_{GC}\). The QR decomposition [7] of matrix \(J_{GC}\) is:

\[
J_{GC} = QR.
\]

Let \(r = \text{rank}(J_{GC})\). Split \(Q\) into \([Q_1, Q_2]\), where \(Q_2 \in \mathbb{R}^{c \times (c-r)}\). \(Q_2\) forms a orthonormal basis for the null space of \(J_{GC}^T\), meaning \(J_{GC}^T Q_2 = 0\) or \(Q_2^T J_{GC} = 0\). Pre-multiplying both sides by \(Q_2^T\) yields:

\[
Q_2^T J_{GC} V_{GC} = Q_2^T J_R \dot{\theta},
\]

or

\[
Q_2^T J_R \dot{\theta} = 0.
\]

Equation (7) is a set of constraint equations for the inputs \(\dot{\theta}\). To make use of these equations, let \(C_{\dot{\theta}} = (Q_2^T J_R) \in \mathbb{R}^{p \times q}\) and \(\text{rank}(C_{\dot{\theta}}) = p\). The QR decomposition of \(C_{\dot{\theta}}^T\) is:

\[
C_{\dot{\theta}}^T = [QC_1 QC_2] R_C,
\]

where \(QC_2 \in \mathbb{R}^{p \times p}\). Then \(C_{\dot{\theta}} QC_2 = 0\), meaning \(QC_2\) is an orthonormal basis for the null space of \(C_{\dot{\theta}}\). At this point, we can choose independent generalized velocity inputs \(\dot{\theta}_{g}\) such that

\[
\dot{\theta} = QC_2 \dot{\theta}_{g}.
\]

However, since neither \(C_{\dot{\theta}}\) nor \(QC_2\) are unique and both change as the robot configuration changes, the generalized inputs \(\dot{\theta}_{g}\) have no physical interpretation and their relationship with the actual joint velocities \(\dot{\theta}\) is unclear.

Since (8) is of limited use, we take another step. We want our actual joint velocities \(\dot{\theta}\) to match some desired joint velocities \(\dot{\theta}_{d}\), or \(\dot{\theta} \approx \dot{\theta}_{d}\). Combining this with (8), we have:

\[
QC_2 \dot{\theta}_{g} \approx QC_2 \dot{\theta}_{d}.
\]

To get as close as possible in the least squares sense to \(\dot{\theta}_{d}\), use the pseudo-inverse [11] of \(QC_2\):

\[
\dot{\theta}_{g} = QC_2^+ \dot{\theta}_{d} = (Q_{C_2}^T QC_2)^{-1} Q_{C_2}^T \dot{\theta}_{d}.
\]

Since the columns of \(QC_2\) are orthonormal, \(QC_2^+\) reduces to \(Q_{C_2}^T\). Noticing that \(\dot{\theta} = QC_2 \dot{\theta}_{g}\), we can pre-multiply both sides of (9) by \(QC_2\) to get:
\[ Q_{C2} \dot{\theta}_d = Q_{C2} Q_{C2}^T \dot{\theta}_d, \]
or
\[ \dot{\theta} = Q_{C2} Q_{C2}^T \dot{\theta}_d = J_{in} \dot{\theta}_d. \]

Equation (10) can be thought of as a transformation that takes the desired velocities \( \dot{\theta}_d \), which can be arbitrary, and transforms them such that \( \dot{\theta} \) satisfy the constraints (6) while remaining as close as possible to \( \dot{\theta}_d \).

First, it eliminates the need to deal with independent generalized velocities \( \dot{\theta}_g \), which have no physical meaning. We can instead directly specify a desired set of joint velocity inputs \( \dot{\theta}_d \) and get a set of actual inputs \( \dot{\theta} \) which satisfies the constraints (6) of the robot/ground system. Second, \( \dot{\theta} \) is guaranteed to be as close as possible to \( \dot{\theta}_d \) in the least squares sense. Third, (10) gives us control over the type of motion we want: for instance, if we want a motion trajectory that minimizes the PVC joint angles \( \gamma \), then we set \( \gamma_d = \gamma^d = 0 \). The actual \( \gamma \) values will then remain as close to 0 as the system constraints permit.

### B. Holonomic Constraints and Stabilization

The robot/ground system is modeled as a hybrid series-parallel mechanism. Each wheel is itself a kinematic chain between the platform and the ground, and there are three such chains in parallel. The closure constraints [10] for the parallel mechanism specify that each kinematic chain must end at the same frame (in this case, \{P\}). Let \( T_{AB} \) be the \( 4 \times 4 \) homogeneous rigid body transform between frames \( A \) and \( B \). Then the closure constraints for the robot are

\[ T_{GP, \text{wheel}1} = T_{GP, \text{wheel}2} = T_{GP, \text{wheel}3}. \]

These can be interpreted as ensuring that the robot platform remains rigid: proper relative lengths and orientations are preserved. Equations (11) can be written as

\[ T_{GP, \text{wheel}1} - T_{GP, \text{wheel}2} = 0, \]
\[ T_{GP, \text{wheel}1} - T_{GP, \text{wheel}3} = 0, \]

which are algebraic equations of the form \( C(q) = 0 \). To avoid having to solve a mixture of differential and algebraic equations, \( C(q) \) is differentiated to obtain:

\[ \dot{C}(q) = \frac{\partial C}{\partial q} \dot{q} = J(q) \dot{q}. \]

Because (13) are velocity-level constraints, during integration error can accumulate leading to violation of the position-level constraints \( C(q) = 0 \). Many different algorithms have been proposed to deal with this issue [8]. We choose the method by Yun [14] which is based on Baumgarte’s widely-used method [2] because it is simple to implement, has a clear interpretation, and is effective for our simulation. Yun suggests replacing (13) with:

\[ J(q) \dot{q} + \sigma C(q) = 0, \quad \sigma > 0, \]

which for any arbitrary initial condition \( C_0 \) has the solution

\[ C(t) = C_0 e^{-\sigma t}. \]

Clearly, (15) converges exponentially to the desired constraints \( C(q) = 0 \) even if the constraints become violated at some point during the simulation. For our simulation, we found that values of \( \sigma \) between 1 and 10 produced good results (\( ||C(q)||_2 < 3 \times 10^{-4} \)).

### C. Definition of Output Velocities

We are interested in the motion of the platform resulting from the input joint velocities \( \dot{\theta} \). The time derivative of the coordinate transform relating the ground frame \( \{G\} \) and the platform frame \( \{P\} \) is:

\[ \dot{T}_{GP} = \begin{bmatrix} \dot{R}_{GP} & \dot{p}_{GP} \\ 0 & 0 \end{bmatrix}. \]

The linear velocity of the origin of the platform frame relative to the ground frame is \( v_P = \dot{p}_{GP} \). The rotational velocity of the platform expressed in the platform frame is

\[ \omega_P = \left( R_{GP} \dot{R}_{GP} \right)^\vee, \]

where the \( \vee \) operator extracts the \( 3 \times 1 \) vector components from the skew-symmetric matrix \( R^T \dot{R} \). These output velocities are coupled in a \( 6 \times 1 \) vector and are written as a linear combination of \( \dot{\theta} \) and \( \dot{\eta} \):

\[ V_P = \begin{bmatrix} v_P \\ \omega_P \end{bmatrix} = \Phi_{VP} \begin{bmatrix} \dot{\theta} \\ \dot{\eta} \end{bmatrix}. \]

### D. Forward Kinematics Equations

We now have all of the tools that we need to make a complete set of ordinary differential equations (ODEs) to model the robot/ground system.

Equations (10) relate the desired and actual joint velocities of the system. The rolling contact equations (5) are the non-holonomic system constraints. The stabilized holonomic constraints (14) ensure that the wheels remain in the proper position and orientation relative to one another. The platform velocities are calculating according to (18). As all of these ODEs are linear in the velocity terms, they can be collectively written in the form:

\[ M(q) \dot{q} = f(q) \]

where

\[ M(q) \dot{q} = \begin{bmatrix} I & 0 & 0 & 0 \\ -\Phi_{VP1} & -\Phi_{VP2} & I & 0 \\ J_1 & J_2 & J_3 & J_4 \\ 0 & I & 0 & -[C K] \end{bmatrix} \begin{bmatrix} \dot{\theta} \\ \dot{\eta} \\ V_{PG} \\ V_C \end{bmatrix}, \]

\[ f(q) = \begin{bmatrix} J_1 \dot{\theta}_d \\ 0 \\ -\sigma C(q) \\ 0 \end{bmatrix}, \]

\[ \Phi_{VP} = [\Phi_{VP1} \Phi_{VP2}] \text{ and } J(q) = [J_1 J_2 J_3 J_4]. \]
E. Adaptability of the Modeling Method

Our formulation is adaptable to other vehicle designs of arbitrary complexity: one simply has to create new coordinate transforms $T_{GP}$ which reflect the geometry of the new system. All other equations will remain identical in structure to those presented here. This makes our modeling method versatile and powerful for realistic kinematic simulations of outdoor vehicles operating on rough terrains.

V. RESULTS AND DISCUSSION

The kinematic simulation was run on several different surfaces and for various inputs. MATLAB's ODE suite was used to solve (19) and the Spline Toolbox was used to generate the ground surfaces. We present results for two surfaces: a high plateau and a randomly-generated hilly terrain.

A. Climbing a Hill

As a demonstration of the usefulness of our modeling method, here we present a simulation of the 3-wheeled mobile robot climbing a steep hill onto a plateau. To the authors' knowledge, this simulation, which precisely represents the rolling motion of the wheels on a complex ground surface, is not possible with other existing methods. Fig. 4 shows the 3-wheeled robot climbing the hill.

The hill climbing simulation was run for 20 seconds with the following desired inputs:
- Steering rate $\dot{\phi}_1 = 0$
- Driving rates $\dot{\alpha}_{1,2,3} = 1 \text{ rad/sec} \approx 57.3 \text{ deg/sec}$
- PVC joint rates $\dot{\gamma}_{2,3} = 0$.

Fig. 5 plots the $\dot{\theta}$ inputs and the steering and PVC angles along with their desired values $\dot{\theta}_d$.

B. Random Terrain

Our simulation also works for more general and complex surfaces. Fig. 6 shows the 3-wheeled robot negotiating a randomly-generated ground surface.

The inputs for this simulation were the same as for the hill climbing simulation in the previous section. Fig. 7 plots the paths of the 3 wheel/ground contact points in the ground $x$-$y$ plane. It also shows the projections of the wheel centers in that plane, to show that the wheels tilt as the robot traverses the uneven terrain. The platform velocities $V_p$ are plotted in Fig. 8.

Fig. 9 plots the $L_2$ error in satisfaction of the holonomic constraints (12) and the rolling contact kinematic equations (5). Fig. 9 shows that the constraint equations are well satisfied during the course of the simulation.

It is evident from Fig. 5 that only relatively small changes (less than 5°) in wheel camber angle are necessary to eliminate kinematic slip and thus improve motion precision and predictability.

VI. CONCLUSION AND FUTURE WORK

In this paper we have presented a novel way of modeling wheeled vehicles on outdoor terrains. In order to precisely model the way that three dimensional wheels roll over uneven terrain, we adapted concepts developed for modeling dextrous robot manipulators. To our knowledge, the union of the worlds of WMR modeling and dextrous manipulator modeling is novel. Our method provides a concise and meaningful set of ordinary differential equations which completely
describe the kinematics of the robot/ground system. Also, our method is easily adaptable to other vehicle designs of arbitrary complexity; one simply has to create a new set of coordinate transforms which reflect the geometry of the system.

The purpose of the simulation is to validate a new concept for design of off-road vehicle wheel suspensions. We used our modeling techniques to simulate the motion of a three-wheeled vehicle with two Passive Variable Camber joints. The results demonstrate that the vehicle can negotiate an extreme terrain without kinematic slip and with relatively small changes in camber angle (less than 5°).

Our next step will be a dynamic simulation of the PVC-equipped robot. This will allow a study of dynamic slip and power consumption with and without PVC. We are also designing an experimental set-up by which we can verify that a real wheel and axle with a PVC joint can negotiate an uneven surface with reduced or eliminated kinematic and dynamic slip and less power consumption than a regular wheel/axle combination.

REFERENCES