Terrain Response Estimation Using an Instrumented Rocker-Bogie Mobility System

Timothy Setterfield, Alex Ellery

***Abstract*—This paper presents a procedure to model the drawbar pull and resistive torque of an unknown terrain as a function of normal load and slip using on-board rover instruments. *Kapvik*, a planetary micro-rover prototype with a rocker-bogie mobility system, is simulated in two dimensions. A suite of sensors is used to take relevant measurements; in addition to typical rover measurements, forces above the wheel hubs and rover forward velocity are sensed. An estimator determines the drawbar pull, resistive torque, normal load and slip of the rover. The collected data is used to create a polynomial fit model which closely resembles the real terrain response.**

***Index Terms*—Terrain response, rocker-bogie, terramechanics, drawbar pull, resistive torque, rover mobility, *Kapvik*.**

I. Introduction

In order to analyze the trafficability of upcoming terrain or optimize the performance of a planetary rover using traction control it is necessary to know the terrain response. An assessment of the upcoming terrain using forward looking range data coupled with a model of the terrain response allows an assessment of whether the upcoming terrain can be traversed. Optimal control of a rover for trafficability or for power consumption, achieved through the favourable distribution of forces amongst the rover wheels [1], [2], also depends on knowledge of the terrain response. For wheels driving in deformable terrain, there is no proportional relationship between applied wheel torque and the force delivered by the terrain. Additionally, the force and moment delivered by the terrain is sensitive to normal load and slip in a nonlinear manner.

Most planetary rovers use relatively rigid metal wheels and drive in deformable sand-like terrain. Terramechanics equations for analyzing this scenario were developed by J.Y. Wong and A.R. Reece in the late 1960's [3]. These equations require many soil properties, most of which must be found empirically. Measuring these properties requires several instruments that are unlikely to be included on planetary exploration missions because of their mass and relatively low utility. As a result, researchers have sought methods to estimate soil properties during regular rover operation.

One such method for estimating two important soil properties (cohesion and internal friction angle) was presented by Iagnemma et al. [4] using a simplified stress distribution and the assumption that wheel sinkage can be measured. Since there are several other soil properties involved in Wong's terramechanics equations, re-creating the terrain response relationships from these properties alone requires that representative values be chosen for the remainder of the soil properties.

High-fidelity dynamic simulations of an articulated planetary rover were conducted by Ding et al. using Lagrangian mechanics and Wong's terramechanics equations [5]. In these simulations the wheeled mobile robot was treated as an articulated multibody system with a moving base. Wong's terramechanics equations were used to solve for the moments and forces exerted on the wheels.

Ray et al. were able to re-create the terrain response of a four wheeled mobile robot using an Extended Kalman-Bucy Filter and a fifth unpowered wheel to measure velocity [6].

The rocker-bogie planetary rover mobility system has been used on the Sojourner Rover, the Mars Exploration Rovers [7], and the Curiosity Rover; it also comprises the mobility system of *Kapvik*, a 30 kg micro-rover prototype developed for the Canadian Space Agency (Figure 1). The left and right sides of the rover body are connected to two rocker links via a rotary joint. An internal differential ensures that the rocker joint angles are equal and opposite with respect to the body. The front wheel is attached to the front end of the rocker; another link, termed the bogie, is attached to the rear end of the rocker via a free rotary joint. Two rear wheels are attached to either end of the bogie. The mobility system thus has three wheels per side. Above each wheel is a single-axis force-sensing load cell, added to aid in the estimation of terrain response. A linear bearing protects the load cell from off-axis loading. The rover's mobility system was designed, manufactured, and assembled at Carleton University in parallel with the research presented in this paper. The major dimensions of *Kapvik* are shown in Table I.

Wheel Dimension	Value	Units
Wheelbase l_w	700	mm
Total Length	850	mm
Total Width	782	mm
Height (without arm)	350	mm
Wheel radius r_w	75	mm
Wheel width b_w	70	mm

TABLE I: *Kapvik* major dimensions.

The objective of the research presented herein is to develop a procedure for estimating the terrain response of a homogeneous soil using only the on-board instruments of *Kapvik* or a similar rover. Instead of obtaining soil properties

Timothy Setterfield is a Master's Graduate of the Department of Mechanical and Aerospace Engineering at Carleton University, Ottawa, ON, Canada; he is currently working at the European Space Agency. e-mail: tim.setterfield@gmail.com.

Alex Ellery is a Professor at the Department of Mechanical and Aerospace Engineering at Carleton University, Ottawa, ON, Canada, and the Canadian Research Chair of Space Robotics and Space Technology.

Manuscript received August 23, 2011

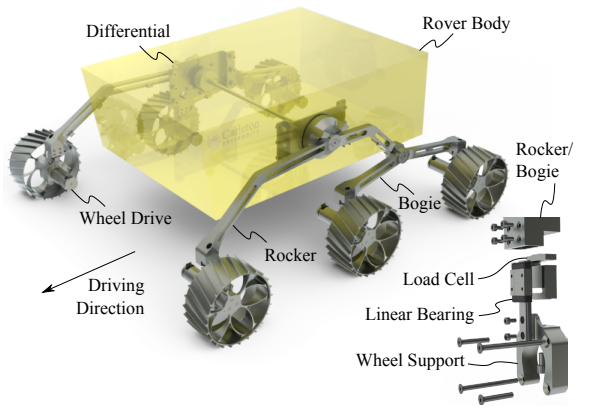


Fig. 1: A three dimensional rendering of the *Kapvik* micro-rover's rocker-bogie mobility system and the load cell assembly used for single-axis force sensing.

and using them to calculate the net traction relationships, this research focuses on directly estimating the net traction relationships. The method presented herein does not require representative values of soil properties to be chosen. The major net traction relationships are simplified into two-dimensional polynomial equations. Although this fails to directly capture information about the soil properties, it allows the full terrain response models to be recaptured from estimated data points via polynomial fitting. This paper presents: the simplification of terramechanics relationships into polynomial equations; the multibody dynamic simulation of the *Kapvik* micro-rover in two dimensions; and the development of an estimator capable of determining the terrain response using only on-board sensors. Although this procedure is presented in the context of a rover with a rocker-bogie mobility system, this should not be considered restrictive: the same techniques could be used with other wheeled rovers.

II. Terramechanics Equations

Interaction between a wheel and the soil beneath it produces a distribution of normal stresses σ and shear stresses τ_s. The two net effects of this are a moment, termed resistive torque τ_R, and a net forward force, termed drawbar pull DP. These effects are obtained by integrating the stresses around the wheel rim [3]. Wong's terramechanics equations for a rigid wheel driving on deformable terrain are used in this paper.

A. Wong's Equations

When a wheel's forward velocity v_w is less than the product of wheel radius and angular velocity $r_w\omega$, the non-dimensional term slip i is used to describe the degree to which the wheel is slipping. When a wheel's forward velocity v_w is greater than the product of wheel radius and angular velocity $r_w\omega$ the non-dimensional term skid i is used to describe the degree to which the wheel is skidding. The variable i is positive when the wheel is slipping and negative when the wheel is skidding [8]–[11].

$$i = 1 - \frac{v_w}{r_w\omega} \qquad \text{when } |r_w\omega| \geq |v_w| \qquad \text{(slip)} \qquad (1)$$

$$i = \frac{r_w\omega}{v_w} - 1 \qquad \text{when } |r_w\omega| < |v_w| \qquad \text{(skid)} \qquad (2)$$

The relationships below are all developed for a wheel in slip, where $i \geq 0$. The case for skidding is handled in the next section.

Soil flow under a wheel was empirically found by Wong to form forward and rearward flow zones, with the maximum normal stress occurring at the transition point between the two (at an angle of θ_m between $\theta = 0$ and $\theta = \theta_0$ in Figure 2). The location of θ_m is given by the following relationship [3]:

$$\theta_m = \theta_0 \left(c_1 + c_2 i\right) \qquad (3)$$

where θ_0 is the total wheel-soil contact angle, and c_1 and c_2 are empirically determined constants.

The normal stress under the wheel follows a different relationship in the forward and rearward flow zones [3].

$$\sigma = \left(k_1 + k_2 b_w\right)\left[\frac{r_w}{b_w}\left(\cos\theta - \cos\theta_0\right)\right]^{n_s}$$
$$\text{when} \quad \theta_m \leq \theta \leq \theta_0 \quad \text{(forward region)} \qquad (4)$$

$$\sigma = \left(k_1 + k_2 b_w\right)\left[\frac{r_w}{b_w}\left(\cos\left[\theta_0 - \frac{\theta}{\theta_m}\left(\theta_0 - \theta_m\right)\right] - \cos\theta_0\right)\right]^{n_s}$$
$$\text{when} \quad 0 \leq \theta < \theta_m \quad \text{(rearward region)} \qquad (5)$$

where k_1 and k_2 are pressure-sinkage constants, b_w is the width of the wheel, and n_s is the soil deformation exponent.

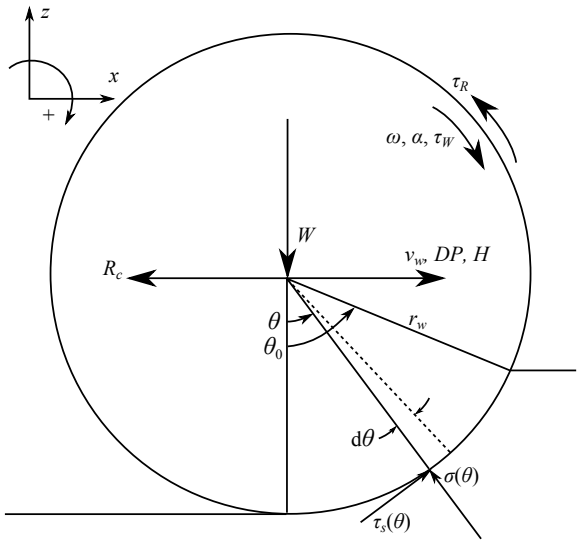


Fig. 2: Wheel coordinates, stresses and dynamics.

The shear stress τ_s at a point on the wheel rim is dependent on soil properties, the normal stress σ at that point, and the shear displacement [3]. The shear displacement of the soil for a wheel in steady-state is a function of the slip i.

$$\tau_s = \left(c + \sigma\tan\phi\right)\left(1 - e^{\frac{-r_w}{K_s}\left[\left(\theta_0 - \theta\right) - \left(1 - i\right)\left(\sin\theta_0 - \sin\theta\right)\right]}\right) \qquad (6)$$

where c is soil cohesion, ϕ is internal friction angle, and K_s

is the shear deformation parameter.

Forces perpendicular to the ground include the normal load W, the perpendicular component of normal stress σ, and the perpendicular component of shear stress τ_s. Since the analysis is performed for a wheel moving with a constant velocity entirely parallel to the ground, the forces perpendicular to the ground sum to zero.

$$-W + b_w r_w \left(\int_0^{\theta_0} \sigma \cos\theta d\theta + \int_0^{\theta_0} \tau_s \sin\theta d\theta \right) = 0 \quad (7)$$

A root finding algorithm must be used to solve Equation 7 for the wheel-soil contact angle θ_0, given normal load W and slip i [3]. Once the wheel-soil contact angle θ_0 has been found, the drawbar pull DP and resistive torque τ_R can be calculated:

$$DP = b_w r_w \left(\int_0^{\theta_0} \tau_s \cos\theta d\theta - \int_0^{\theta_0} \sigma \sin\theta d\theta \right) \quad (8)$$

$$\tau_R = b_w r_w^2 \int_0^{\theta_0} \tau_s d\theta \quad (9)$$

B. Skidding Wheel

Wheels undergoing skid have received very little attention in the literature compared with wheels undergoing slip. One of the reasons for this is likely that in challenging environments where traction analysis is necessary, skidding is almost never encountered.

Recent experiments by Ding et al. on rover wheels driving on planetary soil simulant were conducted down to skids of $i = -0.4$ [11]. Although this study does not cover the full spectrum of possible slips/skids (i.e. from $-1 \leq i \leq 1$), it does give clear insight into the shapes of the drawbar pull DP and resistive torque τ_R curves for a skidding wheel in deformable terrain. The shapes of the curves in skid are very similar to those for slip, but are approximately anti-symmetric about the vertical axis $(i = 0)$ intercept. The Pacejka "Magic Tyre Formula", used extensively for modelling the dynamics of road vehicles, produces a curve of the character described above [12]. The drawbar pull curve for skidding is the same shape as the drawbar pull curve for slipping, but is anti-symmetric about the vertical axis intercept [6], [12]. Lhomme-Desages et al. [8] use a similar model for drawbar pull that is also anti-symmetric about the vertical axis intercept. In this paper, curves for both drawbar pull DP and resistive torque τ_R were made to be anti-symmetric about the vertical axis intercept. When the wheel is skidding $(i < 0)$, the following equations can be used to obtain the drawbar pull DP and resistive torque τ_R using the anti-symmetric curves:

$$DP(i) = -DP(i = -i) + 2DP(i = 0) \quad | \quad i < 0 \quad (10)$$

$$\tau_R(i) = -\tau_R(i = -i) + 2\tau_R(i = 0) \quad | \quad i < 0 \quad (11)$$

where DP for $i \geq 0$ is calculated using Equation 8, and τ_R for $i \geq 0$ is calculated using Equation 9.

C. Soil Properties

The soil properties used in this paper were mostly taken from a thorough experimental study of rover wheel performance on a planetary soil simulant by Ding et al. [11]. The soil simulant is designed to closely resemble lunar soil; the soil properties are summarized in Table II.

Soil Property	Value	Units	Reference
Soil deformation exponent n_s	1.1	-	Ding soil simulant [11]
Cohesion c	250	Pa	Ding soil simulant [11]
Internal angle of friction ϕ_s	31.9	$^\circ$	Ding soil simulant [11]
Shear deformation parameter K_s	11.4	mm	Ding soil simulant [11].
Soil modulus of cohesion k_c	15.6	kPa/m$^{n_s-1}$	Ding soil simulant [11]
Soil modulus of friction k_ϕ	2407.4	kPa/m^{n_s}	Ding soil simulant [11]
Pressure-sinkage constant k_1	12.0	kPa	Ding soil simulant [11] †
Pressure-sinkage constant k_2	1845.3	kPa/m	Ding soil simulant [11] †
Maximum stress angle modulus c_1	0.18	-	Empirical, loose sand [3]
Maximum stress angle modulus c_2	0.32	-	Empirical, loose sand [3]

TABLE II: Soil Properties. † These values are given as the similar properties soil modulus of cohesion k_c and soil modulus of friction k_ϕ in [11], but are converted using $k_1 = k_c b_w^{n_s-1}$ and $k_2 = k_\phi b_w^{n_s-1}$.

D. Numerical Evaluation of Traction Parameters

Equations 4, 5, and 6 allow the stress distributions to be calculated given soil properties, wheel-soil contact angle θ_0 and slip i. A large set of points can be sampled for Riemann sum integration, or a total of five points can be sampled for Simpson's rule integration. An example stress distribution and its approximation using five points is shown in Figure 3.

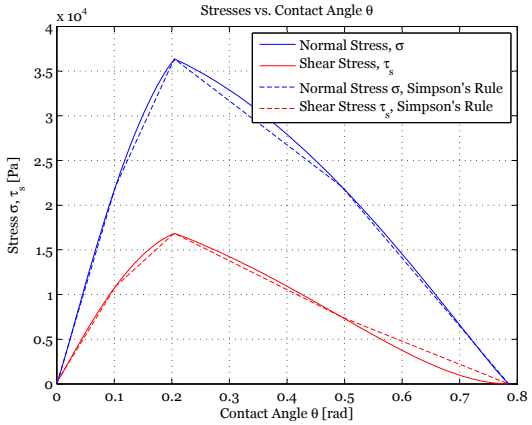


Fig. 3: An example stress distribution using the *Kapvik* wheel dimensions in Table I, soil properties in Table II, a wheel contact angle θ_0 of 45°, and a slip i of 0.25. Note the distinct transition between the forward and rearward regions.

A root finding method in MATLAB is used to find the smallest angle θ_0 that will support a given normal load W at a given slip i. Drawbar pull DP and resistive torque τ_R are then evaluated by performing Riemann sum or Simpson's

rule integration of Equations 8 and 9 around the wheel rim. Simpson's rule integration provides an accurate result with a decreased computational cost. The full procedure for finding the traction parameters is outlined in Algorithm 1.

Algorithm 1 Calculate drawbar pull DP and resistive torque τ_R given normal load W, wheel centroid velocity v_w and wheel angular velocity ω.

1: Calculate wheel slip/skid given v_w and ω using Equation 1 or 2.
2: If the wheel is skidding, use Equations 10 and 11 to convert the problem into one for which slip i is positive.
3: Calculate wheel-soil contact angle θ_0 by finding the smallest root of Equation 7 over the interval $0 < \theta_0 < \pi/2$.
4: Compute the normal stresses σ and shear stresses τ_s at sample points, using Equations 4, 5, and 6.
5: Using either Riemann sums or Simpson's rule, perform the integrals of shear stress in Equations 8 and 9 to calculate the drawbar pull DP and the resistive torque τ_R; use the points sampled in the previous step.

It can be seen from this development that drawbar pull DP and resistive torque τ_R are exclusive functions of normal load W and slip/skid i when the soil properties are known. Example plots are shown in Figure 4 for a normal load W of 50 N.

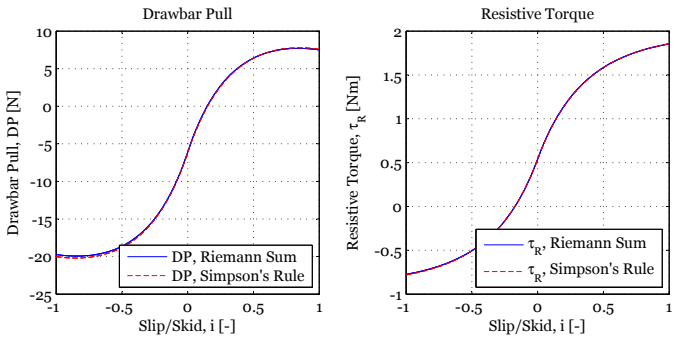


Fig. 4: The relationship between drawbar pull DP (left), resistive torque τ_R (right), and slip i for normal load $W = 50\,\text{N}$.

E. Equation Simplification

Both drawbar pull DP and resistive torque τ_R as a function of normal load W and slip i can be approximated by O^{th} order two dimensional polynomial fits. The fits are represented with $(O+1)(O+2)/2$ coefficients p_{kl}, where p_{kl} is the coefficient for the term $i^k W^l$. Simplifying the equations in this way greatly expedites dynamic rover simulations, since finding the root of Equation 7 is computationally expensive; it also allows the terrain response relationships to be reconstructed from estimated data.

Using the soil properties defined in Table II, a 100×100 array of results was calculated over the parameter space $0 \le i \le 1$ and $0.1\,\text{N} \le W \le 200\,\text{N}$. The publicly available `polyfitweighted2` MATLAB function was used to find 4th order polynomial fits to the data. A 100×100 array of results was again calculated, this time using the polynomial model. The root mean square (RMS) errors for the 10,000 calculated data points for the drawbar pull and resistive torque 4th order approximations over this parameter space were 0.1080 N and 0.008033 N m respectively, for calculated values spanning -9.406 N to 10.195 N and 0 N m to 7.874 N m. The error distributions over the parameter space are shown in Figure 5. The largest errors are at the outer edges of the parameter space; however, even at their maximums, the errors induced by using the polynomial fits are acceptable. The 4th order fits were used herein for simulation. Polynomial fits for resistive torque and drawbar pull – not shown in this paper – were performed using different soil properties to ensure that the suitability of approximating these functions with a polynomial was not soil property dependent.

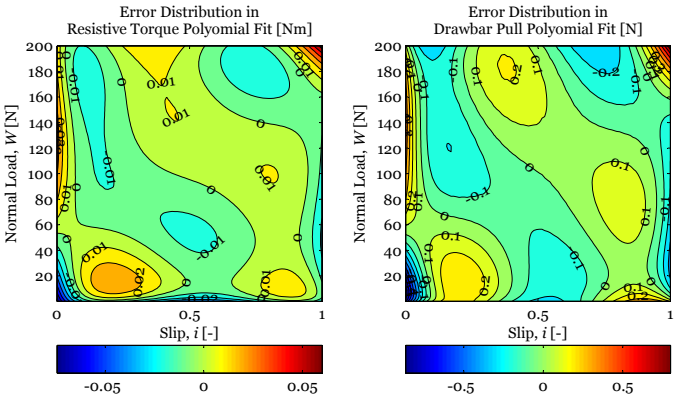


Fig. 5: Contour plots showing the error distribution in 4th order polynomial approximated values of resistive torque (left) and drawbar pull (right) when compared with the results using Wong's classical terramechanics models.

III. Multibody Dynamics

A rover with a rocker-bogie mobility system comprises an articulated multibody system with a moving base [5]. Lagrangian mechanics were used to perform a two dimensional simulation of the *Kapvik* micro-rover driving over rolling terrain. The two simulation dimensions are the world horizontal X_w and the world vertical Z_w, chosen so that the rover can be simulated traversing slopes. A single side of the rocker-bogie mobility system was considered. This is equivalent to the three dimensional case for which the terrain extends infinitely into the page – a good approximation for slowly changing terrain in which the roll of the rover is negligible. The simulation methodology used in this paper closely resembles that used by Ding et al. in their high-fidelity simulation of rover dynamics [5].

A. Coordinates, Naming Conventions, and Configuration

The coordinates and naming conventions of the different joints are shown in Figure 6. The combined center of mass of the body and the rocker is denoted *0*; the rocker "joint" is denoted *r*; the bogie joint denoted *1*; the wheel joints are denoted *2*, *3*, and *4*. A vector $\boldsymbol{l_{kl}}$ represents the vector from k to l when the rover is resting on level ground. The vector $\boldsymbol{c_{11}}$ represents the vector from joint *1* to its center of mass. The wheel centers of mass are located at the wheel joints, so $\boldsymbol{c_{22}} = \boldsymbol{c_{33}} = \boldsymbol{c_{44}} = \boldsymbol{0}$. The angular displacement of the rover

body from the horizontal is given by θ_{0_y}. The joint angles of the bogie and the three wheels are $\{q_1, q_2, q_3, q_4\}$ respectively. The wheel-terrain contact angles with respect to the world horizontal are given by $\{\gamma_1, \gamma_2, \gamma_3\}$. In deformable terrain, the wheel does not actually have a single discrete contact point. However, in simulation, a single effective point of contact dictating the rovers dynamics can be used [1].

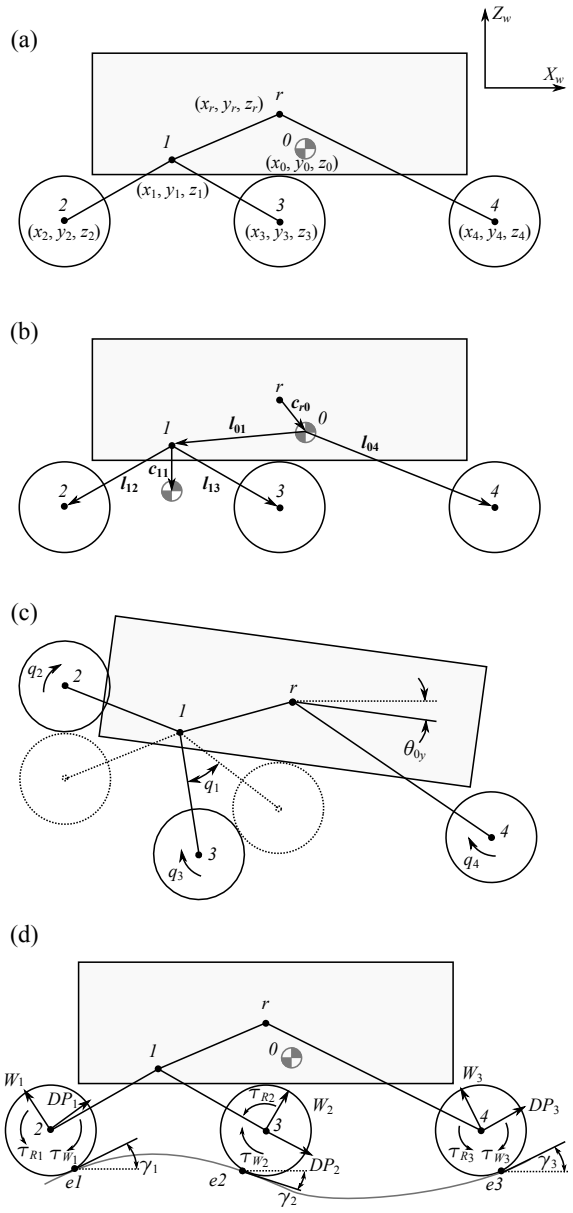


Fig. 6: Two dimensional rocker bogie system configuration variables, internal vectors, and forces.

The configuration of the rover at any given time can be fully described by the configuration vector $\mathbf{\Phi}$.

$$\mathbf{\Phi} = \begin{bmatrix} x_0 & y_0 & z_0 & \theta_{0_x} & \theta_{0_y} & \theta_{0_z} & q_1 & q_2 & q_3 & q_4 \end{bmatrix}^T \tag{12}$$

where $\{x_0, y_0, z_0\}$ are the Cartesian coordinates of the combined body/rocker center of mass in the world frame, $\{\theta_{0_x}, \theta_{0_y}, \theta_{0_z}\}$ are the rotation angles of the body about the $\{X_w, Y_w, Z_w\}$ axes respectively, and $\{q_1, q_2, q_3, q_4\}$ are the joint angles. Note that coordinates y_0, θ_{0_x}, and θ_{0_z} are not required to describe the two dimensional system, but are included to facilitate easy rotations and cross products.

B. Simulation Methodology

The Lagrangian of a system L is defined as:

$$L = T - V \tag{13}$$

The term T represents the kinetic energy of the system and for the rover is given by:

$$T = \frac{1}{2}\dot{\mathbf{\Phi}}^T \left(\sum_{i=0}^{n_v} m_i \boldsymbol{J_{v_i}}^T \boldsymbol{J_{v_i}} + \boldsymbol{J_{\omega_i}}^T \boldsymbol{I_i} \boldsymbol{J_{\omega_i}} \right) \dot{\mathbf{\Phi}}$$
$$T = \frac{1}{2}\dot{\mathbf{\Phi}}^T \boldsymbol{H} \dot{\mathbf{\Phi}} \tag{14}$$

where the Jacobian matrices of velocities $\boldsymbol{v_i}$ and $\boldsymbol{\omega_i}$ with respect to generalized velocity $\dot{\mathbf{\Phi}}$ are given by $\boldsymbol{J_{v_i}}$ and $\boldsymbol{J_{\omega_i}}$ respectively, $\boldsymbol{H}$ is the system inertia matrix, and the number of joints n_v is equal to 4 (the rocker "joint" r is considered rigid because of the two dimensional nature of the simulation).

The term V represents the gravitational potential energy of the system.

$$V = g \begin{bmatrix} z_0 & z_1 & \dots & z_{n_v} \end{bmatrix} \begin{bmatrix} m_0 & m_1 & \dots & m_{n_v} \end{bmatrix}^T \tag{15}$$

where g is the local gravitational acceleration, $\{z_0, \dots, z_{n_v}\}$ are the heights of the mass centers of each body in world coordinates, and $\{m_0, \dots, m_{n_v}\}$ are the masses of each body.

The linear and angular velocity of the center of mass of link i are respectively given by:

$$\boldsymbol{v_i} = \boldsymbol{v_0} + \boldsymbol{\omega_0} \times (\boldsymbol{p_{c_i}} - \boldsymbol{r_0}) + \sum_{j=1}^{n_v} \boldsymbol{L}_{i,j} \boldsymbol{N_j} \times (\boldsymbol{p_{c_i}} - \boldsymbol{p_j}) \dot{q}_j \tag{16}$$

$$\boldsymbol{\omega_i} = \boldsymbol{\omega_0} + \sum_{j=1}^{n_v} \boldsymbol{L}_{i,j} \boldsymbol{N_j} \dot{q}_j \tag{17}$$

where $\boldsymbol{L}$ is an array for which $\boldsymbol{L}_{i,j} = 1$ if j is on the kinematic chain from *0* to i and $\boldsymbol{L}_{i,j} = 0$ otherwise, $\boldsymbol{N_j}$ is the normal vector along the axis of rotation of joint j, and $\dot{q}_j$ is angular rate of joint j.

A skew symmetric matrix can be used to turn the cross product into matrix multiplication. For two vectors $\boldsymbol{a} = [a_1 \;\; a_2 \;\; a_3]^T$ and $\boldsymbol{b} = [b_1 \;\; b_2 \;\; b_3]^T$, a skew symmetric matrix $\tilde{\boldsymbol{a}}$ can be constructed with the following properties [5]:

$$\boldsymbol{a} \times \boldsymbol{b} = \tilde{\boldsymbol{a}}\boldsymbol{b} = \begin{bmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{bmatrix} \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix}$$
$$\boldsymbol{b} \times \boldsymbol{a} = -\tilde{\boldsymbol{a}}\boldsymbol{b} = \tilde{\boldsymbol{a}}^T \boldsymbol{b} \tag{18}$$

Using the insights above, the Jacobians describing the velocity of the center of mass and angular velocity for a single

link can be found such that $\boldsymbol{v_i} = \boldsymbol{J_{v_i}}\dot{\boldsymbol{\Phi}}$ and $\boldsymbol{\omega_i} = \boldsymbol{J_{\omega_i}}\dot{\boldsymbol{\Phi}}$.

$$\boldsymbol{J_{v_i}} = \Big[\boldsymbol{I}_{3\times3}\ \tilde{\boldsymbol{r}}_{\boldsymbol{0i}}{}^{T}\ \boldsymbol{L}_{i,1}\boldsymbol{N_1} \times (\boldsymbol{p_{c_i}} - \boldsymbol{p_1})\ \ \ldots\ \ \ldots\ \boldsymbol{L}_{i,4}\boldsymbol{N_4} \times (\boldsymbol{p_{c_i}} - \boldsymbol{p_4})\Big] \tag{19}$$

$$\boldsymbol{J_{\omega_i}} = \Big[\boldsymbol{0}_{3\times3}\ \boldsymbol{I}_{3\times3}\ \boldsymbol{L}_{i,1}\boldsymbol{N_1}\ \ \ldots\ \ \boldsymbol{L}_{i,4}\boldsymbol{N_4}\Big] \tag{20}$$

where $\boldsymbol{I}_{p\times p}$ is a $p \times p$ identity matrix, $\boldsymbol{0}_{p\times p}$ is a $p \times p$ zero matrix, and $\boldsymbol{r_{0i}} = \boldsymbol{p_{c_i}} - \boldsymbol{r_0}$ is the vector from the combined body/rocker center of mass *0* to the center of mass of link i.

Stacking the Jacobians creates a link Jacobian $\boldsymbol{J_a}$, which when multiplied by the generalized velocity $\dot{\boldsymbol{\Phi}}$, gives a vector containing the full set of link center of mass velocities $\dot{\boldsymbol{X}}_{\boldsymbol{a}}$.

$$\dot{\boldsymbol{X}}_{\boldsymbol{a}} = \begin{bmatrix} \boldsymbol{v_1} \\ \boldsymbol{\omega_1} \\ \vdots \\ \boldsymbol{v_{n_v}} \\ \boldsymbol{\omega_{n_v}} \end{bmatrix} = \boldsymbol{J_a}\dot{\boldsymbol{\Phi}} = \begin{bmatrix} \boldsymbol{J_{v_1}} \\ \boldsymbol{J_{\omega_1}} \\ \vdots \\ \boldsymbol{J_{v_{n_v}}} \\ \boldsymbol{J_{\omega_{n_v}}} \end{bmatrix} \dot{\boldsymbol{\Phi}} \tag{21}$$

The velocity and angular velocity Jacobians for the body *0* are easily deduced since its velocity and angular velocity are explicitly included in the generalized velocity $\dot{\boldsymbol{\Phi}}$.

$$\boldsymbol{J_{v_0}} = \begin{bmatrix}\boldsymbol{I}_{3\times3} & \boldsymbol{0}_{3\times3} & \boldsymbol{0}_{3\times4}\end{bmatrix} \tag{22}$$

$$\boldsymbol{J_{\omega_0}} = \begin{bmatrix}\boldsymbol{0}_{3\times3} & \boldsymbol{I}_{3\times3} & \boldsymbol{0}_{3\times4}\end{bmatrix} \tag{23}$$

The system inertia matrix $\boldsymbol{H}$ can then be calculated using the term in parentheses in Equation 14.

The Euler-Lagrange equation gives utility to the Lagrangian:

$$\frac{d}{dt}\frac{\partial L}{\partial \dot{\boldsymbol{\Phi}}} - \frac{\partial L}{\partial \boldsymbol{\Phi}} = \boldsymbol{H}\ddot{\boldsymbol{\Phi}} + \dot{\boldsymbol{H}}\dot{\boldsymbol{\Phi}} - \frac{\partial L}{\partial \boldsymbol{\Phi}} = \boldsymbol{\tau_g} \tag{24}$$

where the system has n degrees of freedom, and $\boldsymbol{\tau_g}$ is a vector of generalized forces acting on the system.

A continuously differentiable function $z_{wc}(x_{wc})$ is created to constrain the path of the wheel centroids. The effective terrain (i.e. the path traced by the effective terrain contact points as in [1]) is formed by offsetting the wheel centroid path by the wheel radius r_w. Given a wheel centroid path $z_{wc}(x_{wc})$, the corresponding terrain coordinates x_t and z_t are given by:

$$\begin{bmatrix} x_t \\ z_t \end{bmatrix} = \begin{bmatrix} x_{wc} \\ z_{wc}(x_{wc}) \end{bmatrix} + r_w \frac{\left[\frac{dz_{wc}}{dx_{wc}}|_{x_{wc}}, -1\right]^T}{\sqrt{1 + \frac{dz_{wc}}{dx_{wc}}^2}} \tag{25}$$

The wheel centroids are constrained to the wheel centroid path using a vector of constraints $\boldsymbol{\Psi}$.

$$\boldsymbol{\Psi} = \begin{bmatrix}\Psi_1 \\ \Psi_2 \\ \Psi_3\end{bmatrix} = \begin{bmatrix} p_{2_z} - z_{wc}(p_{2_x}) \\ p_{3_z} - z_{wc}(p_{3_x}) \\ p_{4_z} - z_{wc}(p_{4_x}) \end{bmatrix} = \boldsymbol{0}_{3\times1} \tag{26}$$

where p_{i_x} and p_{i_z} are the x and z components of joint i's position respectively.

Algebraic constraints $\boldsymbol{\Psi}$ to the position, velocity, and acceleration of the system respectively can be implemented as follows [13]:

$$\boldsymbol{\Psi}(\boldsymbol{\Phi}, t) = 0 \qquad | \ \boldsymbol{\Psi} \in R^{n_c} \tag{27}$$

$$\dot{\boldsymbol{\Psi}}(\boldsymbol{\Phi}, t) = \boldsymbol{\Psi}_t(\boldsymbol{\Phi}, t) + \boldsymbol{\Psi}_{\boldsymbol{\Phi}}(\boldsymbol{\Phi}, t)\dot{\boldsymbol{\Phi}} = 0 \,|\, \boldsymbol{\Psi}_{\boldsymbol{\Phi}} \in R^{n_c \times n} \tag{28}$$

$$\ddot{\boldsymbol{\Psi}}(\boldsymbol{\Phi}, t) = \dot{\boldsymbol{\Psi}}_t(\boldsymbol{\Phi}, t) + \dot{\boldsymbol{\Psi}}_{\boldsymbol{\Phi}}(\boldsymbol{\Phi}, t)\dot{\boldsymbol{\Phi}} + \boldsymbol{\Psi}_{\boldsymbol{\Phi}}(\boldsymbol{\Phi}, t)\ddot{\boldsymbol{\Phi}} = 0 \tag{29}$$

where $\boldsymbol{\Psi}_{\boldsymbol{\Phi}}$ is the Jacobian of the constraints $\boldsymbol{\Psi}$ with respect to the configuration vector $\boldsymbol{\Phi}$, $\boldsymbol{\Psi}_t$ is the derivative of the constraints $\boldsymbol{\Psi}$ with respect to time t, and $n_c = 3$ is the number of constraints on the system.

The generalized forces $\boldsymbol{\tau_g}$ consist of applied forces $\boldsymbol{\tau_a}$ and constraint forces $\boldsymbol{\tau_c}$. The constraint forces deliver zero power, represent the effect of the normal load reaction forces in generalized coordinates, and can be set to [14]:

$$\boldsymbol{\tau_c} = \boldsymbol{\Psi_{\Phi}}^T \boldsymbol{\lambda} \tag{30}$$

where $\boldsymbol{\lambda} \in R^{n_c}$ is a set of Lagrange multipliers $\{\lambda_1, \ldots, \lambda_{n_c}\}$ that form a linear combination of the columns of $\boldsymbol{\Psi_{\Phi}}$ and allow the constraints to be satisfied. Note that with the constraints imposed, $\boldsymbol{\lambda}_i$ represents the Z_w component of the normal reaction force W_z for wheel i; the X_w component for wheel i can also be found: $W_x = \boldsymbol{\Psi}_{\boldsymbol{\Phi}i,1}\boldsymbol{\lambda}_i$.

To ensure that the roundoff error does not cause violation of the constraints, Baumgarte stabilization is used [13]. Using Baumgarte stabilization, Equation 29 is replaced with an equation that incorporates both $\dot{\boldsymbol{\Psi}}$ and $\boldsymbol{\Psi}$:

$$\ddot{\boldsymbol{\Psi}}(\boldsymbol{\Phi}, t) + 2\alpha\dot{\boldsymbol{\Psi}}(\boldsymbol{\Phi}, t) + \beta^2\boldsymbol{\Psi}(\boldsymbol{\Phi}, t) = 0 \tag{31}$$

where α and β are appropriately chosen constants.

Combining Equations 24, 27, 28, 29, 30 and 31 into a single system of equations, the dynamic system is described by:

$$\begin{bmatrix} \boldsymbol{H} & \boldsymbol{\Psi_{\Phi}}^T \\ \boldsymbol{\Psi_{\Phi}} & \boldsymbol{0} \end{bmatrix} \begin{bmatrix} \ddot{\boldsymbol{\Phi}} \\ -\boldsymbol{\lambda} \end{bmatrix} = \begin{bmatrix} \boldsymbol{\tau_a} - \dot{\boldsymbol{H}}\dot{\boldsymbol{\Phi}} + \frac{\partial L}{\partial \boldsymbol{\Phi}} \\ \zeta \end{bmatrix} \tag{32}$$

$$\zeta = -\dot{\boldsymbol{\Psi}}_t - \dot{\boldsymbol{\Psi}}_{\boldsymbol{\Phi}}\dot{\boldsymbol{\Phi}} - 2\alpha\left(\boldsymbol{\Psi}_t + \boldsymbol{\Psi}_{\boldsymbol{\Phi}}\dot{\boldsymbol{\Phi}}\right) - \beta^2\boldsymbol{\Psi}$$

If the n_c constraint equations are independent, Equation 32 can be solved through matrix inversion [13].

In order to solve this system of equations, the following variables are required: $\boldsymbol{H}$, $\dot{\boldsymbol{H}}$, $\boldsymbol{\Psi}$, $\boldsymbol{\Psi}_t$, $\dot{\boldsymbol{\Psi}}_t$, $\boldsymbol{\Psi}_{\boldsymbol{\Phi}}$, $\dot{\boldsymbol{\Psi}}_{\boldsymbol{\Phi}}$, and $\frac{\partial L}{\partial \boldsymbol{\Phi}}$. Equations for these variables were found but are not included herein. The term $\frac{\partial T}{\partial \boldsymbol{\Phi}}$ in $\frac{\partial L}{\partial \boldsymbol{\Phi}}$ was found to be negligible for a slowly moving rover and was excluded from the simulation in order to reduce the simulation time by approximately one third.

The set of generalized applied forces $\boldsymbol{\tau_a}$ consists of the effects of: resistive wheel torques $\boldsymbol{\tau_R}$, drawbar pull $\boldsymbol{\tau_{DP}}$, and internal wheel torques $\boldsymbol{\tau_W}$. Generalized forces represent forces

or moments in the coordinates of the configuration vector $\boldsymbol{\Phi}$.

$$\boldsymbol{\tau_a} = \boldsymbol{\tau_R} + \boldsymbol{\tau_{DP}} + \boldsymbol{\tau_W} \tag{33}$$

$$\boldsymbol{\tau_R} = \boldsymbol{J_w}^T \boldsymbol{F_{\tau_R}} \tag{34}$$

$$\boldsymbol{F_{\tau_R}} = \begin{bmatrix} \boldsymbol{F_{\tau_{R1}}} & \boldsymbol{F_{\tau_{R2}}} & \boldsymbol{F_{\tau_{R3}}} \end{bmatrix}^T \tag{35}$$

$$\boldsymbol{F_{\tau_{Ri}}} = \begin{bmatrix} 0 & 0 & 0 & 0 & -\tau_{R_i} & 0 \end{bmatrix} \tag{36}$$

$$\boldsymbol{\tau_{DP}} = \boldsymbol{J_w}^T \boldsymbol{F_{\tau_{DP}}} \tag{37}$$

$$\boldsymbol{F_{\tau_{DP}}} = \begin{bmatrix} \boldsymbol{F_{\tau_{DP1}}} & \boldsymbol{F_{\tau_{DP2}}} & \boldsymbol{F_{\tau_{DP3}}} \end{bmatrix}^T \tag{38}$$

$$\boldsymbol{F_{\tau_{DP}i}} = \begin{bmatrix} DP_i \cos\gamma_i & 0 & -DP_i \sin\gamma_i & 0 & 0 & 0 \end{bmatrix} \tag{39}$$

$$\boldsymbol{\tau_W} = \begin{bmatrix} \mathbf{0}_{1\times 7} & (\tau_{W_1} - b_d\dot{q}_2) & (\tau_{W_2} - b_d\dot{q}_3) & (\tau_{W_3} - b_d\dot{q}_4) \end{bmatrix}^T \tag{40}$$

where $\boldsymbol{F_{\tau_{Ri}}}$ and $\boldsymbol{F_{DP_i}}$ are the set of external forces and moments in world coordinates that are applied to the centroid of wheel i; they have the form $\boldsymbol{F_{ext}} = \begin{bmatrix} F_x & F_y & F_z & M_x & M_y & M_z \end{bmatrix}$ and are projected onto generalized coordinates using a Jacobian $\boldsymbol{J_w}$; τ_{R_i} and DP_i are the resistive torque and drawbar pull respectively on wheel i given the current normal load W_i and slip i_i; b_d is a viscous damping coefficient; and τ_{W_i} is the motor torque times the gear ratio and transmission efficiency, applied to wheel i.

C. Simulation Implementation

The result of solving Equation 32 is a value for second time derivative of configuration $\ddot{\boldsymbol{\Phi}}$. To propagate the configuration forward, a configuration state vector $\boldsymbol{\Phi_s}$ is required:

$$\boldsymbol{\Phi_s} = \begin{bmatrix} \dot{\boldsymbol{\Phi}} \\ \boldsymbol{\Phi} \end{bmatrix} \qquad \dot{\boldsymbol{\Phi}}_{\boldsymbol{s}} = \begin{bmatrix} \ddot{\boldsymbol{\Phi}} \\ \dot{\boldsymbol{\Phi}} \end{bmatrix} \tag{41}$$

Each evaluation of $\dot{\boldsymbol{\Phi}}_{\boldsymbol{s}}$ is function of $\boldsymbol{\Phi_s}$, the applied wheel torques $\{\tau_{W_1}, \tau_{W_2}, \tau_{W_3}\}$, and the previously evaluated generalized constraint forces $\boldsymbol{\tau_c}$. Numerical integration is used to propagate the configuration state vector $\boldsymbol{\Phi_s}$ forward. A block diagram of the simulation implementation is shown in Figure 7.

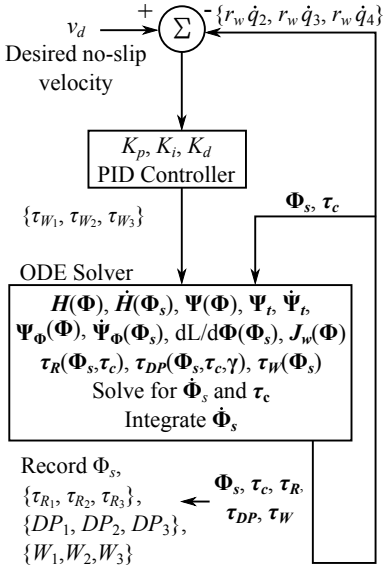


Fig. 7: Block diagram of simulation implementation.

A proportional integral derivative (PID) controller was used to individually control the wheel speeds so that the approximate no-slip wheel velocities $\{r_w\dot{q}_2, r_w\dot{q}_3, r_w\dot{q}_4\}$ were equal to the desired velocity v_d. The controller was run at a frequency of 50 Hz; the configuration state vector $\boldsymbol{\Phi_s}$ and all values of interest were also recorded at this frequency. Because of the sensitivity of drawbar pull DP and resistive torque τ_R to slip i, the MATLAB ordinary differential equation solver `ode45` was used, with default settings, for integration between time steps. Over the duration of a single time step the control input torques $\{\tau_{W_1}, \tau_{W_2}, \tau_{W_3}\}$ and normal loads $\{W_1, W_2, W_3\}$ were presumed to be constant.

The parameters used in simulation were selected so as to represent one side of the *Kapvik* micro-rover and are shown in Table III. Inertias were taken about the Y_w axis only. The wheel dimensions from Table I were used.

IV. Terrain Response Estimation

This section develops the estimation of resistive torque τ_R, drawbar pull DP, normal load W and slip i. It is assumed that there is no *a priori* knowledge of soil properties or terrain response; the estimator builds a terrain response model from scratch. Estimation is accomplished through the use of two Unscented Kalman Filters [15] running simultaneously. A total of three estimators are presented here. Estimator 1, with state vector $\boldsymbol{x_1}$, is a general estimator which estimates a large set of states essential for determining the terrain response, including resistive torques $\{\tau_{R_1}, \tau_{R_2}, \tau_{R_3}\}$ and slips $\{i_1, i_2, i_3\}$. It is run regardless of the type of force sensors available. Estimator 2, with state vector $\boldsymbol{x_2}$, estimates the normal loads $\{W_1, W_2, W_3\}$ and the drawbar pulls $\{DP_1, DP_2, DP_3\}$ in the case where two-axis force sensors are available. Estimator 3, with state vector $\boldsymbol{x_3}$ estimates the normal loads $\{W_1, W_2, W_3\}$ and the drawbar pulls $\{DP_1, DP_2, DP_3\}$ in the case where single-axis force sensors are available. The linearized observability of the filters has been proved, but is not included herein.

A set of sensor measurements $\boldsymbol{z}$ is taken; each is assumed to be a Gaussian random variable. The sensor measurements are used to create the measurement vector $\boldsymbol{y_1}$ for use in Estimator 1. Some of the measurements in $\boldsymbol{y_1}$ are taken directly from the sensor measurements, and some are calculated using a combination of sensor measurements. The measurement vectors $\boldsymbol{y_2}$ and $\boldsymbol{y_3}$, used in Estimators 2 and 3 respectively, contain sensor measurements as well as measurements calculated using a combination of state estimates from Estimator 1.

A. Sensors and Sensor Measurements

To perform accurate terrain response estimation, a comprehensive suite of sensors is required. Fortunately, several of these sensors are typically included on planetary rover platforms. The required measurements include wheel angular velocities, bogie angle, rover accelerations, rover pitch rate, forward velocity, and forces above the wheel hubs (either two-axis or single-axis). The set of measurements is illustrated in Figure 8. In this paper a measurement of x is denoted $\breve{x}$ and the current time step is denoted k.

Parameter	Value	Units	Source
Mass of body and rocker m_0	12.49	kg	SolidWorks model
Mass of link 1 m_1	1.562	kg	SolidWorks model
Mass of a wheel m_2, m_3, m_4	0.3174	kg	SolidWorks model
Inertia of body and rocker I_0	0.5045	kg m^2	SolidWorks model
Inertia of link 1 I_1	0.05248	kg m^2	SolidWorks model
Wheel inertia I_w	0.003584	kg m^2	SolidWorks model and drivetrain specs
Wheel and gear inertia referred to wheel I_{wg}	1.082	kg m^2	SolidWorks model and drivetrain specs
Wheel angular viscous damping coefficient b_d	2.28855	N m s	Experiment
Vector $\boldsymbol{c_{11}}$	$\begin{bmatrix}0 & 0 & -0.07322\end{bmatrix}^T$	m	SolidWorks model
Vector $\boldsymbol{c_{22}} = \boldsymbol{c_{33}} = \boldsymbol{c_{44}}$	$\begin{bmatrix}0 & 0 & 0\end{bmatrix}^T$	m	SolidWorks model
Baumgarte stabilization constant α	500	-	Trial and error
Baumgarte stabilization constant β	500	-	Trial and error
Desired velocity v_d	2	cm/s	Design speed
Proportional control gain K_p	25	-	Trial and error
Integral control gain K_i	0.5	-	Trial and error
Derivative control gain K_d	12.5	-	Trial and error

TABLE III: Parameters used in simulation.

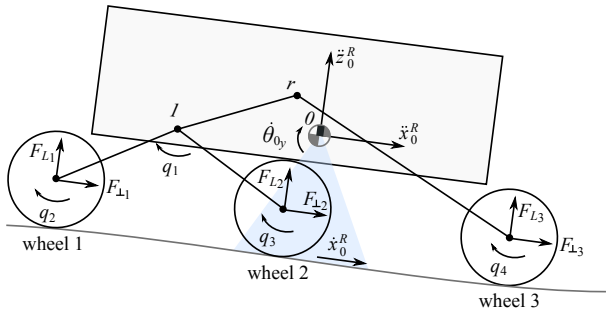


Fig. 8: Sensor measurements.

Wheel angle measurements $\{\breve{q}_2, \breve{q}_3, \breve{q}_4\}$ are provided by motor encoders with a quadrature output which provide the number of counts traversed by the wheel. This is converted into wheel angle given the number of counts per turn and the gear ratio. Bogie angle measurement $\breve{q}_1$ is provided by a hollow shaft grey code absolute encoder. Measured rover accelerations in rover coordinates $\{\breve{\ddot{x}}_0^R, \breve{\ddot{z}}_0^R\}$ and measured rover pitch rate $\breve{\dot{\theta}}_{0_y}$ are provided by an inertial measurement unit (IMU) mounted to the rover body. A measurement of rover body velocity along the rover's forward direction $\breve{\dot{x}}_0^R$ could be taken using a vision-based velocimeter under development at Carleton University. The velocimeter uses a downward facing camera placed at the rover's center of mass to take a sequence of images; optic flow is then used to deduce the rover velocity. A distance sensor beside the camera could be used to scale the measured optic flow from camera to world coordinates; the distance sensor would make the velocity measurement independent of wheel sinkage depth and rover pitch, which could alter the distance from the camera to the terrain. Note that the body velocity in the rover's z direction $\dot{z}_0^R$ is not measured. The rover is equipped with either single-axis (e.g. *Kapvik*) or two-axis force sensors placed above the wheels. These sensors measure either the forces $\{\breve{F}_{L_1}, \breve{F}_{L_2}, \breve{F}_{L_3}\}$ (single-axis) or the forces $\{\breve{F}_{L_1}, \breve{F}_{L_2}, \breve{F}_{L_3}, \breve{F}_{\perp_1}, \breve{F}_{\perp_2}, \breve{F}_{\perp_3}\}$ (two-axis) shown in Figure 8. Either type of sensor can be used in terrain response estimation, but performance is better using a two-axis force sensor. These measurements form a vector of sensor measurements $\boldsymbol{z}$.

$$\boldsymbol{z} = \begin{bmatrix}\breve{q}_1 & \breve{\dot{\theta}}_{0_y} & \breve{\dot{x}}_0^R & \breve{\ddot{x}}_0^R & \breve{\ddot{z}}_0^R & \breve{q}_2 & \breve{q}_3 & \breve{q}_4 & \dots \\ \dots & \breve{F}_{L_1} & \breve{F}_{L_2} & \breve{F}_{L_3} & \breve{F}_{\perp_1} & \breve{F}_{\perp_2} & \breve{F}_{\perp_3}\end{bmatrix}^T \quad (42)$$

The expected sensor noises are shown in Table IV.

Measured Variable *	Sensor	Noise Std. Dev. (σ_*)	Units
Bogie angle $\breve{q}_1$	16-bit Absolute encoder	9.155×10^{-4}	°
Pitch rate $\breve{\dot{\theta}}_{0_y}$	H3-IMU-HP02-0300	0.56	°/s
Velocity in rover coordinates $\breve{\dot{x}}_0^R$	Velocimeter	1	mm/s
Accelerations in rover coordinates $\{\breve{\ddot{x}}_0^R, \breve{\ddot{z}}_0^R\}$	H3-IMU-HP02-0300	1.245×10^{-3}	m^2/s
Wheel positions $\{\breve{q}_2, \breve{q}_3, \breve{q}_4\}$	Motor encoders †	2.143×10^{-5}	°
Wheel vertical loads $\{\breve{F}_{L_1}, \breve{F}_{L_2}, \breve{F}_{L_3}\}$	Force sensor ‡	3.255×10^{-2}	N
Wheel perpendicular loads $\{\breve{F}_{\perp_1}, \breve{F}_{\perp_2}, \breve{F}_{\perp_3}\}$	Force sensor ‡	3.255×10^{-2}	N

TABLE IV: Sensor measurement noises. † 500 count/turn 3-channel quadrature encoders with 1400:1 gear ratio. ‡ 200 N full-scale force sensor sampled by a 10-bit analog to digital converter.

B. Estimator 1: The General Estimator

The general estimator estimates 22 states related to terrain response estimation. In this paper an estimate of x is denoted $\hat{x}$.

The states estimated by the general estimator are:

$$\boldsymbol{x_1} = \Big[\dot{q}_2\ \dot{q}_3\ \dot{q}_4\ \tau_{R_1}\ \tau_{R_2}\ \tau_{R_3}\ \theta_{0_y}\ q_1\ \dot{\theta}_{0_y}\ \dot{q}_1\ \ddot{\theta}_{0_y}\ \ddot{q}_1\ \dots \\ \dots\ \ddot{x}_0\ \ddot{z}_0\ M_{0f_w}\ M_{1f_w}\ F_{xf_w}\ F_{zf_w}\ \dot{x}_0^R\ i_1\ i_2\ i_3\Big]^T \quad (43)$$

where $\{M_{0f_w}, M_{1f_w}, F_{xf_w}, F_{zf_w}\}$ are the net effects of the wheel forces (the moment exerted about the lumped body/rocker center of mass *0*, the moment exerted about the

bogie joint *1*, the net force in the world horizontal direction X_w, and the net force in the world vertical direction Z_w respectively), and $\{i_1, i_2, i_3\}$ are the wheel slip(s)/skid(s). Note that $\{M_{0f_w}, M_{1f_w}, F_{xf_w}, F_{zf_w}\}$ describe the net effects of wheel *linear* forces, but do not include the effects of wheel resistive torques.

An estimate of configuration $\hat{\mathbf{\Phi}}$ can be calculated for subsequent use. The wheel angles $\{q_2, q_3, q_4\}$ and the position of the rover in the world frame $\{x_0, y_0, z_0\}$ are irrelevant in the subsequent applications of $\hat{\mathbf{\Phi}}$. Thus, the estimated configuration $\hat{\mathbf{\Phi}}$ is given by:

$$\hat{\mathbf{\Phi}}_k = \begin{bmatrix} 0 & 0 & 0 & 0 & \hat{\theta}_{0_{y_k}} & 0 & \hat{q}_{1_k} & 0 & 0 & 0 \end{bmatrix}^T \tag{44}$$

Similarly, the estimated generalized velocity $\hat{\dot{\mathbf{\Phi}}}$ is given by:

$$\hat{\dot{\mathbf{\Phi}}}_k = \begin{bmatrix} \hat{\dot{x}}^R_{0_k} \cos\hat{\theta}_{0_{y_k}} & 0 & \hat{\dot{x}}^R_{0_k} \sin\hat{\theta}_{0_{y_k}} & 0 & \dots \\ \dots & \hat{\dot{\theta}}_{0_{y_k}} & 0 & \hat{\dot{q}}_{1_k} & \hat{\dot{q}}_{2_k} & \hat{\dot{q}}_{3_k} & \hat{\dot{q}}_{4_k} \end{bmatrix}^T \tag{45}$$

Note that this estimate of $\hat{\dot{\mathbf{\Phi}}}$ involves the assumption that the velocity in the rover's z direction $\dot{z}^R_0$ is zero. This assumption necessitates that the rover be used only on gently rolling terrain for which $\dot{z}^R_0 \ll \dot{x}^R_0$ and is explored in further detail in Sections IV-F and V-B.

1) Process Model: The process model is a nonlinear discrete-time model. The state is propagated forward using the function f_1:

$$\hat{\boldsymbol{x}}_{\mathbf{1}k+1} = f_1\left(\hat{\boldsymbol{x}}_{\mathbf{1}k}, \boldsymbol{u}_k\right) \tag{46}$$

where $\boldsymbol{u}_k$ is the control vector, equal to the set of input wheel torques:

$$\boldsymbol{u}_k = \begin{bmatrix} \tau_{W_{1_k}} & \tau_{W_{2_k}} & \tau_{W_{3_k}} \end{bmatrix}^T \tag{47}$$

The first three states are wheel angular velocities. Their process model is straightforward, propagating the previous velocity forward by the estimator time step T_e.

$$\hat{\dot{q}}_{i_{k+1}} = \hat{\dot{q}}_{i_k} + \hat{\ddot{q}}_{i_k} T_e \quad | \quad i = \{2, 3, 4\} \tag{48}$$

$$\hat{\ddot{q}}_{i_{k+1}} = \left(\tau_{W_{i-1_k}} - \hat{\tau}_{R_{i-1_k}} - \hat{\dot{q}}_{i_k} b_d\right) {I_{wg}}^{-1} \tag{49}$$

where $\hat{\ddot{q}}_i$ is the estimated angular acceleration of joint i; although not a state, it is used in the process model. Negligible inertial terms in $\boldsymbol{H}$ relating $\ddot{q}_i$ to $\ddot{\theta}_{0_y}$ and $\ddot{q}_1$ are neglected.

The following three states are the wheel resistive torques. As no *a priori* model exists for these states, a random walk is assumed, and the predicted estimate is set to the previous estimate.

$$\hat{\tau}_{R_{i\,k+1}} = \hat{\tau}_{R_{i_k}} \quad | \quad i = \{1, 2, 3\} \tag{50}$$

The process models for θ_{0_y} and q_1 use second order Taylor series approximations; the process models for $\dot{\theta}_{0_y}$, and $\dot{q}_1$ use first order Taylor series approximations.

$$\hat{\theta}_{0_{y_{k+1}}} = \hat{\theta}_{0_{y_k}} + \hat{\dot{\theta}}_{0_{y_k}} T_e + \hat{\ddot{\theta}}_{0_{y_k}} T_e^2/2 \tag{51}$$

$$\hat{q}_{1_{k+1}} = \hat{q}_{1_k} + \hat{\dot{q}}_{1_k} T_e + \hat{\ddot{q}}_{1_k} T_e^2/2 \tag{52}$$

$$\hat{\dot{\theta}}_{0_{y_{k+1}}} = \hat{\dot{\theta}}_{0_{y_k}} + \hat{\ddot{\theta}}_{0_{y_k}} T_e \tag{53}$$

$$\hat{\dot{q}}_{1_{k+1}} = \hat{\dot{q}}_{1_k} + \hat{\ddot{q}}_{1_k} T_e \tag{54}$$

The acceleration states all use a particular row of Equation 32, solved for the state variable of interest. When using the dynamic equations in the estimator, the terms $\dot{\boldsymbol{H}}\dot{\mathbf{\Phi}}$ and $\frac{\partial T}{\partial \mathbf{\Phi}}$ are not included.

$$\hat{\ddot{\theta}}_{0_{y_{k+1}}} = \Big(\hat{M}_{0f_{w_k}} - \hat{\boldsymbol{J}}_{\boldsymbol{w}5,5_k}\hat{\tau}_{R_{1_k}} - \hat{\boldsymbol{J}}_{\boldsymbol{w}11,5_k}\hat{\tau}_{R_{2_k}} \ \dots \\ \dots - \hat{\boldsymbol{J}}_{\boldsymbol{w}17,5_k}\hat{\tau}_{R_{3_k}} - \left[\frac{\partial \hat{V}}{\partial \mathbf{\Phi}}\right]_5 - \hat{\boldsymbol{H}}_{5,1_k}\hat{\dot{x}}_{0k} - \hat{\boldsymbol{H}}_{5,3_k}\hat{\dot{z}}_{0k} \ \dots \\ \dots - \hat{\boldsymbol{H}}_{5,7_k}\hat{\dot{q}}_{1k} - \hat{\boldsymbol{H}}_{5,8_k}\hat{\dot{q}}_{2k} - \hat{\boldsymbol{H}}_{5,9_k}\hat{\dot{q}}_{3k} - \hat{\boldsymbol{H}}_{5,10_k}\hat{\dot{q}}_{4k}\Big)\hat{\boldsymbol{H}}_{5,5_k}{}^{-1} \tag{55}$$

$$\hat{\ddot{q}}_{1_{k+1}} = \Big(\hat{M}_{1f_{w_k}} - \hat{\boldsymbol{J}}_{\boldsymbol{w}5,7_k}\hat{\tau}_{R_{1_k}} - \hat{\boldsymbol{J}}_{\boldsymbol{w}11,7_k}\hat{\tau}_{R_{2_k}} \ \dots \\ \dots - \left[\frac{\partial \hat{V}}{\partial \mathbf{\Phi}}\right]_7 - \hat{\boldsymbol{H}}_{7,1_k}\hat{\dot{x}}_{0k} - \hat{\boldsymbol{H}}_{7,3_k}\hat{\dot{z}}_{0k} \ \dots \\ \dots - \hat{\boldsymbol{H}}_{7,5_k}\hat{\dot{\theta}}_{0_{y\,k}} - \hat{\boldsymbol{H}}_{7,8_k}\hat{\dot{q}}_{2k} - \hat{\boldsymbol{H}}_{7,9_k}\hat{\dot{q}}_{3k}\Big)\hat{\boldsymbol{H}}_{7,7_k}{}^{-1} \tag{56}$$

$$\hat{\dot{x}}_{0_{k+1}} = \Big(\hat{F}_{x_{w_k}} - \hat{\boldsymbol{H}}_{1,5_k}\hat{\dot{\theta}}_{0_{y\,k}} - \hat{\boldsymbol{H}}_{1,7_k}\hat{\dot{q}}_{1k}\Big)\hat{\boldsymbol{H}}_{1,1_k}{}^{-1} \tag{57}$$

$$\hat{\dot{z}}_{0_{k+1}} = \Big(\hat{F}_{z_{w_k}} - \left[\frac{\partial \hat{V}}{\partial \mathbf{\Phi}}\right]_3 - \hat{\boldsymbol{H}}_{3,5_k}\hat{\dot{\theta}}_{0_{y\,k}} - \hat{\boldsymbol{H}}_{3,7_k}\hat{\dot{q}}_{1k}\Big)\hat{\boldsymbol{H}}_{3,3_k}{}^{-1} \tag{58}$$

where $\frac{\partial \hat{V}}{\partial \mathbf{\Phi}}$, $\hat{\boldsymbol{H}}$, and $\hat{\boldsymbol{J}}_{\boldsymbol{w}}$ are obtained using the estimated configuration $\hat{\mathbf{\Phi}}$.

The net moments M_{0f_w} and M_{1f_w} and forces F_{xf_w} and F_{zf_w} resultant from the wheel forces $\{DP_1, DP_2, DP_3, W_1, W_2, W_2\}$ are assumed to follow a random walk. Their process models are then:

$$\hat{M}_{0f_{w_{k+1}}} = \hat{M}_{0f_{w_k}} \qquad \hat{M}_{1f_{w_{k+1}}} = \hat{M}_{1f_{w_k}}$$
$$\hat{F}_{xf_{w_{k+1}}} = \hat{F}_{xf_{w_k}} \qquad \hat{F}_{zf_{w_{k+1}}} = \hat{F}_{zf_{w_k}} \tag{59}$$

The process model for the velocity along the rover $\dot{x}^R_0$ is a first order Taylor series approximation using the estimated acceleration of the rover along the rover body.

$$\hat{\dot{x}}^R_{0\,k+1} = \hat{\dot{x}}^R_{0\,k} + \left(\hat{\ddot{x}}_{0k}\cos\hat{\theta}_{0_{y_k}} - \hat{\ddot{z}}_{0k}\sin\hat{\theta}_{0_{y_k}}\right)T_e \tag{60}$$

The slip(s)/skid(s) of the rover wheels are assumed to follow a random walk.

$$\hat{i}_{i_{k+1}} = \hat{i}_{i_k} \quad | \quad i = \{1, 2, 3\} \tag{61}$$

2) Measurements: The measurement vector $\boldsymbol{y}_1$ for the general estimator is given by:

$$\boldsymbol{y_1} = \begin{bmatrix} \breve{\dot{q}}_2 & \breve{\dot{q}}_3 & \breve{\dot{q}}_4 & \breve{\theta}_{0_y} & \breve{q}_1 & \breve{\ddot{x}}^R_0 & \breve{\ddot{z}}^R_0 & \breve{\dot{x}}^R_0 & \breve{i}_1 & \breve{i}_2 & \breve{i}_3 \end{bmatrix}^T \tag{62}$$

The measurements of wheel angular velocities are obtained by applying the first order finite difference method to the wheel angle measurements.

$$\breve{\dot{q}}_{i_k} = \frac{\breve{q}_{i_k} - \breve{q}_{i_{k-1}}}{T_e} \quad | \quad i = \{2,3,4\} \tag{63}$$

The wheel angular velocities and resistive torques are made observable through this measurement [6].

A measurement of the rover's pitch angle $\breve{\theta}_{0_y}$ is created using the direction of the gravity vector as sensed by the IMU. Under the slow acceleration or steady-state conditions on the rover this method is valid.

$$\breve{\theta}_{0_{y_k}} = \tan^{-1}\left(\frac{-\breve{\ddot{x}}^R_{0_k}}{\breve{\ddot{z}}^R_{0_k}}\right) \tag{64}$$

This measurement is used to make $\dot{\theta}_{0_y}$ and $\ddot{\theta}_{0_y}$ observable. Using a pitch rate measurement $\dot{\theta}_{0_y}$, which is obtainable from the IMU, was found to decrease the performance of the estimator; this is due to the low signal to noise ratio of this sensor for a slowly moving rover. As a result, a pitch rate measurement was not included in the measurement vector $\boldsymbol{y_1}$.

The measurements $\breve{q}_1$, $\breve{\ddot{x}}^R_0$, $\breve{\ddot{z}}^R_0$, and $\breve{\ddot{x}}^R_0$ are taken directly from the sensor measurement vector $\boldsymbol{z}$.

To create a measurement of slip, a measured configuration vector $\breve{\boldsymbol{\Phi}}$ and its derivative $\breve{\dot{\boldsymbol{\Phi}}}$ are required.

$$\breve{\boldsymbol{\Phi}}_k = \begin{bmatrix} 0 & 0 & 0 & 0 & \breve{\theta}_{0_{y_k}} & 0 & \breve{q}_{1_k} & 0 & 0 & 0 \end{bmatrix}^T \tag{65}$$

$$\breve{\dot{\boldsymbol{\Phi}}}_k = \begin{bmatrix} \breve{\dot{x}}^R_{0_k}\cos\breve{\theta}_{0_y} & 0 & -\breve{\dot{x}}^R_{0_k}\sin\breve{\theta}_{0_y} & \dots \\ \dots & 0 & \breve{\dot{\theta}}_{0_{y_k}} & 0 & \breve{\dot{q}}_{1_k} & \breve{\dot{q}}_{2_k} & \breve{\dot{q}}_{3_k} & \breve{\dot{q}}_{4_k} \end{bmatrix}^T \tag{66}$$

The measured link Jacobian $\breve{\boldsymbol{J}}_{\boldsymbol{a}k}$ and measured set of joint velocities $\breve{\dot{\boldsymbol{X}}}_{\boldsymbol{a_k}}$ can then be calculated as in Equation 21.

$$\breve{\dot{\boldsymbol{X}}}_{\boldsymbol{a}k} = \breve{\boldsymbol{J}}_{\boldsymbol{a}k}\breve{\dot{\boldsymbol{\Phi}}}_k \tag{67}$$

The measurements of slip can then be calculated using the wheel linear and angular velocity magnitudes found above and Equation 1 or 2. It is assumed that the direction of rover motion is positive with positive X_w.

$$\breve{i}_{i_k} = i\left(v_w, \omega\right) = i\left(\text{sgn}\left(\breve{\dot{\boldsymbol{X}}}_{\boldsymbol{a}(6i+1)_k}\right)\left|\breve{\boldsymbol{v}}_{\boldsymbol{i+1}_k}\right|, \breve{\dot{q}}_{i+1_k}\right) \tag{68}$$

for $i = \{1,2,3\}$ and where $\breve{\boldsymbol{v}}_{\boldsymbol{i+1}_k}$ is the wheel centroid velocity vector of wheel i, a term in $\breve{\dot{\boldsymbol{X}}}_{\boldsymbol{a}}$ (see Equation 21).

3) Measurement Model: The measurement model h_1 predicts the expected set of measurements based on the current estimated state vector $\hat{\boldsymbol{x}}_{\boldsymbol{1}k}$. The estimated measurement of a variable x is denoted $\hat{\breve{x}}$ in this paper.

$$\hat{\boldsymbol{y}}_{\boldsymbol{1}k} = h_1\left(\hat{\boldsymbol{x}}_{\boldsymbol{1}k}\right) \tag{69}$$

The wheel angular velocities $\{\dot{q}_2, \dot{q}_3, \dot{q}_4\}$, rover pitch θ_{0_y}, and bogie joint angle q_1 are states and thus have a simple measurement model.

$$\hat{\breve{\dot{q}}}_{2_k} = \hat{\dot{q}}_{2_k} \quad \hat{\breve{\dot{q}}}_{3_k} = \hat{\dot{q}}_{3_k} \quad \hat{\breve{\dot{q}}}_{4_k} = \hat{\dot{q}}_{4_k} \tag{70}$$

$$\hat{\breve{\theta}}_{0_{y_k}} = \hat{\theta}_{0_{y_k}} \quad \hat{\breve{q}}_{1_k} = \hat{q}_{1_k} \tag{71}$$

The estimated accelerations in world coordinates $\{\hat{\ddot{x}}_0, \hat{\ddot{z}}_0\}$ must be rotated into rover coordinates to match the measurements $\{\breve{\ddot{x}}^R_0, \breve{\ddot{z}}^R_0\}$. The effect of gravity, which is measured by the accelerometers, must also be included. Note that gravity, although directed downwards, produces the same signal as an upward acceleration of g.

$$\hat{\breve{\ddot{x}}}^R_{0_k} = \hat{\ddot{x}}_{0_k}\cos\hat{\theta}_{0_{y_k}} - \hat{\ddot{z}}_{0_k}\sin\hat{\theta}_{0_{y_k}} - g\sin\hat{\theta}_{0_{y_k}} \tag{72}$$

$$\hat{\breve{\ddot{z}}}^R_{0_k} = \hat{\ddot{x}}_{0_k}\sin\hat{\theta}_{0_{y_k}} + \hat{\ddot{z}}_{0_k}\cos\hat{\theta}_{0_{y_k}} + g\cos\hat{\theta}_{0_{y_k}} \tag{73}$$

The slips $\{i_1, i_2, i_3\}$ are states and thus have a simple measurement model.

$$\hat{\breve{i}}_{1_k} = \hat{i}_{1_k} \quad \hat{\breve{i}}_{2_k} = \hat{i}_{2_k} \quad \hat{\breve{i}}_{3_k} = \hat{i}_{3_k} \tag{74}$$

C. Estimator 2: Wheel Force Estimator with Two-Axis Force Sensors

The wheel force estimators both estimate nine states: the wheel-terrain contact angles, the wheel normal loads, and the wheel drawbar pulls.

$$\boldsymbol{x_2} = \begin{bmatrix} \gamma_1 & \gamma_2 & \gamma_3 & W_1 & W_2 & W_3 & DP_1 & DP_2 & DP_3 \end{bmatrix}^T \tag{75}$$

The two-axis force sensor senses two orthogonal components of force and thus given the wheel-terrain contact angle γ_i, can resolve the sensed forces into normal load W_i and drawbar pull DP_i for wheel i.

1) Process Model: The process model f_2 is used to propagate the states forward.

$$\hat{\boldsymbol{x}}_{\boldsymbol{2}k+1} = f_2\left(\hat{\boldsymbol{x}}_{\boldsymbol{2}k}\right) \tag{76}$$

Terrain angle evolution is an inherently random process [1] and thus is modelled as a random walk. Since the estimator has no *a priori* knowledge of the terrain response, both normal load W and drawbar pull DP must be treated as unmodelled phenomena; their process models are also assumed to be a random walk. Thus the expected next state vector is simply the same as the previous one.

$$\hat{\boldsymbol{x}}_{\boldsymbol{2}k+1} = \hat{\boldsymbol{x}}_{\boldsymbol{2}k} \tag{77}$$

2) Measurements: The measurement vector contains wheel-terrain contact angle measurements, and force sensor measurements.

$$\boldsymbol{y_2} = \begin{bmatrix} \breve{\gamma}_1 & \breve{\gamma}_2 & \breve{\gamma}_3 & \breve{F}_{L_1} & \breve{F}_{L_2} & \breve{F}_{L_3} & \breve{F}_{\perp_1} & \breve{F}_{\perp_2} & \breve{F}_{\perp_3} \end{bmatrix}^T \tag{78}$$

The estimated configuration vector $\hat{\boldsymbol{\Phi}}$ and estimated generalized velocity $\hat{\dot{\boldsymbol{\Phi}}}$ calculated in Estimator 1 are used in the measurement of wheel-terrain contact angles. Since the velocity of the wheel centroid will be parallel to the terrain, the angle of the velocity vector with respect to the horizontal will also represent the wheel-terrain contact angle. The procedure for creating the measurements of wheel-terrain contact angle is as follows:

$$\hat{\dot{\boldsymbol{X}}}_{\boldsymbol{a}k} = \hat{\boldsymbol{J}}_{\boldsymbol{a}k}\hat{\dot{\boldsymbol{\Phi}}}_k$$

$$\breve{\gamma}_{i_k} = \tan^{-1}\left(\frac{-\hat{v}_{i+1_{z_k}}}{\hat{v}_{i+1_{x_k}}}\right) = \tan^{-1}\left(\frac{-\hat{\dot{\boldsymbol{X}}}_{\boldsymbol{a}(6i+3)_k}}{\hat{\dot{\boldsymbol{X}}}_{\boldsymbol{a}(6i+1)_k}}\right) \tag{79}$$

where $\hat{\boldsymbol{J}}_{\boldsymbol{a}k}$ is a function of $\hat{\boldsymbol{\Phi}}_k$, and $\hat{v}_{i+1_{x_k}}$ and $\hat{v}_{i+1_{z_k}}$ are respectively the X_w and Z_w components of wheel i's centroid velocity.

The wheel-terrain contact angle estimation presented above is similar to a method used by Iagnemma et al. [1] except that owing to the addition of a velocimeter, does not rely on the assumption of low slip.

The measurements $\{\breve{F}_{L_1}, \breve{F}_{L_2}, \breve{F}_{L_3}, \breve{F}_{\perp_1}, \breve{F}_{\perp_2}, \breve{F}_{\perp_3}\}$ are taken directly from the two-axis force sensors.

3) Measurement Model: The measurement model h_2 predicts the expected set of measurements based on the current estimated state vector $\hat{\boldsymbol{x}}_{\boldsymbol{2}k}$. The wheel-terrain contact angles $\{\gamma_1, \gamma_2, \gamma_3\}$ are states and thus have a simple measurement model.

$$\hat{\breve{\gamma}}_{i_k} = \hat{\gamma}_{i_k} \quad | \quad i = \{1, 2, 3\} \tag{80}$$

When the rover is traversing a planar surface, the sensing axes of the force sensors are perfectly aligned with the normal loads and drawbar pulls. However, in the general case this is not true and the estimated normal loads and drawbar pulls must be rotated in order to predict the force sensor measurements.

$$\hat{\breve{F}}_{L_{i_k}} = \hat{W}_{i_k} \cos\left(\hat{\gamma}_{i_k} - \hat{\varepsilon}_{i_k}\right) - \hat{DP}_{i_k} \sin\left(\hat{\gamma}_{i_k} - \hat{\varepsilon}_{i_k}\right) \tag{81}$$

$$\hat{\breve{F}}_{\perp_{i_k}} = \hat{W}_{i_k} \sin\left(\hat{\gamma}_{i_k} - \hat{\varepsilon}_{i_k}\right) + \hat{DP}_{i_k} \cos\left(\hat{\gamma}_{i_k} - \hat{\varepsilon}_{i_k}\right) \tag{82}$$

for $i = \{1, 2, 3\}$ and where $\hat{\varepsilon}_{1_k} = \hat{\varepsilon}_{2_k} = \hat{\theta}_{0_{y_k}} + \hat{q}_{1_k}$ and $\hat{\varepsilon}_{3_k} = \hat{\theta}_{0_{y_k}}$.

D. Estimator 3: Wheel Force Estimator with Single-Axis Force Sensors

The state vector for Estimators 2 and 3 are identical.

$$\boldsymbol{x_3} = \boldsymbol{x_2} \tag{83}$$

Single-axis force sensors may be used on a rover to provide force sensing above the wheel hubs with a minimal impact on mass and complexity. The *Kapvik* micro-rover is equipped with single-axis force sensors above the wheel hubs.

1) Process Model: The process model is identical to the previous estimator: all states are assumed to undergo a random walk.

$$\hat{\boldsymbol{x}}_{\boldsymbol{3}k+1} = \hat{\boldsymbol{x}}_{\boldsymbol{3}k} \tag{84}$$

2) Measurements: The measurement vector contains a set of measurements identical to the previous estimator; however, the perpendicular forces $\{F_{\perp_1}, F_{\perp_2}, F_{\perp_3}\}$ are calculated artificially since they are not explicitly sensed.

$$\boldsymbol{y_3} = \boldsymbol{y_2} \tag{85}$$

The procedure for calculating measurements of wheel-terrain contact angles $\{\breve{\gamma}_1, \breve{\gamma}_2, \breve{\gamma}_3\}$ is identical to the previous estimator (see Equation 79). The measurements $\{\breve{F}_{L_1}, \breve{F}_{L_2}, \breve{F}_{L_3}\}$ are taken directly from the single-axis force sensors. As a result, one component of each wheel force is known. The remaining components $\{\breve{F}_{\perp_1}, \breve{F}_{\perp_2}, \breve{F}_{\perp_3}\}$ must be solved for using the estimated net effects of wheel forces $\{\hat{M}_{0f_w}, \hat{M}_{1f_w}, \hat{F}_{xf_w}, \hat{F}_{zf_w}\}$. As shown in Figure 9, forces $F_{\perp_1}$ and $F_{\perp_2}$ have the same line of action. Given the net effects of wheel forces observed it is impossible to differentiate the two forces and they are combined together into $F_{\perp_{12}} = F_{\perp_1} + F_{\perp_2}$. With two unknowns, $F_{\perp_{12}}$ and $F_{\perp_3}$, only two of the four equations are required to find a solution. The equation for M_{1f_w} is used to solve for $F_{\perp_{12}}$. With knowledge of $F_{\perp_{12}}$, the equation for F_{xf_w} is used to solve for $F_{\perp_3}$. Referring to Figure 9, the forces $F_{\perp_{12}}$ and $F_{\perp_3}$ are solved for as follows:

$$\breve{F}_{\perp_{12_k}} = \left(-\hat{M}_{1f_{w_k}} + \left(\breve{F}_{L_{1_k}} - \breve{F}_{L_{2_k}}\right) b\right) \frac{1}{a} \tag{86}$$

$$\breve{F}_{\perp_{3_k}} = \Big(\hat{F}_{xf_{w_k}} - \left(\breve{F}_{L_{1_k}} + \breve{F}_{L_{2_k}}\right) \sin\hat{\varepsilon}_{1_k} \ \ldots$$
$$\ldots \ - \breve{F}_{\perp_{12_k}} \cos\hat{\varepsilon}_{1_k} - \breve{F}_{L_{3_k}} \sin\hat{\varepsilon}_{3_k}\Big) \frac{1}{\cos\hat{\varepsilon}_{3_k}} \tag{87}$$

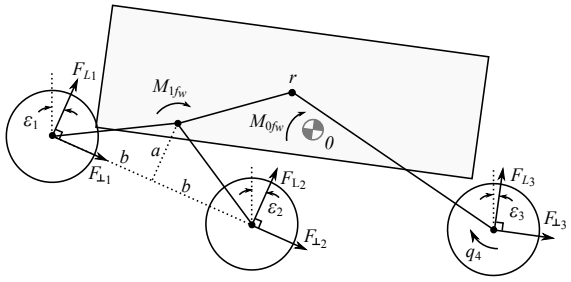


Fig. 9: Forces applied at the wheel shown in sensor directions.

Thus, it is possible to provide an accurate measurement of $F_{\perp_3}$. The combined force $\breve{F}_{\perp_{12}}$ can then be apportioned based on the sensed loads $\breve{F}_{L_1}$ and $\breve{F}_{L_2}$, which up to moderate values of $\{\varepsilon_1, \varepsilon_2\}$ approximately represent the normal loads W_1 and W_2.

$$\breve{F}_{\perp_{1_k}} = \frac{\breve{F}_{\perp_{12_k}} \breve{F}_{L_{1_k}}}{\breve{F}_{L_{1_k}} + \breve{F}_{L_{2_k}}} \qquad \breve{F}_{\perp_{2_k}} = \frac{\breve{F}_{\perp_{12_k}} \breve{F}_{L_{2_k}}}{\breve{F}_{L_{1_k}} + \breve{F}_{L_{2_k}}} \tag{88}$$

This is an approximation, since the relationship between drawbar pull and normal load is not a proportional one. Additionally, F_{L_1} and F_{L_2} do not directly represent the normal loads, and $F_{\perp_1}$ and $F_{\perp_2}$ do not directly represent the drawbar pulls, as can be seen in Equations 81 and 82. Nevertheless, apportioning the forces in this way provides a reasonable approximation. A similar procedure is performed in [6].

3) Measurement Model: The measurement model for the single-axis wheel force estimator is identical to that for the two-axis wheel force estimator.

$$\hat{\boldsymbol{y}}_{\boldsymbol{3}k} = h_3(\hat{\boldsymbol{x}}_{\boldsymbol{3}k}) = h_2(\hat{\boldsymbol{x}}_{\boldsymbol{3}k}) \tag{89}$$

E. Creating the Model

The result of running the estimators developed above is a set of estimated terrain response values $\{\tau_R, DP, W, i\}$ for each wheel. These are all treated as sample points in the two terrain response relationships: resistive torque τ_R as a function of normal load W and slip i, and drawbar pull DP as a function of normal load W and slip i.

Creating a two dimensional polynomial fit with thousands of data points can be computationally expensive. To make it more manageable, sample data can be temporally averaged, decreasing the number of sample points.

A two dimensional polynomial function is fit to the sample points. The complexity of the relationship modelled dictates the required order O of the fit to be used. Over the normal loads of $0.1\,\mathrm{N} \leq W \leq 200\,\mathrm{N}$ used for the simulation model, a 4^{th} order polynomial was found to be most appropriate. Over the normal loads in which the rover will typically operate ($35\,\mathrm{N} \leq W \leq 70\,\mathrm{N}$ in *Kapvik* simulations below), the relationship is less complex and 2^{nd} order relationships were found to be most appropriate. Fits of higher order result in highly erroneous data at the edge of and outside the parameter space spanned by the sample data.

The drawbar pulls and to a lesser extent the normal loads estimated using single-axis force sensors (Estimator 3) will be less accurate on wheels 1 and 2 than on wheel 3. If desired, wheel 3 can be used exclusively to create the model, serving as a sensing wheel. Drawbar pull estimates using wheels 1 and 2 will be biased toward a proportional relationship between drawbar pull and normal load because of the force apportioning performed in Equation 88.

F. Proposed Usage of Estimator

The terrain response estimation algorithm requires that the terrain be homogeneous and gently rolling. Forward looking range data could be used to assess the suitability of the terrain ahead and autonomously make a decision about whether or not to perform terrain response estimation. During a traverse, the rover could log all of the essential sensor measurements for terrain response estimation. When stopped, the estimator could be run to obtain $DP(W,i)$ and $\tau_R(W,i)$ data points from the recently traversed terrain. Temporally averaged data could then be added to the existing dataset. A maximum number of recent points to be kept in this dataset could be chosen in advance based on memory and processing time available. A polynomial fit to the dataset could then be performed, forming a local model for terrain response that could subsequently be used in trafficability predictions, cost function based path planning, or traction control. For example, the terrain response acquired during a traverse over relatively benign terrain could be used to assess the feasibility of climbing an upcoming incline. If desired, a history of polynomial coefficients could be kept in memory to capture the evolution of terrain response over the duration of the mission.

Ideally, the estimator would also be able to run in real-time, providing current estimates of resistive torques, drawbar pulls, wheel normal loads, wheel slips, and wheel-terrain contact angles, amongst other states. Knowledge of these states could be used to perform reactive traction control, with forces being optimally applied, or high slip conditions being avoided. Real-time estimator speeds have not been achieved in the MATLAB simulation of the estimator; optimization of the algorithm for real-time operation lies outside the scope of this paper.

V. Simulation Results and Discussion

Simulations of the rover traversing the six representative terrains shown in Figure 10 were performed with a desired speed v_d of 2 cm/s, the nominal speed for *Kapvik*. Unbiased Gaussian noise of the magnitudes outlined in Table IV was added to the sensor measurements. Since the rover moves slowly, the estimator was run at 5 Hz using temporally averaged measurements; this reduced computation time.

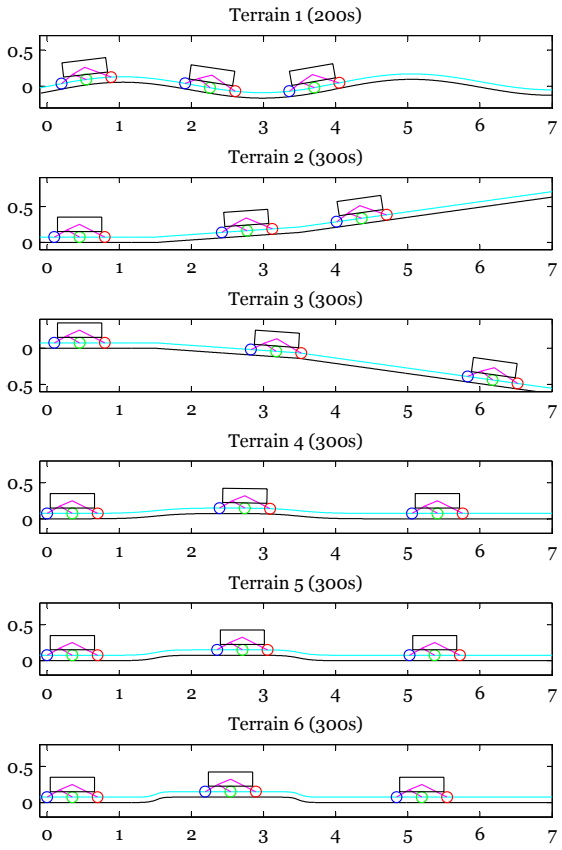


Fig. 10: The six terrains used to validate the estimator, shown with the simulation duration in parentheses. Terrain 1 is a gently rolling terrain. Terrains 2 and 3 are predominantly flat uphill and downhill terrains respectively, each with slopes of $0°$, $\pm 4°$ and $\pm 8°$ and short transitions. Terrains 4, 5, and 6 are step increases of half a *Kapvik* wheel diameter formed by sigmoid functions, where the transition region decreases in distance from Terrain 4-5 and Terrain 5-6. The rover is shown at the beginning, middle, and end of its traverse.

A. Estimator Performance

The root mean square estimation errors of the important terrain response estimation states over the simulation runs are shown in Table V. Slip $\hat{i}$ and resistive torque $\hat{\tau}_R$ are consistently estimated well by Estimator 1. As expected, the performance of the wheel force estimator using single-axis force sensors (Estimator 2) is poorer than that using two-axis force sensors (Estimator 3) for wheels 1 and 2. As the drawbar pull and normal load do not follow a proportional relationship as implied in Equation 88, a systematic error is introduced for wheels 1 and 2. However, the drawbar pull for wheel 3 is estimated with an accuracy nearly identical to that when using two-axis force sensors. Errors in estimated forces $\{\hat{W}, \hat{DP}\}$ increase in magnitude for Terrains 4-6 as the transition region shortens. This is a result of the assumption that $\dot{z}_0^R \approx 0$ and is explained in Section V-B.

RMS Errors	$\hat{i}_1$ [-]	$\hat{i}_2$ [-]	$\hat{i}_3$ [-]	$\hat{W}_1$ [N]	$\hat{W}_2$ [N]	$\hat{W}_3$ [N]	$\hat{\tau}_{R_1}$ [Nm]	$\hat{\tau}_{R_2}$ [Nm]	$\hat{\tau}_{R_3}$ [Nm]	$\hat{DP}_1$ [N]	$\hat{DP}_2$ [N]	$\hat{DP}_3$ [N]
Terrain 1, Estimators 1 and 2	0.015	0.015	0.016	0.15	0.095	0.3	0.008	0.0051	0.0055	0.88	0.55	0.47
Terrain 1, Estimators 1 and 3	0.015	0.015	0.016	0.21	0.17	0.31	0.008	0.0051	0.0055	1.8	2.3	0.49
Terrain 2, Estimators 1 and 2	0.012	0.012	0.012	0.094	0.048	0.076	0.0038	0.0028	0.003	0.53	0.23	0.35
Terrain 2, Estimators 1 and 3	0.012	0.012	0.012	0.1	0.06	0.078	0.0038	0.0028	0.003	0.65	0.67	0.4
Terrain 3, Estimators 1 and 2	0.011	0.011	0.011	0.059	0.056	0.069	0.0024	0.0023	0.0024	0.31	0.18	0.27
Terrain 3, Estimators 1 and 3	0.011	0.011	0.011	0.065	0.065	0.071	0.0024	0.0023	0.0024	0.41	0.42	0.31
Terrain 4, Estimators 1 and 2	0.012	0.012	0.012	0.06	0.048	0.092	0.0032	0.0025	0.0027	0.32	0.23	0.32
Terrain 4, Estimators 1 and 3	0.012	0.012	0.012	0.067	0.061	0.096	0.0032	0.0025	0.0027	0.56	0.79	0.38
Terrain 5, Estimators 1 and 2	0.013	0.013	0.011	0.086	0.067	0.21	0.0039	0.0029	0.0031	0.73	0.51	0.71
Terrain 5, Estimators 1 and 3	0.013	0.013	0.011	0.11	0.14	0.21	0.0039	0.0029	0.0031	0.88	1.5	0.75
Terrain 6, Estimators 1 and 2	0.02	0.02	0.017	0.15	0.12	0.37	0.0068	0.0047	0.005	1.7	1	1.3
Terrain 6, Estimators 1 and 3	0.02	0.02	0.017	0.23	0.34	0.38	0.0068	0.0047	0.005	1.7	2.6	1.3

TABLE V: Root mean square errors in the simulated state estimates $\{\hat{i}, \hat{W}, \hat{\tau_R}, \hat{DP}\}$.

Figure 11 shows the estimated and real slips and normal loads over the 200 s Terrain 1 simulation. Slip is accurately estimated, with some white noise. Normal load is estimated with similar accuracy using both Estimators 2 and 3; as a result, only the results from Estimator 2 are shown.

Figure 12 shows the estimated and real resistive torques τ_R and drawbar pulls DP for the same simulation. Resistive torques are very well estimated because of the precise measurement of wheel angles $\{q_2, q_3, q_4\}$ (Figure 12a). Drawbar pulls are well estimated by Estimator 2 using two-axis force sensors (Figure 12b). A small systematic error is caused by the imperfect estimation of wheel-terrain contact angles, as outlined in Section V-B. The white noise in drawbar pull is more prevalent for wheels 1 and 3 than for wheel 2. This is because the estimate of wheel-terrain contact angle $\hat{\gamma}_2$ is less sensitive to noise in the estimated rover pitch $\hat{\theta}_{0_y}$ than are $\hat{\gamma}_1$ and $\hat{\gamma}_3$. Since the estimated rover pitch $\hat{\theta}_{0_y}$ is noisier than the other variables affecting the estimate of wheel-terrain contact angle, wheels 1 and 3 experience more white noise than wheel 2. The systematic error caused by force apportioning performed in Equation 88 is evident in the drawbar pull estimates for wheels 1 and 2 (Figure 12c).

B. Effect of Vertical Velocity Assumption

The assumption that the velocity of *0* along the body vertical $\dot{z}_0^R$ is approximately zero reduces the performance of the estimator in situations where $\dot{z}_0^R \not\approx 0$. This is because the wheel-terrain contact angles $\{\gamma_1, \gamma_2, \gamma_3\}$ are calculated from the estimated angles of the wheel centroid velocity vectors with respect to the horizontal. The resultant error in calculated wheel terrain contact angles is approximately the angle of the rover velocity with respect to its forward direction $\delta \approx \tan^{-1}\left(-\dot{z}_0^R/\dot{x}_0^R\right) \approx -\dot{z}_0^R/\dot{x}_0^R$. When used to apportion the sensed/calculated wheel forces $\{F_{L_1}, F_{L_2}, F_{L_3}, F_{\perp_1}, F_{\perp_2}, F_{\perp_3}\}$ into the desired forces $\{DP_1, DP_2, DP_3, W_1, W_2, W_3\}$, the error in calculated wheel-terrain contact angles $\{\gamma_1, \gamma_2, \gamma_3\}$ limits the achievable accuracy. The approximate magnitude of the error is found by solving Equations 81 and 82 for DP and W and comparing their values with and without the error term δ.

$$E_{DP_i} = [F_{L_i}\sin(\gamma_i - \delta - \varepsilon_i) - F_{\perp_i}\cos(\gamma_i - \delta - \varepsilon_i)] - [F_{L_i}\sin(\gamma_i - \varepsilon_i) - F_{\perp_i}\cos(\gamma_i - \varepsilon_i)] \quad (90)$$

$$E_{W_i} = [F_{L_i}\cos(\gamma_i - \delta - \varepsilon_i) + F_{\perp_i}\sin(\gamma_i - \delta - \varepsilon_i)] - [F_{L_i}\cos(\gamma_i - \varepsilon_i) + F_{\perp_i}\sin(\gamma_i - \varepsilon_i)] \quad (91)$$

where E_{DP_i} and E_{W_i} are the calculated errors in drawbar pull and normal load respectively for wheel i. A representative case is chosen where $\gamma_i = 0$, $W_i = 50$ N, and $DP_i = 0$, but $\delta \neq 0$; this represents steady state operation where the wheel being analyzed is on level ground but the velocity along the body vertical $\dot{z}_0^R$ is non-zero; this situation is possible if the rover is pitched. In Figure 13, as $(\gamma_i - \varepsilon_i)$ is altered, F_{L_i} and $F_{\perp_i}$ are altered to accurately reflect the redistribution of forces. The resulting relationship is shown in Figure 13 and shows that miscalculation of wheel terrain contact angle has a larger effect on drawbar pull DP_i than on normal load W_i.

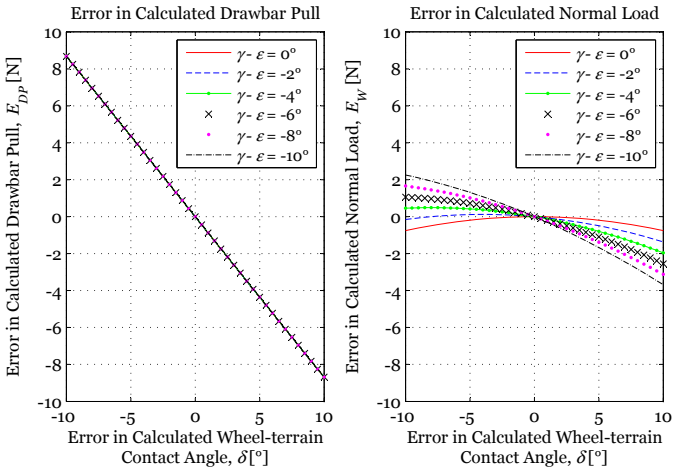
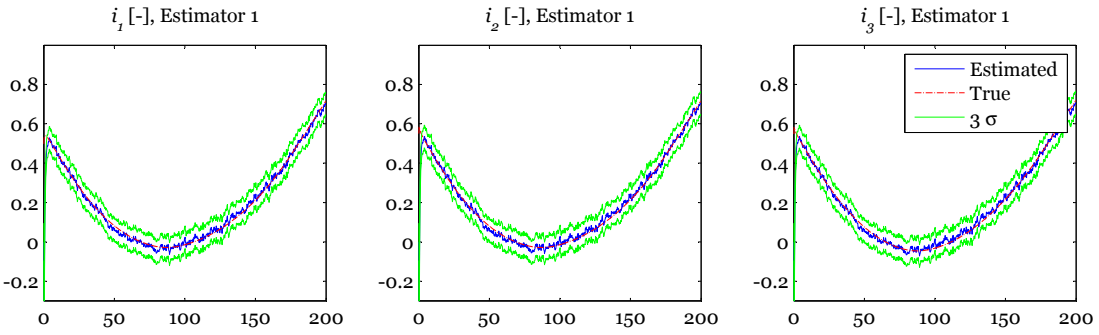
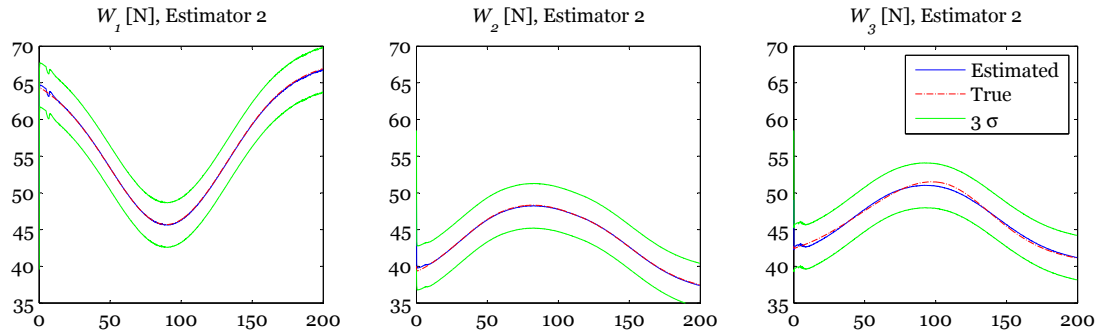


Fig. 13: The sensitivity of calculated drawbar pull (left) and normal load (right) to errors in calculated wheel-terrain contact angle δ.

A non-zero velocity $\dot{z}_0^R$ on a rover using a rocker-bogie mobility system is induced by a variety of terrain types which are not gently rolling. The approximate magnitude of $\dot{z}_0^R$ and resultant severity of the error in wheel-terrain contact angle,



(a) Estimated slip for wheels 1, 2, and 3 (left to right), Estimator 1.



(b) Estimated normal load for wheels 1, 2, and 3 (left to right), Estimator 2.

Fig. 11: Estimated normal loads W and slips i over the 200 s simulation (time in [s] on x axis). The estimation of normal load is of comparable accuracy between Estimators 2 and 3.

normal load, and drawbar pull estimates can be judged based on the curvature of the terrain κ compared with the wheelbase of the rover l_w. Terrains 4-6 have similar profiles but with increasing levels of curvature. Figure 14 demonstrates how the severity of the errors scale with the terrain's curvature for Terrains 4-6. As the terrain curvature κ approaches the reciprocal of the rover's wheelbase $(1/l_w)$, the likelihood of increased velocity of the rover along its body vertical $\dot{z}_0^R$ increases; the assumption that $\dot{z}_0^R \approx 0$ loses validity, and the estimator's performance degrades. In operation, forward looking range data could be used to assess the curvature of the upcoming terrain and decide whether or not to perform terrain response estimation during the traverse. Terrains that include curvatures $\kappa \geq 1/l_w$ should not be used for the terrain response estimation described herein.

C. Terrain Response Model Re-creation

To reduce the amount of data used to reconstruct the terrain response models, the estimates were temporally averaged to 1 sample every 2 seconds from the original sampling rate of 5 samples/s. Based on the requirement that $\kappa < 1/l_w$ from Section V-B and *Kapvik*'s wheelbase l_w from Table I, the curvature of viable terrain must be less than $1.43\,\mathrm{m/m^2}$; using this requirement, data points from the rover's traverses of Terrains 1-5 were included, but data points from the rover's traverse of Terrain 6 were excluded. A two dimensional, 2nd order polynomial function with was then fit to the data. The resulting models for resistive torque and drawbar pull using Estimators 1 and 2 are shown in Figures 15 and 16 respectively. The models using Estimators 1 and 3 are very similar and are not included herein. The coefficients for the estimated model are compared with an optimal two dimensional fit to the real model in Table VI. The distribution of errors in the resultant models for resistive torque and drawbar pull are shown in Figures 17 and 18 respectively. The estimated fits provide a good approximation of the actual terrain response in the traversed parameter space.

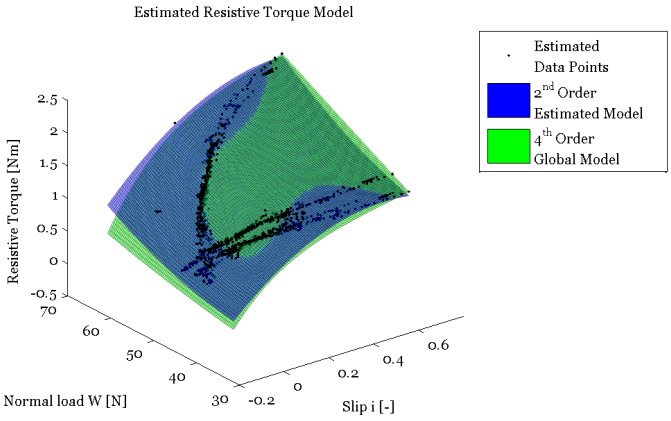
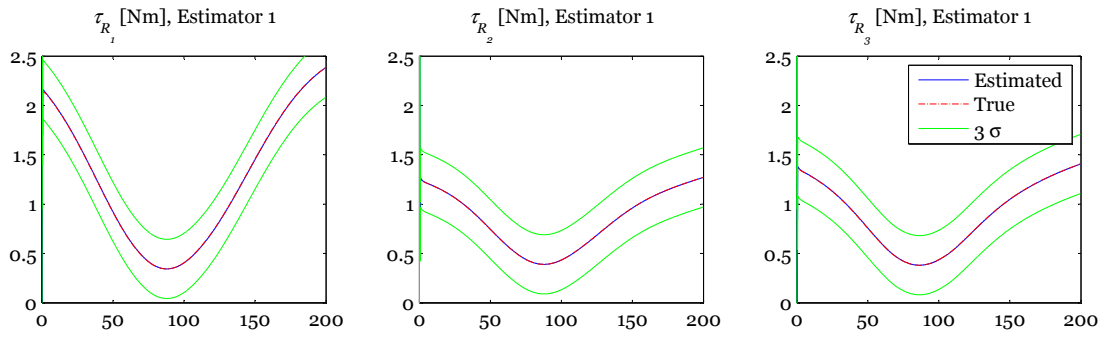
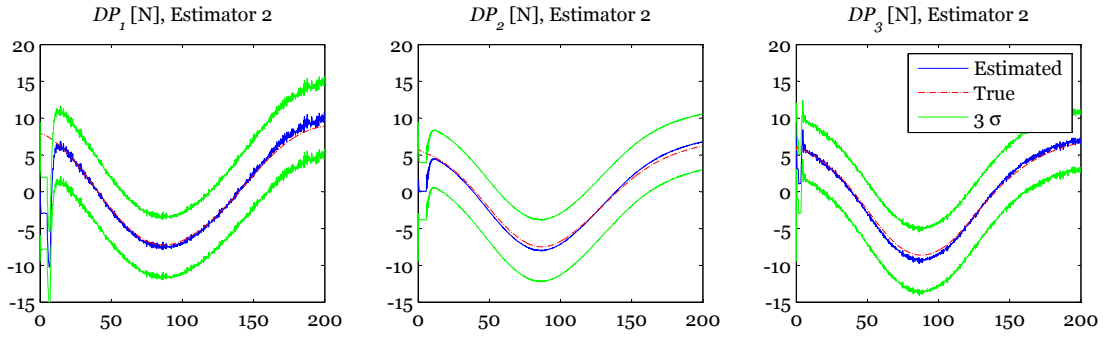


Fig. 15: Resistive torque fit using Estimators 1 and 2.

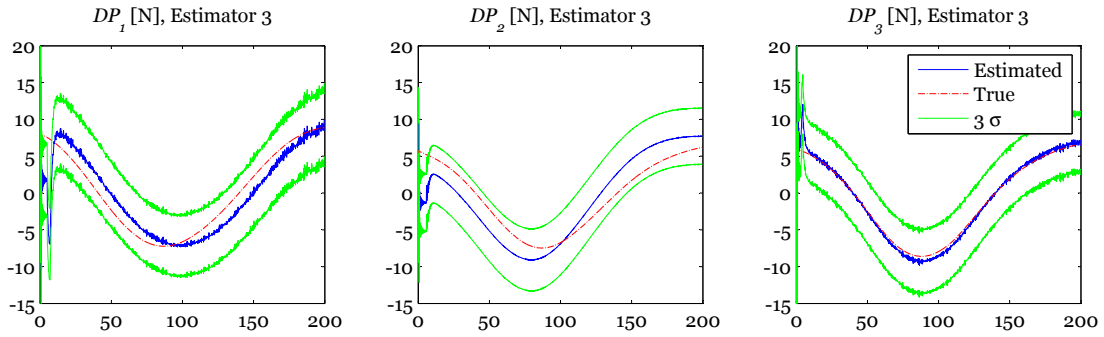
As shown in Figures 17 and 18, the errors in the estimated models are most severe near the edge of the traversed parameter space, and close to zero near to the estimated data points.



(a) Estimated resistive torque for wheels 1, 2, and 3 (left to right), Estimator 1.



(b) Estimated drawbar pull for wheels 1, 2, and 3 (left to right), Estimator 2.



(c) Estimated drawbar pull for wheels 1, 2, and 3 (left to right), Estimator 3.

Fig. 12: Estimated resistive torque τ_R and drawbar pull DP over the 200 s simulation (time in [s] on x axis) on Terrain 1.

	p_{00}	p_{10}	p_{01}	p_{20}	p_{11}	p_{02}
τ_R Local Model	-0.367	2.16	0.0139	-2.7	0.0266	8.2×10^{-5}
$\hat{\tau_R}$ Fit, Est. 1 and 2	0.777	2.35	-0.0327	-2.48	0.0156	5.79×10^{-4}
DP Local Model	-4.17	31.5	-0.032	-38.8	0.274	-1.66×10^{-4}
$\hat{DP}$ Fit, Est. 1 and 2	11.8	35.7	-0.685	-38.3	0.122	0.00673
$\hat{DP}$ Fit, Est. 1 and 3	9.33	35.5	-0.579	-37.8	0.127	0.00562

TABLE VI: Coefficients p_{kl} of the 2nd order polynomial fits to the optimal local model and estimated data. The local model was created by performing a 2nd order polynomial fit to a 100×100 array of global model sample points over the traversed parameter space.

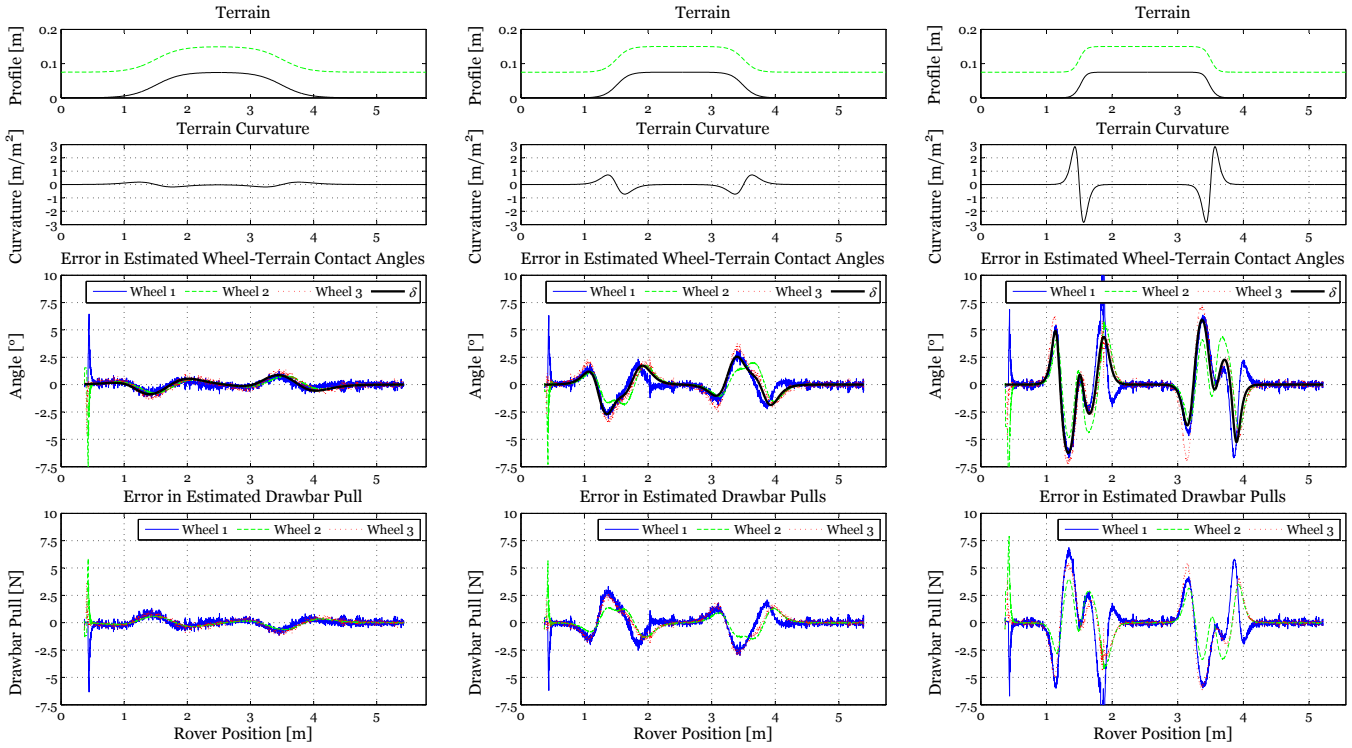
The area enclosed by estimated data points also has minimal error. This shows that predictions made using the estimated models would be most accurate for values of slip and normal load similar to or enclosed by those already sampled. The root mean square errors in the estimated model are shown in Table VII. The RMS errors as well as the error distributions of Figures 17 and 18 are calculated using 100×100 arrays of values over the traversed parameter space. The magnitudes of the RMS errors are relatively small, meaning that the estimated model could be effectively used for trafficability analysis, cost function based path planning, or traction control.

RMS Errors	τ_R[Nm]	DP[N]
Estimators 1 and 2	0.0551	0.747
Estimators 1 and 3	0.0554	0.670

TABLE VII: Root mean square error of the 2nd order estimated terrain response models compared with the 4th order global model over the traversed parameter space.

VI. Conclusion

Rovers exploring other planets will require knowledge of the terrain's response in order to assess terrain trafficability,



(a) Terrain 4. (b) Terrain 5. (c) Terrain 6.

Fig. 14: The effect of terrain curvature on the assumption that $\dot{z}_0^R \approx 0$ and resultant errors in estimated wheel-terrain contact angles and drawbar pulls. The resultant errors very closely follow the error term δ, which is a result of the assumption that $\dot{z}_0^R = 0$.

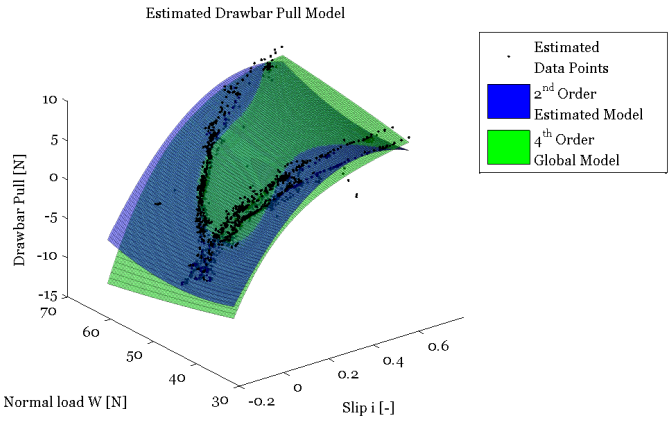


Fig. 16: Drawbar pull fit using Estimators 1 and 2.

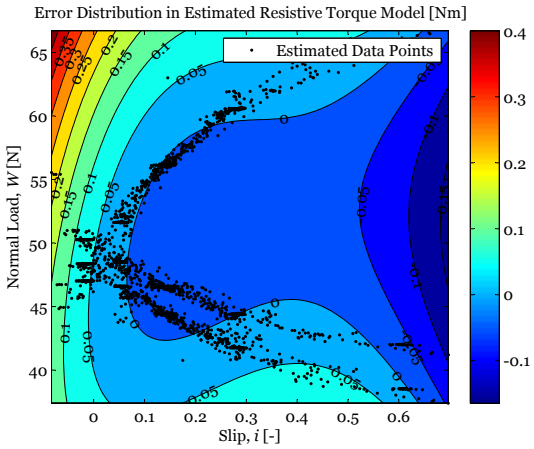


Fig. 17: Distribution of errors in 2nd order estimated resistive torque model over traversed parameter space compared with the 4th order global model.

perform cost function based path planning, or perform optimal traction control. Since bringing additional equipment to characterize the soil is expensive and impractical, soil properties or terrain response must be found using on-board rover sensors.

In this paper, resistive torque and drawbar pull, the key outputs of Wong's terramechanics equations, are simplified into two dimensional polynomial functions of normal load and slip. This expedites simulation by eliminating the requirement to use root-finding routines. Additionally, it provides a model that can be reconstructed if all the required variables are estimated.

Fully dynamic 2D simulation was performed on one side of *Kapvik*, a 30 kg planetary micro-rover prototype with a rocker-bogie mobility system. The rover was simulated traversing six different terrains; the simulations incorporated Wong's terramechanics relations to properly represent the wheel-soil interactions.

A suite of on-board sensors, including conventional sensors as well as a velocimeter and force sensors, were used to

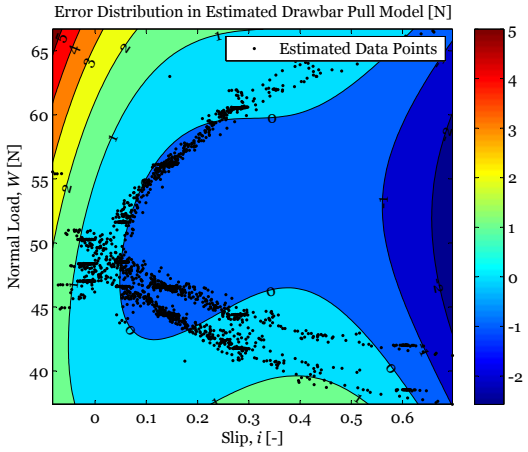


Fig. 18: Distribution of errors in 2nd order estimated drawbar pull model over traversed parameter space compared with the 4th order global model.

take measurements critical to the observation of the terrain response. These sensors, coupled with Unscented Kalman Filters, were used in simulation to accurately reconstruct the terrain response relationships over the traversed parameter space.

REFERENCES

[1] K. Iagnemma, "Traction Control of Wheeled Robotic Vehicles in Rough Terrain with Application to Planetary Rovers," *The International Journal of Robotics Research*, vol. 23, no. 10-11, pp. 1029–1040, Oct. 2004. [Online]. Available: http://ijr.sagepub.com/cgi/doi/10.1177/0278364904047392

[2] A. Krebs, F. Risch, T. Thueer, J. Maye, C. Pradalier, and R. Siegwart, "Rover control based on an optimal torque distribution - Application to 6 motorized wheels passive rover," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, Taipei, Taiwan, 2010, pp. 4372–4377.

[3] J. Wong and A. Reece, "Prediction of rigid wheel performance based on the analysis of soil-wheel stresses: Part I. Performance of Driven Rigid Wheels," *Journal of Terramechanics*, vol. 4, no. 1, pp. 81–98, 1967.

[4] K. Iagnemma, S. Kang, H. Shibly, and S. Dubowsky, "Online Terrain Parameter Estimation for Wheeled Mobile Robots With Application to Planetary Rovers," *IEEE Transactions on Robotics*, vol. 20, no. 5, pp. 921–927, 2004.

[5] L. Ding, K. Nagatani, K. Sato, A. Mora, K. Yoshida, H. Gao, and Z. Deng, "Terramechanics-based High-Fidelity Dynamics Simulation for Wheeled Mobile Robot on Deformable Rough Terrain," in *2010 IEEE International Conference on Robotics and Automation*, 2010, pp. 4922–4927.

[6] L. R. Ray, D. C. Brande, and J. H. Lever, "Estimation of net traction for differential-steered wheeled robots," *Journal of Terramechanics*, vol. 46, no. 3, pp. 75–87, 2009. [Online]. Available: http://dx.doi.org/10.1016/j.jterra.2008.03.003

[7] R. Lindemann and C. Voorhees, "Mars Exploration Rover Mobility Assembly Design, Test and Performance," *2005 IEEE International Conference on Systems, Man and Cybernetics*, pp. 450–455, 2005. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1571187

[8] D. Lhomme-Desages, C. Grand, and J. Guinot, "Trajectory control of a four-wheel skid-steering vehicle over soft terrain using a physical interaction model," in *2007 IEEE International Conference on Robotics and Automation*. IEEE, 2007, pp. 1164–1169. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=4209246

[9] G. Ishigami, "Terramechanics-based Analysis and Control for Lunar / Planetary Exploration Robots," Doctor of Philosophy Thesis, Tohoku University, 2008.

[10] L. Ding, H. Gao, Z. Deng, K. Yoshida, and K. Nagatani, "Slip Ratio for Lugged Wheel of Planetary Rover in Deformable Soil: Definition and Estimation," in *The 2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2009, pp. 3343–3348.

[11] L. Ding, H. Gao, Z. Deng, K. Nagatani, and K. Yoshida, "Experimental study and analysis on driving wheels performance for planetary exploration rovers moving in deformable soil," *Journal of Terramechanics*, vol. 48, no. 1, pp. 27–45, Feb. 2011. [Online]. Available: http://linkinghub.elsevier.com/retrieve/pii/S0022489810000601

[12] H. B. Pacejka and E. Bakker, "The Magic Formula Tyre Model," *Vehicle System Dynamics*, vol. 21, no. S1, pp. 1–18, 1992. [Online]. Available: http://www.informaworld.com/openurl?genre=article&doi=10.1080/00423110408969999

[13] J. Garcia de Jalon and J. Bayo, *Kinematic and Dynamic Simulation of Multibody Systems: The Real-Time Challenge*. Springer-Verlag, 1994.

[14] D. Dimitrov, "Multibody simulation dynamics of a multibody system (Euler-Lagrange formulation)," in *Multibody Simulation Course Notes*. Orebro University, 2009.

[15] D. Simon, *Optimal State Estimation*. John Wiley & Sons, 2006.

[16] K. Iagnemma, "Rough-Terrain Mobile Robot Planning and Control with Application to Planetary Exploration," Doctor of Philosophy Thesis, Massachusetts Institute of Technology, 2001.

[17] K. Yoshida, H. Hamano, and T. Watanabe, "Slip-based Traction Control of a Planetary Rover," in *Experimental Robotics VIII*, 5th ed. Springer, 2003, pp. 644–653.