The purpose of this project was to develop methods and software to determine whether a given Small Unmanned Ground Vehicle (SUGV) can traverse a given terrain, when both the SUGV and the terrain are not known exactly. A simulation model of a real-world SUGV (iRobot PackBot) was developed in the ADAMS environment and used to simulate traversal of a variable-height step obstacle. For this project, a user subroutine was successfully integrated into the ADAMS model to predict deformable track-terrain interaction. A parameterizable UGV vehicle system model was implemented using the ADAMS command language. A key element of this model is a slip-sinkage model. This simulation model was validated using real-world data collected in a step validation fixture with sand and a variable-height curb.

In a related effort, this project developed methods for using UGV sensor data to estimate variables and parameters needed for traction force prediction. The methods were evaluated using data collected from experiments with a PackBot traversing various deformable and non-deformable surfaces.

Unmanned Ground Vehicle, simulation, ADAMS, PackBot, deformable terrain, parameter estimation, simulation validation
STTR A07-T026 Final Report

USACE ERDC, Vicksburg, MS
March 7, 2011

Griffin Technologies
University of Texas at Austin
Southwest Research Institute
**PREdicting Mobility using STATistics (PreMoStat)**

**Purpose:**
- To develop methods and software to determine whether a given Small Unmanned Ground Vehicle (SUGV) can traverse a given terrain, when both the SUGV and the terrain are not known exactly:
  - Develop efficient UGV simulations that incorporate statistical variability in vehicle-terrain interactions
  - Establish experimental methods to validate statistical models for vehicle-terrain interactions for typical obstacles/terrains
  - Evaluate simulation efficacy using an Army-relevant SUGV on testbed terrains and obstacles

**Results:**
- A novel, efficient statistical simulation framework for predicting off-road robot mobility
- Quantification of prediction accuracy on realistic terrains and obstacles
- Validated models for vehicle-terrain interactions

**Payoff:**
- Increased survivability, reliability, and mission effectiveness in all terrain conditions
- Insight into observed robot performance in tests and in the field
- Model for integration into UGV simulation environments

**Phase I**
- Develop Monte Carlo model of PackBot climbing step, varying step height and surface friction
- Validated predicted traverse with empirical test data

**Phase II**
- Extend methods to deformable terrain (e.g., sand)
Phase 1 Summary

Collected real-world data on non-deformable steps

Built and validated a baseline mobility model

Found two key parameters: friction and step height

Confirmed that the model predicts the ability of a PackBot to climb a non-deformable step obstacles
Validation Data

Built a validation fixture at SwRI
- Platform motion using Vicon motion capture system
- PackBot internal data measurements
- On-board SwRI power logger
- Reference video

Collected non-deformable surface test data
- 5 Non-deformable surfaces, multiple heights, standard speed

Collected deformable surface validation data using sand
- 2 types of cement curbs, 4” and 1” radius
- Range of curb heights, from success to failure
- 4 speeds
- 2 sand depths
- 4 yaw angles
Fixture Parameters

- Curb height $h$
- Curb material $\mu$
- Curb radius/shape $r$
- Yaw angle relative to curb $\theta_1$
- Surface material depth $d$
- Velocity of the robot $v$
- Surface material properties (non-deformable)
- Flipper angle $\theta_2$
The Fixture

- Motion Capture Camera
- Movable Curb
- PackBot
- Sand Smoothing Device
- Gantry for Swapping Curbs
- Alternate Curb
- Landing Platform
- Sand/Soil Bin

Sand/Soil tray is fixed, curb and landing platform move vertically
Fine Height Adjustment

Adjustment mechanism raises/lowers curb 1mm per turn of the crank
Sand Depth and Leveling

Leveling device is adjusted to set the depth of sand
Non-Deformable Surfaces

Testing the surfaces used in Phase 1 with 4” radius concrete curb
Some Failure Modes in Sand

- **No Climb**
- **Spun Out**
- **High-Center**
- **Dug In**
Video

(Please click to play)
Disposition of Data

We used the data for validating the simulation model.

A set of Matlab functions were developed to process and synchronize the data.

The data was provided to performers on the DARPA Maximizing Mobility and Manipulation (M3) program.

All of this data is available for further distribution.

- Would need to look into any iRobot proprietary issues for release outside the Government.

SwRI continues to refine their data collection processes and is available to collect further data.
Developing a Simulation Model
Overview

1. What Was Done
2. Why Was This Done
3. Results
4. Comparison of Simulation and Test Results
5. Animations of Simulation Results
6. Development of Deformable Terrain Subroutine
7. Discussion of ADAMS Solver Simulation Process
8. Discussion on Slip-Sinkage Model
9. ADAMS Model Demo
10. Concluding Remarks
What Was Done

- Developed a subroutine for integrating specific track-terrain interaction models into ADAMS for predicting mobility on deformable terrains.

- Utilized ADAMS command language to efficiently build a parameterizable system model (rigid-bodies, forces, constraints, etc.)

- Developed a means for integrating slip-sinkage into the track-terrain interaction model.

- These new methods were fully implemented in ADAMS and evaluated using data collected during experiments with an iRobot PackBot at SwRI.
Why Was This Done

- Need a parameterizable track-terrain model for small scale UGV (PackBot) that could be used for statistical mobility prediction studies.

- Having a reliable parameterizable system model can have saving benefits (money and time) as opposed to physical testing.

- The ability to model highly deformable terrain interaction does not exist in ADAMS, so a customized subroutine was required.

- The ability to evaluate small UGVs operating in extreme maneuvers requires we be able to account for slip-sinkage.
ADAMS Vehicle Model

- Chassis, Sprocket and Idler geometries imported from CAD system (SolidWorks).

- Track segment geometry was created in ADAMS environment allowing for parameterization.

- Solid model geometries can easily be changed by simply importing the modified geometry file.
Improved Dynamics Model

- Phase 1 PackBot psuedo-track model was a set of cascaded wheels
- Cleats engaging/disengaging with terrain introduced bouncing that reduced accuracy and increased simulation time
- Results show improved simulation performance even though new model has more bodies and constraints

**Phase 1 Psuedo-Track Model**

**Pseudo-track**
- Cleats were attached to a set of cascaded wheels
  (Note curvature of track near cleat)

**Segmented Track, 36 segments**
- Parameterized - cleat height, taper angle, etc.
- Spring/damper system used to constrain motion of track segments relative to one another

**Phase 2 Full-Track Model**
ADAMS 3-D contact model was implemented to compute both the normal and friction forces acting at the track segment-curb interface.

The normal force is computed using:

\[ F_{normal} = k_s \Delta x^n + b_s f(\Delta x)\Delta \dot{x} \]  
(Eq. 6)

which is essentially a non-linear spring/damper system.

The friction force is modeled as

\[ F_{friction} = \mu (v_{slip}) F_{normal} \]  
(Eq. 7)

where the relationship between the coefficient of friction and slip velocity is shown on the figure to the right.

\( \Delta x \) = Penetration depth
\( \Delta \dot{x} \) = Penetration rate
Track Model

- Discretized track belt (36x segments)
- 3-D contacts between segments, sprocket, idler and support
- Bushing elements constrain motion between joining segments
- In-plane constraint between segment and track support
- Parameterizable cleat geometry
Various deformable terrain models have been presented in the literature for predicting both the normal and shear loads acting at the track-terrain interface.

Two of the more popular methods were used for this study because of their ease of implementation and the fact that test methods exist to extract values of their empirical parameters. However, it is noted that PreMoStat model is not limited to these methods.

To start, a set of 3 orthogonal force elements are applied to each segment’s center-of-mass (cm), which move with the body frame. For ease of explanation, we will assume an instance in time when the x, y, z force components are along the longitudinal, lateral and vertical directions, respectively.

The next sections discuss how both the normal and shear loads are computed and distributed among Fx, Fy, and Fz.
Shear Displacement Model

To generate traction force, shear displacement of the terrain must occur.

The total shear displacement for a single track segment is defined by:

\[ j = \sqrt{j_x^2 + j_y^2} \]

where the components of the shear displacement are computed by integrating the slip velocity in the corresponding direction starting from initial contact with the terrain:

\[ j_i = \int_0^{t_c} v_i dt \quad \text{for} \quad i = x, y \]

\( j \)  = absolute shear displacement
\( j_x \)  = shear displacement along x-axis of SM
\( j_y \)  = shear displacement along y-axis of SM
\( v_x \)  = slip velocity expressed along x-axis of SM
\( v_y \)  = slip velocity expressed along y-axis of SM
\( t_c \)  = accumalitve time in contact with the terrain
\( SM \)  = marker fixed to segment
Shear Force Model

The shear force model used for this analysis is based on a shear stress-shear displacement relationship proposed in [3] and is given by:

$$\tau = (c + \sigma \tan \phi)\left(1 - e^{-K/j}\right) \quad \text{(Eq. 8)}$$

The total shear force acting on a given track segment is given by:

$$F_{\text{shear}} = \tau_{\text{base}} A_{\text{base}} + \tau_{\text{cleat}} A_{\text{cleat}} \quad \text{(Eq. 9)}$$

Therefore, the shear force components are given by:

$$F_x = -F_{\text{shear}} \cos \theta \quad \text{(Eq. 10)}$$
$$F_y = -F_{\text{shear}} \sin \theta$$

where,

$$\theta = \tan^{-1}\left(\frac{v_y}{v_x}\right)$$

$c$, $\phi$, $K$ = empirical parameters
$\sigma$ = normal pressure
$v_x$ = slip velocity along x-axis
$v_y$ = slip velocity along y-axis
$A_{\text{base}}$ = base normal contact area
$A_{\text{cleat}}$ = cleat normal contact area
Normal Force Model

Therefore, the normal force acting on an individual track segment is computed as:

\[ F_N = \left( K_{ss}^{n} \cdot k \cdot z_{base} \cdot A_{base} \right) + \left( K_{ss}^{n} \cdot k \cdot z_{cleat} \cdot A_{cleat} \right) + \left( b_{damp} \cdot \dot{z}_{base} \right) \quad \text{(Eq. 11)} \]

where we have introduced a damping term to minimize oscillations allowing for better numerical stability and performance.

Therefore, the vertical force component is given by:

\[ F_Z = -F_N \quad \text{(Eq. 12)} \]

- \( k \) = pressure sinkage parameter
- \( z_{base} \) = static sinkage of base
- \( z_{cleat} \) = static sinkage of cleat
- \( A_{base} \) = area of base
- \( A_{cleat} \) = area of cleat
- \( \dot{z}_{base} \) = sinkage rate of base
- \( b_{damp} \) = damping parameter
- \( TM \) = marker fixed to terrain
- \( SM \) = marker fixed to segment
Control System

A simple PID controller was implemented to control the kinematic speed of the vehicle, which is similar to how the physical PackBot control system operates.

\[
V_{Kin} = \frac{1}{2} R_W (\omega_R + \omega_L)
\]

\[
\omega_{R,L} = \text{right/left sprocket speed}
\]

\[
\tau_{MR,ML} = \text{right/left motor torques}
\]

\[
\tau_{SR,SL} = \text{right/left sprocket torques}
\]

\[
R_W = \text{sprocket hub radius}
\]
Results

- A user subroutine was successfully developed for predicting deformable track-terrain interaction and was implemented into an ADAMS UGV system model.

- A parameterizable UGV vehicle system model was successfully developed using the ADAMS command language allowing for a more automated building process.

- A slip-sinkage model was integrated into the deformable terrain model and shows promising results considering its simplicity in formulation.

- Because of the extensive detail of the track model, simulation of ~9 (sec) real time takes ~2hrs to complete.
Simulation Test Case

To validate the ADAMS PackBot model, simulation results were compared to test data collected at SwRI.

The test cases consist of the PackBot traversing over a curb obstacle set to various heights and radii with the base of the curb being a loose sandy terrain.

The speed of the PackBot was set to operate at different reference speeds, which was also accounted for in the simulation using a PID control scheme.

A motion capture system was used to measure the position/orientation and velocities of the PackBot.
Simulation Results

The following test case was simulated and compared:

- Curb Height = 163 (mm)
- Curb Radius = 4 (in.)
- Reference Speed = 300 (mm/sec)

Max Static sinkage ~ 11.6 (mm) ~32 %
Max Slip-sinkage   ~ 24.2 (mm)  ~68 %
Max Total sinkage ~ 35.8 (mm)

---

[Graphs showing data comparisons]
Animation With No Slip-Sinkage

This animation shows a simulation result of the ADAMS PackBot model traversing a curb of height 117 (mm) and radius of 1 (in.) at 300 (mm/sec).

This version of the deformable terrain subroutine does not account for slip-sinkage, but only static sinkage.
Animation With Slip-Sinkage

This animation shows a simulation result of the ADAMS PackBot model traversing a curb of height 117 (mm) and radius of 1 (in.) at 300 (mm/sec).

This version of the deformable terrain subroutine accounts for slip-sinkage which is evident when compared to the previous animation.
Developing Terrain Subroutine

ADAMS allows one to define customized force elements using user-defined subroutines coded in either Fortran or C++.

Because of the complex nature of deformable terrain interaction, including large terrain deformations, it was necessary for our work to develop a user-defined force subroutine.

To accomplish this task, we used a bottom-up process by breaking down the subroutine into multiple functions.

- Function → computing normal forces
- Function → compute shear forces
- Function → terrain query
- Function → error checking

Each of these functions were developed and tested on elementary models to simplify the debugging process.

PreMoStat_GFOSUB.f

```
--- PreMoStat subroutine used to model deformable terrain ---------------------

subroutine GFOSUB(id, time, par, opar, dflag, iflag, result)
  implicit none
  -------------------------- Type and dimension statements ---------------------
  integer: id
  double precision: tparm(1), par(1,:), tflag(1), iflag(1)
  logical: dflag(1), iflag(1)

date: PreMoStat_GFOSUB.f

termini: id, tparm(1), par(1,:), dflag(1), iflag(1)

--- PreMoStat GFOSUB 23-Mar-2012

21 KB Fortran Source

File Name: PreMoStat GFOSUB.f

--- PreMoStat GFOSUB 23-Mar-2012

21 KB Fortran Source

File Name: PreMoStat GFOSUB.f
```

Check_Params.f

Comp_NonLinear_Debri.f

Comp_Shear_Debri_Params.f

Find_Terrain_Element.f

PreMoStat_GFOSUB.f
ADAMS Solver Simulation Process

ADAMS Solver (numerical engine), passes the kinematic states of the $i^{th}$ track segment to the PreMoStat.dll subroutine. In turn, the subroutine computes the normal and shear forces and passes this result back to ADAMS Solver. This process is repeated for each track segment at a given iteration time step.

Parameters

- $x_i, y_i, z_i$ = position of track segment
- $v_x_i, v_y_i, v_z_i$ = velocity of track segment
- $j_x_i, j_y_i$ = shear displacement of track segment
- $F_x_i, F_y_i, F_z_i$ = forces applied to track segment
**Slip-Sinkage**

Shearing of the terrain leads to a phenomenon referred to as slip-sinkage.

As shearing increases, additional sinkage is introduced resulting in bulldozing.

The total sinkage of the vehicle is the sum of the static sinkage and the additional sinkage due to shearing:

\[ z_{\text{total}} = z_o + z_j \quad \text{(Eq. 1)} \]

The figure to the right shows how the total sinkage varies with both shear displacement and normal pressure. The difficulty of implementing slip-sinkage is separating the contributions of the two components.

Experience has shown that for small UGVs (like PackBot), the component of total sinkage due to slip is dominant.

Testing has also shown that slip-sinkage can significantly effect the mobility of small UGVs under extreme maneuvers such as step climbing and zero radius turns.
In [1], Lyasko used a conservation of energy approach and proposed that the total sinkage of a tracked vehicle can be captured using:

\[ z = K_{ss}z_o \]  
(Eq. 2)

where

\[ K_{ss} = \frac{1+i}{1-0.5i} \]  
(Eq. 3)

The significance of this approach is that the total sinkage is expressed as a function of slip (or shear displacement) and static sinkage, both of which can be computed directly. Another advantage of this approach is that no additional empirical parameters have been introduced.

- \( z_o \) = static sinkage
- \( z \) = total sinkage (static + sinkage due to slip)
- \( i \) = slip, which can expressed as \( j/x \) for straight-line motion
- \( j \) = shear displacement
- \( x \) = position of track segment relative to track
The normal pressure acting on each track segment is based on a pressure sinkage relationship proposed in [2] and is given by:

\[ p = \left( \frac{k_c}{b} + k_\phi \right) z_o^n = k \cdot z_o^n \]  
(Eq. 4)

Accounting for slip-sinkage and inserting (Eq. 2) into (Eq. 4) and rearranging yields:

\[ p = K_{ss}^{-n} \cdot k \cdot z^n \]  
(Eq. 5)

Notice as slip (or shear displacement) increases, the value of \( K_{ss}^{-n} \) decreases reducing the stiffness \( k \), which in turn yields additional sinkage.

To account for terrain “memory”, a primitive terrain query model was implemented.

The value of \( K_{ss}^{-n} \) for each element of the discretized terrain is “tracked” over the simulation.
Conclusions

- The PreMoStat UGV (PackBot) model is in a state to where it is ready to be extensively compared with test results, including investigating statistical variability.

- However, to improve the accuracy of the UGV (PackBot) system model, the values of certain parameters need to be measured/estimated, namely:
  - Soil parameters related the shear stress
  - Parameters related to damping and stiction through the drivetrain
  - Control parameters

- Although the effects of slip-sinkage have been accounted, the associated losses (bulldozing) should be integrated to more accurately predict mobility performance, especially for small radius turn maneuvers.
References


Vehicle-Based Track-Terrain Parameter Estimation
The Project

We developed methods for using UGV sensor data to estimate variables and parameters needed for traction force prediction.

The methods were evaluated using data collected from experiments with an iRobot PackBot traversing various deformable and non-deformable surfaces.

<table>
<thead>
<tr>
<th>Data</th>
<th>K, m</th>
<th>Cohesion, Kpa</th>
<th>Phi, degree</th>
<th>Fmax, N</th>
</tr>
</thead>
<tbody>
<tr>
<td>Obsvn 1</td>
<td>0.0012</td>
<td>740</td>
<td>15.697</td>
<td>43.789</td>
</tr>
<tr>
<td>Obsvn 2</td>
<td>0.0015</td>
<td>735</td>
<td>16.075</td>
<td>44.901</td>
</tr>
<tr>
<td>Obsvn 3</td>
<td>0.0014</td>
<td>750</td>
<td>16.415</td>
<td>45.905</td>
</tr>
</tbody>
</table>
Why was this done

We lack validated instruments for determining terrain parameters that can be used to predict small UGV mobility.

Need a way to collect data for guiding nominal parameter settings for vehicle-terrain interaction in UGV simulation studies.

There appear to be no databases available that can provide insight into the type of statistical variability we might expect to see in the parameters needed for models commonly used in vehicle mobility predictions.
Results and Conclusions

We showed that an instrumented robotic vehicle can be used to estimate time-varying track slip and vehicle slip angle, variables critical in determining:

a) longitudinal and lateral friction coefficients in a baseline skid-steer model, and

b) cohesion, friction angle, and deformation modulus for model-based traction force prediction.

These preliminary results support related efforts reported in the literature, and suggest that SUGVs with onboard sensors may provide a means for building a mobility database for small-scale vehicle platforms.
Recommendations

Additional testing is needed to assess accuracy against results from standard instruments, if available, as well as to generate a preliminary database useful for statistical

These ‘sensor-endowed’ vehicles are being developed and deployed on diverse terrains, and there could be a way to use this data for the benefit of prediction and design.

Support continued efforts to enable development of a mobility database using vehicle-based ‘data mining’.


In progress: journal article for J. of Terramechanics + another to J. of Dyn. Sys. Measurement and Control (ASME)
Methodology Used

The approach developed combines the use of Extended Kalman Filters (EKF) and Generalized Newton Raphson (GNR) methods in a multi-tiered algorithm.

Implicit in this approach are model bases that approximate the vehicle dynamics and the vehicle-terrain interaction.

Experiments were designed accordingly:

1. Ad hoc U-turns during sand pit testing
2. Field and indoor testing on various terrains
3. Prepared straight-line tests on sand and soil + step
Indoor tests employed the Vicon MoCap to track vehicle motion, while outdoor tests used a differential GPS.
Algorithm - 1

Small-Scale Tracked Vehicle

Kinematic Model

Observations

Estimation Algorithm

Inputs

Outputs

Truth Table

Compute

GNR Algorithm

Motor Model

Truth Table

Lateral

Motor Model

Observations

Estimation Algorithm

Outputs

Straight Line Motion Model

Verified Soil Properties

\( \alpha \) = slip angle

\( c \) = soil cohesion

\( F_L \) = tractive force left track

\( F_R \) = tractive force right track

\( i \) = slip on left/right track

\( i_L \) = slip on left track

\( i_R \) = slip on right track

\( k \) = deformation parameter

\( \text{obsvn} \) = observations

\( \phi \) = soil friction angle

\( \psi \) = yaw angle

\( \psi' \) = yaw rate/turning rate

\( I_L \) = current left motor

\( I_R \) = current right motor

\( \omega_L \) = angular speed left sprocket

\( \omega_R \) = angular speed right sprocket

\( \mu \) = lateral coefficient of friction

\( \mu_r \) = longitudinal coefficient of friction

\( x \) = displacement in x-direction body fixed coordinates

\( y \) = displacement in y-direction body fixed coordinates

\( X \) = displacement in X-direction inertial coordinate

\( Y \) = displacement in Y-direction inertial coordinate
Having a good measure of slip is critical to traction force parameter estimation.

Each track has slip,

\[ i = 1 - \frac{V}{r \omega} = 1 - \frac{V}{V_t} = \frac{V_t - V}{V_t} = \frac{V_j}{V_t} \]

Track slip impacts shear displacement along the track-terrain interface.

Our process: directly calculated slip using onboard measurements of encoder speeds together with approximated speeds based on position measurements made using either MoCap (indoor) or DGPS (outdoor).

This data, along with PackBot onboard data, was passed to algorithm.
From an ideal kinematic steer model for a tracked vehicle, introduced slip in track and side slip, so full (kinematic) state equations become:

\[
x(k) = \begin{bmatrix} X(k) \\ Y(k) \\ \Psi(k) \\ i_L(k) \\ i_R(k) \\ \alpha(k) \end{bmatrix}
\]

\[
\dot{x}(k) = \begin{bmatrix} \frac{r}{2}[\omega_L (1 - i_L) + \omega_R (1 - i_R)](\cos \psi + \tan \alpha \sin \psi) \\ \frac{r}{2}[\omega_L (1 - i_L) + \omega_R (1 - i_R)](\sin \psi - \tan \alpha \cos \psi) \\ \frac{r}{2B}[\omega_R (1 - i_R) - \omega_L (1 - i_L)] \\ 0 \\ 0 \\ 0 \end{bmatrix}
\]

*Assume slip and side slip dynamics are zero, lacking dynamic model. So we need SNC to compensate for this model uncertainty.
Estimation Algorithm – EKF

Observation model:

\[ Z_k = [X_k; Y_k; \Psi_k] \]

\[ z_k = Z_k - G(X^*_k, t_k) \]

\[ \tilde{H} = \frac{\partial G(X^*_k, t_k)}{\partial X_k} \]

Kinematic model solved each step

\[ \dot{X} = F(X^*, t) \Rightarrow X^*(t_{k-1}) = \hat{X}_{k-1} \]

\[ \dot{\phi}(t, t_{k-1}) = A(t)\phi(t, t_{k-1}) \forall, \quad \phi(t_{k-1}, t_{k-1}) = I \]
Discrete Kalman Filter

Discrete System Model

\[ x_{k-1} \]

Measurement Model

\[ H_k \]

\[ \Phi_{k-1} \]

\[ \Delta \]

\[ z_k = Z_k - H_k x_k^{(-)} \]

Observation Model

\[ K_k \]

\[ \hat{x}_k^{(-)} \]

Updated State Estimate

\[ x_k^{(+)} = \hat{x}_k^{(-)} + K_k z_k \]

Kalman Gain

\[ x(k) = \begin{bmatrix} X(k) \\ Y(k) \\ \Psi(k) \\ i_L(k) \\ i_R(k) \\ \alpha(k) \end{bmatrix} \]
Slip Estimation - Sand Court
Coefficients of Friction

Note: Probably should conduct some comparisons to existing data sets to see how well these coefficients predict trajectories.
Algorithm – 3

\( \alpha = \) slip angle
\( c = \) soil cohesion
\( F_L = \) tractive force left track
\( F_R = \) tractive force right track
\( i = \) slip on left/right track
\( i_L = \) slip on left track
\( i_R = \) slip on right track
\( k = \) deformation parameter
\( obsvn = \) observations
\( \phi = \) soil friction angle
\( \psi = \) yaw angle
\( \dot{\psi} = \) yaw rate/turning rate
\( I_L = \) current left motor
\( I_R = \) current right motor
\( \omega_L = \) angular speed left sprocket
\( \omega_R = \) angular speed right sprocket
\( \mu = \) lateral coefficient of friction
\( \mu_l = \) longitudinal coefficient of friction
\( x = \) displacement in x-direction
\( y = \) displacement in y-direction
\( X = \) displacement in X-direction inertial coordinate
\( Y = \) displacement in Y-direction inertial coordinate
Frictional Dry Sand


### TABLE 2.3 Terrain Values

<table>
<thead>
<tr>
<th>Terrain</th>
<th>Moisture Content (%)</th>
<th>n</th>
<th>$k_c$ lb/in.$^{*1}$</th>
<th>kPa/m$^{*1}$</th>
<th>$k_f$ lb/in.$^{*2}$</th>
<th>kPa/m$^{*2}$</th>
<th>c lb/in.$^2$</th>
<th>kPa</th>
<th>$\phi$ deg</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dry sand</td>
<td>0</td>
<td>1.1</td>
<td>0.1</td>
<td>0.99</td>
<td>3.9</td>
<td>1528.43</td>
<td>0.15</td>
<td>1.04</td>
<td>28°</td>
</tr>
<tr>
<td>(Land Locomotion Lab., LLL)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sandy loam (LLL)</td>
<td>15</td>
<td>0.7</td>
<td>2.3</td>
<td>5.27</td>
<td>16.8</td>
<td>1515.04</td>
<td>0.25</td>
<td>1.72</td>
<td>29°</td>
</tr>
<tr>
<td>Sandy loam</td>
<td>22</td>
<td>0.2</td>
<td>7</td>
<td>2.56</td>
<td>3</td>
<td>43.12</td>
<td>0.2</td>
<td>1.38</td>
<td>38°</td>
</tr>
<tr>
<td>Michigan (Strong, Pressure)</td>
<td>11</td>
<td>0.9</td>
<td>11</td>
<td>52.53</td>
<td>6</td>
<td>1127.97</td>
<td>0.7</td>
<td>4.83</td>
<td>20°</td>
</tr>
<tr>
<td></td>
<td>23</td>
<td>0.4</td>
<td>15</td>
<td>11.42</td>
<td>27</td>
<td>808.96</td>
<td>1.4</td>
<td>9.65</td>
<td>35°</td>
</tr>
</tbody>
</table>

Data

<table>
<thead>
<tr>
<th>Data</th>
<th>K, m</th>
<th>Cohesion, kPa</th>
<th>Phi, degree</th>
<th>Fmax, N</th>
</tr>
</thead>
<tbody>
<tr>
<td>Obsvn 1</td>
<td>0.01197</td>
<td>1210</td>
<td>28.361</td>
<td>84.115</td>
</tr>
<tr>
<td>Obsvn 2</td>
<td>0.00925</td>
<td>1350</td>
<td>25.386</td>
<td>73.944</td>
</tr>
<tr>
<td>Obsvn 3</td>
<td>0.01627</td>
<td>1085</td>
<td>30.913</td>
<td>93.3</td>
</tr>
<tr>
<td>Obsvn 4</td>
<td>0.022</td>
<td>1350</td>
<td>36.91</td>
<td>117.04</td>
</tr>
<tr>
<td>Obsvn 5</td>
<td>0.009</td>
<td>1530</td>
<td>23.03</td>
<td>66.25</td>
</tr>
</tbody>
</table>
Frictional Dry Sand

<table>
<thead>
<tr>
<th>Terrain</th>
<th>Moisture Content (%)</th>
<th>n</th>
<th>$k_e$ (lb/in.(^2))</th>
<th>$k_e$ (kN/m(^2))</th>
<th>$c$ (lb/in.(^2))</th>
<th>$c$ (kPa)</th>
<th>$\phi$ (deg)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dry sand</td>
<td></td>
<td>0</td>
<td>1.1</td>
<td>0.99</td>
<td>3.9</td>
<td>0.15</td>
<td>1.04</td>
</tr>
<tr>
<td>(Land Locomotion Lab., LLL)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sandy loam (LLL)</td>
<td>15</td>
<td>0.7</td>
<td>2.3</td>
<td>5.27</td>
<td>16.8</td>
<td>0.25</td>
<td>1.72</td>
</tr>
<tr>
<td>Sandy loam</td>
<td>22</td>
<td>0.2</td>
<td>7</td>
<td>2.56</td>
<td>3</td>
<td>0.2</td>
<td>1.38</td>
</tr>
<tr>
<td>Sandy loam (Strong, Buchele)</td>
<td>11</td>
<td>0.9</td>
<td>11</td>
<td>52.53</td>
<td>6</td>
<td>0.7</td>
<td>4.83</td>
</tr>
<tr>
<td>Michigan (Thailand)</td>
<td>23</td>
<td>0.4</td>
<td>15</td>
<td>11.42</td>
<td>27</td>
<td>1.4</td>
<td>9.65</td>
</tr>
<tr>
<td>Sandy loam (Hanamoto)</td>
<td>26</td>
<td>0.3</td>
<td>5.3</td>
<td>2.79</td>
<td>6.8</td>
<td>2.0</td>
<td>13.79</td>
</tr>
<tr>
<td>Clayey soil (Thailand)</td>
<td>38</td>
<td>0.5</td>
<td>12</td>
<td>13.19</td>
<td>16</td>
<td>0.6</td>
<td>4.14</td>
</tr>
<tr>
<td>Heavy clay (Waterways Experiment Stn., WES)</td>
<td>55</td>
<td>0.7</td>
<td>7</td>
<td>16.03</td>
<td>14</td>
<td>0.3</td>
<td>2.07</td>
</tr>
<tr>
<td>Lean clay (WES)</td>
<td>22</td>
<td>0.2</td>
<td>45</td>
<td>16.43</td>
<td>120</td>
<td>10</td>
<td>68.95</td>
</tr>
<tr>
<td>LETE sand (Wong)</td>
<td>32</td>
<td>0.15</td>
<td>5</td>
<td>1.52</td>
<td>10</td>
<td>2</td>
<td>13.79</td>
</tr>
<tr>
<td>Upland sandy loam (Wong)</td>
<td>0.79</td>
<td>32</td>
<td>102</td>
<td>42.2</td>
<td>5301</td>
<td>0.19</td>
<td>1.3</td>
</tr>
<tr>
<td>Rubicon sandy loam (Wong)</td>
<td>51</td>
<td>1.10</td>
<td>7.5</td>
<td>74.6</td>
<td>5.3</td>
<td>0.48</td>
<td>3.3</td>
</tr>
<tr>
<td>North Gower clayey loam (Wong)</td>
<td>43</td>
<td>0.66</td>
<td>3.5</td>
<td>6.9</td>
<td>9.7</td>
<td>0.54</td>
<td>3.7</td>
</tr>
<tr>
<td>Grenville loam (Wong)</td>
<td>46</td>
<td>0.73</td>
<td>16.3</td>
<td>41.6</td>
<td>24.5</td>
<td>0.88</td>
<td>6.1</td>
</tr>
<tr>
<td>Snow (U.S.)</td>
<td>24</td>
<td>1.01</td>
<td>0.008</td>
<td>0.06</td>
<td>20.9</td>
<td>0.45</td>
<td>3.1</td>
</tr>
<tr>
<td>(Harrison)</td>
<td></td>
<td>1.6</td>
<td>0.07</td>
<td>4.37</td>
<td>0.08</td>
<td>0.15</td>
<td>1.03</td>
</tr>
<tr>
<td>Snow (Sweden)</td>
<td></td>
<td>1.6</td>
<td>0.04</td>
<td>2.49</td>
<td>0.10</td>
<td>0.09</td>
<td>0.62</td>
</tr>
<tr>
<td></td>
<td></td>
<td>1.44</td>
<td>0.3</td>
<td>10.55</td>
<td>0.05</td>
<td>0.87</td>
<td>0.6</td>
</tr>
</tbody>
</table>

Source: Reference 2.3, 2.4, 2.35, and 2.36.

### TABLE 4.2-1 SUMMARY OF DISCRETE KALMAN FILTER EQUATIONS

<table>
<thead>
<tr>
<th>System Model</th>
<th>$\Delta k = \Phi k - 1 \Sigma k - 1 + \Psi k - 1$, $\Psi k \sim N(0, Q_k)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Measurement Model</td>
<td>$\Sigma k = H k \Sigma k + \Xi k$, $\Xi k \sim N(0, R_k)$</td>
</tr>
<tr>
<td>Initial Conditions</td>
<td>$E[\hat{X}(0)] = \hat{X}_0$, $E[(\hat{X}(0) - \hat{X}_0)(\hat{X}(0) - \hat{X}_0)^T] = P_0$</td>
</tr>
<tr>
<td>Other Assumptions</td>
<td>$E[\Psi_k] = 0$ for all $j, k$</td>
</tr>
<tr>
<td>State Estimate Extrapolation</td>
<td>$\hat{X}_k(-) = \Phi k - 1 \hat{X}_k - 1(+)$</td>
</tr>
<tr>
<td>Error Covariance Extrapolation</td>
<td>$P_k(-) = \Phi k - 1 P_k - 1(+) \Phi k - 1^T + Q_k - 1$</td>
</tr>
<tr>
<td>State Estimate Update</td>
<td>$\hat{X}_k(+) = \hat{X}_k(-) + K_k [\hat{X}_k - H_k \hat{X}_k(-)]$</td>
</tr>
<tr>
<td>Error Covariance Update</td>
<td>$P_k(+) = [I - K_k H_k] P_k(-)$</td>
</tr>
<tr>
<td>Kalman Gain Matrix</td>
<td>$K_k = P_k(-) H_k^T [H_k P_k(-) H_k^T + R_k]^{-1}$</td>
</tr>
<tr>
<td>Definitions</td>
<td>$F(\hat{X}(t),t) = \frac{\partial f(\hat{X}(t),t)}{\partial \hat{X}(t)} \hat{X}(t) = \hat{X}(t)$</td>
</tr>
<tr>
<td></td>
<td>$H_k(\hat{X}_k(-)) = \frac{\partial h_k(\hat{X}_k(t),t)}{\partial \hat{X}_k(t)} \bigg</td>
</tr>
</tbody>
</table>

### TABLE 6.1-1 SUMMARY OF CONTINUOUS-DISCRETE EXTENDED KALMAN FILTER

<table>
<thead>
<tr>
<th>System Model</th>
<th>$\dot{X}(t) = f(\hat{X}(t),t) + \Psi(t)$ ; $\Psi(t) \sim N(0, Q(t))$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Measurement Model</td>
<td>$\dot{X}_k = h_k(\hat{X}_k(t)) + \Xi_k$ ; $k = 1, 2, \ldots$ $\Xi_k \sim N(0, R_k)$</td>
</tr>
<tr>
<td>Initial Conditions</td>
<td>$\hat{X}(0) \sim N(\hat{X}_0, P_0)$</td>
</tr>
<tr>
<td>Other Assumptions</td>
<td>$E[\Psi(t) \Psi(t)^T] = 0$ for all $k$ and all $t$</td>
</tr>
<tr>
<td>State Estimate Propagation</td>
<td>$\dot{\hat{X}}(t) = f(\hat{X}(t),t)$</td>
</tr>
<tr>
<td>Error Covariance Propagation</td>
<td>$\dot{P}(t) = F(\hat{X}(t),t) P(t) + P(t) F^T(\hat{X}(t),t) + Q(t)$</td>
</tr>
<tr>
<td>State Estimate Update</td>
<td>$\hat{X}(t) = \hat{X}(t) + K(t) [\hat{X}(t) - H_k(\hat{X}(t),t)]$</td>
</tr>
<tr>
<td>Error Covariance Update</td>
<td>$P(t) = [I - K(t) H_k(\hat{X}(t),t)] P(t)$</td>
</tr>
<tr>
<td>Gain Matrix</td>
<td>$K(t) = (1 - K(t) H_k(\hat{X}(t),t)) P(t)$</td>
</tr>
<tr>
<td>Definitions</td>
<td>$F(\hat{X}(t),t) = \frac{\partial f(\hat{X}(t),t)}{\partial \hat{X}(t)} \hat{X}(t) = \hat{X}(t)$</td>
</tr>
<tr>
<td></td>
<td>$H_k(\hat{X}_k(t),t) = \frac{\partial h_k(\hat{X}_k(t),t)}{\partial \hat{X}_k(t)} \bigg</td>
</tr>
</tbody>
</table>
Ideal Kinematic Steer Models

For a *kinematic* model of a differentially-driven vehicle, we assume there is **no slip**, and that the wheels have controllable speeds, $\omega_1$ and $\omega_2$. The velocity of the CG in the local reference frame has a net effect from each wheel, composed as,

$$
\begin{align*}
\dot{x}_1 &= R_w \omega_1 \\
\dot{x}_2 &= R_w \omega_2
\end{align*}
$$

Note these are the velocities at the wheels.

$$
\therefore \dot{x} = \frac{1}{2} (\dot{x}_1 + \dot{x}_2) = \frac{1}{2} R_w (\omega_1 + \omega_2)
$$

The lateral motion is constrained, so, \( \dot{y} = 0 \)

The yaw rate is also composed by the net (constrained) motion of the two wheels, and you can show that:

$$
\dot{\psi} = \frac{R_w}{B} (\omega_1 - \omega_2)
$$

So the velocities in the global reference frame are,

$$
\dot{\mathbf{q}}_l = \begin{bmatrix} \dot{X} \\ \dot{Y} \\ \dot{\psi} \end{bmatrix} = \Psi(\psi) \cdot \dot{\mathbf{q}} = \begin{bmatrix} \cos \psi & -\sin \psi & 0 \\ \sin \psi & \cos \psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} \frac{R_w}{2} \cos \psi (\omega_1 + \omega_2) \\ \frac{R_w}{2} \sin \psi (\omega_1 + \omega_2) \\ \frac{R_w}{B} (\omega_1 - \omega_2) \end{bmatrix}
$$
Introduce slip

With slip, the velocity of each track is,

\[ i = \frac{r \omega - V}{r \omega} \Rightarrow V = r \omega [1 - i] \]

The velocity of the body CG is then,

\[ V = \frac{1}{2} (V_R + V_L) = \frac{r}{2} \left[ \omega_R (1 - i_R) + \omega_L (1 - i_L) \right] \]

We can estimate the turning radius from, \( \frac{V}{R} = \psi \), and the turning rate,

\[ \dot{\psi} = \frac{r}{2R} \left[ \omega_R (1 - i_R) + \omega_L (1 - i_L) \right] \]

so,

\[ R = B \frac{2}{2} \left[ \omega_R (1 - i_R) - \omega_L (1 - i_L) \right] \]
Kinematic Steer with Slip

\[
\tan \alpha = \frac{-\dot{y}}{\dot{x}}
\]

\[
i = \frac{r \omega - V}{\max(r \omega, V)}
\]

\[
\dot{X} = \frac{r}{2} \left[ \omega_L \left(1 - i_L \right) + \omega_R \left(1 - i_R \right) \right] (\cos \psi + \tan \alpha \sin \psi)
\]

\[
\dot{Y} = \frac{r}{2} \left[ \omega_L \left(1 - i_L \right) + \omega_R \left(1 - i_R \right) \right] (\sin \psi - \tan \alpha \cos \psi)
\]

\[
\psi = \frac{r}{2B} \left[ \omega_R \left(1 - i_R \right) - \omega_L \left(1 - i_L \right) \right]
\]
Conclusions

We can begin understanding how and when the level of uncertainty makes UGV operation impractical.

Closer to having a practical means for quantifying how and when a small UGV is influenced by terrain variability.

The implementation on a mobile stand-alone system may be a reasonable next-step.

It should be possible to incorporate different traction models into this methodology.

Prior information about and/or online estimation of vehicle-terrain interaction parameters can help improve UGV traversability on terrains having significant variability.
Points of Contact

Griffin Technologies
- Eric Krotkov  ekrotkov@griffin-technologies.com
- Doug Hackett  dhackett@griffin-technologies.com

University of Texas at Austin
- Raul Longoria  longoria@mail.utexas.edu

Southwest Research Institute
- Andy Moore  andrew.moore@swri.org