On-line Mobile Robot Model Identification using Integrated Perturbative Dynamics

Forrest Rogers-Marcovitz and Alonzo Kelly

Abstract We present an approach to the problem of real-time identification of vehicle motion models based on fitting, on a continuous basis, parametrized slip models to observed behavior. Our approach is unique in that we generate parametric models capturing the dynamics of systematic error (i.e. slip) and then *predict* trajectories for arbitrary inputs on arbitrary terrain. The integrated error dynamics are linearized with respect to the unknown parameters to produce an observer relating errors in predicted slip to errors in the parameters. An Extended Kalman filter is used to identify this model on-line. The filter forms innovations based on residual differences between the motion originally predicted using the present model and the motion ultimately experienced by the vehicle. Our results show that the models converge in a few seconds and they reduce prediction error for even benign maneuvers where errors might be expected to be small already. Results are presented for both a skid-steered and an Ackerman steer vehicle.

1 Introduction

Autonomous vehicle research has continuously demonstrated that a platform's precise understanding of its own mobility is a key ingredient of competent machines that perform [4]. Since ground vehicles are propelled over the earth by the two forces of gravity and traction, agile autonomous mobility relies fundamentally on understanding and exploiting the interactions between the terrain and tractive devices like wheels and tracks. Furthermore the propulsive forces depend critically on the composition of the surface over which the vehicle moves. Therefore, unless the terrain is homogeneous for the duration of an entire mission, and the vehicle models are already well calibrated, we must conclude that a capacity to calibrate (i.e. identify) vehicle models in real-time is a core requirement of high performance ground

Forrest Rogers-Marcovitz and Alonzo Kelly
Robotics Institute, Carnegie Mellon University, e-mail: {forrest,alonzo}@cmu.edu

robots. In principle, it should be possible to calibrate vehicle models in real-time based on observing residual differences between the motion originally predicted by the platform and the motion ultimately experienced. In a machine learning context, this process would be called *self-supervised learning* whereas in the adaptive control community it would be called *on-line identification*.

We concentrate on the problem of calibrating the faster-than-real-time models which occur in mobile robotic predictive control and motion planning. Such cases include obstacle avoidance and path following situations where the predicted motion of the vehicle is the basis for decisions on where to go. Our models calibrate the mapping between the control inputs and *predicted* state trajectory. Certainly this mapping depends on terrain conditions, gravity, reaction and inertial forces but we have found it unnecessary to express forces explicitly in the models. Our models nonetheless provide a fairly accurate model of the underlying dynamics over the regime of performance of present-day robots.

Motion models of ground robots have many uses. Model-based approaches have been applied to estimate longitudinal wheel slip and to detect immobilization of mobile robots [10]. Analytical models also exist for steering maneuvers of planetary rovers on loose soil [3]. The aspects of wheel-terrain interaction that are needed for accurate models are neither well known nor easily measurable in realistic situations. Other researchers have addressed the problem of model identification for ground robots. Algorithms have been developed to learn soil parameters given wheel-terrain dynamic models [8].

Our colleagues have constructed an artificial neural network that was used to learn a forward predictive model of the Crusher Unmanned Ground Vehicle [2]. This model gave good results, with fast predictions, without making any assumptions about the vehicle to ground interaction. It was trained off-line and does not attempt to adapt to varying terrain properties.

Whereas published methods are designed to use measurements to estimate present state for feedback controllers, our method learns a predictive model by capturing the underlying dynamics as a function of all of input space. Some published methods lump all of the unknown soil parameters into slip ratios and a slip angle and use velocity measurements to aid in estimation. An Extended Kalman Filter (EKF) and a Sliding Mode Observer (SMO) have been developed to estimate the slip ratio parameters by [11] and [5]. An EKF has also been used to estimate the slip angles and longitudinal slippage for a wheeled mobile robot [6].

Our method also relies on more reliable pose residuals rather than measurements of velocity. In our past work, we have developed off-line calibration techniques [1] including techniques for learning vehicle slip rates [7]. In this paper, we extend those techniques to work for real time calibration. We first develop a general land vehicle model in Section 2. This model, along with the pose residual observations, is integrated into the EKF in Section 3. Section 4 describes the experimental set-up, and the results are presented in Section 5. We present our conclusions in Section 6.

2 Vehicle Model

For any vehicle moving in contact with a surface, there are three degrees of freedom of in-plane motion as long as the vehicle remains in contact with the terrain, (Figure 1). Errors in motion prediction can therefore be reduced, without loss of generality, to instantaneous values of forward slip rate, δV_x, side slip rate, δV_y, and angular slip rate, $\delta\omega$.

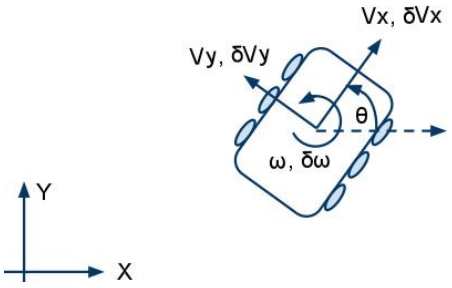


Fig. 1 Vehicle Dynamics.

Given these slip rates and the vehicle's commanded linear and angular velocity, we have the following kinematic differential equation for the time derivatives of the vehicle's 2D position and heading with respect to a ground fixed frame of reference, known hereafter as the *world frame*.

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} c\theta & -s\theta & 0 \\ s\theta & c\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \left\{ \begin{bmatrix} V_x \\ V_y \\ \omega \end{bmatrix} + \begin{bmatrix} \delta V_x \\ \delta V_y \\ \delta\omega \end{bmatrix} \right\} \tag{1}$$

$$c\theta = \cos(\theta),\ \ s\theta = \sin(\theta)$$

This model is the general case for a vehicle moving on a surface with three degrees of velocity freedom. We expressed the velocity in terms of a commanded component and an error component. Both are expressed in body coordinates because slip is likely to be simpler to express in these coordinates. The model is also relevant to rough terrain when expressed in body coordinates because the instantaneous linear velocity remains in the plane containing the wheel contact points. Ignoring suspension deflections, this plane is fixed in body coordinates. No nonholonomic constraints are present in the unperturbed model above, but they can be easily added. The perturbations, of course, are elements that will not respect such constraints even if they nominally exist. In plain terms, if the wheels are slipping sideways, the perturbations will tell us that.

We elected to use a velocity driven model because these are natural inputs for most of our robots. However, this is not a requirement of our approach. Any differential equation, such as one driven by forces or fuel flow rates, could be used. It is important to express the model in terms of control inputs rather than measure-

ments since the model will be used to predict motion before the terrain is traversed. Furthermore, the perturbation terms need not be in input space. They could be expressed as additive errors to the state $\delta\underline{x}$ or even the state derivatives $\delta\underline{\dot{x}}$. They need only span the dimensions necessary to characterize the actual errors of interest.

The calibration question is how best to model the perturbations. They will depend, in general, on terrain composition and vehicle state along with the applied, constraint, and inertial forces that terrain and vehicle state imply. We can parameterize a 3×1 vector of the slip velocity $\underline{\delta u}$, expressed as $\underline{\delta u}(\underline{u},\underline{p})$. These slip velocities depend, in our model, on the commanded velocities $\underline{u}(V_x, V_y, \omega)$ and on a set of learned slip parameters, $\underline{p}$. Accordingly, the *perturbed* system dynamics can be written in general as:

$$\underline{\dot{x}} = \underline{f}(\underline{x},\underline{u},\underline{p}) \tag{2}$$

The use of wheel rotational velocities, or integrals involving them, to form observations is not desirable since such measurements cannot observe slip directly. In addition, wheels may slip sideways and such motion would not be reflected in measurements of wheel rotations, even though it is the dominant effect of interest in many cases. Although many approaches to system identification are based on fitting the data to the differential equation, we prefer to exploit the high relative accuracy of position data (e.g. RTK GPS or visual odometry) by observing the effects of the slip velocities on the integrated dynamics. This formulation also allows the position measurements to be under-sampled relative to the command frequency; for example, low-cost GPS may only provide an position update at 1 Hz while the vehicle is commanded at 100 Hz. The vehicle path is found by integrating the equations of motion.

$$\underline{x} = \begin{bmatrix} x \\ y \\ \theta \end{bmatrix} = F(\underline{u}(t),\underline{p}) = \int f(\underline{x}(t),\underline{u}(t),\underline{p})dt \tag{3}$$

In order to produce regular observations, we reset the integral to a local origin before using the current estimate of slip parameters to produce the difference between measured and predicted changes in pose for a given path segment. To avoid a double delta symbol in the notation, let us define the uppercase vector $\underline{\mathrm{X}}$ to be the difference between two states expressed in ground fixed coordinates:

$$\underline{X}(k) = \underline{x}(t_k + n) - \underline{x}(t_k) \tag{4}$$

This gives us an integrated observation.

$$\Delta\underline{X} = \underline{X}_{measured} - \underline{X}_{predicted} \tag{5}$$

3 Extended Kalman Filter

Any vehicle's performance will depend on terrain characteristics which vary over space and time due to such effects as weather and seasonal vegetation. Only an on-line system can adapt to these changes as fast as the local environment changes during the vehicle's motion from place to place. In an operational setting, the trajectory followed by the robot will not normally be chosen to simplify model identification. In any case, there is no clear way to tell when the terrain is about to change so the system must be able calibrate on any arbitrary trajectory.

An Extended Kalman Filter (EKF) is used whose state vector is composed entirely of the parameters governing perturbative (i.e. slip) velocity. The vehicle pose estimation system functions as a sensor for the identification filter. It may use its own Kalman filter but in this paper *the* EKF will always mean the identification EKF used to identify slip parameters. In the EKF, uncertainty in both states (parameters) and the measurements is correctly treated. The transition and observation models below are used in the standard EKF algorithm.

3.1 Transition model

Our parameters are assumed to be constant over a short segment once the terrain and the inputs are known. Therefore, the state transition model is adequately modeled as static in (6). The process noise, $\underline{\varepsilon}_t$, of covariance Q_t describes the parameter uncertainty. The uncertainty models can be adaptive, allowing the noise amplitude to be increased, for example, during terrain transitions that are detected (from rapid changes in slip) or predicted (from perception). The transition matrix is simply the identity matrix, (7), so it does not appear in the transition model.

$$\underline{p}_t = g(\underline{p}_{t-1}) + \underline{\varepsilon}_t = \underline{p}_{t-1} + \underline{\varepsilon}_t \tag{6}$$

$$G_t = \frac{\partial g(\underline{p}_{t-1})}{\partial \underline{p}_t} = I_{5,5}\ (IdentityMatrix) \tag{7}$$

3.2 Observation Model

Each computation of the motion prediction dynamics integral (3) is called a *path segment*. The observation is the measured relative pose between the start and end of the current path segment, $\underline{X}_{measured}$, whereas the predictive measurement is the same quantity as predicted by the current parameter estimates and the sequence of commands issued over the segment, (8). The observation noise, $\underline{\delta}_t$, is the expected

noise of the difference between start and end poses, with covariance R_t. More detail on the observation noise is presented in Section 3.3.

$$\underline{z_t} = h(\underline{u}(t), \underline{p_t}) + \underline{\delta_t} \tag{8}$$

$$\underline{z_t} = \underline{X}_{measured} \tag{9}$$

$$h(\underline{u}(t), \underline{p_t}) = \underline{X}_{predicted} = \underline{F}(\underline{u}(t), \underline{p_t}) \tag{10}$$

Given an observed difference between the predicted and measured relative pose of a path segment, the measurement Jacobian, H, allows us to find the changes needed in the slip parameters to correct the relative pose residual. Since pose is only 3D in the plane and any sophisticated model will have many parameters, we have an under determined system. Optimal estimation, via the Kalman gain, can compute changes in n parameters to correspond to 3 pose residual errors, (11).

$$\Delta \underline{p} = K \Delta \underline{X} = PH^T [HPH^T + R]^{-1} \Delta \underline{X} \tag{11}$$

The Jacobian, H, which linearizes the change in the integrand $\underline{F}$ due to the change in parameters $\underline{p}$, can be evaluated on an arbitrary trajectory in (12).

$$H = \frac{\partial \underline{F}}{\partial \underline{p}} = \frac{\partial}{\partial \underline{p}} \int f(\underline{x}(t), \underline{u}(t), \underline{p}) dt = \int \frac{\partial}{\partial \underline{p}} f(\underline{x}(t), \underline{u}(t), \underline{p}) dt \tag{12}$$

The last step in (12) uses Leibniz's rule to convert the derivative of the integral to the integral of the derivative. Numerical finite differences on the integral could also be performed, but Leibniz' rule gives a closed form solution for our formulation. The chain rule is used for the inner derivative to simplify the analytical differentiation, (13).

$$\frac{\partial \underline{f}}{\partial \underline{p}} = \frac{\partial \underline{f}}{\partial \underline{\delta u}} \frac{\partial \underline{\delta u}}{\partial \underline{p}} \tag{13}$$

The first needed derivative, describing the change in predicted motion relative to the change in slip rates, is simply the rotation matrix in the motion model.

$$\frac{\partial \underline{f}}{\partial \underline{\delta u}} = \begin{bmatrix} c\theta & -s\theta & 0 \\ s\theta & c\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{14}$$

Slip rate is represented as a second order polynomial surface over input space. This approach allows us to learn, not only the present slip, but a model for how slip depends on arbitrary inputs - even arbitrary functions of time. The polynomial terms are formed over commanded speed V, and angular velocity is dropped from the command vector in favor of curvature, κ. The result of these formulation decisions is a *slip surface* that is a general paraboloid over this input space.

$$\begin{aligned}
\delta V_x &= \alpha_{1,x}\,\kappa + \alpha_{2,x}V + \alpha_{3,x}\,\kappa V + \alpha_{4,x}\,\kappa^2 + \alpha_{5,x}V^2 \\
\delta V_y &= \alpha_{1,y}\,\kappa + \alpha_{2,y}V + \alpha_{3,y}\,\kappa V + \alpha_{4,y}\,\kappa^2 + \alpha_{5,y}V^2 \\
\delta V_\omega &= \alpha_{1,\omega}\,\kappa + \alpha_{2,\omega}V + \alpha_{3,\omega}\,\kappa V + \alpha_{4,\omega}\,\kappa^2 + \alpha_{5,\omega}V^2
\end{aligned} \tag{15}$$

Constant terms were not used since they cause phantom drift when the vehicle is, in fact, not moving. Commanded side velocity was not considered as our vehicles (skid-steered and Ackerman-steered) have no commanded side velocity. Gravity, expressed in the body frame, would likely be needed in the model for slopes, but we have not performed such experiments yet. Additional terms may be added for other relevant inputs, disturbances, or state variables.

Ideally, the inputs used should be whatever the system actually accepts as inputs at the level of abstraction that is being calibrated. For example, one could calibrate a decoupled model of a Mars rover which accepts 4 wheel velocities or a state space model of the same system that presents a curvature-accelerator interface like an automobile.

The slip rate surface parameters are grouped into a column vector to form the 21 state vector of the EKF. Equation (16) gives the second needed derivative used in inner derivative, (13).

$$\underline{p} = [\alpha_{1,x}\ \alpha_{2,x}\ \alpha_{3,x}\ \alpha_{4,x}\ \alpha_{5,x}\ \alpha_{1,y}\ \alpha_{2,y}\ \alpha_{3,y}\ \alpha_{4,y}\ \alpha_{5,y}\alpha_{1,\omega}\ \alpha_{2,\omega}\ \alpha_{3,\omega}\ \alpha_{4,\omega}\ \alpha_{5,\omega}]^\top$$

$$\begin{aligned}
\frac{\partial \underline{\delta u}}{\partial \underline{p}} = U &= \begin{bmatrix} C & 0_{1,5} & 0_{1,5} \\ 0_{1,5} & C & 0_{1,5} \\ 0_{1,5} & 0_{1,5} & C \end{bmatrix} \\
C &= \left[\kappa,\ V,\ \kappa V,\ \kappa^2,\ V^2\right]
\end{aligned} \tag{16}$$

3.3 Uncertainty

Given the EKF observation residual, (17), the observation uncertainty is a combination of the measurement uncertainties of the initial and final state plus the uncertainty in the integrand, F, which depends on both the initial pose and parameter uncertainties, (18). Equation (18) assumes that the two pose measurements are uncorrelated - which we find to be an adequate model for RTK GPS. For systems that use WAAS, or lower grade GPS, a model of the correlation of the two relative pose errors can be used. GPS errors are of course not white or Gaussian, but a simple check such as a validation gate should be able to detect large GPS jumps and reject any such measurements. Good inertial navigation systems or visual odometry are likely to be better solutions than GPS since they exhibit excellent short term accuracy that can observe wheel slip.

$$\Delta \underline{z} = \underline{z} - h(\underline{p}) = \underline{x}_f - \underline{x}_i - F(\underline{x}_i, \underline{p}, \underline{u}_i^f) \tag{17}$$

$$R_z = R_{x_f} + R_{x_i} + R_F \tag{18}$$

$$R_F = R_{F_{x_i}} + R_{F_p} = J_{x_i} R_{x_i} J_{x_i}^\top + J_p R_p J_p^\top \tag{19}$$

In estimating the uncertainty in $\underline{X}_{predicted}$, it is very important to correctly account for the fact that the predicted relative pose error can be caused by a combination of errors in the initial conditions and errors in the parameters used in the integral. That is, a lateral pose error could be caused simply by an error in the yaw angle used as initial conditions.

The second term in (19) describes the uncertainty in the *predicted* relative pose due to errors in the slip parameters. We can take the parameter uncertainty and it's Jacobian straight from the EKF; thus $J_p = H$ in (12) and $R_p = P$, the parameter covariance, in (11).

The first term in (19) describes the uncertainty in the *predicted* relative pose due to error in the *measured* initial pose. The pose measurement uncertainty provided by the position estimation system is used for this. For the Jacobian, the partial inside the integrand is taken with respect to the initial state rather than the current state at each point in time as the integral in computed.

$$J_{x_i} = \frac{\partial \underline{F}}{\partial \underline{x_i}} = \frac{\partial}{\partial \underline{x_i}} \int f(\underline{x_i}, \underline{x}, \underline{p}, \underline{u}) dt = \int \frac{\partial \underline{f}(\underline{x_i}, \underline{x}, \underline{p}, \underline{u})}{\partial \underline{x_i}} dt \tag{20}$$

We can isolate the dependence on the initial angle by rewriting the world-relative yaw angle, θ, as the sum of the constant initial angle, θ_i, and the time varying deviation from the initial angle, $\Delta\theta(t)$. This can be represented as a product of two rotation matrices.

$$f(\underline{x_i}, \underline{x}, \underline{p}, \underline{u}) = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} c\theta_i & -s\theta_i & 0 \\ s\theta_i & c\theta_i & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} c\Delta\theta & -s\Delta\theta & 0 \\ s\Delta\theta & c\Delta\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} V_x(\underline{p}) \\ V_y(\underline{p}) \\ \omega(\underline{p}) \end{bmatrix} \tag{21}$$

The Jacobian is now straightforward since the rotation through the constant initial angle can be taken outside the integral. We define $\Delta\underline{x}_f^i$ as the state change expressed in a coordinate system fixed to the initial pose.

$$\frac{\partial \underline{F}(\underline{x_i}, \underline{p}, \underline{u_i^f})}{\partial \underline{x_i}} = \frac{\partial}{\partial \underline{x_i}} \begin{bmatrix} c\theta_i & -s\theta_i & 0 \\ s\theta_i & c\theta_i & 0 \\ 0 & 0 & 1 \end{bmatrix} \int f(\underline{x}, \underline{p}, \underline{u}) dt = \frac{\partial}{\partial \underline{x_i}} \begin{bmatrix} c\theta_i & -s\theta_i & 0 \\ s\theta_i & c\theta_i & 0 \\ 0 & 0 & 1 \end{bmatrix} * \Delta\underline{x}_f^i \tag{22}$$

The derivative of a matrix with respect to a vector produces a third order tensor (3x3x3) in general. We can simplify this complexity by taking the derivative of the rotation matrix with respect to each element of the vector of initial conditions and then multiplying by the relative pose change, $\Delta\underline{x}_f^i$, to get a single column of the Jacobian. First we take the derivative with respect to initial heading.

$$\frac{\partial \underline{F}(\underline{x_i}, \underline{p}, \underline{u_i^f})}{\partial \theta_i} = \begin{bmatrix} -s\theta_i & -c\theta_i & 0 \\ c\theta_i & -s\theta_i & 0 \\ 0 & 0 & 1 \end{bmatrix} * \Delta\underline{x}_f^i = \begin{bmatrix} -\Delta y \\ \Delta x \\ 0 \end{bmatrix} \tag{23}$$

The changes in position produced, Δx and Δy, are expressed in world coordinates. Because the rotation matrix is independent of the initial position, (x_i, y_i), the other two columns of the Jacobian are zero. This gives us the final Jacobian:

$$J_{x_i} = \begin{bmatrix} 0 & 0 & -\Delta y \\ 0 & 0 & \Delta x \\ 0 & 0 & 0 \end{bmatrix} \tag{24}$$

The result is intuitively correct since this is the Jacobian of a rotation of a displacement vector rotated through an angle at the start. If we multiply out the whole first uncertainty term, we get:

$$J_{x_i} R_{x_i} J_{x_i}^{\top} = \begin{bmatrix} \Delta y^2 & -\Delta y \Delta x & 0 \\ -\Delta y \Delta x & \Delta x^2 & 0 \\ 0 & 0 & 0 \end{bmatrix} \sigma_{\theta_i \theta_i} \tag{25}$$

The uncertainty propagation can be used to estimate the error in trajectory prediction for planning purposes. Vehicle motion planners can improve performance and better avoid obstacles if they know the likely regions occupied by the vehicle after executing an arbitrary trajectory. The uncertainty in the integrand, R_F, found in Equation (19), gives the uncertainty in the predicted pose from the uncertainty in the parameters and the uncertainty in the initial state.

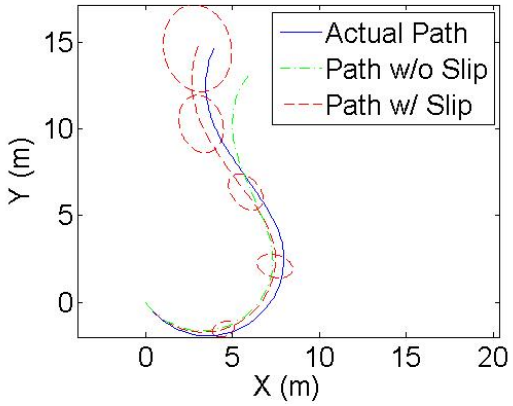


Fig. 2 1-σ Position Uncertainty along the path with slip prediction compared to the actual path and ideal path without slip prediction.

Figure 2, shows a typical arbitrary vehicle path. The solid line shows the actual vehicle path over 10 seconds. The dash-dot line shows the path generated with no slip prediction. The dashed line is the path generated with the described slip prediction along with 1-σ position uncertainty bounds every 2 seconds found via Equation (19). The predicted path closely tracks the actual path while providing useful uncertainty bounds for the trajectory planner.

4 Experimental Set-Up

Data has been collected on an automated LandTamer, a six wheeled skid-steered vehicle with a hydraulic/gear drive system, that has been retrofitted for autonomy, Fig-3(a). Skid-steered vehicles are often used as outdoor mobile robots due to their robust mechanical structure and high maneuverability. For such a vehicle, execution of paths with any curvature will create wheel slip which makes both kinematic and dynamic modeling difficult. The Land Tamer vehicle is retrofitted with a high end Novatel SPAN Inertial Measurement Unit (IMU) unit and with Real-Time Kinematic Global Positioning System (RTK GPS). The RTK receiver reports the vehicle's position to centimeter accuracy.

(a) Land Tamer Vehicle (b) recBot eGator Vehicle

Fig. 3 Test Vehicles

In one experiment, data was collected on the Land Tamer vehicle in a gravel lot after a heavy rain. Mud and wet gravel added to the terrain variability. Data collection occurred as the vehicle was commanded to drive in circles at various curvatures and speeds. Four speeds of $\{0.25,0.50,0.75,1.0\}$ m/s and three curvatures of $\{0.4,0.5,0.6\}$ m^{-1} were used. The vehicle started at the smallest curvature and smallest speed; it then was commanded the other speeds in increasing order. This was repeated for the other two curvatures. The test was continuous, and data from the transitions were included in the data set to assess the need to model transients (i.e. times when the inputs are changing). These first data sets were collected on flat ground to eliminate the need for perception of the ground slope used in prediction. The EKF used the collected data, in temporal order, to optimize the slip model parameters. The data collection time was just over nine minutes.

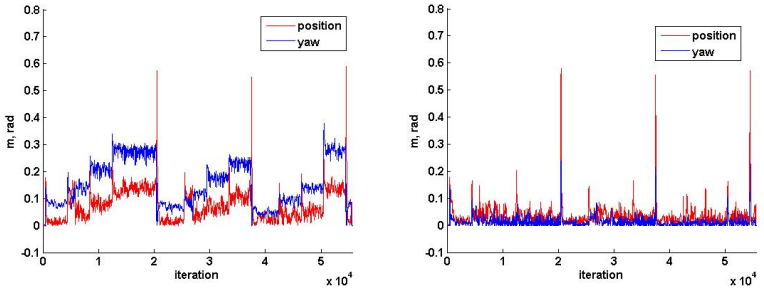
Data was also collected on our "recBot" vehicle, a medium size drive-by-wire UGV, Fig-3(b). The recBots are built based on the Deere eGator vehicle which is well suited to operations on light off-road terrain. The vehicle is Ackerman-steered providing contrast to the previous skid-steered collected data. The skid rates depend, in this case, on the turn angle of the front wheels. Curvature was calculated from the turning angle, γ, and the wheelbase distance, a, $\kappa = \frac{1}{a}\tan\gamma$. The recBot was retrofitted with a similar high-end INS system. Data was collected as the recBot

was randomly driven around on a grass lawn for just over five and half minutes at speeds up to 4.8 m/s. The grass was mostly level and flat, although tractor treads in the ground provided additional variance in the skip rates.

5 Results

Overlapping one-second path segments were used for computing the path residuals. It is significant that residuals were large enough over such a short time period to permit identification of the model despite the sensor noise. Each iteration occurred with pose measurement updates at 100 Hz; thus, the segment start points were separated by 10 milliseconds. Keeping track of the past internal derivatives, $\partial \underline{f}/\partial \underline{p}$, along the length of the path segment made the on-line integration very efficient. The entire EKF algorithm can easily run faster than real-time.

As the EKF was run on the collected Land Tamer circular data, a future path segment is predicted from the current pose with the current estimated slip parameters. The remaining predicted pose residuals from the measured end pose were minimal with a few exceptions during transients (perhaps due to un-modeled actuation delays), Figure 4(b). For comparison, the relative pose residuals computed without slip prediction are shown in Figure 4(a).



(a) Pose Residuals without Slip Prediction (b) Pose Residuals with Slip Prediction

Fig. 4 Predicted Pose Residuals, 1 second path segments

Parameter values converge in a few seconds once the vehicle enters a new speed/curvature regime, and the models are formulated to be valid throughout input space. The system cannot distinguish innovations caused by changes in terrain from those caused by parameter errors so the net result is to adapt to terrain changes in the same way at the same rate. Stochastic models have also been successfully identified in an off-line setting [7] in order to quantify the remaining variation after the systematic model converges. Future work will integrate the stochastic model into the EKF for on-line identification.

It is significant that the slip rate surface parameters were all initialized to zero, making no assumptions about the vehicle-ground interaction. This is more evidence of rapid on-line adaptation to the terrain. The slip surface parameter variation over the EKF run is shown in Figure 5 versus the algorithm iteration index. The parameters are clearly adjusting to incorporate new knowledge when new operating regimes are first encountered.

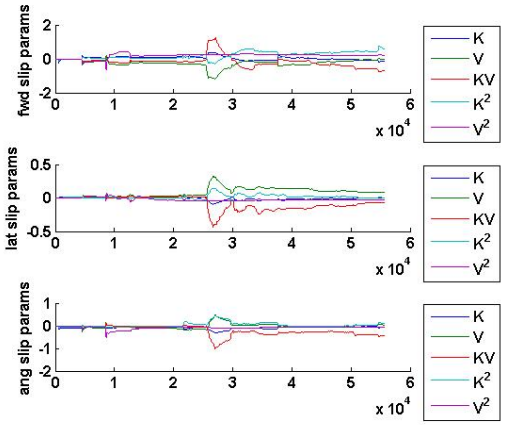


Fig. 5 Parameter Variation during EKF iterations

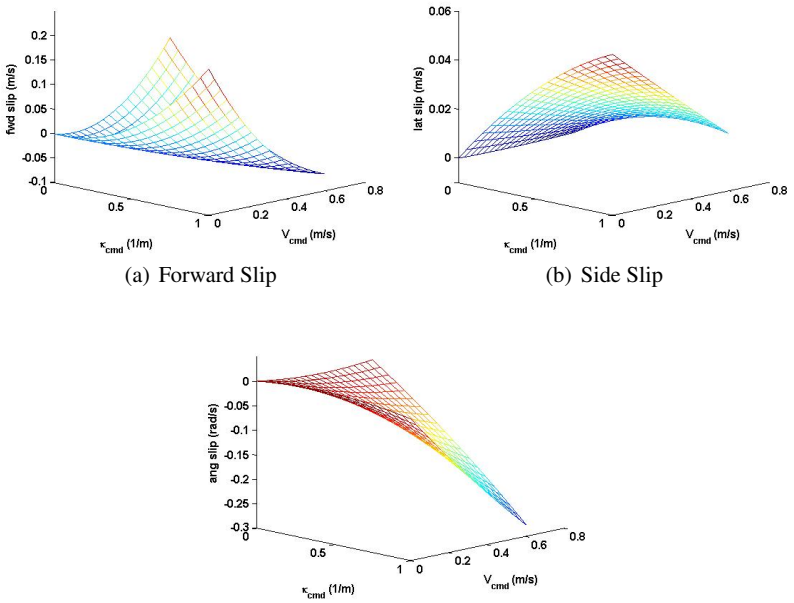
The final learned slip rate surfaces for the LandTamer are shown in Figure 6. The surfaces are the final predicted mean slip rates over the observed commanded curvatures and speeds. The surfaces themselves tell us that lateral slip is nearly nonexistent for such a vehicle and the dominant slip directions are best characterized as rotational and forward.

Even for the benign conditions in this dataset, the EKF relative position estimation over 1 second path segments was two times better than the result obtained with no slip prediction, Table 1. The EKF provides an improvement of ten times reduction in the predicted path residual orientation error over path prediction with no slip rates.

Table 1 Relative Pose Error Comparison with Standard Deviation

Algorithm	Position (m)	Orientation (rad)
None	$.075 \pm .058$	$.157 \pm .084$
EKF	$.034 \pm .042$	$.014 \pm .019$

The improvements are more pronounced as the path segment length or prediction time increases, Figure 7. While the position error for a path segment, with no predicted slip, increases rapidly as the path time length increases, the EKF position



(a) Forward Slip (b) Side Slip

(c) Angular Slip

Fig. 6 Slip Surfaces learned via EKF

error is increases at a much slower rate. With a path time length of six seconds, we see a position error improvement from 1.6 meters to under 21 centimeters - an error reduction of over about 8 times. Clearly a motion model for this vehicle on this terrain can be calibrated, and it can be calibrated well.

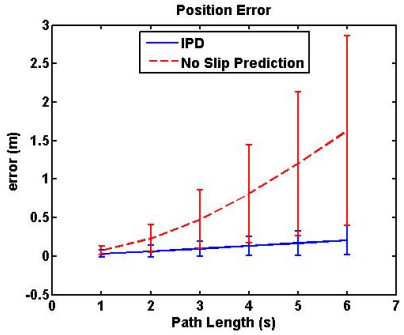


Fig. 7 Land Tamer Predicted Position Residual with regard to Path Length

The EKF was also run on the collected RecBot data. Recall this test was performed at speeds up to 4.8 m/s and steering inputs were arbitrary. The prediction was more difficult due to less frequent (10 Hz) pose updates, which affects the path integration, and an inaccurate steering wheel angle encoder. In addition, the vehicle was driven at high speeds through more aggressive maneuvers. While the EKF error residuals were not as small, the relative improvements, over not using slip in the motion prediction, are similar to those of the Land Tamer data set. The position error grows rapidly as the path segment length is increased, Figure 8. At six seconds, the position error decreases around 3 times. The other plots for the RecBot are not shown here due to space considerations.

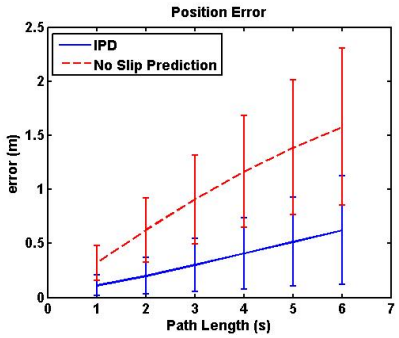


Fig. 8 RecBot Predicted Position Residual with regard to Path Length

6 Conclusion

A method has been developed for real-time identification of vehicle dynamic models. The method uses perturbative models to represent slip in all degrees of freedom, and the model relates slip velocities to arbitrary input signals. Our residuals are based on pose predictions generated from integrating the model over a short period of time. Though more complex than classical techniques, the approach permits the direct use of high quality sensing like RTK GPS, or even under-sampled pose measurements, as ground truth. This technique requires the linearization of an integral of the system dynamics with respect to the model parameters. The process is clearly observable over all of input space for our vehicles and it converges in a few seconds on the tested terrain.

The vehicle motion model is a general formulation that applies to any vehicle type and any terrain. In this paper, we have applied it to both skid-steered and Ackerman-steered vehicles on loose gravel and grass. Future work will expand the EKF for rolling terrain, for a range of vehicle types, and for multiple terrain types. In addition, command delay transients will be simultaneously learned. Ultimately, perception based slip prediction will permit vehicle control and motion planning to predict the terrain characteristics based on its previewed geometry and appear-

ance. Our formulation also leads to an equivalent capacity to calibrate a stochastic dynamics model by using the integral of the associated linear variance equation.

Our method improves prediction for a few seconds of trajectory prediction while adapting rapidly to terrain. The improved models produced by this technique will lead to significant improvements in the performance of model predictive controllers, particularly in difficult terrains or during aggressive maneuvers.

Acknowledgments

This work was supported by the U. S. Army Research Office under contract Real-Time Identification of Wheel Terrain Interaction Models for Enhanced Autonomous Vehicle Mobility (contract number W911NF-09-1-0557).

References

1. M. Alarfaj and F. Rogers-Marcovitz, "Interaction of Pose Estimation and Online Dynamic Modeling for a Small Inspector Spacecraft," *ESA Small Satellites Systems and Services 4S Symposium*, Funchal, Madeira, Portugal, May 2010.
2. M. Bode, "Learning the Forward Predictive Model for an Off-Road Skid-Steer Vehicle," tech. report CMU-RI-TR-07-32, Robotics Institute, Carnegie Mellon University, September, 2007.
3. G. Ishigami, A. Miwa, K. Nagatani, and K. Yoshida, "Terramechanics-Based Model for Steering Maneuver of Planetary Exploration Rovers on Loose Soil," *Journal of Field Robotics*, Vol. 24, No. 3, March, 2007, pp. 233-250.
4. A. Kelly, A. Stentz, O. Amidi, M. Bode, D. Bradley, A. Diaz-Calderon, M. Happold, H. Herman, R. Mandelbaum, T. Pilarski, P. Rander, S. Thayer, N. Vallidis, and R. Warner. "Toward Reliable Off Road Autonomous Vehicles Operating in Challenging Environments," *The International Journal of Robotics Research*, Vol 25, No 5/6, 2006.
5. E. Lucet, C. Grand, D. Sall, and P. Bidaud, "Dynamic sliding mode control of a four-wheel skid-steering vehicle in presence of sliding", *Romansy*, Tokyo, Japan, July 2008.
6. C. B. Low and D. Wang, "Integrated Estimation for Wheeled Mobile Robot posture, velocities, and wheel skidding perturbations," *IEEE International Conference on Robotics and Automation*. Rome, Itatly, August 2007.
7. F. Rogers-Marcovitz, "On-line Mobile Robotic Dynamic Modeling using Integrated Perturbative Dynamics," tech. report CMU-RI-TR-10-15, Robotics Institute, Carnegie Mellon University, April 2010.
8. L. Seneviratne, Y. Zweiri, S. Hutangkabodee, Z. Song, X. Song, S. Chhaniyara, S. Al-Milli, and K. Althoefer. "The modeling and estimation of driving forces for unmanned ground vehicles in outdoor terrain," *International Journal of Modeling, Identification, and Control*, Vol. 6, No. 1, 2009
9. X. Song, L. D. Seneviratne, K. Althoefer, and Z. Song, "A Robust Slip Estimation Method for Skid-Steered Mobile Robots," *Intl. Conf. on Control, Automation, Robotics, and Vision*, Hanoi, Vietnam, Dec. 2008
10. C. Ward, and K. Iagnemma, "A Dynamic Model-Based Wheel Slip Detector for Mobile Robots on Outdoor Terrain," *IEEE Transactions on Robotics*, Vol. 24, No. 4, pp. 821-831, Aug. 2008.
11. J. Yi, H. Wang, J. Zhang, D. Song, S. Jayasuriya, and J. Liu, "Kinematic Modeling and Analysis of Skid-Steered Mobile Robots With Applications to Low-Cost Inertial-Measurement-Unit-Based Motion Estimation," *IEEE Transactions on Robotics*, Vol. 25, No. 6, Oct. 2009.