

A Unified Perturbative Dynamics Approach to Online Vehicle Model Identification

Neal Seegmiller, Forrest Rogers-Marcovitz, Greg Miller and Alonzo Kelly

Abstract The motions of wheeled mobile robots are largely governed by contact forces between the wheels and the terrain. Inasmuch as future wheel-terrain interactions are unpredictable and unobservable, high performance autonomous vehicles must ultimately learn the terrain by feel and extrapolate, just as humans do. We present an approach to the automatic calibration of dynamic models of arbitrary wheeled mobile robots on arbitrary terrain. Inputs beyond our control (disturbances) are assumed to be responsible for observed differences between what the vehicle was initially predicted to do and what it was subsequently observed to do. In departure from much previous work, and in order to directly support adaptive and predictive controllers, we concentrate on the problem of predicting candidate trajectories rather than measuring the current slip. The approach linearizes the nominal vehicle model and then calibrates the perturbative dynamics to explain the observed prediction residuals. Both systematic and stochastic disturbances are used, and we model these disturbances as functions over the terrain, the velocities, and the applied inertial and gravitational forces. In this way, we produce a model which can be used to predict behavior across all of state space for arbitrary terrain geometry. Results demonstrate that the approach converges quickly and produces marked improvements in the prediction of trajectories for multiple vehicle classes throughout the performance envelope of the platform, including during aggressive maneuvering.

1 Introduction

We concentrate in this paper on the problem of calibrating the faster-than-real-time models that are used in mobile robot predictive control and motion planning. In obstacle avoidance, lane change maneuvers, and path following the predicted motion

All Authors
Robotics Institute, Carnegie Mellon University, e-mail: {nseegmil, forrest, gmiller, alonzo}@rec.ri.cmu.edu

of the vehicle is the basis for the precise specification of the inputs. The mapping between inputs and resultant behavior depends critically on terrain conditions which vary significantly over time and space so it cannot be pre-programmed. The mapping must be either perceived from non-contact sensing or learned from experience. We take the latter approach in this paper.

Motion models of ground robots have many uses, but the aspects of wheel-terrain interaction that are needed for accurate models are neither well known nor easily measurable in realistic situations. Published methods are mostly concerned with robust path-following (e.g. [9]), or the estimation of instantaneous wheel slip for feedback controllers. Some have developed terramechanics-based models for slip estimation that require knowledge of tire constants and soil parameters. These have been applied to detect immobilization [13], and to investigate steering maneuvers for planetary rovers on loose soil [5]. Some methods lump all unknown tire and soil parameters into slip ratios and angles; extended Kalman filters have been developed for real-time estimation of slip ratios using velocity measurements [14] [8].

Other researchers have addressed the problem of model identification for ground robots; [11] provides an algorithm to learn soil parameters, and [2] learns a mapping between slip ratios and visual terrain classes. However, there is little precedent in the literature for the calibration of *predictive* models despite the fact that they are fundamental to virtually every decision that a mobile robot makes. The only precedent known to us is [3] where our colleagues constructed an artificial neural network that was trained offline. Our method learns a predictive model by capturing the underlying dynamics as a function of all of input space and it is calibrated online based on trajectories executed under normal operation.

The literature on identifying stochastic differential equations is even less developed, at least in robotics applications. In statistics literature there are regression techniques for parameterized (or heteroscedastic) covariances (e.g. [4]), but not in the context of a differential equation. One of the authors [7] presented methods for calibration of odometry error models which are similar to the methods used here. By contrast, [1] presents offline coordinate ascent methods for tuning Kalman filters automatically. Other than these two references, we can find nothing that even slightly anticipates our efforts here to calibrate stochastic dynamics online. In this paper we present a kind of meta Kalman filter, running in real time, which calibrates the uncertainty in the system model used in the pose estimation filter.

Classical model identification techniques often require the numerical differentiation of measurements. For example, the *Springer Handbook of Robotics* explains a method of identifying the inertial parameters of a manipulator which requires the bandpass filtered 1st and 2nd derivatives of joint angle data [12]. Unlike these classical techniques, our method exploits the excellent short term accuracy of pose sensing that is available on mobile robots (inertial navigation, real-time kinematic GPS, or visual odometry) by using measurements of relative pose rather than velocity. In effect, we integrate the model rather than differentiate the measurements. In our recent initial work, we have developed online calibration techniques for learning vehicle slip rates [10]. In this paper, we extend those techniques to a more elegant formulation of the perturbative dynamics that incorporates all of initial condition

errors, 3D terrain, and stochastic disturbances, all using the same underlying model. We first develop a somewhat general 3D kinematic vehicle model in Section 2. This model, along with the pose residual observations, is integrated into an EKF in Section 3 to calibrate deterministic slip and in Section 4 to calibrate residual random disturbance covariance. The results are presented in Section 5 along with the experimental set-up which is followed by brief conclusions in Section 6.

2 System Modeling

Our fundamental approach is to linearize the system dynamics in order to capture, in a dynamical model, the first order evolution of pose prediction error caused by input disturbances. The disturbances are inputs to deterministic and stochastic differential equations and their values depend on the state of the environment and of the vehicle. Once the model is linearized, the perturbative dynamics provide a derived system model describing the mean behavior of deterministic error. Likewise, stochastic calculus provides the first order evolution of the state covariance, so the same linearization can be used to estimate the remaining random error in a probabilistic sense. In both cases, the calibration process is performed online using a Kalman filter and, in this way, the system can adapt rapidly to changes in the terrain. The state vectors in the identification systems are the parameters which characterize the disturbances as functions over the terrain, the inputs, and the applied forces.

For any vehicle moving over a terrain surface, ignoring the suspension deflections, there are three instantaneous degrees of freedom of motion as long as the vehicle remains in contact with the terrain, (Figure 1).

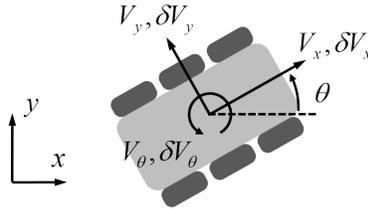


Fig. 1 Vehicle Dynamics. Three degrees of freedom remain in the general case after terrain contact is enforced. Disturbances and inputs are expressed in the body frame.

The true inputs to a vehicle motion model typically have dimensions of power, force, linear velocity, and curvature or angular velocity. However, we are interested in predictive models of the platform under the influence of its control system, so the system boundary encloses the controller as well. We find a velocity driven model to be most appropriate. This choice also simplifies the perturbative dynamics by making them driftless and making the transition matrix available in closed form.

Given the vehicle's commanded linear and angular velocities, we have the following unconstrained kinematic differential equation for the time derivatives of the position and yaw with respect to a ground fixed frame of reference.

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} c\theta c\beta & c\theta s\beta s\gamma - s\theta c\gamma & 0 \\ s\theta c\beta & s\theta s\beta s\gamma + c\theta c\gamma & 0 \\ 0 & 0 & \frac{c\gamma}{c\beta} \end{bmatrix} \begin{bmatrix} V_x \\ V_y \\ V_\theta \end{bmatrix} \quad (1)$$

$$c = \cos(), s = \sin(), \gamma = \text{roll}, \beta = \text{pitch}, \theta = \text{yaw}$$

A more precise model would be a 6 degree of freedom differential equation subject to 3 constraints requiring the suspension and the roll γ , pitch β , and altitude z to adjust the wheel contact patches to remain in contact with the terrain. This level of precision would complicate the formulation and require a numerical solution for a kind of constrained transition matrix. Such precision is also unwarranted here because the above model is the nominal model only. The context is already one of characterizing deviations from this nominal model.

The terrain is assumed to be rigid and the predicted attitude angles above are computed by a perception system. If the terrain is not rigid, attitude and altitude disturbances can be added to the model, but note that such disturbances do not have the dimensions of velocities so their effects do not compound with time as do errors in dead reckoning. In plainer terms, the attitude error at the end of a 3 second prediction depends, to first order, on the terrain model at that instant rather than the history of attitude errors to that point. Conversely, wheel slip is a velocity disturbance which must be integrated to ascertain its first order effect on terminal position error. For this reason, we omit attitude and altitude errors from the perturbative model and thereby avoid the need to linearize (or even represent) the constraints at all.

Subject to the above caveats, this model is the general case for a vehicle moving on an arbitrary rigid surface with the (maximum possible) three degrees of velocity freedom. The model is relevant to rough terrain motion prediction because it is the inputs, rather than the state, which are confined to 3 degrees of freedom. Our general motion model, with pose vector $\underline{\rho} = [x \ y \ \theta]^T$ and input vector $\underline{u} = [V_x \ V_y \ V_\theta]^T$, is therefore a differential equation of the form:

$$\dot{\underline{\rho}} = \underline{f}(\underline{\rho}, \underline{u}) \quad (2)$$

2.1 Linearized Dynamics

We will now develop the linearized perturbation dynamics of this system. The first and most crucial step is to compute the linear relationship, known as the *transition matrix*, between the pose at any point in time and that at any later point in time. We can estimate the effects of the disturbances $\delta \underline{u}(t)$ on the pose errors $\delta \underline{\rho}(t)$ by writing the linear perturbation dynamics:

$$\delta \underline{\dot{\rho}} = F(t) \delta \underline{\rho}(t) + G(t) \delta \underline{u}(t) \quad (3)$$

The Jacobians $F(t)$ and $G(t)$ depend on the reference trajectory and are taken with respect to the variables defined in Eq. (1). The attitude angles are treated as known inputs based on earlier comments so the system dynamics are really of the form $\underline{\dot{\rho}} = f(\underline{\rho}, \underline{u}, \underline{\Omega})$ for attitude angles $\underline{\Omega}$. After a little manipulation (see [6] for detail), the Jacobians reduce to:

$$F(t) = \frac{\partial f}{\partial \underline{\rho}} = \begin{bmatrix} 0 & 0 & -\dot{y} \\ 0 & 0 & \dot{x} \\ 0 & 0 & 0 \end{bmatrix} \quad (4)$$

$$G(t) = \frac{\partial f}{\partial \underline{u}} = \begin{bmatrix} c\theta c\beta & c\theta s\beta s\gamma - s\theta c\gamma & 0 \\ s\theta c\beta & s\theta s\beta s\gamma + c\theta c\gamma & 0 \\ 0 & 0 & \frac{c\gamma}{c\beta} \end{bmatrix} \quad (5)$$

The transition matrix of the system is easy to derive in this case because all powers of the system Jacobian F vanish. Consider the matrix integrating factor:

$$\Psi(t, \tau) = \int_{\tau}^t F(\zeta) d\zeta = \begin{bmatrix} 0 & 0 & -\Delta y \\ 0 & 0 & \Delta x \\ 0 & 0 & 0 \end{bmatrix} \quad (6)$$

The predicted *history point displacements* are defined as $\Delta x(t, \tau) = x(t) - x(\tau)$ and $\Delta y(t, \tau) = y(t) - y(\tau)$. It is well-known in linear systems theory that when the above matrix commutes with itself, as it does in this case, its matrix exponential is the transition matrix:

$$\Phi(t, \tau) = e^{\Psi(t, \tau)} \approx I + \Psi(t, \tau) = \begin{bmatrix} 1 & 0 & -\Delta y \\ 0 & 1 & \Delta x \\ 0 & 0 & 1 \end{bmatrix} \quad (7)$$

It will also be convenient to define the *input transition matrix*:

$$\Gamma(t, \tau) = \Phi(t, \tau)G(\tau) \quad (8)$$

Now the main result for this section is given by what we will call the *vector superposition integral*:

$$\delta \underline{\rho}(t) = \Phi(t, t_0) \delta \underline{\rho}(t_0) + \int_{t_0}^t \Gamma(t, \tau) \delta \underline{u}(\tau) d\tau \quad (9)$$

This result expresses how the effects of errors in initial conditions $\delta \underline{\rho}(t_0)$ and the (systematic) input disturbances $\delta \underline{u}(\tau)$ are projected forward and integrated over time to produce the errors in the pose.

3 Systematic Model Identification

This section formulates a system identification solution which utilizes the linearized perturbation dynamics. Because the system state is always the integral of its velocity, producing the correct velocity will produce the correct state, whether the velocity errors are caused by slip or some other phenomena. We will therefore, without loss of generality, represent errors in motion prediction in terms of instantaneous values of forward slip rate δV_x , lateral slip rate δV_y , and angular slip rate δV_θ . Such input perturbations are additive to the inputs so they are expressed in the coordinates of the body frame where they are likely to be constant under steady state conditions.

Of course, the slip rates will certainly depend on the terrain and the trajectory so they will not be constant under non steady-state conditions. Therefore, the general relationship between time varying slip rates and pose errors is not a function; it is the functional given by Eq. (9) that depends on the entire error history. In this case, the relevant theory for finding an unknown function is variational optimal control. The real-time solution of the resulting Euler-Lagrange (partial differential) equations seems ill-advised, so we will use parameterization to convert to a more conventional estimation problem. The small disturbance inputs will be assumed to depend on an unknown set of parameters $\underline{\alpha}$, so that $\delta \underline{u}(\underline{\alpha}) = [\delta V_x \ \delta V_y \ \delta V_\theta]^T$. Substituting into Eq. (9):

$$\delta \underline{\rho}(\underline{\alpha}, t) = \Phi(t, t_0) \delta \underline{\rho}(t_0) + \int_{t_0}^t \Gamma(t, \tau) \delta \underline{u}(\underline{\alpha}, \tau) d\tau \quad (10)$$

Once the reference trajectory is specified, this is a vector-valued function of a vector of parameters and time. A suggested parameterization over the terrain and inputs is provided in Section 3.3.

3.1 Linearizing the Perturbation Integral

For the identification system, we will need the linear algebraic relationship between the error inputs $\delta \underline{u}$ and the error pose $\delta \underline{\rho}(t)$ that is predicted to occur at the end of the prediction interval $(t - t_0)$. For our parameterized pose errors, the derivative of Eq. (10) is a Jacobian matrix. Since differentiation can be moved inside the integral sign by Leibniz rule and inside the matrix product by the rules of matrix differentiation, the derivative of Eq. (10) is:

$$J_\alpha = \frac{\partial \delta \underline{\rho}(\underline{\alpha}, t)}{\partial \underline{\alpha}} = \int_{t_0}^t \Gamma(t, \tau) \frac{\partial \delta \underline{u}(\underline{\alpha}, \tau)}{\partial \underline{\alpha}} d\tau = \int_{t_0}^t \Gamma(t, \tau) U_\alpha(\tau) d\tau \quad (11)$$

3.2 Formulating the Kalman Filter

This section formulates a Kalman filter that calibrates the predictive model online based on experience. We will reinterpret the system dynamics in Eq. (1) (with slip velocities added) as a measurement process where the change in pose $\Delta \underline{\rho}$ is reinterpreted as an observation of an unknown $\delta \underline{u}$.

$$\underline{h}(\underline{x}) = \Delta \underline{\rho}_{pred} = \int_{t_0}^t \begin{bmatrix} c\theta c\beta & c\theta s\beta s\gamma - s\theta c\gamma & 0 \\ s\theta c\beta & s\theta s\beta s\gamma + c\theta c\gamma & 0 \\ 0 & 0 & \frac{c\gamma}{c\beta} \end{bmatrix} \begin{bmatrix} V_x(\tau) + \delta V_x(\tau) \\ V_y(\tau) + \delta V_y(\tau) \\ V_\theta(\tau) + \delta V_\theta(\tau) \end{bmatrix} d\tau \quad (12)$$

Note that the prediction is a nonlinear function of the angular slip $\delta V_\theta(\tau)$ because it affects the yaw angle in the rotation matrix *inside* the integral. It is for this reason that we chose to use the full nonlinear prediction in an extended Kalman filter rather than use the linearized dynamics in Eq. (9). We used the linearized dynamics, rather, to compute the Jacobian. We have used the notation $\underline{\rho}(t)$ for the system motion state, and called it a *pose*, in order to distinguish it from the state vector \underline{x} of the identification Kalman filter.

The overall approach is to predict, at regular intervals, given the input trajectory $\underline{u}(t)$ for the last few seconds, a prediction of the change in pose $\underline{\rho}(t) - \underline{\rho}(t_0)$. This prediction is then compared to an actual measurement of the pose change to form an innovation that the filter must explain in terms of errors in the slip parameters. It would be possible to use recently produced terrain models in order to calibrate perception errors as well, but we have chosen to provide the attitude angles from the historical state as known inputs.

Note that our integrated dynamics formulation introduces the three gauge freedoms of motion in the plane due to the introduction of the initial conditions. If all measurements were transformed by a rigid planar transform, the innovations would be unaffected because they must use the initial pose measurement as initial conditions. One approach is to define error states that absorb the error of the initial measurement relative to the others, but we have chosen instead to form the innovation in the coordinates of the initial body frame to eliminate the initial conditions altogether.

3.3 Systematic Model

Our models are predictive, so they must be formulated in terms of predictable quantities regardless of whatever measurements of recent motions may be available. Measurements of applied forces at the wheels are not typically available, but the inertial forces caused by wheel reactions can be measured directly or computed from velocities.

Slip velocities are represented as functions of linear and angular velocity, their products (representing lateral acceleration), and applied gravitational force (com-

puted from attitude angles). This representation includes the *slip angle* of automotive engineering as the coefficient relating lateral slip to longitudinal velocity, but it permits other linear relationships to be learned as well. The approach allows us to learn a model for how slip depends on arbitrary terrain and inputs - even as they vary over time. Such a general model is a prerequisite for identifying the system online based on any trajectories it executes. None of our vehicles accept a lateral velocity command, so the slip velocity is parameterized over commanded speed V_x , and angular velocity V_θ . The result of these formulation decisions is a *slip surface* defined over this input space, expressed in coordinates fixed to the body. Of course, this model can be modified arbitrarily to suit different situations.

$$\begin{aligned}\delta V_x &= \alpha_1 V_x + \alpha_2 |V_\theta| + \alpha_3 V_x |V_\theta| + \alpha_4 g_x \\ \delta V_y &= \alpha_5 V_x + \alpha_6 V_\theta + \alpha_7 V_x V_\theta + \alpha_8 g_y \\ \delta V_\theta &= \alpha_9 V_x + \alpha_{10} V_\theta + \alpha_{11} V_x V_\theta + \alpha_{12} g_x + \alpha_{13} g_y\end{aligned}\tag{13}$$

The quantity $|V_\theta|$ appears in order to force longitudinal slip to be an even function of turn direction. The components of gravity g_x and g_y are computed from the known magnitude of gravity and the attitude angles.

The identification filter state vector comprises the parameters, $\underline{\alpha}$:

$$\underline{x} = [\alpha_1 \ \alpha_2 \ \dots \ \alpha_N]^T\tag{14}$$

The elimination of the gauge freedoms comes at the cost of introducing a measurement transformation. The transformed measurement is the difference between the pose estimation system measurements of terminal and initial pose ($\underline{\rho}_{f,meas}$ and $\underline{\rho}_{i,meas}$ respectively), converted to initial pose coordinates.

$$\underline{z} = \underline{\rho}_{f,meas}^i = (R_i^w)^{-1}(\underline{\rho}_{f,meas} - \underline{\rho}_{i,meas})\tag{15}$$

The Jacobian is $H = J_\alpha$, a $3 \times N$ matrix derived above in Section 3.1. The measurement uncertainty is derived from the uncertainty in the pose estimation system, being careful to express any needed correlations, including the correlation introduced in Eq. (15) by the conversion of coordinates.

4 Stochastic Model Identification

The random error behavior of the system can be “calibrated” in the sense that the covariance of pose predictions can be required to agree with the observed scatter of earlier predictions. Such an approach can be confusing because we are calibrating the equation that normally serves as the covariance dynamics in a Kalman filter - with another Kalman filter.

The underlying system dynamics are not linear as described above. It was possible to compute a nonlinear predictive measurement $\underline{h}(\underline{x})$ in the systematic case, but

here we must be content with a linear approximation or attempt nonlinear covariance propagation online. We chose a linear approximation because the computation must be fast and the trajectories are always relatively short. Once the decision of a linear filter is made, the random error dynamics are easy to derive from the systematic. Recall that the relationship between input noise covariance $\Xi(t)$ and pose covariance $\Pi(t)$ in continuous time is given by what we will call the *matrix superposition integral*:

$$\Pi(t) = \Phi(t, t_0)\Pi(t_0)\Phi(t, t_0)^T + \int_{t_0}^t \Gamma(t, \tau)\Xi(\tau)\Gamma(t, \tau)^T d\tau \quad (16)$$

All of the (trajectory dependent) matrices above are known except for the input random disturbance covariance $\Xi(t) = \text{Exp}[\delta\underline{u}(t)\delta\underline{u}(t)^T]$. Once it is known, the pose prediction covariance $\Pi(t) = \text{Exp}[\delta\underline{\rho}(t)\delta\underline{\rho}(t)^T]$ can be computed. The notation is chosen to distinguish these matrices from analogous ones called P and Q in the Kalman filter used to estimate them.

The linearization of the prediction integral proceeds analogously to the deterministic case. To save space, we will present only the highlights. The Jacobian of the pose covariance Π taken with respect to the input covariance Ξ is the derivative of a matrix with respect to a matrix - a 4th order tensor. In the simplest case, the noise sources are constants, assumed not to depend on the trajectory, and $\Xi(t)$ is a 3×3 symmetric positive definite matrix with 6 independent elements. The Jacobian can also be regarded as a set of six matrices. The derivative of $\Pi(t)$ with respect to the (i, j) element of Ξ is:

$$J_{\xi_{ij}} = \frac{\partial \Pi(t)}{\partial \xi_{ij}} = \int_{t_0}^t \underline{\gamma}_i(t, \tau)\underline{\gamma}_j(t, \tau)^T d\tau \quad (17)$$

where $\underline{\gamma}_i(t, \tau)$ is the i th column of $\Gamma(t, \tau)$.

While a Kalman filter with a “state matrix” can be defined, it is conceptually simpler to collect the 6 independent elements of Π into a state vector and the independent elements of Ξ into a measurement vector and reorganize the Jacobian as appropriate. The Kalman filter is then analogous to the deterministic case where the measurement is the sample covariance matrix S computed from the pose innovations *after* the systematic component of error has been removed.

$$S(t_k) = \frac{1}{(n-1)} \sum_{k=1}^{k=n} \delta\underline{\rho}(t_k)\delta\underline{\rho}(t_k)^T, \quad \delta\underline{\rho} = \underline{\rho}_{f,meas}^i - \underline{\rho}_{f,pred}^i \quad (18)$$

When calibrating online, the luxury of repeating the same path multiple times to observe scatter is unavailable. It takes just a little effort to define the predicted variance of n samples taken from n different distributions. It can be shown from the total probability theorem [7] that the average of all of the predicted covariance matrices for each of the trajectories is the predicted covariance for the sample. However, our experiments have produced very consistent estimates of covariance based

on presenting innovations to the stochastic identification filter one single predicted covariance $\Pi(t)$ at a time.

5 Results

Experiments were conducted on Crusher, a six-wheeled skid-steered vehicle with an advanced active suspension. Crusher is capable of autonomously driving through deserts, mountains, forests, wetlands, and other extreme environments. In order to show applicability to other platforms, tests were also conducted on the LandTamer (skid-steered, hydraulic) and RecBot (Ackerman-steered, electric). In all cases, a high-end IMU and differential GPS unit were used for ground truth motion measurement. Our method should work just as well with visual odometry or any other system that measures motion, on the scale of a few seconds, with error significantly less than the prediction errors being resolved.



Fig. 2 From left to right: Crusher, LandTamer, RecBot

The first results presented are for the Crusher vehicle driving on rough, grassy terrain at Camp Roberts in California. In this dataset Crusher traverses steep slopes of up to 29° (see Fig. 3) which enables the identifier to learn the dependence of slip on the x and y components of the gravity vector, as well as commanded forward and angular velocities. The vehicle is commanded to drive at speeds up to 6 m/s and angular velocities up to 4 rad/s. The improvement in predicted change in pose at the end of two-second path segments is shown in Fig. 4. The standard deviation of along track error and cross track error are reduced by 71% and 81% respectively. The standard deviation of heading prediction error is reduced by 90%. Note that the mean error is also reduced from 1.9 meters to near zero. In these and all other scatter plots, all of the data is processed but only 1000 data points are plotted. The data points are equally spaced in time and span the entire experiment.

The time to converge depends primarily on the initial parameter estimates and the diversity of input path segments. Fig. 5(a) shows the pose prediction performance on holdout Camp Roberts data after calibrating for limited periods ranging from 0 to 300 seconds. Based on pose prediction error, the filter converges within 4 minutes of

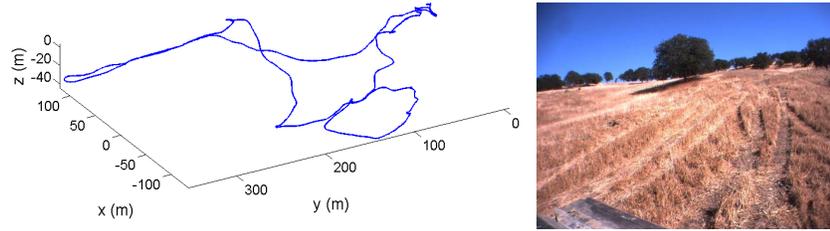


Fig. 3 *Left.* The path of the Crusher vehicle in the Camp Roberts test. During the test Crusher traverses steep slopes; roll angles range from -28 to 29° and pitch ranges from -22 to 17° . *Right.* Image captured by one of Crusher’s cameras; Crusher traverses both a dirt road and tall dry grass.

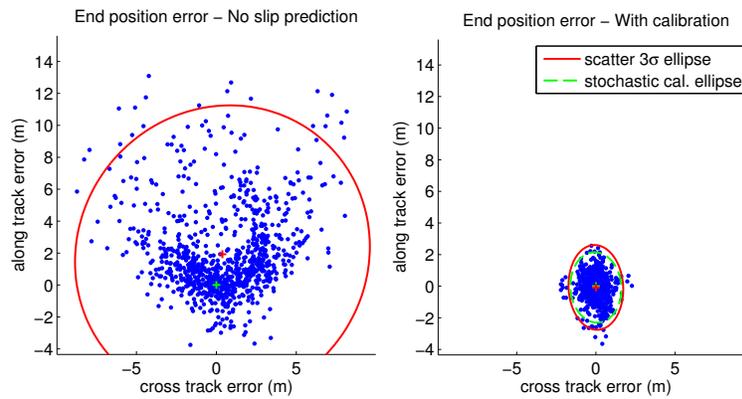


Fig. 4 Scatter plots of along track and cross track error for the Crusher vehicle at Camp Roberts. Each point represents predicted pose error at the end of a two-second path segment. The left figure shows predicted pose error with no slip calibration, the right figure shows *online* prediction error during calibration. The three standard deviation error ellipse of the points is shown by the solid red line. The dashed green line ellipse (just inside the red ellipse) is the average pose uncertainty predicted by the stochastic calibration filter.

driving when starting from the uncalibrated case (i.e. all slip parameters initialized to zero). When starting from the mean calibration for the Crusher vehicle on other datasets, the filter converges in only seconds, Fig. 5(b).

The slip surfaces learned by the filter on the Camp Roberts dataset are shown in Fig. 6. Note that forward slip is negatively correlated with the absolute value of the commanded angular velocity, $|V_\theta|$. Lateral slip depends primarily on centripetal acceleration ($V_x V_\theta$) as expected. Angular slip is predominantly a linear function of the commanded angular velocity. As expected, the filter learned a positive correlation between forward slip and the x component of the gravity vector ($\delta V_x = \dots + 0.256g_x$) and between lateral slip and the y component ($\delta V_y = \dots + 0.043g_y$).

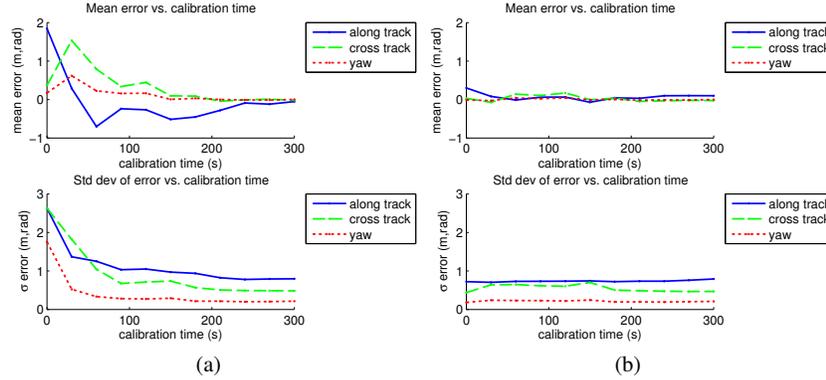


Fig. 5 Plots of the mean and standard deviation of pose prediction error vs. calibration time on the Camp Roberts dataset. The model is calibrated for 0-300 seconds of driving then evaluated on holdout data (the remainder of the 13 minute data log). Figure (a) shows results when starting from a completely uncalibrated model (all slip parameters initialized to zero), and Figure (b) shows results when starting from the average calibration on other Crusher datasets.

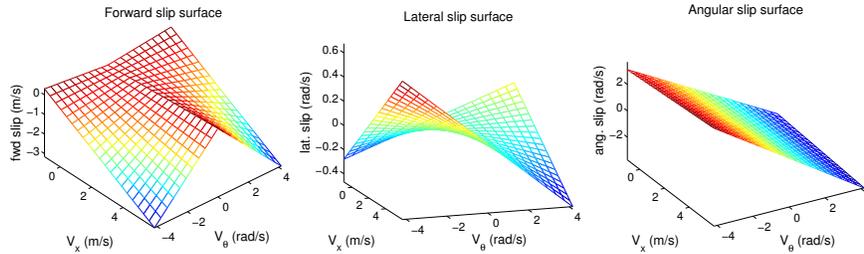


Fig. 6 Plots of the slip surfaces for the Crusher Camp Roberts dataset. These surfaces show the *predicted* forward, lateral, and angular slip as a function of the commanded forward and angular velocity (V_x and V_θ), according to (13). These surfaces correspond to zero x and y components of the gravity vector (i.e. driving on flat ground).

Figure 7 provides a visual summary of the extreme cases that are being predicted well in the Crusher datasets. First, it is important to note that Crusher’s effective turn rate is only a third of the commanded rate. This is partly due to the fact that four of six wheels are slipping sideways and resisting motion in a tight turn. There is evidently no yaw rate feedback used to compensate these gross errors. Furthermore, when turning on a hill, the wheels on the high side carry no load and are therefore unable to generate traction. The result is that turning becomes impossible and turn commands cause more or less translational motion perpendicular to the terrain gradient. These effects and others depend on the orientation of the terrain gradient in the body frame, just as our model is formulated to learn them.

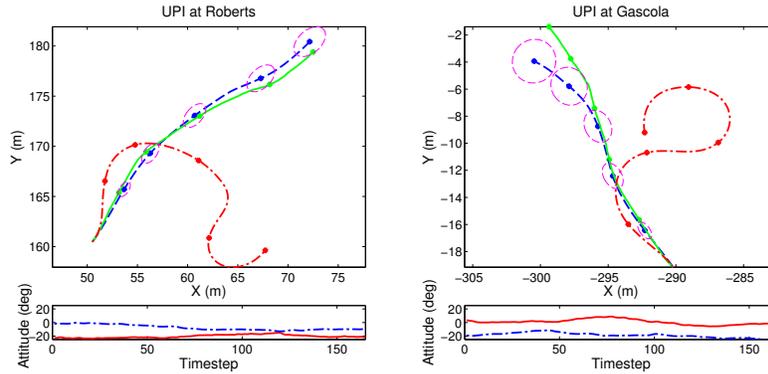


Fig. 7 Predicted and measured paths under different conditions for Crusher. The plan views show predicted paths without slip modeling (dash-dot red) with slip modeling (dashed blue) and measured path (solid green). Attitude signals roll (solid red) and pitch (dash-dot blue) are shown underneath. The paths are ten seconds long. The ellipses are estimates of uncertainty at two second intervals. *Left.* Crusher diving at Camp Roberts with a -20° roll. Commanded turns in either direction are correctly predicted to be impossible. *Right.* Crusher driving uphill at Gascola on 15° to 20° slope. 25% longitudinal wheel slip is correctly predicted.

Crusher was also driven on muddy slopes during a rainstorm at Gascola, Pennsylvania (see Fig. 8). Large reductions in motion prediction error were observed similar to the Camp Roberts results despite the difficult weather conditions (see Fig. 9). The standard deviation of along track, cross track, and heading error were reduced by 71%, 82%, and 87% respectively.

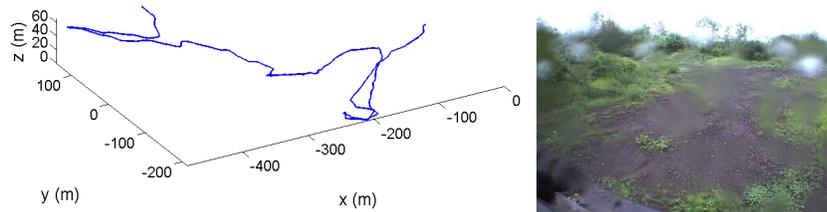


Fig. 8 *Left.* Path of the Crusher vehicle in the Gascola test. During the test, roll angles range from -25° to 25° and pitch ranges from -26° to 16° . *Right.* Image captured by one of Crusher’s cameras; Crusher traverses muddy terrain and vegetation. Note the raindrops on the lens in the image.

In another experiment, data was collected on the LandTamer vehicle in a muddy gravel lot after a heavy rain. Data was collected as the vehicle was commanded to drive in circles at various speeds and curvatures ($0.25\text{-}1.0\text{ m/s}$ and $0.4\text{-}0.6\text{ m}^{-1}$). As

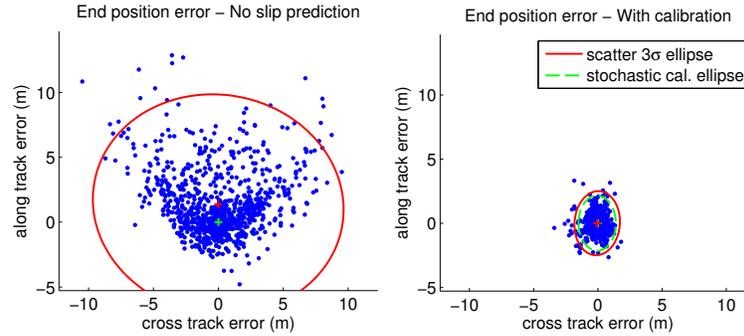


Fig. 9 Scatter plots of along track and cross track error for the Crusher vehicle at Gascola. Each dot represents predicted pose error two seconds in the future.

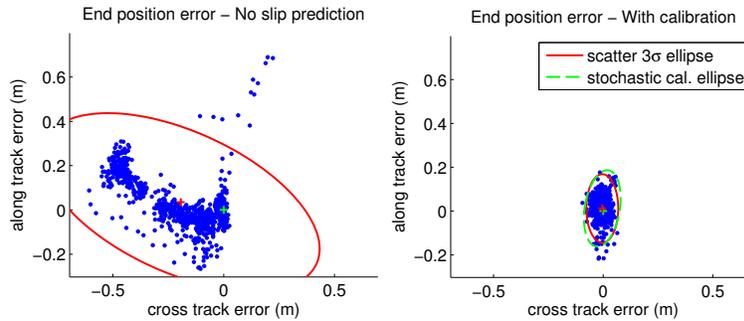


Fig. 10 Scatter plots of along track and cross track error for the LandTamer vehicle driving circles on wet gravel. Each dot represents predicted pose error two seconds in the future.

seen in the error scatter plot, Fig. 10, the large cross track error bias was almost completely removed by the calibration of wheel slip.

Finally, tests were also conducted on the RecBot vehicle, a medium size drive-by-wire UGV. The vehicle is Ackerman-steered in contrast to the previous skid-steered data sets. Data was collected as the RecBot was driven randomly 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, except for large tractor treads that added variance to the slip rates. Results for the RecBot vehicle are presented in Fig. 11.

In addition to learning models of wheel slip, our formulation can be used to learn models of the vehicle powertrain. The powertrain model maps nominal velocity commands (V_x , V_θ) to actual wheel angular velocities (as measured by encoders) and is learned online in parallel with the slip model. The powertrain modeling is not the dominant effect in our experiments, so we omit the details due to space limitations.

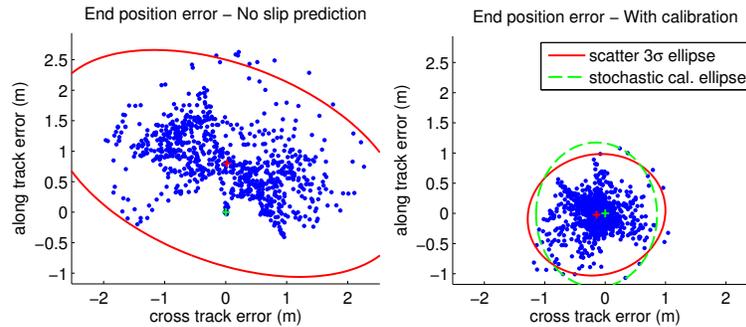


Fig. 11 Scatter plots of along track and cross track error for the RecBot vehicle driving in a grassy yard. Each dot represents predicted pose error four seconds in the future. Improvements are not as pronounced here because the terrain and the trajectories are less extreme.

6 Conclusion

The capacity to adapt to changes in traction while predicting aggressive maneuvers accurately is a fundamental requirement of self-sufficient, high-performance robots. Because the relevant mechanical properties of the terrain are not directly observable, predicting motion accurately has always been regarded as a difficult problem. We undertook it anyway because we felt that even a poor model was better than the present state of the art which ignores such issues completely. We have developed a capacity to enable mobile robots to be much more informed about their own mobility, both in terms of the mapping from their inputs to outputs and in terms of the residual random error in that mapping.

While our black box approach teaches us little about the underlying terramechanics, it has been quite successful in predicting motion under conditions of significant wheel slip, on steep slopes, during aggressive maneuvers. In the presented experiments, Crusher experienced lateral accelerations of up to 0.5 g's as a result of slopes, and 0.3 g's as a result of maneuvers. Our approach applies without modification across multiple vehicle classes including Crusher which represents the possible worst case of high speed skid steer platforms. Because our vehicle model is velocity-based and not force-based, skidding with locked brakes and certain other slip scenarios are not predicted, but will be investigated in future work.

The implementation is straightforward and lightweight from a processing point of view. Path integrals are the same computations already performed in model predictive control, but we need only the path followed and only one path segment every few seconds. The systematic and stochastic Kalman filters are of 13 and 6 states respectively and can run at frequencies in excess of 100 Hz. In effect, the computations are negligible compared to the processing load of motion control and pose estimation. We hope to show in future work how our identification system can coexist with a model predictive control system and a pose estimation system so that both benefit

from continuously calibrated, more accurate models of the platform motion during aggressive maneuvering.

Acknowledgements This research was made with U.S. Government support under and awarded by the Army Research Office (W911NF-09-1-0557), the Army Research Laboratory (W911NF-10-2-0016), and by the DoD, Air Force Office of Scientific Research, National Defense Science and Engineering Graduate (NDSEG) Fellowship, 32 CFR 168a.

References

1. Abbeel, P., Coates, A., Montemerlo, M., Ng, A.Y., Thrun, S.: Discriminative training of kalman filters. In: Proc. Robotics: Science and Systems. Cambridge, USA (2005)
2. Angelova, A., Matthies, L., Helmick, D., Sibley, G., Perona, P.: Learning to predict slip for ground robots. In: Proc. IEEE International Conference on Robotics and Automation (2006)
3. Bode, M.: Learning the forward predictive model for an off-road skid-steer vehicle. Tech. Rep. CMU-RI-TR-07-32, Robotics Institute, Pittsburgh, PA (2007)
4. Hoff, P., Niu, X.: A covariance regression model (2011). ArXiv:1102.5721v1 [stat.ME]
5. Ishigami, G., Miwa, A., Nagatani, K., Yoshida, K.: Terramechanics-based model for steering maneuver of planetary exploration rovers on loose soil. *J. Field Robotics* **24**(3), 233–250 (2007)
6. Kelly, A.: Linearized error propagation in odometry. *The International Journal of Robotics Research* **23**(1), 179 – 218 (2004)
7. Kelly, A.: Fast and easy systematic and stochastic odometry calibration. In: Proc. IEEE International Conference on Intelligent Robots and Systems (2006)
8. Low, C.B., Wang, D.: Integrated estimation for wheeled mobile robot posture, velocities, and wheel skidding perturbations. In: Proc. IEEE International Conference on Robotics and Automation, pp. 2355–2360 (2007)
9. Lucet, E., Grand, C., Sallé, D., Bidaud, P.: Dynamic sliding mode control of a four-wheel skid-steering vehicle in presence of sliding. In: Proc. RoManSy. Tokyo, Japan (2008)
10. Rogers-Marcovitz, F., Kelly, A.: On-line mobile robot model identification using integrated perturbative dynamics. In: Proc. International Symposium on Experimental Robotics (2010)
11. Seneviratne, L., Zweiri, Y., Hutangkabodee, S., Song, Z., Song, X., Chhaniyara, S., Al-Milli, S., Althoefer, K.: The modeling and estimation of driving forces for unmanned ground vehicles in outdoor terrain. *International Journal of Modeling, Identification, and Control* **6**(1), 40–50 (2009)
12. Siciliano, B., Khatib, O. (eds.): Springer Handbook of Robotics, chap. 14. Springer, Berlin, Heidelberg (2008)
13. Ward, C., Iagnemma, K.: A dynamic-model-based wheel slip detector for mobile robots on outdoor terrain. *IEEE Transactions on Robotics* **24**(4), 821–831 (2008)
14. Yi, J., Wang, H., Zhang, J., Song, D., Jayasuriya, S., Liu, J.: Kinematic modeling and analysis of skid-steered mobile robots with applications to low-cost inertial-measurement-unit-based motion estimation. *IEEE Transactions on Robotics* **25**, 1087–1097 (2009)