2007 IEEE International Conference on
Robotics and Automation
Roma, Italy, 10-14 April 2007

Model-Based Wheel Slip Detection for Outdoor Mobile Robots

Chris C. Ward and Karl Iagnemma

Abstract— **This paper introduces a model-based approach to estimating longitudinal wheel slip and detecting immobilized conditions of autonomous mobile robots operating on outdoor terrain. A novel tire traction/braking model is presented and used to calculate vehicle dynamic forces in an extended Kalman filter framework. Estimates of external forces and robot velocity are derived using measurements from wheel encoders, IMU, and GPS. Weak constraints are used to constrain the evolution of the resistive force estimate based upon physical reasoning. Experimental results show the technique accurately and rapidly detects robot immobilization conditions while providing estimates of the robot's velocity during normal driving. Immobilization detection is shown to be robust to uncertainty in tire model parameters. Accurate immobilization detection is demonstrated in the absence of GPS, indicating the algorithm is applicable for both terrestrial applications and space robotics.**

I. Introduction

MOBILE robot position estimation systems typically rely (in part) on wheel odometry as a direct estimate of displacement and velocity [1],[2]. On high-traction terrain and in combination with periodic GPS absolute position updates, such systems can provide an accurate estimate of a robot's position. However, when driving over low-traction terrain, deformable terrain, steep hills, or during collisions with obstacles, an odometry-based position estimate can quickly accumulate large errors due to wheel slip. Between absolute position updates, an odometry-based system is unable to differentiate between a robot that is immobilized with its wheels spinning and one that is driving normally. Autonomous robots should quickly detect that they are immobilized in order to take appropriate action, such as planning an alternate route away from the low-traction terrain region or implementing a traction control algorithm. Additionally, robust position estimation is required for accurate map registration.

Wheel slip can be accurately estimated through the use of encoders by comparing the speed of driven wheels to that of undriven wheels [3]; however this does not apply for all-wheel drive vehicles or those without redundant encoders. Ojeda and Borenstein have proposed comparing redundant wheel encoders against each other and against yaw gyros as a fuzzy indicator of wheel slip, even when all wheels are driven [4], and have also proposed a motor current-based slip estimator [5]; however this technique requires accurate current measurement and terrain-specific parameter tuning. A body of work exists in the automotive community related to traction control and anti-lock braking systems (ABS); however, this work generally applies at significantly higher speeds than is typical for autonomous robots [6],[7].

A large amount of work has utilized Kalman filters with inertial and absolute measurements to enhance dead reckoning and estimate lateral slip [8]-[10]. The notion of an effective tire radius, which can indirectly compensate for some longitudinal slip, is presented in [11]. None of this work, however, explicitly considers the effects of longitudinal wheel slip or vehicle immobilization.

A potentially simple approach to detecting robot slip and immobilization is to analyze GPS measurements. However, nearby trees and buildings can cause signal loss and multipath errors and changing satellites can cause position and velocity jumps [12],[13]. Additionally, GPS provides low frequency updates (e.g. typically near 1 Hz [14]) making GPS alone too slow for immobilization detection.

Another potentially simple approach could rely on comparison of wheel velocities to a robot body velocity estimate derived from integration of a linear acceleration measurement in the direction of travel. As shown in a companion paper [15]; however, such an approach is not robust at low speeds during travel on rough, outdoor terrain.

Here a method is presented for detecting robot wheel slip and immobilization that does not require redundant wheel encoders or motor current measurements. The proposed approach uses a dynamic vehicle model fused with wheel encoder, inertial measurement unit (IMU), and (optional) GPS measurements in an extended Kalman filter to create an estimate of the robot's longitudinal velocity. An insight of this approach is the realization that a robot becomes immobilized due to an external force resisting motion. The proposed algorithm utilizes a novel tire traction/braking model in combination with sensor data to estimate external resistive forces acting upon the robot and calculate the robot's acceleration and velocity. Weak constraints are used to constrain the evolution of the resistive force estimate based upon physical reasoning. The algorithm has been shown to accurately detect immobilized conditions on a variety of terrain types and provide an estimate of the robot's velocity during "normal" driving. The algorithm has been run in real time onboard a mobile robot and has proven robust to periods of GPS and IMU dropout.

This paper is organized as follows. In Section II the

Manuscript received September 14, 2006. This work was supported by the DARPA LAGR program. The authors are with the Department of Mechanical Engineering, Massachusetts Institute of Technology, Cambridge, MA 02139 USA (e-mail: chriscw@mit.edu, kdi@mit.edu).

vehicle dynamic model and tire model are presented. In Section III the slip detection algorithm is described. In Section IV experimental results are presented. In Section V conclusions are drawn from this work and future work is suggested.

I. Dynamic Models

A. Robot Configuration

The robot configuration considered in this work is shown in Fig. 1. The robot has four rubber pneumatic tires and is a front-wheel differential-drive configuration with undriven rear wheels that are freely-rotating castors mounted to a rear pivot joint suspension. Robot body and tire forces are shown in Fig. 2. The dynamic models presented below are specific to this robot configuration; however the modeling process is adaptable to other wheeled vehicle configurations.

B. Vehicle Dynamics

Modeled forces acting on the robot include gravity, a lumped external disturbance force, and tire forces acting at the four tire-terrain contact patches (Fig. 2). The disturbance force represents external forces such as wind resistance or the force caused by collision with an obstacle. In this work we limit the disturbance force to forces resisting vehicle motion.

Tire forces are composed of normal, traction/braking, rolling resistance, and lateral force components. For the robot considered, the rear traction/braking forces can be assumed negligible since these wheels are undriven. The rear lateral forces can be neglected because the rear castors spin freely.

The vehicle acceleration along the body x-axis is:

$$\dot{v}_{bx} = \frac{1}{m}\left(\sum_{i=1}^{2} F_{i,tract} + \sum_{i=1}^{4} F_{i,roll\,res} + F_{disturb} - mg\sin(\varphi)\right) = f_{tire} + a_{dist,bx} - g\sin(\varphi) \quad (1)$$

where m is the total vehicle mass, g is the acceleration due to gravity, f_{tire} and $a_{dist,bx}$ are the equivalent x-axis body accelerations due to tire forces and the disturbance force. Assuming the vehicle's axis of yaw rotation is approximately the point midway between the front tires and neglecting any yaw moment due to gravity, the vehicle's yaw acceleration is:

$$\ddot{\psi} = \frac{c}{2J}\left(F_{1,tract} + F_{1,roll\,res} - F_{2,tract} - F_{2,roll\,res}\right) = g_{tire} \quad (2)$$

where J is the vehicle's moment of inertia about the body z-axis and c is the distance between front wheel centers.

C. Normal Forces

Calculation of the robot's normal forces with arbitrary body roll (θ) and pitch (φ) is in general an underconstrained problem. Methods proposed in the literature [16], [17] typically consider a simplified 2-wheeled "bicycle" model, which can be applied when roll effects are ignored. In [18] it is suggested that normal forces be estimated by considering the elasticity of the terrain. A rigid body solution can also be found (utilizing the Moore-Penrose Generalized Inverse), assuming point tire-soil contact [19].

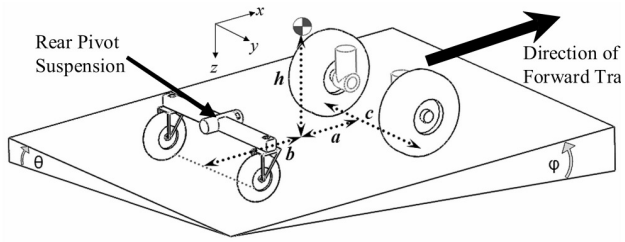

Fig. 1. Robot kinematic parameters and body-fixed coordinate system.

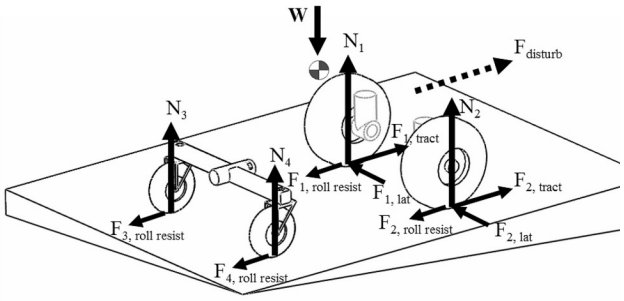

Fig. 2. Diagram showing vehicle and tire forces.

For the robot configuration considered in this work, assuming zero moment about the passive rear suspension pivot joint allows the rear left and right normal forces to be assumed equal. With this assumption the normal force calculation is no longer underconstrained and an explicit solution exists. For normal force calculations it is also assumed that the vehicle longitudinal acceleration is negligibly small, which is generally valid for slow-moving robots. The normal forces are:

$$N_1 = \frac{W}{2}\cos(\varphi)\left(\frac{1}{a+b}\left(b\cos(\theta) - h\tan(\varphi)\right) - \frac{2h}{c}\sin(\theta)\right) \quad (3)$$

$$N_2 = \frac{W}{2}\cos(\varphi)\left(\frac{1}{a+b}\left(b\cos(\theta) - h\tan(\varphi)\right) + \frac{2h}{c}\sin(\theta)\right) \quad (4)$$

$$N_3 = N_4 = \frac{W}{2(a+b)}\left(h\sin(\varphi) + a\cos(\varphi)\cos(\theta)\right). \quad (5)$$

D. Traction/Braking Model

A large body of research has been performed on modeling tire forces on rigid and deformable terrain. Most models are semi-empirical and express tire traction/braking forces as functions of wheel slip i and wheel skid i_s, where [20]:

$$i = 1 - \frac{v}{r\omega} \text{ and } i_s = 1 - \frac{r\omega}{v} \quad (6),(7)$$

where r is the tire radius and ω is the wheel angular velocity.

Here a unified, explicitly differentiable traction/braking model is proposed that captures the critical elements of the models proposed in the literature. The traction/braking force is expressed as a function of the wheel's relative velocity, rather than slip. A relative velocity-based formulation does not introduce the singularities found in slip-based formulations and is consequentially easier to apply within an

extended Kalman filter framework. The simplified model is:

$$F_{Traction} = N\left(\text{sign}(v_{rel})C_1\left(1-e^{-A_t|v_{rel}|}\right)+C_2 v_{rel}\right) \quad (8)$$

where v_{rel} is the velocity of the tire relative to the ground: $v_{rel} = r\omega - v_{fwd}$, and v_{fwd} is the tire's forward velocity, computed as: $v_{fwd} = v_{bx} \pm 0.5c\dot{\psi}$, where $\dot{\psi}$ is the robot yaw rate, and C_1, A_t, and C_2 are constants.

The simplified model is continuously differentiable and can predict both traction and braking force, without a need to distinguish the two cases. This model requires three terrain/tire dependant parameters as opposed to the popular "Magic Formula" which requires six. C_1 and A_t are positive constants which can be viewed as the maximum tire-terrain traction coefficient and the slope of the traction curve in the low relative velocity region, respectively. C_2 is the slope in the high relative velocity range and can be positive or negative depending on the terrain.

Fig. 3 shows a plot of traction/braking versus wheel slip for lines of constant wheel velocity using the proposed traction/braking model, as well as a representative traction force curve from the literature [20]. Assuming the robot typically operates near a nominal velocity, the proposed model can be interpreted as a pseudo-linearization around the nominal operating velocity.

E. Rolling Resistance Model

Rolling resistance is generally modeled as a combination of static and velocity dependant forces [20], [17]. Here a function with form similar to (8) is proposed as a continuously differentiable formulation of the rolling resistance, with the static force smoothed at zero velocity to avoid a singularity. The rolling resistance is:

$$F_{roll\,res} = -\text{sign}(v_{fwd})N\left(R_1\left(1-e^{-A_{roll}|v_{fwd}|}\right) + R_2|v_{fwd}|\right) \quad (9)$$

where R_1, A_{roll}, and R_2 are positive constants.

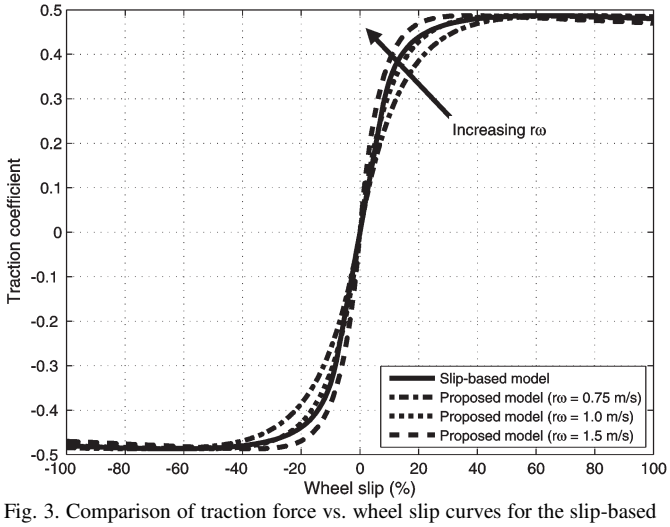

Fig. 3. Comparison of traction force vs. wheel slip curves for the slip-based model and the proposed simplified model at various wheel speeds.

II. Slip Detector Algorithm

A. Extended Kalman Filter

The slip detector algorithm utilizes an extended Kalman filter (EKF) to integrate sensor measurements with the nonlinear vehicle model. The EKF structure requires that the discrete, nonlinear process model be written in the form $\hat{x}_k^- = f(\hat{x}_{k-1}, u_k, w_{k-1})$, where $\hat{x}_k^-$ is the *a priori* estimate of the state vector, x, at time step k and f is a nonlinear function of the previous state estimate, $\hat{x}_{k-1}$, the current input vector, u_k, and process noise, w_{k-1}. The measurement vector, z, is a nonlinear function, h, of the true, current state vector and sensor noise ν such that: $z_k = h(x_k, \nu_k)$.

The standard EKF time update equations using the notation of [21] and Joseph's form of the covariance update equation [22] are used. The relations $f(\hat{x}_{k-1}, u_k, 0)$ and $h(\hat{x}_k^-, 0)$ express the estimated state and measurement vectors, $\hat{x}_k^-$ and $\hat{z}$, by evaluating the nonlinear process and measurement equations, assuming zero noise. Q and R are process and measurement noise covariance matrices and A_k, W_k, H_k, and V_k are process and measurement Jacobians.

B. State Space Model Formulation

The vehicle and sensor dynamics are formulated as a state space model using the following state vector:

$$\mathbf{x} = [x_1, x_2, x_3, x_4, x_5, x_6, x_7]^T = [v_{bx}, b_{ax}, a_{dist,bx}, \omega_l, \omega_r, \dot{\psi}, b_{g\psi}]^T,$$

where ω_l and ω_r are the angular velocities of the left and right front wheels and b_{ax} and $b_{g\psi}$ are the accelerometer x-axis and yaw gyro walking biases, respectively, which are part of the IMU error model suggested in [23].

Using the above state vector, the vehicle dynamics can be written as:

$$\dot{\mathbf{x}} = \begin{bmatrix} f_{tire}(\mathbf{x},\theta,\varphi) + x_3 - g\sin(\varphi) \\ -\frac{1}{\tau_{ax}}x_2 \\ 0 \\ f_{controller,l}(u) \\ f_{controller,r}(u) \\ g_{tire}(\mathbf{x},\theta,\varphi) \\ -\frac{1}{\tau_{gz}}x_7 \end{bmatrix} + \begin{bmatrix} w_1 \\ \sqrt{\frac{2f_s\sigma_{abx}^2}{\tau_{ax}}}w_2 \\ w_3 \\ w_4 \\ w_5 \\ w_6 \\ \sqrt{\frac{2f_s\sigma_{gbz}^2}{\tau_{gz}}}w_7 \end{bmatrix} \quad (10)$$

where w_i are zero mean white noise and $f_{controller}(u)$ is the wheel acceleration which is a function of the robot's onboard velocity controller and the desired velocity. When estimating the vehicle dynamics, we neglect $f_{controller}$, assuming it is negligible compared with the process noise. The *a priori* estimate of the robot state is found by discretizing the state equations and neglecting the zero-mean process noise, w_i.

C. Measurement Model

The algorithm utilizes measurements from the IMU, GPS, and front wheel encoders. The measurement vector is:

$$\mathbf{z} = [z_1, z_2, z_3, z_4, z_5]^T = [\ddot{x}_{IMU}, \dot{\psi}_{IMU}, \dot{x}_{GPS}, \omega_{l,enc}, \omega_{r,enc}]^T,$$

where $\ddot{x}_{IMU}$ and $\dot{\psi}_{IMU}$ are IMU measurements of x-axis acceleration and yaw rate, $\dot{x}_{GPS}$ is the component of the GPS velocity measurement along the body x-axis, and $\omega_{l,enc}$ and $\omega_{r,enc}$ are the left and right front wheel encoder angular

velocities. The measurement vector can be written as:

$$\mathbf{z} = h(\mathbf{x},\nu) = \begin{bmatrix} f_{tire}(\mathbf{x},\theta,\varphi) + c_{ax} + x_2 + x_3 \\ x_6 + c_{gz} + x_7 \\ x_1 \\ x_4 \\ x_5 \end{bmatrix} + \begin{bmatrix} \nu_1 \\ \nu_2 \\ \nu_3 \\ \nu_4 \\ \nu_5 \end{bmatrix} \quad (11)$$

where c_{ax} and c_{gz} are constant offsets of the x-axis accelerometer and yaw gyro respectively. To approximate the constant offsets, they are initialized to the average of the first n IMU measurements, subtracting out the acceleration due to gravity from the acceleration measurement. When the robot is at rest, the constant offsets are updated with new measurements using the exponential moving average [24]:

$$EMA_{current} = (measurement_{current} - EMA_{prev})\left(\frac{2}{1+p}\right) + EMA_{prev} \quad (12)$$

which is an approximation of the time average of the measurement over the last p samples, with a higher weight given to the most recent measurements. The EMA is not guaranteed to converge to the true value of the constant offsets, however it is easily and recursively calculated and has yielded good results in practice. The estimated measurement vector, $\hat{\mathbf{z}}$, is found by evaluating $h(\hat{\mathbf{x}}_k^-, 0)$.

D. Weak Constraints

The disturbance, x_3, and accelerometer walking bias, x_2, have both been modeled as random walks. Practically, the only difference between these variables in the model are that x_3 appears in the calculation of $\dot{x}_1$ while x_2 does not, and that x_3 is assigned a larger covariance in the matrix Q so that x_3 can evolve more quickly than x_2.

Although a direct measure of the disturbance force is generally not available, rules governing its evolution can be developed based upon insight into the physical nature of the disturbance. These rules are implemented using weak constraints described in [25] and implemented in a vehicle model in [11]. Unlike ad hoc solutions, weak constraints are a principled method for integrating rules and constraints into the Kalman filter framework which correctly update the state vector and system covariance matrix. Weak constraints can be viewed as virtual measurements or observations.

Table I summarizes the weak constraints employed in this work. The middle column presents the condition that must be met for the constraint to be applied and the right column gives the measurement innovation to be used in the EKF. For some of the conditions the variable *VelDir* is used, defined as: $VelDir = \text{sign}(\omega_l + \omega_r)$, such that *VelDir* equals 1 if the wheels are driving forward, 0 if the wheels are stopped, and -1 if the wheels are driving in reverse. i_{EMA} is the EMA (12) of the average of the left and right front wheel slip.

E. Slip and Immobilization Detection

The extended Kalman filter provides an estimate of the robot's forward velocity and the front wheels' angular velocities. Using these estimates, a criterion for detecting when the robot is immobilized is desired. A natural choice

TABLE I. SUMMARY OF WEAK CONSTRAINTS USED.

Description	Condition	"Measurement" Innovation
1) The modeled disturbance force should only oppose motion.	if $(\text{sign}(\hat{x}_{k,3}^-) = VelDir)$ AND $(\text{sign}(\hat{x}_{k,3}^-) \neq 0)$	$(z_{k,WC1} - h_{WC1}(\hat{x}_k^-, 0)) = (0 - \hat{x}_{k,3}^-)$
2) The disturbance should act quickly. The disturbance should not gradually increase such that the wheel EMA of the slip slowly increases. Only applies when the average wheel slip is small. $min_{\Delta i}$, $thresh_i$, and α are user-defined constants	if $\left(0 < \frac{\Delta i_{EMA}}{\Delta t} < min_{\Delta i}\right)$ AND $\left(i_{EMA} < thresh_i\right)$	$(z_{k,WC2} - h_{WC2}(\hat{x}_k^-, 0)) = (\alpha\, \hat{x}_{k,3}^- - \hat{x}_{k,3}^-)$ $0 < \alpha < 1$
3) The disturbance can stop the robot, but should not pull the robot backwards. If the robot is moving backwards, then either it is sliding down a hill and the disturbance should be zero, or the estimated disturbance is too high and should be reduced.	if $(\text{sign}(\hat{x}_{k,1}^-) = -VelDir)$ $\Rightarrow$ if $\text{sign}(f_{tire} - g\sin(\varphi)) = -VelDir$ then $\rightarrow$ a) else $\rightarrow$ b)	$(z_{k,WC3} - h_{WC3}(\hat{x}_k^-, 0)) =$ a) $(0 - \hat{x}_{k,3}^-)$ b) $(\max(VelDir[g\sin(\varphi) - f_{tire},\ \hat{x}_{k,3}^-]) - \hat{x}_{k,3}^-)$
4) When robot is fully stopped, the disturbance force and walking biases should tend to zero for calibration of the IMU constant biases. t_{stop} is a constant. T is the length of time the condition has been met.	if $(\omega_l = 0)$ AND $(\omega_r = 0)$ for $T \geq t_{stop}$	$(z_{k,WC4} - h_{WC4}(\hat{x}_k^-, 0)) =$ $([0,0,0]^T - [\hat{x}_{k,2}^-, \hat{x}_{k,3}^-, \hat{x}_{k,7}^-]^T)$
5) In practice there may be known limits on the magnitude of a state. These can be implemented as weak constraints as a precaution to keep the states bounded. Under normal operation, these constraints will rarely be called. max_{xi} are constants.	if $(\text{abs}(\hat{x}_{k,i}^-) > max_{x_i})$ In practice i = 1, 2, 3, and 7 have been used.	$(z_{k,WC5} - h_{WC5}(\hat{x}_k^-, 0)) =$ $(\text{sign}(\hat{x}_{k,i}^-)\, max_{xi} - \hat{x}_{k,i}^-)$
6) Similar to 5, we can also impose limits on the rate of change of a state. For example, we may know that a robot's acceleration will never exceed a certain limit.	if $\left(\frac{\text{abs}(\hat{x}_{k,i}^- - \hat{x}_{k-1,i}^-)}{\Delta t}\right) > max_{\Delta i}$ In practice i = 1 and 2 have been used.	$(z_{k,WC6} - h_{WC6}(\hat{x}_k^-, 0)) =$ $((\hat{x}_{k-1,i}^- + \text{sign}(\hat{x}_{k,i}^- - \hat{x}_{k-1,i}^-) max_{\Delta i}\Delta t) - \hat{x}_{k,i}^-)$

for an "immobilized" metric is the wheel slip (6). In practice, the calculated wheel slip can be noisy. To improve robustness, the EMA (12) of the average of the left and right wheel slips is calculated and immobilization is detected if the EMA is larger than a threshold value. The threshold value is chosen empirically. A low threshold value allows the detector to react quickly, however can be prone to falsely detecting immobilized conditions. In practice, since measurement noise can cause large variations in calculated slip at low speeds, the threshold can be chosen to vary with speed. Immobilization is not detected if the robot is braking (i.e. $v_{bx} > r\omega$). The above technique represents one possible criterion for detecting immobilization which has worked well in practice; however other criteria are possible.

III. Experimental Results

A. Robot Description

An autonomous mobile robot developed for the DARPA LAGR (Learning Applied to Ground Robots) program [26] has been used to experimentally validate the algorithm (Fig. 4). The robot is 1.2 m long x 0.7 m wide x 0.5 m and has the kinematic configuration discussed in Section IIA. The robot is equipped with 4096 count per revolution front wheel encoders, an Xsens MT9 IMU, and a Garmin GPS 16 differential GPS. The robot has been used to collect data to process offline using a Matlab implementation of the slip detector, as well as to run an online C++ implementation on one of the robot's 2.0 GHz Pentium M computers.

Fig. 4. The LAGR robot.

B. Algorithm Performance

The algorithm was applied to 21 experimental test runs. During these tests, the robot traveled approximately 120 meters over a range of terrain types including loose mulch, loose gravel over hard dry soil, mud, and various grasses. The robot was driven at speeds ranging from 0.1 m/s to 1 m/s. The test runs include 20 instances of the robot coming to a complete stop with the wheels still spinning, which were initiated by holding the robot back using a spring scale.

The slip detector correctly identified each of these 20 instances as immobilized with an average detection time of 0.4 seconds. All data with the robot driving freely or sitting at rest was correctly labeled as normal driving, with the exception of two false positives. In total, less than 0.2% of the data points were falsely labeled as immobilized.

Fig. 5 shows a plot of the robot driving unconstrained over grass at 1 m/s until $t \sim 16$ s, after which the robot attempts to drive forward while restrained with a spring scale, producing 100% wheel slip. In the top plot, it can be seen that the filter's estimated robot velocity (x_1) follows the measured wheel velocity during the unconstrained driving. The middle plot shows the estimated disturbance (x_3), which remains small while the robot is driving normally. Just after the robot stops at $t \sim 14.5$ s, suspension displacement creates a small spike in the disturbance. The bottom plot shows the EMA of the wheel slip. While driving normally, the wheel slip is estimated at approximately 3%, which is physically reasonable. The increased slip while accelerating and braking is also expected. The detector correctly labeled the unconstrained driving as driving normally.

After $t \sim 16$ s the velocity estimate shows that the robot accelerates against the spring, but quickly becomes immobilized. The disturbance estimate approaches a near-constant resistive value ranging from -2.8 to -3.1 $m \cdot s^{-2}$ while the robot is immobilized, before returning to zero when the wheels stop spinning. During this test, the spring scale measured a 325 N force holding the robot back. The equivalent body acceleration for the 117 kg robot is 2.8 $m \cdot s^{-2}$, which closely agrees with the estimated disturbance. The slip EMA quickly approaches 100% and the detector identifies the robot as immobilized at $t = 17.56$ s.

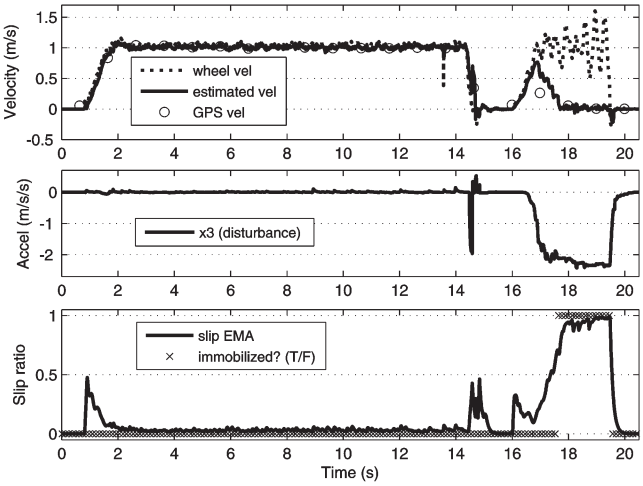

Fig. 5. Example of robot driving normally, then becoming immobilized.

GPS velocity estimates for this test were available at 1 Hz, slower than desired for detection. Additionally, GPS returns the average velocity over the previous time step, and thus the measurement is truly accurate for 0.5 seconds prior to the reported measurement time. In this example, immobilization could not be detected by GPS until $t \sim 18.5$ s, nearly one second slower than the proposed algorithm.

C. Algorithm Performance without GPS

The 21 experimental datasets were reprocessed without including GPS velocity measurements (i.e. using wheel encoder and IMU measurements only). The algorithm again

TABLE II. SENSITIVITY OF FALSE IMMOBILIZED FLAGS TO CHANGES IN TIRE PARAMETERS.

Parameter		C_1		A_1		R_1		R_2		A_{roll}	
Parameter Change	Nominal	+20%	-20%	+20%	-20%	+20%	-20%	+20%	-20%	+20%	-20%
# False Positives	2	1	5	2	3	2	2	2	2	2	2

correctly identified all 20 immobilizations. The false immobilization detections increased from two to four (0.35% of all data points). These results suggest the algorithm can be applied on systems lacking reliable GPS, such as mobile robots in urban surroundings, underwater, or where GPS is not available such as for Mars rovers. However, GPS can increase accuracy and improve performance when available.

D. Sensitivity to Tire Model Parameters

To study the algorithm's sensitivity to tire model parameter values, the 21 experimental data sets were reprocessed, individually varying one of the five tire constants by ±20%. In all 210 tests, the algorithm correctly identified all 20 immobilizations. The number of false positives for each case is summarized in Table II. It was observed that the algorithm performance was most sensitive to changes in C_1. Increasing C_1 increases the maximum modeled traction, making the model less likely to estimate that traction has been lost and the wheels are slipping. Conversely, decreasing C_1 reduces the modeled available traction, increasing the likelihood of wheel slip in the model and causing an increase in the number of false immobilization detections. Even in the worst case, only 0.3% of the data points were falsely labeled immobilized. In summary, the algorithm appears quite robust to errors in estimated tire model parameters. It should be noted that the algorithm's velocity estimate appeared less accurate in the majority of the non-nominal cases.

IV. CONCLUSIONS AND SUGGESTIONS FOR FUTURE WORK

A dynamic model-based slip detector has been proposed that has proven effective at detecting robot immobilization over a variety of terrains. The detector utilizes a novel tire traction/braking model and weak constraints to estimate external forces acting on the robot. The algorithm can be applied to any vehicle with an IMU, wheel encoders, and (optionally) GPS. Sensitivity analysis has indicated that accurate immobilization detection is possible with relatively coarse engineering estimates of tire-model parameters. The algorithm also yields reasonably accurate estimates of the robot's velocity and could potentially be implemented in a position estimation system that is robust to wheel slip.

Current work focuses on techniques for autonomously adapting the tire model parameters which would allow the algorithm to provide highly accurate velocity estimates as well as improve the slip detection time and reliability over variable terrain. Current work is also exploring fusing the output of multiple slip detection algorithms to increase detection speed and accuracy.

ACKNOWLEDGMENT

The authors would like to thank Andrew Lookingbill at Stanford University for his help collecting experimental data.

REFERENCES

[1] J. Borenstein, H. R. Everett, L. Feng, *"Where am I?" Sensors and Methods for Mobile Robot Positioning*, Univ. of Michigan, April 1996, http://www-personal.umich.edu/~johannb/shared/pos96rep.pdf.

[2] Y Fuke, E. Krotkov, "Dead reckoning for a lunar rover on uneven terrain," *Proc. of 1996 IEEE Intl. Conf. on Robotics and Automation*.

[3] F. Gustafsson, "Slip-based tire-road friction estimation," *Automatica*, vol. 33, no. 6, pp. 1087-1099, 1997.

[4] L. Ojeda, G. Reina, J. Borenstein, "Experimental results from FLEXnav: An expert rule-based dead-reckoning system for Mars rovers," *IEEE Aerospace Conference 2004*, Big Sky, MT, 2004.

[5] L. Ojeda, D. Cruz, G. Reina, J. Borenstein, "Current-based slippage detection and odometry correction for mobile robots and planetary rovers," *IEEE Transactions on Robotics*, vol. 22, no. 2, April 2006.

[6] H. Tan, Y. Chin, "Vehicle antilock braking and traction control: a theoretical study," *Intl. Jrnl. of Systems Science*, vol. 23, no. 3, 1992.

[7] *Automotive Handbook*, 5th ed., Robert Bosch GmbH, Germany, 2000.

[8] R. Anderson, D Bevly, "Estimation of slip angles using a model based estimator and GPS," *Proc. of the 2004 American Control Conf.*, v. 3.

[9] A. Kelly, "A 3D state space formulation of a navigation Kalman filter for autonomous vehicles," CMU Tech. Report CMU-RI-TR-94-19.

[10] M. Wadda, K. Yoon, H. Hashimoto, "High accuracy road vehicle state estimation using extended Kalman filter," *Proc. of 2000 IEEE Intelligent Transportation Systems*, Dearborn, MI, October 2000.

[11] S. Julier, H. Durrant-Whyte. "On the role of process models in autonomous land vehicle navigation systems," *IEEE Trans. on Robotics and Automation*, vol. 19, no. 1, 2003.

[12] S. Sukkarieh, E. Nebot, H. Durrant-Whyte, "A high integrity IMU/GPS navigation loop for autonomous land vehicle applications," *IEEE Trans. on Robotics and Automation*, vol. 15, no. 3, June 1999.

[13] B. Hofmann-Wellenhof, H. Lichtenegger, J. Collins, *Global Positioning System: Theory and Practice*, 5th ed., Springer, 2001.

[14] *GPS 16/17 Series Technical Specifications*, Garmin Intl., Inc., 2005.

[15] C. Ward, K. Iagnemma, "Classification-based wheel slip detection and detector fusion for outdoor mobile robots," *2007 IEEE Int. Conf. on Robotics and Automation*.

[16] T. Gillespie, *Fundamentals of Vehicle Dynamics*. Society of Automotive Engineers, Warrendale, PA. 1992, ch. 1.

[17] M. G. Bekker, *Theory of Land Locomotion*, Univ. of Michigan Press, Ann Arbor, 1956, ch. 9 & 6.

[18] M. Hung, D. Orin, "Dynamic simulation of actively-coordinated wheeled vehicle systems on uneven terrain," *Proc. of the 2001 IEEE Intl. Conf. on Robotics and Automation*, Seoul, Korea, May 2001.

[19] V. Kumar, K. Waldron, "Force distribution in closed kinematic chains," *IEEE Jrnl. of Robotics and Automation*, vol. 4, no. 6, 1988.

[20] J. Y. Wong, *Theory of Ground Vehicles*, 3rd ed., Wiley, 2001, ch. 1.

[21] G. Welch, G. Bishop, "An introduction to the Kalman filter," *SIGGRAPH 2001*.

[22] M. Grewal, A. Andrews. *Kalman Filtering: Theory and Practice*, Englewood Cliffs, NJ: Prentice Hall, 1993, ch. 7.

[23] W. Flenniken, J. Wall, D. Bevly, "Characterization of various IMU error sources and the effect on navigation performance," *Proc. of ION GNSS 2005*.

[24] "Moving Averages", July 2006, http://www.stockcharts.com/education/IndicatorAnalysis/indic_movingAvg.html

[25] J. Geeter, H. Brussel, J. Schutter, "A Smoothly Constrained Kalman Filter," *IEEE Trans. on Pattern Analysis and Machine* Intelligence, vol. 19, no. 10, October 1997.

[26] "Learning Applied to Ground Robots," accessed July 24, 2006, http://www.darpa.mil/ipto/Programs/lagr/vision.htm.