

# Near-Optimal Navigation of High Speed Mobile Robots on Uneven Terrain

Karl Iagnemma, Shingo Shimoda, and Zvi Shiller

**Abstract**—This paper proposes a method for near-optimal navigation of high speed mobile robots on uneven terrain. The method relies on a layered control strategy. A high-level planning layer generates an optimal desired trajectory through uneven terrain. A low-level navigation layer guides a robot along the desired trajectory via a potential field-based control algorithm. The high-level planner is guaranteed to yield optimal trajectories but is computationally intensive. The low-level navigation layer is sub-optimal but computationally efficient. To guard against failures at the navigation layer, a model-based lookahead approach is employed that utilizes a reduced form of the optimal trajectory generation algorithm. Simulation results show that the proposed method can successfully navigate a mobile robot over uneven terrain while avoiding hazards. A comparison of the method’s performance to a similar algorithm is also presented.

## I. INTRODUCTION AND PREVIOUS WORK

UNMANNED ground vehicles (UGVs) are expected to play significant roles in future military, planetary exploration, and materials handling applications [1]. Many applications require UGVs to move at high speeds on rough, poorly characterized terrain. Ideally, UGVs would follow optimal (i.e. minimum time or maximum speed) trajectories during these operations to maximize efficiency, productivity, or other metrics.

Optimal performance is difficult to achieve in practice for several reasons. First, UGVs generally have access only to coarse-grained (i.e. several vehicle lengths spacing per data point) map data during the trajectory planning stage. Thus, while planned trajectories may be theoretically optimal at the data resolution available, they are likely to be sub-optimal, or even infeasible, at the data resolution relevant to the navigation task (i.e. several data points per vehicle length). Second, during high speed navigation, UGVs will likely encounter unexpected hazards that must be quickly (i.e. O(ms)) avoided. To avoid these hazards, navigation algorithms must be computationally efficient while considering important vehicle dynamic effects such as rollover and side slip.

Despite the wide interest in motion planning, few off-road trajectory planners have been developed [2]-[6]. Off-road trajectory planners cannot rely on a binary representation of

obstacles and free space, which is common to most work on motion planning. Instead, traversability over uneven terrain is determined not only by the size of obstacle, but also by terrain slope and curvature, and vehicle dynamics and speed. A kinematic planner was presented in [3] that computes the shortest feasible path for off-road vehicles. While the selected path is ensured to be statically safe, it does not account for vehicle dynamics and speed. A genetic algorithm is used in [4] to synthesize paths from segments, each evaluated for static stability and for satisfying mission constraints. Another genetic-based planner [6] uses fuzzy logic to account for obstacle height in each terrain region, which in turn determines vehicle speed. A similar approach to represent traversability is used in [7].

Off-road planners that explicitly consider vehicle dynamics typically search for an optimal path using dynamic simulations to determine the traversability or cost of specific terrain segments [5, 8]. An exception is the global trajectory planner first presented in [2]. It determines traversability directly by computing the maximum speed above which a vehicle will rollover or skid along a given terrain segment. The global search first selects a series of “best” traversable paths, which are then further optimized to minimize travel time. Early work derived speed limits for a simple point mass robot model [2]. Recent work considered more detailed vehicle models [9].

Artificial potential fields have long been successfully employed for robot navigation. First works were performed by Khatib as a real-time obstacle avoidance method for manipulators [10]. Ge et al. applied a potential field method for dynamic control of a mobile robot, with moving obstacles and goal [11]. Latombe applied potential field methods to general robot path planning [12]. Path planning using artificial potential fields has also been applied to nonholonomic systems [13]. Potential field navigation for wheeled robots on natural terrain has also been explored [14]. In general, potential field methods have been used for planning and control of low-speed systems, usually with a binary obstacle representation.

This paper presents a method for near-optimal navigation of high speed mobile robots on uneven terrain. The method relies on a layered control strategy. A high-level planning layer generates an optimal desired trajectory that is represented as a series of waypoints. The trajectory generation method is based on the global physics-based planner first presented in [2]. In the scenario considered here, this trajectory is formulated off-line based on coarse

Karl Iagnemma is with the Department of Mechanical Engineering, Massachusetts Institute of Technology, Cambridge MA, USA (e-mail: [kdi@mit.edu](mailto:kdi@mit.edu)). Shingo Shimoda is with the RIKEN BMC, JAPAN (e-mail: [shimoda@bmc.riken.jp](mailto:shimoda@bmc.riken.jp)). Zvi Shiller is with the Department of Mechanical Engineering-Mechatronics, College of Judea and Samaria, Ariel, Israel ([shiller@yosh.ac.il](mailto:shiller@yosh.ac.il)).

topographical map data, and thus computational constraints are minimal. A low-level navigation layer then guides the robot along the desired trajectory via a potential field-based control algorithm based on previous work by the authors [15]. In this method, a potential field is defined in the two-dimensional “trajectory space” of the robot path curvature and longitudinal velocity based on fine-grained elevation data gathered from on-board sensors [16]. Previous work required ad hoc tuning of potential field gains to yield safe robot navigation in practice. Here, to guard against failure a model-based lookahead approach is employed that utilizes a reduced form of the optimal trajectory generation algorithm. Simulation results show that the proposed method can successfully navigate a mobile robot over uneven terrain while avoiding hazards. A comparison of the method’s performance to a similar algorithm is also presented.

## II. OPTIMAL TRAJECTORY PLANNING

Motion planning over rough terrain requires the selection of a feasible path, and the computation of some velocity profile along that path. If optimal performance is not required, the path can be generated first to ensure that the UGV is statically stable using a kinematic planner, such as in [3]. Then, the velocity profile along that path can be computed to ensure that the vehicle is also dynamically stable. While computationally efficient, such a path-velocity decomposition cannot yield the time optimal trajectory since the kinematic search is not guided by the vehicle’s dynamic performance. For this reason, the search for the optimal trajectory is normally done in the vehicle’s  $2n$  dimensional state-space, for an  $n$  dimensional configuration space.

To avoid an expensive search in the state-space, the global trajectory optimization used here is formulated as a two stage optimization that combines a global “kinematic” graph search over the terrain with a local trajectory optimization [2]. The global search selects promising candidate paths for local optimization, thus trading an expensive search in the state-space for many simpler searches in the configuration space. A byproduct of this approach is the generation of local minima in addition to the global optimal trajectory [2, 17].

### A. Terrain Representation

Terrain is here represented by a smooth bi-cubic B patch, which is a parametric surface made of a mesh of cubic splines. The need for a smooth surface representation stems from the local optimization, which requires a smooth path over the terrain surface.

Topologically, the patch is a warped rectangle in a three dimensional space. This terrain representation does not distinguish between obstacles and uneven terrain. Obstacles are simply integrated into the B-patch. The control points of the patch are generated by placing a uniform grid on the map-range data. The resolution of this grid depends on the map range. For short-range planning (up to 100 m), the resolution is chosen empirically at half the UGV width. This

ensures that obstacles the size of the vehicle and larger are depicted by the B-patch. For long range planning (up to several kilometers), the resolution is selected at 100 m between control points to account for primary terrain features (i.e. hills, ravines, etc.).

### B. Velocity Limits

The global search produces path candidates for local trajectory optimization. Since we seek the time optimal trajectory, the global search selects paths along which the UGV can sustain high speeds without violating dynamic constraints such as rollover, excessive side slip, and maintaining ground contact. Velocity limits (above which some of the dynamic constraints may be violated) are computed by mapping the dynamic constraints to constraints on the vehicle’s speed and tangential acceleration. For long range planning, the UGV is modeled as a suspended point mass [2, 18]. For short range planning, the vehicle is modeled as a rigid body [9].

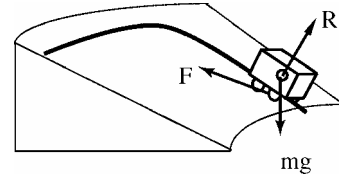


Fig. 1: External forces acting on a point mass UGV model.

Velocity limits for a point mass model are derived by first expressing the three external forces shown in Fig. 1 in terms of the UGV’s speed and tangential acceleration (see [2,18] for a detailed derivation):

$$\begin{aligned} f_t &= mgk_t + m\ddot{s} \\ f_q &= mgk_q + m\kappa n_q \dot{s}^2 \\ R &= mgk_r + m\kappa n_r \dot{s}^2 \end{aligned} \quad (1)$$

where  $f_t$  and  $f_q$  are components of the friction force tangent and normal to the path,  $\kappa$  is the path curvature,  $k$  is a unit vector pointing opposite of the gravity force,  $n$  is a unit vector pointing in the direction of the path center of curvature, and the subscripts denote projections along the path coordinate frame,  $t, q, r$ .

We can now express the dynamic constraints in terms of the external forces (1). The sliding constraint becomes:

$$f_t^2 + f_q^2 \leq \mu^2 R^2 \quad (2)$$

The tip-over constraint becomes:

$$f_q^2 \leq \left(R \frac{b}{h}\right)^2 \quad (3)$$

where  $h$  is the height of the center of mass and  $b$  is the lateral distance between the wheels. The contact constraint is simply:

$$R > 0 \quad (4)$$

Substituting (1) into the dynamic constraints (2-4) yields three constraints on vehicle speed. For example, the sliding constraint yields the following velocity constraint [18]:

$$a\dot{s}^4 + 2b\dot{s}^2 + c \geq 0 \quad (5)$$

where the coefficients  $a$ ,  $b$ ,  $c$  are determined from terrain geometry, path direction and curvature, and the terrain tractive coefficient. Staying below the speed limits obtained from the three dynamic constraints ensures that the vehicle does not roll, skid, or lose contact with the terrain.

### C. The Global Search

The maximum velocity  $\dot{s}_m$ , above which the vehicle cannot follow the given path is an excellent candidate for measuring traversability since it accounts for the effects of vehicle dynamics, terrain topography, and surface friction. A zero value implies that the vehicle is statically unstable, and the given path segment is hence not traversable, whereas a nonzero value implies that the given path segment is traversable at some nonzero speed. The velocity limit thus offers a *scalar* function that can distinguish between traversable and non-traversable path segments in the configuration space.

Dividing the path arc length by the velocity limit produces a simple cost function that has units of time and can be computed for each path segment along the graph used to represent the terrain:

$$J = \int_{s_1}^{s_2} \frac{ds}{\dot{s}_m} \quad (6)$$

Note that (6) resembles the cost function used to minimize time except that here the actual vehicle speed  $\dot{s}$  is replaced with the speed limit  $\dot{s}_m$ . Thus, using cost function (6) produces the fastest traversable paths, assuming that the vehicle travels at its maximum safe speed. The ability to select a traversable path at high speeds without the need for an expensive search in the state-space greatly contributes to the efficiency of this approach.

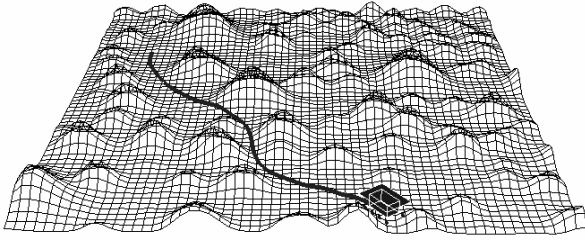


Fig. 2: A global optimal path over rough terrain.

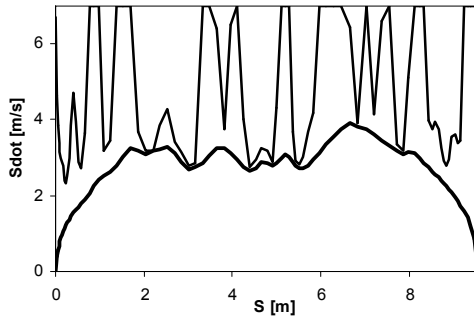


Fig. 3: An optimal velocity profile along the path in Fig. 2. The velocity profile stays below the velocity limits.

The global search first generates a series of “best” traversable paths. These paths are good initial guesses for a local trajectory optimization that will minimize motion time and take into account the feasible vehicle acceleration along the path (ignored during the global grid search). The local optimization consists of a parameter optimization over the control points of a B spline that define the path over the B-patch. The end result of this two stage process is a globally optimal trajectory that minimizes motion time from start to goal, while considering vehicle dynamics, terrain topography, and the vehicle’s dynamic constraints. Figs. 2 and 3 show an optimal path and optimal velocity profile computed by the global search between given end points on uneven terrain. The optimal velocity profile, shown in Fig. 3, is computed by switching between the maximum and minimum tangential acceleration to avoid crossing the velocity limit curve. Obviously, the optimal velocity profile represents the ultimate vehicle speeds along the path since any attempt to move faster would cross the velocity limits, which in turn would cause the vehicle to either roll, skid, or lose contact with the terrain.

### III. TRAJECTORY SPACE NAVIGATION WITH POTENTIAL FIELDS

Here, a potential field-based navigation method is employed as the low-level navigation layer. It takes as an input an optimal trajectory (computed as described in Section II) represented as a list of closely-spaced waypoints. A more thorough presentation can be found in [15].

The trajectory space,  $TS \in \mathcal{R}^2$ , is defined as a two-dimensional space of a UGV’s instantaneous path curvature and longitudinal velocity [16]. This space clearly cannot describe the complete vehicle state, but can rather capture important UGV state and configuration information and serve as a physically intuitive description of the current vehicle status. A UGV’s “position” in  $TS$  is a curvature-velocity pair denoted  $\tau = (v, k)$ . Note that in this work only positive longitudinal velocities are considered.

The trajectory space is a convenient space for navigation for two reasons. First, the trajectory space maps easily to the UGV actuation space (generally consisting of the throttle and steering angle). Navigation algorithms performed in the trajectory space will select command inputs that obey vehicle nonholonomic constraints. Second, dynamic constraints related to UGV rollover and side slip are easily expressible in the trajectory space, since these constraints are functions of the UGV velocity and path curvature. These constraints can also capture effects such as terrain inclination and roughness.

The coordinate systems used in this work are shown in Fig. 4. A body frame  $\mathcal{B}$  is fixed to the vehicle, with its origin at the vehicle center of mass. The position of the vehicle in the inertial frame  $\mathcal{I}$  is expressed as the position of the origin of  $\mathcal{B}$ . The vehicle attitude is expressed by  $x$ - $y$ - $z$  Euler angles using the vehicle yaw  $\theta$ , roll  $\phi$ , and pitch  $\psi$  defined in  $\mathcal{B}$ .

To perform navigation a potential field is constructed in the trajectory space based on dynamic constraints, waypoint

locations, and hazard locations. These issues are discussed below.

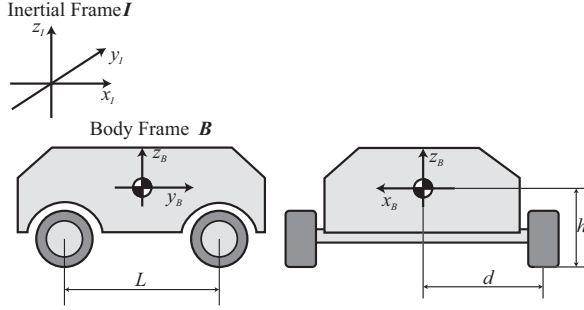


Fig. 4. Definition of coordinate system.

#### A. Potential Field Constraint Definitions

Potential fields are defined based on UGV rollover and side slip constraints. As in the high-level planning layer, these constraints are computed from rigid body models. A rollover constraint for a UGV on uneven terrain can be expressed as:

$$\kappa_r(v) = \frac{dg_z \pm hg_x}{hv^2} \quad (7)$$

where  $\kappa_r$  is the maximum admissible path curvature and  $g_*$  is the gravitational acceleration of the  $z$ -axis direction in  $B$ . The two solutions to (7) correspond to travel on positive/negative inclination slopes with nonzero  $g_x$  components reflecting the effect of terrain roll. A potential field is then defined as:

$$PF_r(v, \kappa) = \begin{cases} K_r \left( 1 - \frac{(\kappa - \kappa_{MAX})^2}{(\kappa_r(v) - \kappa_{MAX})^2} \right) & \kappa_r \leq \kappa < \kappa_{MAX} \\ 0 & 0 \leq \kappa < \kappa_r \end{cases} \quad (8)$$

where  $\kappa_{MAX}$  is the maximum attainable path curvature based on kinematic steering constraints, and  $K_r$  is a positive gain parameter to modulate the potential field amplitude.

A side slip constraint for a UGV on uneven terrain can be expressed as:

$$\kappa_s(v) = \frac{-g_x \pm \mu g_z}{v^2} \quad (9)$$

where  $\kappa_s$  is the maximum admissible path curvature and  $\mu$  is the terrain tractive coefficient. A potential field is then defined as:

$$PF_s(v, \kappa) = \begin{cases} K_s \left( 1 - \frac{(\kappa - \kappa_{MAX})^2}{(\kappa_s(v) - \kappa_{MAX})^2} \right) & \kappa_s \leq \kappa < \kappa_{MAX} \\ 0 & 0 \leq \kappa < \kappa_s \end{cases} \quad (10)$$

Again,  $K_s$  is a positive gain parameter to modulate the potential field amplitude.

#### B. Potential Field-Based Waypoint Navigation

For navigation between waypoints, a desired path curvature and velocity must be computed at each instant based (at minimum) on the relative location of the robot and the waypoint. A method for computing a path curvature based on knowledge of UGV steering kinematics and waypoint position is presented in [15]. A potential field corresponding to the current desired waypoint location can then be defined as follows:

$$PF_g(\kappa) = K_g(\kappa - \kappa_d)^2 \quad (11)$$

where  $\kappa_d$  is the desired steering angle and  $K_g$  is a positive gain parameter to modulate the potential field amplitude.

#### C. Potential Field for Desired Velocity

A potential field for the desired UGV velocity can be simply expressed as follows:

$$PF_v(v) = K_{v1}(v - v_d)^{K_{v2}} \quad (12)$$

where  $v_d$  is the desired velocity and  $K_{v1}$  and  $K_{v2}$  are positive gain parameters to modulate the potential field amplitude.  $v_d$  can be a function of position to reflect mission objectives.

#### D. Potential Field for Hazard Locations

Consider a UGV approaching a hazard as shown in Fig. 5. Here  $\kappa_1$  and  $\kappa_2$  are the maximum and minimum path curvatures that intersect the hazard. A potential field for hazard locations is constructed based on the following observations:

- Path curvatures between  $\kappa_1$  and  $\kappa_2$  can be safely followed until the UGV is near the hazard;
- The potential field magnitude should be greater at high speed than at low speed since control accuracy generally decreases with increasing speed;
- Relative locations of waypoints and hazards should influence the hazard potential field value (i.e. to allow close passage to hazards to achieve a waypoint).

From these observations, a potential field for hazard locations is defined as follows:

$$PF_o(v, \kappa) = \frac{K_o(K_{ov}v + 1)}{(K_{od}O_d + 1)(K_{oa}A_d + 1)} \exp\left(-\frac{(\kappa - X)^2}{2\sigma}\right) \quad (13)$$

where  $O_d$  is the Euclidean distance between vehicle and hazard,  $A_d$  is the angle between the UGV heading and the waypoint location,  $X = (\kappa_1 + \kappa_2)/2$ ,  $\sigma = (\kappa_1 - \kappa_2)/2$ , and  $K_o$ ,  $K_{od}$ ,  $K_{oa}$ , and  $K_{ov}$  are positive gain parameters to modulate the potential field amplitude.

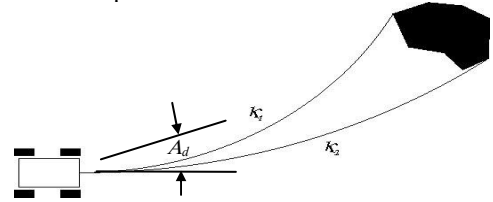


Fig. 5. Minimum and maximum path curvatures intersecting with a hazard.

#### E. Definition of Net Potential Field

A net potential field is generated as the sum of all proposed potential fields, as follows:

$$PF(v, \kappa) = PF_r(v, \kappa) + PF_s(v, \kappa) + PF_g(\kappa) + PF_v(v) + \sum_{i=1}^n PF_o^i(v, \kappa) \quad (14)$$

where  $n$  is the number of hazards present and  $PF_o^i$  is the potential function corresponding to the  $i^{th}$  hazard. An illustration of a net potential field is shown in Fig. 6.

At every timestep, a desired path curvature and velocity are determined by calculating the gradient of the net potential field at the robot position in  $TS$ , then moving in the direction of maximum descent. Details related to these computations

and issues related to potential field local minima and maxima are described in [15].

The potential field navigation method described here has been shown to perform well in simulation and experiments. However, studies have shown that terrain with high-frequency undulation can potentially lead to rollover and side slip failures. This is due to the fact that the potential field constraints related to rollover and side slip are computed as a function of the average local terrain inclination. Thus, large localized values of inclination can lead to failure.

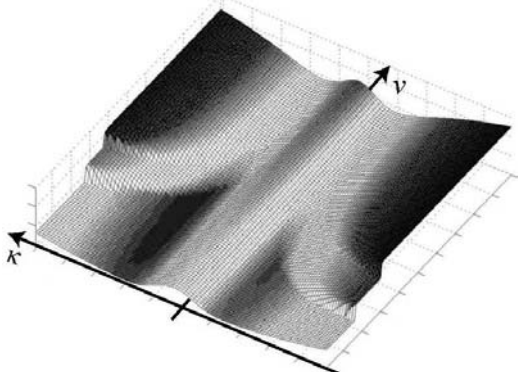


Fig. 6. Illustration of net potential field.

To guard against failures at the navigation layer, a model-based lookahead approach has been developed that utilizes a reduced form of an optimal trajectory generation algorithm. This approach is described next.

#### IV. NEAR-OPTIMAL POTENTIAL FIELD NAVIGATION

A common weakness of potential field methods is that it is difficult to guarantee convergence and bound system performance. While the potential field-based navigation method described here has been shown to perform effectively in simulation and experimental studies, the authors have observed failures in cases where significant high-frequency terrain undulation exists. This is due to the fact that potential functions related to rollover and side slip constraints are computed as a function of the average local terrain inclination. Thus, large localized values of inclination can lead to failure.

To guard against failure at the navigation layer a model-based lookahead approach is employed that utilizes a reduced form of the optimal trajectory generation algorithm. At every timestep, this approach essentially forward simulates a robot model over a short time horizon, then determines whether the robot will violate any dynamic constraints along the resulting trajectory. Violation of dynamic constraints is determined by comparing the robot's predicted velocity profile to the maximum safe allowable velocity profile computed via the method described in Section II. If a constraint is violated, the robot's desired velocity is reduced to the maximum safe velocity along the trajectory. This has the effect of imposing a "safe speed limit" on the robot at the navigation layer.

The algorithm for calculating a safe robot maneuver at every timestep is described here, with  $T$  representing the time horizon duration and  $t$  the "virtual time" in a forward

simulation loop. The value of  $T$  is here chosen empirically. The algorithm is composed of the following steps:

1. The value of the net potential field at the robot's current position in  $TS$  is calculated from Eqn. (14);
2. The gradient of the net potential field is computed, and a desired maneuver (i.e. a  $(\nu, \kappa)$  pair) is chosen in the direction of maximum descent;
3. The predicted trajectory of the robot is computed via forward simulation of a rigid body model subject to the desired maneuver over time  $dt$ ;
4. Steps 1-3 are repeated while  $t < T$ ;
5. A maximum safe velocity profile is computed over the predicted path via the approach described in Section II;
6. The predicted robot velocity profile is compared to the maximum safe velocity profile. If the predicted velocity profile exceeds the maximum safe velocity profile at any point, the robot's desired velocity is reduced to the maximum safe velocity along the trajectory.

This approach attempts to exploit the computational efficiency of the potential field-based navigation with the safety guarantees implicit in optimal trajectory planning. Since low-order rigid body models are used in forward simulation, computational demands are negligible.

#### V. SIMULATION RESULTS

Dynamic simulations were conducted of a small UGV traveling at high speeds over uneven terrain. The UGV parameters were as follows (see Fig. 4): length  $L = 0.27$  m, half-width  $d = 0.124$  m, c.g. height  $h = 0.06$  m, wheel diameter = 0.12 m. The potential field parameters were set as follows:  $K_r = 800$ ,  $K_s = 800$ ,  $K_g = 0.3$ ,  $K_{\nu l} = 0.5 \times 10^{-5}$ ,  $K_{\nu 2} = 4$ ,  $K_o = 1500$ ,  $K_{od} = 0.05$ ,  $K_{oa} = 10$ ,  $K_{ov} = 0.07$ ,  $T = 1.0$  s.

Randomized rough terrain was generated using a fractal method modified to incorporate gross terrain undulation and discrete "peaks." Rough terrain with fractal number of 2.05, grid spacing of 2 wheel diameters, and height scaling of 35 wheel diameters was employed. The near-optimal navigation method described above was used to determine the desired UGV steering angle and velocity. PD control was employed for steering angle and velocity control.

A representative simulation result is shown in Figs. 7-9. Fig. 7 shows a representative terrain, the optimal waypoints generated by the high-level planner, and the actual path followed by the low-level navigation layer. It can be seen that the UGV successfully navigates through the desired waypoints, thus approximating an optimal path while avoiding hazards. Fig. 8 shows a plot of the UGV velocity along the path compared to the maximum achievable velocity as computed by the model-based lookahead algorithm. The actual velocity remains near to the optimal velocity, and deviates from the desired velocity during hazard avoidance maneuvers and turns with large curvature. Fig. 9 shows plots of the UGV roll angle and slip angle during the simulation, illustrating the challenging nature of the terrain and effectiveness of the algorithm in limiting these quantities.

Table 1 shows the results of 25 simulation trials. The proposed near-optimal navigation method safely controlled

the UGV in all terrains. The average percent difference in total navigation time between the optimal method and the proposed method was 9.6%. It was observed that the “standard” potential-field based navigation scheme (i.e. without lookahead analysis) failed on 24% of the simulation trials, with a 23.6% average slower navigation time than the optimal traveling time. It should be noted that safer navigation with the “standard” approach is achievable, however at the expense of greater increase in navigation time.

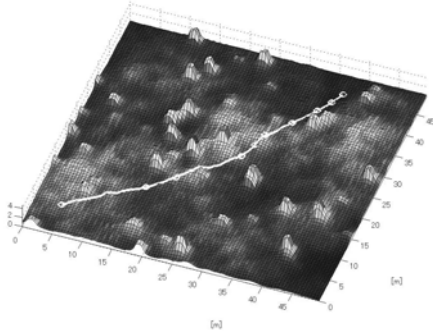


Fig. 7. Representative terrain map and waypoints generated by high-level planning layer and path generated by low-level navigation layer.

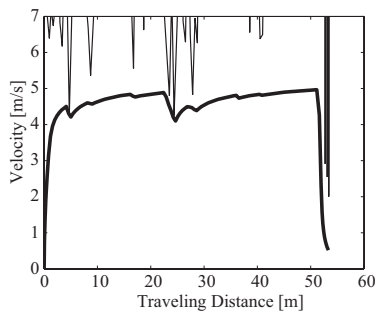


Fig. 8. UGV velocity (thick) and computed velocity limit curve (thin).

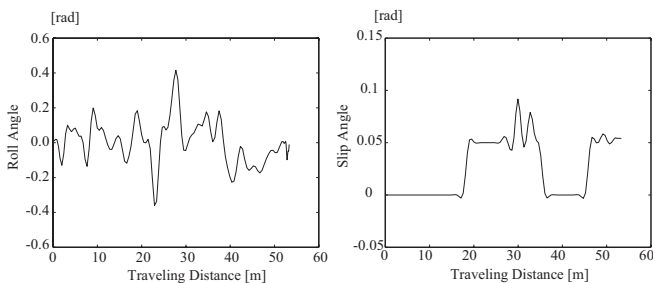


Fig. 9. UGV Roll angle (left) and slip angle (right).

TABLE I  
SIMULATION RESULTS OF 25 TERRAIN TRAVERSALS

	Incidence of Failure [%]	Increase over Optimal Time [%]
Near-optimal	0.0	9.6
“Standard” Potential-field	24.0	23.6

## VI. CONCLUSIONS

This paper proposes a method for near-optimal navigation of high speed mobile robots on uneven terrain. A high-level planning layer generates an optimal desired trajectory, and a low-level navigation layer guides a robot along the desired trajectory via a potential field-based algorithm. To guard against failures at the navigation layer, a model-based

lookahead approach was presented that utilizes a reduced form of the optimal trajectory generation algorithm. Simulation trials show that the proposed method can safely navigate a mobile robot along a near-optimal trajectory over uneven terrain while avoiding hazards.

## ACKNOWLEDGMENTS

K. Iagnemma acknowledges the support of the U.S. Army Research Office grant number W911NF-05-1-0166. Z. Shiller acknowledges the support of the Israeli Space Agency grant 01-99-08430 and the support of Dr. David Paslin for the Robotics Research Laboratory at the College of Judea and Samaria.

## REFERENCES

- [1] S. Fish, “UGV’s in Future Combat Systems,” *Proceedings of SPIE - The International Society for Optical Engineering*, v 5422, *Unmanned Ground Vehicle Technology VI*, pp. 288-291, Apr 2004.
- [2] Z., Shiller and Y.R., Gwo, “Dynamic motion planning of autonomous vehicles.” *IEEE Transactions on Robotics and Automation*, 7(2):241-249, April 1991.
- [3] T. Simeon and B. Dacre-Wright, “A practical motion planner for all-terrain mobile robots,” *International Conference on Intelligent Robots and Systems*, pp. 1357-1363, 1993.
- [4] S. Farritor, H. Hacot and S. Dubowsky, “Physics-based planning for planetary exploration,” *Proceedings of the IEEE Int. Conf. on Robotics and Automation*, 1998.
- [5] M. Cherif, “Motion Planning for All-Terrain Vehicles: A Physical Modeling Approach for Coping with Dynamic and Contact Interaction Constraints,” *IEEE Transactions on Robotics and Automation*, vol. 15, no. 2, pages 202-218, April 1999.
- [6] M. Tarokh, R. Chan and C. Song, “Path planning of rovers using fuzzy logic and genetic algorithm,” *Proc. World Automation Conf., ISORA-026*, pp. 1-7, 2000.
- [7] H. Seraji, “Traversability index: A new concept for planetary rovers,” *Proceedings of IEEE Int'l. Conf. on Robotics and Automation*, 1999.
- [8] A. Kelly and A. Stentz, “An Approach to Rough Terrain Autonomous Mobility,” *International Conference on Mobile Planetary Robots*, 1997.
- [9] M. P. Mann and Z. Shiller: “Dynamic Stability of Off-Road Vehicles: A Geometric Approach,” *Proceedings of IEEE Int. Conf. on Robotics and Automation*, pp. 3705-3710, 2006
- [10] O. Khatib, “Real-time Obstacle Avoidance for Manipulators and Mobile Robots,” *International Journal of Robotics Research*, Vol. 5, No. 1, pp. 90-98, 1986
- [11] S. Ge and Y. Cui, “Dynamic Motion Planning for Mobile Robots Using Potential Field Method,” *Autonomous Robots*, Vol. 13, 2002
- [12] J. Barraquand and B. Langlois and J. Latombe, “Numerical Potential Field Techniques for Robot Path Planning,” *IEEE Trans. on Systems, Man, and Cybernetics*, Vol. 22, No.2, pp. 224-241, 1992
- [13] H. Tanner, S. Loizou, and K. Kyriakopoulos, “Nonholonomic Navigation and Control of Cooperating Mobile Manipulators,” *IEEE Transaction on Robotics and Automation*, Vol. 19, No. 1, 2003
- [14] H. Haddad, M. Khatib, S. Lacroix and R. Chatila., “Reactive Navigation in Outdoor Environments using Potential Fields,” *Proc. of Intl. Conf. on Robotics and Automation*, pp. 1232-1237, 1998.
- [15] Shimoda, S., Kuroda, Y., and Iagnemma, K., “High Speed Navigation of Unmanned Ground Vehicles on Uneven Terrain using Potential Fields,” to appear in *Robotica*, 2007
- [16] Spenko, M., Kuroda, Y., Dubowsky, S., and Iagnemma, K., “Hazard Avoidance for High Speed Unmanned Ground Vehicles in Rough Terrain,” *Journal of Field Robotics*, Vol. 23, No. 5, pp. 311-331, 2006
- [17] Shiller, Z., Fujita, Y., Ophir, D., Nakamura, Y., “Computing a set of Local Optimal Paths through Cluttered Environments and Over Open Terrain,” *IEEE Int. Conf. on Robotics and Automation*, 2004.
- [18] Z., Shiller, “Obstacle Traversal for Space Exploration,” *Proceedings of the IEEE Int. Conf. on Robotics and Automation*, 2000.