Receding Horizon Model-Predictive Control for Mobile Robot Navigation of Intricate Paths

Thomas M. Howard, Colin J. Green, and Alonzo Kelly

Abstract As mobile robots venture into more difficult environments, more complex state-space paths are required to move safely and efficiently. The difference between mission success and failure can be determined by a mobile robots capacity to effectively navigate such paths in the presence of disturbances. This paper describes a technique for mobile robot model predictive control that utilizes the structure of a regional motion plan to effectively search the local continuum for an improved solution. The contribution, a receding horizon model-predictive control (RHMPC) technique, specifically addresses the problem of path following and obstacle avoidance through geometric singularities and discontinuities such as cusps, turn-in-place, and multi-point turn maneuvers in environments where terrain shape and vehicle mobility effects are non-negligible. The technique is formulated as an optimal controller that utilizes a model-predictive trajectory generator to relax parameterized control inputs initialized from a regional motion planner to navigate safely through the environment. Experimental results are presented for a six-wheeled skid-steered field robot in natural terrain.

1 Introduction

Mobile robot navigation is the challenge of selecting intelligent actions from the continuum of possible actions that make progress towards achieving some goal under the constraints of limited perceptual information, computational resources, and planning time of the system. It also often viewed as the problem of balancing path following and obstacle avoidance in autonomous system architectures. Regional motion planning is the problem of planning beyond the sensor horizon.

A state-space trajectory is typically defined as a vector valued function of monotonic time (t). There are, however, circumstances where time is replaced by po-

Robotics Institute, Carnegie Mellon University, Pittsburgh, PA 15213-3980, USA, email: {thoward,cjgreen,alonzo}@ri.cmu.edu

tentially nonmonotonic functions of distance (s) or heading (ψ) to form a path. Path representations are used to achieve behaviors that allows velocity to remain unspecified. A cusp is a point in a trajectory where linear velocity changes sign. While cusps are discontinuous in path curvature, they are not discontinuous in state space trajectories and are perfectly feasible motions. Furthermore, the concept of forward on a path is not well-defined for cusps (and likewise for point turns) whereas forwards in time always has meaning. The capacity of a state space trajectory representation to remove discontinuities and permit a forward horizon to be defined are the basis of our preference for this representation.

A reference trajectory is the state-space trajectory ($\mathbf{x}(t)$[1]) provided by a regional motion planner (or other form of global guidance). The reference actions ($\mathbf{u}(\mathbf{x},t)$) are the inputs which cause the vehicle to follow the path perfectly in the absence of disturbances. In the presence of disturbances, the reference input signals that correspond to a disturbance free trajectory must be augmented by corrective actions to null the following error over some time horizon.

1.1 Motivation

As mobile robots navigate intricate motion plans composed of cusps, turn-in-place, and multi-turn maneuvers, the geometric singularities and discontinuities of these inflection points become problematic. Commonly applied techniques cannot generally reason about actions beyond these problematic points, which can endanger the system or impede path following performance by limiting the navigation horizon.

Consider the situation illustrated in Figure 1. In this example, the mobile robot deviates from the reference trajectory from disturbances including errors in modeling dynamics, terramechanical properties, and mobility.

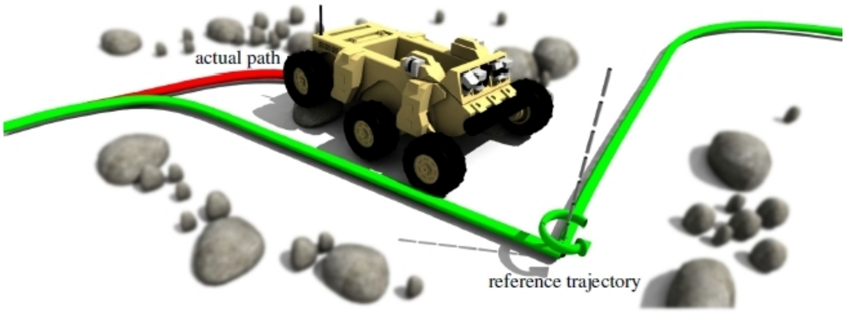


Fig. 1 An illustration of a vehicle attempting to to follow a reference trajectory with geometric singularities.

[1] The state ($\mathbf{x}$) contains the vehicle position, orientation, velocity, or any other quantity of interest

The popular class of pursuit algorithms [1] will round path corners, avoid cusps, and fail for turn-in-place maneuvers where the pursuit point becomes undefined. In contexts where such intricate maneuvers were generated by a path planner in order to avoid obstacles, a pursuit planner is inadequate. Sampling-based obstacle avoidance techniques [6] sometimes fail for intricate path navigation because of the computational resources required to search the entire input or state space densely enough to find an acceptable solution.

For effective intricate path navigation, a technique is needed which can exploit the reference trajectory structure to search in the local continuum for actions which minimize path deviation and avoid obstacles. This is the process of parametric relaxation, the technique of rendering a functional on a few parameters in order to permit relaxation of a trajectory (for optimization purposes) by searching a small number of degrees of freedom.

1.2 Related Work

There has been substantial research in the problem of developing effective, efficient mobile robot navigators. Early path following controllers operate on the assumption of tracking a single lookahead point and have been greatly extended in the literature [4]. In [12], effective search spaces for navigation in roads and trails were produced by generating nudges and swerves to the motion that reacquires the lane center.

An alternative approach involves sampling in the input space of the vehicle. In [6], navigation search spaces were generated by sampling in the input space of curvature. This approach also estimated the response of each action through a predictive motion model subject to the initial state constraints to more accurately predict the consequences of the actions. Egographs [8] represent a technique for generating expressive navigation search spaces offline by precomputing layered trajectories for a discrete set of initial states. Precomputed arcs and point turns comprised the control primitive sets used to autonomously guide planetary rovers for geologic exploration [2] where convolution on a cost or goodness map determined the selected trajectory. This approach was an extension of Morphin [11], an arc-planner variant where terrain shape was considered in the trajectory selection process. Another closely related algorithm is the one presented in [3], where an arc-based search space is evaluated based on considering risk and interest.

Other techniques such as rapidly-exploring random trees [7] have been effectively used to generate search spaces around the mobile robot to navigate cluttered, difficult environments and generate sophisticated maneuvers including u-turns. [9] presents a reactive path following controller for a unicycle type mobile robot built with a Deformable Virtual Zone to navigate paths without the need for global path replanning.

1.3 Discriminators

The main contribution of this work is the development of a receding horizon model-predictive controller (RHMPC) that effectively navigates intricate paths in complex environments. The algorithm leverages the capacity to generate the reference controls for a given reference trajectory. This capability exists because the sequence of reference controls can be generated by a trajectory generator that understands the association between actions and the corresponding state-space trajectory. Our particular preference is parameterized controls, but the key issue is that the controls are known, however represented, which correspond exactly to the reference trajectory. The corresponding reference trajectory inputs are already available in many regional motion planner implementations, so this simply requires that this additional information is passed to the navigator with the reference trajectory. Field experiments results demonstrate that the proposed technique can effectively navigate intricate paths composed of path singularities and discontinuities.

2 Technical Approach

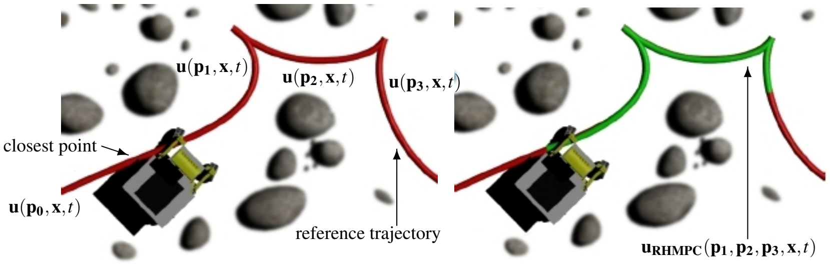
This section describes the issues related to navigation of intricate paths generated by regional motion planners, the methods by which parameterized controls are generated, and the trajectory optimization techniques used to generate corrective actions. The trajectory follower is formulated as an optimal control problem:

$$\begin{aligned} \text{minimize:} \quad & J(\mathbf{x},\mathbf{u},t) \\ \text{subject to:} \quad & \dot{\mathbf{x}} = \mathbf{f_{PMM}}(\mathbf{x},\mathbf{u},t) \\ & \mathbf{x}(t_I) = \mathbf{x_I} \\ & \mathbf{u}(\mathbf{x}) \in \mathbf{U}(\mathbf{x}), \quad t \in [t_I, t_F] \end{aligned} \tag{1}$$

The problem is one of determining actions from a set of functions ($\mathbf{U}(\mathbf{x},t)$) to represent the control inputs ($\mathbf{u}(\mathbf{x},t)$) which, when subject to the predictive motion model ($\mathbf{f_{PMM}}(\mathbf{x},\mathbf{u},t)$), minimize a penalty function ($J(\mathbf{x},\mathbf{u},t)$). An additional requirement for the trajectory follower is that the resulting control must be defined for a specific period of time or distance. This allows the optimized path to be evaluated for hazards to ensure vehicle safety.

2.1 Control Parameterization

One of the most difficult problems in motion planning involves reducing the continuum of actions to a manageable space to search. The trajectory following technique that we present uses a portion of the reference controls, which may be only piecewise continuous, as the initial guess. First, the reference trajectory is divided into the primitives used by the motion planner as shown in Figure 2(a). For each action, there exists a set of controls that, when applied to the system, produce a path segment of a certain shape. Parameterized freedom vectors ($\mathbf{p_i}$) control the shape of each set of inputs ($\mathbf{u}(\mathbf{p_i},\mathbf{x},t)$) that define the reference trajectory.


$\mathbf{u}(\mathbf{p}_1,\mathbf{x},t)$
$\mathbf{u}(\mathbf{p}_2,\mathbf{x},t)$
$\mathbf{u}(\mathbf{p}_3,\mathbf{x},t)$
closest point
$\mathbf{u}(\mathbf{p}_0,\mathbf{x},t)$
reference trajectory
$\mathbf{u}_{\mathbf{RHMPC}}(\mathbf{p}_1,\mathbf{p}_2,\mathbf{p}_3,\mathbf{x},t)$

(a) Reference trajectory segmentation (b) Horizon-limited action extraction

Fig. 2 An illustration of the parameterized action initialization in the RHMPC technique.

The initial guess for the parameterized control inputs ($\mathbf{u_{RHMPC}}(\mathbf{p_{RHMPC}},\mathbf{x},t)$) is defined by the sequence of trajectory segments between the nearest state and the predefined fixed control horizon (Figure 2(b)). In this example, the free parameters of the receding horizon model-predictive controller ($\mathbf{p_{RHMPC}}$) are defined by a concatenation of free parameters in the control inputs:

$$\mathbf{p_{RHMPC}} = \left[\mathbf{p_1}\ \mathbf{p_2}\ \mathbf{p_3}\right]^T \tag{2}$$

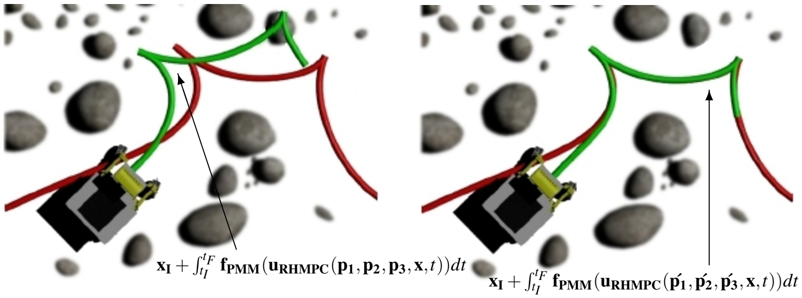
2.2 Path Deviation Optimal Control

Once the control input parameterization is determined, the next step is to modify the parameters to compensate for disturbances, approximations, and errors in the motion model. This technique seeks to minimize a cost function ($J(\mathbf{x},\mathbf{u},t)$) by modifying a set of control inputs:

$$J(\mathbf{x},\mathbf{u},t) = \Phi(\mathbf{x}(t_I),t_I,\mathbf{x}(t_F),t_F) + \int_{t_I}^{t_F} \mathscr{L}(\mathbf{x}(t),\mathbf{u}(\mathbf{p},\mathbf{x}),t)dt \tag{3}$$

The initial corrective action is evaluated through the predictive motion model subject to the initial state constraints to obtain a cost estimate as illustrated in Figure 3. While the gradient of the cost function with respect to the parameterized control input freedom exceeds a threshold, the algorithm adjusts the control inputs to minimize the integrated penalty function ($\mathscr{L}(\mathbf{x},\mathbf{u},t)$). The parameterized freedoms are modified iteratively through any standard optimization technique, such as gradient descent, as the cost function gradient is determined entirely numerically:

$$\mathbf{p}_{\mathbf{RHMPC}_i} = \mathbf{p}_{\mathbf{RHMPC}_{i-1}} - \alpha \nabla J(\mathbf{x}, \mathbf{u}, t), \quad i \geq 1 \tag{4}$$


$\mathbf{x_I} + \int_{t_I}^{t_F} \mathbf{f_{PMM}}(\mathbf{u_{RHMPC}}(\mathbf{p_1}, \mathbf{p_2}, \mathbf{p_3}, \mathbf{x}, t))dt$
$\mathbf{x_I} + \int_{t_I}^{t_F} \mathbf{f_{PMM}}(\mathbf{u_{RHMPC}}(\mathbf{p'_1}, \mathbf{p'_2}, \mathbf{p'_3}, \mathbf{x}, t))dt$

(a) Simulation of the initial action

(b) Simulation of a corrective action with optimized parameters

Fig. 3 An illustration of the parameterized action correction in the RHMPC technique.

2.3 Integrating Observed Cost Information

There are several situations when a mobile robot should intentionally deviate from the reference trajectory. Navigating around recently observed static and dynamic obstacles faster than the replanning rate of the regional motion planner is important when perceptual information is frequently updated. Another reason for deviation is the suboptimality of the reference trajectory itself. One solution is to stop and request a refined or alternative plan. A potentially better method is to include cost information into the utility functional optimized by the receding horizon model-predictive controller to determine the obstacle avoidance maneuver. The presented technique is naturally suited to deform the current action for local obstacle avoidance and path smoothing. The desired behaviors can be integrated by modifying the cost function to include a weighted penalty for obstacle cost.

3 Implementation

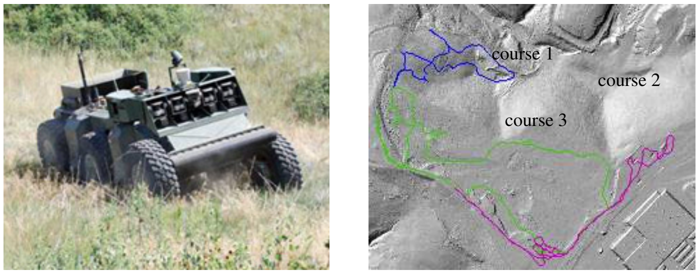
The regional motion planer used to generate feasible reference trajectories for these experiments runs A* on a graph composed of regularly arranged discrete nodes in a state space, similar to [10]. The connectivity between nodes in the discretized graph was provided by a motion template consisting of forward, reverse, and turn-in-place trajectories with lengths varying between $3m$ and $9m$. This particular implementation operated on a $60m$ x $60m$ vehicle-centered cost map. Reference trajectory updates were provided by the regional motion planner at a rate of $2Hz$.

The resulting reference trajectory is a series of sequential independent parameterized trajectories. Intricate paths composed of multiple cusps and/or turn-in-place actions often result from the diversity of edges in motion planning graphs and the complexity of the environment. The model-predictive trajectory generator [5] was used in both the motion template generation and the path deviation optimal control. Actions in the motion template were composed of constant linear velocities and either second-order spline curvature functions parameterized by distance or constant angular velocity functions parameterized by heading. Generic spline classes defined by sequences of individual command profiles parameters were optimized by the receding horizon model-predictive controller. The corrective actions were generated by the receding horizon model-predictive controller at a rate of $20Hz$.

4 Experiments

A set of experiments were designed as a comparison between a navigator that used the presented trajectory follower and one that directly executed the regional motion plan. Both systems used the same version of a lattice planner that searches dynamically feasible actions which was specifically designed for the test platform. Each field experiment was required to achieve a series of waypoints in an environment with updating perceptual information generated by an on-board perception system combined with limited overhead prior data.

The platform for the field experiments was Crusher (Figure 4(a)), a six-wheeled skid steered outdoor mobile robot. The multi-kilometer experiments were conducted at a test site in Pittsburgh, Pennsylvania with variable off-road terrain (Figure 4(b)).



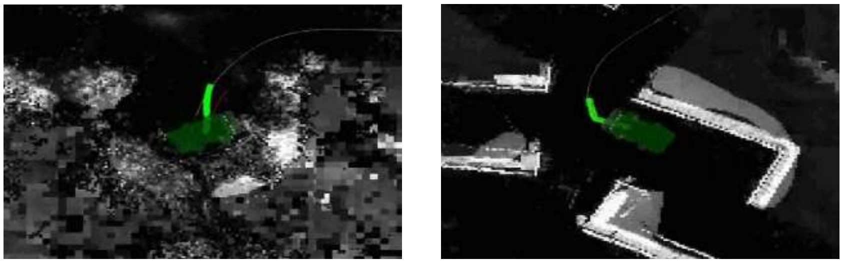
(a) Crusher (b) The field experiment courses

Fig. 4 The mobile robot and test environment for the trajectory follower field experiments.

Integrated path cost was the main metric used to measure success for the field experiments, which is related to the risk, mobility, and traversability for the vehicles configuration in the environment. While inherently unitless and scaleless, it provides the best metric for measuring performance because both the motion planner and the trajectory follower optimize this quantity.

5 Results

This section presents the results of the three field experiments comparing the performance of the presented trajectory follower to a system directly executing the regional motion plan. Figure 5 shows several selected examples from the field experiments where the receding horizon model-predictive control was used to navigate intricate paths in varied, natural terrain. Two different situations are shown involving geometric singularities and discontinuities including cusps and turn-in-place actions where the generated RHMPC action is shown as a solid green line.



(a) Planning a control that reasons about a turn-in-place action and compensates for a path following disturbance

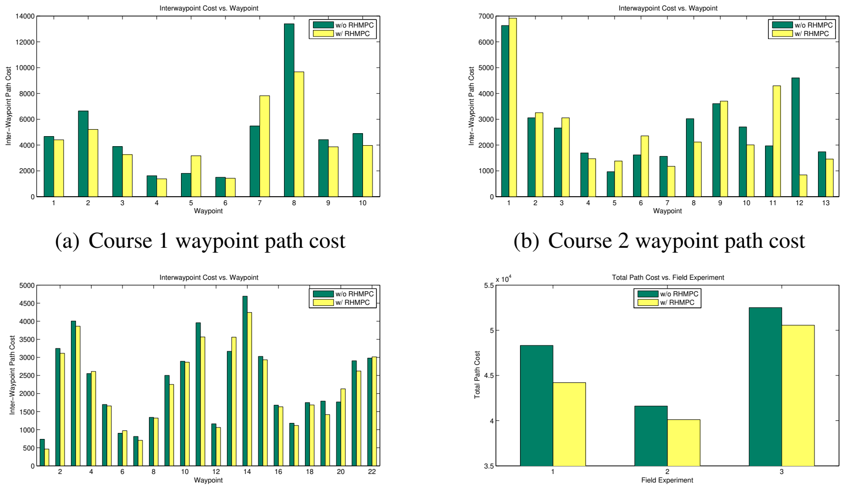
(b) Planning a control with a constant horizon through a future turn-in-place action

Fig. 5 Selected examples of the RHMPC navigator used in the multi-kilometer field experiments.

Figure 5(a) shows the receding horizon model-predictive control determined for following of a trajectory with an initial turn-in-place action with path following disturbance. The current vehicle state is off the reference trajectory and a corrective control is determined which adjusts the length of the angular velocity command and bends the straight segment to reacquire the path in a feasible manner. The last example shown in Figure 5(b) involves planning through a future turn-in-place action between two nominally straight segments. This example shows the flexibility of the technique, where the turn-in-place action does not necessarily need to start or end at a specific point in the receding horizon model-predictive control.

Figure 6 shows the integrated cost of each systems between each waypoint. It is useful to look at each waypoint-waypoint segment separately because each one can be considered to be an independent trial. On average, the system using the trajectory follower slightly outperformed or achieved a similar level of performance of the alternative system.

For portions of the course where disturbances relative to the predicted motion are uncommon or the local cost gradient was small near the disturbances very little



(a) Course 1 waypoint path cost

(b) Course 2 waypoint path cost

(c) Course 3 waypoint path cost

(d) Total path cost

Fig. 6 The waypoint and total path cost for a series of comparison runs on three courses

improvement would be expected, with more improvement expected in cases where small system disturbances can quickly lead to significantly different path cost. Figure 6(d) shows the total integrated cost for each of the three field experiments. It is important to note that while the trajectory follower did not outperform the alternative system between every waypoint, it did improve the overall performance of the system by up to 7.2%. The variability in the results is expected because of the chaotic nature of outdoor mobile robots were any number of small changes can cause the robot to select a significantly different path.

6 Conclusions

The receding horizon model-predictive control algorithm enables mobile robots to navigate intricate paths by utilizing the paths by relaxing parameterized controls that correspond exactly to the path shape. This technique enables planning through inflection points and turn-in-place actions in paths to better reason about the recovery trajectory. This method makes it possible to intelligently search the local continuum for an action which minimizes path following error and/or avoids obstacles. It also enables several other important behaviors including the capacity to define a utility function in situations where pursuit planners fail and the ability to correctly follow path discontinuities like cusps which are otherwise feasible motions. Several multi-kilometer field experiments demonstrated that the inclusion of the presented trajectory follower as a mobile robot navigator improves upon the metric that the regional motion planner minimizes.

Acknowledgments

This research was conducted at the Robotics Institute of Carnegie Mellon University under contract to NASA/JPL as part of the Mars Technology Program. Field experiments on the Crusher mobile robot platform were sponsored by the Defense Advanced Research Project Agency (DARPA) under contract Unmanned Ground Combat Vehicle PerceptOR Integration (contract number MDA972-01-9-0005). The views and conclusions contained in this paper are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the U.S. Government.

References

1. O. Amidi. Integrated mobile robot control. Technical Report CMU-RI-TR-90-17, Carnegie Mellon, 1990.
2. J.J. Besiadecki, P.C. Leger, and M.W. Maimone. Tradeoffs between directed and autonomous driving on the Mars Exploration Rovers. *International Journal of Robotics Research*, 26(91):91–104, January 2007.
3. D. Bonnafous, Sm. Lacroix, and Thierry Siméon. Motion generation for a rover on rough terrains. In *Proceedings of the 2001 IEEE/RSJ International Conference on Intelligent Robots and Systems*, volume 2, pages 784–789, October 2001.
4. R.C. Coulter. Implementation of the pure pursuit path tracking algorithm. Technical Report CMU-RI-TR-92-01, Carnegie Mellon, 1992.
5. T.M. Howard and A. Kelly. Optimal rough terrain trajectory generation for wheeled mobile robots. *International Journal of Robotics Research*, 26(2):141–166, February 2007.
6. A. Kelly and T. Stentz. Rough terrain autonomous mobility - Part 2: An active vision and predictive control approach.
7. Y. Kuwata, A. Fiore, J. Teo, E. Frazzoli, and J.P. How. Motion planning for urban driving using RRT. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1681–1686, September 2008.
8. A. Lacze, Y. Moscovitz, N. DeClaris, and K. Murphy. Path planning for autonomous vehicles driving over rough terrain. In *Proceedings of the 1998 IEEE ISIC/CIRA/ISAS Joint Conference*, pages 50–55, September 1998.
9. L. Lapierre, R. Zapata, and P. Lepinay. Combined path-following and obstacle avoidance control of a wheeled robot. *International Journal of Robotics Research*, 26(4):361–375, April 2007.
10. M. Pivtoraiko, R. Knepper, and A. Kelly. Optimal, smooth, nonholonomic mobile robot motion planning in state lattices. Technical Report CMU-RI-TR-07-15, Carnegie Mellon, 2007.
11. R. Simmons, E. Krotkov, L. Chrisman, F. Cozman, R. Goodwin, M. Hebert, L. Katragadda, S. Koenig, G. Krishnaswamy, Y. Shinoda, W.L. Whittaker, and P. Klarer. Experience with rover navigation for lunar-like terrains. In *Proceedings of the 1995 IEEE Conference on Intelligent Robots and Systems*, volume 1, pages 441–446, August 1995.
12. S. Thrun, M. Montemerlo, H. Dahlkamp, D. Stavens, A. Aron, J. Diebel, P. Fong, J. Gale, M. Halpenny, G. Goffmann, K. Lau, C. Oakley, M. Palatucci, V. Pratt, and P. Stang. Stanley: The robot that won the DARPA Grand Challenge. *Journal of Field Robotics*, 23(9):661–692, 2006.