ACHIEVING INCREASED MOBILITY AND AUTONOMY FOR GROUND VEHICLES OVER ROUGH TERRAIN

Panagiotis Tsiotras and Raghvendra V. Cowlagi

Daniel Guggenheim School of Aerospace Engineering

Georgia Institute of Technology, Atlanta, GA 30332-0150

ABSTRACT

We summarize some recent results on modeling and control of ground vehicles navigating in high-speed over rough terrain. We start with the modeling of expert race (rally) driving techniques, and we then propose a new graph-search method to bridge the gap between the path-planning and trajectory generation layers in the motion planning control hierarchy. The latter result is of independent theoretical interest, as it can be applied to any graph search problem when transitions between the nodes of the graph depend on the prior history of the path.

1. INTRODUCTION

Reducing the risk for human lives while operating in a hazardous or hostile environment has led to the development of unmanned, autonomous and semi-autonomous vehicles for both commercial and military applications. A typical military mission involves driving the vehicle from point A to point B, avoiding any obstacles, while minimizing the exposure to danger; see Fig. 1. In general, this involves driving with maximum velocity.
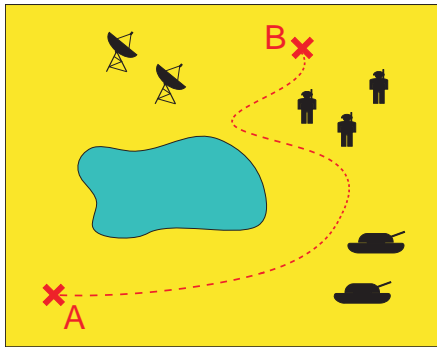

Fig. 1: A typical military mission involves an autonomous vehicle entering a hazardous area, while avoiding obstacles and/or minimizing its exposure to enemy threats and countermeasures.

Achieving autonomous operation in an open, unknown, unstructured terrain, while minimizing time still remains a major challenge in the development of land vehicles. This is due to the uncertainty of the environment the vehicle operates in, the poor characterization of the complex physics

at the interface between the wheels and the ground, the dynamics of the vehicle itself, and the overall integration of the vehicle's control algorithm with the information gathered from the onboard sensors. Autonomous driving of wheeled vehicles at high speed, in particular, adds a new level of complexity due to the time constraints imposed by the small reaction times and the nonlinear characteristics of the vehicle at those extreme regimes.

The emphasis on high-speed is motivated by the advantages in terms of reaction time, minimization of exposure to danger for vehicles operating in enemy territory, increase of supply line capacity, reduction in delivery time of materiel, etc (Fig. 2(a)). We should also mention the fact, confirmed by several Army studies, that the difficulty of successfully engaging and hitting a target increases disproportionately with the target speed (Fig. 2(b)). In summary, increasing the speed of a vehicle has a an immediate and positive correlation on the success of its mission.
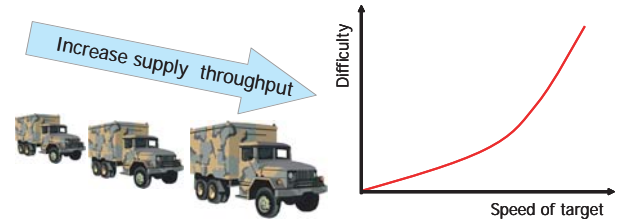
(a) Convoys throughout can be increased with vehicle speed.

(b) Success of hitting a target decreases with target speed.

Fig. 2: A safe increase of the vehicle's operating speed will lead to increase convoy speed and an decrease in successful ambush attacks.

In this paper we summarize several recent results we have developed over the past couple of years on the path planning and motion planning problem for ground vehicles navigating in an environment full of obstacles at high speed. At the core of our approach is the modeling of expert human race driving techniques. Indeed, the fastest off-road vehicles can be found in open country rally-cross racing. Rally racing requires a great deal of practice and skill (Frere 1992; Jenkinson 1959). Rally race drivers operate the vehicle at the limits of its handling capacity with very slim safety margins. They induce oversteer and skidding through corners to optimize their trajectory and prepare the vehicle for the next maneuver; see Fig. 3.

		Form Approved
Report Documentation Page		*Form Approved* *OMB No. 0704-0188*

Public reporting burden for the collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to a penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.

1. REPORT DATE **DEC 2008**	2. REPORT TYPE **N/A**	3. DATES COVERED **-**

4. TITLE AND SUBTITLE	5a. CONTRACT NUMBER
Achieving Increased Mobility And Autonomy For Ground Vehicles Over Rough Terrain	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **Daniel Guggenheim School of Aerospace Engineering Georgia Institute of Technology, Atlanta, GA 30332-0150**	8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S)
	11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release, distribution unlimited

13. SUPPLEMENTARY NOTES
See also ADM002187. Proceedings of the Army Science Conference (26th) Held in Orlando, Florida on 1-4 December 2008, The original document contains color images.

14. ABSTRACT

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**	**UU**	**8**	

Standard Form 298 (Rev. 8-98)
Prescribed by ANSI Std Z39-18

Fig. 3: Race drivers often deliberately induce skidding in corners to minimize reaction time and reorient the vehicle correctly for the next maneuver.

At the level of the vehicle dynamics, when a vehicle operates at high speed and over rough terrain, the common non-slipping conditions are no longer valid. We have used saturated friction models, which take into consideration the possibility of the wheel's reduced or complete loss of adhesion with the ground (Canudas de Wit et al. 2003; Tsiotras et al. 2004; Velenis et al. 2005). Furthermore, high-speed driving, and rally racing in particular, incorporates several specialized techniques (e.g., power oversteer, Kansei drift, braking and feint drift, pendulum turn, clutch kick, handbrake drift, etc.) which are non-common during everyday driving. The study of these techniques, their understanding, and their consistent reproduction is also of prime importance in this context.

At the path and motion planning levels, several of the currently available path-planning algorithms for autonomous vehicles (cf. (LaValle and Kuffner 2001), (Hsu et al. 2002), (Frazzoli et al. 2002)) are tailored towards the high-end of the unmanned vehicle spectrum. Smaller-scale, low-cost vehicles may not have the on-board resources to implement some of the sophisticated algorithms available in the literature. Intelligent path planners for these vehicles need to explicitly address the constraints imposed by the hardware in a consistent manner and as early on in the control design process as possible. In Section 3 we propose a computationally efficient method to incorporate kinodynamic constraints, imposed on the motion-planning problem by the vehicle dynamics and the environment. Along with the multi-resolution path planner proposed in (Tsiotras and Bakolas 2007; Cowlagi and Tsiotras 2008, 2007; Bakolas and Tsiotras 2008), these results constitute a computationally efficient overall architecture for path planning and motion planning for ground vehicles operating in high speed.

2. MODELING EXPERT RALLY DRIVING TECHNIQUES

In this section by presenting empirical information on Trail-Braking, one of the commonly used rally racing maneuvers. Trail-Braking is a technique used by rally drivers to negotiate single corners at high speeds (O'Neil 2006a). Typically, an average/novice driver negotiates a corner by first braking to regulate the speed, then by releasing the

brakes and steering the vehicle along the corner, and finally by accelerating after the exit of the corner. In Trail-Braking deceleration of the vehicle by braking continues even after steering has commenced. It is used when the approach speed to the corner is high. An example of the steering and throttle/braking trail-braking commands to negotiate a 90 deg left corner as shown in Fig. 4.
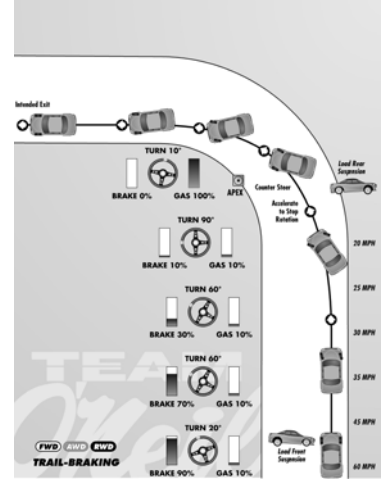

Fig. 4: Empirical description of the Trail-Braking maneuver; from (O'Neil 2006b).

2.1 Minimum-Time Cornering

We have shown (Velenis et al. 2007b,a, 2008) that a TB maneuver can the reproduced as the result of a minimum-time cornering problem subject to certain endpoint boundary conditions. The vehicle model used is a standard single-track vehicle model that includes load transfer effects. Load transfer (owing to inertia effects during acceleration and deceleration) are crucial when maneuvering over a surface with reduced friction, and cannot be neglected. The model is summarized in the equations below

$$
\begin{aligned}
m\ddot{x} &= f_{Fx}\cos(\psi+\delta) - f_{Fy}\sin(\psi+\delta) \\
&\quad + f_{Rx}\cos\psi - f_{Ry}\sin\psi \tag{1} \\
m\ddot{y} &= f_{Fx}\sin(\psi+\delta) + f_{Fy}\cos(\psi+\delta) \\
&\quad + f_{Rx}\sin\psi + f_{Ry}\cos\psi \tag{2} \\
I_z\ddot{\psi} &= (f_{Fy}\cos\delta + f_{Fx}\sin\delta)\ell_F - f_{Ry}\ell_R \tag{3} \\
I_F\dot{\omega}_F &= T_F - f_{Fx}r \tag{4} \\
I_R\dot{\omega}_R &= T_R - f_{Rx}r, \tag{5}
\end{aligned}
$$

where the tire forces are computed from $f_{ij} = f_{iz}\mu_{ij}$ ($i = F, R$, $j = x, y$), where f_{iz} is the normal load at each of the front and rear axles, and μ_{ij} is the longitudinal and lateral friction coefficients of the front and rear tires. The friction coefficients μ_{ij} can be calculated using, for instance, Pacejka's Magic formula (Bakker et al. 1987), normalized by the corresponding axle normal load f_{iz}.

The front and rear axle normal loads, including the load

transfer effect, are given by

$$f_{Fz} = \frac{\ell_R mg - hmg\mu_{Rx}}{L + h\left(\mu_{Fx}\cos\delta - \mu_{Fy}\sin\delta - \mu_{Rx}\right)}, \quad (6)$$

$$f_{Rz} = mg - f_{Fz}, \quad (7)$$

where $L = \ell_F + \ell_R$ and h is the vertical distance of the center of mass of the vehicle from the ground.

By applying optimal control theory and imposing that the vehicle completes all turning before exiting the corner, we obtain the trajectory shown in Fig. 5. The vehicle induces large slip angles in order to initiate turning about its yaw axis early on, and complete all turning while still inside the corner (Fig. 6(b)). The strategy is quite different than neutral steering that tends to keep the vehicle slip angle small (dash-dot line in Fig. 6(b)).
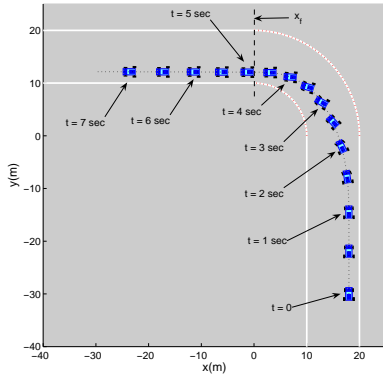

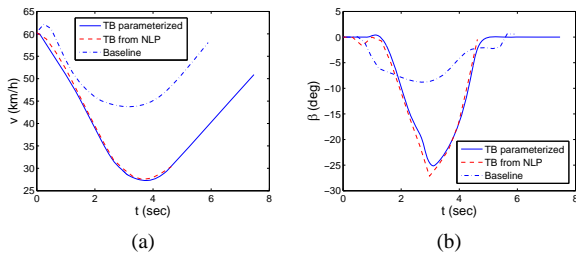Fig. 5: Trail-Braking cornering: vehicle trajectory from numerical optimization.

Fig. 6: Velocity and sideslip angle histories for a Trail-Braking maneuver; results from numerical optimization.

The results from the optimization have allowed us to introduce a simple parameterization of the control inputs that simplifies considerably the problem. The parameterization uses specific time instances to switch between constant or linear functions for the steering and throttle/braking commands as shown in Fig. 7. These input parameterization criteria are valid for a wide range of corner geometries; see Fig. 8.

The previous results have been validated against a high-fidelity vehicle model (CarSim) including all four wheels,
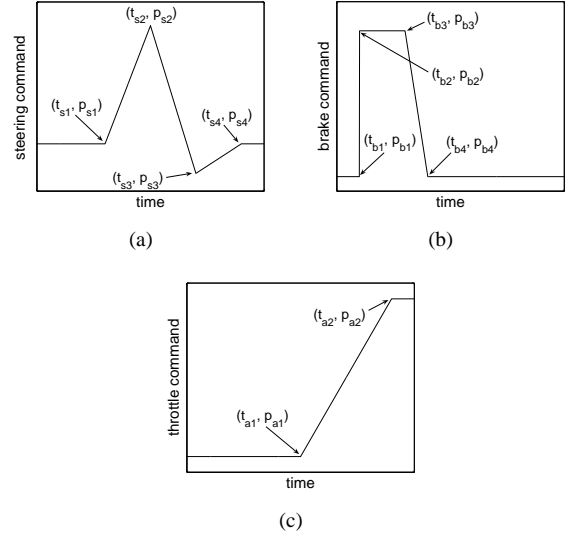

Fig. 7: Parameterized steering, braking and throttle inputs for Trail-Braking.

suspension dynamics etc. The optimization criterion used during this step is given by

$$\hat{\mathcal{J}} = W_t t_f + W_r e_r + W_d e_d(t_f) + W_\psi e_\psi(t_f) \\ + W_v e_v(t_f) + W_y e_y(t_f), \quad (8)$$

where, t_f is the final time, $e_r = \sum_{k=1}^{N} e_r(t_k)$ is the cumulative absolute value of the position error from the road limits, $e_d(t_f)$ is the absolute value of the lateral deviation of the vehicle from the inner limit of the road at t_f, $e_\psi(t_f)$ is the final absolute orientation error, $e_v(t_f)$ is the final absolute lateral velocity of the vehicle, and $e_y(t_f)$ is the final absolute yaw rate error of the vehicle. The weights W_i ($i = t, c, d, \psi, v, y$) are used for non-dimensionalization and to adjust the relative significance between the terms in the right-hand-side of (8). The results of these high-fidelity simulations, and for different corner geometries, are shown in Fig. 9.

3. KINODYNAMIC MOTION PLANNING

The exact solution of the path-planning problem in an unstructured, dynamically changing environment is known to be computationally intractable (in the language of algorithmic complexity it is "NP-hard"). In other words, the on-line computation of optimal, safe, collision-free paths (and the corresponding trajectories) constitutes an insurmountable undertaking with current computer technology.

To tackle the computational complexity, in the literature the problem is typically decomposed into two sequential steps, namely, the path-generation step, followed by the trajectory-generation step. During the first step a path from a given initial point to a given destination in the environment is computed, such that the path does not intersect any obstacles. In the second step, a time parameterization along the path yields a *trajectory* that must be followed by the
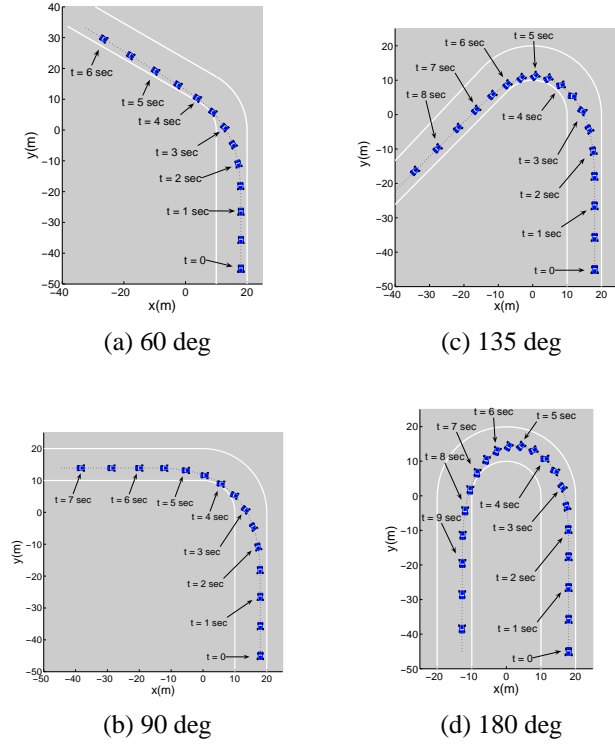
(a) 60 deg

(c) 135 deg

(b) 90 deg

(d) 180 deg

Fig. 8: Optimal trajectory through 60, 90, 135, and 180 deg corners; results from numerical optimization.

vehicle. During the last step, the trajectory needs to obey the constraints imposed by the associated vehicle dynamics, while also minimizing a certain cost function, e.g. time of travel. The computation of a suitable trajectory from the starting point to the goal destination is called the motion planning problem.

The previous decomposition of the motion planning problem to a path planning problem followed by a suitable time parameterization is somewhat artificial. Nonetheless, the approach is popular in practice since its use simplifies the overall problem. Apart from the obvious lack of optimality, this approach of solving separately the geometric and dynamic parts of the problem, may also lead to dynamically infeasible paths. The reason being that the geometric path planner has no prior knowledge of the dynamic limitations of the vehicle. If we want to ensure that the overall scheme will always generate feasible paths, we have to bridge the gap between the geometric and dynamic layers. This can only be achieved if certain information about the dynamic envelope of the vehicle is passed on to the path planner (dashed-line in Fig. 10).

In this section we describe a new scheme to include information about the class of dynamically feasible paths early on (viz at the geometric layer). Furthermore, we do this in an numerically efficient manner that is based on a non-trivial modification of Dijkstra's algorithm for the solution of shortest-path problems on graphs. In that sense, our algorithm is of more general interest than just vehicle path/motion planning. It can be used to search for shortest
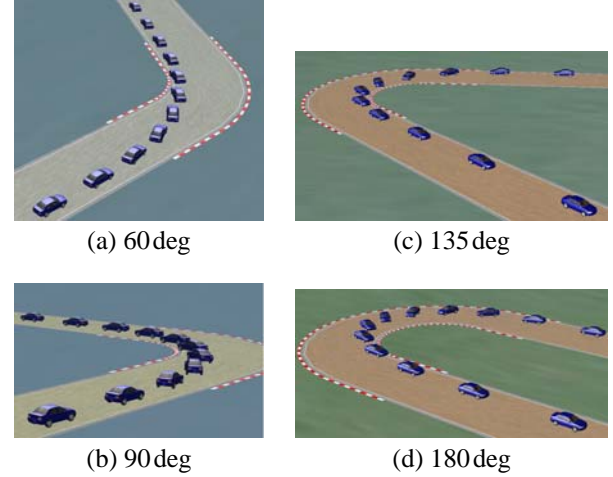

(a) 60 deg

(c) 135 deg

(b) 90 deg

(d) 180 deg

Fig. 9: Trail-Braking through the 60, 90, 135, and 180 deg corners; results of high-fidelity model validation.
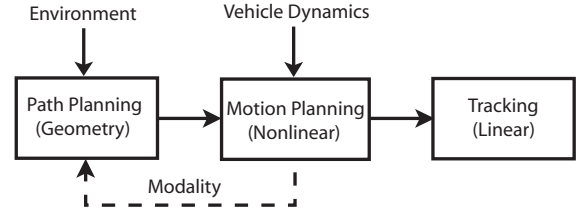

Fig. 10: The traditional control hierarchy for solving robot navigation problems (solid arrows). The second layer is typically the most difficult one as it needs to consider the full nonlinear dynamics of the vehicle. Better performance can be achieved by coupling the path-planning and motion-planning (feasible trajectory generation) layers, perhaps via the vehicle's modalities.

paths on a graph whenever the node transitions depend on the prior history of visited nodes.

3.1 Capturing Curvature Information

Current path planning algorithms based on cell decompositions of the obstacle-free space work exceedingly well for generating paths when no kinematic or dynamic constraints are present. A survey of path planning algorithms can be found in (Latombe 1991; LaValle 2006). However, the motion of the actual vehicle must obey such constraints. For instance, a vehicle always has a certain minimum turn radius r_{min} (Fig. 11). A path whose curvature does not satisfy this curvature constraint cannot be followed by the vehicle even at very low speeds. Driving a high-speed adds additional restrictions on the allowable, feasible paths. Without additional assumptions, there is no guarantee that a feasible trajectory satisfying these constraints will even exist within the channel of cells computed by a (purely geometric) path planner.

At first glance, one may argue that this is only an artifact of an inappropriate choice of the edge cost function in the associated graph representing the environment and the obstacles. Below we provide a counter-example to this
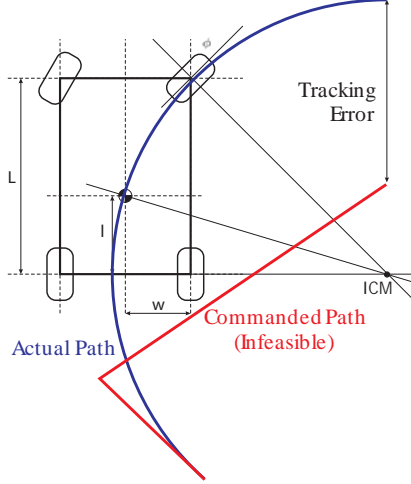
Fig. 11: A vehicle has a minimum turn radius. A path planning algorithm needs to take this information into consideration. Most path planning algorithms do not do that, leading to infeasible paths and, subsequently, to poor trajectory tracking performance.
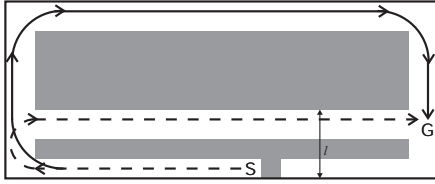

Fig. 12: Counterexample for path planning without kinematic constraint.

argument.

Consider the path planning problem depicted in Fig. 12, where S denotes the initial position, G denotes the goal, and the dark areas are obstacles. Consider two vehicles A and B, whose minimum radii of turn are kinematically constrained by r_{min}^A and r_{min}^B respectively, such that $r_{min}^A \leq \ell$ and $r_{min}^B > \ell$. Clearly, the dashed path in Fig. 12 is feasible for vehicle A, but not for vehicle B. A path planning algorithm for B ought to result in the bold path shown in Fig. 12.

Figure 13 depicts the same problem with a uniform cell decomposition. The channel containing the dashed path of Fig. 12 is denoted by cells with bold outlines. Such a channel is obviously not traversable by vehicle B. However, notice that no *pair of successive cells* is *by itself* infeasible, i.e., a channel defined by two successive cells alone always contains a feasible path. Stated differently, for any two adjacent cells, there is no cell-dependent property associated with the two adjacent cells that can be penalized by an edge cost function in order to prevent the graph search from generating a channel such as the one shown in Fig. 13(a).

It may be further argued that a feasible path is guaranteed to exist in *any* channel if the dimensions of the cells are large enough. Indeed, in (Bereg and Kirkpatrick 2005) it is shown that a curvature-bounded path with local cur-
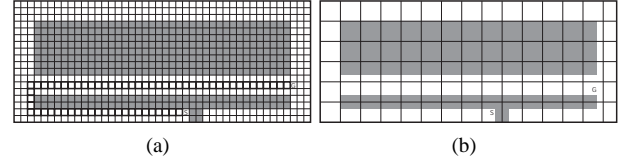

Fig. 13: (a) No pair of successive cells is by itself infeasible; (b) Cells are too large, all cells are MIXED.

vature less than or equal to $1/r_{min}$ exists in a polygonal channel if the width w of the channel satisfies the inequality $w \geq \tau r_{min}$, where τ satisfies a certain polynomial equation. The above counter-example also serves to illustrate that such a choice of cells may be too restrictive in practice. Figure 13(b) illustrates that the dimensions of the cells may be too large to capture details of the environment, i.e., the number of MIXED cells could be too large for the cell decomposition to be useful for path planning.

3.2 Working with Multiple Cells

In this paper, we propose the following approach to plan a path using cell decompositions, while incorporating path curvature information: On the topological graph associated with a given cell decomposition, we define a cost function based on $k-$tuples (i.e., histories) of nodes, for some fixed $k > 2$, such that the elements of each $k-$tuple are pairwise adjacent. The question of feasibility of traversal through $k-$tuples of cells (rather than traversal through two successive cells only) allows for more general definitions of "edge" cost functions. In particular, we can introduce suitable costs that capture the maximum approximate curvature for any path lying inside the channel. As a result, we can ensure that a feasible *trajectory* will always exist inside the computed channel of cells at the geometric, path-planning layer, even before invoking the motion-planning task.

3.3 Algorithm For History-Based Search

Next, we briefly describe an algorithm for searching for optimal paths in graphs with a history-based cost function. A rigorous description, along with analytical results on the optimality and performance of the algorithm, is given in (Cowlagi and Tsiotras 2009).

Consider a uniform cell decomposition $\mathscr{C}_d$ consisting of N cells. We may construct a graph $\mathscr{G} \equiv (\mathscr{V}, \mathscr{E})$, such that each element in the set of nodes $\mathscr{V}$ corresponds to a unique, obstacle-free cell. We label the nodes as $1, 2, \ldots, N$. Two nodes are *adjacent* if the corresponding cells are geometrically adjacent[1]. The edge set $\mathscr{E}$ consists of all pairs (i, j), $i, j \in \mathscr{V}$ with nodes i and j adjacent. We now describe two path planning problems on graph $\mathscr{G}$ associated with $\mathscr{C}_d$.

Let $g(\cdot)$ be an edge cost function defined for each pair

---

[1]We consider 4-connectivity for this work, that is, cells that have two vertices in common are said to be adjacent.

of adjacent nodes in $\mathscr{G}$. For a given $H \geq 1$, let $\tilde{g}_{H+1}(\cdot)$ be cost function with each sequence of nodes of length $H + 2$, in which pairs of successive nodes are adjacent. For given initial and terminal nodes i_S, $i_G \in \mathscr{V}$, an *admissible path* $\pi \equiv (j_0^\pi, j_1^\pi, \ldots, j_P^\pi)$ in $\mathscr{G}$ is a sequence of nodes in which all pairs of successive nodes are adjacent.

Problem 1. Let the cost of an admissible path π be

$$J(\pi) \;=\; \sum_{k=1}^{P} g((j_{k-1}^\pi, j_k^\pi)). \qquad (9)$$

Find an admissible path π^* in $\mathscr{G}$ such that $J(\pi^*) \leq J(\pi)$ for every admissible path π in $\mathscr{G}$.

Problem 2. Let the cost associated with π be

$$\tilde{\mathscr{J}}(\pi) = \sum_{k=H+1}^{P} \tilde{g}_{H+1}\left((j_{k-H-1}, j_{k-H}, \ldots, j_k)\right). \qquad (10)$$

Find an admissible path π^* in $\mathscr{G}$ such that $\tilde{\mathscr{J}}(\pi^*) \leq \tilde{\mathscr{J}}(\pi)$ for every admissible path π in $\mathscr{G}$.

Problem 1 is the usual graph search problem associated with a cell decomposition. A class of algorithms used to solve Problem 1 are the *label correcting algorithms*. A label correcting algorithm progressively searches for the least cost path starting from i_S and ending at node $i \in \mathscr{V}$, by iteratively reducing an estimate $d(i)$ of the least cost to i, called the *label* of the node i. The algorithm also maintains a list of nodes, called the OPEN list, whose labels can potentially be reduced from their current value, as well as a *backpointer* $b(i)$, which records the immediate predecessor of each node $i \in \mathscr{V}$ in the optimal path from i_S to i. Detailed descriptions of the general label correcting algorithm and particular examples, such as the Bellman-Ford, Dijkstra, and A* algorithms, may be found in (Bertsekas 2000), (Cormen et al. 2001) and (LaValle 2006).

On the other hand, Problem 2 encapsulates the history-based search that we would like to perform. The simplest example of $\tilde{g}_{H+1}$ is a binary cost function, i.e., one which classifies a $H + 2$-history of cells as "good" (low value) or "bad" (high value). Figure 14 illustrates histories (unique up to translation, rotation, and reflection operations) that may be classified as "good" for $H = 3$.

The proposed algorithm to solve Problem 2 is an extension of the standard label correcting algorithm described above. While a formal description of the proposed algorithm may be found in (Cowlagi and Tsiotras 2009), the primary idea is that of recording *multiple* histories (of length $H + 1$) of each node , and, consequently, multiple labels. The multiple histories in the proposed algorithm replace the backpointer of the standard label correcting algorithm. Figure 15 illustrates this idea.

Suppose that the node being processed at a particular instant during the execution of the algorithm corresponds to the cell with the bold, dark outline in Fig. 15(a). Suppose
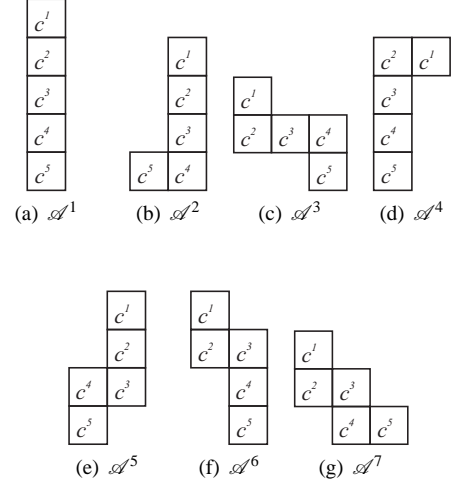

(a) $\mathscr{A}^1$ (b) $\mathscr{A}^2$ (c) $\mathscr{A}^3$ (d) $\mathscr{A}^4$

(e) $\mathscr{A}^5$ (f) $\mathscr{A}^6$ (g) $\mathscr{A}^7$

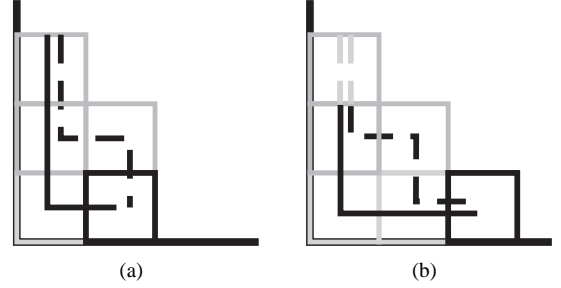Fig. 14: Illustration of tiles of "good" histories for $H = 3$

(a)   (b)

Fig. 15: Idea of recording multiple histories.

that the cost function under consideration is the binary cost function introduced above, with $H = 3$, such that the histories shown in Fig. 14 have low cost. Next, assume that the cost of the channel containing the bold path in Fig. 15(a) has a slightly lower cost than that of the channel containing the dotted path. Figure 15(b) shows the next iteration of the algorithm, where the new bold cell corresponds to the node being processed. Clearly, the channel containing the bold path has a high cost, while the channel containing the dotted path has a low cost. However, if the algorithm had discarded the channel with the dotted path in the earlier iteration, then in the next iteration, it would have been "stuck" with the channel containing the bold path, thus incurring a high cost despite a low cost alternative being available. To avoid such situations, the proposed algorithm records *all* histories of a given, finite length, and maintains labels for each of those histories. When a particular node is processed, its label is computed using *all* histories and labels of its predecessors, thus ensuring that no history is "forgotten".

Figure 16 shows the application of the proposed algorithm to the motivating problem of Fig. 12, using the binary cost function discussed before. The channel of cells marked in red is the result of the proposed algorithm, while the channel marked in blue is the result of performing a standard graph search on $\mathscr{G}$ using a cost defined on the edge set only ($H = 0$). The corresponding geometric paths

of minimum possible curvature inside the channels are also shown in the same figure. The red curve is longer of course, but it has lower maximum curvature.
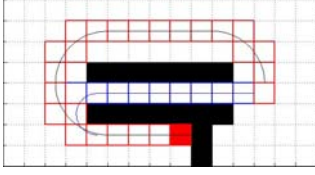

Fig. 16: The example of Fig. 12 solved using the proposed algorithm.

Figure 17 shows a more interesting example, where the environment is cluttered with randomly distributed obstacles. As before, the channel consisting of red cells is the result of the proposed algorithm, whereas the channel consisting of cells marked in blue is the result of performing a standard graph search. The corresponding geometric paths of minimum possible curvature inside the channels are also shown in the same figure. Most interestingly, Fig. 17(b) shows the corresponding optimal velocity profile from the solution of the minimum-time problem along each path (Velenis and Tsiotras 2008). Although the red path is longer, a vehicle following this path will take less time than a vehicle following the shorter (but with more and sharper turns) blue path. This example demonstrates the benefits of the proposed algorithm for the solution motion planning problems.
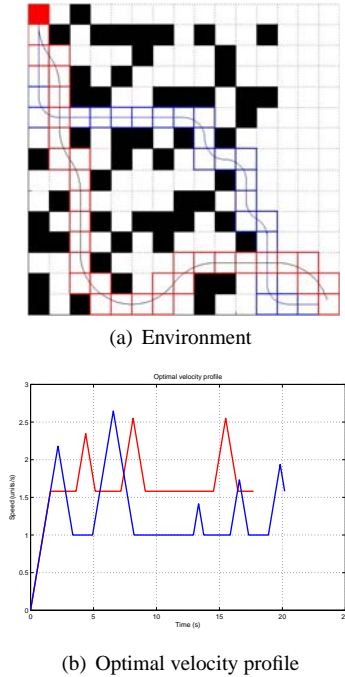

(a) Environment

(b) Optimal velocity profile

Fig. 17: Example of a cluttered environment.

In closing, it is worth-mentioning that it is easy to extend the proposed algorithm to search for an optimal channel as well as for a kinodynamically feasible path lying within that channel at the same time. Figure 18 shows simulation results of the application of such an extension to the generation of paths with maximum curvature $1/r_{\min}$, for a given $r_{\min}$. Note that, unlike the results in Fig. 16, the search for the optimal channel and construction of a geometric path within that channel was done *simultaneously* in the case of Fig. 18.

(a) $r_{\min} = 3.0$ (b) $r_{\min} = 3.5$

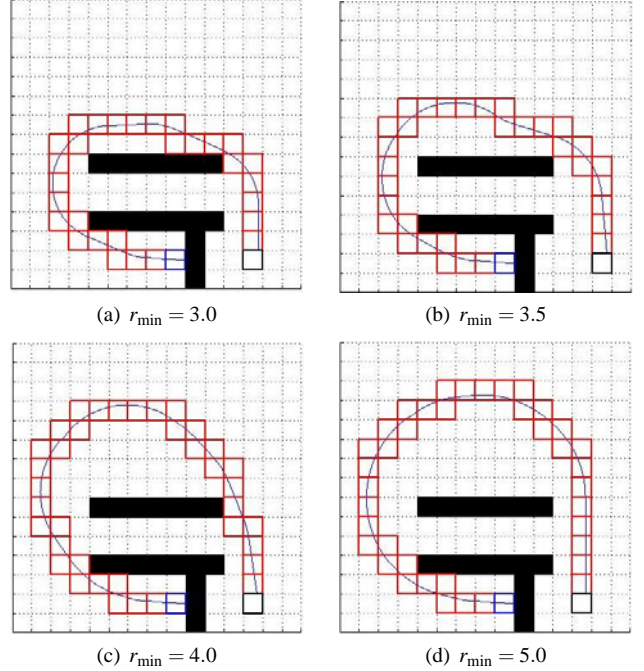(c) $r_{\min} = 4.0$ (d) $r_{\min} = 5.0$

Fig. 18: Resultant channels and trajectories for different curvature bounds. The cell marked in blue is the start cell. The environment is similar to that in the motivating example of Fig. 12. Channels and trajectories computed simultaneously.

CONCLUSIONS

We present recent results on the problem of path and trajectory generation for high-speed vehicles. In the first part of the paper we present a mathematical formalism that uses optimal control theory to generate trail-braking, high-speed maneuvers for a large variety of corner geometries. In the second part of the paper we propose a new algorithm for path planning that takes into account the kinodynamic constraints imposed by the vehicle dynamics and or the environment. The algorithm can be used in cases when the computational resources on-board the vehicle are limited.

ACKNOWLEDGMENTS

REFERENCES

Bakker, E., L. Nyborg, and H. Pacejka, 1987: Tyre modelling for use in vehicle dynamics studies. SAE Paper No. 870421, SAE Paper No. 870421.

Bakolas, E. and P. Tsiotras, 2008: Multiresolution path planning via sector decompositions compatible to onboard sensor data. *AIAA Guidance, Navigation, and Control Conference*, Honolulu, HI, aIAA Paper 2008-7238.

Bereg, S. and D. Kirkpatrick, 2005: Curvature-bounded traversals of narrow corridors. *Proc. Twenty-first Annual Symposium on Computational Geometry*, Pisa, Italy, 278–287.

Bertsekas, D. P., 2000: *Dynamic Programming and Optimal Control*, Vol. 1. Athena Scientific, Belmont, MA.

Canudas de Wit, C., P. Tsiotras, E. Velenis, M. Basset, and G. Gissinger, 2003: Dynamic friction models for road/tire longitudinal interaction. *Vehicle System Dynamics*, **39 (3)**, 189–226.

Cormen, T. H., C. E. Leiserson, R. L. Rivest, and C. Stein, 2001: *Introduction to Algorithms*. 2d ed., MIT Press.

Cowlagi, R. and P. Tsiotras, 2007: Beyond quadtrees: Cell decomposition for path planning using the wavelet transform. *46th IEEE Conference on Decision and Control*, New Orleans, LA, 1392–1397.

Cowlagi, R. and P. Tsiotras, 2008: Multiresolution path planning with wavelets: A local replanning approach. *American Control Conference*, Seattle, WA, 1220–1225.

Cowlagi, R. V. and P. Tsiotras, 2009: An algorithm for finding optimal paths in graphs for history-based cost functions, *submitted to the 2009 American Control Conference,* St. Louis, MO.

Frazzoli, E., M. A. Dahleh, and E. Feron, 2002: Real-time motion planning for agile autonomous vehicles. *Journal of Guidance, Control, and Dynamics*, **25 (1)**, 116–129.

Frere, P., 1992: *Sports Car and Competition Driving*. Bentley Publishers, Cambridge, MA.

Hsu, D., R. Kindel, J.-C. Latombe, and S. Rock, 2002: Randomized kinodynamic motion planning with moving obstacles. *The International Journal of Robotics Research*, **21 (3)**, 233–255.

Jenkinson, D., 1959: *The Racing Driver: The Theory and Practice of Fast Driving*. Robert Bentley, Inc., Cambridge, MA.

Latombe, J.-C., 1991: *Robot Motion Planning*. Kluwer Academic Publishers.

LaValle, S. M., 2006: *Planning Algorithms*. Cambridge University Press.

LaValle, S. M. and J. J. Kuffner, Jr., 2001: Randomized kinodynamic planning. *The International Journal of Robotics Research*, **20 (5)**, 378–400.

O'Neil, T., 2006a: Private communication, private communication.

O'Neil, T., 2006b: *Rally Driving Manual*. Team O'Neil Rally School and Car Control Center.

Tsiotras, P. and E. Bakolas, 2007: A hierarchical on-line path-planning scheme using wavelets. *European Control Conference*, Kos, Greece, 2806–2812.

Tsiotras, P., E. Velenis, and M. Sorine, 2004: A lugre tire friction model with exact aggregate dynamics. *Vehicle System Dynamics*, **42 (3)**, 195–210.

Velenis, E. and P. Tsiotras, 2008: Minimum-time travel for a vehicle with acceleration limits: Theoretical analysis and receding horizon implementation. *Journal of Optimization Theory and Applications*, **138 (2)**, 275–296.

Velenis, E., P. Tsiotras, C. Canudas de Wit, and M. Sorine, 2005: Dynamic tire friction models for combined longitudinal and lateral vehicle motion. *Vehicle System Dynamics*, **43 (1)**, 3–29.

Velenis, E., P. Tsiotras, and J. Lu, 2007a: Aggressive maneuvers on loose surfaces: Data analysis and input parametrization. *15th IEEE Mediterranean Control Conference*, Athens, Greece.

Velenis, E., P. Tsiotras, and J. Lu, 2007b: Modeling aggressive maneuvers on loose surfaces: The cases of trail-braking and pendulum-turn. *European Control Conference*, Kos, Greece, 1233–1240.

Velenis, E., P. Tsiotras, and J. Lu, 2008: Optimality properties and driver input parameterization for trail-braking cornering. *European Journal of Control*, **14 (4)**.