Dynamic motion planning of autonomous vehicles

2 authors, including:

Dynamic Motion Planning of Autonomous Vehicles

Zvi Shiller, *Member, IEEE,* and Yu-Rwei Gwo

Abstract-This paper presents a method for planning the motions of autonomous vehicles moving on general terrains. The method obtains the geometric path and vehicle speeds that minimize motion time considering vehicle dynamics, terrain topography, obstacles, and surface mobility. The terrain is represented by a smooth cubic B patch, and the geometric path consists of a B spline curve mapped to the surface. The time optimal motions are computed by first obtaining the best obstacle-free path from all paths represented by a uniform grid. This path is further optimized with a local optimization, using the optimal motion time along the path as the cost function and the control points of a *B* spline as the optimizing parameters. Examples are presented that demonstrate the method for a simple dynamic model of a vehicle moving on a mountainous terrain.

I. INTRODUCTION

THE problem of motion planning of autonomous vehicles consists of selecting the geometric path and vehicle speeds so as to avoid obstacles and to minimize some cost function, such as time or energy. While extensive work has focused on computing the geometric path, little attention has been given to selecting the optimal vehicle speeds. Selecting the wrong speeds can cause the vehicle to loose its path, or to waste time or energy.

In this paper a method is presented that obtains the time optimal and collision-free paths between given end points, considering vehicle dynamics, terrain topography, and surface mobility. The terrain is represented by a continuous B patch, and the path is represented by a *B* spline curve, constructed on the surface. This is in contrast to most current methods that consider flat surfaces, and paths made of simple geometric shapes, such as straight lines and circular arcs [5], [7], or other specific functions designed to smooth path curvature [6], [13]. Terrain elevation has been addressed in a limited number of works, ignoring the effects of vehicle dynamics [2], [14]. Vehicle dynamics were considered in [3] for planning feasible vehicle motions in the plane, without addressing the issues of motion optimality and obstacle avoidance.

The motion planning problem is formulated as a three-stage optimization. First, the optimal speeds along a given path are computed to minimize motion time, using an algorithm previously developed in [1] and later modified in [18]. To minimize motion time, the algorithm selects highest possible speeds that do not violate dynamic constraints on vehicle motions, such as tangential and side slip, tip-over, and separation between vehicle

and ground. The local optimal path that avoids obstacles and further minimizes motion time is obtained by a parameter optimization, using the motion time along the path as the cost function and the control points of a *B* spline as path parameters [17]. The initial guess to this optimization that is likely to converge to the global optimal path is obtained by a graph search, following the method presented in [16]. Obstacles and regions of different cultural types are considered using a mobility factor that multiplies the original coefficient of friction between the vehicle and ground. A mobility of zero is assigned to unaccessible regions, and a mobility of one is assigned to perfect roads. Other regions that allow restricted motions, such as forests and rough terrain, are assigned mobility values between zero and one.

The method presented combines various aspects of motion planning of autonomous vehicles in a unified way. Vehicle kinematics, dynamics, and terrain topography are used to formulate a single velocity curve, representing the upper bounds on vehicle motions along a given path. This allows computing a single valued cost function, facilitating the search for a better path at a higher optimization level, free of the detailed computation of the multiparameter objective function. To allow such a unified approach, we resort to simplified kinematic and dynamic models. Accurate models of rolling vehicles are quite involved and are beyond the scope of this paper. As demonstrated in the examples, even the simplest model provides very useful information for planning the motions of rolling vehicles.

While the method presented minimizes motion time, the approach is general and applicable to other path-dependent cost functions, such as energy, distance from obstacles, time spent in hostile environments, etc. The method can be extended to any path representation that maps a finite set of parameters to a geometric curve on the surface.

In this paper, we first derive vehicle kinematics, solving for its orientation and steering angle along the path. In Section III, vehicle dynamics are derived in terms of path geometry, and the constraints on vehicle motions are formulated in Section IV. The terrain and the path representations are given in Section V, and the local and global optimizations are briefly discussed in Section VI. Section VII includes several examples that demonstrate the approach for a vehicle moving on three-dimensional terrains with obstacles.

II. VEHICLE KINEMATICS

The inverse kinematics problem of a rolling vehicle consists of solving for vehicle orientation and the required steering angle that will move a given vehicle fixed point along a desired path Choosing the center of mass as the guiding point, and assuming no sliding, makes the problem nonholonomic since only incremental solutions for vehicle orientation and the steering angle are possible (see [l0] and the references therein for examples of nonholonomic systems).
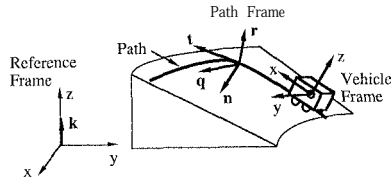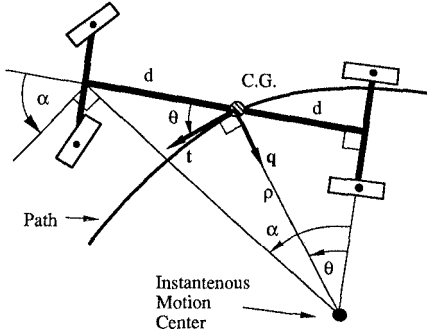
Fig. 1.   Vehicle and path fixed frames.

Fig. 2.   The steering angle α.

We consider a four-wheeled vehicle with two fixed axles, driven by fixed rear wheels and steered by the front wheels. A rectangular coordinate frame xyz is attached to the vehicle at its center of mass, and an inertial frame tqr is fixed to the terrain at every point along the path, as shown in Fig. 1. The r vector is normal to the surface, t is tangent to the path, and q is normal to both. A unit vector n in the rq plane is defined in the direction of the center of path curvature. The vectors q and n coincide only on flat surfaces.

At every point along the path, the z axis of the vehicle fixed frame is aligned with the r vector. The orientations of the other vehicle axes, assumed to be parallel to the tq plane, are determined based on the orientation and the steering angle at the previous point. For perfect rolling, the rolling axis of every wheel must pass through the instantaneous motion center (IMC). If the front axle is fixed, this results in a different steering angle of each of the front wheels. We consider only the average steering angle α as shown in Fig. 2, thus violating the assumption of perfect rolling. By fixing the front axle and assuming planar contact points, we simplify the analysis of the reaction forces. This, however, restricts the terrain to be smooth and obstacles to be large relative to vehicle size.

Referring to Fig. 2 and using simple trigonometric identities, we solve for the steering angle α that will move the mass center along the path in the direction of the tangent t

$$\tan \alpha = \frac{2d}{\sqrt{\rho^2 - d^2}} \tag{1}$$

where d is half the distance between the axles (for a symmetric vehicle), and ρ is the distance to the IMC from the mass center, obtained by intersecting the lines of the rear axle and the q vector, as shown in Fig. 2. Note that, generally, ρ is not the radius of path curvature. It can be easily shown that α is related to the angle θ between the x and t vectors by

$$\alpha = \tan^{-1}\left(2 \tan \theta\right). \tag{2}$$

The angle θ is a function of the current vehicle position x and the desired direction of motion t. Since x is the result of the
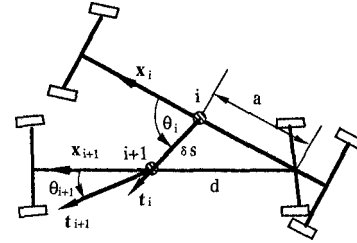

Fig. 3.   **The** vehicle at two subsequent points along the path.

previous motions, we can solve only for the incremental changes in θ.

Fig. 3 shows the vehicle at two positions: before and after an incremental move of δs along the path from some point i. Point $i + 1$ is located at the end of the vector $\delta s t_i$ from point i. During the incremental move, the rear axle slides along the x_i direction to a distance a from the point i. The new locations of the mass center and the rear axle define the new vehicle orientation. Note that θ_{i+1} is the angle between x_{i+1} and t_{i+1}. First, express x_i in terms of θ_i

$$x_i = \cos \theta_i t_i - \sin \theta_i q_i \tag{3}$$

where the t_i and q_i are known for every point i along the path. The vehicle changes its orientation from x_i to x_{i+1}

$$x_{i+1} = \frac{a}{d}x_i + \frac{\delta_s}{d}t_i \tag{4}$$

where

$$a = -\delta s \cos \theta_i + \overline{\sqrt{d^2 - \delta s^2 \sin^2 \theta_i}}.$$

The angle θ_{i+1} is obtained by the dot product

$$\theta_{i+1} = \cos^{-1}\left(x_{i+1} \cdot t_{i+1}\right). \tag{5}$$

Substituting (4) into (5) yields θ_{i+1} in terms of the known orientation at point i and the new direction of motion t_{i+1}

$$\theta_{i+1} = \cos^{-1}\left(\frac{a}{d}x_i \cdot t_{i+1} + \frac{\delta s}{d}t_i \cdot t_{i+1}\right). \tag{6}$$

The angle θ obtained by (6) is used in (2) to solve for the next steering angle. Equation (6) can be integrated from the initial point along the path, given some initial condition on θ. Note that we do not need α for the integration. Here, the steering angle is required only for verifying that it stays within the range allowed by the steering mechanism.

Although the path is specified along the terrain, we assume that it represents the motions of the center of mass. This simplifies vehicle kinematics and dynamics while introducing a small error due to the ignored relative motions between the center of mass and the path.

III. Vehicle Dynamics

In this section, the dynamic equations of a simple vehicle model are derived. The wheels consist of massless rigid disks, making point contacts with ground. The moment of inertia of the vehicle is neglected, assuming   the mass is concentrated at one point. At each contact point, there are three unknown forces: two friction and one normal force. For four contact points, this adds to a total of 12 unknowns. The six equations of motion are insufficient to solve for all the reaction forces, and the problem is indeterminate. Approximate solutions can be

found by assuming equal force distribution between any collinear forces and by stating linear relations between the normal and the friction forces [8]. The models using these assumptions are cumbersome and beyond the scope of this paper. Therefore, except for the tip-over constraint, we transfer the reaction and friction forces to the mass center. In the following, F and R represent the summations of the external friction and reaction forces acting on the vehicle, respectively.

It is useful to first define some identities. The vectors t and n, which uniquely specify path geometry, are obtained by differentiating the path, p(S), with respect to path distance S

$$t = p_s = \frac{\partial p}{\partial S} \qquad (7)$$

$$n = \frac{p_{ss}}{|p_{ss}|}$$

where

$$p_{ss} = \frac{\partial^2 p}{\partial S^2}.$$

Gravity is defined in the opposite direction of the unit vector k fixed to an inertial reference frame. The projections of the unit vectors k and n on t and q are obtained by the dot products

$$k_t = k \cdot t \qquad k_q = k \cdot q \qquad k_r = k \cdot r \qquad (8)$$

and

$$n_t = n \cdot t \qquad n_q = n \cdot q \qquad n_r = n \cdot r. \qquad (9)$$

The external forces acting on the vehicle consist of the friction forces between the vehicle and ground, the normal forces, and the gravity force. The total friction force F, tangent to the tq plane, is represented in path coordinates as

$$F = f_t t + f_q q \qquad (10)$$

where f_t and f_q are the components tangent and normal to the path, respectively. The equation of motion of the vehicle can be written in terms of the tangential speed $\dot{S}$ and the tangential acceleration $\ddot{S}$

$$f_t t + f_q q + Rr - mgk = m\kappa n\dot{S}^2 + mt\ddot{S} \qquad (11)$$

where R is the magnitude of the reaction force in the r direction, and m is the lumped vehicle mass. The projections of the external forces in the t, q, and r directions are obtained by dot multiplying both sides of (11) with the vectors t, q, and r

$$f_t = mgk_t + m\ddot{S} \qquad (12)$$

$$f_q = mgk_q + m\kappa n_q \dot{S}^2 \qquad (13)$$

$$R = mgk_r + m\kappa n_r \dot{S}^2 \qquad (14)$$

where κ is path curvature. It is convenient to express the friction forces also in the xy coordinates

$$f_x = f_t \cos\theta - f_q \sin e \qquad (15)$$

$$f_y = f_t \sin\theta + f_q \cos 8. \qquad (16)$$

Substituting (12) and (13) into (15) and (16) yields

$$f_x = m\cos\theta \ddot{S} - m\kappa n_q \sin\theta \dot{S}^2 + mg(k_t \cos\theta - k_q \sin 0) \qquad (17)$$

$$f_y = m\sin\theta \ddot{S} + m\kappa n_q \cos\theta \dot{S}^2 + mg(k_t \sin\theta + k_q \cos\theta). \qquad (18)$$

Equations (12) to (18) can be used to determine the required forces for the prescribed motion, or alternatively, to determine the feasible speed and acceleration for given limits on the friction and normal forces.

IV. DYNAMIC CONSTRAINTS

Several constraints between the vehicle and ground are considered to ensure vehicle dynamic stability along the path. The constraints treated in this paper include: a) limits on engine torque, b) limits on the coefficient of friction (sliding constraint), c) positive contact between the vehicle and ground (contact constraint), and d) tip-over. Generally, each constraint can be transformed to constraints on vehicle tangential speed and acceleration as shown below.

A. Engine Torque Constraint

The engine torque produces the friction force f_t tangent to the path at the contact points. A positive torque produces a force in the direction of motion, while a negative torque, or a braking force, produces a force in the opposite direction. This force is bounded by the maximum equivalent engine force F_{max} and the maximum braking force F_{min}

$$F_{min} \leq f_t \leq F_{max}. \qquad (19)$$

These limits are assumed to be constant and independent of speed. Substituting (12) into (19) yields the feasible acceleration range due to constraints on the equivalent engine force

$$F_{min}/m - gk_t \leq \ddot{S} \leq F_{max}/m - gk_t. \qquad (20)$$

The engine torque constraint limits only the acceleration since the effect of speed on engine torque was not considered.

B. Sliding Constraint

If the friction force required by vehicle motion (speed and acceleration) exceeds the limits

$$F^2 = f_t^2 + f_q^2 \leq \mu^2 R^2 \qquad (21)$$

the vehicle will slide and deviate from the desired path. Substituting (12)–(14) in (21) yields a quadratic inequality equation in S

$$\ddot{S}^2 + 2gk_t\ddot{S} + \kappa^2(n_q^2 - \mu^2 n_r^2)\dot{S}^4$$
$$+ 2g\kappa(k_q n_q - \mu^2 k_r n_r)\dot{S}^2 + g^2(k_q^2 + k_t^2 - \mu^2 k_r^2) \leq 0. \qquad (22)$$

Solving (22) for $\ddot{S}$ yields the feasible range of acceleration along the path due to the sliding constraint

$$\ddot{S}_d \leq \ddot{S} \leq \ddot{S}_a \qquad (23)$$

where

$$\ddot{S}_a = -gk_t + \sqrt{a\dot{S}^4 + 2b\dot{S}^2 + c}$$

$$\ddot{S}_d = -gk_t - \sqrt{a\dot{S}^4 + 2b\dot{S}^2 + c}$$

and

$$a = \kappa^2 \left(\mu^2 n_r^2 - n_q^2 \right)$$

$$b = g\kappa \left(\mu^2 k_r n_r - k_q n_q \right)$$

$$c = g^2 \left(\mu^2 k_r^2 - k_q^2 \right).$$

Note that the maximum acceleration is not necessarily positive, nor is the maximum deceleration always negative; this depends on k_t, which represents the forward slope of the path. In addition to limits on the acceleration, (23) also expresses limits on the feasible vehicle speeds along the path. The argument under the square root should be positive; otherwise, (22) cannot be satisfied (the acceleration must be a real number to retain its physical meaning)

$$A = a\dot{S}^4 + 2b\dot{S}^2 + c \geq 0. \tag{24}$$

The feasible speed range is determined by the roots of (24). Only the positive roots are of interest since we assume forward motion. Depending on the values of the coefficients a, b, and c, it is possible to obtain a feasible speed range that does not include $\dot{S} = 0$. In such a case, the vehicle is statically unstable due to surface characteristics (traction and geometry).

Plotting the velocity limits for all points along the path creates a velocity limit curve in the phase plane S-S, representing the upper bound on vehicle speeds above which the constraint is violated. To ensure vehicle stability for the entire speed range, we consider only the lower feasible regions, excluding feasible islands at high speeds for which the motion is possible.

C. Contact Constraint

The speeds of autonomous vehicles moving on bumpy roads need to be carefully selected in order to maintain continuous contact with ground. Vehicles bouncing off the ground can easily tip over or loose their absolute position. A positive contact with ground is guaranteed if the normal force R applied on the vehicle is positive. Setting $R = 0$ in (14), we obtain the maximum speed allowed by the contact constraint

$$\dot{S} \leq \sqrt{-\frac{gk_r}{\kappa n_r}}. \tag{25}$$

The projection of r on k, $k_{,}$, is always positive; otherwise, the vehicle moves along an inverted slope. Equation (25) therefore applies only to cases where n_r is negative. This occurs when path curvature points opposite to the direction of surface normal r. Considering the reaction forces at each contact point would result in more conservative velocity limits than the ones expressed in (25).

D. Tip-Over Constraint

The tip-over constraint is obtained by expressing the condition for which the vehicle is about to tip over in terms of $\dot{S}, \ddot{S}$, and terrain topography. The limiting case for this constraint occurs when the entire weight shifts to one side of the vehicle, and the wheels on the other side are about to lose contact with ground. Referring to Fig. 4, if the vehicle is about to tip over counter clockwise (CCW), then the total reaction force R is applied on the left wheel, and the friction force f_y points downward, as shown (the friction force can be applied in the other direction, but then the vehicle will not tip over CCW). The vehicle will *not* tip over CCW if the resultant moment created by the reaction and the friction forces around its center of mass is
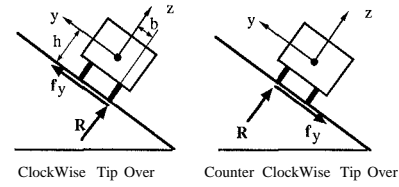

Fig. 4.   Tip over constraint.

positive (note that a positive f_y points in the positive y direction)

$$f_y h + Rb \geq 0 \tag{26}$$

where h and b are defined in Fig. 4 and R is defined in (14). The condition for *not* tipping over CCW is therefore

$$f_y \geq -R\frac{b}{h}. \tag{27}$$

The condition for not tipping over in the clockwise direction is

$$f_y \leq R\frac{b}{h}. \tag{28}$$

Equations (27) and (28) can be combined to one constraint equation

$$f_y^2 \leq \left(R\frac{b}{h} \right)^2. \tag{29}$$

Substituting (14) and (18) into (29) transforms the tip over constraint to quadratic inequality equation in S and S

$$a_1 \ddot{S}^2 + 2\left(a_2 \dot{S}^2 + a_3 \right) \ddot{S} + a_4 \dot{S}^4 + 2a_5 \dot{S}^2 + a_6 \leq 0 \tag{30}$$

where

$$a_1 = \sin^2 \theta$$

$$a_2 = \kappa n_q \sin \theta \cos \theta$$

$$a_3 = g \sin \theta \left(k_t \sin \theta + k_q \cos \theta \right)$$

$$a_4 = \kappa^2 \left(n_q^2 \cos^2 \theta - \beta^2 n_r^2 \right)$$

$$a_5 = \kappa g \left(n_q \cos \theta \left(k_t \sin \theta + k_q \cos \theta \right) - \beta^2 n_r k_r \right)$$

$$a_6 = g^2 \left[\left(k_t \sin \theta + k_q \cos \theta \right)^2 - \beta^2 k_r^2 \right]$$

and $\beta = b/h$. Equation (30) can be solved for the feasible acceleration as a function of S. Similarly to the sliding constraint, the limits on $\dot{S}$ are derived by satisfying the inequality

$$A = \left(a_2^2 - a_1 a_4 \right) \dot{S}^4 + 2\left(a_2 a_3 - a_1 a_5 \right) \dot{S}^2 + a_3^2 - a_1 a_6 \geq 0. \tag{31}$$

This inequality defines another velocity limit curve that must be satisfied to ensure safe vehicle motions.

E. Velocity Limit Curve

The feasible acceleration is determined by the intersection of the feasible acceleration ranges given in (20), (23), and (30). In addition, vehicle speeds should not exceed the limits expressed in (24), (25), and (31). The feasible speeds form a region that is a subset of the region formed by the intersection of the feasible regions of the individual constraints, considered separately. This creates a unified velocity limit curve below the union of the
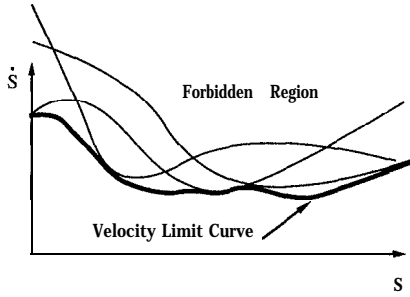
Fig. 5. A unified velocity limit curve.

individual curves, as shown schematically in Fig. 5. The velocity limit curve represents a unified mapping of the various dynamic constraints to the phase plane S-$\dot{S}$. It allows planning safe vehicle motions simply by ensuring that vehicle speeds stay below the velocity limit. Staying below the limit curve does not, however, guarantee that a given velocity profile is feasible. Staying below the time optimal velocity profile does.

V. TERRAIN AND PATH REPRESENTATIONS

Terrain representation is central to determining vehicle dynamic stability. A continuous representation is essential as it allows for generating directly smooth paths for which the gradient, tangent, and normal vectors at every point can be computed. We use B-spline patches to represent the terrain and B-spline curves to represent the path. B-spline patches can be generated from a topographic map or specified interactively using computer graphics interfaces.

A. Surface Representation by B Spline Patches

A B-spline patch is a parametric surface made of a mesh of B splines. A typical point q on a single patch in a three-dimensional space is a function of two parameters u and w [12].

$$q(v, w) = VMRM^T W^T \qquad (32)$$

where,

$$V = [v^3 v^2 v1], \qquad v = [0, 1]$$

and

$$w = [w^3 w^3 w1], \qquad w = [0, 1].$$

M is the 4 x 4 matrix specifying the type of spline used to construct the patch, and R is the 4 x 4 matrix of 16 control points. Each control point has an xy location, usually defined as the grid point on a uniform mesh, and a z value that represents its height. A large surface can be composed by several patches.

B. Path Representation

A smooth curve on the surface is obtained by parameterizing v and w with a single parameter u

$$p(u) = V(u)MRM^T W^T(u). \qquad (33)$$

A line in the U-W space is mapped to a continuous curve on the B patch. The line in the u-w space is represented by the B spline

$$\begin{bmatrix} v(u) \\ w(u) \end{bmatrix} = UMR_c \qquad (34)$$

where

$$U = [u^3 u^2 u1], \qquad u = [0, 1].$$

R_c is a 4 x 4 x 2 array of the control points in the u-w space, and u is the independent parameter along the B spline. B splines are second-order continuous, which is found to be essential for time-optimal paths. We chose to represent the curve by a B spline, but any other continuous representation will suffice as long as it provides a finite set of parameters for each curve. The path is initially obtained as a function of the parameter u; then it is rescaled in terms of the distance along the path S.

C. Obstacles and Surface Mobility Representations

Obstacles and areas with different surface quality are represented by rectangle primitives in the u-w space, mapped to the B-patch surface using (32). A point is determined to be inside an obstacle simply by comparing its u and w values to those of the obstacle. We restrict the obstacle-avoidance problem to gross motions, assuming large obstacles and wide pathways compared to vehicle size. Collision is detected simply by testing the collision of the contact points. Collision avoidance is achieved by adding a penalty to the cost function, as explained later.

To account for surfaces with various cultural types or tractions, we represent surface characteristics with a mobility factor $\eta = [0, 1]$. A mobility of one is assigned to paved roads, and zero to areas not accessible to the vehicle, such as large obstacles, lakes, or dangerous zones. A mobility value between zero and one is assigned to areas where vehicle motions are restricted and slowed down due to surface quality and other obstructions such as mud, sand, trees, small stones, and mines. The mobility factor η multiplies the original friction coefficient μ. The resulting effect is to lower the limit curve, reducing the maximum average speed allowed in a specific region.

VI. PATH OPTIMIZATION

Distance is the simplest cost function that can be obtained with a two-dimensional graph search. For more complex cost functions, such as motion time or energy, the search must be done in three dimensions (position and speed). To avoid a three-dimensional search, the problem is divided into two smaller problems: 1) a one-dimensional optimization of the motions along a specified path, and 2) a search for the optimal path in the position space. This separation of the problems makes the approach computationally practical for a variety of cost functions and constraints. It has been used for optimizing the motions of realistic six-dimensional robotic manipulators [16],[17].

A. Time Optimal Motions Along Specified Paths

The optimization along a given path follows an algorithm originally developed for minimizing motion times of robotic manipulators, subject to actuator constraints and the full nonlinear manipulator dynamics [1]. This algorithm has been extended to include general constraints such as the limitation on the gripping force and the maximum tolerable acceleration of the payload [15], [17]. The transformation of the dynamic constraints to velocity limit curves follows the basic principles presented in [1] and [17].

Following the time-optimization algorithm, vehicle motions are optimized by maximizing the acceleration or deceleration at every point along the path, while avoiding the limit curves [1]. The maximum acceleration or deceleration at every point is obtained by the intersection of the feasible acceleration ranges defined in (20), (23), and (30). The main computational effort in this part is to compute the switching points between the accelera-
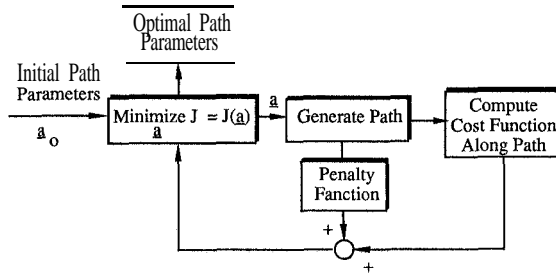
Fig. 6.   The parameter optimization procedure.

tion and deceleration. Efficient computation of the switching points is presented in [19]. The original algorithm was recently modified to account for singular points where the assumption of maximum acceleration and deceleration fails [18]. The reader is referred to [1], [17], and [18] for a full description of the method. The computation time of the time-optimal motions along a given path is of the order of less than 1 s on a Silicon Graphics IRIS-4D. The efficient computation of the motion time along the path makes this method suitable for evaluating the cost function in a local path optimization.

B. Local Optimization

The objective of the local optimization is to find the path with the minimum time, in the vicinity of an initial guess. The problem is formulated as a parameter optimization by representing smooth paths with a relatively small number of parameters, such as the control points of a B spline, using the motion time along the path as the cost function. The dynamic constraints, discussed earlier, are transparent to this optimization, allowing the use of unconstrained optimization procedures. The ability to consider constraints of various types in a unified way is one of the main advantages of this approach.

Using the control points as path parameters, the unconstrained optimization minimizes a cost function J through variations of path parameters $\underline{a}$:

$$\min_{\underline{a}} J = J(\underline{a}) \tag{35}$$

where the vector $\underline{a}$ represents the control points of the B spline in the u-w space, and J is any single valued cost function that can be evaluated for every path. The problem stated in (35) can be solved with standard parameter optimization methods [11]. A typical parameter optimization procedure is shown schematically in Fig. 6.

The lack of analytical relations between path parameters and the cost function restricts the use of gradient-type methods. Optimization procedures that compute the gradient by finite differences are suitable for smooth cost function over the entire parameter space. That is often difficult to achieve due to the discrete nature of the path representation and the approximations involved in computing the cost function. A pattern search method that does not require gradients has been most successful in the majority of the problems treated here [4]. Since the pattern search is essentially a heuristic method, the optimality of the solutions obtained cannot be guaranteed.

Constraints on path geometry due to obstacles or unsafe regions create voids in the solution space, requiring the use of constrained optimization methods. To retain the unconstrained nature of the problem, the objective function is augmented with a penalty value that adds a high cost for violating a constraint. It is convenient to use the number of infeasible points along the
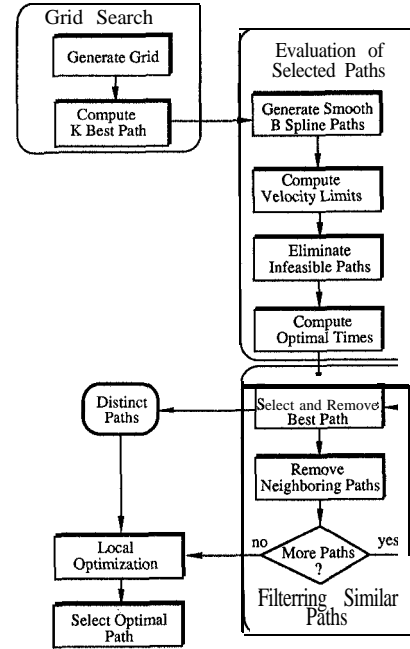

Fig. 7.   Global search for the time optimal path.

path to formulate the penalty function. The augmented cost function is

$$J_c = J(\underline{a}) + wn \tag{36}$$

where w is a weighting factor, and n is the number of infeasible points. A point along the path might be infeasible if the velocity limit is zero, the steering angle exceeds its limit, or the vehicle passes through an obstacle of zero mobility. This penalty value is zero if the path is completely feasible, and it increases gradually when the path penetrates into a forbidden region. The effect of the penalty function is to repel the path from infeasible regions. This penalty function is not differentiable, but neither is the cost function. The use of a pattern search reduces the sensitivity of the optimization to this problem.

C. Global Search for the Initial Path

The initial path for the local optimization is obtained with a global search, following the method presented in [16]. The algorithm, described schematically in Fig. 7, consists of a graph search in the position space for a set of paths that are likely to be in the vicinity of the global optimum. We briefly outline the method. The reader is referred to [16] for further details.

The position space is initially represented by a uniform grid in the u-w space from which all infeasible grid points due to the tip-over constraint and obstacles were eliminated. To avoid generating paths with tight turns, the grid is represented by a special graph that accounts for the directions of motion and departures from each grid point. The cost for each grid segment
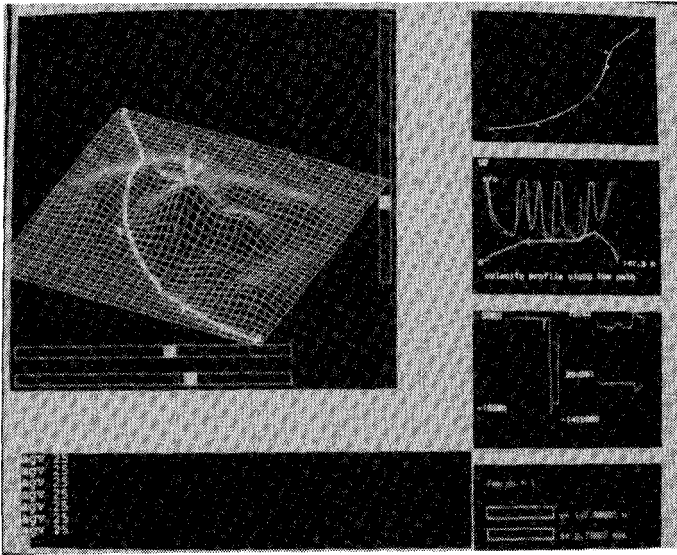
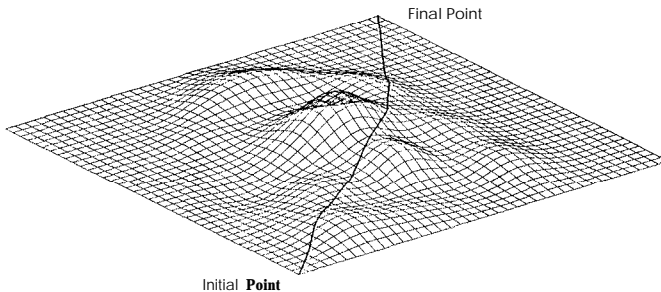Fig. 8.   The graphic display for interactive motion planning.
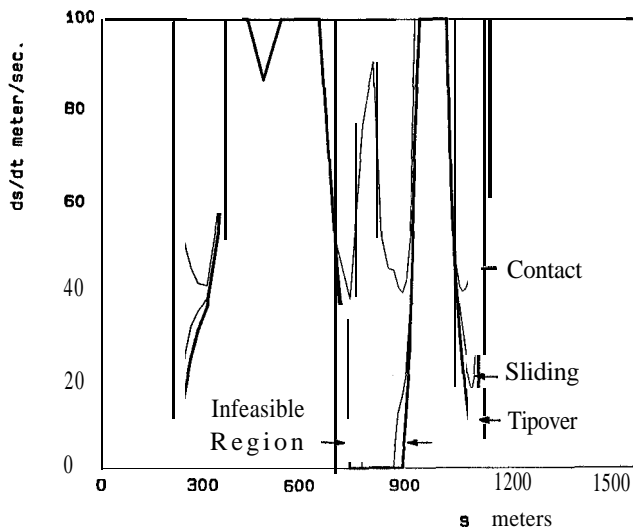

Fig. 9.   The shortest path.

Fig. 10.   The velocity limit curve for the shortest path.

consists of a lower bound estimate on the motion time, computed by integrating the velocity limit along the segment. To obtain at least one path in the vicinity of every potential local optimum, a large set of best paths are obtained along the grid, using the Dryfus search method for the K best paths [9]. These paths are smoothed by B splines, using the grid points as control points. The continuous limit curve is then computed for each path to test their feasibility due to the sliding or to the tip-over constraint at

zero speeds. Some paths become infeasible after the smoothing due to path curvature that is ignored along the grid.

The feasible paths are evaluated and filtered, based on their optimal motion time, to retain a smaller set of distinct paths. The filtering process consists of eliminating all paths within a tube of some radius D_{max} around the best path. This is repeated for the next best path, until all paths in the original set have been either selected to represent a distinct region or have been eliminated. The retained paths are assumed each to converge to a different potential local optimum in the local optimization. D_{max} is selected heuristically based on the grid size and the anticipated size of the convergence regions.

This method can be guaranteed to converge to the global optimum only if the grid size is sufficiently small and the number of paths in the initial set is sufficiently large. For more practical grid sizes, like the one used in the examples shown below, the global optimality of the solution is not guaranteed. However, the solutions obtained for the cases shown appear to be the global optimum. Further, this search discovered local optimal solutions that were not anticipated with manual selection of initial guesses.

VII. EXAMPLES

The method has been implemented in an interactive computer program on a Silicon Graphics IRIS-4D Workstation. User interfaces have been developed to invoke the various levels of the optimization: computation of the velocity limit and the optimal trajectory along a given path, a local optimization, and a global search for the best initial path. Fig. 8 shows a typical graphics display of the program. The main window displays the terrain, the path, and the obstacles, colored according to their mobility factors. The top right window displays the B spline in the u-w space. The user can modify the path by moving the control points in this window. The next window displays the velocity limit curves and the time-optimal trajectory (if the path is feasible). The third window displays data on the required force, energy, and friction force. The lower window displays the distance and motion time for interactive evaluation and optimization of the path.

The following examples were generated by this system, using a vehicle with the parameters given in Table I.

A. Minimum Distance Path

In the first example, the distance between two points on the terrain was minimized. Setting the cost function as the path distance, the global search, followed by the local optimization, obtained the shortest path shown in Fig. 9, with a total distance of 1460 m. Fig. 10 shows the velocity limit curves for this path, from which it is evident that the path is infeasible. The infeasible region along the path corresponds to the section where the velocity limit is zero. This occurs due to the steep slopes at the center of the terrain where the sliding and the tip over constraints are violated.

B. Time-Optimal Motions with No Obstacles

The next example obtains the time-optimal path between the same points as in the previous example. Using a grid resolution of 10 x 10, the first 240 paths found by the global search were considered. The filtering procedure resulted in six distinct paths; all are shown in Fig. 11. Each of these paths represents the best path within a tube of 200-m radius. The best three local optimal paths obtained by the local optimization are shown in Fig. 12,

TABLE I
VEHICLE PARAMETERS

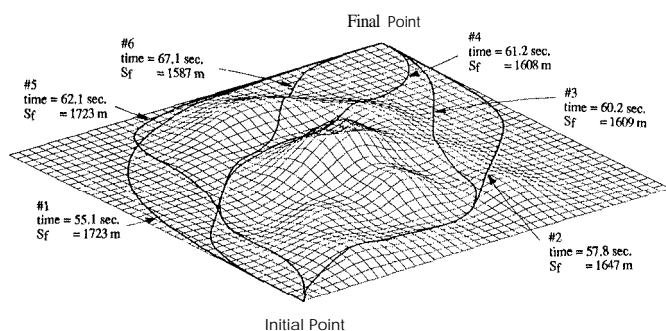|			
Mass	= 2000 kg	Length (2d)	= 4 m
Max Engine Force	= 3000 N	Aspect Ratio (1/β)	= 2
Max Braking Force	= -15 000 N	Max. Steering Angle	= 300^0
Friction Coefficient	= 0.7	Terrain Scale	= 1000 × 1000 m

Fig. 11.　Near-optimal paths selected by the global search after filtering.
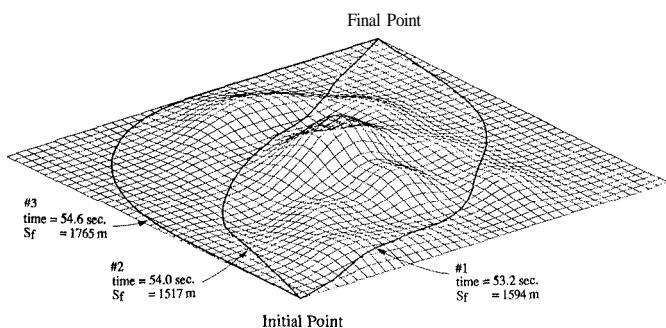

Fig. 12.　Three local time optimal paths.

with motion times of 53.2, 54.0, and 54.6 s. Some of the filtered paths converged to the same local optimum (paths 2 and 3, and paths 1 and 5). Fig. 13 shows the velocity limit curves and the optimal trajectory for the best path. Here, the tip-over constraint dominates the limit curves due to the high aspect ratio. In the current implementation, we use the lowest limit curve to construct the optimal trajectory, neglecting the coupling between the constraints.

Computation time for the global search was about 20 s, including the generation of the cost along the grid, and the search for the best 240 paths. A typical computation of an optimal trajectory along a given path was less then 1 s, including the graphic display of the terrain, the path, and the optimal trajectory. Filtering took less than 1 s, and the local optimization for a typical path with 10 control points required about 15 min. The computation of a global optimal path required a total computation time in the order of 1 h.

C. Time-Optimal Motions with Obstacles

In this example, obstacles with zero mobility were added to the original terrain. The obstacles were placed to block the local optimal paths obtained in the previous example in order to force the optimization to find other solutions. Here too, the best 240 paths obtained by the global search were selected and filtered. The best three paths obtained by the local optimization are shown in Fig. 14, with motion times of 57.4, 65.0, and 73.9 s. All three paths are quite different from the time-optimal paths
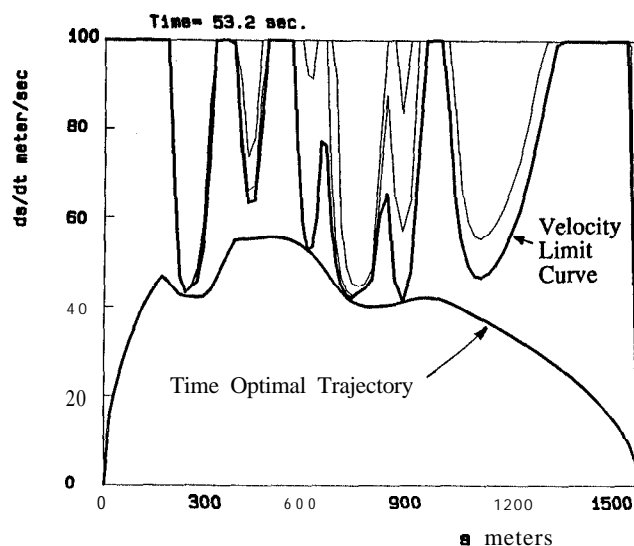

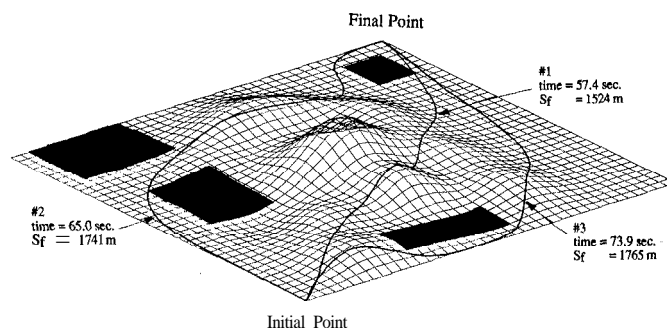Fig. 13.　The time optimal trajectory for the global optimal path.

Fig. 14.　Time optimal paths with obstacles.

obtained with no obstacles. Path 1 is the global optimum for this case. In fact, it is close to a local optimal path, obtained with no obstacles. This local optimum was not found in the previous example because of its relatively high motion time. Paths 2 and 3 are variations of paths 2 and 1 of the previous example, modified to avoid the obstacles.

VIII. CONCLUSIONS

A method for planning optimal vehicle motions on general terrains has been presented. It accounts for vehicle kinematics, dynamics, terrain topography, obstacles, and regions with various mobility. The terrain and the path are represented by B-spline patches and curves, allowing general geometric shapes. The method obtains the path and vehicle speeds that minimize motion time, subject to engine torque limits, a given coefficient of friction, and vehicle kinematic and dynamic parameters.

For a given path, constraints on vehicle motions, such as sliding, tip over, and contact, are transformed to velocity limit curves, providing upper bounds on vehicle speeds. The optimal motions along the path are obtained by a previously developed

method that maximizes the acceleration or deceleration at all points, while avoiding the limit curve. A local optimal path is obtained by a parameter optimization procedure, optimizing over the control points of the B spline, using the motion time as the cost function. The initial guess to the local optimization is obtained by a global graph search. The approach is general and can be extended to include cost functions other than distance and time, for instance, energy consumption, minimum threat, etc. in military as well as civilian applications.

The method has been implemented in an interactive computer program on a Silicon Graphics IRIS-4D workstation. Examples are presented that demonstrate the approach for planning the motions of a rolling vehicle moving along a mountainous terrain, with and without obstacles. For the case shown, the time-optimal path was found to be significantly different from the unsafe minimum distance path, demonstrating the importance of terrain topography and vehicle dynamics. Obstacle avoidance was demonstrated by placing obstacles to block the unconstrained local optimal solutions. The best solution obtained by the grid search was found always to be in the vicinity of a local optimal path, but not necessarily the global optimum. Computation time for a full optimization was in the order of 1 h, spending most of the time on the local optimization. The computation time can be reduced by limiting the resolution of the map and the path, or by increasing the grid resolution and avoiding the local optimization.

A simple dynamic model is used to demonstrate the approach. Except for the tip over constraint, the reaction and friction forces are translated to the mass center to avoid solving explicitly for the reaction forces at the contact points. Exact models of rolling vehicles are generally complex due to the indeterminate nature of the problem and are beyond the scope of this paper. Future work will focus on formulating the dynamic constraints treated here with more realistic models. This method can be used in other applications, such as determining safe speeds for motor vehicles, using the time-optimal velocity profile along the traveled path as an upper bound for the actual vehicle speeds.

REFERENCES

[1] J. E., Bobrow, S. Dubowsky, and J. S. Gibson, "Time-optimal control of robotic manipulators," *Int. J. Robotics Res.,* vol. *4, no. 3, 1985.*

[2] D. Gaw and A. Meystel, "Minimum-time navigation of an unmanned mobile robot in a 2-l /2D world with obstacles," in *Proc. IEEE Int. Conf. Robotics Automat.,* vol. *3, 1986,* pp. 1670-1677.

[3] T. J. Greattinger and B. H. Krogh, "Evaluation and time scaling of trajectories for wheeled mobile robots," *ASME J. Dynamic Syst., Measurement, Control,* vol. 111, pp. 222-231, June 1989.

[4] R. Hooke and T. A. Jeeves, "Direct search solution of numerical and statistical problems," *J. Assoc. Comput. J., vol. 3,* pp. 212-229, 1961.

[5] P. Jacobs and J. Canny, "Planning smooth paths for mobile robots," in *Proc. IEEE Int. Conf. Robotics Automat.,* (Scottsdale, AZ), May 1989, pp. 2-7.

[6] Y. Kanayama, and B. I. Hartman, "Smooth local path planning for autonomous vehicles," in *Proc. IEEE Int. Conf. Robotics Automat.,* (Scottsdale, AZ), May 1989, pp. 1265-1270.

[7] Y. Kanayama and S. I. Yuta "Vehicle path specification by a sequence of straight lines," *IEEE J. Robotics Automat.,* vol. *4, no. 3,* pp. *265-276,* June 1988.

[8] V. Kumar and K. J. Waldron, "Actively coordinated vehicle systems," *ASME J. Mechanisms, Transmissions, Automat. Design,* vol. 111, pp. 223-231. June *1989.*

[9] E. L. Lawler, *Combinatorial Optimization: Networks and Matroids.* New York: Holt, Rinehart and Winston, 1976, pp. 98- 100.

[10] Z. Li and J. Canny, "Motion of two robotic bodies with rolling constraint," *IEEE Trans. Robotics Automat.,* vol. *6,* no. 1, pp. *62-72,* Feb. 1990.

[11] D. G. Luenberger, *Linear and Nonlinear Programming,* Reading, MA: Addison-Wesley, 1984.

[12] M. E. Mortenson, *Geometric Modeling. New* York: Wiley, 1985.

[13] W. Nelson, "Continuous-curvature paths for autonomous vehi-cles," in *Proc. IEEE Int. Conf. Robotics Automat.* (Scotts-dale, AZ), May 1989, vv. 1260-1264.

[14] A. M. Parodi, "Multi-goal real-time global path planning for an autonomous land vehicle using a high-speed graph search proces-sor," *IEEE Int. Conf. Robotics Automat.,* vol. 1, 1985, pp. 161-167.

[15] Z. Shiller and S. Dubowsky, "On the optimal control of robotic manipulators with actuator and end-effector constraints," in *Proc. IEEE Int. Conf. Robotics Automat.* (St. Louis, MO), Mar. 1985, pp. 614-620.

[16] "Global time optimal motions of robotic manipulators in the presence of obstacles," in *Proc. IEEE Conf. Robotics Au-tomat.* (Philadelphia, PA), Avr. 1988, pp. 370-375.

[17] "Time optimal path planning for robotic manipulators in the presence of obstacles, with actuator, gripper and payload con-straints," *Int. J. Robotics Res.,* vol. 8, no. 6, pp. 3-18, Dec. 1989.

[18] Z. Shiller and H. H. Lu, "Robust computation of path con-strained time optimal motions." *IEEE Conf. Robotics Au-tomat.* (Cincinnati, OH), May '1990, pp. 144–149.

[19] J-J. E. Slotine and H. S. Yang, "Improving the efficiency of time optimal path following algorithms," *IEEE Trans. Robotics Automat.,* vol. 5, no. 1, pp. 118-124, Feb. 1989.
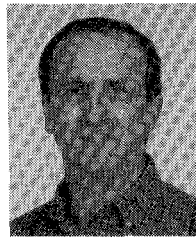
Zvi Shiller (S'85-M'87) received the B.Sc. degree from Tel-Aviv University, Tel-Aviv, Is-rael, in 1976, and the M.S. and Sc.D. degrees from the Massachusetts Institute of Technology, Cambridge, in 1984 and 1987, respectively, all in mechanical engineering.

Since 1987, he has been an Assistant Profes-sor in the Mechanical, Aerospace and Nuclear Engineering Department at the University of California, Los Angeles. Between 1978 and 1982, he worked as a Design Engineer for the Israeli Aircraft Industries. His research interests include motion plan-ning and control of computer-coordinated mechanical systems, multi-body dynamics, robotics, and autonomous vehicles.

Yu-Rwei Gwo received the B.Sc. degree from National Taiwan University, Taiwan, in 1985 and the M.S. degree from University of California, Los Angeles, in 1990, both in me-chanical engineering. His research interests in-clude mobile robots, obstacle avoidance, and artificial intelligence.