Smooth Local Path Planning for Autonomous Vehicles

Yutaka Kanayama and Bruce I. Hartman

Community and Organization Research Institute
Department of Computer Science
University of California
Santa Barbara, CA 93106

Abstract

Two cost functions of paths for smoothness are defined; Path curvature and the derivative of path curvature. Through these definitions, two classes of simple paths are obtained; the set of circular arcs and the set of cubic spirals. A cubic spiral is a curve whose tangent direction is described by a cubic function of path distance s. These sets of simple paths are used for solving path planning problems of symmetric posture (position and orientation) pairs. For a non-symmetric posture pair, we use two simple paths as a solution. In order to find those paths, we use the fact that the locus of split postures is a circle or a straight line. A posture q is said to be a split posture of a pair (p_1, p_2) of postures, if p_1 and q are symmetric and so are q and p_2. The resultant solutions are smoother than those obtained by one of the authors using clothoid curves. This algorithm has been successfully implemented on the autonomous mobile robot Yamabico-11 at UCSB.

1. Introduction

We proposed a robot's path description method in which a path is described by a sequence of postures (a posture is a position with an orientation) instead of a sequence of curve segments [1]. For instance, a path shown in Figure 1 is described by a sequence $(p_1, p_2, \cdots, p_8)$ of eight postures. In this method, we are to find each path segment which is determined solely by two endpostures. Thus, in Figure 1, we are to solve a local path-finding problem for pairs of postures (p_1, p_2), ... , (p_7, p_8) independently. This paper describes a method of finding the "smoothest" local path joining a given pair of postures.

Smoothness of paths is essential for mobile robot navigation, because unsmooth motions may cause slippage of wheels which degrades the robot's deadreckoning ability. In order to control "smoothness" of paths, we propose to define the cost of a path for smoothness. A unit cost for smoothness at a point on a path is proposed as (1) square of its curvature, or (2) square of the derivative of its curvature in this paper. The total cost of a path is taken as the integration of either of these incremental costs. Using calculus of variations, we obtained circles, or a new class of curves which we call *cubic spirals* for solutions. The class of cubic spirals has an advantage that the curvature is continuous at each specified posture, because the curvature is null there. The curvature continuity is essential for vehicle path design.

When navigating itself, an autonomous vehicle has only two degrees of freedom; velocity and curvature. Therefore, if we focus only on the static features of robot's paths, a path can be naturally described by its curvature $\kappa(s)$ of the traveling distance s. (The traveling distance is the robot's only coordinate.) This is the reason why we do not adopt spline curves in formalizing this smooth path planning theory. It is not easy for the robot to control (x, y) in a Cartesian coordinate system.

Komoriya and Hongo independently adopted a description method using a sequence of straight lines and circular arcs [2][3] and so do many other researchers. This paper gives a mathematical interpretation to their work; if we take curvature itself as the cost of a path, the solutions are circles and straight lines. However, one major problem in the method is that the curvature of a generated path is not necessarily continuous and thus, strictly speaking, a vehicle is not able to follow them smoothly [4]. The use of clothoid curves, or Cornu spirals, was proposed and implemented on Stanford "mobi" [5]. This class of curves has linear curvature change and has been used in highway and railway curve design. The paths have curvature continuity. Our results show that cubic spirals are better in terms of the maximum curvature.

This algorithm has been successfully implemented on an autonomous mobile robot Yamabico-11 which has been developed at the University of California at Santa Barbara. This local path planning function is expressed in the form of elementary *move* and *stop* functions integrated into a higher-

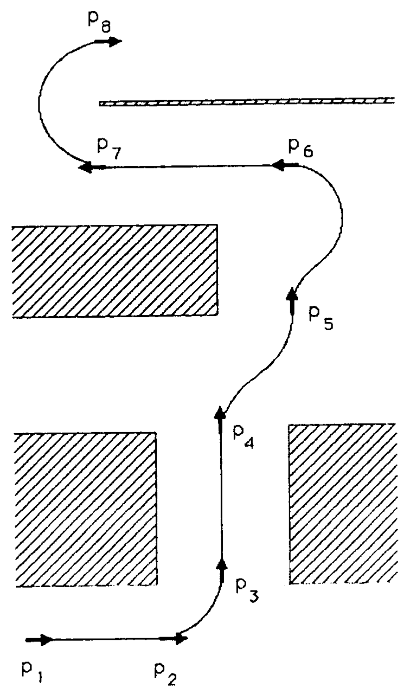

Fig. 1 Path Specification by Postures

CH2750-8/89/0000/1265$01.00 © 1989 IEEE

level mobile robot language, MML, a dialect of the C language [6][7][8][9]. For complete proofs for some Propositions, refer to [10].

2. Overview of the Method

We define a 2D global Cartesian coordinate system in a world for a mobile robot. Let us call a combination of position and orientation (x, y, θ) a *posture*. The direction θ is taken counter-clockwise from the X-axis. Any vehicle possesses this three degrees of freedom in its positioning. If $p = (x, y, \theta)$ is a posture, "point p" stands for the point (x, y). Let $p_1 = (x_1, y_1, \theta_1)$ and $p_2 = (x_2, y_2, \theta_2)$ be two postures with $(x_1, y_1) \neq (x_2, y_2)$. The orientation β from the point (x_1, y_1) to (x_2, y_2) is (Fig. 2),

$$\beta = \tan^{-1}\left(\frac{y_2 - y_1}{x_2 - x_1}\right) \tag{1}$$

The pair of two postures p_1 and p_2 are said to be *symmetric* if and only if the the following relation is satisfied:

$$\theta_1 - \beta = -(\theta_2 - \beta) \tag{2}$$

We write $sym(p_1, p_2)$ if p_1 and p_2 are symmetric. For instance, in Figure 1, $sym(p_1, p_2)$, $sym(p_2, p_3)$, $sym(p_3, p_4)$, $sym(p_6, p_7)$, and $sym(p_7, p_8)$. But $\bar{\ }sym(p_4, p_5)$ and $\bar{\ }sym(p_5, p_6)$, where a $\bar{\ }$ symbol means negation. If $sym(p_1, p_2)$ and there is an intersection p_0 of the "forward ray" of p_1 and the "backward ray" of p_2 (or if there is an intersection p_0 of the "backward ray" of p_1 and "forward ray" of p_2), the triangle $p_0 p_1 p_2$ is equilateral (Fig. 2). However, in some cases, there is no such an intersection associated with a symmetric posture pair, for instance, (p_1, p_2) and (p_7, p_8) in Figure 1.

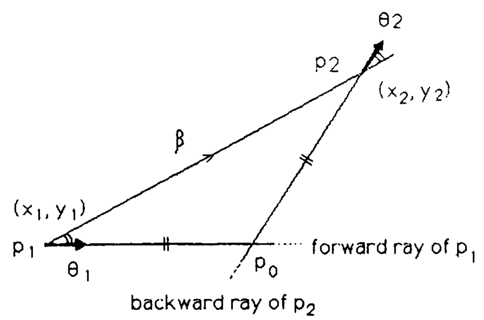

Fig. 2 Symmetric Postures

A *size d* and an *deflection* α of a pair (p_1, p_2) of a symmetric posture pair are defined as the Euclidean distance of the two points and the angle between the directions of its postures respectively:

$$d = ((x_2 - x_1)^2 + (y_2 - y_1)^2)^{1/2} \tag{3}$$

$$\alpha = \theta_2 - \theta_1 \tag{4}$$

A posture q is said to be a *split posture* of a pair of postures (p_1, p_2) if $sym(p_1, q)$ and $sym(q, p_2)$.

We assume that the tangent direction of a path Π (with a finite length) is defined everywhere on the path. For each point (x, y) on a path Π, we associate a posture (x, y, θ), where θ is a tangent direction of Π at the point. As a special case, for an endpoint of the path, we associate an endposture. A path (with a finite length) is said to be *simple* if its endpostures are symmetric. For instance, every circular arc is simple (cf. Proposition 1).

In our method of finding solutions, the following preparation steps are required: Define a cost function for smoothness of paths. Thus, for each path Π, its cost $cost(\Pi)$ is explicitly determined. Through this definition of a cost, we define a set S of simple paths, in which there is one and only one simple path with a given size d and a deflection α. After this preparation is done, we are ready to describe the local path planning method when being given two postures $p_1 = (x_1, y_1, \theta_1)$ and $p_2 = (x_2, y_2, \theta_2)$, where $(x_1, y_1) \neq (x_2, y_2)$.

The Algorithm

(Case I) If $sym(p_1, p_2)$, find a path in S with the same size and deflection of (p_1, p_2) and make appropriate translation and rotation to fit the pair of end-postures. This resultant simple path is expressed as $\Pi(p_1, p_2)$. (Notice that the size and deflection completely characterize a simple path.)

(Case II) If $\bar{\ }sym(p_1, p_2)$, first, find the locus of split postures of (p_1, p_2). Next we find a split posture q which makes the sum of costs, $\mathrm{cost}(\Pi(p_1, q)) + \mathrm{cost}(\Pi(q, p_2))$, minimum.

3. Path Representation

Let us define the representation of closed paths with finite length. A path Π is a pair (l, κ) where l is a positive *length*, and κ is a continuous curvature function: $[-l/2, l/2] \to R$ of Π, where R is the set of real numbers. We stipulate that $x(0) = y(0) = \theta(0) = 0$. That is, we are adopting a standard positioning convention of paths. The tangent direction function θ is given by,

$$\theta(s) = \int_0^s \kappa(t)dt \quad -l/2 \leq s \leq l/2 \tag{5}$$

Therefore, a point $(x, y) = (x(s), y(s))$ on Π is given by,

$$\begin{cases} x = x(s) = \int_0^s \cos\theta(t)\, dt \\ y = y(s) = \int_0^s \sin\theta(t)\, dt \end{cases} \tag{6}$$

The *endpostures* of the path Π is $(x(-l/2), y(-l/2), \theta(-l/2))$ and $(x(l/2), y(l/2), \theta(l/2))$. The following Proposition relates the "symmetric" property of the curvature of a path, and the symmetric property of endpostures of the path.

Proposition 1. If a curvature function κ is "symmetric", namely, if $\kappa(-s) = \kappa(s)$ holds, so do the following relations,

$$\begin{cases} \theta(s) = -\theta(s) \\ x(s) = -x(-s) \\ y(s) = y(-s) \end{cases} \tag{7}$$

for all s. Therefore, in a special case of $s = l/2$, its endpostures are symmetric.

4. Cost Functions and Complete Set of Simple Paths

Let us define two cost functions of paths for smoothness:

$$\mathrm{cost}_1(\Pi) = \int_0^l F_1 ds = \int_0^l \kappa^2 ds = \int_0^l (\dot{\theta})^2 ds \tag{8}$$

$$\mathrm{cost}_2(\Pi) = \int_0^l F_2 ds = \int_0^l (\dot{\kappa})^2 ds = \int_0^l (\ddot{\theta})^2 ds \tag{9}$$

Although we do not consider any dynamic properties of robot motion control, we can interpret these definitions from the viewpoint of dynamics. Assuming constant-velocity navigation of a vehicle along the path Π, the instantaneous centripetal acceleration of the vehicle is proportional to its curvature $\kappa(s)$. Notice that the radius of the osculating circle at $(x(s), y(s))$ is $r(s) = 1/\kappa(s)$. Therefore, in the first definition of Equation (8), we are taking the integration of (square of) acceleration as its path cost. In the definition of $cost_2$ of Equation (9), we take the integration of square of the *variation of acceleration* (or *jerk*) as a path cost. Since jerk is considered to be minimized for comfortable vehicle control, it would be rather reasonable to adopt this definition. Let us characterize classes of paths which make the path costs of Equations (8) and (9) minimum:

Proposition 2. For a fixed length l, the following path minimizes $\mathrm{cost}_1(\Pi)$ of Equation (8):

$$\begin{cases} \dot{\theta}(s) = A \equiv \kappa(s) \\ \theta(s) = As + B \end{cases} \tag{10}$$

where A and B are integral constants.

Proof. $F_1 = (\dot{\theta})^2$ by Equation (8). By a textbook on calculus of variations [11], we have

$$\frac{\partial F_1}{\partial \theta} - \frac{d}{ds}\left(\frac{\partial F_1}{\partial \dot{\theta}}\right) = 0$$

Therefore,

$$-\frac{d}{ds}(2\dot{\theta}) = -2\frac{d^2\theta}{ds^2} = 0$$

Equation (10) is obtained, by simply integrating this two times. □

Paths given by Equation (10) are merely circular arcs or straight lines. Since the curvature function is constant and is "symmetric", each circular arc is simple by Proposition 1. In this case, the set S of simple paths are the set of all circular arcs with various sizes and deflections.

Proposition 3. For a fixed length l, the following path minimizes $\mathrm{cost}_2(\Pi)$ of Equation (9):

$$\begin{cases} \dot{\theta}(s) = \frac{1}{2}As^2 + Bs + C \equiv \kappa(s) \\ \theta(s) = \frac{1}{6}As^3 + \frac{1}{2}Bs^2 + Cs + D \end{cases} \tag{11}$$

where A, B, C, and D are integral constants.

Proof. $F_2 = \ddot{\theta}^2$ by Equation (9). Similarly, we have

$$\frac{\partial F_2}{\partial \theta} - \frac{d}{ds}\left(\frac{\partial F_2}{\partial \dot{\theta}}\right) + \frac{d^2}{ds^2}\left(\frac{\partial F_2}{\partial \ddot{\theta}}\right) = 0$$

Therefore,

$$2\frac{d^2}{ds^2}\ddot{\theta} = 2\frac{d^4\theta}{ds^4} = 0$$

Equation (11) is obtained, by integrating this four times. □

These curves are more complex than circles. Let us call this class of curves *cubic spirals*, since the direction function θ is a cubic function. Figure 3 shows an example of curvature and tangent direction functions of a cubic spiral. Figure 4 shows an example of a whole cubic spiral. We will use only the part of a cubic spiral $[s_1, s_2]$ in Figure 3, between the two points where curvature is null. Since the curvature of this part is clearly "symmetric", the path is also simple. Hereafter, a *cubic spiral* refers to that finite simple path out of a whole spiral. Figure 5 shows a class of cubic spirals with unit length. These are said to be *standard* cubic spirals.

Once, the use of clothoid curves was proposed by the authors [5]. (Precisely speaking, *clothoid pairs* were used.) This class of curves also possesses the advantage of having curvature continuity (See Conclusion).

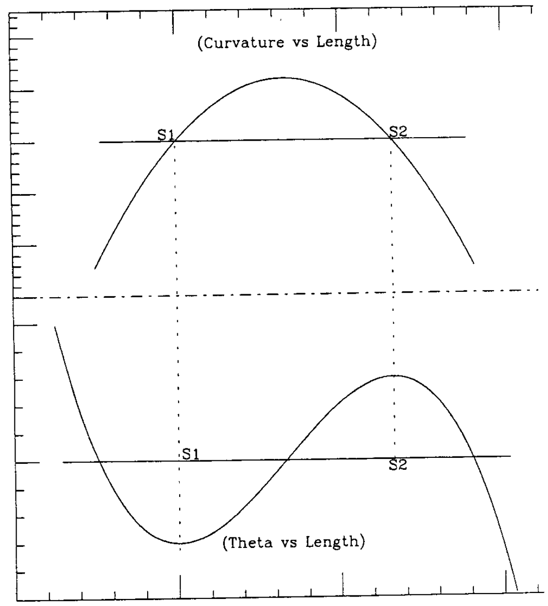

Fig. 3 Curvature and Tangent Direction of Cubic Spiral

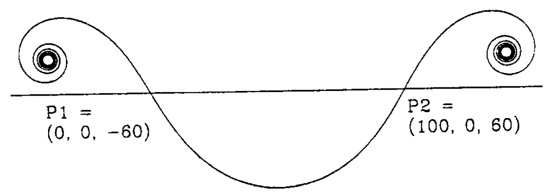

Fig. 4 A Whole Cubic Spiral

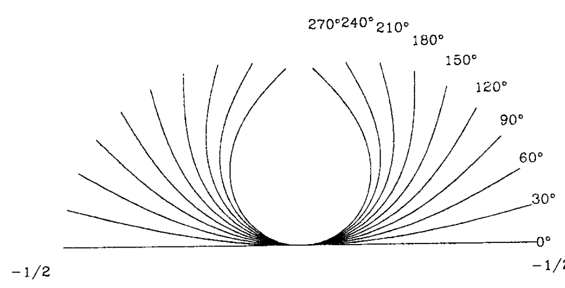

Fig. 5 Standard Cubic Spirals

5. Solving Symmetric Cases

In our smooth path planning problem, finding a simple path for a given symmetric posture pair (p_1, p_2) is the most elementary part (Section 2). In this case, the size d and the deflection α of (p_1, p_2) are given. The following Proposition gives a circle solution:

Proposition 4. Given the distance d and deflection α of a circular arc Π, its length, curvature and cost are:

$$l = \frac{\alpha/2}{\sin(\alpha/2)} d \tag{12}$$

$$\kappa(s) = \frac{2\sin(\alpha/2)}{d} \tag{13}$$

$$\mathrm{cost}_1(\Pi) = \frac{2\alpha\sin(\alpha/2)}{d} \tag{14}$$

In the cubic spiral case, we need to characterize the set of cubic spirals. By the definition of the cubic spiral, its curvature is,

$$\kappa(s) = c\left(\frac{l^2}{4} - s^2\right), \quad \text{for } s \in [-l/2, l/2] \tag{15}$$

where c is a constant. A cubic spiral with a unit length is called *standard*. Let $D(\alpha)$ denote the size of a standard cubic spiral with a deflection α. For instance, $D(0) = 1$, $D(\pi/2) = 0.8558$, and $D(\pi) = 0.4861$.

Proposition 5.

(I) All cubic spirals with the same deflection α are similar to each other.

(II) If the size d and deflection α of a cubic spiral Π is given, its length, curvature, and cost are:

$$l = \frac{d}{D(\alpha)} \tag{16}$$

$$\kappa(s) = \frac{6\alpha D(\alpha)^3}{d^3}\left(\frac{d^2}{4D(\alpha)^2} - s^2\right) \tag{17}$$

$$\mathrm{cost}_2(\Pi) = \frac{12\alpha^2 D(\alpha)^3}{d^3} \tag{18}$$

where θ_0 is a constant.

(III)

$$D(\alpha) = 2\int_0^{1/2} \cos\left(\alpha(3/2 - 2s^2)s\right) ds \tag{19}$$

Figures 6 and 7 show symmetric cubic spirals for a simple left-turn and u-turn, respectively. They also clarify the differences among solutions using circles, cubic spirals, and clothoids. The length of the circle is minimum while that of the clothoid is maximum. Maximum curvature of the cubic spiral is less than that of the clothoid curve. This is one of the advantages of cubic spirals. To compensate for this, however, we need to have a bigger $\dot{\kappa}$ at the start and end of the path.

6. Solving Non-Symmetric Cases

The next question is how to solve the problem if the input pair of postures $p_1 = (x_1, y_1, \theta_1)$ and $p_2 = (x_2, y_2, \theta_2)$ with $(x_1, y_1) \neq (x_2, y_2)$ is non-symmetric. The pair (p_1, p_2) is said to be *parallel* if and only if $\theta_1 = \theta_2$.

Proposition 6. The locus of points of split postures (x, y, θ) of an input posture pair $p_1 = (x_1, y_1, \theta_1)$ and $p_2 = (x_2, y_2, \theta_2)$ is:

(I) If (p_1, p_2) is parallel,

The line determined by the points p_1 and p_2:

$$(x - x_1)(y - y_2) = (x - x_2)(y - y_1) \tag{23}$$

(II) If (p_1, p_2) is not parallel,

The following circle which goes through the points p_1 and p_2:

$$\left((x - x_1)(x - x_2) + (y - y_1)(y - y_2)\right)\tan\left(\frac{\theta_2 - \theta_1}{2}\right) = (x - x_1)(y - y_2) - (x - x_2)(y - y_1) \tag{24}$$

The center p_c of the circle (24) is:

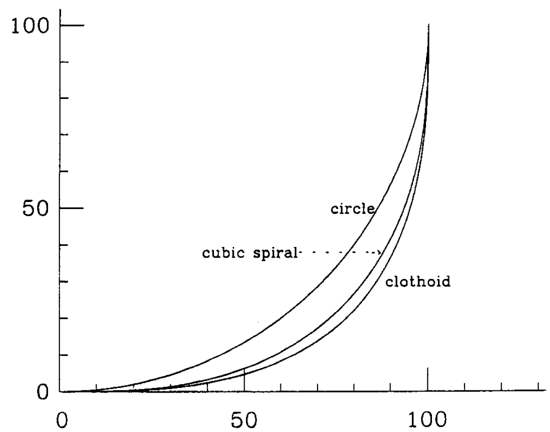

Fig. 6 Simple Curves with $\alpha = \pi/2$

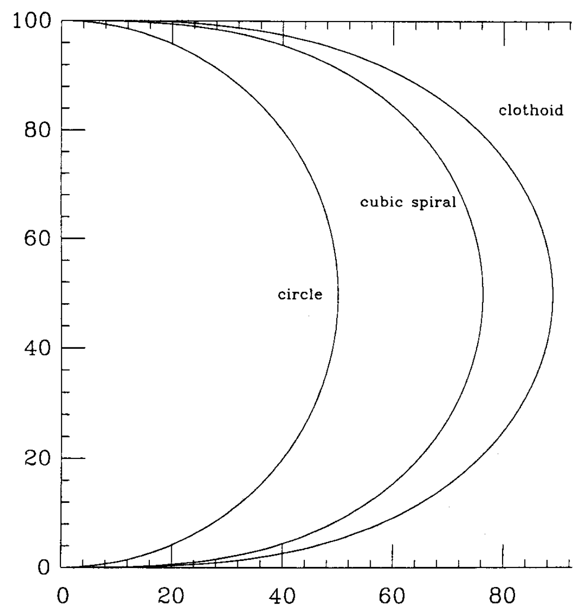

Fig. 7 Simple Curves with $\alpha = \pi$

$$p_c = (x_c, y_c) = \left(\frac{x_1+x_2+c(y_1-y_2)}{2}, \frac{y_1+y_2+c(x_2-x_1)}{2}\right), \quad (25)$$

where $c \equiv cot((\theta_2-\theta_1)/2)$.

Proof. By Equations (2),

$$\begin{cases} \dfrac{\theta_1+\theta}{2} = \tan^{-1}\left(\dfrac{y-y_1}{x-x_1}\right) \\ \dfrac{\theta+\theta_2}{2} = \tan^{-1}\left(\dfrac{y_2-y}{x_2-x}\right) \end{cases}$$

By taking the difference of both sides to cancel θ,

$$\frac{\theta_2-\theta_1}{2} = \tan^{-1}\left(\frac{y_2-y}{x_2-x}\right) - \tan^{-1}\left(\frac{y-y_1}{x-x_1}\right)$$

By applying the tan function to the both sides,

$$\tan\left(\frac{\theta_2-\theta_1}{2}\right) = \tan\left(\tan^{-1}\left(\frac{y_2-y}{x_2-x}\right) - \tan^{-1}\left(\frac{y-y_1}{x-x_1}\right)\right)$$

$$= \frac{\dfrac{y-y_2}{x-x_2} - \dfrac{y-y_1}{x-x_1}}{1+\left(\dfrac{y-y_2}{x-x_2}\right)\left(\dfrac{y-y_1}{x-x_1}\right)} = \frac{(x-x_1)(y-y_2)-(x-x_2)(y-y_1)}{(x-x_1)(x-x_2)+(y-y_1)(y-y_2)}$$

This is equal to Equation (24). In the parallel case, its left hand side is zero and Equation (23) is obtained. □

Figures 8 and 9 shows examples of the sets of split postures for a parallel and non-parallel cases respectively. In non-parallel cases, if $\alpha > 0$, three points p_c, p_1 and p_2 are placed counterclockwise (p_c is the center of the locus, see Figure 12); if $\alpha < 0$, three points p_c, p_1 and p_2 are placed clockwise.

In selecting a split point, we further stipulate the following in order to avoid impractical solutions:

(I) In parallel cases ($\alpha = 0$), we select a split point on the segment $\overline{p_1p_2}$.

(II) In case of $\alpha > 0$, we select a split point on the arc which is taken between p_1 and p_2 in counterclockwise order.

(III) In case of $\alpha < 0$, we select a split point on the arc which is taken between p_1 and p_2 in clockwise order.

Under these restrictions, we are to find a split posture q which makes the total of $\mathrm{cost}(\Pi(p_1, q)) + \mathrm{cost}(\Pi(q, p_2))$ minimum.

6.1. Parallel Cases

Let $\theta \equiv \theta_1 = \theta_2$.

Proposition 7. For either cost function $cost_1(\Pi)$ or $cost_2(\Pi)$, the least cost split point of $p_1 = (x_1, y_1, \theta)$ and $p_2 = (x_2, y_2, \theta)$ is

$$q = \left(\frac{x_1+x_2}{2}, \frac{y_1+y_2}{2}, \beta-(\theta-\beta)\right) \quad (26)$$

where $\beta = \tan^{-1}\left(\dfrac{y_2-y_1}{x_2-x_1}\right)$

This solution leads to a combination of two identical simple paths which are mirror images of each other. Figure 10 shows several resultant paths of parallel cases.

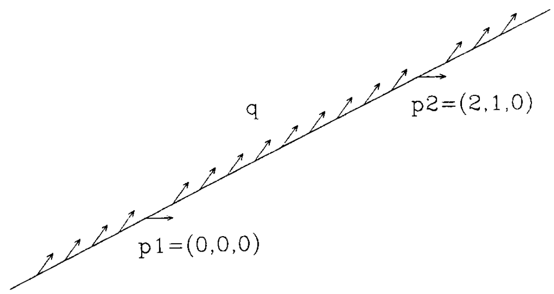

Fig. 8 Split Postures in Parallel Case

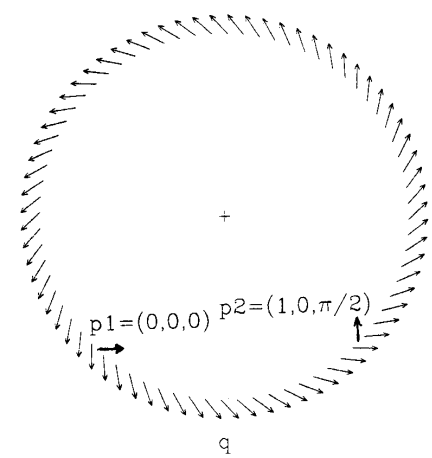

Fig. 9 Split Postures in Non-Parallel Case

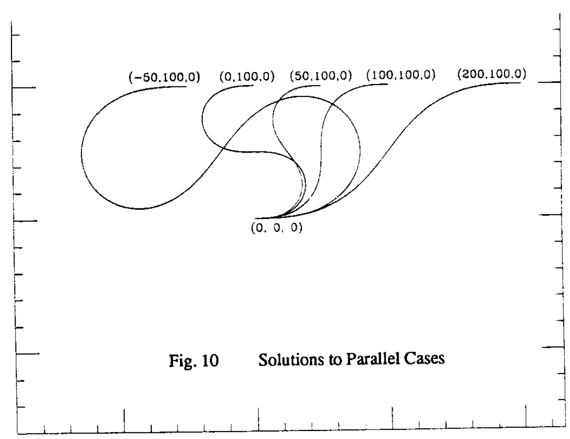

Fig. 10 Solutions to Parallel Cases

6.2. Non-Parallel Cases

In this case, the locus of split points is a circle of Equation (23).

Proposition 8. In a non-parallel case, there is a split point on the permissible arc of Equation (24) such that the sum of the costs of two cubic spirals is minimum.

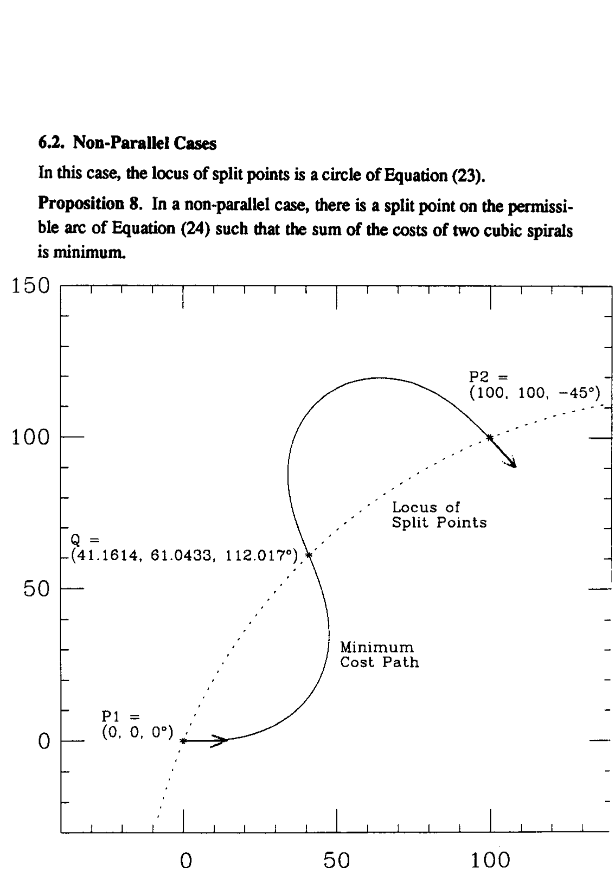

Fig. 11 Example of Non-Parallel Case

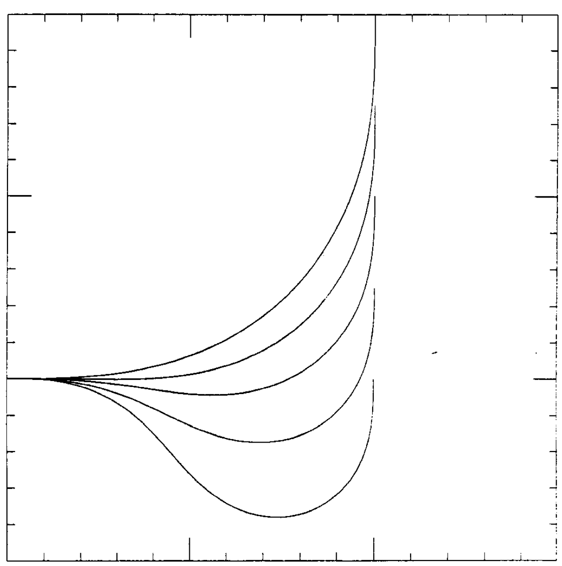

Fig. 12 Series of Non-Parallel Case Solutions

Proof. Since the total cost function is positive and continuous. □

Because no analytical method for finding the minimum solution has been found yet, we adopt an iterative method based on the Newton-Raphson algorithm. In Figure 11, the input postures are $p_1 = (0, 0, 0)$ and $p_2 = (100, 100, -\pi/4)$. This posture pair is neither symmetric nor parallel. The locus of possible split points is indicated by the broken line whose center is (136.603, −36.603), which is out of bounds of this figure. The cost is minimum at the split point Q shown in the figure. Figure 12 shows a sequence of non-parallel case solutions.

7. Conclusion

In this paper, two definitions of smoothness of vehicle paths is proposed; the cost of a path is its curvature or the derivative of its curvature. After implementing on our mobile robot Yamabico-11, we confirmed the vehicle moves far smoother than when the clothoid method is used. Let us compare the maximum curvature $\kappa(\alpha)$ at the middle point of a standard cubic spiral, and the maximun curvature $\kappa'(\alpha)$ at the middle point of a standard cubic spiral pair, with the same deflection α. Let R be their ratio:

$$R(\alpha) = \frac{\kappa(\alpha)}{\kappa'(\alpha)} = \frac{3}{4}\frac{D(\alpha)}{D'(\alpha)} \tag{27}$$

The functions $D(\alpha)$ is defined by Equation (19). $D'(\alpha)$ is the sizes of the standard clothoid pair with a deflection α. Typical values of R are: $R(\alpha) \approx 0.75$ for small α, $R(\pi/4) = 0.7528$, $R(\pi/2) = 0.7624$, $R(3\pi/4) = 0.7832$, and $R(\pi) = 0.8309$. Thus, curvatures of cubic spirals are always smaller for normal deflections, and the smoothness of cubic spirals are numerically certified.

A cost function for path planning is said to be *size-free* if the problem is enlarged by n, the solution is also enlarged by n. Both cost functions proposed here are clearly size-free. Also, a cost function for path planning is said to be *continuous* if a small change in the problem causes a small change in the solution. Again, both cost functions are continuous. These are among some essential meta-properties in path planning theories.

References

[1] Y. Kanayama, "A Path/Motion Description Method and Its Application to Mobile Robot Language Design", Technical Report of Department of Computer Science at University of California at Santa Barbara, February 1989

[2] Hongo T., Arakawa, H., Sugimoto, G., Tange, K. and Yamamoto, Y. 1985. "An Automatic Guidance System of a Self-Controlled Vehicle - The Command System and the Control Algorithm ---," Proc. IECON.

[3] Komoriya, K., Tachi, S. and Tanie, K. 1986. "A Method of Autonomous Locomotion for Mobile Robots," Advanced Robotics, vol. 1, no. 1, pp. 3-19

[4] I. J. Cox, "Branche: An Autonomous Robot Vehicle for Structured Environments", Proc. IEEE International Conference on Robotics and Automation, pp. 978-982, 1988

[5] Y. Kanayama and Miyake N., "Trajectory Generation for Mobile Robots", Robotics Research, vol. 3, MIT Press, pp. 333-340, 1986

[6] Y. Kanayama, A. Nilipour and C. A. Lelm, "A Locomotion Control Method for Autonomous Vehicles", Proc. IEEE Conference on Robotics and Automation, pp. 1315-1317, 1988

[7] B. I. Hartman, Y. Kanayama, and T. Smith, "Model and Sensor Vased Navigation by an Autonomous Mobile Robot", Proc. of International Conference on Advanced Robotics, June 1989, to appear.

[8] Y. Kanayama, "A Path/Motion Description Method and Its Application to Mobile Robot Language Design", Technical Report of the Department of Computer Science at University of California at Santa Barbara, TRCS89-05, February 1989

[9] Y. Kanayama and T. Noguchi, "Spatial Learning by an Autonomous Mobile Robot with Ultrasonic Sensors", Technical Report of the Department of Computer Science at University of California at Santa Barbara, TRCS89-06, February 1989

[10] Y. Kanayama and B. I. Hartman, "Smooth Local Path Planning for Autonomous Vehicles", Technical Report of the Department of Computer Science at University of California at Santa Barbara, TRCS88-15, June 1988

[11] R. Weinstock, "Calculus of Variations", New York, Dover Publishing Inc., 1974