JOURNAL OF AEROSPACE COMPUTING, INFORMATION, AND COMMUNICATION
Vol. 4, December 2007

Planning Continuous Curvature Paths Using Constructive Polylines

Joshua Henrie* and Doran Wilde†
Brigham Young University, Provo, Utah 84602

DOI: 10.2514/1.32776

Previous methods for planning clothoid based continuous curvature paths aim at minimizing path length. However, minimal length paths are not always smooth, natural, and drivable. We discuss a method of generating clothoid-based trajectories using constructive polylines. The goal of the motion planner is to create a path for a large car-like vehicle in human driving environments, thus, the trajectories must be smooth, drivable, and natural enough to be used on roadways. We show several examples of trajectories developed for a DARPA Urban Challenge vehicle.

Nomenclature

κ	curvature, $d\theta/ds$
α	sharpness, $d\kappa/ds$
s	distance along a clothoid
L	length along a curve
λ	angle of circular arc (radians)
δ	change of heading (deflection); for a single clothoid: $\delta = \phi - \theta_0$, (radians)
ψ_0	angle from initial heading to d, (radians)
ψ_f	angle from d to final heading, (radians)
φ	angle from initial point to final point (along d) (radians)
θ	angle of heading with 0 along x-axis (radians)
η	change of angle between adjacent polylines, $(\theta_f - \theta_0)/2$, (radians)
C_0	complex number in x-y plane, $x_0 + iy_0$
$C(s)$	Fresnel cosine
$S(s)$	Fresnel sine
$F(s)$	complex clothoid
A, B	angles of triangles, (radians)
a, b	sides of triangles
d	length of line between initial and final points
d_1	polyline for a clothoid curve

Received 15 June 2007; revision received 15 October 2007; accepted for publication 16 October 2007.

* Graduate Student, Electrical and Computer Engineering, 459 Clyde Building.
† Associate Professor, Electrical and Computer Engineering, 459 Clyde Building, and Senior Member IEEE.

d_2 polyline used to scale clothoid curve
d_3 polyline for a circular arc
r radius of circular arc, $r = 1/\kappa$

Subscripts
$(\ )_{max}$ maximum
$(\ )_0$ initial
$(\ )_f$ final
$(\ )_c$ values pertaining to a clothoid

I. Introduction and Motivation

MOTION planning has been defined as the study of open loop controls that steer a mobile robot from a specified initial state to a specified final state over a certain time interval.[1] Motion planning can be done by constructing trajectories that have steering controls (curvature and sharpness) as well as velocity controls (velocity and acceleration/deceleration) embedded in the data structure of the trajectory.

Paths that are called "optimal paths" in the previous literature refer to minimal length paths. Dubins showed that a length-optimal path connecting an initial state and a final state of a mobile robot consists of only straight line segments and circular arc segments of maximal allowable curvature.[2] Reeds and Shepp gave a complete characterization of possible shortest paths.[3] They extended Dubins work by adding cusps (reversals in travel direction) to their paths, and they enumerated the possible types of optimal paths showing that an optimal path required no more than five segments and no more than two direction reversals. Fraichard and Sheuer showed that under the continuous curvature assumption that near optimal paths consist of straight lines, circular arc segments of maximal allowable curvature, and clothoid curves.[4] These curves have maximum allowable sharpness (i.e., change of curvature with respect to distance traveled) and are used to connect the straight-line segments to the circular arc segments. Fraichard and Sheuer paths are near optimal and within 10% of the length of Reeds and Shepp paths.[4]

Our motivation is to generate a smooth, natural, drivable trajectory between two points and then determine the maximum velocity, and hence minimize travel time along the trajectory. Since time-optimal trajectories do not necessarily require that mobile robots perform turns at maximal sharpness and curvature, we have been investigating methods for constructing continuous curvature trajectories where the curvature and sharpness are left as variables. Therefore, instead of constructing paths/trajectories from maximal curvature circular arcs, maximal sharpness clothoids, and line segments, we are investigating a new method for computing trajectories based on constructive polylines that allows us to compute curvature and sharpness of the trajectory as a function of the distance traveled and the headings.

In our method, a trajectory is computed using constructive polylines. These polylines connect the points where the curvature of the trajectory is zero, which includes straight lines and inflection points. Furthermore, the polylines under curves are subdivided at the point where the trajectory is parallel to the construction line as shown Fig. 1. These construction polylines are defined and used differently than the control lines used by Walton and Meek to draw clothoid arcs in a graphic editor.[5]

Section II describes an important primitive method, called "Clothoid_Turn," used for motion planning trajectories described by constructive polylines as we have proposed. To motion plan a curve (Fig. 2(a)), you simply need to know (1.) the length d of the polyline segment, and (2.) the heading deflection δ, which is also the angle at which

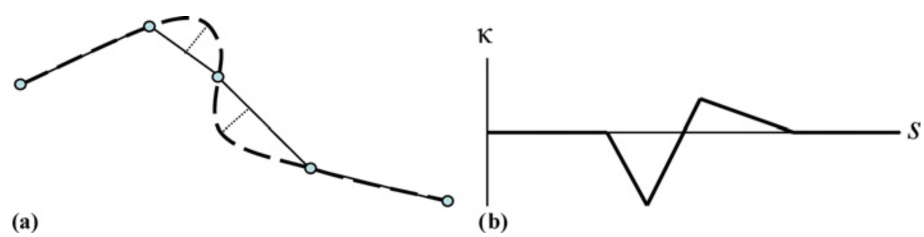

Fig. 1 (a) A trajectory and the associated constructive polyline. (b) The curvature plot for the trajectory on the left.

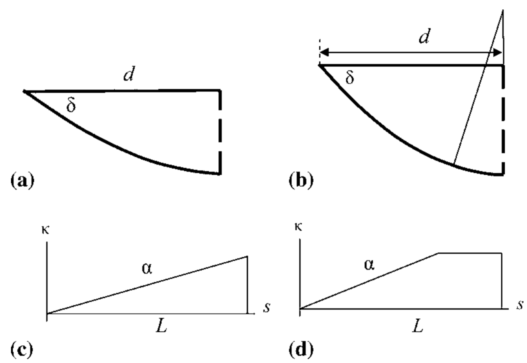

Fig. 2 (a) Clothoid segment given heading deflection δ and polyline length d. (b) Clothoid/arc segment given heading deflection δ and polyline length d. (c) Curvature diagram for clothoid segment with arc-length L, final curvature κ, and sharpness (slope) α. (d) Curvature diagram for clothoid/arc segment with arc-length L, final curvature κ, and sharpness (slope) α.

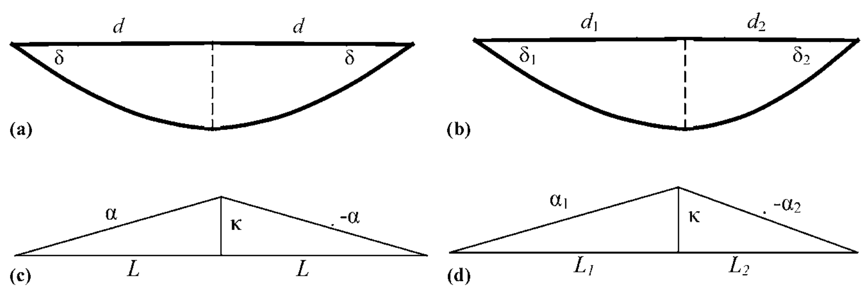

Fig. 3 (a) Symmetric curve. (b) Asymmetric curve. (c) Symmetric curvature diagram. (d) Asymmetric curvature diagram.

the curve is incident to the polyline. The method then computes the arc-length L, the final curvature κ, and the sharpness α (Fig. 2(c)). If a maximum curvature $\kappa_{\max}$ is specified, then two types of curves can be generated. If the final curvature is less than or equal to the maximum curvature $\kappa_{\max}$ then a simple clothoid (Fig. 2(a) and 2(c)) is generated, otherwise, a combination clothoid/arc curve is generated (Fig. 2(b) and 2(d)).

This method only really plans half of a curve. If the curve is symmetric (Fig 3(a) and 3(c)), then a single call to the clothoid turn method plans both halves of the curve, and $d = Length(polyline)/2$. If the curve is asymmetric (Fig. 3(b) and 3(d)), then two calls to the clothoid turn method must be made, where $d_1 + d_2 = Length(polyline)$. A curve of two clothoids that begins and ends with zero curvature is called a clothoid set.

II. Clothoid Equations and Planning a Basic Clothoid Curve

Clothoids are curves with continuous increasing/decreasing linear curvature and are defined in the complex plane by Eq. (1) which is derived by Kimia et al.[6]

$$F(s) = C_0 + \operatorname{sign}(\alpha)\sqrt{\frac{\pi}{|\alpha|}}e^{i(\theta_0-(\kappa_0^2/2\alpha))}\left[C\left(\frac{\kappa_0+\alpha s}{\sqrt{\pi|\alpha|}}\right) - C\left(\frac{\kappa_0}{\sqrt{\pi|\alpha|}}\right) + iS\left(\frac{\kappa_0+\alpha s}{\sqrt{\pi|\alpha|}}\right) - iS\left(\frac{\kappa_0}{\sqrt{\pi|\alpha|}}\right)\right], \alpha \neq 0 \tag{1}$$

The Fresnel sine and cosine are defined as:

$$C(s) = \int_0^s \cos\left(\frac{\pi}{2}u^2\right)\mathrm{d}u \tag{2}$$

$$S(s) = \int_0^s \sin\left(\frac{\pi}{2}u^2\right)\, \mathrm{d}u \tag{3}$$

Equation (1) is valid for all clothoids except where $\alpha = 0$ which is the case for straight lines and circular arcs. To simplify our analysis it is assumed that the initial and final points of a clothoid sets will begin and end with zero curvature, $\kappa_0 = 0$ (Fig. 3(c) and 3(d)). In essence, this means that the vehicle's wheels must be straight upon entering or exiting a trajectory. Assuming $\kappa_0 = 0$ greatly simplifies the process of creating continuous curvature trajectories by allowing the catenation of arbitrary clothoid sets and straight-line segments. If the vehicle exits a trajectory with non-zero curvature, an additional clothoid must be added in order to create a continuous curvature trajectory that can connect to a straight line. However, if the vehicle exits a trajectory with zero curvature then it may enter a tangential straight-line segment without needing to add additional clothoids for the transition.

Without loss of generality, a trajectory can be computed for the normal case $\theta_0 = 0$ and $C_0 = 0$, and then rotated and translated into its correct orientation. Section II.B further explains the process of rotating and translating a clothoid curve.

After applying the assumptions of $\kappa_0 = 0$, $\theta_0 = 0$, and $C_0 = 0$, Eq. (1) reduces to Eq. (4).

$$F(s) = \sqrt{\frac{\pi}{|\alpha|}}\left[\mathrm{sign}(\alpha) C\left(\frac{\alpha s}{\sqrt{\pi|\alpha|}}\right) + i S\left(\frac{\alpha s}{\sqrt{\pi|\alpha|}}\right)\right], \quad \alpha \neq 0 \tag{4}$$

Although Eq. (4) has been significantly reduced from Eq. (1), it is stated only in terms of the unknown curve length, s, and sharpness, α. For our motion planner we assume that the only knowns to the trajectory planner are the initial and final points and their associated headings, θ_0 and θ_f. Therefore, the length, curvature, and sharpness of the desired clothoid for the trajectory are not inputs. The change in heading, also known as deflection[4] ($\delta = \theta_f - \theta_0$), can be easily computed from the initial and final points. To derive the primitive clothoid turn method referred to in the introduction; rewrite Eq. (4) in terms of δ and κ, Eq. (9), instead of s and α. The derivation for Eq. (9) is given below.

A curvature diagram showing the curvature, κ, with respect to the distance along the curve, s, (Fig. 4(a)) is a convenient diagram to work with because it visually shows all aspects of a clothoid, except position: curvature (vertical axis), sharpness (slope), length (horizontal axis), and change of heading (area under the curve). Figure 4 shows a clothoid in the s-κ domain and the x-y domain. From Fig. 4a, the relationships between variables δ and κ and variables s and α are visible (Eqs. (5)).

$$\kappa = \alpha s, \quad \delta = \frac{\alpha s^2}{2}, \quad \alpha = \frac{2\delta}{\kappa^2} \tag{5}$$

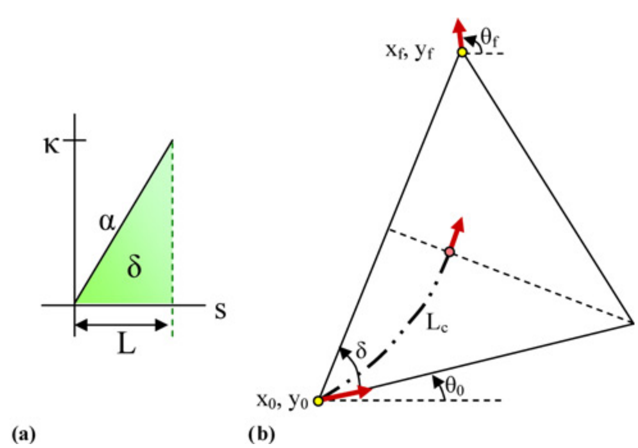

Fig. 4 (a) A clothoid in s-κ domain. (b) A clothiod in x-y domain.

Using the Eq. (5) the inner terms of the Fresnel integrals in Eq. (4) are put into terms of δ.

$$\frac{\alpha s}{\sqrt{\pi|\alpha|}} = \left\{ \begin{array}{l} \alpha > 0\colon \sqrt{\dfrac{\alpha^2 s^2}{\pi\alpha}} = \sqrt{\dfrac{\alpha s^2}{\pi}} = \sqrt{\dfrac{2\delta}{\pi}} \\ \alpha < 0\colon (-1)\sqrt{\dfrac{(-\alpha)^2 s^2}{\pi(-\alpha)}} = (-1)\sqrt{\dfrac{(-\alpha)s^2}{\pi}} = (-1)\sqrt{\dfrac{2(-\delta)}{\pi}} \end{array} \right\} = \frac{\delta}{|\delta|}\sqrt{\frac{2|\delta|}{\pi}} = \operatorname{sign}(\delta)\sqrt{\frac{2|\delta|}{\pi}} \tag{6}$$

Next, perform change of variables on the constant term in front of the Fresnel equations in Eq. (4) to terms of δ and κ by using the same Eq. (5).

$$\sqrt{\frac{\pi}{|\alpha|}} = \left\{ \begin{array}{l} \alpha > 0\colon \sqrt{\dfrac{\pi}{\alpha}} = \dfrac{\sqrt{2\pi\delta}}{\kappa} \\ \alpha < 0\colon \sqrt{\dfrac{\pi}{-\alpha}} = \dfrac{\sqrt{2\pi(-\delta)}}{-\kappa} \end{array} \right\} = \sqrt{\frac{\delta}{|\alpha|}} = \frac{\sqrt{2\pi|\delta|}}{|\kappa|} \tag{7}$$

For a single clothoid beginning with zero curvature as shown in Fig. 4,

$$\operatorname{sign}(\alpha) = \frac{\alpha}{|\alpha|} = \frac{\kappa}{|\kappa|} = \frac{\delta}{|\delta|} = \operatorname{sign}(\delta). \tag{8}$$

Substituting Eqs. (6)–(8) into Eq. (4), the final transformed clothoid equation is

$$F\left(\frac{2\delta}{\kappa}\right) = \frac{\sqrt{2\pi|\delta|}}{|\kappa|}\left[C\left(\sqrt{\frac{2|\delta|}{\pi}}\right) + \operatorname{sign}(\delta)\, i\, S\left(\sqrt{\frac{2|\delta|}{\pi}}\right)\right]. \tag{9}$$

A. Translating Clothoid Equation to the x and y Plane

To translate Eq. (9) from the complex plane to the x and y plane, let

$$F\left(\frac{2\delta}{\kappa}\right) = x\left(\frac{2\delta}{\kappa}\right) + i y\left(\frac{2\delta}{\kappa}\right) \tag{10}$$

where

$$x\left(\frac{2\delta}{\kappa}\right) = \frac{\sqrt{2\pi|\delta|}}{|\kappa|} C\left(\sqrt{\frac{2|\delta|}{\pi}}\right) \tag{11}$$

$$y\left(\frac{2\delta}{\kappa}\right) = \operatorname{sign}(\delta)\frac{\sqrt{2\pi|\delta|}}{|\kappa|} S\left(\sqrt{\frac{2|\delta|}{\pi}}\right). \tag{12}$$

An important and key observation from Eqs. (11) and (12) is that κ determines the size of the clothoid and δ determines the shape of the clothoid. Therefore, pull κ out of Eqs. (11) and (12) and use it to scale the size of the clothoid without altering the shape of the clothoid.

$$x\left(\frac{2\delta}{\kappa}\right) = \frac{1}{|\kappa|} x(2\delta) \tag{13}$$

$$y\left(\frac{2\delta}{\kappa}\right) = \frac{1}{|\kappa|} y(2\delta) \tag{14}$$

where the shape of the clothoid is determined by

$$x(2\delta) = \sqrt{2\pi|\delta|}\, C\left(\sqrt{\frac{2|\delta|}{\pi}}\right) \tag{15}$$

$$y(2\delta) = \operatorname{sign}(\delta)\sqrt{2\pi|\delta|}\, S\left(\sqrt{\frac{2|\delta|}{\pi}}\right). \tag{16}$$

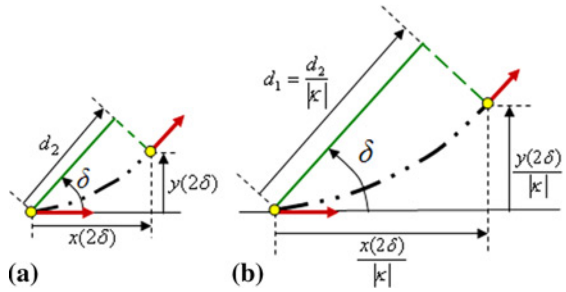

Fig. 5 Clothoid scaling. (a) Clothoid for when $\kappa = 1$. (b) Clothoid scaled by $1/|\kappa|$.

B. Rotating and Scaling Clothoids

The rotating and scaling of clothoids is a key result of this paper and the basis of the proposed method to construct trajectories. Figure 5 shows similar clothoids where Fig. 5(a) is produced from Eqs. (15) and (16) and Fig. 5(b) is produced from Eqs. (11) and (12). Equations (13) and (14) describe the relationship between the two figures. Because the clothoids in Fig. 5 are similar, they share the same initial and final headings and deflection. However, x, y, d_2, and the length of the curve are all scaled linearly with $1/|\kappa|$. Thus, the distance between the beginning and ending points in Fig. 5(b) can be spanned by scaling the clothoid in Fig. 5(a).

1. Rotating and Scaling Algorithm

To scale and rotate a clothoid first let $\kappa = 1$ and compute $x(2\delta)$ and $y(2\delta)$ which are functions of only δ. Next, rotate $x(2\delta)$, $y(2\delta)$ by $-\delta$; the x-component of the resulting point is d_2 in Eq. (17).

$$d_2 = \cos(\delta)x(2\delta) + \sin(\delta)y(2\delta) \tag{17}$$

As described in the next section, the distance d_1 in Fig. 5(b) is computed by Eq. (20). After computing both d_1 and d_2 the curvature, κ, can be computed from Eq. (18).

$$\kappa = \text{sign}(\delta)\frac{d_2}{d_1} \tag{18}$$

Once κ is computed, the length and sharpness of the scaled clothoid (Fig. 5(b)) can be found using Eq. (19). The result is a clothoid turn with the desired length and change of heading. The clothoid turn algorithm with inputs d_1 and δ and outputs κ, α and L can be seen if Fig. 6.

$$\alpha = \frac{\kappa^2}{2\delta}, \quad L = \frac{\kappa}{\alpha} \tag{19}$$

III. Constructing Trajectories Using Clothoids, Arcs, and Lines

Trajectories between two points begin and end with zero curvature and consist of a combination of straight lines, arcs, and clothoid turns. The most basic trajectory consists of a single straight line. The next most basic trajectory

```
Clothoid_Turn ( d1, δ )
{
    if δ == 0  //Straight Line
    {
        κ = 0
        α = 0
        L = d1
    }
    else if δ > 0 //Left Turning Clothoid
    {
        d2 = cos(δ) C(2δ) + sin(δ) S(2δ)
        κ = d2 / d1
        α = κ^2 / 2δ
        L = κ / α
    }
    else if δ < 0 //Right Turning Clothoid
    {
        d2 = cos(δ) C(2δ) - sin(δ) S(2δ)
        κ = -d2 / d1
        α = κ^2 / 2δ
        L = κ / α
    }
}
```

Fig. 6 Clothoid turn algorithm. Outputs of the algorithm are κ, α, and L.

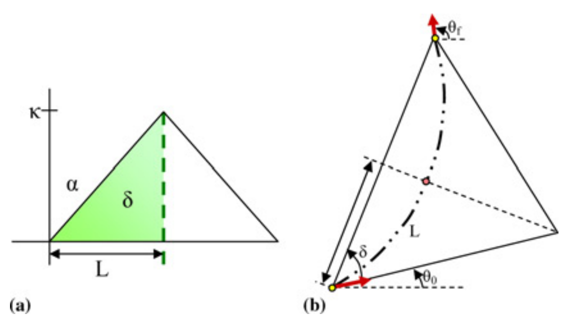

Fig. 7 (a) Symmetric clothoid set with unconstrained curvature in s-κ plane. (b) Symmetric clothoid set in the x-y plane.

consists of two clothoids of the same sharpness, the second clothoid mirroring the first (Fig. 7) is called a symmetric clothoid set. To create a basic symmetric clothoid set choose d_1 such that

$$d_1 = \frac{\sqrt{(x_f - x_0)^2 + (y_f - y_0)^2}}{2}. \tag{20}$$

A d_1 from Eq. (20) creates a clothoid turn that covers half the distance between the initial and final points. Mirroring the clothoid covers the second half of the distance and a trajectory is created (Fig. 7).

A clothoid set is a basic trajectory because it is based on a single construction polyline and begins and ends with zero curvature. At this point, we expand the definition of a clothoid set to include an optional arc. Adding an arc to a clothoid set creates the same type of trajectory as in Fig. 7 but with upper-bounded curvature. This extended

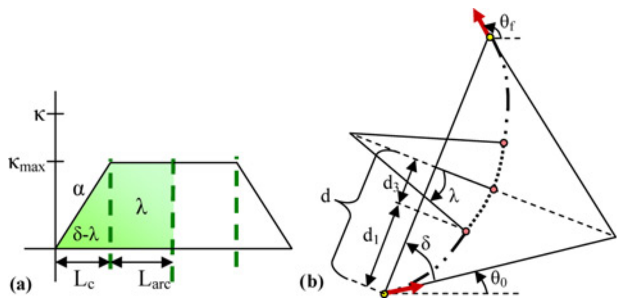

Fig. 8 (a) Clothoid set with an arc to constrain curvature in the s-κ plane. (b) A clothoid set with inserted arc in the x-y plane. Note that d^1 is less than the d^1 in Eq. (20), and d^3 makes up the remaining distance to the halfway point between the initial and final points of the desired trajectory. L^c is the length of the clothoid and L^{arc} is the length of the arc. The angle of the arc, λ (i.e. the change of heading of the arc), is a part of the change of heading of half the trajectory.

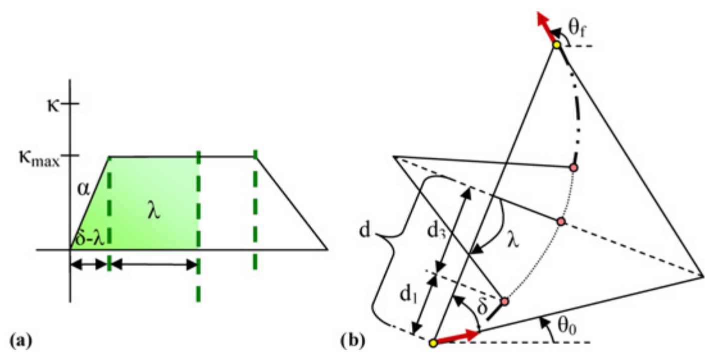

Fig. 9 (a) A trajectory with two clothoids of different sharpness and an arc to constrain curvature in the s-κ plane. (b) A trajectory with two clothoids of different sharpness and inserted arc in the x-y plane.

definition of a clothoid set is similar to a "CC-Turn" by Fraichard and Scheuer[4] except it is not required to have maximum sharpness and curvature.

Developing a clothoid set with an arc is similar to the process of creating a clothoid set without an arc. To develop a symmetric clothoid set, first create a clothoid with a maximum curvature of κ_{max}. Then add an arc to finish the remaining half of the distance between the beginning and ending points. The arc is then mirrored and a mirror of the first clothoid is added on the end (Fig. 8).

It should be noted that choosing d_1 to be half the distance between the initial and final points is not required, but it is a simple and effective solution. The same principles of trajectory generation apply to asymmetric clothoid sets (Fig. 9).

IV. Application of Trajectories

This section discusses some trajectories created for common driving maneuvers of a DARPA Urban Challenge vehicle using clothoid-arc sets and lines. Each of the trajectories follow the specifications that a trajectory begins and ends with zero curvature, the vehicle's minimum turning radius (i.e., maximum curvature, κ_{max}) is not violated, and the only information given to the system are the initial and final points and their associated headings.

A. Generic Turn Trajectory

The goal of the generic turn method is to create a trajectory between any two given points with any two headings, assuming there is sufficient distance between the points that a trajectory can be created within the constraints stated in the introduction of this section. Figure 10 shows a generic turn in the s-κ plane and in the x-y plane. To create

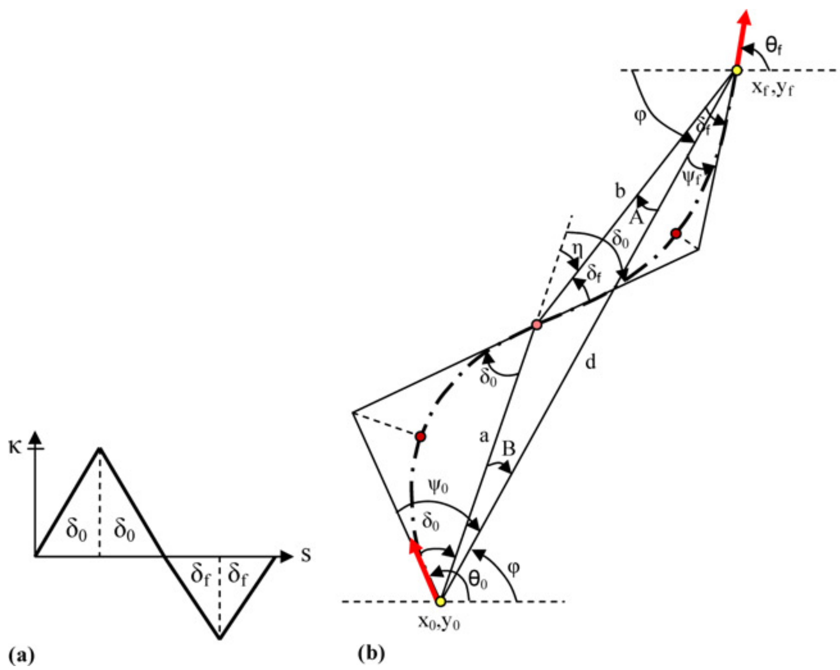

Fig. 10 (a) Generic turn trajectory in the s-κ plane. (b) Generic turn trajectory in the x-y plane (note that d here is different from d in Fig. 9b).

a trajectory between points x_0, y_0 and x_f, y_f, the lengths of polylines a and b, and the deflection angles δ_0 and δ_f must be computed. The only input variables of the system are θ_0, θ_f, x_0, y_0, x_f, and y_f. From these inputs, values for ψ_0, ψ_f, η, φ, and d are found using Eqs. (24), (25), (26), (21), and (20), respectively. It is important to note that half of the lengths of a and b will replace d in Fig. 8(b) of the clothoid turn. Using the geometry in Fig. 10(b), equations for constructive polylines a and b are derived using the Law of Sines. Equations (21)–(26) are also obtained from the figure and are used to compute values for angles δ_0 and δ_f.

$$\varphi = \operatorname{atan}\left(\frac{y_f - y_0}{x_f - x_0}\right) \tag{21}$$

$$\theta_0 = \varphi - B - \delta_0 \tag{22}$$

$$\theta_f = \varphi + A + \delta_f \tag{23}$$

$$\psi_0 = \varphi - \theta_0 = \delta_0 + B \tag{24}$$

$$\psi_f = \theta_f - \varphi = \delta_f + A \tag{25}$$

$$\eta = A + B = \delta_0 + \delta_f \tag{26}$$

For the first clothoid set in the generic turn, the deflection is labeled as δ_0 and the second clothoid set deflection is δ_f. Thus, the total deflection from the initial point to the final point is the sum of all angles δ_0 and δ_f (Eq. (27)).

$$\theta_f - \theta_0 = 2(\delta_0 + \delta_f) = 2\eta \tag{27}$$

Because A, B, δ_0, and δ_f cannot be computed from Fig. 10(b) and Eqs. (21)–(27), they must be fixed before the construction polylines can be determined. We use Table 1 to summarize the constraints of Eqs. (22)–(26) that restrict the values for A, B, δ_0, and δ_f. The sums of rows and columns, ψ_0, ψ_f, and η, are functions of inputs in Eqs. (21), (24), (25), and (26). The horizontal rows sum to $A + B = \eta$, (Eq. (26)); $\delta_0 + \delta_f = \eta$, (Eq. (26)); and $\psi_0 + \psi_f = 2\eta$. The vertical columns sum to $A + \delta_f = \psi_f$, (Eq. (25)); $B + \delta_0 = \psi_0$, (Eq. (24)); and $\eta + \eta = 2\eta$.

Table 1 Constraints of Eqs. (22)–(26).

A	B	η
δ_f	δ_0	η
ψ_f	ψ_0	2η

Table 2 Null Space Added to Constraints.

$A+q$	$B-q$	η
δ_f-q	δ_0+q	η
ψ_f	ψ_0	2η

While Table 1 constrains the variables A, B, δ_0, and δ_f, there remains one degree of freedom in the system, $A+B=\delta_0+\delta_f$. To characterize an entire family of solutions, we introduce a null space parameter q. By adding a null space to Table 1 it allows A, B, δ_0, and δ_f, to be adjusted with respect to each other while maintaining the integrity of the constraints. Table 2 shows that given a solution (A, B, δ_0, δ_f) then ($A+q$, $B-q$, δ_0+q, δ_f-q) is also a solution for any value of q.

To characterize the entire family of solutions we only need to find a single solution within the family. Although any number of solutions satisfy the equations in Table 1, Eq. (26) immediately suggests two candidate solutions: $A=\delta_0$, $B=\delta_f$ and $A=\delta_f$, $B=\delta_0$. By plugging in the first candidate solution into Eqs. (22)–(26), they produce to the following equations.

$$\theta_0=\varphi-\delta_0-\delta_f \tag{28}$$

$$\theta_f=\varphi+\delta_0+\delta_f \tag{29}$$

$$\psi_0=\delta_0+\delta_f \tag{30}$$

$$\psi_f=\delta_0+\delta_f \tag{31}$$

$$\eta=\delta_0+\delta_f \tag{32}$$

Table 3 summarizes the constraints for the solution $A=\delta_0$ and $B=\delta_f$. Summing the columns and rows of the table show that the solution is valid, but only when $\psi_0=\psi_f$ (Eqs. (30) and (31)). The constraint $\psi_0=\psi_f$ added to the other constraints of the table results in a full rank system and is therefore a single point solution.

The next step is to check if the second candidate solution, $A=\delta_f$ and $B=\delta_0$, is valid and provides a desirable family of solutions. This solution reduces Eqs. (22)–(26) to the following equations.

$$\theta_0=\varphi-2\delta_0 \tag{33}$$

$$\theta_f=\varphi+2\delta_f \tag{34}$$

$$\psi_0=2\delta_0 \tag{35}$$

$$\psi_f=2\delta_f \tag{36}$$

$$\eta=\delta_0+\delta_f \tag{37}$$

Table 3 Constraints for the solution $A=\delta_0$ and $B=\delta_f$.

δ_0	δ_f	$\delta_0+\delta_f=\eta$
δ_f	δ_o	η
ψ_f	ψ_o	$\psi_o+\psi_f=2\eta$

Table 4 Constraints for the solution $A = \delta_f$, $B = \delta_0$, and free parameter q.

$\frac{\psi_f}{2} + q$	$\frac{\psi_0}{2} - q$	$\frac{\psi_0 + \psi_f}{2} = \eta$
$\frac{\psi_f}{2} - q$	$\frac{\psi_0}{2} + q$	$\frac{\psi_0 + \psi_f}{2} = \eta$
ψ_f	ψ_o	$\psi_o + \psi_f = 2\eta$

Again, Table 4 summarizes the constraints for the solution $A = \delta_f$, $B = \delta_0$ and free parameter q. Each column and row of the table sum correctly, therefore $A = \delta_f$ and $B = \delta_0$ is a valid solution. Furthermore, there are no other additional constraints introduced to the system. Thus, the solution $A = \delta_f$ and $B = \delta_0$ produces a family of valid solutions for any value of q.

Using Eqs. (35) and (36) δ_0 and δ_f can be computed and therefore polylines a and b can be found for each solution within the solution set using the Law of Sines (Eqs. (38) and (39)) on triangle abd in Fig. 10(b). With the length of constructive polylines a and b computed, the generic turn problem reduces to two clothoid set trajectories. Figure 11

Generic_Turn ($x_0, y_0, \theta_0, x_f, y_f, \theta_f, q$)
{

$$\varphi = \operatorname{atan}\left(\frac{y_f - y_0}{x_f - x_0}\right)$$

$$d = \sqrt{(x_f - x_0)^2 + (y_f - y_0)^2}$$

$$\psi_0 = \varphi - \theta_0$$

$$\psi_f = \theta_f - \varphi$$

$$\eta = \psi_0 - \psi_f$$

$$A = \frac{\psi_f}{2} + q \text{ (see Table 4)}$$

$$B = \frac{\psi_0}{2} - q \text{ (see Table 4)}$$

$$\delta_0 = \frac{\psi_0}{2} + q \text{ (see Table 4)}$$

$$\delta_f = \frac{\psi_f}{2} - q \text{ (see Table 4)}$$

$$a = \frac{d \sin(A)}{\sin(\eta)}$$

$$b = \frac{d \sin(B)}{\sin(\eta)}$$

Clothoid_Turn ($\frac{a}{2}$, δ_0) $\rightarrow \kappa_1, \alpha_1, L_1$

Clothoid_Turn ($\frac{b}{2}$, δ_f) $\rightarrow \kappa_2, \alpha_2, L_2$

}

Trajectory $\leftarrow$ clothoid ($\kappa_0 = 0, \alpha = \alpha_1, L = L_1$) +
clothoid ($\kappa_0 = \kappa_1, \alpha = -\alpha_1, L = L_1$) +
clothoid ($\kappa_0 = 0, \alpha = \alpha_2, L = L_2$) +
clothoid ($\kappa_0 = \kappa_2, \alpha = -\alpha_2, L = L_2$)

Fig. 11 Algorithm for a symmetric generic turn trajectory. The output of the generic turn algorithm is a trajectory of four concatenated clothoid turns.

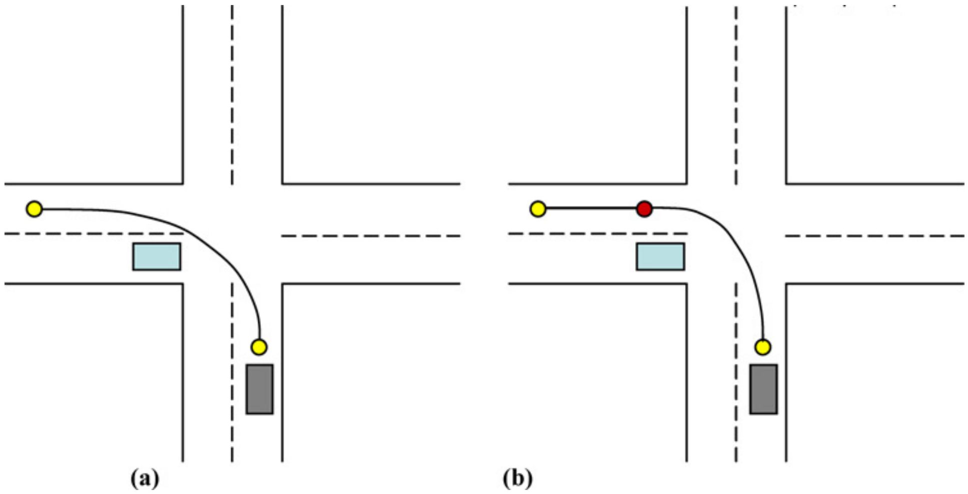

Fig. 12 (a) Undesirable shallow left turn with a possible collision. (b) A left turn consisting of a curve and a straight segment to avoid a collision.

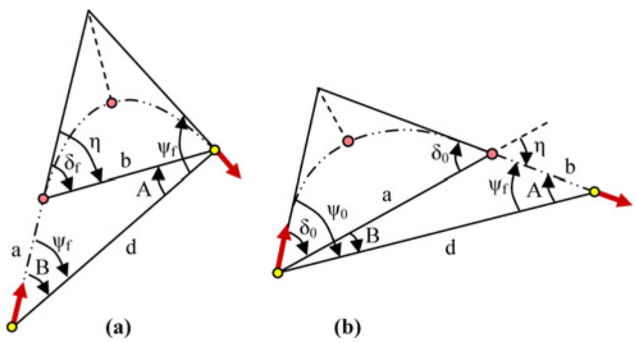

Fig. 13 (a) Straight-curve trajectory. (b) Curve-straight trajectory.

outlines the algorithm for the generic turn.

$$a = \frac{d\sin(A)}{\sin(\pi - \eta)} = \frac{d\sin\left(\psi_f/2 + q\right)}{\sin(\eta)} \tag{38}$$

$$b = \frac{d\sin(B)}{\sin(\pi - \eta)} = \frac{d\sin\left(\psi_0/2 - q\right)}{\sin(\eta)} \tag{39}$$

B. Special Case Maneuvers

Sometimes two clothoid sets produce a trajectory that, although valid, is undesirable, such as in the case of making a shallow left turn at an intersection. If a second vehicle stops at the left part of the intersection, the motion planner should not plan a trajectory such that the vehicles collide (Fig. 12(a)). For these types of situations, it is easier to have special case turns that have a straight line either before or after a clothoid trajectory (Fig. 12(b)). In this section, we discuss how to create specific maneuvers for an Urban Challenge vehicle.

1. Trajectories for Turns and Straight Segments

The solution $A = \delta_f$ and $B = \delta_0$ for any value of q has been shown to be valid (Table 4), but what about the case when $\delta_0 = 0$? By choosing a specific member of the solution set (i.e., by choosing q) the generic turn method can produce a straight segment along polyline a, followed by one clothoid set using polyline b (Fig. 13(a)). This is called

Table 5 Constraints for the straight-curve trajectory for $q = -\psi_0/2$. (see Table 4).

$\frac{\psi_f}{2} - \frac{\psi_0}{2}$	ψ_0	$\frac{\psi_0 + \psi_f}{2} = \eta$
$\frac{\psi_f}{2} + \frac{\psi_0}{2}$	0	$\frac{\psi_0 + \psi_f}{2} = \eta$
ψ_f	ψ_0	$\psi_o + \psi_f = 2\eta$

Table 6 Constraints for the curve-straight trajectory for $q = \psi_f/2$ (see Table 4).

ψ_f	$\frac{\psi_0}{2} - \frac{\psi_f}{2}$	$\frac{\psi_0 + \psi_f}{2} = \eta$
0	$\frac{\psi_0}{2} + \frac{\psi_f}{2}$	$\frac{\psi_0 + \psi_f}{2} = \eta$
ψ_f	ψ_0	$\psi_o + \psi_f = 2\eta$

Table 7 Constraints for the lane change trajectory for $q = 0$ and $\theta_0 = \theta_f$ (see Table 4).

0	0	$0 = \eta$
δ_f	δ_0	$0 = \eta$
ψ_f	ψ_0	$0 = \eta$

a straight-curve trajectory and it is produced by setting $q = -\psi_0/2$. Applying q to the generic turn, Table 4, results in a straight-curve trajectory, Table 5, where $\delta_0 = 0$.

The generic turn method can also produce a trajectory consisting of a clothoid set on polyline a, followed by a straight segment of length b (Fig. 13(b)) by choosing another specific member of the solution set (i.e., by choosing a different q). This is called a curve-straight trajectory, and is produced by setting $q = \psi_f/2$. And again applying q to the generic turn, Table 4, results in a curve-straight trajectory, Table 6, where $\delta_f = 0$.

Because the solutions of the maneuvers are members of the solution set of the generic turn, the algorithms of the straight-curve and curve straight maneuvers consist of a single call to Generic_Turn( ), passing the appropriate value of q.

2. Lane Change Trajectory

One of the requirements of the DARPA Urban Challenge is that the vehicle must be able to pass another vehicle; the lane change maneuver does just that. The trajectory is used when the beginning and ending point headings are the same (i.e., $\theta_0 = \theta_f$) but the points are offset by some x and y distance (Fig. 14). This maneuver allows the path planner to provide just two points and let the motion planner take care of the details. Without the lane change maneuver, the path planner would have to provide three points and their headings to complete the maneuver using left and right turns. Again, because the lane change trajectory is a member of the generic turn solution set, the algorithm requires a single call to Generic_Turn( ) where $\theta_0 = \theta_f$ and $q = 0$ (see Table 7).

3. U-Turn Trajectory

The u-turn is another maneuver required by the Urban Challenge. This trajectory comes about when the total deflection is approximately π. The actual trajectory is a mirrored clothoid set with d from Eq. (20) passed to a single call of Clothoid_Turn( ). The trajectory consists of two clothoids (Fig. 15(a)): clothoid ($\kappa_0 = 0, \alpha = \alpha$,

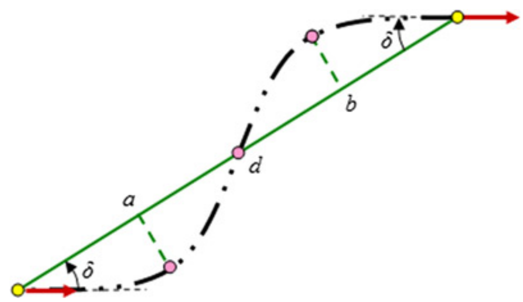

Fig. 14 Lane change trajectory maneuver

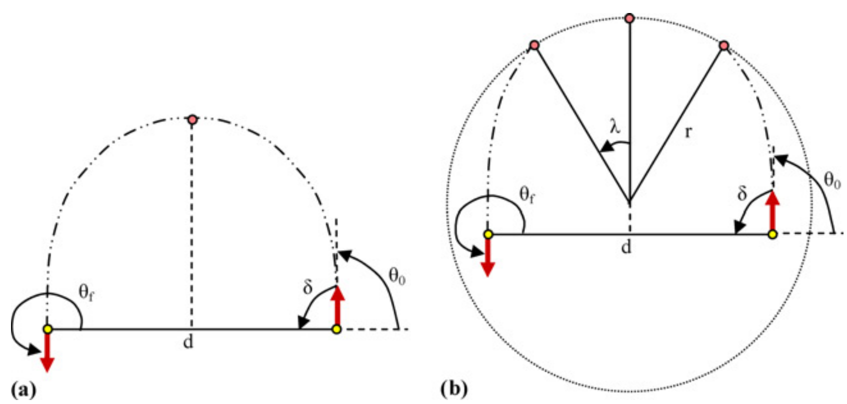

Fig. 15 (a) U-turn trajectory, (b) U-turn trajectory with an arc.

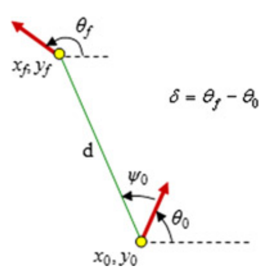

Fig. 16 Using ψ_0 and δ to determine which maneuver to execute.

$L = L) + \text{clothoid}(\kappa_0 = \kappa, \alpha = -\alpha, L = L)$; or two clothoids and an arc between them (Fig 15(b)): clothoid $(\kappa_0 = 0, \alpha = \alpha, L = L_c) + \text{arc}(\kappa_0 = \kappa_{\max}, \alpha = 0, L = L_{\text{arc}}) + \text{clothoid}(\kappa_0 = \kappa_{\max}, \alpha = -\alpha, L = L_c)$.

C. When to Execute a Trajectory Maneuver

The motion planner must know how to determine which maneuver to execute. One way is to examine the values and signs of ψ_0 and δ (as shown in Fig. 16). For example, for the maneuvers we have discussed previously:

- if $\psi_0 > 0$ and $\delta > 0$ then a left turn is performed,
- if $\psi_0 < 0$ and $\delta < 0$ then a right turn is performed,
- if $\delta = 0$ then a left or right lane change maneuver is executed depending on the sign of ψ_0, or
- if $\psi_0 = \pi/2$ and $\delta = \pi$ then a left u-turn is executed.

Other maneuvers such as U-turn-straight or straight-U-turn trajectories have not been discussed.

It is important to note that the feasibility of a maneuver depends on the length of d. If d is too small, then it might not be possible to perform a given maneuver. For example, imagine taking a u-turn to a point only 2 feet away. Maneuvers depend upon d, ψ_0, and δ. We believe that the minimum length d for which a maneuver is feasible (Eq. (40)) is a function of ψ_0 and δ. We do not give this function here but it is a matter of future research.

$$d \geq f(\delta, \psi_0) \tag{40}$$

Maneuvers that cannot be performed are deemed "Illegal" meaning that either (1) a trajectory for that maneuver cannot be created that is smooth and drivable, (2) the maneuver is not usually legal or wise to perform in normal traffic situations (e.g., a right hand u-turn), (3) the maneuver is not simple and may require driving in reverse.

A further advantage of analyzing the legality of maneuvers is that it adds another check between the path planner and motion planner for drivability and safety. Should the motion planner encounter an illegal maneuver condition, it returns an error to the path planner stating that the motion planner cannot create a safe and drivable trajectory.

V. Conclusion

We showed that the deflection δ of a clothoid determines its shape, and the curvature κ determines its size. The x and y components of the clothoid curve and hence the length of the curve, all scale linearly with $1/|\kappa|$ while the shape of the clothoid curve stays the same. Thus, given the deflection of a curve, a clothoid curve can be created, and then given the length of its associated polyline, the curve can be scaled to the correct size. We have used this principle to design the "Clothoid_Turn" primitive method that allows us to generate clothoids using constructive polylines. Using this method, we compute paths that are more complicated between an initial point and final point using only the coordinates of those points and their associated headings. This is the basis of our motion planner.

Using polylines and the clothoid turn method, we developed a clothoid set consisting of two clothoid curves which is a turn similar to a "C" path by Reeds and Shepp[3] except that it uses clothoid arcs as well as circular arcs. The u-turn is a special case of this type of path. We also developed a generic turn algorithm that produces a path consisting of two clothoid sets similar to a "CC" path by Reeds and Shepp.[3] We then showed that the straight-curve turn, the curve-straight turn, and the lane change are all special cases of the generic turn. All of the maneuvers discussed in Section IV.B are smooth, natural, and drivable trajectories.

For future research, we are developing paths using the clothoid turn method that include more than two polylines and allow for reversal in direction. We are also trying to characterize feasible paths by finding a function that will return the minimal length of d required for a valid trajectory (Eq. 40). We are also continuing to develop a set of heuristics to determine the type of maneuver to perform, and how to parameterize the generic turn for different maneuvers.

References

[1] Kolmanovsky, I., and McClamroch, N.H., "Developments in Nonholonomic Control Problems," *IEEE Control Systems*, Vol. 15, No. 6, 1995, pp. 20–36.

[2] Dubins, L.E., "On Curves of Minimal Length with a Constraint on Average Curvature, and with Prescribed Initial and Terminal Positions and Tangents," *American Journal of Mathematics*, Vol. 79, 1957, pp. 497–517.

[3] Reeds, J.A., and Shepp, L.A., "Optimal Paths for a Car that Goes both Forwards and Backwards," Pacific *Journal of Mathematics*, Vol. 145, No. 2, 1990, pp. 367–393.

[4] Fraichard, T., and Scheuer, A., "From Reeds and Shepp's to Continuous-Curvature Paths," *Robotics, IEEE Transactions on [see also Robotics and Automation, IEEE Transactions on]*, Vol. 20, No. 6, 2004, pp. 1025–1035.

[5] Walton, D.J., and Meek, D.S., "A Controlled Clothoid Spline," *Computers and Graphics*, Vol. 29, No. 3, 2005, pp 353–363.

[6] Kimia, B. B., Frankel, I., and Popescu, A., "Euler Spiral for Shape Completion," *International Journal of Computer Vision*, Vol. 54, No. 1–3, 2003, pp. 157–180.

Christopher Rouff
Associate Editor