2012 American Control Conference
Fairmont Queen Elizabeth, Montréal, Canada
June 27-June 29, 2012

Tracking Expanding Star Curves Using Guidance Vector Fields

Eric W. Frew, Dale Lawrence
University of Colorado, Boulder, CO, 80302

Abstract—**This paper describes a constructive method for creating provably stable vector fields for a class of star shaped, three-dimensional curves. Using the approach, vector fields can be designed that provide essentially global exponential convergence to a desired curve expressed in cylindrical coordinates. Additional terms are added to the vector fields to account for time-varying curves and to enable normalization of the vector field in order to keep the resulting speed bounded. Simulation results demonstrate the method for tracking a variety of different closed curves and for following a family of patterns when time-varying elements are included.**

I. Introduction

Vector field control methods provide a rigorous framework for deriving control actions with provable stability guarantees for a large class of path following problems. In particular, vector field methods are well suited for constructing control laws that provide convergence to limit cycle behaviors that follow specified patterns or closed curves [1].

A variety of different approaches have been used to construct stable guidance vector fields for path tracking. Differential geometry can be used to define control decisions in a local Frenet-Serret frame that moves with the curve [2], [3]. The vector field emerges from regulation to the origin in the local frame. The method can break down in places along the curve where the Frenet-Serret frame is undefined. A second approach uses feedback control on course angle to create vector fields that converge to circular or straight line paths [4]. An alternative approach defines the control frame such that the state is always on the desired path while the origin of the frame is moved to the desired location [5], [6]. This approach enables complex patterns through open-loop specification of control actions, but converges slowly to the desired location. Finally, vector fields can be derived for classes of n-dimensional systems where the desired curve is defined as the intersection of $n-1$ (possibly time varying) surfaces with codimension equal to one [7]. That approach relies on the ability to specify surfaces with certain additional properties that yield the desired curve.

This paper describes a constructive method for deriving provably stable vector fields for a class of star shaped, three-dimensional curves. The method builds on the Lyapunov vector field approach described in Ref. 1. That work introduced a method for designing families of vector fields that are essentially globally asymptotically stable by construction to a circular limit cycle. The design method was extended to obtain other paths that are diffeomorphic to a circle while preserving global stability. In this paper the method is applied in cylindrical coordinates. A general mapping from actual coordinates to parametric coordinates is utilized such that the vector field is first derived for a circular pattern and then transformed back into actual coordinates while preserving stability. This paper extends the approach of Ref. 1 by allowing for time varying curves and by describing how formulation in cylindrical coordinates enables local approximation for tracking curves specified only by a discrete set of control points.

II. Guidance Vector Field Control onto a Stationary Unit Circle at Unit Height

Let $\mathbf{p} \in \mathbb{R}^3$ be the position of the vehicle to be controlled so that the control input $\mathbf{u}$ is the velocity of the vehicle, i.e. $\dot{\mathbf{p}} = \mathbf{u}$. A guidance vector field $\mathbf{g}(\mathbf{p})$ is defined such that setting $\mathbf{u} = \mathbf{g}(\mathbf{p})$ drives the vehicle into a limit cycle that corresponds to motion around a circle of unit radius at unit distance above the ground in a plane with normal in the z-direction. Classes of vector fields can be derived for this problem [1], but only a single particular vector field is considered here.

This paper considers vector fields expressed in cylindrical coordinates. Let $\mathbf{x} = [r, \theta, z]^T$ be the cylindrical coordinates of the vehicle and let the vehicle velocity expressed in the cylindrical coordinate system be $\dot{\mathbf{p}} = [v_r, v_\theta, v_z]^T$. Note that

$$\dot{\mathbf{x}} = \begin{bmatrix} \dot{r} \\ \dot{\theta} \\ \dot{z} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1/r & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \dot{\mathbf{p}} = \mathbf{M}_r \cdot \dot{\mathbf{p}} = \begin{bmatrix} v_r \\ v_\theta/r \\ v_z \end{bmatrix} \tag{1}$$

Define the vector field

$$\mathbf{g}(\mathbf{p}) = \mathbf{g}(\mathbf{x}) = \begin{bmatrix} -k_r\alpha(r-1) \\ 2\alpha r \\ -k_z\alpha(z-1) \end{bmatrix} \tag{2}$$

where $k_r > 0$ and $k_z > 0$ are constants that can be used to adjust the strength of the contraction of the vector field to the curve and $\alpha = \alpha(\mathbf{x}) > 0$ is a normalization term that is selected so that the vector field produces bounded magnitudes.

Consider the Lyapunov function

$$V(\mathbf{x}) = (r-1)^2 + (z-1)^2. \tag{3}$$

Taking its derivative yields

$$\dot{V}(\mathbf{x}) = \nabla V(\mathbf{x}) \cdot \dot{\mathbf{x}} = \nabla V(\mathbf{x}) \cdot \mathbf{M}_r \cdot \dot{\mathbf{p}} = \nabla V(\mathbf{x}) \cdot \mathbf{M}_r \cdot \mathbf{u}. \tag{4}$$

978-1-4577-1096-4/12/$26.00 ©2012 AACC

Expanding the gradient and setting the input to the vector field, i.e. setting $\mathbf{u} = \mathbf{g}(\mathbf{x})$ gives

$$\dot{V}(\mathbf{x}) = 2\,[r-1 \quad 0 \quad z-1]\begin{bmatrix} -k_r\alpha\cdot(r-1) \\ 2\alpha \\ -k_z\alpha\cdot(z-1) \end{bmatrix} \tag{5}$$

which reduces to

$$\dot{V} = -2k_r\alpha(r-1)^2 - 2k_z\alpha(z-1)^2 \tag{6}$$

If the magnitude of the vector field and the initial position are bounded, then the normalization term $0 < \alpha_0 \leq \alpha(\mathbf{x}) < \infty$ is bounded. Finally, let $k = \min(k_r, k_z)$ so the derivative of the Lyapunov function can be bounded such that

$$\dot{V} \leq -2k\alpha_0(r-1)^2 - 2k\alpha_0(z-1)^2 = -2k\alpha_0 V. \tag{7}$$

Equation (7) shows that by the comparison principle [8] the Lyapunov function has exponential convergence to zero. This means the vehicle position converges to limit cycle motion on a circle with unit radius and unit height, with transverse velocity $v_\theta = 2\alpha$.

III. Tracking an Expanding Star Curve

This section assumes that the desired curve is an expanding star pattern, meaning each point on the curve can be described by a radius $r_D(\theta, t)$ and height $z_D(\theta, t)$ that are unique functions of the angle θ. It is assumed that the star curve is differentiable so that $r_D^{'} = \partial r_D/\partial\theta$ and $z_D^{'} = \partial z_D/\partial\theta$ are bounded for all θ and for all t. Further, it is assumed that the speed of the expansion is bounded, i.e. that $\partial r_D/\partial t \leq w_r < \infty$ and $\partial z_D/\partial t \leq w_z < \infty$. This pattern is diffeomorphic to a unit circle at unit height in a plane normal to the z-direction. The control approach is to transform the vehicle position into a new coordinate system where the desired curve is the unit circle at unit height, calculate the vector field in this coordinate system, and then transform it back into the original coordinate system with additional terms as needed to account for the expansion.

Given the point $\mathbf{x} = [r, \theta, z]^T$ in "physical coordinates", let $\mathbf{q} = [\rho, \phi, \zeta]^T$ be a point in "parametric coordinates" determined by the transformation

$$\mathbf{q}(\mathbf{x}, t) = \begin{bmatrix} \rho \\ \phi \\ \zeta \end{bmatrix} = \begin{bmatrix} r/r_D(\theta, t) \\ \theta \\ z/z_D(\theta, t) \end{bmatrix} \tag{8}$$

which is easily inverted

$$\mathbf{x}(\mathbf{q}, t) = \begin{bmatrix} r \\ \theta \\ z \end{bmatrix} = \begin{bmatrix} \rho\cdot r_D(\phi, t) \\ \phi \\ \zeta\cdot z_D(\phi, t) \end{bmatrix}. \tag{9}$$

The time derivatives of the different vectors are related by

$$\dot{\mathbf{x}} = \frac{\partial\mathbf{x}}{\partial\mathbf{q}}\cdot\dot{\mathbf{q}} + \frac{\partial\mathbf{x}}{\partial t} \tag{10}$$

where the Jacobian of this inverse transformation is

$$\mathbf{J} = \frac{\partial\mathbf{x}}{\partial\mathbf{q}} = \begin{bmatrix} r_D & \frac{r}{r_D}r_D^{'} & 0 \\ 0 & 1 & 0 \\ 0 & \frac{z}{z_D}z_D^{'} & z_D \end{bmatrix} \tag{11}$$

and

$$\frac{\partial\mathbf{x}}{\partial t} = \begin{bmatrix} \rho\frac{\partial r_D}{\partial t} \\ 0 \\ \zeta\frac{\partial z_D}{\partial t} \end{bmatrix} = \begin{bmatrix} \frac{r}{r_D}\frac{\partial r_D}{\partial t} \\ 0 \\ \frac{z}{z_D}\frac{\partial z_D}{\partial t} \end{bmatrix} \tag{12}$$

where the dependence of r_D and z_D on θ (ϕ) and t were dropped for convenience.

The control approach here is to define the vector field in parametric equations so that we can make $\dot{\mathbf{q}} = \mathbf{M}_q \cdot \mathbf{g}(\mathbf{q})$ where the vector field $\mathbf{g}(\mathbf{q})$ is defined by (2). In order to achieve this the control input needs to be

$$\mathbf{u} = \mathbf{M}_r^{-1}\dot{\mathbf{x}} = \mathbf{M}_r^{-1}(\mathbf{J}\cdot\mathbf{M}_q\cdot\mathbf{g}(\mathbf{q}) + \frac{\partial\mathbf{x}}{\partial t}). \tag{13}$$

To see that this control law tracks the expanding curve, consider the Lyapunov function

$$\begin{aligned} V(\mathbf{x}, t) &= \frac{1}{r_D^2}(r - r_D)^2 + \frac{1}{z_D^2}(z - z_D)^2 \\ &= (\rho-1)^2 + (\zeta-1)^2 = V(\mathbf{q}(\mathbf{x}, t)). \end{aligned} \tag{14}$$

Taking its derivative gives

$$\begin{aligned} \dot{V}(\mathbf{q}) &= \nabla_q V \cdot \left[\frac{\partial\mathbf{q}}{\partial\mathbf{x}}\cdot\dot{\mathbf{x}} + \frac{\partial\mathbf{q}}{\partial t}\right] \\ &= \nabla_q V \cdot \left[\frac{\partial\mathbf{q}}{\partial\mathbf{x}}\cdot\mathbf{M}_r\cdot\mathbf{u} + \frac{\partial\mathbf{q}}{\partial t}\right]. \end{aligned} \tag{15}$$

Substituting (13) into (15) gives

$$\dot{V}(\mathbf{q}) = \nabla_q V \cdot \left[\mathbf{M}_q\cdot\mathbf{g}(\mathbf{q}) + \mathbf{J}^{-1}\frac{\partial\mathbf{x}}{\partial t} + \frac{\partial\mathbf{q}}{\partial t}\right]. \tag{16}$$

From (8)

$$\mathbf{J}^{-1} = \frac{\partial\mathbf{q}}{\partial\mathbf{x}} = \begin{bmatrix} \frac{1}{r_D} & -\frac{r}{r_D^2}r_D^{'} & 0 \\ 0 & 1 & 0 \\ 0 & -\frac{z}{z_D^2}z_D^{'} & \frac{1}{z_D} \end{bmatrix} \tag{17}$$

and

$$\frac{\partial\mathbf{q}}{\partial t} = \begin{bmatrix} -\frac{r}{r_D^2}\frac{\partial r_D}{\partial t} \\ 0 \\ -\frac{z}{z_D^2}\frac{\partial z_D}{\partial t} \end{bmatrix}. \tag{18}$$

Combining (12), (17), and (18),

$$\begin{aligned} \mathbf{J}^{-1}\frac{\partial\mathbf{x}}{\partial t} + \frac{\partial\mathbf{q}}{\partial t} &= \begin{bmatrix} \frac{1}{r_D} & -\frac{r}{r_D^2}r_D^{'} & 0 \\ 0 & 1 & 0 \\ 0 & -\frac{z}{z_D^2}z_D^{'} & \frac{1}{z_D} \end{bmatrix}\begin{bmatrix} \frac{r}{r_D}\frac{\partial r_D}{\partial t} \\ 0 \\ \frac{z}{z_D}\frac{\partial z_D}{\partial t} \end{bmatrix} \\ &\quad + \begin{bmatrix} -\frac{r}{r_D^2}\frac{\partial r_D}{\partial t} \\ 0 \\ -\frac{z}{z_D^2}\frac{\partial z_D}{\partial t} \end{bmatrix} \\ &= \begin{bmatrix} \frac{r}{r_D^2}\frac{\partial r_D}{\partial t} \\ 0 \\ \frac{z}{z_D^2}\frac{\partial z_D}{\partial t} \end{bmatrix} + \begin{bmatrix} -\frac{r}{r_D^2}\frac{\partial r_D}{\partial t} \\ 0 \\ -\frac{z}{z_D^2}\frac{\partial z_D}{\partial t} \end{bmatrix} \\ &= \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}. \end{aligned} \tag{19}$$

Substituting (19) back into (16)

$$\dot{V}(\mathbf{q}) = \nabla_q V \cdot \mathbf{M}_q \cdot \mathbf{g}(\mathbf{q}). \tag{20}$$

Expanding the terms gives

$$\dot{V}(\mathbf{x}) = 2\left[\rho - 1 \quad 0 \quad \zeta - 1\right]\begin{bmatrix} -k_r\alpha(\rho-1) \\ 2\alpha \\ -k_z\alpha(\zeta-1)\end{bmatrix} \tag{21}$$

so

$$\begin{aligned}\dot{V} &= -2k_z\alpha(\rho-1)^2 - 2k_r\alpha(\zeta-1)^2 \\ &\leq -2k\alpha_0[(\rho-1)^2 + (\zeta-1)^2] \\ &= -2k\alpha_0 V.\end{aligned} \tag{22}$$

As before, $0 < \alpha_0 \leq \alpha < \infty$ is bounded and $k = \min(k_z, k_r)$, so the Lyapunov function has exponential convergence to zero and the vehicle state converges to the limit cycle behavior $r = r_D$ and $z = z_D$.

A. *Desired Speed*

Recall that the parameter α was introduced in order to set the desired speed of the vehicle. The vector field equation (2) can be rewritten

$$\mathbf{g}(\mathbf{x}) = \alpha\begin{bmatrix} -k(r-1) \\ 2r \\ -k(z-1)\end{bmatrix} = \alpha \cdot \mathbf{h}(\mathbf{x}). \tag{23}$$

Using this new description the control law (13) can be rewritten as

$$\mathbf{u} = \alpha \cdot \mathbf{v}_1 + \mathbf{v}_2 \tag{24}$$

where

$$\mathbf{v}_1 = \mathbf{M}_r^{-1}\mathbf{J}\mathbf{M}_q\mathbf{h}(\mathbf{q}) \tag{25}$$

and

$$\mathbf{v}_2 = \mathbf{M}_r^{-1}\frac{\partial \mathbf{x}}{\partial t}. \tag{26}$$

The speed (squared) of the vehicle given this control law is

$$\begin{aligned}\|\mathbf{u}\|^2 = \mathbf{u}^T\mathbf{u} &= (\alpha\cdot\mathbf{v}_1 + \mathbf{v}_2)^T(\alpha\cdot\mathbf{v}_1+\mathbf{v}_2) \\ &= \alpha^2(\mathbf{v}_1^T\mathbf{v}_1) + 2(\mathbf{v}_1^T\mathbf{v}_2)\alpha + \mathbf{v}_2^T\mathbf{v}_2.\end{aligned} \tag{27}$$

Setting the vehicle speed to the desired speed u_{des} and rearranging gives a quadratic in α

$$\alpha^2(\mathbf{v}_1^T\mathbf{v}_1) + 2(\mathbf{v}_1^T\mathbf{v}_2)\alpha + (\mathbf{v}_2^T\mathbf{v}_2 - u_{des}^2) = 0. \tag{28}$$

It is assumed that the vehicle speed is always fast enough to track the curve, meaning we must have $u_{des}^2 > \mathbf{v}_2^T\mathbf{v}_2$. In this case the solutions to (28) are real with positive root

$$\alpha = \frac{-\mathbf{v}_1^T\mathbf{v}_2 + \sqrt{(\mathbf{v}_1^T\mathbf{v}_2)^2 - (\mathbf{v}_1^T\mathbf{v}_1)(\mathbf{v}_2^T\mathbf{v}_2 - u_{des}^2)}}{\mathbf{v}_1^T\mathbf{v}_1} > 0. \tag{29}$$

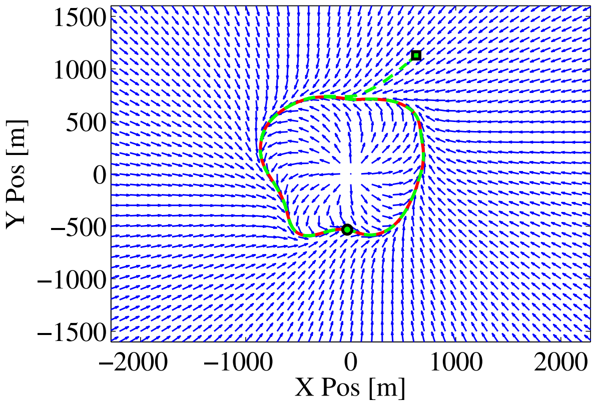

Fig. 1. Vector field and example robot path for a curve defined by a radial Fourier basis set.

B. *Determining the Star Curve*

In some cases the curve to be followed is given by a set of points that lie on the curve, rather than an analytical expression. In that case an analytical approximation can be derived that can be used to calculate the vector field. Start with a set of control points in polar coordinates $\mathbf{p_i} = [r_i, \theta_i, z_i]^T$ that lie on the desired curve. In order to derive an approximation, assume the radius r_i and height z_i can be be written as the weighted sum of basis functions of θ_i, i.e. let $r_i(\theta_i) = \sum_{j=1}^N \gamma_j f_j(\theta_i)$.

Because the approximation is defining a star pattern, the basis functions should be chosen to yield unique values for $\theta_i \in [0, 2\pi)$ and to be differentiable at $\theta_i = \{0, 2\pi\}$. Therefore, the Fourier basis make a natural choice for the functions $f_i(\theta)$. This work assumes the radius r_i (likewise the height z_i) is related to the angle θ_i by

$$r(\theta_i) = r_0 + \sum_{j=1}^{n}(\alpha_j \cdot \sin j\theta_i + \beta_j \cdot \cos j\theta_i) \tag{30}$$

where r_0, α_j, and β_j are weights and n is the number of functions used in the fit, and can be used to trade off between smoothness and error at the control points. Since (30) is linear in the weights, they can be approximated by a least squares fit over all control points to yield the curve

$$\hat{r}(\theta) = \hat{r}_0 + \sum_{j=1}^{n}(\hat{\alpha}_j \cdot \sin j\theta + \hat{\beta}_j \cdot \cos j\theta). \tag{31}$$

The fitting process described by (31) can be applied to the desired radius and heights separately to give analytical approximations of the complete star curve. Figure 1 shows the result of a fit with n = 20 (41 total weights) for a curve with about 200 control points at the same height $z_D = 0$. The curve fit was used as the desired curve to derive the vector field over the entire space (Figure 1). One example path based on this vector field is also given.

One of the strengths of the vector field control approach described by (13) is that it only requires knowledge of the desired curve at the particular value of θ corresponding to the

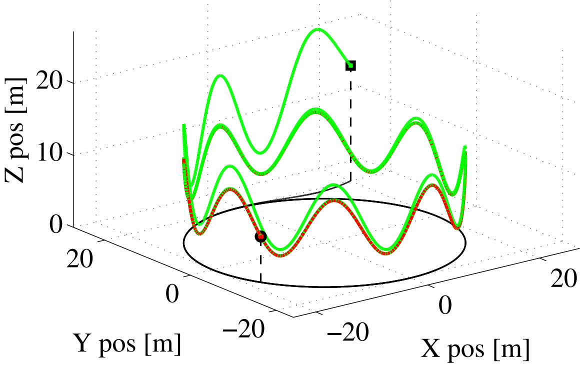

Fig. 2. Robot path

current robot position. As a result, the complete curve fitting described by (31) does not need to be performed to derive a control action. Rather, it is sufficient to perform a local fit around the control points near $\theta_i = \theta$.

IV. Examples

This section demonstrates how the control approach can be applied to various closed curves or patterns. In each case the vector field is normalized such that $u_{des} = 20$.

A. Sine Wave Height

The first example defines a curve with constant radius, but spatially varying height. The desired curve is specified by the desired radius $r_D(\theta) = 20$ and height $z_D = 8 + 4\sin 8\theta$. The red curve in Figure 2 denotes the resulting desired curve. The Cartesian position coordinates of the robot as a function of time are given in Figure 3a. The planar components of motion vary slowly with large amplitude while the height component varies faster. The height follows a sinusoidal spatial pattern which yields a temporal pattern that is more jagged. The exponential convergence of the Lyapunov function in this case can be seen in Figure 3b. The three dimensional path of the robot as it converges to the curve is given in green in Figure 2.

B. Sine Wave Radius

The next example defines a curve with spatially varying radius and constant height: $r_D = 20 + 4\sin 6\theta, z_D = 8$. The desired curve is in red in Figure 4. The position of the robot as a function of time is given in Figure 5a. In this example the planar position components show the beat frequencies due to the spatial variations in radius while the heights converges exponentially in about 10 seconds. Figure 5b shows the exponential convergence of the Lyapunov function to zero. The path of the robot is given in green in Figure 4.

C. Corkscrew

By adding time variations to the specification of the desired curve, vector fields are created that stabilize to a family of

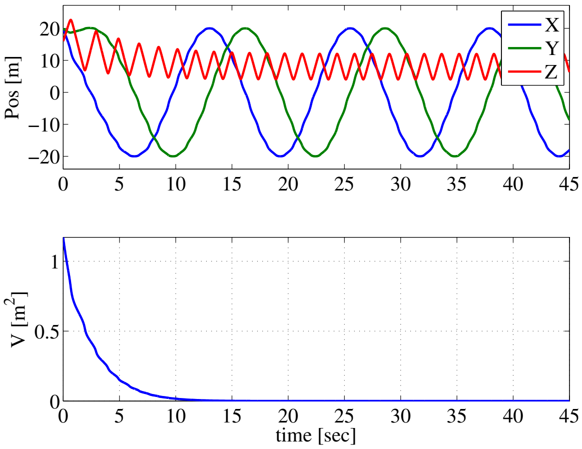

Fig. 3. a.) Position and b.) Lyapunov function versus time.

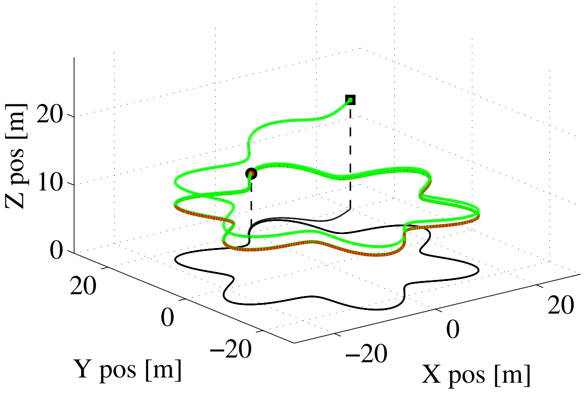

Fig. 4. Robot path

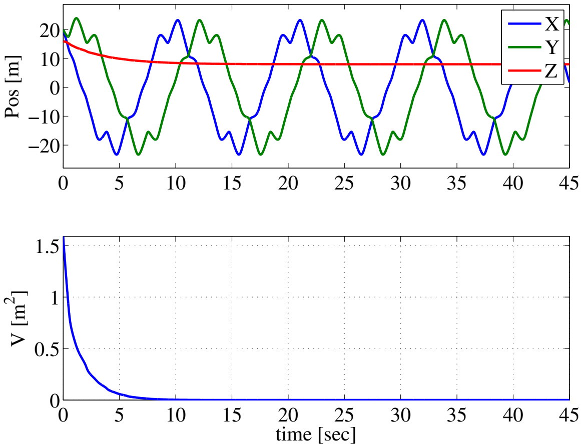

Fig. 5. a.) Position and b.) Lyapunov function versus time.

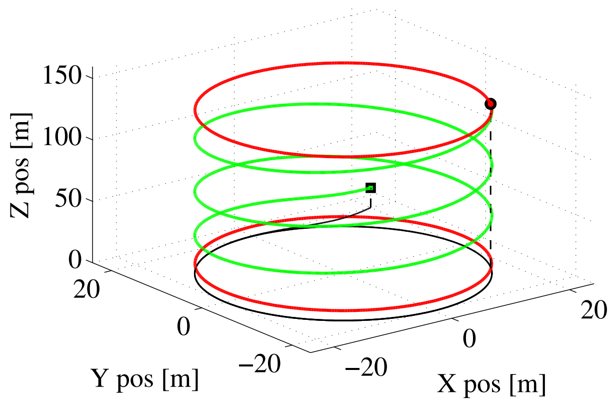

Fig. 6. Robot path

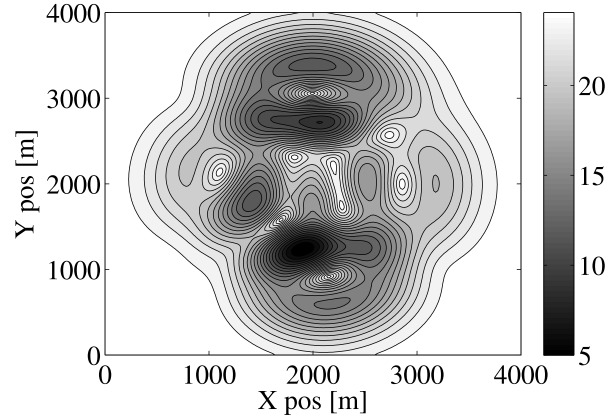

Fig. 8. Normal expansion speed.

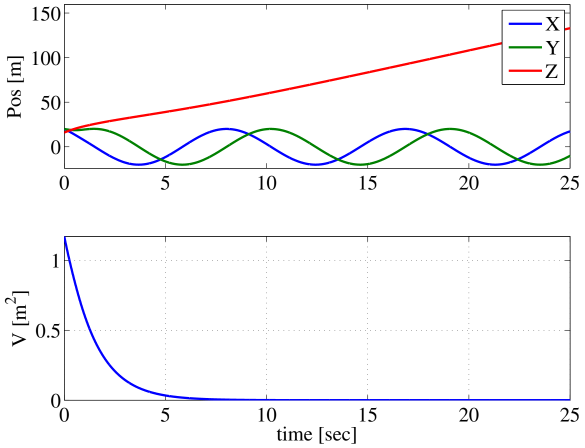

Fig. 7. a.) Position and b.) Lyapunov function versus time.

curves or that achieve other motion patterns. Convergence to helical curves is achieved by letting the desired height change with time. In that case, if the desired curve's vertical velocity $\frac{\partial z_D}{\partial t} = v_{z0}$ is constant and the vector field speed u_{des} is constant, the helix angle of the steady state motion is $\tan^{-1} \frac{\beta}{\sqrt{1-\beta^2}}$ where $\beta = v_{z0}/u_{des}$.

Figure 6 and Figure 7 show the results where the desired curve is given by $r_D = 20$ and $z_D = 8 + 5t$. The two red circles at the bottom and top of Figure 6 denote the desired curve at the start and final times, respectively. The green curve represents the robot path. The path converges to the desired circle initially and then produces the helical shape as it moves upward. Figure 7a shows the position of the robot versus time. The planar position components oscillate while the height increases at a constant rate.

D. Isotropic Outward Expansion

This final example integrates the various elements of the control approach described here: tracking unusually shaped curves, tracking an expanding curve, tracking a curve only specified by a set of control points. The particular example consists of a robot tracking a curve that expands outward in the ground plane with normal velocity that is a function of the position of the curve. Propagation of an outward expanding curve or wave is difficult in general, so this work uses ordered upwind methods to approximate the expansion over a grid [9]. These methods have been shown to converge to exact solutions as the grid spacing goes to zero, otherwise they interpolate values in an upwinding manner that preserves causality and approximates the viscosity solution of the underlying Hamilton-Jacobi-Bellman equation [9].

In this example a curve expands isotropically outward with a speed $v_{curve}(\mathbf{p})$ that is a function of position. Figure 8 shows a mesh plot of the speed field over the environment. The speed varies from 5 to 25 m/s over the course of the domain, with several peaks of local maximum and minimum. Ordered upwind methods [9], [10] are used to track the expanding curve over a 201 by 201 grid with 20 m spacing. At a given control sample time, the desired curve is taken to be the AcceptedFront [9], [10] of the expansion process, which is the set of points that the curve would reach just at or before the sample time.

The ordered upwind methods propagate expansion of the desired curve outward normal to the curve. At a given position $\mathbf{p}_D$ on the curve, this corresponds to the velocity vector in cylindrical coordinates of

$$\mathbf{v}_{wave} = \frac{v_{curve}(\mathbf{p}_D)}{\sqrt{r_D^2 + (r_D^{'})^2}} [r_D \quad r_D^{'} \quad 0]^T. \tag{32}$$

For (13) only the radial component is needed, so we take

$$\frac{\partial r_D}{\partial t} = \frac{v_{curve}(\mathbf{p}_D) \cdot r_D}{\sqrt{r_D^2 + (r_D^{'})^2}}. \tag{33}$$

In this example the robot is given a constant speed of u_{des} = 40 m/s so it can track the expanding curve at all times.

Figure 9 shows snapshots of the curve and robot path at different times during the simulation. These snapshots start at time $t = 63.25$ s which is approximately halfway into the run. At this time the robot has already converged to the desired curve and tracks it as it expands outward. The red curve in each

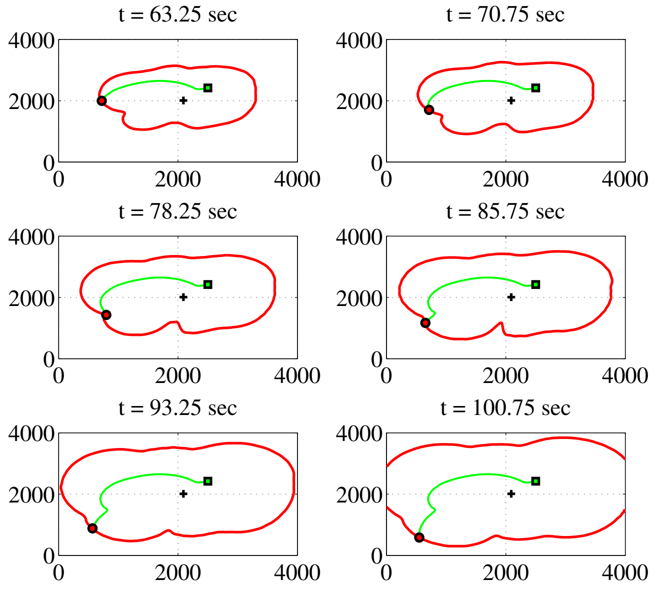

Fig. 9. Desired curve and robot path at different times.

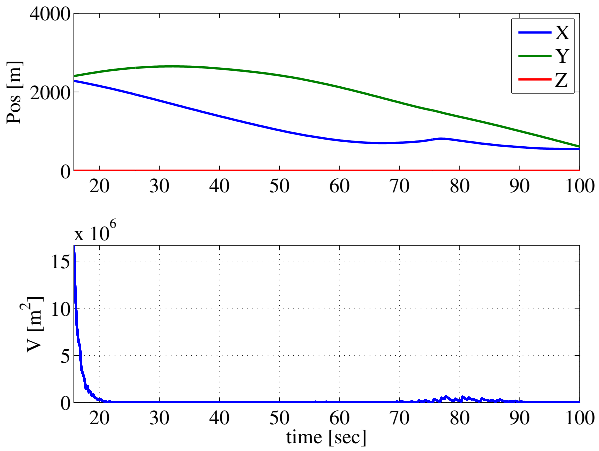

Fig. 10. a.) Position and b.) Lyapunov function versus time.

subplot shows the desired curve. The desired curve started as a circle of radius $r_D = 1$ at the cross marker in the center of the plots. The non-uniform speed field (Figure 8) leads to the irregular shaped curves. Of note are the notches in the curves, especially the one in the lower left portion of the curve, which the robot passes through at $t = 78.25$ s.

Figure 10 shows the robot position and Lyapunov function versus time. In this case the Lyapunov function first converges toward zero like the previous examples. At approximately $t = 60$ s the Lyapunov function and tracking errors grow. This behavior is due to the approximation of the desired curve around the sharp turn visible in the first two subplots of Figure 9. The ordered upwind methods use a grid spacing of 20 m, creating discrete jumps in the location of the curve. As a result, the effective radial velocity is different from the value used for $\partial r_D/\partial t$ and position error is induced. The error is greatest around the notch in the curve because the jump can be large as the curve pinches in on itself. By construction, the approximated desired curve is always a differentiable star shape even in cases where the ordered upwind expansion process creates fronts that would violate this assumption.

V. Conclusion

This paper presented a vector field control approach for tracking expanding, three-dimensional star curves. A method was presented that utilized a transformation from physical to "parametric" cylindrical coordinates in order to construct an essentially globally stable control law. Additional terms were added to the control law to account for time varying expansion of the desired curve. While the method described here is similar in spirit to the one described in Ref. 7, this method differs in that a constructive process is described that preserves the stability properties of a basic control algorithm while providing exponential convergence (as opposed to the asymptotic convergence demonstrated in [7]). Simulation results showed how the approach could be used to follow a variety of patterns and closed curves.

The results presented here focused only on the stability of the vector field itself and did not consider additional algorithms needed for use by real platforms. In particular, real platforms will have acceleration limits that constrain the curvature of feasible paths and require tracking control to bring the robot onto the vector field. Designing vector fields that satisfy given acceleration constraints is an open question, even when the desired curve is known to satisfy them.

References

[1] D. A. Lawrence, E. W. Frew, and W. J. Pisano, "Lyapunov vector fields for autonomous uav flight control," *AIAA Journal of Guidance, Control, and Dynamics*, vol. 31, no. 5, pp. 1220–1229, September 2008.

[2] R. A. Wise and R. T. Rysdyk, "Uav coordination for autonomous target tracking," in *Collection of Technical Papers - AIAA Guidance, Navigation, and Control Conference*, vol. 5. AIAA, August 2006, pp. 3210–3231.

[3] R. Rysdyk, "Unmanned aerial vehicle path following for target observation in wind," *Journal of Guidance, Control, and Dynamics*, vol. 29, no. 5, pp. 1092–1100, 2006.

[4] D. R. Nelson, D. B. Barber, T. W. McLain, and R. W. Beard, "Vector field path following for miniature air vehicles," *IEEE Transactions on Robotics and Automation*, vol. 23, no. 3, pp. 519–529, 2007.

[5] S. Hernandez and D. A. Paley, "Three-dimensional motion coordination in a spatiotemporal flowfield," *IEEE Transactions on Automatic Control*, vol. 55, no. 12, pp. 2805–2810, 2010.

[6] L. DeVries and D. A. Paley, "Multi-vehicle control in a strong flowfield with application to hurricane sampling," Portland, OR, 2011.

[7] V. M. Goncalves, L. C. A. Pimenta, C. A. Maia, B. C. O. Dutra, and G. A. S. Pereira, "Vector fields for robot navigation along time-varying curves in n-dimensions," *IEEE Transactions on Robotics*, vol. 26, no. 4, pp. 647–659, 2010.

[8] H. Khalil, *Nonlinear Systems*. Upper Saddle River, NJ: Prentice Hall, 2001.

[9] J. A. Sethian and A. Vladimirsky, "Ordered upwind methods for static hamilton-jacobi equations: Theory and algorithms," *SIAM Journal on Numerical Analysis*, vol. 41, no. 1, pp. 325 – 363, 2003.

[10] E. W. Frew, "Combining area patrol, perimeter surveillance, and target tracking using ordered upwind methods," in *Proceedings of the 2009 IEEE International Conference on Robotics and Automation*, Kobe, Japan, May 2009, pp. 3123–3128.