Proceedings of the 34th
Conference on Decision & Control
New Orleans, LA - December 1995

FP09 4:50

On line path following by recursive spline updating

R. Frezza G. Picci

Dipartimento di Elettronica e Informatica
Università di Padova, Italy
frezza@dei.unipd.it

Abstract

In this paper we address the problem of tracking an unknown contour by commanding the translation and rotational velocities of a vehicle. We propose an on-line contour estimator based on a recursive spline approximation and we discuss its applicability to a model-based predictive control strategy.

1. Introduction

An autonomous vehicle moving on a plane must follow a contour described by some unknown curve Γ in the plane. The contour may describe the boundary of some unknown obstacle or one of the borders of an unknown road that the vehicle must follow. In most applications it is particularly important that the vehicle avoids impact with the border. A sensor (typically a TV camera) provides measurements of the distance of the curve from the x axis of the moving frame (we assume this coincides with the optical axis of the camera) at N points placed at fixed distances $\{x_1, \ldots, x_N\}$ from the origin of the vehicle-fixed coordinate system $\{x, y\}$ (the origin conventionally coincides with the optical center of the camera). See fig. (1).

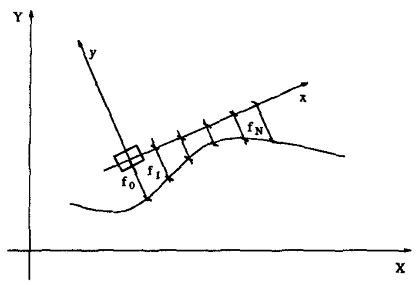

Figure 1: Path–following of an unknown trajectory in a plane.

The above is a prototype problem of autonomous navigation and has been studied by many [?] but mostly in an ad hoc context. A solution which uses the paradigmes of modern control and estimation theory was first proposed by Dickmanns and his group [2, 3]. Dickmann's work is based on setting up a stochastic state-space model of the dynamics of the unknown contour as seen by an observer sitting on the moving vehicle-fixed frame. An extended Kalman filter built from this model serves to estimate on–line the contour. The contour estimate is then tracked by applying a suitable state feedback control law.

The idea is quite appealing and has shown to work well for tracking borders of highways. These borders however satisfy rather stringent geometric conditions which were suitably incorporated in the a priori state-space model of the contour. In fact it is not clear how to generalize Dickmann's approach to more general type of contours and it seems worthy trying to build a modeling philosophy which will work for more general situations. This is the main motivation of this paper.

Let the sensor provide N noisy measurements $\{y_1, \ldots, y_N\}$ of the N "future" distances $\{f_1, \ldots, f_N\}$ of the vehicle from the contour, say

$$y_k = \frac{f_k}{x_k} + n_k \quad k = 1, \ldots, N$$

where x_k is the distance from the origin of k-th measurement line and the n_k's are uncorrelated white measurement noise processes. From these data the on-board computer must reconstruct on-line a current local model of the chunk of curve seen on the image plane. The reconstruction should be continuously updated based on both the current measurements and on some a priori model of the contour.

Assume further that the controller drives the vehicle by imposing the translational (v) and angular velocity (ω) of the camera-fixed frame. Then the on-board local reconstruction of the environment changes depending on the imposed motion. In particular the contour model permits to predict at each instant t a set of N future distances $\{\hat{f}_1, \ldots, \hat{f}_N\}$ of the vehicle from the contour corresponding to the chosen control actions. This framework is clearly reminiscent of "predictive control" [5], altough it does not necessarily rely on building an actual predictor.

A sensible "predictive control" strategy to avoid impact with the border and at the same time to achieve careful tracking would be to regulate not just the current distance f_0 but as many future distances f_k as possible. In fact one would like to know how many future f_k it wold be possible to regulate or track by our controller.

[0]Support by the Italian Space Agency with Grant # RS–106 is gratefully acknowledged.

In this paper we propose a finite dimensional contour estimator algorithm based on a recursively updated spline approximation of the contour. The dynamical model of the contour is called the *splinator.* We begin here a study of the attainable performances of systems of this kind.

2. Problem formulation

At time t the camera mounted on the vehicle "sees" a chunck of the unknown contour Γ. We assume the contour is described by the graph of some function

$$w = \Gamma(z) \quad 0 \leq z \tag{1}$$

in the inertial frame ($\{z, w\}$), and that the chunck seen by the camera is

$$y = \gamma(x, t) \quad 0 \leq x \leq L, \tag{2}$$

where $\{x, y\}$ is the coordinate frame fixed with the vehicle (and the camera). The N noisy measurements of the "future" distances $\mathbf{f} = [f_1, \ldots, f_N]'$ of the vehicle from the contour, provided by the camera (see fig. (1)), are then essentially sample-values in space of the function γ , i.e.

$$\begin{aligned} y_k(t) &= \frac{\gamma(x_k, t)}{x_k} + n_k(t) \quad k = 1, \ldots, N \\ &= \frac{f_k(t)}{x_k} + n_k(t). \end{aligned} \tag{3}$$

The contour Γ seen from the vehicle-fixed reference frame moves as a rigid object with traslational and angular velocities $\mathbf{v}(t)$ and $\omega(t)$ (here $\mathbf{v}(t)$ is expressed relative to the moving frame). Because of the rigid motion of the vehicle, the chunck (2) of the contour Γ seen on the image plane changes in time. The rigid motion constraint implies that a generic point P of the contour Γ of inertial coordinates $[z, \Gamma(z)]'$, seen from the moving frame, moves according to the transformation

$$\begin{aligned} \begin{bmatrix} \dot{x}(t) \\ \dot{y}(t) \end{bmatrix} &= \frac{d}{dt} R(t) \left\{ \begin{bmatrix} z \\ \Gamma(z) \end{bmatrix} - \begin{bmatrix} z_0(t) \\ w_0(t) \end{bmatrix} \right\} \\ &= \begin{bmatrix} 0 & -\omega(t) \\ \omega(t) & 0 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} - \begin{bmatrix} v_x(t) \\ v_y(t) \end{bmatrix} \end{aligned}$$

where $[x, y]'$ are the coordinates of P with respect to the camera frame,

$$R(t) := \begin{bmatrix} \cos\psi(t) & \sin\psi(t) \\ -\sin\psi(t) & \cos\psi(t) \end{bmatrix}$$

$\omega(t) := \dot{\psi}(t)$, $\mathbf{v}(t) = [v_x(t) \; v_y(t)]'$ is the traslational velocity of the vehicle and $[z_0(t) \; w_0(t)]'$ is the position of the origin of the moving frame at time t. In the following we shall assume that the vehicle can move only by keeping the instantaneous traslation velocity vector always directed in the direction of the x axis i.e. that $v_y(t)$ is identically zero. If we imagine the x axis as being the longitudinal symmetry axis of the vehicle, this corresponds to the usual constraint of "no lateral slipping". Henceforth we shall write v_x as v.

Now for small enough rotation angles ψ it will generally happen that the contour Γ can still be described by an explicit functional relation (namely $y = \gamma(x, t)$) also in the moving coordinate frame. An explicit equation for the dynamics of $\gamma(x, t)$ can be obtained easily from the previous equation by just substituting the first scalar equation

$$\dot{x}(t) = -\omega(t)\gamma(x, t) - v(t) \tag{4}$$

into the second, which reads

$$\frac{d}{dt}\gamma(x, t) = \frac{\partial\gamma(x, t)}{\partial t} + \frac{\partial\gamma(x, t)}{\partial x}\dot{x}(t) = \omega(t)x(t) \tag{5}$$

so that one obtains the following partial differential equation

$$\frac{\partial\gamma(x, t)}{\partial t} = \omega(t)x - \frac{\partial\gamma(x, t)}{\partial x}(\omega(t)\gamma(x, t) + v(t)) \tag{6}$$

This equation describes the dynamics of the contour in the coordinate frame fixed with the vehicle. We shall show in this paper how, on the basis of (6), one can estimate the unknown contour on-line. Note that this equation is nonlinear and may show finite escape time effects. Intuitively, one should expect this phenomenon to occur when the rigid motion has twisted the original contour enough so that an explicit description ($y = \gamma(x, t)$) of the curve in the mobile frame is no longer possible. In this spirit equation (6) could well be called a *Riccati-type* PDE since it generalizes the classical well-known Riccati equation for the motion of a homogeneous straight line under rotation around the origin. Similar (although finite dimensional) Riccati-type equations have been obtained by Ghosh in his work on perspective systems [4].

Equation (6) describes an *infinite dimensional* dynamical system with inputs $v(t)$ and $\omega(t)$. Note that the "state" of this system is not simply the contour chunck $\gamma(x, t)$ for $x \in [0, L]$. A simple first order finite differences approximation of equation (6)

$$\begin{aligned} \gamma(x, t + dt) = \gamma(x, t) - \omega x dt + \\ +(\gamma(x + dx, t) - \gamma(x, t))(v - \omega\gamma(x, t))dt \end{aligned} \tag{7}$$

shows that to integrate forward in time (6), the knowledge of $\gamma(x, t)$ $0 \leq x \leq L$ and of the inputs is not sufficient. In particular, for this approximation of (6), it is clear that to determine $\gamma(x, t + dt)$ for $x \in [0, L]$ one needs to know $\gamma(x, t)$ for $x \in [0, L + dx]$. Therefore, to compute $\gamma(x,\tau)$ for any $\tau > t$ in $x \in [0, L]$ one needs to know $\gamma(x,t)$ for $x \in [0, +\infty)$. This is, clearly, impossible. Therefore, it is necessary to introduce some assumptions on the shape of the unknown contour for $x > L$ in inertial coordinates.

We may, to start, assume some regularity or smoothness; for example that the curve is $\mathcal{C}^2$ (twice continuously differentiable), or, as in [3] that its curvature is a piecewise linear function of the arclength.

Since the contour Γ is unknown, the a priori model should be a stochastic model embodying the a priori knowledge of the shape, possibly with specification of the statistical uncertainty parameters. A general model class is

$$\frac{d}{dz}\xi(z) = A\xi(z) + Bn(z) \qquad (8a)$$

$$\Gamma(z) = C\xi(z) \quad z \geq 0 \qquad (8b)$$

where $\xi(z)$ is a state vector of suitable dimension, $n(z)$ is spatial white noise and A, B, C are constant matrices. A particular case of (8) is obtained by describing the contour as a random walk (of appropriate order) in the variable z. In the following we shall mostly refer to the random walk model. Note that although one may have a priori informations on which parameters A, B, C are appropriate to describe a given class of contours, the variance of the noise driving the state model is in general poorly known and needs to be adjusted experimentally.

It is easy to see that equation (6) can be integrated forward in time in the strip $x \in [0, L]$, $t \geq 0$ if the inputs and the following boundary conditions are assigned

$$\begin{cases} \gamma(x,0) = \gamma_0(x) & x \in [0,L] \\ \frac{\partial\gamma(L,t)}{\partial x} = \delta_L(t) & t \geq 0 \end{cases} \qquad (9)$$

which represent a possible "state space" for the dynamical system (6). The problem is, however, infinite dimensional and, since the sensor provides only a finite dimensional vector of N noisy measurements at each time, it is convenient to transform it to a finite dimensional one.

3. B-splines

We model the chunck of the contour Γ seen by the on board camera at time t with a cubic B-spline

$$s(x,t) = \sum_{i=1}^{N+2} p_i(t) B_i(x) \quad 0 \leq x \leq L, t \geq 0.$$

The $B_i(x)$ functions are known in the literature with the name of basic splines [1]. They are cubic polynomials with compact support on $[x_{i-1}, x_{i+3}]$ which generate a partition of unity on the nodes $[x_1, \ldots, x_N]$.

Once the nodes sequence $[x_1, \ldots, x_N]$ is given the basic spline functions $B_{i,k}(x)$ of order k can be computed by the following recursive formula

$$\begin{aligned} B_{i,k}(x) &= \frac{x - x_i}{x_{i+k-1} - x_i} B_{i,k-1}(x) + \\ &\quad + \frac{x_{i+k} - x}{x_{i+k} - x_{i+1}} B_{i+1,k-1}(x) \end{aligned}$$

with the initial condition

$$B_{i,1}(x) = \begin{cases} 1 & x_i \leq x < x_{i+1} \\ 0 & otherwise \end{cases}$$

In this paper, whenever a basic spline is written without its order k explicit then $k = 3$. In the textbooks usually $B_{i+1}(x)$, $B_{i+2}(x)$ and $B_{i+3}(x)$ are the only three basic cubic spline functions different from zero in the node x_i. For a simpler notation we shall instead shift the index of the basic spline functions from $(i+1)$ to i.

The local spline approximation of the contour interpolates the N values $\mathbf{f}(t) = [\gamma(x_1,t), \ldots, \gamma(x_N,t)]$ so that for $i = 1, \ldots, N$

$$\begin{aligned} \gamma(x_i,t) &= p_i(t)B_i(x_i) + p_{i+1}(t)B_{i+1}(x_i) + \\ &\quad + p_{i+2}(t)B_{i+2}(x_i) \end{aligned}$$

which can be written in matrix form as

$$\mathbf{f} = \mathbf{B}\mathbf{p} \qquad (10)$$

where $\mathbf{p} = [p_1, \ldots, p_{N+2}]$ is the vector of coefficients of the spline and $\mathbf{B}$ is the following $N \times (N+2)$ tridiagonal matrix

$$\mathbf{B}_{i,j} = [B_j(x_i)] \quad i = 1, \ldots, N; j = 1, \ldots, N+2.$$

Clearly, the interpolating conditions alone are not sufficient to determine the $N+2$ spline coefficients p_i. Two more conditions are necessary. One can, for example, assign the first and second derivative of the spline at the end node x_N or the second derivatives of the spline at both end-points x_1 and x_N. We shall assign the two extra conditions *dynamically* on the basis of the a priori model of the contour in an inertial reference frame.

The dynamics of the *Splinator*

The *splinator* is a nonlinear *finite dimensional* dynamical system, whose states, inputs and outputs are, respectively, the spline coefficients $\mathbf{p}$, the velocities v and ω and the values $\mathbf{f}$.

The splinator describes the lateral dynamics of the vehicle with respect to the unknown contour. In particular, writing relationship (6) at the N interpolating points $\mathbf{x} = [x_1, \ldots, x_N]^T$ we get a system of N first order differential equations describing the dynamics of the N outputs $\mathbf{f}$, for $i = 1, \ldots, N$

$$\frac{df_i(t)}{dt} = -\omega(t)x_i + s'(x_i,t)(v(t) - \omega(t)f_i(t)) \quad (11)$$

where $s'(x_i,t) = \sum_{j=1}^{N+2} B_j'(x_i)p_j(t)$.

A-priori model in inertial coordinates
Two more dynamic equations are determined from the a priori stochastic model of the contour. For example, since the shape of the contour beyond $x_N = L$ is completely unknown, the local "a priori" model could be meant to describe the time evolution of the derivative of the spline approximation $s'(x_N,t)$ at the end node $x_N = L$. An "a priori" model could be a second order random walk. Denoting $\xi_1 := s'(L,t)$, we have in this case

$$\begin{cases} \dot{\xi}_1 = \xi_2 v \\ \dot{\xi}_2 = \nu \end{cases} \quad (12)$$

where $\xi_2 \simeq s''(L,t)$ and ν is white noise of appropriate variance. More generally, we may want to take into account other kinds of uncertainties in the curve like spline approximation error, measurement errors, uncertainties in the velocities v and ω etc.. In this case, one may consider a more general linear uncertainty model like the following

$$\begin{cases} \xi_1 = \sum_{i=1}^{N+2} c_{i,1}p_i \\ \xi_2 = \sum_{i=1}^{N+2} c_{i,2}p_i \\ \dot{\xi}_1 = \nu_1 \\ \dot{\xi}_2 = \nu_2 \end{cases} \quad (13)$$

In any case this leads to $N+2$ interpolation conditions of the form

$$\begin{bmatrix} f_1 \\ \vdots \\ f_N \\ \xi_1 \\ \xi_2 \end{bmatrix} = \begin{bmatrix} & & \mathbf{B} & \\ & & & \\ c_{1,1} & \cdots & \cdots & c_{N+2,1} \\ c_{1,2} & \cdots & \cdots & c_{N+2,2} \end{bmatrix} \begin{bmatrix} p_1 \\ \vdots \\ \vdots \\ p_{N+2} \end{bmatrix}$$
$$= \bar{\mathbf{B}}\mathbf{p}. \quad (14)$$

where the $N+2 \times N+2$ matrix $\bar{\mathbf{B}}$ must be invertible. The dynamic equations of the splinator can, then, be determined by combining

$$\dot{\mathbf{p}} = \bar{\mathbf{B}}^{-1} \begin{bmatrix} \dot{f}_1 \\ \vdots \\ \dot{f}_N \\ \dot{\xi}_1 \\ \dot{\xi}_2 \end{bmatrix}. \quad (15)$$

with the kynematic differential equation (11) for $\mathbf{f}$ and with the "stochastic" end point conditions (13).

4. Observability and contour estimation

The splinator is a nonlinear finite dimensional dynamical system and observability is a local property which depends on the state $\mathbf{p}$. Local observability at $\mathbf{p}_0$ of the splinator implies that any infinitesimal change of the state $\mathbf{p}$ in a neighborhood of $\mathbf{p}_0$ can be detected in the outputs $\mathbf{f}$ or in their time derivatives.

It is easy to show that, as long as the translation velocity $v(t)$ is not null, the state of the splinator is locally observable from the inputs and the outputs at any point $\mathbf{p} \in \Re^{N+2}$.

The splinator is a nonlinear finite dimensional dynamical system and observability is a local property which depends on the state $\mathbf{p}$. Local observability at $\mathbf{p}_0$ of the splinator implies that any infinitesimal change of the state $\mathbf{p}$ in a neighborhood of $\mathbf{p}_0$ can be detected in the outputs $\mathbf{f}$ or in their time derivatives.

It is easy to show that, as long as the translation velocity $v(t)$ is not null, the state of the splinator is locally observable from the inputs and the outputs at any point $\mathbf{p} \in \Re^{N+2}$.

By contradiction, let's assume that the splinator is locally unobservable around some point of $\Re^{N+2}$ and that $v(t) \neq 0$. Unobservability implies the existence of a nonzero vector of spline coefficients $\mathbf{p}(t)$ such that the observations $\mathbf{f} = \mathbf{B}\mathbf{p}$ are all zero at time t.

Then, equations (11) imply that to the unobservable vector $\mathbf{p}$ corresponds a contour whose measured values $\mathbf{f}$ satisfy

$$\dot{f}_i(t) = -\omega(t)x_i + v(t)s'(x_i,t) \quad i = 1, \ldots, N.$$

For this contour, the observability codistribution $\mathcal{C}$ can be easily computed, it is

$$\mathcal{C} = \begin{bmatrix} d\mathbf{f} \\ d\dot{\mathbf{f}} \end{bmatrix} = \begin{bmatrix} \mathbf{B} \\ v(t)\mathbf{B}' \end{bmatrix},$$

where d denotes the gradient with respect to $\mathbf{p}$. Since $v(t) \neq 0$, unobservability of $\mathbf{p}$ implies also $\mathbf{B}'\mathbf{p} = 0$. This is a contradiction since there is no non-trivial cubic spline which vanishes on the nodes together with its first derivative.

If, instead, $v(t) = 0$ then $d\dot{\mathbf{f}} = 0$ and all spline coefficient vectors $\mathbf{p}$ corresponding to contours satisfying $f_i(t) = s(x_i,t) = 0$ are locally unobservable. An example of an unobservable contour is shown in fig. (2).

An extended Kalman filter based on the splinator model can be applied to estimate on–line the unknown contour.

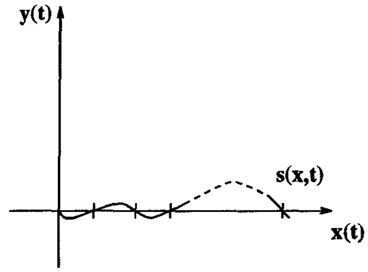

Figure 2: Example of "unobservable" contour if $v(t) = 0$.

5. Controllability of the contour

Studying the controllability of the splinator means investigating how one can change the shape of the contour, as seen by an observer sitting on the vehicle, controlling the inputs $v(t)$ and $\omega(t)$.

Equations (15) describing the dynamics of the splinator are affine in the controls v and ω and can be written in the following form

$$\dot{\mathbf{p}} = \bar{\mathbf{B}}^{-1}(\mathbf{g}_1(\mathbf{p})v + \mathbf{g}_2(\mathbf{p})\omega) \tag{16}$$

where

$$\mathbf{g}_1(\mathbf{p}) = \begin{bmatrix} \mathbf{B}'\mathbf{p} \\ 0 \\ 0 \end{bmatrix}$$

and $\mathbf{g}_2(\mathbf{p})$ is the $N+2$ dimensional vector defined by

$$\begin{cases} g_{2,i}(\mathbf{p}) = -x_i - s'(x_i,t)s(x_i,t) & i+1,\ldots,N \\ g_{2,i}(\mathbf{p}) = 0 \quad i = N+1, N+2. \end{cases}$$

The controllability distribution is determined by $\mathbf{g}_1$, $\mathbf{g}_2$ and their Lie-brackets. Clearly, the integral manifold of the distribution cannot be of dimension larger than N since the last two components of both $\mathbf{g}_1$ and $\mathbf{g}_2$ are zero. This means that ξ_1 and ξ_2 are not controllable. It means, as expected, that the inertial shape of the contour cannot be changed acting on v and ω.

6. Contour tracking

Three types of predictive control strategies will be discussed. The first is a simple myopic quadratic control based on minimizing the square distance of the current f_k's from a desired current sampled contour. The second is still a quadratic cost problem but computed by standard receding horizon predictive control on the predicted minimum distance of the vehicle from the contour. The third control is computed by imposing the vehicle trajectory on an appropriate path updated on-line.

Quadratic control

The values of the controls $v(t)$ and $\omega(t)$ are determined minimizing the following cost function

$$\min_{v(t),\omega(t)} J = \frac{1}{2}\left\{\sum_{i=1}^{M}(\hat{f}_i(t+\Delta t|t) - d)^2 + \rho(v(t) - v_d)^2\right\}$$

where d is the desired distance at which the vehicle should follow the contour, v_d is the desired velocity at which the vehicle should track the contour and $\hat{f}_i(t+\Delta t|t)$ are the prediction of the distances f_i at time $t+\Delta t$ given the estimate of the contour updated at time t

$$\hat{f}_i(t+\Delta t|t) = \sum_{j=1}^{N+2} B_j(x_i)\hat{p}_i(t+\Delta t|t).$$

In general, the goal $f_i(t+\Delta t) = d$ for $i = 1,\ldots,M$ cannot be achieved because the values f_i depend on the inertial shape of the contour which is not controllable. The controls tend to drive the vehicle so that

$$d = \sum_{i=1}^{M} \frac{\hat{f}_i(t+\Delta t|t)}{M}.$$

Predictive control

Let $\Sigma_I = [x,y,z]^T$ be an inertial coordinate frame that at time t coincides with the frame fixed with the camera and the vehicle. With respect to Σ_I the motion of the vehicle is described by the following differential equations

$$\begin{cases} \dot{x} = v\cos(\phi) \\ \dot{y} = v\sin(\phi) \\ \dot{\phi} = \omega \end{cases}$$

with initial conditions $\{x(t), y(t), \phi(t)\} = 0$. Let $r(x,y)$ be the minimum distance between a point of coordinates $[x,y,0]^T$ and the on-line estimate of the contour at time t which, in the coordinate frame Σ_i, is $[x, \sum_{i=1}^{N+2} B_i(x)\hat{p}_i(t|t), 0]^T$.

The values of the controls $v(t)$ and $\omega(t)$ are determined minimizing the following cost function

$$\min_{\substack{v(t),\, v(t+\Delta t),\ldots,\, v(t+k\Delta t) \\ \omega(t),\, \omega(t+\Delta t),\ldots,\, \omega(t+k\Delta t)}} W$$

where

$$\begin{aligned} W &= \sum_{i=1}^{k}(r(x(t+i\Delta t), y(t+i\Delta t)) - d)^2 + \\ &\quad + \rho_i(v(t+i\Delta t) - v_d) \end{aligned}$$

with

$$\begin{cases} x(t+i\Delta t) = \sum_{j=0}^{i-1} v(t+j\Delta t)\cos(\phi(t+j\Delta t))\Delta t \\ y(t+i\Delta t) = \sum_{j=0}^{i-1} v(t+j\Delta t)\cos(\phi(t+j\Delta t))\Delta t \\ \phi(t+i\Delta t)\sum_{j=0}^{i-1}\omega(t+j\Delta t)\Delta t \end{cases}.$$

Spline control

Let $s_c(x) = \sum_{j=1}^{M+2} B_j(x) r_j$ be the cubic B-spline that, in the coordinate frame fixed with the vehicle, satisfies at time t

$$\begin{cases} s_c(0) = 0 \quad s_c'(0) = 0 \\ s_c(x_i) = \sum_{j=1}^{N+2} B_j(x_i)\hat{p}_j(t|t) + d \quad i = 2, \ldots, M \\ s_c'(x_M) = \sum_{j=1}^{N+2} B_j'(x_M)\hat{p}_j(t|t) \end{cases}$$

where $M \leq N$. Since the heading of the vehicle at time t points in the direction of the tangent vector to the spline s_c in $x = 0$ and the spline has continuous second order derivative $d^2 s_c(x,t)/dx^2$, $s_c(x)$ is a feasible trajectory for the vehicle. The controls that drive the vehicle along this trajectry at longitudinal speed v_d are

$$\begin{cases} v(t) = v_d \\ \omega(t) = \frac{d^2 s_c(0)}{dx^2} v(t) \end{cases}.$$

Applying these controls the vehicle reaches the right configuration with respect to the unknown contour following the spline $s_c(x)$ which we call the "control spline". Note that, once the vehicle is in the right configuration, the control spline $s_c(x)$ coincides with the on–line estimate of the unknown contour $s(x,t)$, see fig. (3).

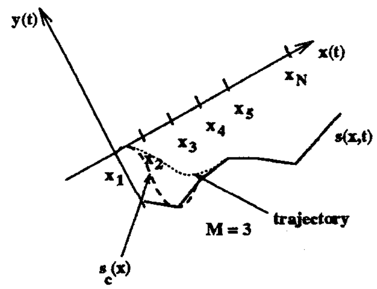

Figure 3: The vehicle must follow the contour $s(x,t)$ drawn with continuous line at distance $d = 0$. The dashed line is the spline $s_c(x)$ at time t with $M = 3$, and the dotted, thinner, line is the trajectory followed by the vehicle.

References

[1] C. De Boor, "A Practical Guide to Splines", Springer–Verlag, 1978.

[2] E. D. Dickmanns and V. Graefe, "Dynamic monocular machine vision, " *Machine Vision and Application,* vol. 1, pp. 223-240, 1988.

[3] E. D. Dickmanns and V. Graefe, "Applications of dynamic monocular machine vision, " *Machine Vision and Application,* vol. 1, pp. 241-261, 1988.

[4] B.K. Ghosh and E.P. Loucks, A perspective theory for motion and shape estimation in machine vision, *SIAM J. on Control and Optimiz.*, june 1995.

[5] M. Morari, Some Control Problems in the Process Industries, in *Essays on Control: Perspectives in the Theory and its Applications*, pp. 55–78, (survey paper persented at the second ECC Conference, Groningen, 1993), Birkhauser.