Interactive design of constrained variational curves

Wieger Wesselink [a,*], Remco C. Veltkamp [a,b]

[a] *Department of Computing Science, Technical University of Eindhoven, P.O. Box 513, 5600 MB Eindhoven, The Netherlands*

[b] *Department of Interactive Systems, CWI, Kruislaan 413, 1098 SJ Amsterdam, The Netherlands*

Abstract

A constrained variational curve is a curve that minimizes some energy functional under certain interpolation constraints. Modeling curves using constrained variational principles is attractive, because the designer is not bothered with the precise representation of the curve (e.g. control points). Until now, the modeling of variational curves is mainly done by means of constraints. If such a curve of least energy is deformed locally (e.g. by moving its control points) the concept of energy minimization is lost. In this paper we introduce deform operators with built-in energy terms. We have tested our ideas in a prototype system for modeling uniform B-spline curves. Through the use of widgets, the user can interactively modify the range of influence and other properties of the operators. Experiments show that these operators offer a very intuitive way of modeling.

Keywords: Variational curves; Energy functionals; Interactive design

1. Introduction

A variational curve is a curve that minimizes a given functional that represents the energy of the curve. Modeling these variational curves (variational modeling) is a very powerful way of modeling. It allows the design of smooth curves satisfying a number of interpolation constraints, in a very easy way. The associated energy functional typically depends on local properties of the curve such as the tangent vector and curvature. It is called the internal energy of the curve. A well known example of such a functional is the bend energy. Because of its physical background, the bend energy has been studied heavily, especially in the context of minimal energy splines (see (Birkhoff and de

---

* Corresponding author. Email addresses of the authors: {wieger,remco}@win.tue.nl.

Boor, 1965; Horn, 1983; Jou and Han, 1990; Brunnett et al., 1993; Brunnett and Kiefer, 1994)). Minimal energy spline curves interpolate a sequence of points while minimizing the bend energy. Other energy functionals have been studied as well (see (Hagen and Schulze, 1991; Roulier et al., 1991)).

There are two basic ways of interaction possible in the design process of a variational curve. In the first place, constraints can be imposed upon the curve. The most common constraints are point and normal interpolation. They can be used to sketch a rough outline of a curve. In the second place an internal energy functional can be chosen. With the choice of an internal energy the designer has a certain kind of global control over the shape of the curve.

These two ways of interaction are useful, but sometimes they are not flexible enough. Flexibility can be provided by operators that deform the curve locally. In the case of control point based curves (e.g. Bézier curves or B-spline curves) this can easily be done by repositioning a few control points. However, this does not fit in the concept of variational modeling, because the deformed curve does in general not minimize an energy functional.

This paper introduces some design operators that do not suffer from this problem. The effect of such an operator is defined by an energy functional. Since this functional depends on properties from outside the curve, it is called external. The internal and external energies E_{int} and E_{ext} are combined into one functional, the total energy E_{tot} of the curve, which has to be minimized:

$$E_{tot} = E_{int} + E_{ext}. \tag{1}$$

We have tested the concepts in a prototype system for modeling planar uniform B-spline curves. However, the idea is also applicable to Bézier curves or virtually any other class of curves. In the prototype system the design operators are represented by widgets that let the user specify a portion of the curve on which the operator works, a weight factor, and possibly some additional parameters. By increasing the weight factor the user can increase the effect of the operator. In order to improve the speed of the computation the energy functionals are approximated with quadratic expressions. This results in a quadratic minimization problem with linear constraints, which is solved using a standard method. Experiments show that these design operators can be useful tools for interactive variational modeling.

2. Internal energy

The internal energy of a curve is the part of the total energy (1) that depends only on properties of the curve itself. It determines the global shape of the curve. This section discusses the internal energy functionals that we use.

The internal energy that occurs most in the literature is the bend energy (E_{bend}):

$$E_{bend}(x) = \int \kappa^2(t) \|x'(t)\| \, dt, \tag{2}$$

with κ the curvature of the curve x.

Experience has shown that this functional yields smooth curves, so it seems an appropriate choice. However, if the length of the curve is not restricted in any way, an absolute minimum of (2) often does not exist (see (Birkhoff and de Boor, 1965)): in these cases the bend energy can be decreased by introducing large loops. In order to deal with this unwanted effect the length of the curve has to be restricted. A common way to do this is to combine the bend energy with another functional, the length of a curve. As this functional makes the curve resist stretching, it is called stretch energy ($E_{stretch}$):

$$E_{stretch}(x) = \int \|x'(t)\| \, dt. \tag{3}$$

We use a convex combination of the bend and the stretch energy:

$$E_{int}(x) = \alpha E_{bend}(x) + (1 - \alpha)E_{stretch}(x), \tag{4}$$

for some α in the interval $[0, 1]$.

For an interactive modeling program the expressions (2) and (3) are too complicated, since the evaluation and minimization of these expressions would be computationally too expensive. Therefore they are approximated with the following ones:

$$\bar{E}_{bend}(x) = \int \|x''(t)\|^2 \, dt, \tag{5}$$

$$\bar{E}_{stretch}(x) = \int \|x'(t)\|^2 \, dt. \tag{6}$$

These approximations are frequently used (see for instance (Celniker and Gossard, 1991)). The advantage of these approximations is that for e.g. B-spline curves they are quadratic functions of the control points (see Section 5). This is probably the reason for their popularity, since quadratic functions can be efficiently minimized. A disadvantage of the approximations (5) and (6) is that they are parameterization dependent. Generally speaking, the approximations will be worse when the absolute value of the derivative of the curve is fluctuating more. Note that for an arclength parameterized curve ($\|x'(t)\| \equiv 1$), the approximations would be exact.

3. Design operators

This section discusses two ways of interaction in variational curve modeling. The first one is to specify a number of constraints. This can be used to sketch a rough outline of the curve. Then we introduce a new way of modeling curves. The new way is to use external energy operators, so called because they produce the external part of the total energy (1) of a curve. The goal of these operators is to offer flexible ways of deforming a curve locally.

Fig. 1. Left: variational curve with point interpolation constraints. Right: variational curve with normal constraint.
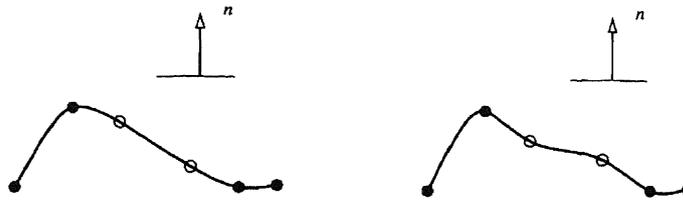
Fig. 2. Left: original curve and a director operator. Right: resulting constrained variational curve.

3.1. Constraints

We use two kinds of constraints: point and normal interpolation. The point interpolation constraint is illustrated in Fig. 1 (left), which shows a variational curve interpolating prescribed points. In order to prevent the curve from collapsing to a single point in its strive to minimize its energy, at least two specified endpoints should be interpolated.

A normal interpolation constraint forces the curve to attain a shape so as to have the prescribed normal at some point of the curve. This is illustrated in Fig. 1 (right), which shows a variational curve interpolating a prescribed normal.

3.2. External energy operators

We use three types of operators to add an energy term to the external energy functional: a director, a point attractor, and a curve attractor. These operators generate a corresponding energy term: E_{dir}, E_{pat}, and E_{cat} respectively.

3.2.1. Director

The director is a design operator that tries to push the tangent along a segment of the curve in a specified direction, as illustrated in Fig. 2. The small circles on the curve identify the parameter interval $[v, w]$ on the curve to which the operator is applied. Let n be a unit vector normal to the prescribed tangent vector. In any point on the curve the quantity $(x' \cdot n)^2$ measures the orthogonality between the tangent vector x' at the point and the vector n. So, the larger the inner product, the larger the deviation between prescribed and given tangent vector.

Therefore, a suitable energy term that tries to direct the tangent of the curve in the parameter interval $[v, w]$ in the specified direction is:

Fig. 3. Left: original curve and a point attractor. Right: resulting constrained variational curve.

Fig. 4. Left: original curve and a curve attractor. Right: resulting constrained variational curve.

$$E_{\text{dir}}(x) = \int_{v}^{w} (x'(t) \cdot n)^2 \, dt. \tag{7}$$

Minimizing this expression amounts to minimizing the total deviation between the prescribed and given tangent along the curve segment.

3.2.2. Point attractor

The point attractor is a design operator that pulls the curve towards a point, as illustrated in Fig. 3. The grey dot is the point that attracts the curve; the point on the curve to which the attractor applies is graphically represented by a circle.

Suppose we want the point on the curve with parameter t_0 be attracted by the point p. An energy term that has this effect is:

$$E_{\text{pat}}(x) = \|x(t_0) - p\|^2. \tag{8}$$

The minimal value of this expression is achieved when the curve's point $x(t_0)$ and p coincide. However, other energy terms will often prevent this. For example, if the point attractor will bend the curve, then the bend energy will increase. The bend energy component in the total energy of the curve will then compete with the energy of the point attractor to obtain a minimum, and will restrain the curve from interpolating p.

3.2.3. Curve attractor

The curve attractor is a design operator that pulls the design curve towards another curve. We restrict ourselves to polynomial attractor curves (for instance line segments). This is illustrated in Fig. 4. The grey line is the attractor which is applied to the curve segment between the two circles.

Let ℓ be the attractor curve. Suppose we want to let the part of the curve x in the interval $[v, w]$ be attracted by ℓ. An energy term that will attract this part of x to ℓ is:

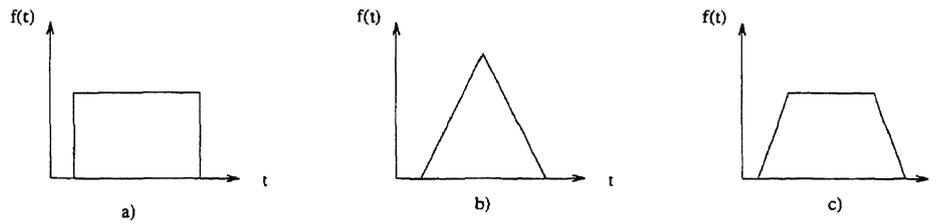$$E_{\text{cat}}(x) = \int_{v}^{w} \|x(t) - \ell(t)\|^2 \, dt. \tag{9}$$

Fig. 5. Three examples of weight functions.

Minimizing this expression amounts to minimizing the total deviation between the curve segment and the attractor curve. As for the point attractor, this energy term has to compete with other terms, which may restrain the design curve from interpolating the attractor curve.

4. Combining the energy terms

The various internal and external energy functionals from the previous sections are now added together to form the total energy of a curve. In this addition the external energy operator terms are weighted, in order to control their influence. We use two ways to do this. First, the energy functional of an external energy operator can be weighted as a whole by multiplying it with a constant factor. Second, if a finer control is needed a weight can be attached to each individual point of the curve.

A point attractor can only be weighted with a constant factor c, because the parameter interval of the curve that is affected by the point attractor consists of only one point:

$$\bar{E}_{\text{pat}}(x) = c \, \|x(t_0) - p\|^2. \tag{10}$$

Increasing c has the effect that the distance between $x(t_0)$ and p will be lowered.

The director and curve attractor can also be weighted in the second way. For this, a weight function f has to be included in the corresponding energy functionals:

$$\bar{E}_{\text{dir}}(x) = \int_v^w f(t) \, (x'(t) \cdot n)^2 \, dt, \tag{11}$$

$$\bar{E}_{\text{cat}}(x) = \int_v^w f(t) \, \|x(t) - \ell(t)\|^2 \, dt. \tag{12}$$

We have experimented with three types of piecewise linear weight functions, illustrated in Fig. 5. The constant weight function from Fig. 5a corresponds to multiplying the energy functional with a constant factor. The weight functions of Figs. 5b and 5c make the energy contributions more important in the middle of their parameter interval.

5. Computation

This section deals with the computational aspects of modeling with constraints and external energy functionals. A detailed analysis is given of how the different energy functionals can be computed and how the variation problem is solved. We start with giving a definition of B-spline curves, for these are the curves that we use in our prototype system.

5.1. B-spline curves

A nonuniform B-spline curve of degree d with knot sequence $\{t_0, \ldots, t_{m+2d-2}\}$ and control points $\{P_0, \ldots, P_{m+d-1}\}$ is defined as (see (Farin, 1993)):

$$x(t) = \sum_{i=0}^{m+d-1} P_i N_i^d(t),$$ (13)

where the basis functions N_i^d are defined by the de Boor recursion relations:

$$N_i^d(t) = \frac{t - t_{i-1}}{t_{i+d-1} - t_{i-1}} N_i^{d-1}(t) + \frac{t_{i+d} - t}{t_{i+d} - t_i} N_{i+1}^{d-1}(t),$$

$$N_i^0(t) = \begin{cases} 1 & \text{if } t_{i-1} \leqslant t < t_i, \\ 0 & \text{otherwise.} \end{cases}$$

The curve is defined over the domain $[t_{d-1}, \ldots, t_{m+d-1}]$. m is the number of curve segments if all knots in the domain are simple. A B-spline curve is called uniform if the spacing between the knots is constant ($t_i - t_{i-1} = h$ for all i).

In the rest of this section, the vector that contains the components of the control points of a B-spline curve will be denoted as $\mathcal{P}$:

$$\mathcal{P} = [P_{0x} \, P_{1x} \, \ldots P_{(m+d-1)x} \, P_{0y} \ldots P_{(m+d-1)y}]^T.$$ (14)

The subscripts x and y stand for the x- and y-coordinates respectively. The dimension of $\mathcal{P}$ will be denoted as r (so $r = 2m + 2d$).

5.2. Constraints

In case of a curve in B-spline representation, a point interpolation constraint $x(t_0) = p$ takes the following form :

$$\sum_{i=0}^{m+d-1} P_i N_i^d(t_0) = p.$$

Hence a point interpolation constraint consists of two equations $C_x^T \mathcal{P} = 0$ and $C_y^T \mathcal{P} = 0$, one for the x-coordinates and one for the y-coordinates, where C_x and C_y are suitably chosen r-vectors and $\mathcal{P}$ is the vector defined in (14). Note that these equations are

linear in $\mathcal{P}$, if t_0 is fixed. The value of t_0 is determined automatically as the value corresponding to the point of the curve closest to p.

A normal constraint $x'(t_0) \perp n$ takes the following form:

$$n_x \sum_{i=0}^{m+d-1} P_{ix} \dot{N}_i^d(t_0) + n_y \sum_{i=0}^{m+d-1} P_{iy} \dot{N}_i^d(t_0) = 0.$$

If t_0 is fixed, a normal constraint can be written in the form of the linear constraint $C^T \mathcal{P} = 0$, with C a suitably chosen r-vector. The value of t_0 is determined as the value corresponding to the point of the curve closest to the tail of the arrow representing the normal.

5.3. Computation of energy functionals

The energy functionals can be conveniently expressed in the vector $\mathcal{P}$. First it will be shown that all functionals are quadratic functions of $\mathcal{P}$ and then the integrals that actually have to be computed will be reviewed.

Substitution of the B-spline representation (13) into the bend energy functional (5) gives the following expression:

$$\bar{E}_{\text{bend}}(x) = \int_{t_{d-1}}^{t_{m+d-1}} \| x''(t) \|^2 \, dt = \sum_{i=0}^{m+d-1} \sum_{j=0}^{m+d-1} P_i \cdot P_j \int_{t_{d-1}}^{t_{m+d-1}} \ddot{N}_i^d(t) \ddot{N}_j^d(t) \, dt.$$

Substitution into the stretch energy functional (6) gives:

$$\bar{E}_{\text{stretch}}(x) = \int_{t_{d-1}}^{t_{m+d-1}} \| x'(t) \|^2 \, dt = \sum_{i=0}^{m+d-1} \sum_{j=0}^{m+d-1} P_i \cdot P_j \int_{t_{d-1}}^{t_{m+d-1}} \dot{N}_i^d(t) \dot{N}_j^d(t) \, dt.$$

These two energies can be written in matrix form as $\bar{E}_{\text{bend}}(x) = \mathcal{P}^T A_{\text{bend}} \mathcal{P}$ and $\bar{E}_{\text{stretch}}(x) = \mathcal{P}^T A_{\text{stretch}} \mathcal{P}$, which are quadratic in $\mathcal{P}$. The elements of the $r \times r$-matrices A_{bend} and A_{stretch} contain integrals of products of B-spline basis functions.

Substitution of the B-spline representation into the director energy functional (11) gives

$$\bar{E}_{\text{dir}}(x) = \int_v^w f(t)(x'(t) \cdot n)^2 \, dt$$

$$= \sum_{i=0}^{m+d-1} \sum_{j=0}^{m+d-1} (P_i \cdot n)(P_j \cdot n) \int_v^w f(t) \dot{N}_i^d(t) \dot{N}_j^d(t) \, dt.$$

Likewise, the point attractor energy functional (10) expands into

$$\bar{E}_{\text{pat}}(x) = c \, \|x(t_0) - p\|^2$$

$$= c \sum_{i=0}^{m+d-1} \sum_{j=0}^{m+d-1} P_i \cdot P_j N_i^d(t_0) N_j^d(t_0) - 2c \sum_{i=0}^{m+d-1} P_i \cdot p N_i^d(t_0) + c\,p \cdot p,$$

and the curve attractor energy functional (12) expands into

$$\bar{E}_{\text{cat}}(x) = \int_v^w f(t) \|x(t) - \ell(t)\|^2 \, dt$$

$$= \sum_{i=0}^{m+d-1} \sum_{j=0}^{m+d-1} P_i \cdot P_j \int_v^w f(t) N_i^d(t) N_j^d(t) \, dt$$

$$- 2 \sum_{i=0}^{m+d-1} P_i \cdot \int_v^w f(t)\ell(t) N_i^d(t) \, dt + \int_v^w f(t)\ell(t)^2 \, dt.$$

These energy functionals can be written as $\bar{E}_{\text{dir}}(x) = \mathcal{P}^{\mathsf{T}} A_{\text{dir}} \mathcal{P} + B_{\text{dir}}^{\mathsf{T}} \mathcal{P}$, $\bar{E}_{\text{cat}}(x) = \mathcal{P}^{\mathsf{T}} A_{\text{cat}} \mathcal{P} + B_{\text{cat}}^{\mathsf{T}} \mathcal{P} + c_{\text{cat}}$, and $\bar{E}_{\text{pat}}(x) = \mathcal{P}^{\mathsf{T}} A_{\text{pat}} \mathcal{P} + B_{\text{pat}}^{\mathsf{T}} \mathcal{P} + c_{\text{pat}}$. Again, the A's are $r \times r$-matrices, the B's are r-vectors, and the c's are constants. So, all the functionals are of the form:

$$E(x) = \mathcal{P}^{\mathsf{T}} A \mathcal{P} + B^{\mathsf{T}} \mathcal{P} + c. \tag{15}$$

5.3.1. Matrix elements

Now that we have seen that the energy functionals can be written as quadratic functions in $\mathcal{P}$, it is time to take a closer look at the matrices and A and vectors B that constitute these quadratic functions (15). Recall that we restrict ourselves to piecewise linear weight functions f in $\bar{E}_{\text{dir}}$ and $\bar{E}_{\text{cat}}$, and that the curve ℓ in $\bar{E}_{\text{cat}}$ is polynomial. With this in mind, a careful examination of the energy functionals learns that the computation of the matrices and vectors involves the computation of the following primitive functions:

$$\int t^k \dot{N}_i^d(t) \dot{N}_j^d(t) \, dt, \quad k = 0, 1,$$

$$\int \dot{N}_i^d(t) \dot{N}_j^d(t) \, dt,$$

$$\int t^k N_i^d(t) N_j^d(t) \, dt, \quad k = 0, 1, \tag{16}$$

$$\int t^k N_i^d(t) \, dt, \qquad k = 0, \ldots, \text{degree}(\ell) + 1,$$

for $0 \leqslant i, j \leqslant m+d-1$. Since the B-spline basis functions N_i^d are piecewise polynomial, these primitive functions have to be computed for every separate interval $[t_k, t_{k+1}]$.

A basis function N_i^d is in the interval $[t_k, t_{k+1}]$ a degree d polynomial with coefficients that depend on the knots $\{t_{k-d+1}, \ldots, t_{k+d}\}$. So the integrals (16) are also polynomials,

with coefficients that depend on these knots. We have computed these coefficients with Mathematica (see (Wolfram, 1991)) for the case $d = 3$. They consist of large sums of terms, each of them depending on the knots t_i.

Our prototype modeling system deals with the restricted class of uniform B-spline functions with simple knots $t_i = i$, $i = 0, \ldots, m + 4$. For this class of functions, the amount of computation can be reduced significantly. The integrals (16) only have to be computed for $0 \leqslant i, j \leqslant 3$, because of the following property of these uniform B-spline functions:

$$N_i^d(t) = N_{i-k}^d(t - kh), \quad k \in \mathbb{N}, \ k \leqslant i.$$

All these polynomials have to be computed only once. Once they are stored, the computation of the matrices A and vectors B is nothing more than substitution of the parameter values v and w in these polynomials.

5.4. Solving the variational problem

In the previous subsection we have seen that all the energy functionals that we use are quadratic functions of the control points of a curve. Therefore the total energy, which is a linear combination of these functionals, is also a quadratic function in $\mathcal{P}$. Furthermore we have seen that the interpolation constraints are linear in $\mathcal{P}$. So we have reduced the variational problem to a quadratic programming problem (see (Luenberger, 1989)) with linear constraints

$$\text{Minimize} \quad \mathcal{P}^T A \mathcal{P} + 2B^T \mathcal{P} + c \tag{17}$$

$$\text{subject to} \quad C \mathcal{P} = D, \tag{18}$$

where A is an $r \times r$-matrix, B is an r-vector, C is an $s \times r$-matrix, D is an s-vector, and c is a number (s is the number of constraints).

We solve this system in two steps. First the constraint equations (18) are used to eliminate some variables from the minimization problem (17). Then the resulting unconstrained quadratic minimization problem is solved using LU-decomposition.

6. Interactive design

We have implemented a prototype system with the constraints and external energy operators mentioned. The polynomial splines are represented in the uniform B-spline scheme. The functionality provided by the prototype system is the following. The user can:

- Determine the number of control points of the curve.
- Specify point interpolation constraints and normal constraints.
- Apply external energy operators of the types director, point attractor, and curve attractor.
- Determine the relative weight of the stretch and bend energy, by setting the value of α in equation (4).
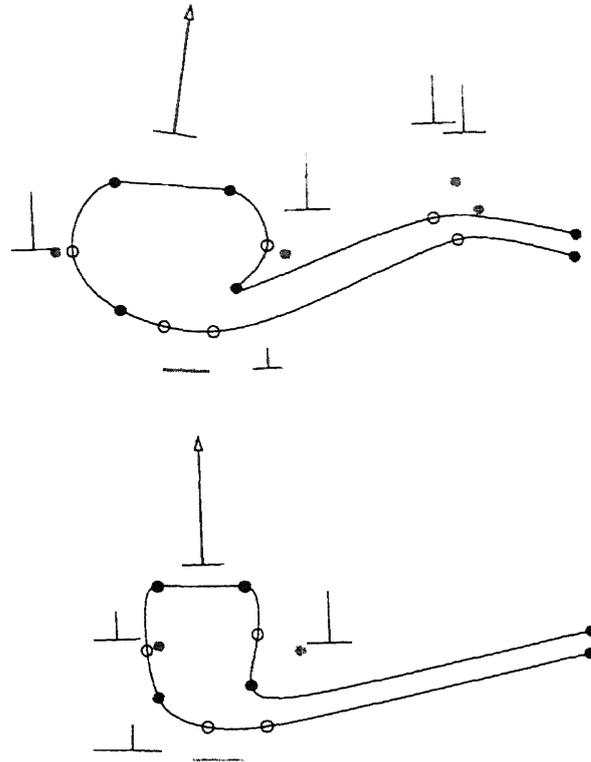
Fig. 6. Pipe-shaped constrained variational curves.

The point interpolation constraint is specified by pointing at the position of the point. The normal constraint is specified by pointing at the position of the tail of the arrow representing the normal vector, and positioning the tip of the arrow to adjust the direction of the vector. The interpolation point and the arrow can also be dragged around. The parameter values to which the point interpolation and the normal constraints apply are determined automatically as the value corresponding to the closest point of the curve, as mentioned in Section 5.2.

For the external energy operators the parameter values must be determined by the user. For the point attractor, the user determines the parameter value t_0 in equation (10) by indicating a point on the curve. The value of t_0 then becomes the parameter value corresponding to that point. The point on the curve is graphically represented by a circle (see Fig. 3). Similarly, for the director and the curve attractor, the user interactively determines the parameter interval bounds v and w in equations (11) and (12). The points on the curve corresponding with the interval bounds are graphically represented by circles (see Figs. 2 and 4).

The external operators are represented by upside-down T-shaped widgets (see Fig. 6), which can be manipulated to control their effect. The normal vector n of the director operator can be set by grabbing an endpoint of the widget and rotating the widget around

the center of its base. The direction of the base of the widget is the direction imposed on the curve segment to which the operator applies. The position of the point attractor is simply pointed at. The type of curve of the curve attractor is limited to a line. That line is set by positioning the endpoints independently.

Each external energy operator can be weighted by a step function, a hat function or a truncated hat function. The user selects which of these weight functions is in effect. The height of the stick perpendicular to the base is a measure of the strength of the operator, i.e. the maximal value of the weight function. The strength can be changed by grabbing the endpoint of the stick of the widget and adjusting its height. Other parameters of the weight functions are fixed.

We have combined the normal constraint with a point interpolation constraint, such that the normal constraint applies to the position on the curve that interpolates the point (point-normal interpolation constraint). It has an intuitive appeal to apply a normal constraint at a prescribed point that must be interpolated.

Likewise, we have combined the director and the line attractor with interpolation constraints. This is done in three ways: with a point interpolation constraint at the middle of the parameter interval, with a point-normal interpolation constraint at the middle of the parameter interval, and with a point interpolation constraint at both ends of the parameter interval. Again, it has an intuitive appeal to apply the operators around or between prescribed points that must be interpolated.

Fig. 6 shows an example of two curves designed with the constraints and the external energy operators described in this paper. The constraints and the operators provide local shape control, the "rest" of the shape is determined by minimizing the internal energy, while the total curve still minimizes an energy functional. The example shows two pipe-shaped curves. The top of the head is flattened with a director operator. The amount of bulging of the head is influenced by the point attractors at both sides. A line attractor is used to pull the bottom downwards. In the upper pipe, the two lines of the stem are bent with two point attractors. Varying the strengths of all these operators varies the shape of the pipes.

7. Concluding remarks

We have introduced a way to interactively model variational curves by means of operators that affect the total energy of the curve. These operators effectively control groups of control points. In this way we can deform the curve locally, while the energy minimizing principles remain satisfied. By contrast, the internal energy of a curve has a global effect on its shape. This gives additional flexibility over the standard way of variational modeling (with constraints only). We have tested our ideas in a prototype system for modeling uniform B-spline curves. The user can interactively modify the range of influence, the strength, and other properties of the operators, through the use of interaction widgets. Experiments show that these operators offer an intuitive way of modeling.

One may wonder what the actual physical meaning is of the constructed total energy of a curve. The bend energy component E_{bend} has a physical justification, although it is an approximation. Other internal energy functionals can be developed that have
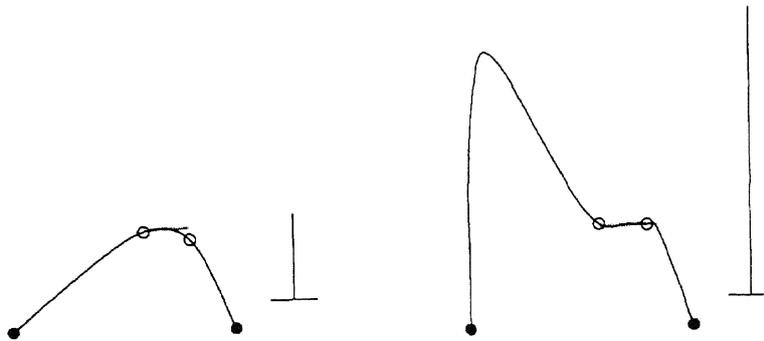
Fig. 7. Increasing the strength of the director of the left curve may lead to disturbance of the resulting curve, shown right.

some desired effect, see for example the metrics described in (Roulier et al., 1991). The external components E_{dir}, E_{pat}, and E_{cat} are designed heuristically so as to have the desired effect. The composition of the internal and external energy functionals is a rather arbitrary weighted sum of the individual components.

A recent development in image processing is formed by active contours (see for example (Kass et al., 1987) and (Williams, 1992)). Active contours are variational curves which are guided towards contours in an image, under the influence of certain "image forces". These image forces are strongly related to the external energy operators in this article. They are also used to deform a curve and they are also expressed with energy terms. An important difference is that these image forces depend on the image data and they cannot be used as a tool for interactive modeling. Another difference is that active contours are time-dependent. This results in an increase of complexity and therefore in most practical implementations the modeling curves are polylines.

The modeling by means of external energy functionals also has its limitations. When the weight factor of an external energy functional is increased to a very high level, the corresponding part of the curve will almost exactly meet the specifications of the operator. For instance, in the case of a director, this portion of the curve will become very flat. But, as is illustrated in the right side of Fig. 7, the neighborhood of this segment of the curve can be heavily disturbed. An explanation for this behavior is that the energy in this portion of the curve dominates the total energy of the curve so much, that the energy of the portions in the neighborhood is neglected. So the neighborhood of this portion will do anything to decrease the energy in this portion, even if this causes them to oscillate widely.

In principle, the weight factor of the attractor operators could be made negative in order to simulate repelling behavior. However, if the weight factor is too negative, the total energy of the curve is dominated by the negative term, and the curve tries to minimize its energy by shooting to infinity.

When the user applies an external energy operator to a segment of the curve between certain parameter values, the lengths of the curve segment corresponding to the parameter range before and after minimization are generally not the same. This is caused by the

fact that the curves are not arclength parameterized.

To alleviate these two problems we can use hierarchical spline curves. A hierarchical spline is the sum of a base curve and additional spline segments. The segments added to the base curve are typically used to model local detail. Each external energy operators could add a new spline segment to the hierarchical spline to provide the local refinement. However, this would make the final curve dependent of the order in which the operators are applied. In our current concept, the operators have no order dependent effect, since they affect one energy functional for the whole curve.

In the case of curve attractors, we have only experimented with straight line segments. Under normal circumstances this will be no severe restriction, because the curve will only try to approximate the attractor curve, while keeping the bend energy low. This means that the attractor curve will not be interpolated, unless the interpolation of it happens to coincide with a state of low energy of the curve. Therefore, straight line segments are sufficiently flexible.

Acknowledgement

References

Birkhoff G. and de Boor, C. (1965), Piecewise polynomial interpolation and approximation, in: Garabedian, H.L., ed., *Approximation of Functions*, Elsevier, Amsterdam, 164–190.

Brunnett, G., Hagen, H. and Santarelli, P. (1993), Variational design of curves and surfaces, Surveys Math. Industry 3, 1–27.

Brunnett, G. and Kiefer, J. (1994), Interpolation with minimal-energy splines, Computer Aided Design 26, 137–144.

Celniker, G. and Gossard, D. (1991), Deformable curve and surface finite-elements for free-form shape design, Comput. Graph. 25, 257–266.

de Boor, C. (1978), *A Practical Guide to Splines*, Springer, Berlin.

Farin, G. (1993), *Curves and Surfaces for CAGD*, 3rd ed., Academic Press, San Diego, CA.

Hagen, H. and Schulze, G. (1991), Variational principles in curve and surface design, in: Hagen, H. and Roller, D., eds., *Geometric Modelling — Methods and Applications*, Springer, Berlin.

Horn, B. (1983), The curve of least energy, ACM Trans. Math. Software 9, 441–460.

Jou, E. and Han, W. (1990), Minimal-energy splines I: plane curves with angle constraints, Math. Meth. Appl. Sci. 13, 351–372.

Kass, M., Witkin, A. and Terzopoulos, D. (1988), Snakes: active contour models, Internat. J. Comput. Vision 1, 321–331.

Luenberger, D.G. (1989), *Linear and Non-Linear Programming*, 2nd ed., Addison-Wesley, Reading, MA.

Press, W.H., Flannery, B.P., Teukolsky, S.A. and Vetterling, W.T. (1988), *Numerical Recipes in C, The Art of Scientific Computing*, Cambridge University Press.

Roulier, J. Rando, T. and Piper, B. (1991), Fairness and monotone curvature, in: Chui, C.K., ed., *Approximation Theory and Functional Analysis*, Academic Press, New York.

Schweikert, D.G. (1966), An interpolating curve using a spline in tension, J. Math. Phys. 45, 312–317.

Williams, D.J. and Shah, M. (1992), A fast algorithm for active contours and curvature information, CVGIP: Image Understanding 1, 14–26.

Wolfram, S. (1991), *Mathematica: A System for Doing Mathematics by Computer*, Addison Wesley, Reading, MA.