CS6491-2015 P3: Worm

Xiong Ding, Subhajit Das

1 Objective

Our objectives are the following:

1. Animate a morph between two input curves

2. Compute and display the inflation - minimal tube tangent to both input curves

2 Definitions and input

Given two Curves A and B both sharing common start and end points in a 3D space. In order to make the morph between these curves,we will compute the medial curve C by computing the medial points $M_1, M_2, M_3......M_N$. Subsequently, we would make transverse curves in the form of series of Parabolas spaced $D/2$ distance from each other, considering the two input curves can accommodate a sphere or ball of radius r in between them. Here, D is the diameter of the biggest circle tangential to both curves A and B.

3 Approach

3.1 Definition of curves and the tangent vector

Both curves A and B are controlled by 5 control points $\{ct_1, ct_2, ct_3, ct_4, ct_5\}$ individually to form a quintic Bézier curve, which has an explicit formula:

$$P(t) = (1-t)^4 P_0 + 4(1-t)^3 t P_1 + 6(1-t)^2 t^2 P_2 + 4(1-t) t^3 P_3 + t^4 P_4 \tag{1}$$

It's derivative is

$$P'(t) = 4(1-t)^3(P_1 - P_0) + 12(1-t)^2 t(P_2 - P_1) + 12(1-t)t^2(P_3 - P_2) + 4t^3(P_4 - P_3) \tag{2}$$

The curves are formed by computing the central points of the Ball with diameter D. The points on the curves are obtained by Linear Interpolation as below:

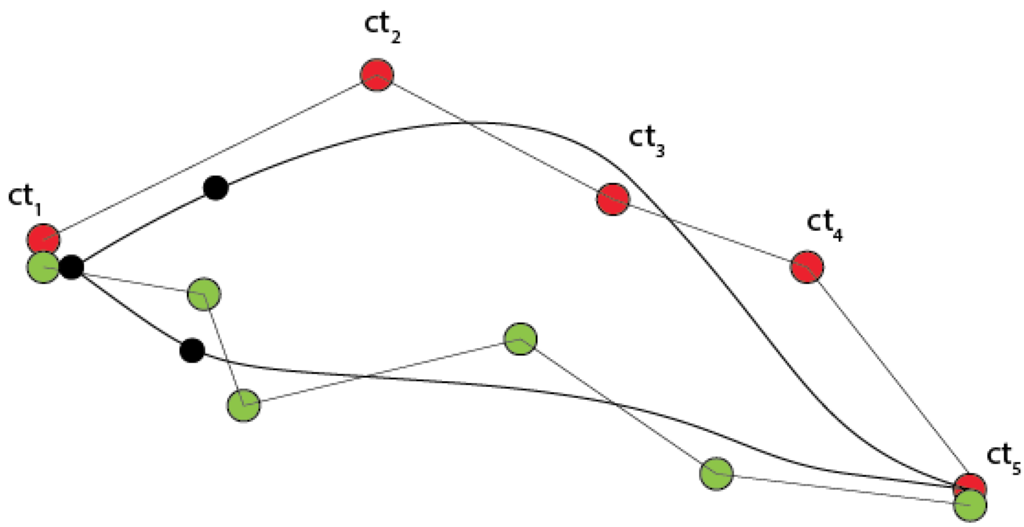

Figure 1: Shows two quintic bezier curve with 5 control points

$$\begin{aligned}
\mathrm{x} = \mathrm{L}(\mathrm{ct}_0, s, ct_1)\ y = L(ct_1, s, ct_2) \\
\mathrm{z} = \mathrm{L}(\mathrm{ct}_2, s, ct_3)\ w = L(ct_3, s, ct_4) \\
\mathrm{xx} = \mathrm{L(x,s,y)}\ \ \mathrm{yy} = \mathrm{L(y,s,z)}\ \mathrm{zz} = \mathrm{L(z,s,w)} \\
\mathrm{xxx} = \mathrm{L(xx,s,yy)} \\
\mathrm{yyy} = \mathrm{L(yy,s,zz)} \\
\mathrm{cPt} = \mathrm{L(xxx,s,yyy)}
\end{aligned} \tag{3}$$

where, ct_0, ct_1... etc are the control points of the curves and cPt is the point on the quintic Bezier Curve.

3.2 Calculate the First Median Point on the Medial Curve

Let A_0 be the start point of both curves A and B and medial curve C. From the parametric equation given in 1 we found two points A_1 and B_1 on curves A and B respectively very close to start point A_0. We first calculate the Angle Bisector Vector AB as below:

$$\begin{aligned}
\text{Vector A1} = \mathrm{V}(\mathrm{A}_0, A_1) \\
\text{Vector B1} = \mathrm{V}(\mathrm{A}_0, B_1) \\
\text{Vector BS} = \mathrm{A}(\mathrm{A}_1, B_1)
\end{aligned} \tag{4}$$

Now we need to compute the first median point M_0 on this Vector BS. That is given by the following equation,

$$M_0 = P(A_0, dis, BS) \tag{5}$$

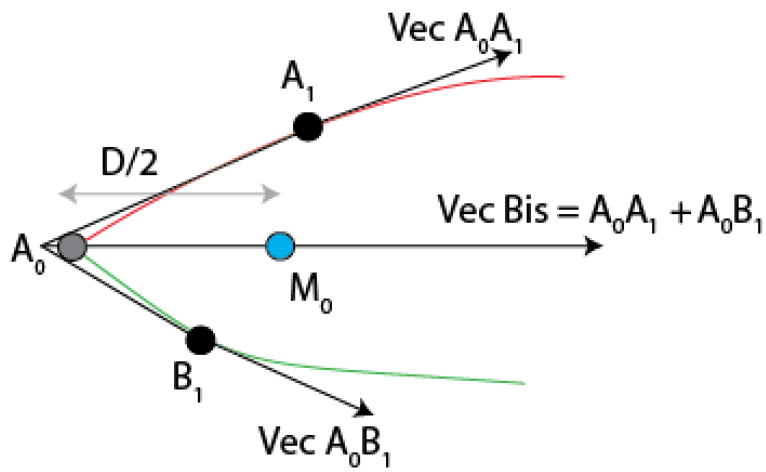

Figure 2: Shows two quintic bezier curve with 5 control points

where dis is the minimum value of $D/2$, magnitude($A1$) or magnitude($B1$) Here,D is the diameter of the fitting ball between curves A and B, such that it is tangential to both the curves.

3.3 Calculate the next median Point by guessing

Calculate the tangent vectors $tanV_1$ and $tanV_2$ at points A_1 and B_1 respectively as below:

$$c_1 = 4 * (1-t)^3 c_2 = 12 * (1-t)^2 * t c_3 = 12 * (1-t) * t^2 c_4 = t^3$$

$$\begin{aligned}
&\text{Vector } v1 = \text{V}(ct_0,ct_1) \\
&\text{Vector } v2 = \text{V}(ct_1,ct_2) \\
&\text{Vector } v3 = \text{V}(ct_2,ct_3) \\
&\text{Vector } v4 = \text{V}(ct_3,ct_4) \\
&\text{Vector } tanV_1 = (c_1\text{*v1}+c_2\text{*v2}) + (c_3\text{*v3} + c_4\text{*v4})
\end{aligned} \tag{6}$$

where $tanV_1$ is the calculated tangent vector. We can compute $tanV_2$, the respective tangent vectors at point A_1 and B_1 using aforementioned principles.

We compute the Bisection between the tangent vectors $tanV_1$ and $tanV_2$ and find two points on the bisector by the following calculation:

First, we get the normalized normal Vector $\underline{N}$:

$$\begin{aligned}
Vector\underline{\text{N}} &= \text{tanV}_1 \text{ x } tanV_2 \\
Vector\underline{\text{NN}} &= \text{N}/||\text{N}||
\end{aligned} \tag{7}$$

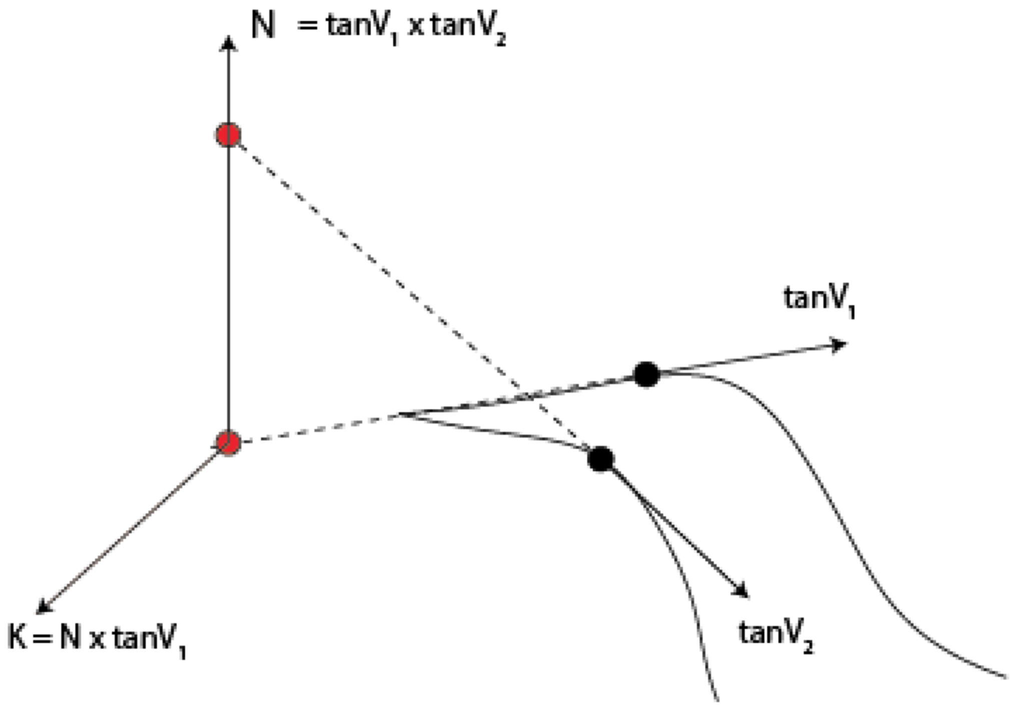

Figure 3: Shows the two bisection point obtained by computing the normal Vector N

We get the vector $A1B1$ between points A_1 and B_1. Also we compute the normalized perpendicular vector K by taking cross product between $tanV_1$ and vector N. This allows to compute the point $P1$ as below:

$$\begin{gathered} float \text{ val= } dot(\text{A1B1},K)/\text{dot}(tanV_1,K) \\ Point \text{ P1= } P(\text{B}_1,\text{val},tanV_2) \end{gathered} \tag{8}$$

We do the same to find the other point $P2$.

We add the points $P1$ and $P2$ to find the middle point $P3$. Also we add the normalized $tanV_1$ vector and normalized $tanV_2$ vector to get vector V. This helps us find $P4$ as below:

$$Point \text{ P4= } P(\text{P3,V}) \tag{9}$$

Thus the final points we obtained from computing the bisection are $P3$ and $P4$. We use these points to compute the Vector t between them and thus we get the guessed Median Point MG_1 as below:

$$Point\text{MG}_1 = \text{P}(M_0,D/2,\text{t}) \tag{10}$$

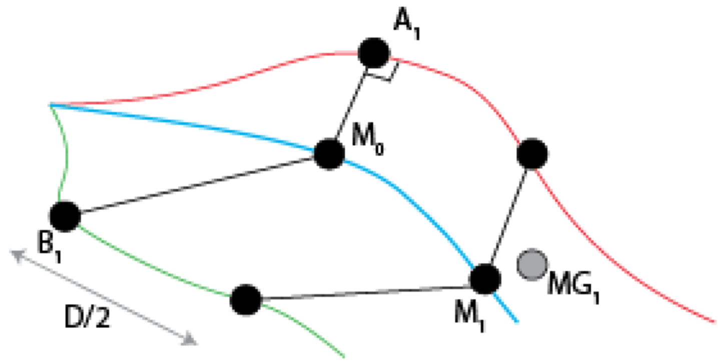

Figure 4: Shows the concept of parallel transport to find the next Median Point based on the current one

3.4 Update the median point based on the Guess

Now that we have the guessed median point, we know that its not precise. To compute the new median point from the guessed median point, we first find the projection of point MG_1 on curve A and B respectively.

To find the projection we shoot rays to each of the center point of the balls of the curve A and find the ray with least distance, $dis1$ to the point MG_1.

Then if it is not the first point on the curve on the curve, then we check the projection side of the point MG_1 to the line between center balls points $c[id-1]$ and $c[id]$. We check to find if its on the line. If it is on the line , then we re compute the projection of point MG_1 on the $line(c[id-1], c[id])$ and compute the new distance $dis2$.

Similarly if it is not the last point on the curve, then we check the projection side of the point MG_1 to the line between center balls points $c[id]$ and $c[id+1]$. We check to find if its on the line. If it is on the line , then we re compute the projection of point MG_1 on the $line(c[id], c[id+1])$ and compute the new distance $dis3$.

Now we compare the three distances $dis1, dis2dis3$ and get the projected point PA_1 which has the least distance.We do the same process for curve B and find the projected point PB_1.

Likewise, we also calculate the tangent vectors $tanPA_1$ and $tanPB_1$ at points PA_1 and PB_1 respectively. At this point with the help of these two points and tangent vectors, we call the bisection method as explained above to compute the two bisected points $BB1$ and $BB2$.

We create a line between $BB1$ and $BB2$ and find the projection of MG_1 on this line which is basically the updated median point as below:

$$Point\ \mathrm{M}_1 = \mathrm{project}(MG_1, BB1, BB2) \tag{11}$$

3.5 Calculate the Median Line

Computing the median lines needs computation of all the median points. We can get the First Median Point and the next Median Point as aforementioned. We iterate the computation part of guessing the next median and updating the same point with precision by feeding last computed median point as the first median point at each iteration. We keep a check such that the distance between the last median point and the end center point of the curve should be greater than $D/2$.

$$\begin{aligned} while(n(V(last.M, A.C[A.n-1])) > D/2 \quad i < 70) \\ Point \text{ p= } guessNextMedian() \\ Point \text{ pnew= } updateMedianPoint() \end{aligned} \tag{12}$$

4 Implementation details and some cautions

4.1 Class Nearest

This Class in rope.pde contains the data structure to store the closest projection in a rope to a certain point. It stores, the projection point, projection distance, the id of the projected point and whether the point is in the middle of an edge.

4.2 Class Median

This Class in rope.pde creates the data structure to save the median point P with two other Nearest Objects, storing the respective nearest points on each Curve A and B.

All the medians are stored in an arraylist *mps*

4.3 Class Rope

The given class rope is used to make a new MPS object which is the median curve. we implement the computation of curve points in the function *calRope*() under rope.pde.Likewise we implemented $calTangent()$ to compute the tangent vectors at any point on the curve.

The projection of any point on the curve is implemented in $calProject()$ function which returns a Nearest object. It takes the id of the point on the curve and the external point to be projected as input.

4.4 Median Point Implementation

The function $calfirstMedian()$ computes the first median point by taking three points as input.

The function $guessNextMedian()$ returns the next best guessed median point on the median curve by taking the last computed median point as an input

The function $updateMedianPoint()$ returns the precise next median point on the median curve by taking the guessed median point as input along with the id's of the last projected points on the left and right curve.

4.5 Median Curve Implementation

The function $calMedian()$ iterates to compute the first median point, guessed median point and updated median point repeatedly, till the length of the median curve cannot fit any new median point to make the curve.

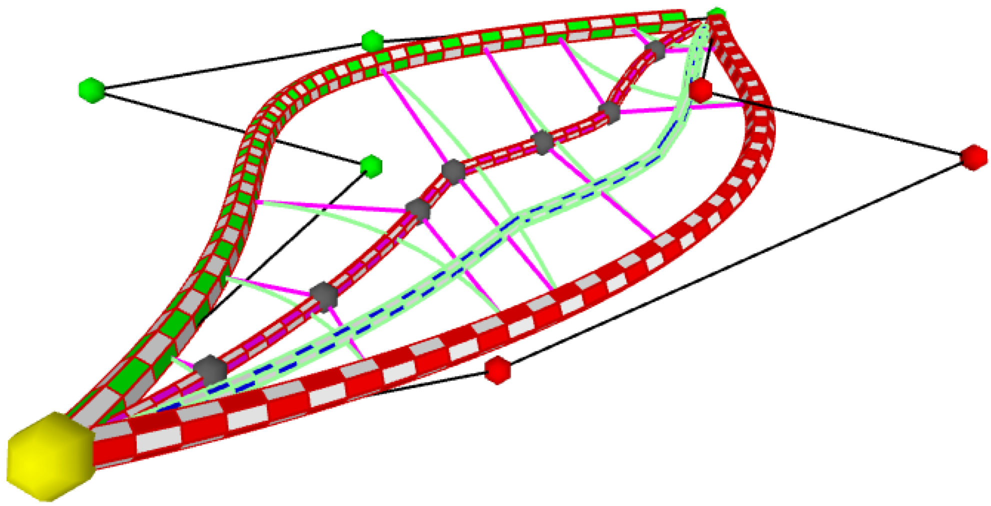

Figure 5: Screenshot of the animating morph via the median curve from Curve A to B

5 Man-made structure

6 Conclusion, discussion and reliability