Fitting and fairing Hermite-type data by matrix weighted NURBS curves

Xunnian Yang
Zhejiang University
Outline

I. Motivation and definition of matrix weighted NURBS curves
II. Matrix wt. NURBS curves with almost circular /helical precision
III. Fitting Hermite-type data by matrix wt. NURBS curves
IV. Fairing Hermite-type data by matrix wt. NURBS curves
V. Summary
I. Motivation and definition of matrix weighted NURBS
Motivation: Geometric setting of weights

• **NURBS curves** with real weights

\[
P(t) = \frac{\sum_{i=0}^{n} \omega_i P_i N_{i,k} (t)}{\sum_{i=0}^{n} \omega_i N_{i,k} (t)}
\]

- **real weights:** \( \omega_i \in \mathbb{R}^+ \)
- **control points:** \( P_i \in \mathbb{R}^d \)
- **basis functions:** \( N_{i,k} (t) \)

- Effect of individual weight is clear
- Difficult to edit multi-weights

• **Q1:** Can the weights be chosen in a more clear geometric way?
Motivation: Fairness control

• A NURBS curve is fair when
  ➢ its curvature (normal) varies smoothly, or
  ➢ curvature, torsion, etc. vary smoothly

• The formulae are complex for NURBS curves

\[
k(t) = \frac{\|P'(t) \times P''(t)\|}{\|P'(t)\|^3} \quad \tau(t) = \frac{\det(P'(t), P''(t), P'''(t))}{\|P'(t) \times P''(t)\|^2}
\]

• Naive optimization of the curvature/torsion is impractical

• Q2: Does there exist simple way to construct fair
  NURBS curves?
Motivation: Fitting precision to control points

- Higher order NURBS curves are more fair but shrinks much
- Only linear precision from the control polygon
- The curves tend to be straight by recursive smoothing

\[ P_{l+1}() \]

- Q3: Can a NURBS curve lie close to control points or have non-linear precision?
Real weights vs. matrix weights
Matrix weighted NURBS curves

$$Q(t) = \left[ \sum_{i=0}^{n} M_i N_{i,k}(t) \right]^{-1} \sum_{i=0}^{n} M_i P_i N_{i,k}(t) \quad M_i \in R^{d \times d}$$

• By this model
  - the weights can be defined in geometric way
  - the obtained curves can have nonlinear precision
  - data fitting and fairing will become simple and efficient
Observation: intrinsic geometry representation via Hermite data

\[ \bar{k}_i \mathbf{n}_i = \frac{\mu_i}{2} \left( k_{i-1}^- \mathbf{n}_{i-1} + k_{i+1}^- \mathbf{n}_{i+1} \right) \]

\[ \bar{k}_i \mathbf{n}_i = \frac{2}{l^2} \left( P_i - \frac{1}{2} P_{i-1} - \frac{1}{2} P_{i+1} \right) \]

\[ k_{i-1}^+ = \frac{2(P_{i-1} - P_i) \cdot \mathbf{n}_{i-1}}{\|P_{i-1} - P_i\|^2} \quad k_{i+1}^- = \frac{2(P_{i+1} - P_i) \cdot \mathbf{n}_{i+1}}{\|P_{i+1} - P_i\|^2} \]

\[ \frac{1}{2} (M_{i-1} + M_{i+1}) P_i = \frac{1}{2} M_{i-1} P_{i-1} + \frac{1}{2} M_{i+1} P_{i+1} \]

where \( M_i = I + \mu_i \mathbf{n}_i \mathbf{n}_i^T \)

\[ P_i = \left( \frac{1}{2} M_{i-1} + \frac{1}{2} M_{i+1} \right)^{-1} \left( \frac{1}{2} M_{i-1} P_{i-1} + \frac{1}{2} M_{i+1} P_{i+1} \right) \]

Points on a curve equal to weighted combinations of neighboring points with matrix weights!
Matrix weight and least squares fitting

\[ P_i = \left( \frac{1}{2} M_{i-1} + \frac{1}{2} M_{i+1} \right)^{-1} \left( \frac{1}{2} M_{i-1} P_{i-1} + \frac{1}{2} M_{i+1} P_{i+1} \right) \]

\[ F(P) = \frac{1}{2} (P - P_{i-1})^2 + \frac{1}{2} (P - P_{i+1})^2 + \mu_i \left\{ \frac{1}{2} [(P - P_{i-1}) \cdot n_{i-1}]^2 + \frac{1}{2} [(P - P_{i+1}) \cdot n_{i+1}]^2 \right\} \]

= \min

The selected point is the solution to least squares fitting to neighboring points and the tangent lines passing through the points.
Definition: Matrix weighted NURBS curves with point-normal control pairs

\[ F(Q(t)) = \sum_{i=0}^{n} \omega_i N_{i,k}(t)(Q(t) - P_i)^2 + \sum_{i=0}^{n} \omega_i \mu_i N_{i,k}(t)[(Q(t) - P_i) \cdot n_i]^2 \]

\[
\frac{\partial F(Q(t))}{\partial Q(t)} = 0
\]

\[ Q(t) = \left[ \sum_{i=0}^{n} M_i N_{i,k}(t) \right]^{-1} \sum_{i=0}^{n} M_i P_i N_{i,k}(t) \]

where \( M_i = \omega_i \left( I + \mu_i n_i n_i^T \right) \)
Definition: Matrix weighted NURBS curves with point-tangent control pairs

\[ F(Q(t)) = \sum_{i=0}^{n} \omega_i N_{i,k}(t)(Q(t) - P_i)^2 + \sum_{i=0}^{n} \omega_i \mu_i N_{i,k}(t)[A_i(Q(t) - P_i)]^2 \]

where \( A_i = I - t_i t_i^T \)

\[ \frac{\partial F(Q(t))}{\partial Q(t)} = 0 \]

\[ Q(t) = \left[ \sum_{i=0}^{n} M_{i} N_{i,k}(t) \right]^{-1} \sum_{i=0}^{n} M_{i} P_i N_{i,k}(t) \]

where \( M_i = \omega_i (I + \mu_i A_i) \)
Evaluation: Homogeneous coordinates representation

- NURBS curves with matrix weights

\[ Q(t) = \left[ \sum_{i=0}^{n} M_i N_{i,k}(t) \right]^{-1} \sum_{i=0}^{n} M_i P_i N_{i,k}(t) \quad M_i \in R^{d\times d} \]

- Homogeneous coordinates representation

\[ \overline{Q}(t) = \sum_{i=0}^{n} \overline{Q}_i N_{i,k}(t) \quad \overline{Q}_i = (M_i, M_i P_i) \in R^{(d+1)\times d} \]
**Matrix weighted NURBS vs NURBS**

- Any NURBS curve is a special matrix weighted NURBS curve;
  \[
P(t) = \frac{\sum_{i=0}^{n} \omega_i P_i N_{i,k}(t)}{\sum_{i=0}^{n} \omega_i N_{i,k}(t)} = \left[\sum_{i=0}^{n} M_i N_{i,k}(t)\right]^{-1} \sum_{i=0}^{n} M_i P_i N_{i,k}(t)
\]

- A matrix weighted NURBS curve can be converted to a NURBS curve (with a higher degree);
  \[
  Q(t) = \left[\sum_{i=0}^{n} M_i N_{i,k}(t)\right]^{-1} \sum_{i=0}^{n} M_i P_i N_{i,k}(t) = \frac{\sum_{i=0}^{n_1} \overline{\omega}_i P_{i,k_1}(t)}{\sum_{i=0}^{n_1} \overline{\omega}_i N_{i,k_1}(t)}
  \]
NURBS vs. matrix weighted NURBS

- NURBS curves of degree $n$
- Matrix weighted NURBS curves of degree $n$
- NURBS curves of degree $nd$, $d$ is the dimension
II. Matrix weighted NURBS curves with almost circular/helical precision
Effects of control normals

\[
Q(t) = \left[ \sum_{i=0}^{n} M_i N_{i,k}(t) \right]^{-1} \sum_{i=0}^{n} M_i P_i N_{i,k}(t)
= M_i = \omega_i \left( I + \mu_i n_i n_i^T \right)
\]
Matrix wt. NURBS with almost circular precision

• Proposition (Yang X. CAGD 42:40-53,2016):

A matrix wt. NURBS curve of order $k$ passes through its control points when

- The control points and control normals are uniformly sampled from a circular arc;
- The weight matrix $M_i = I + \mu_i n_i n_i^T$, where

$$\mu_i = \frac{\sum_{|j| = 1}^l \lambda_i + j n_i^T (P_i - P_{i+j})}{\sum_{|j| = 1}^l \lambda_i + j n_i^T n_{i+j} n_{i+j}^T (P_{i+j} - P_i)}$$
Matrix wt. NURBS with almost circular precision

Degree=7
Matrix wt. NURBS with almost circular precision

How to improve the fairness of the curve?

Degree=3
Matrix wt. NURBS with almost circular precision

• Modified weight matrix:

\[ M_i = I + \mu_i n_i n_i^T \]

\[
\mu_i = \frac{\sum_{|l|=1}^l \lambda_{i+j}d_{ij}}{\sum_{|l|=1}^l \lambda_{i+j} n_i^T n_{i+j}d_{ij}} \quad d_{ij} = \frac{1}{2} [n_i^T (P_i - P_{i+j}) + n_{i+j}^T (P_{i+j} - P_i)]
\]

• Matrix wt. NURBS curves

- less sensitive to data noise
- (still) have almost circular precision
- more fair shapes even with non-uniformly sampled control data
Matrix wt. NURBS with almost circular precision

Degree=3
Matrix wt. NURBS curves with helical precision

• Matrix wt. NURBS curves with point-tangent control pairs

\[ Q(t) = \left[ \sum_{i=0}^{n} M_i N_{i,k}(t) \right]^{-1} \sum_{i=0}^{n} M_i P_i N_{i,k}(t) \]

\[ M_i = \omega_i \left( I + \mu_i A_i \right) \]

\[ A_i = I - t_i t_i^T \]

• Almost helical precision in 3D space

\[ \omega_i = 1 \quad \mu_q = \frac{\sum_{|j|=1}^{l} \lambda_{q+j} d_{q,q+j} \eta_{q,q+j}}{\sum_{|j|=1}^{l} \lambda_{q+j} d_{q,q+j} n_q^T V_{q+j}} \]
Matrix wt. NURBS curves with helical precision

degree 1

degree 3

degree 5
Control normals vs. control tangents

Quintic B-spline curve

Point-normal based

Point-tangent based
Circular precision vs. helical precision

• Spatial matrix wt. NURBS curves
  ➢ with point-normal control pairs, or
  ➢ with point-tangent control pairs

<table>
<thead>
<tr>
<th># points</th>
<th>B-spline curve</th>
<th>point-normal</th>
<th>point-tangent</th>
</tr>
</thead>
<tbody>
<tr>
<td>20</td>
<td>0.370825</td>
<td>0.191567</td>
<td>0.109114</td>
</tr>
<tr>
<td>40</td>
<td>0.102556</td>
<td>0.019438</td>
<td>0.010415</td>
</tr>
<tr>
<td>60</td>
<td>0.046716</td>
<td>0.003962</td>
<td>0.002583</td>
</tr>
<tr>
<td>80</td>
<td>0.026509</td>
<td>0.001281</td>
<td>0.000937</td>
</tr>
<tr>
<td>100</td>
<td>0.017028</td>
<td>0.000519</td>
<td>0.000414</td>
</tr>
<tr>
<td>120</td>
<td>0.011847</td>
<td>0.000248</td>
<td>0.000205</td>
</tr>
</tbody>
</table>
III. Fitting Hermite-type data by matrix weighted NURBS curves
NURBS curve fitting: resolve control points

- Popular, many methods;
- Not shape preserving;
- Not easy for high quality (fairness).

\[ P(t) = \frac{\sum_{i=0}^{n} \omega_i P_i N_{i,k}(t)}{\sum_{i=0}^{n} \omega_i N_{i,k}(t)} \]
Direct fitting by matrix wt. NURBS curves

• Basics of 2D curves
  – A planar matrix weighted NURBS curve has (almost) circular precision;
  – A smooth planar curve can be approximated by arc splines with arbitrary tolerances (Meek & Walton 1995);

• Curve reconstruction
  – with properly sampled Hermite data (points+normals);
  – direct construction of matrix wt. NURBS curve with point-normal control pairs;
Fitting arc spline by matrix wt. NURBS

degree 1
Fitting arc spline by matrix wt. NURBS

degree 3
Fitting arc spline by matrix wt. NURBS

degree 6
Direct fitting by matrix w.t. NURBS curves

Quintic B-spline curve

Matrix w.t. NURBS curve
Direct fitting by matrix weighted NURBS curves

• Basics of 3D curves
  – A spatial matrix weighted NURBS curve has (almost) helical precision;
  – A smooth space curve can be approximated by helical arcs splines (Sloss 1970);

• Curve reconstruction
  – with properly sampled Hermite data (points+tangents);
  – direct construction of matrix wt. NURBS curve with point-tangent control pairs;
Interpolation of end/selected points

• An open matrix wt. NURBS curve can interpolate its end control point by
  ➢ using multiple end control points
    • The curve has degenerate derivatives at the end
  ➢ using multiple knots at the end
    • Additional boundary derivatives should be assigned
  ➢ adding local symmetric control points (proposed)
    • It can yield uniform parameterization
    • No additional derivatives should be assigned
Addling local symmetric control points

\[ P_{n+j} = P_{n-j} - 2a_j t_n \]
\[ a_j = (P_{n-j} - P_n) \cdot t_n \]

\[ b_j = (P_{n-j} - P_n) \cdot b_n \]
Cubic B-spline

matrix wt. NURBS of degree 3

matrix wt. NURBS of degree 7
IV. Fairing Hermite-type data by matrix weighted NURBS curves
Why fairing?

• A matrix wt. NURBS curve may not be fair
  – when the data are noisy, or
  – when the sampled points are irregularly spaced

• A fair curve is useful for
  – construction of high quality shapes;
  – tool path generation for CNC machining;
  – ......
How to fair?

• Fairing by re-sampling and refitting
  – Resample points from previously obtained curves
  – Refit new matrix wt. NURBS curves to the sampled data
The fairing algorithm

- **Preprocessing**
  - input points+(estimated)tangents
  - Construct a matrix wt. NURBS curve \( P(t) \)

- **Fairing by repeat the following steps**
  - Sample points \( P(t_i) \), where \( t_i \) are knots
  - Add local symmetric points at boundaries
  - Compute tangent vectors at the points
  - Construct new matrix wt. NURBS curve \( Q(t) \)
  - Replace \( P(t) \) by \( Q(t) \) and repeat again
Initial fitting curve

10 times of fitting
Matrix w.t. NURBS curve

(matrix w.t. NURBS curve
(10 resampling + refitting))
Adding constraints

• Points to be interpolated
  – Boundary points
  – Feature points
  – These points can be fixed during resampling

• The rest points
  – Can be unconstrained, or
  – Constrained in permitted tolerances
quintic B-spline curve;

matrix w.t. NURBS curves before/after fairing
Curve fitting and fairing

Cubic B-spline curve
Matrix w.r.t. NURBS curve
2 refitting by Matrix NURBS
50 refitting by Matrix NURBS
Fitting and fairing by matrix wt. NURBS curves
Summary

• Matrix weighted NURBS curves
  ➢ Compatible with NURBS curves
  ➢ Geometric definition of matrix weights
  ➢ Shape editing using control normals/tangents

• Almost circular/helical precision
  ➢ Direct curve fitting from data
  ➢ Fairing by resampling + refitting
Thank you!