Smooth Shape-Aware Functions with Controlled Extrema

Alec Jacobson\textsuperscript{1}
Tino Weinkauf\textsuperscript{2}
Olga Sorkine\textsuperscript{1}

\textsuperscript{1}ETH Zurich
\textsuperscript{2}MPI Saarbrücken
Real-time deformation relies on smooth, shape-aware functions

input shape + handles
Real-time deformation relies on smooth, shape-aware functions
Real-time deformation relies on smooth, shape-aware functions.
Real-time deformation relies on smooth, shape-aware functions

\[ x'_i = \sum_{j=1}^{H} f_j(x_i) T_j x_i \]
Real-time deformation relies on smooth, shape-aware functions

\[ x'_i = \sum_{j=1}^{H} f_j(x_i) T_j x_i \]
Spurious extrema cause distracting artifacts

unconstrained $\Delta^2$
[Botsch & Kobbelt 2004]

$$x'_i = \sum_{j=1}^{H} f_j(x_i) T_j x_i$$
Spurious extrema cause distracting artifacts

unconstrained \( \Delta^2 \)
[Botsch & Kobbelt 2004]

\[
x'_i = \sum_{j=1}^{H} f_j(x_i) T_j x_i
\]
Bounds help, but don’t solve problem

bounded $\Delta^2$

[Jacobson et al. 2011]

$$x'_i = \sum_{j=1}^{H} f_j(x_i) T_j x_i$$
Bounds help, but don’t solve problem

bounded $\Delta^2$

[Jacobson et al. 2011]

$$x'_i = \sum_{j=1}^{H} f_j(x_i) T_j x_i$$
Gets worse with higher-order smoothness

bounded $\Delta^4$

[Jacobson et al. 2011]

$$x'_i = \sum_{j=1}^{H} f_j(x_i) T_j x_i$$

$\Delta^k$, $k > 2$ oscillate too much
Gets worse with higher-order smoothness

bounded $\Delta^4$  
[Jacobson et al. 2011]

\[
x'_i = \sum_{j=1}^{H} f_j(x_i) T_j x_i
\]

$\Delta^k, k > 2$ oscillate too much

\[
\text{local max}
\]
\[
\text{local min}
\]
We explicitly prohibit spurious extrema

$$x'_i = \sum_{j=1}^{H} f_j(x_i) T_j x_i$$
We explicitly prohibit spurious extrema

\[ x'_i = \sum_{j=1}^{H} f_j(x_i) T_j x_i \]

our $\Delta^4$
Same functions used for color interpolation

\[
x'_i = \sum_{j=1}^{H} f_j(x_i) T_j x_i
\]
Same functions used for color interpolation

\[ \mathbf{c}_i = \sum_{j=1}^{H} f_j(x_i) \mathbf{c}_j \]
Same functions used for color interpolation

unconstrained $\Delta^2$
[Finch et al. 2011]

\[ c_i = \sum_{j=1}^{H} f_j(x_i) c_j \]
 Same functions used for color interpolation

unconstrained $\Delta^2$
[Finch et al. 2011]

$$c_i = \sum_{j=1}^{H} f_j(x_i)c_j$$
Same functions used for color interpolation

unconstrained $\Delta^2$
[Finch et al. 2011]

$$c_i = \sum_{j=1}^{H} f_j(x_i) c_j$$

Our $\Delta^2$
Want same control when smoothing data
Want same control when smoothing data

Exact, but sharp geodesic
Want same control when smoothing data

Exact, but sharp geodesic
Want same control when smoothing data

Exact, but sharp geodesic

Smooth, but extrema are lost
Want same control when smoothing data

Exact, but sharp geodesic

Smooth and maintain extrema
Ideal discrete problem is intractable

\[
\arg \min_f E(f)
\]

Interpolation functions:

\[
E_L(f) = \int_{\mathcal{M}} \| \nabla^k f \|^2 dV, \quad k = 2, 3, \ldots
\]
Ideal discrete problem is intractable

arg \min_{f} \ E(f)

Data smoothing:

\[ E_L(f) = \int_{\mathcal{M}} \| \nabla^k f \|^2 dV, \quad k = 2, 3, \ldots \]

\[ E_D(f) = \sum_{i \in \mathcal{M}} \| h_i - f_i \|^2 \]

\[ E(f) = \gamma_L E_L(f) + \gamma_D E_D(f) \]
Ideal discrete problem is intractable

$$\arg\min_f E(f)$$
Ideal discrete problem is intractable

\[
\arg\min_{f} E(f) \\
\text{s.t. } f_{\max} = \text{known} \\
\quad f_{\min} = \text{known}
\]
Ideal discrete problem is intractable

$$\arg \min_{f} \, E(f)$$

s.t. \( f_{\text{max}} = \text{known} \)

\( f_{\text{min}} = \text{known} \)

linear

\( f_{j} < f_{\text{max}} \)

\( f_{j} > f_{\text{min}} \)
Ideal discrete problem is intractable

\[
\arg \min_{f} E(f)
\]

s.t. \( f_{\text{max}} = \text{known} \)
\( f_{\text{min}} = \text{known} \)

linear

\( f_j < f_{\text{max}} \)
\( f_j > f_{\text{min}} \)

nonlinear

\( f_i > \min_{j \in \mathcal{N}(i)} f_j \)
\( f_i < \max_{j \in \mathcal{N}(i)} f_j \)
Assume we have a feasible solution

\[
\arg \min_f E(f)
\]

s.t. \( f_{\text{max}} = \text{known} \)

\( f_{\text{min}} = \text{known} \)

**linear**

\( f_j < f_{\text{max}} \)

\( f_j > f_{\text{min}} \)

**nonlinear**

\( f_i > \min_{j \in \mathcal{N}(i)} f_j \)

\( f_i < \max_{j \in \mathcal{N}(i)} f_j \)

"Representative function" \( u \)

\( u_j < u_{\text{max}} \)

\( u_j > u_{\text{min}} \) handles

\( u_i > \min_{j \in \mathcal{N}(i)} u_j \) interior

\( u_i < \max_{j \in \mathcal{N}(i)} u_j \)
Assume we have a feasible solution

“Representative function” $U$

- **handles**
  - $u_j < u_{\text{max}}$
  - $u_j > u_{\text{min}}$

- **interior**
  - $u_i > \min_{j \in \mathcal{N}(i)} u_j$
  - $u_i < \max_{j \in \mathcal{N}(i)} u_j$
Copy “monotonicity” of representative

$$\arg\min_f E(f)$$

s.t. \( f_{\text{max}} = \text{known} \)

\( f_{\text{min}} = \text{known} \)

\[(f_i - f_j)(u_i - u_j) > 0 \quad \text{linear} \quad \forall (i, j) \in \mathcal{E}\]

At least one edge in either direction per vertex
Rewrite as conic optimization

**QP**

\[
\begin{align*}
\text{minimize} & \quad \frac{1}{2} \| F f \|^2 + c^T f + \text{const} \\
\text{subject to} & \quad A_{\text{ineq}}^T f \leq b_{\text{ineq}}, \\
& \quad f \leq u_f, \quad f \geq l_f
\end{align*}
\]

**Conic**

\[
\begin{align*}
\text{minimize} & \quad [c^T \ 0 \ 1] [f \ t \ v]^T + \text{const} \\
\text{subject to} & \quad [F \ -I \ 0] [f \ t \ v]^T \preceq [0 \ -\infty] \\
& \quad [F \ -I \ 0] [f \ t \ v]^T \succeq [0 \ b_{\text{ineq}}] \\
& \quad [f \ t \ v]^T \preceq [u_f \ \infty \ \infty] \\
& \quad [f \ t \ v]^T \succeq [l_f \ -\infty \ 0] \\
2v & \geq \sum_i c_i^2
\end{align*}
\]

Optimize with MOSEK

August 9, 2012

Alec Jacobson

# 34
We always have harmonic representative

$$\arg \min_u \frac{1}{2} \int_\Omega \| \nabla u \|^2 dV$$
We always have harmonic representative

\[ \arg \min_u \frac{1}{2} \int_\Omega \| \nabla u \|^2 dV \]

s.t. \( u_{\text{max}} = 1 \)
We always have harmonic representative

$$\arg\min_u \frac{1}{2} \int_{\Omega} \| \nabla u \|^2 dV$$

s.t. $u_{\text{max}} = 1$

s.t. $u_{\text{min}} = 0$
We always have harmonic representative

$$\arg\min_{u} \frac{1}{2} \int_{\Omega} \| \nabla u \|^2 dV$$

s.t. $u_{\text{max}} = 1$

s.t. $u_{\text{min}} = 0$

Works well when no input function exists
Data energy may fight harmonic representative

Anisotropic input data
Data energy may fight harmonic representative

Anisotropic input data

Harmonic representative
Data energy may fight harmonic representative

Anisotropic input data

Harmonic representative
Data energy may fight harmonic representative

Anisotropic input data

Harmonic representative
Data energy may fight harmonic representative

Anisotropic input data

Resulting solution with large $\gamma D$
If data exists, copy topology, too

Anisotropic input data

[Weinkauf et al. 2010] representative
If data exists, copy topology, too

Anisotropic input data

Resulting solution with large $\gamma_D$
Final algorithm is simple and efficient

- *Data smoothing:* topology-aware representative
  - Morse-smale + linear solve \(\sim\)milliseconds
Final algorithm is simple and efficient

- **Data smoothing**: topology-aware representative
  - Morse-smale + linear solve $\sim$milliseconds

- **Interpolation**: harmonic representative
  - Linear solve $\sim$milliseconds
Final algorithm is simple and efficient

- **Data smoothing**: topology-aware representative
  - Morse-smale + linear solve ~milliseconds

- **Interpolation**: harmonic representative
  - Linear solve ~milliseconds

- **Conic optimization**
  - 2D ~milliseconds, 3D ~seconds
Final algorithm is simple and efficient

- **Data smoothing**: topology-aware representative
  - Morse-smale + linear solve \(~\text{milliseconds}\)
- **Interpolation**: harmonic representative
  - Linear solve \(~\text{milliseconds}\)
- **Conic optimization**
  - 2D \(~\text{milliseconds}, 3D \sim\text{seconds}\)

*Interpolation*: functions are precomputed
We preserve troublesome appendages

Bounded $\Delta^2$

Our $\Delta^2$
We preserve troublesome appendages

Bounded $\Delta^2$

Our $\Delta^2$
We preserve troublesome appendages

Bounded $\Delta^2$

Our $\Delta^2$
Our weights attach appendages to body

Extrema glue appendages to far-away handles

[Botsch & Kobbelt 2004, Jacobson et al. 2011]
Extrema glue appendages to far-away handles

[Botsch & Kobbelt 2004, Jacobson et al. 2011]
Our weights attach appendages to body

Our method
Our weights attach appendages to body

Our method
Extrema distort small features

Unconstrained $\Delta^2$ [Botsch & Kobbelt 2004]

weight of middle point
Extrema distort small features

Unconstrained $\Delta^2$ [Botsch & Kobbelt 2004]

weight of middle point
Extrema distort small features

Bounded $\Delta^2$ [Jacobson et al. 2011]

weight of middle point
“Monotonicity” helps preserve small features

Bounded $\Delta^2$ [Jacobson et al. 2011]

Our $\Delta^2$
Spurious extrema are unstable, may “flip”

slightly larger region
Spurious extrema are unstable, may “flip”

slightly larger region
Spurious extrema are unstable, may “flip”

Unconstrained $\Delta^3$ [Botsch & Kobbelt, 2004]
Spurious extrema are unstable, may “flip”

Unconstrained $\Delta^3$ [Botsch & Kobbelt, 2004]
Spurious extrema are unstable, may “flip”

Unconstrained $\Delta^3$ [Botsch & Kobbelt, 2004]
Spurious extrema are unstable, may “flip”

Bounded $\Delta^3$
Spurious extrema are unstable, may “flip”

Bounded $\Delta^3$
Lack of extrema leads to more stability

Our $\Delta^3$
Lack of extrema leads to more stability
Even control continuity at extrema

Original
Even control continuity at extrema

Original

Direct extension of [Botsch & Kobbelt 2004]
Even control continuity at extrema

Original

[Botsch & Kobbelt 2004] + data term
Even control continuity at extrema

Original

Our method without data term
Even control continuity at extrema

Original

Our method with data term
Reproduces results of Weinkauf et al. 2010...

Original noisy data
Reproduces results of Weinkauf et al. 2010...

Original noisy data
Reproduces results of Weinkauf et al. 2010...

Original noisy data

Simplified and smoothed
Reproduces results of Weinkauf et al. 2010...

Original noisy data

Simplified and smoothed
... but 1000 times faster

30K vertices
5 seconds per solve
... but 1000 times faster

30K vertices
5 seconds per solve
... but 1000 times faster

30K vertices
5 seconds per solve
Conclusion: Important to control extrema

- Copy “monotonicity” of harmonic functions
- *Reduces* search-space, but optimization is tractable
Future work and discussion

- Larger, but still tractable subspace?
  - Consider all valid harmonic functions?
Future work and discussion

- Larger, but still tractable subspace?
  - Consider all valid harmonic functions?
- Continuous formulation?
We thank Kenshi Takayama for his valuable feedback. This work was supported in part by an SNF award 200021_137879 and by a gift from Adobe Systems.
Smooth Shape-Aware Functions with Controlled Extrema

MATLAB Demo: http://igl.ethz.ch/projects/monotonic/

Alec Jacobson (jacobson@inf.ethz.ch)
Tino Weinkauf
Olga Sorkine