Earth Movers Distances on Discrete Surfaces

Justin Solomon, Raif Rustamov, Leonidas Guibas  Stanford University
Adrian Butscher  Autodesk Research
Distances in Geometry Processing

Point-to-point “Geodesic”

Feature-to-point

Feature-to-feature

Torus by M. Irons, signed distance by R. Kolluri, curve distance by C. Wu
“Somewhere over here.”
“Exactly here.”
Probabilistic Geometry

\[ \rho(x) \]

Superposition

“One of these two places.”
Fuzzy Distances

Which is closer, 1 or 2?

Query

$p(x, y)$

$p_1(x, y)$

$p_2(x, y)$
Typical Measurement

$p_1(x)$

$p_1(x) - p_2(x)$

$p_2(x)$

$L^p$ norm
KL divergence
Fuzzy Distances

Which is closer, 1 or 2? Equidistant.
Overlap is the wrong measure!
Alternative: Earth Mover’s Distance

Cost to move mass $m$ from $x$ to $y$:

$$m \cdot d(x, y)$$
Alternative: Earth Mover’s Distance

\[
\min_T \sum_{i,j} T_{ij} d(x_i, x_j) \\
\text{s.t. } \sum_j T_{ij} = p_i \\
\sum_i T_{ij} = q_j \\
T \geq 0
\]

\[m \cdot d(x, y)\]

Starts at \(p\)

Ends at \(q\)

Positive

Move mass from one distribution to the other
Earth Mover’s Distance

- Many names
  - Wasserstein distance, transportation distance, Mallows distance

- Theoretically sound
  - Regularity properties, continuous and discrete formulations

- Popular option
  - Computer vision, machine learning, operations, graphics
Computer Graphics Applications

\[
\begin{align*}
\min_T & \sum_{i,j} T_{ij} d(x_i, \cdot) \\
\text{s.t.} & \sum_j T_{ij} = p_i \\
& \sum_i T_{ij} = q_j \\
& T_{ij} \geq 0
\end{align*}
\]

Matrix $T_{ij}$ is too big!

Precompute $d(x_i, x_j)$ for all $i, j$!
Our approach: Use Eulerian Flow

Probabilities *advect* along the surface

*New discretization, optimization, and (consequently) applications!*

Think of probabilities like a fluid
**Alternative Formulation**

**Total work**

\[
\inf_{J} \int_{M} \| J(x) \| \, dx
\]

\[
\text{s.t. } \nabla \cdot J(x) = \rho_1(x) - \rho_0(x)
\]

\[
J(x) \cdot n(x) = 0 \quad \forall x \in \partial M
\]

**Advects from \( \rho_0 \) to \( \rho_1 \)**

**Scales linearly**

**Theoretical version:**

“Beckmann problem”
Hodge Decomposition of $J$

$$J(x) = \nabla f(x) + R \cdot \nabla g(x)$$

Curl-free

Div-free

$$\nabla \cdot J = \Delta f = \rho_1 - \rho_0$$

New idea!
1. $\Delta f = \rho_1 - \rho_0$ \hspace{1cm} Sparse SPD linear solve for $f$

2. $\inf_g \int_M \| \nabla f(x) + \mathcal{R} \cdot \nabla g(x) \| \, dx$ \hspace{1cm} Unconstrained and convex optimization for $g$
Fast Optimization

1. \( \Delta f = \rho_1 - \rho_0 \) - Sparse SPD linear solve for \( f \)

2. \( \inf_g \int_M \| \nabla f(x) + R \cdot \nabla g(x) \| \, dx \) - Unconstrained and convex optimization for \( g \)

- Piecewise-linear FEM, optimized via ADMM
- Spectral approximation (optional)

\[ g(x) = a_1 \phi_1(x) + a_2 \phi_2(x) + a_3 \phi_3(x) + \cdots \]
\[ \Delta \phi_k = \lambda_k \phi_k \]

Satisfies triangle inequality!
Fast Optimization

**function ADMM-WASSERSTEIN**(\(\rho_0, \rho_1\))

- \(\rho_0, \rho_1\) have one value per vertex
- Concatenate \(B_t\)'s vertically to obtain \(B\)

\[
\begin{align*}
f & \leftarrow \Delta^+(\rho_1 - \rho_0) & \text{\(\triangleright\) Solve for gradient part} \\
v & \leftarrow \nabla f & \text{\(\triangleright\) Compute gradient vector field}
\end{align*}
\]

for \(i \leftarrow 1, 2, 3, \ldots\)  
\[
\begin{align*}
z_t & \leftarrow B_t c + w_t - \frac{w_t}{\beta} & \text{\(\triangleright\) Iterate until convergence} \\
\alpha_t & \left\{ \begin{array}{ll}
1 - \frac{1}{\beta\|z_t\|} & \beta\|z_t\| > 1 \\
0 & \text{otherwise}
\end{array} \right. \\
J_t & \leftarrow \alpha_t z_t & \text{\(\triangleright\) Update vector field \(J\)}
\end{align*}
\]

- Update coefficients; can pre-factor
\[
c \leftarrow \left(\sum_t B_t^\top B_t\right)^{-1} \left[\sum_t B_t^\top \left(\frac{w_t}{\beta} + J_t - w_t\right)\right]
\]

\[
y_t \leftarrow y_t + \beta (J_t - B_t c - w_t) & \text{\(\triangleright\) Update dual}
\]

**return** \(J_t\) \(\forall t \in T\)

Iterations are **fast and easy to implement**!
Pointwise Distance

\( W(\rho_0, \rho_1) \rightarrow d(x, y) \)
Proposition: Satisfies triangle inequality.
Pointwise Distance

Proposition: Satisfies triangle inequality.
Volumetric Distance

\begin{align*}
    p &= \int_M x \rho_p(x) \, dx \\
    \implies d_W(p, q) &\equiv \mathcal{W}(\rho_p, \rho_q)
\end{align*}

Use barycentric coordinates (mean value)
Volumetric Distance

Works for negative weights

Reduces to geodesic distance
EMD in Optimization

\[
\min_{\rho_1} \mathcal{W}(\rho_0, \rho_1) \\
\text{s.t. } \rho_1 \text{ outside maze}
\]
min\rho \sum_k W(\rho, \rho_k)^2
Variations of EMD

Avoid center

Distance to feature
What’s Next?

- **Quadratic** ground distance

- **Other representations**
  - Point clouds? Polygon soup? Graphs?

- **Faster optimization**
Earth Movers Distances on Discrete Surfaces

Thanks!
# Timings

<table>
<thead>
<tr>
<th>Mesh</th>
<th>$n_{\text{vert}}$</th>
<th>$d_g$</th>
<th>$d_h$</th>
<th>$d_b$</th>
<th>$d^0_{\text{VV}}$</th>
<th>$d^{20}_{\text{VV}}$</th>
<th>$d^{100}_{\text{VV}}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bearing</td>
<td>3182</td>
<td>0.050</td>
<td>0.002</td>
<td>3.52</td>
<td>3.86</td>
<td>30.8</td>
<td>41.4</td>
</tr>
<tr>
<td>David</td>
<td>5197</td>
<td>0.096</td>
<td>0.003</td>
<td>10.09</td>
<td>6.18</td>
<td>86.5</td>
<td>121.2</td>
</tr>
<tr>
<td>Dog</td>
<td>3716</td>
<td>0.056</td>
<td>0.002</td>
<td>4.66</td>
<td>3.27</td>
<td>38.7</td>
<td>59.8</td>
</tr>
<tr>
<td>Teapot</td>
<td>3900</td>
<td>0.063</td>
<td>0.002</td>
<td>6.25</td>
<td>3.87</td>
<td>45.2</td>
<td>57.9</td>
</tr>
<tr>
<td>Man</td>
<td>10050</td>
<td>0.18</td>
<td>0.006</td>
<td>42.2</td>
<td>23.2</td>
<td>312.0</td>
<td>511.9</td>
</tr>
</tbody>
</table>

Single-source all-targets
# Timings

<table>
<thead>
<tr>
<th>Mesh size</th>
<th>( M \text{ for } d_g )</th>
<th>( M \text{ for } d_h )</th>
<th>( M \text{ for } d_b )</th>
<th>( M \text{ for } d^0_{\nabla\nabla} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( n_{\text{vert}} )</td>
<td>( n_{\text{tri}} )</td>
<td>2</td>
<td>100</td>
<td>2</td>
</tr>
<tr>
<td>2k</td>
<td>4k</td>
<td>0.06</td>
<td>2.60</td>
<td>0.03</td>
</tr>
<tr>
<td>4k</td>
<td>9k</td>
<td>0.13</td>
<td>6.25</td>
<td>0.05</td>
</tr>
<tr>
<td>8k</td>
<td>16k</td>
<td>0.24</td>
<td>11.76</td>
<td>0.10</td>
</tr>
<tr>
<td>16k</td>
<td>32k</td>
<td>0.70</td>
<td>34.93</td>
<td>0.20</td>
</tr>
<tr>
<td>53k</td>
<td>105k</td>
<td>2.74</td>
<td>121.94</td>
<td>0.71</td>
</tr>
<tr>
<td>111k</td>
<td>222k</td>
<td>8.06</td>
<td>432.28</td>
<td>2.04</td>
</tr>
</tbody>
</table>

All-pairs for sample of \( M \) points
Robustness

Perturbation

Isometry and remeshing
Triangle Inequality

Fix $p$ and $q$; red points are where $d(p, \cdot) + d(\cdot, q) < d(p, q)$. 

- $m = 1$ (default)
- $m = 10$
- $m = 100$ 

[Crane et al. 2013] 

This paper 

$d^0_W$, $d^{10}_W$, $d^{100}_W$