Laplacian Growth

Mathias Plans
What is to come

- Laplacian growth
- Laplacian operator
- Graph Laplacian
- Computation
- Use cases
Laplacian field
\[ \Phi = 0 \]

\[ \Phi = 1 \]
$\nabla^2 \Phi = 0$
Maths
\( f(x, y) = x^2 y^3 \)
$\nabla^2 f = 0$
\[ f(x, y) = xy \]
\[ f(x, y) = \sin x + \cos y \]
\[ f(x, y) = e^x \sin y \]
Discretization
\[ \nabla^2 \phi(v) = \sum_{w : d(w, v) = 1} \phi(v) - \phi(w) \]
<table>
<thead>
<tr>
<th>Labelled graph</th>
<th>Degree matrix</th>
</tr>
</thead>
</table>
| ![Labelled graph](https://en.wikipedia.org/wiki/Laplacian_matrix) | \[
\begin{bmatrix}
2 & 0 & 0 & 0 & 0 & 0 \\
0 & 3 & 0 & 0 & 0 & 0 \\
0 & 0 & 2 & 0 & 0 & 0 \\
0 & 0 & 0 & 3 & 0 & 0 \\
0 & 0 & 0 & 0 & 3 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 \\
\end{bmatrix}
\] |

<table>
<thead>
<tr>
<th>Adjacency matrix</th>
<th>Laplacian matrix</th>
</tr>
</thead>
</table>
| \[
\begin{bmatrix}
0 & 1 & 0 & 0 & 1 & 0 \\
1 & 0 & 1 & 0 & 1 & 0 \\
0 & 1 & 0 & 1 & 0 & 0 \\
0 & 0 & 1 & 0 & 1 & 1 \\
1 & 1 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 \\
\end{bmatrix}
\] | \[
\begin{bmatrix}
2 & -1 & 0 & 0 & -1 & 0 \\
-1 & 3 & -1 & 0 & -1 & 0 \\
0 & -1 & 2 & -1 & 0 & 0 \\
0 & 0 & -1 & 3 & -1 & -1 \\
-1 & -1 & 0 & -1 & 3 & 0 \\
0 & 0 & 0 & -1 & 0 & 1 \\
\end{bmatrix}
\] |
$\nabla^2 \phi = 0$
Back to the Laplacian growth

1. Set the initial values
2. Find a harmonic function
3. Select growth locations based on the harmonic function
4. Repeat from step 2 until the growth is desirable
5. Post-process
1. Set the initial values

- Depends on the scene
- ...and the problem
2. Find a harmonic function

\[ \nabla^2 \phi = 0 \]
2. Find a harmonic function

- System of linear equations
- When $n$ is the number of vertices in the graph
  - Gaussian has time complexity of $O(n^3)$
  - Conjugate Gradient and Successive Overrelaxation have time complexity of $O(n^{1.5})$ but are more specialized
  - Multigrid method has time complexity of $O(n)$

\[ \nabla^2 \phi = 0 \]
<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Type</th>
<th>Serial Time</th>
<th>PRAM Time</th>
<th>Storage</th>
<th>#Procs</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dense LU</td>
<td>D</td>
<td>$N^3$</td>
<td>$N$</td>
<td>$N^2$</td>
<td>$N^2$</td>
</tr>
<tr>
<td>Band LU</td>
<td>D</td>
<td>$N^2$</td>
<td>$N$</td>
<td>$N^2(3/2)$</td>
<td>$N$</td>
</tr>
<tr>
<td>Inv(P)*bhat</td>
<td>D</td>
<td>$N^2$</td>
<td>log $N$</td>
<td>$N^2$</td>
<td>$N^2$</td>
</tr>
<tr>
<td>Jacobi</td>
<td>I</td>
<td>$N^2$</td>
<td>$N$</td>
<td>$N$</td>
<td>$N$</td>
</tr>
<tr>
<td>Sparse LU</td>
<td>D</td>
<td>$N^{(3/2)}$</td>
<td>$N^{(1/2)}$</td>
<td>$N^*\log N$</td>
<td>$N$</td>
</tr>
<tr>
<td>CG</td>
<td>I</td>
<td>$N^{(3/2)}$</td>
<td>$N^{(1/2)}*\log N$</td>
<td>$N$</td>
<td>$N$</td>
</tr>
<tr>
<td>SOR</td>
<td>I</td>
<td>$N^{(3/2)}$</td>
<td>$N^{(1/2)}$</td>
<td>$N$</td>
<td>$N$</td>
</tr>
<tr>
<td>FFT</td>
<td>D</td>
<td>$N^*\log N$</td>
<td>log $N$</td>
<td>$N$</td>
<td>$N$</td>
</tr>
<tr>
<td>Multigrid</td>
<td>I</td>
<td>$N$</td>
<td>$(\log N)^2$</td>
<td>$N$</td>
<td>$N$</td>
</tr>
<tr>
<td>Lower Bound</td>
<td>N</td>
<td>$\log N$</td>
<td>$\log N$</td>
<td>$N$</td>
<td>$N$</td>
</tr>
</tbody>
</table>
### Parallelized

<table>
<thead>
<tr>
<th>Method</th>
<th># flops</th>
<th># messages</th>
<th># words sent</th>
</tr>
</thead>
<tbody>
<tr>
<td>SOR</td>
<td>$N^{(3/2)}/p$</td>
<td>$N^{(1/2)}$</td>
<td>$N/p$</td>
</tr>
<tr>
<td>FFT</td>
<td>$N\times\log(N)/p$</td>
<td>$p^{(1/2)}$</td>
<td>$N/p$</td>
</tr>
<tr>
<td>Multigrid</td>
<td>$N/p + \log(p)\times\log(N)$</td>
<td>$(\log(N))^{^2}$</td>
<td>$(N/p)^{(1/2)} + \log(p)\times\log(N)$</td>
</tr>
</tbody>
</table>

p is the number of processes
Reducing $n$

- $n$ is usually very large
- 3D grid of size 1000 has 1 billion vertices
- Can reduce granularity
Reducing $n$

- $n$ is usually very large
- 3D grid of size 1000 has 1 billion vertices
- Can reduce granularity
- Or..
Fast Animation of Lightning Using An Adaptive Mesh

Theodore Kim and Ming C. Lin
University of North Carolina at Chapel Hill
Delta notates the length, not Laplacian.
Approximate it

- For CG purposes, approximation will do
Fast Simulation of Laplacian Growth

Theodore Kim, Jason Sewall, Avneesh Sud and Ming C. Lin
University of North Carolina at Chapel Hill, USA
{kim,sewall,sud,lin}@cs.unc.edu
3. Select growth locations based on the harmonic function

- Can be stochastic
- Single point of growth
- Surface growth
4. Repeat from step 2 until the growth is desirable

- Run a certain number of iterations
- Positive and negative boundaries touch
- Or meet some other criterium
5. Post-process

- Laplacian field alone is useless
- Turn it into a graph
- Volumetric rendering
Uses
(a) Original configuration

(b) Lightning configuration
\[ p_i = \frac{\left( \phi_i \right)^\eta}{\sum_{j=1}^{n} \left( \phi_j \right)^\eta} \]
(a) Original configuration

(b) $\eta = 1$

(c) $\eta = 2$

(d) $\eta = 3$
Lightning is not just one bolt, but many subsequent very similar bolts
Lightning is not just one bolt, but many subsequent very similar bolts

\[
\left( \nabla^2 + \left( \frac{\omega}{c} \right)^2 \right) \phi = -4\pi\rho
\]
Lightning is not just one bolt, but many subsequent very similar bolts

\[ \left( \nabla^2 + \left( \frac{\omega}{c} \right)^2 \right) \phi = -4\pi \rho \]

\[ \nabla^2 \phi = -4\pi \rho \]

Rho is determined by previous bolts
Post-processing

- Convert into a tree
- Determine main path
- Set the “wattage” of each edge
Post-processing

- Convert into a tree
- Determine main path
- Set the “wattage” of each edge

Rendering

- Lightning bolt is actually very narrow
- Construct an APSF kernel
  - Authors created for 2km
- Use APSF kernel for post-processing

https://gamma.cs.unc.edu/LIGHTNING/lightning.pdf
Potential uses

- Movies/animations
- Environments in games
- Gameplay
Potential uses

● Movies/animations
● Environments in games
● Gameplay
  ○ Tesla coil ability in Sprash
\[ \nabla^2 \phi = 0 \]

\[ f(x, y) = e^x \sin y \]

\[ \Phi \]

\[ \text{The End} \]

\[ p_i = \frac{(\phi_i)^\eta}{\sum_{j=1}^{n}(\phi_j)^\eta} \]

\[ \nabla^2 \phi(v) = \sum_{w:d(w,v)=1} \phi(v) - \phi(w) \]
Sources

Main sources:

- https://gamma.cs.unc.edu/LIGHTNING/
- https://gamma.cs.unc.edu/FRAC/laplacian_small.pdf
- https://gamma.cs.unc.edu/FAST_LIGHTNING/
- https://people.eecs.berkeley.edu/~demmel/cs267/

Wikipedia sources:

- https://en.wikipedia.org/wiki/Laplacian_matrix
- https://en.wikipedia.org/wiki/Laplace_operator
- https://en.wikipedia.org/wiki/Laplace%27s_equation
Used videos

- https://youtu.be/xkClgx88jR0
- https://youtu.be/x1piZoh_Nvk
- https://youtu.be/SL6SkFEZnf4
- https://youtu.be/HC0DFVRLV0U

Useful videos

- https://youtu.be/EW08rD-GFh0
- https://youtu.be/XbCvGRjizgg
- https://youtu.be/JQSC0lCPG24
Pictures

- [https://n-e-r-v-o-u-s.com/blog/?p=1536](https://n-e-r-v-o-u-s.com/blog/?p=1536)
- [https://unsplash.com/photos/0G01UI1MQhg](https://unsplash.com/photos/0G01UI1MQhg)
- [https://www.farwayart.com/shop/scenic-dendrite-agate](https://www.farwayart.com/shop/scenic-dendrite-agate)
- [https://stormandsky.com/lightning](https://stormandsky.com/lightning)
- [https://sites.und.edu/timothy.prescott/apex/web/apex.Ch15.S2.html](https://sites.und.edu/timothy.prescott/apex/web/apex.Ch15.S2.html)

Tools

- [https://www.geogebra.org/3d/](https://www.geogebra.org/3d/)
- [https://latexeditor.lagrida.com/](https://latexeditor.lagrida.com/)
- MS Paint