Differential Geometry:
Numerical Integration and Surface Flow

[*Implicit Fairing of Irregular Meshes using Diffusion and Curvature Flow*. Desbrun *et al.*, 1999]

Energy Minimization

Recall:

We have been considering the situation in which we are given an energy $E(\boldsymbol{x})$ that we would like to minimize.

- Mean Curvature Flow $E:\mathbf{R}^{3\text{x}V}\rightarrow\mathbf{R}$, taking vertex positions to areas.
- Circle Packings $E:\mathbf{R}^{V}\rightarrow\mathbf{R}$, taking radii at the vertices to vertex positions to absolute angle-sum deficits.
- Circle Patterns $E:\mathbf{R}^{T}\rightarrow\mathbf{R}$, taking log-radii at triangles to the integral of kite-angle-sum deficits.

Energy Minimization

<u>Recall</u>:

We have been considering the situation in which we are given an energy *E*(***x***) that we would like to minimize.

In each case, the negative gradient of the energy told us how to modify the values to reduce the energy at each time step:

$$\frac{dx}{dt} = -\nabla E(x)$$

and the Hessian gave us the change in the gradient direction.

Preliminaries

Numerical Integration:

Suppose that we have an evolving system $x(t)$ and a function that tells us how the system changes at any particular time as a function of its current state:

$$\frac{dx}{dt} = \Phi\left(x(t)\right)$$

and suppose that we know the initial state $x(0)$, how should we evolve the system?

Preliminaries

<u>Explicit Integration (Forward Euler)</u>:

If we assume that:

$$\left.\frac{dx}{dt}\right|_{t_0} = \Phi\big(x(t_0)\big) \approx \frac{x(t_0+\delta)-x(t_0)}{\delta}$$

Preliminaries

Explicit Integration (Forward Euler):

If we assume that:

$$\left.\frac{dx}{dt}\right|_{t_0} = \Phi\big(x(t_0)\big) \approx \frac{x(t_0+\delta) - x(t_0)}{\delta}$$

We get:

$$x(t_0+\delta) \approx x\big(t_0\big) + \delta \left.\frac{dx}{dt}\right|_{t_0} = x\big(t_0\big) + \delta\,\Phi\big(x(t_0)\big)$$

which is precisely how we evolved the surface when performing mean curvature flow.

Preliminaries

Explicit Integration (Forward Euler):

If we assume that:

$$\left.\frac{dx}{dt}\right|_{t_0} = \Phi\left(x(t_0)\right) \approx \frac{x(t_0+\delta)-x(t_0)}{\delta}$$

We get:

$$x(t_0+\delta) \approx x\left(t_0\right) + \delta \left.\frac{dx}{dt}\right|_{t_0} = x\left(t_0\right) + \delta\,\Phi\left(x(t_0)\right)$$

which i face

when p

Note:

In this interpretation, we treat the derivative at time *t* as a predictor of the state at the next time-step

Preliminaries

<u>Implicit Integration (Backward Euler)</u>:

If we assume that:

$$\left.\frac{dx}{dt}\right|_{t_0} = \Phi\big(x(t_0)\big) \approx \frac{x(t_0) - x(t_0 - \delta)}{\delta}$$

We get:

$$x(t_0) \approx x\big(t_0 - \delta\big) + \delta \left.\frac{dx}{dt}\right|_{t_0} = x\big(t_0 - \delta\big) + \delta\, \Phi\big(x(t_0)\big)$$

$$\Downarrow$$

$$x(t_0 + \delta) \approx x\big(t_0\big) + \delta \left.\frac{dx}{dt}\right|_{t_0 + \delta} = x\big(t_0\big) + \delta\, \Phi\big(x(t_0 + \delta)\big)$$

Preliminaries

Implicit Integration (Backward Euler):

If we assume that:

$$\left.\frac{dx}{dt}\right|_{t_0} = \Phi\big(x(t_0)\big) \approx \frac{x(t_0) - x(t_0 - \delta)}{\delta}$$

We get:

$$x(t_0 + \delta) \approx x\big(t_0\big) + \delta \left.\frac{dx}{dt}\right|_{t_0 + \delta} = x\big(t_0\big) + \delta\, \Phi\big(x(t_0 + \delta)\big)$$

The challenge is that now our updated state depends on change that is defined by the state $x(t_0+\delta)$ that we don't know.

Preliminaries

Implicit Integration (Backward Euler):

$$x(t_0+\delta) \approx x(t_0) + \delta \frac{dx}{dt}\bigg|_{t_0+\delta} = x(t_0) + \delta\,\Phi\big(x(t_0+\delta)\big)$$

We could solve this using starting with an initial guess for the state $y=x(t_0)$ at $t_0+\delta$ and use this to predict the new state:

$$x(t_0+\delta) = x(t_0) + \delta\,\Phi(y)$$

Preliminaries

Implicit Integration (Backward Euler):

$$x(t_0+\delta) \approx x(t_0) + \delta \left.\frac{dx}{dt}\right|_{t_0+\delta} = x(t_0) + \delta\, \Phi\big(x(t_0+\delta)\big)$$

We could solve this using starting with an initial guess for the state $y=x(t_0)$ at $t_0+\delta$ and use this to predict the new state:

$$x(t_0+\delta) = x(t_0) + \delta\, \Phi(y)$$

Then, depending on the difference between y and the predicted state $x(t_0)+\delta\Phi(y)$, we modify our guess.

Preliminaries

Implicit Integration (Backward Euler):

$$x(t_0+\delta) \approx x(t_0) + \delta \left.\frac{dx}{dt}\right|_{t_0+\delta} = x(t_0) + \delta\, \Phi\big(x(t_0+\delta)\big)$$

Q: Given the difference between the guess and the predicted state, how do we modify the guess?

Preliminaries

Implicit Integration (Backward Euler):

$$x(t_0+\delta) \approx x(t_0) + \delta \left.\frac{dx}{dt}\right|_{t_0+\delta} = x(t_0) + \delta\,\Phi\big(x(t_0+\delta)\big)$$

A: Assuming our guess is *y* then the error in our prediction is:

$$\varepsilon = \underbrace{x(t_0) + \delta\,\Phi(y)}_{\text{Prediction}} - \underbrace{y}_{\text{Guess}}$$

Preliminaries

Implicit Integration (Backward Euler):

$$x(t_0+\delta) \approx x(t_0) + \delta \left.\frac{dx}{dt}\right|_{t_0+\delta} = x(t_0) + \delta\,\Phi\big(x(t_0+\delta)\big)$$

A: Assuming our guess is *y* then the error in our prediction is:

$$\varepsilon = x(t_0) + \delta\,\Phi(y) - y$$

Setting our new guess to be *ŷ*, we want the difference between the original prediction and the new prediction to be -ε:

$$-\varepsilon = \underbrace{\big(\delta\,\Phi(\hat{y}) - \hat{y}\big)}_{\text{New Prediction}} - \underbrace{\big(\delta\,\Phi(y) - y\big)}_{\text{Old Prediction}}$$

Preliminaries

Implicit Integration (Backward Euler):

$$x(t_0+\delta) \approx x(t_0) + \delta \left.\frac{dx}{dt}\right|_{t_0+\delta} = x(t_0) + \delta\,\Phi\big(x(t_0+\delta)\big)$$

A: To get the prediction difference:

$$-\varepsilon = \underbrace{\big(\delta\,\Phi(\hat{y}) - \hat{y}\big)}_{\text{New Prediction}} - \underbrace{\big(\delta\,\Phi(y) - y\big)}_{\text{Old Prediction}}$$

we linearize the function Φ. Setting $\hat{y}=y+\Delta$:

$$\Phi(y+\Delta) \approx \Phi(y) + d\Phi_{\hat{y}}\Delta$$

Preliminaries

Implicit Integration (Backward Euler):

$$x(t_0+\delta) \approx x(t_0) + \delta \left.\frac{dx}{dt}\right|_{t_0+\delta} = x(t_0) + \delta\,\Phi\big(x(t_0+\delta)\big)$$

A: To get the prediction difference:

$$-\varepsilon = \underbrace{\big(\delta\,\Phi(\hat{y}) - \hat{y}\big)}_{\text{New Prediction}} - \underbrace{\big(\delta\,\Phi(y) - y\big)}_{\text{Old Prediction}}$$

we linearize the function Φ. Setting $\hat{y}=y+\Delta$:

$$\Phi(y+\Delta) \approx \Phi(y) + d\Phi_{\hat{y}}\Delta$$

Note:

If Φ is the (negative) gradient of an energy, $d\Phi$ is the energy's Hessian.

Preliminaries

<u>Implicit Integration (Backward Euler)</u>:

$$x(t_0+\delta) \approx x(t_0) + \delta \left.\frac{dx}{dt}\right|_{t_0+\delta} = x(t_0) + \delta\,\Phi\big(x(t_0+\delta)\big)$$

A: To get the prediction difference:

$$-\varepsilon = \big(\delta\,\Phi(\hat{y}) - \hat{y}\big) - \big(\delta\,\Phi(y) - y\big)$$

$$\Phi(y+\Delta) \approx \Phi(y) + d\Phi_{\hat{y}}\Delta$$

This gives:

$$-\varepsilon = \big(\delta\,\Phi(\hat{y}) - \hat{y}\big) - \big(\delta\,\Phi(y) - y\big)$$

Preliminaries

Implicit Integration (Backward Euler):

$$x(t_0+\delta) \approx x(t_0) + \delta \left.\frac{dx}{dt}\right|_{t_0+\delta} = x(t_0) + \delta\, \Phi\big(x(t_0+\delta)\big)$$

A: To get the prediction difference:

$$-\varepsilon = \big(\delta\, \Phi(\hat{y}) - \hat{y}\big) - \big(\delta\, \Phi(y) - y\big)$$

$$\Phi(y+\Delta) \approx \Phi(y) + d\Phi_{\hat{y}}\Delta$$

This gives:

$$\begin{aligned} -\varepsilon &= \big(\delta\, \Phi(\hat{y}) - \hat{y}\big) - \big(\delta\, \Phi(y) - y\big) \\ &= \big(\delta\, \Phi(y+\Delta) - y - \Delta\big) - \big(\delta\, \Phi(y) - y\big) \end{aligned}$$

Preliminaries

Implicit Integration (Backward Euler):

$$x(t_0+\delta) \approx x(t_0) + \delta \left.\frac{dx}{dt}\right|_{t_0+\delta} = x(t_0) + \delta\,\Phi\big(x(t_0+\delta)\big)$$

A: To get the prediction difference:

$$-\varepsilon = \big(\delta\,\Phi(\hat{y}) - \hat{y}\big) - \big(\delta\,\Phi(y) - y\big)$$

$$\Phi(y+\Delta) \approx \Phi(y) + d\Phi_{\hat{y}}\Delta$$

This gives:

$$\begin{aligned}
-\varepsilon &= \big(\delta\,\Phi(\hat{y}) - \hat{y}\big) - \big(\delta\,\Phi(y) - y\big) \\
&= \big(\delta\,\Phi(y+\Delta) - y - \Delta\big) - \big(\delta\,\Phi(y) - y\big) \\
&\approx \big(\delta\,\Phi(y) + \delta\, d\Phi_{\hat{y}}\Delta - y - \Delta\big) - \big(\delta\,\Phi(y) - y\big)
\end{aligned}$$

Preliminaries

Implicit Integration (Backward Euler):

$$x(t_0+\delta) \approx x(t_0) + \delta \left.\frac{dx}{dt}\right|_{t_0+\delta} = x(t_0) + \delta\,\Phi\big(x(t_0+\delta)\big)$$

A: To get the prediction difference:

$$-\varepsilon = \left(\delta\,\Phi(\hat{y}) - \hat{y}\right) - \left(\delta\,\Phi(y) - y\right)$$

$$\Phi(y+\Delta) \approx \Phi(y) + d\Phi_{\hat{y}}\Delta$$

This gives:

$$\begin{aligned}
-\varepsilon &= \left(\delta\,\Phi(\hat{y}) - \hat{y}\right) - \left(\delta\,\Phi(y) - y\right) \\
&= \left(\delta\,\Phi(y+\Delta) - y - \Delta\right) - \left(\delta\,\Phi(y) - y\right) \\
&\approx \left(\delta\,\Phi(y) + \delta\, d\Phi_{\hat{y}}\Delta - y - \Delta\right) - \left(\delta\,\Phi(y) - y\right) \\
&= \left(\delta\, d\Phi_{\hat{y}}\Delta - \Delta\right)
\end{aligned}$$

Preliminaries

Implicit Integration (Backward Euler):

$$x(t_0+\delta) \approx x(t_0) + \delta \left.\frac{dx}{dt}\right|_{t_0+\delta} = x(t_0) + \delta\, \Phi\big(x(t_0+\delta)\big)$$

A: Putting all this together, we obtain the modified guess $\hat{y}=y+\Delta$, by solving the system:

$$-\varepsilon = \left(\delta\, d\Phi_{\hat{y}} - 1\right)\Delta$$

to get the offset Δ that takes us from the initial guess $y=x(t_0)$ to the improved guess $\hat{y}=x(t_0+\delta)$.

Preliminaries

Implicit Integration (Backward Euler):

$$-\varepsilon = \left(\delta\, d\Phi_{\hat{y}} - 1\right)\Delta$$

Problem:

Computing the offset Δ required evaluating the derivative $d\Phi$ at $\hat{y}$, but we don't know $\hat{y}$!

Preliminaries

Implicit Integration (Backward Euler):

$$-\varepsilon = \left(\delta\, d\Phi_{\hat{y}} - 1\right)\Delta$$

Solution (Iterative):

Use an iterative approach, defining a sequence of guesses $\{y^0, y^1, \ldots\}$ where we compute the improved guess y^{i+1} by using the derivative matrix computed at y^i.

Preliminaries

Implicit Integration (Backward Euler):

$$-\varepsilon = \left(\delta\, d\Phi_{\hat{y}} - 1\right)\Delta$$

Solution (Iterative):

Use an iterative approach, defining a sequence of guesses $\{y^0, y^1, \ldots\}$ where we compute the improved guess y^{i+1} by using the derivative matrix computed at y^i.

This means that we have to define and solve the linear system $d\Phi_{y^i}$ at each internal iteration.

Preliminaries

Implicit Integration (Backward Euler):

$$-\varepsilon = \left(\delta \, d\Phi_{\hat{y}} - 1\right)\Delta$$

Solution (Semi-Implicit):

Just use the derivative matrix from the initial guess.

Preliminaries

Implicit Integration (Backward Euler):

$$-\varepsilon = \left(\delta\, d\Phi_{\hat{y}} - 1\right)\Delta$$

Solution (Semi-Implicit):

Just use the derivative matrix from the initial guess.

This corresponds to interpreting positional derivatives as backward-predicting while velocity derivatives as forward-predicting.

$$x(t_0+\delta) \approx x\left(t_0\right) + \delta \left.\frac{dx}{dt}\right|_{t_0+\delta} \qquad \Phi\left(y+\Delta\right) \approx \Phi\left(y\right) + d\Phi_y \Delta$$

Preliminaries

Implicit Integration (Backward Euler):

$$-\varepsilon = \left(\delta\, d\Phi_{\hat{y}} - 1\right)\Delta$$

Solution (Semi-Implicit):

Since our initial guess is $y=x(t_0)$, our error is:

$$\varepsilon = x\left(t_0\right) + \delta\,\Phi\left(y\right) - y = \delta\,\Phi\left(x(t_0)\right)$$

Preliminaries

Implicit Integration (Backward Euler):

$$-\varepsilon = \left(\delta\, d\Phi_{\hat{y}} - 1\right)\Delta$$

Solution (Semi-Implicit):

Since our initial guess is $y=x(t_0)$, our error is:

$$\varepsilon = x(t_0) + \delta\,\Phi(y) - y = \delta\,\Phi\left(x(t_0)\right)$$

Since our modified guess will be the state for the next time-step, this gives:

$$x(t_0+\delta) = x(t_0) + \left(I - \delta\, d\Phi_{x(t_0)}\right)^{-1}\left(\delta\,\Phi\left(x(t_0)\right)\right)$$

$$= x(t_0) + \left(\frac{I}{\delta} - d\Phi_{x(t_0)}\right)^{-1}\left(\Phi\left(x(t_0)\right)\right)$$

Preliminaries

Derivatives:

Given a function $F:\mathbf{R}^n \rightarrow \mathbf{R}^n$, the derivative of F is an nxn matrix dF that describes the (tiny) change of each output coefficient as a function of a (tiny) change in each of the input coefficients.

Preliminaries

<u>Derivatives</u>:

Given a function $F:\mathbf{R}^n \rightarrow \mathbf{R}^n$, the derivative of F is an nxn matrix dF that describes the (tiny) change of each output coefficient as a function of a (tiny) change in each of the input coefficients.

In the case that F is linear, then F is its own derivative.

Preliminaries

<u>Mean Curvature Flow</u>:

Given a surface, we defined an energy that was the area of surface and we showed that the gradient of the energy was proportional to the mean curvature vector:

$$-\nabla E\left(\left\{v_1,\ldots,v_n\right\}\right)=-L_{\vec{v}}\vec{v}$$

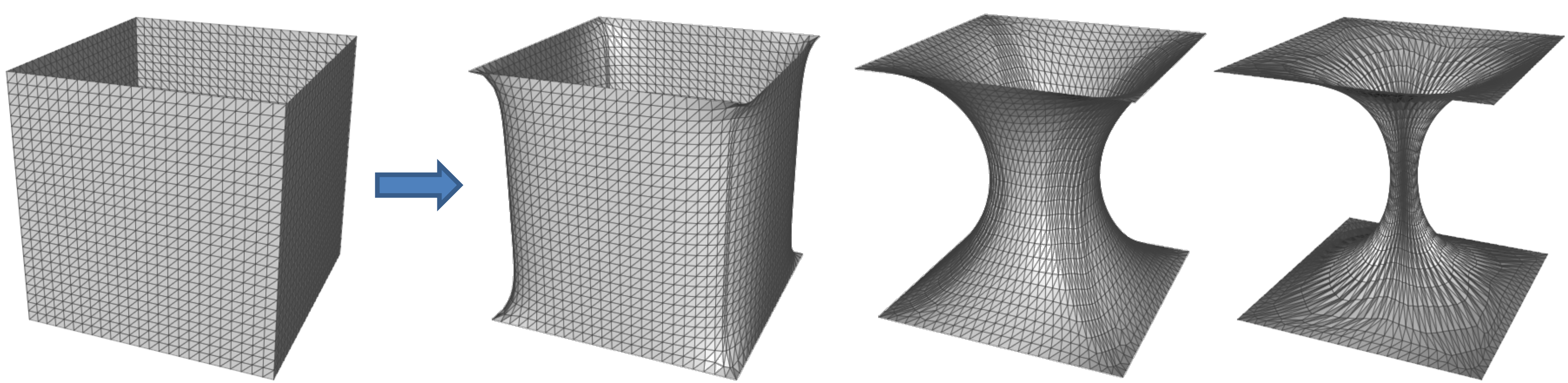
where $L_{\vec{v}}$ is the cotangent-weight Laplacian defined by the vertices $\vec{v}$.

Preliminaries

Mean Curvature Flow:

Thus, to minimize the area, we offset points on the surface in the direction of the negative mean curvature:

$$\vec{v}^{(t+1)} = \vec{v}^{(t)} - \delta L_{\vec{v}(t)} \vec{v}^{(t)}$$



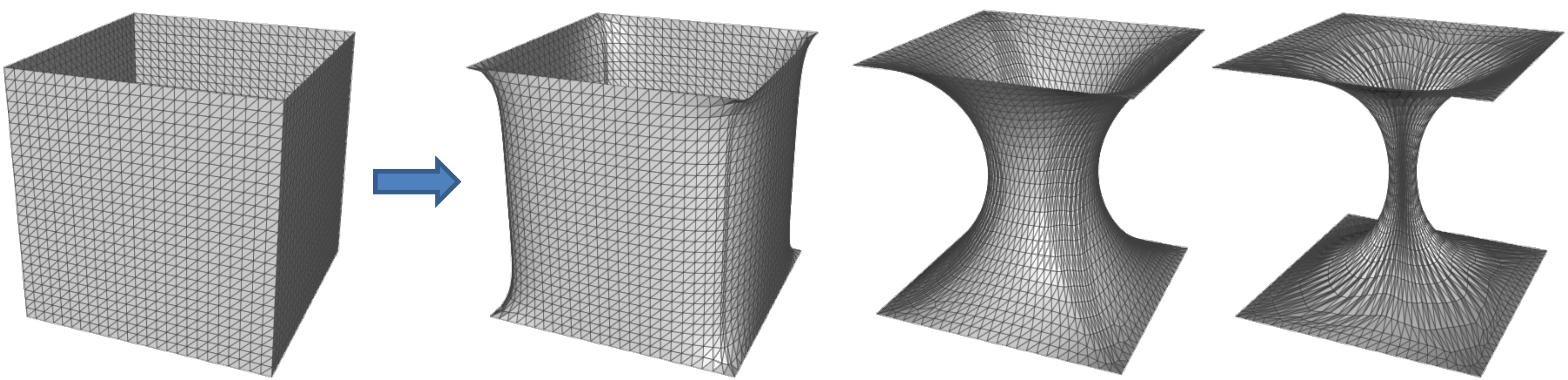
Preliminaries

Mean Curvature Flow:

Thus, to minimize the area, we offset points on the surface in the direction of the negative mean curvature:

$$\vec{v}^{(t+1)} = \vec{v}^{(t)} - \delta L_{\vec{v}(t)} \vec{v}^{(t)}$$

Which amounts to explicit integration.



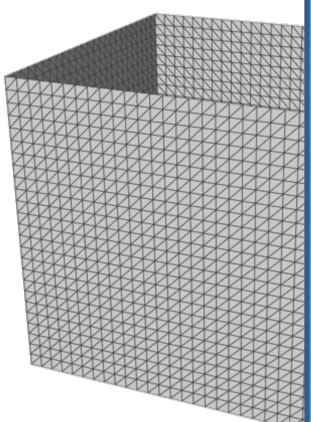
Preliminaries

Mean Curvature Flow:

Thus, to minimize the area, we offset points on the surface in the direction of the negative mean curvature:

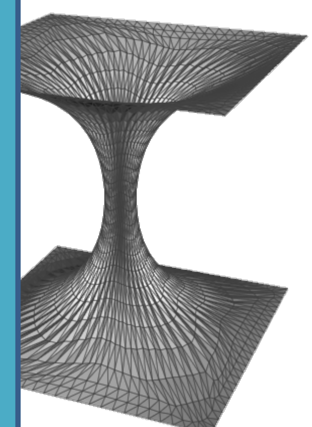
$$\vec{v}^{(t+1)} = \vec{v}^{(t)} - \delta L_{\vec{v}(t)} \vec{v}^{(t)}$$

Which amounts to explicit integration



Note:

Since the geometry changes at each time-step, we have to compute the new cotangent-weight Laplacian, $L_{\vec{v}(t)}$ at each time-step.



Preliminaries

Mean Curvature Flow:

Thus, to minimize the area, we offset points on the surface in the direction of the negative mean curvature:

$$\vec{v}^{(t+1)} = \vec{v}^{(t)} - \delta L_{\vec{v}^{(t)}} \vec{v}^{(t)}$$

In practice, the step-size δ has to be small (proportional to the smallest edge-length).

Can we do better?

Preliminaries

Mean Curvature Flow (Semi-Implicit):

Using a semi-implicit scheme, we can generate more stable integration by solving a linear system at each step:

$$\vec{v}^{(t+1)} = \vec{v}^{(t)} - \left(\frac{I}{\delta} + dL_{\vec{v}(t)} \right)^{-1} \left(L_{\vec{v}(t)} \vec{v}^{(t)} \right)$$

Preliminaries

Mean Curvature Flow (Semi-Implicit):

Using a semi-implicit scheme, we can generate more stable integration by solving a linear system at each step:

$$\vec{v}^{(t+1)} = \vec{v}^{(t)} - \left(\frac{I}{\delta} + dL_{\vec{v}^{(t)}} \right)^{-1} \left(L_{\vec{v}^{(t)}} \vec{v}^{(t)} \right)$$

To do this right, we would differentiate the cotangent-weight entries in the Laplacian matrix $L_{\vec{v}}$ since they depend on the vertex positions.

Preliminaries

<u>Mean Curvature Flow (Semi-Implicit)</u>:

Using a semi-implicit scheme, we can generate more stable integration by solving a linear system at each step:

$$\vec{v}^{(t+1)} = \vec{v}^{(t)} - \left(\frac{I}{\delta} + dL_{\vec{v}(t)}\right)^{-1} \left(L_{\vec{v}(t)}\vec{v}^{(t)}\right)$$

However, if we pretend that they are fixed, then the derivative of the Laplacian matrix is just the Laplacian matrix, and we get:

$$\vec{v}^{(t+1)} = \vec{v}^{(t)} - \left(\frac{I}{\delta} + L_{\vec{v}(t)}\right)^{-1} \left(L_{\vec{v}(t)}\vec{v}^{(t)}\right)$$

Preliminaries

<u>Mean Curvature Flow (Semi-Implicit)</u>:

Using a semi-implicit scheme, we can generate more stable integration by solving a linear system at each step:

$$\vec{v}^{(t+1)} = \vec{v}^{(t)} - \left(\frac{I}{\delta} + dL_{\vec{v}^{(t)}}\right)^{-1}\left(L_{\vec{v}^{(t)}}\vec{v}^{(t)}\right)$$

However, if we pretend that they are fixed, then

<u>Note</u>:

An additional advantage of this simplification is that the derivative of the Laplacian is now of size *nxn* instead of 3*n*x3*n*.