1 Problem Setup

As input, we are given a graph $G = (V_G, E_G)$, where each vertex is a geographic position $s_i \in S^2$, and each edge (i,j) has an associated (Olivier-Ricci) curvature $R_{i,j} \in (-2, 1)$ and an associated latency $t_{i,j} \in \mathbb{R}_{\geq 0}$.

Intuitively, we want to return a surface in $\mathbb{R}^3$ that is the graph of a function $\rho : S^2 \to \mathbb{R}_{>0}$ whose geodesics $g_{i,j}$ between s_i and s_j (and their missing ρ-coordinates) have length $\phi_{i,j}$ that is in a linear relationship with the latency.

The strategy to realize this intuition is to create a mesh $M = (V_M, E_M)$ supported on a subset of S^2 that contains our input positions. Let P be the support. Then for each $s_i \in P$, we want to assign a $\rho_i \in \mathbb{R}_{>0}$, which in turn gives a point $v_i = (s_i, \rho_i) \in V$. This setup is made explicit in `mesh/sphere.py`.

A similar setup is found in `mesh/rectangle.py`, where we use $[0,1]^2$ instead of S^2. In general, this setup

2 Objective/Loss Functions

To enforce that the mesh approximates our desired surface, we define the objective functions

$$
\begin{aligned}
\mathcal{L}_{\text{geodesic}}(M) &\triangleq \sum_{e \in E_G} (\text{least squares residual of edge } e)^2, \\
\mathcal{L}_{\text{smooth}}(M) &\triangleq -\rho^\intercal L_C \rho, \\
\mathcal{L}_{\text{curvature}}(M) &\triangleq \sum_{\substack{v \in V_M \\ v \text{ close to } (i,j)}} \big(\kappa(v) - R_{i,j}\big)^2, \\
\mathcal{L}(M) &\triangleq \lambda_{\text{geodesic}} \mathcal{L}_{\text{geodesic}}(M) + \lambda_{\text{curvature}} \mathcal{L}_{\text{curvature}}(M) + \lambda_{\text{smooth}} \mathcal{L}_{\text{smooth}}(M),
\end{aligned}
$$

where the λ's are tunable hyperparameters. The other variables will be defined in the upcoming subsections. Our goal is then to minimize $\mathcal{L}(M)$.

Note that the loss functions (particularly the geodesic and total ones) also have a dependence on the measured latencies. We omit that as a written parameter because they are treated as fixed (we are really optimizing over the manifold, not over the measured latencies).

2.1 Laplacian

Some mesh notation first. If i and j are two indices vertices for which (v_i, v_j) is a half-edge, let $c(i,j)$ be the index such that $v_i \to v_j \to v_{c(i,j)}$ traces a triangle counterclockwise. Note that this index exists and is unique assuming we have a mesh without boundary. On a mesh with boundary, if no $c(i,j)$ exists, then the half-edge (v_i, v_j) lies on the boundary.

We define the following variables:

$N_{i,j}$	Outward normal of triangle $v_i \to v_j \to v_{c(i,j)}$
$A_{i,j}$	Area of triangle $v_i \to v_j \to v_{c(i,j)}$
$D_{i,j}$	Vertex triangle areas; diagonal
$\theta_{i,j}$	Measure of $\angle v_i v_{c(i,j)} v_j$
L_C^{Neumann}	Cotangent operator with zero-Neumann boundary condition
$L_C^{\text{Dirichlet}}$	Cotangent operator with zero-Dirichlet boundary condition
L_C	Cotangent operator in the no-boundary case; sparse

2.1.1 Forward Computation

We have the following (standard) definition of the Laplace-Beltrami operator on a mesh:

$$
\begin{aligned}
N_{i,j} &= \left(v_i - v_{c(i,j)}\right) \times \left(v_j - v_{c(i,j)}\right), \\
A_{i,j} &= \frac{1}{2} \left\| N_{i,j} \right\|_2, \\
D_{i,j} &= \begin{cases} \frac{1}{3} \displaystyle\sum_{\substack{k \\ (v_i, v_k) \text{ is} \\ \text{a half-edge}}} A_{i,k} & \text{if } i = j, \\ 0 & \text{otherwise,} \end{cases}
\end{aligned}
$$

$$\cot\left(\theta_{i,j}\right) = \frac{\left(v_i - v_{c\left(i,j\right)}\right) \cdot \left(v_j - v_{c\left(i,j\right)}\right)}{2A_{i,j}},$$

$$\left(L_C^{\text{Neumann}}\right)_{i,j} = \begin{cases} \frac{1}{2}\cot\left(\theta_{i,j}\right) & \text{if } \left(v_i, v_j\right) \text{ is a half-edge on } \partial M, \\ \frac{1}{2}\cot\left(\theta_{j,i}\right) & \text{if } \left(v_j, v_i\right) \text{ is a half-edge on } \partial M, \\ \frac{1}{2}\left(\cot\left(\theta_{i,j}\right) + \cot\left(\theta_{j,i}\right)\right) & \text{if } \left(v_i, v_j\right) \text{ is a half-edge not on } \partial M, \\ -\frac{1}{2}\left(\sum\limits_{\substack{k \\ \left(v_i, v_k\right) \text{ is} \\ \text{a half-edge}}} \cot\left(\theta_{i,k}\right) + \sum\limits_{\substack{k \\ \left(v_k, v_i\right) \text{ is} \\ \text{a half-edge}}} \cot\left(\theta_{k,i}\right)\right) & \text{if } i = j \\ 0 & \text{otherwise,} \end{cases}$$

$$\left(L_C^{\text{Dirichlet}}\right)_{i,j} = \begin{cases} \frac{1}{2}\left(\cot\left(\theta_{i,j}\right) + \cot\left(\theta_{j,i}\right)\right) & \text{if } \left(v_i, v_j\right) \text{ is a half-edge and } i, j \notin \partial M, \\ -\frac{1}{2}\sum\limits_{\substack{k \notin \partial M \\ \left(v_i, v_k\right) \text{ and } \left(v_k, v_i\right) \\ \text{are half-edges}}} \left(\cot\left(\theta_{i,k}\right) + \cot\left(\theta_{k,i}\right)\right) & \text{if } i = j \text{ and } v_i \notin \partial M, \\ 0 & \text{otherwise.} \end{cases}$$

Flipping our attention back to meshes without boundary, the two definitions above coincide, so we can write

$$L_C = L_C^{\text{Neumann}} = L_C^{\text{Dirichlet}}.$$

2.1.2 Reverse Computation

For the ease of notation, assume that we are using the spherical setup, so $v_\ell = \rho_\ell s_\ell$.

We compute

$$\frac{\partial v_i}{\partial \rho_\ell} = \begin{cases} s_i & \text{if } \ell = i, \\ 0 & \text{otherwise,} \end{cases}$$

$$\frac{\partial N_{i,j}}{\partial \rho_\ell} = \begin{cases} \left(v_{c\left(i,j\right)} - v_j\right) \times \frac{\partial v_\ell}{\partial \rho_\ell} & \text{if } \ell = i, \\ \left(v_i - v_{c\left(i,j\right)}\right) \times \frac{\partial v_\ell}{\partial \rho_\ell} & \text{if } \ell = j, \\ \left(v_j - v_i\right) \times \frac{\partial v_\ell}{\partial \rho_\ell} & \text{if } \ell = c(i,j), \\ 0 & \text{otherwise,} \end{cases}$$

$$\frac{\partial A_{i,j}}{\partial \rho_\ell} = \frac{1}{4A_{i,j}} N_{i,j} \cdot \frac{\partial N_{i,j}}{\partial \rho_\ell},$$

$$\left(\frac{\partial D}{\partial \rho_\ell}\right)_{i,j} = \begin{cases} \frac{1}{3}\sum\limits_{\substack{k \\ \left(v_i, v_k\right) \text{ is} \\ \text{a half-edge}}} \frac{\partial A_{i,k}}{\partial \rho_\ell} & \text{if } i = j, \\ 0 & \text{otherwise,} \end{cases}$$

$$\frac{\partial}{\partial \rho_\ell}\cot\left(\theta_{i,j}\right) = \begin{cases} \frac{\left(v_j - v_{c\left(i,j\right)}\right) \cdot \frac{\partial v_\ell}{\partial \rho_\ell} - 2\cot\left(\theta_{i,j}\right)\frac{\partial A_{i,j}}{\partial \rho_\ell}}{2A_{i,j}} & \text{if } \ell = i, \\ \frac{\left(v_i - v_{c\left(i,j\right)}\right) \cdot \frac{\partial v_\ell}{\partial \rho_\ell} - 2\cot\left(\theta_{i,j}\right)\frac{\partial A_{i,j}}{\partial \rho_\ell}}{2A_{i,j}} & \text{if } \ell = j, \\ \frac{\left(2v_{c\left(i,j\right)} - v_i - v_j\right) \cdot \frac{\partial v_\ell}{\partial \rho_\ell} - 2\cot\left(\theta_{i,j}\right)\frac{\partial A_{i,j}}{\partial \rho_\ell}}{2A_{i,j}} & \text{if } \ell = c(i,j), \\ 0 & \text{otherwise,} \end{cases}$$

$$\left(\frac{\partial L_C^{\text{Neumann}}}{\partial \rho_\ell}\right)_{i,j} = \begin{cases} \frac{1}{2}\frac{\partial}{\partial \rho_\ell}\cot\left(\theta_{i,j}\right) & \text{if } \left(v_i, v_j\right) \text{ is a half-edge on } \partial M, \\ \frac{1}{2}\frac{\partial}{\partial \rho_\ell}\cot\left(\theta_{j,i}\right) & \text{if } \left(v_j, v_i\right) \text{ is a half-edge on } \partial M, \\ \frac{1}{2}\left(\frac{\partial}{\partial \rho_\ell}\cot\left(\theta_{i,j}\right) + \frac{\partial}{\partial \rho_\ell}\cot\left(\theta_{j,i}\right)\right) & \text{if } \left(v_i, v_j\right) \text{ is a half-edge not on } \partial M, \\ -\frac{1}{2}\left(\sum_{\substack{k \\ \left(v_i, v_k\right) \text{ is} \\ \text{a half-edge}}} \frac{\partial}{\partial \rho_\ell}\cot\left(\theta_{i,k}\right) + \sum_{\substack{k \\ \left(v_k, v_i\right) \text{ is} \\ \text{a half-edge}}} \frac{\partial}{\partial \rho_\ell}\cot\left(\theta_{k,i}\right)\right) & \text{if } i = j \\ 0 & \text{otherwise,} \end{cases}$$

$$\left(\frac{\partial L_C^{\text{Dirichlet}}}{\partial \rho_\ell}\right)_{i,j} = \begin{cases} \frac{1}{2}\left(\frac{\partial}{\partial \rho_\ell}\cot\left(\theta_{i,j}\right) + \frac{\partial}{\partial \rho_\ell}\cot\left(\theta_{j,i}\right)\right) & \text{if } \left(v_i, v_j\right) \text{ is a half-edge and } i, j \notin \partial M, \\ -\frac{1}{2}\sum_{\substack{k \notin \partial M \\ \left(v_i, v_k\right) \text{ and } \left(v_k, v_i\right) \\ \text{are half-edges}}} \left(\frac{\partial}{\partial \rho_\ell}\cot\left(\theta_{i,k}\right) + \frac{\partial}{\partial \rho_\ell}\cot\left(\theta_{k,i}\right)\right) & \text{if } i = j \text{ and } v_i \notin \partial M, \\ 0 & \text{otherwise.} \end{cases}$$

2.2 Geodesic Distance via the Heat Method

Here are the variables used for this part of the computation:

γ	Set of points in V_M
h	Mean half-edge length
δ^γ	Heat source (indicator on γ)
$u^{\gamma,\text{N}}$	Heat flow with zero-Neumann boundary condition
$u^{\gamma,\text{D}}$	Heat flow with zero-Dirichlet boundary condition
u^γ	Heat flow
$q_{i,j}^\gamma$	Intermediate value for computation
$m_{i,j}^\gamma$	Intermediate value for computation
$X_{i,j}^\gamma$	Unit vector in same direction as $\nabla u_{i,j}^\gamma$
$p_{i,j}^\gamma$	Intermediate value for computation
ϕ^γ	Vector of geodesic distances

2.2.1 Forward Computation

Say we want to find the geodesic distances to a set of points $\gamma \subseteq V_M$. Following the Crane et al's Heat Method, we use the (approximate) heat flow u^γ, where

$$\begin{aligned} h &= \text{TODO}, \\ \delta^\gamma &= \begin{cases} 1 & \text{if } v_i \in \gamma, \\ 0 & \text{if } v_i \notin \gamma, \end{cases} \\ u^{\gamma,\text{N}} &= \left(D - h^2 L_C^{\text{N}}\right)^{-1}\delta^\gamma, \\ u^{\gamma,\text{D}} &= \left(D - h^2 L_C^{\text{D}}\right)^{-1}\delta^\gamma, \\ u^\gamma &= \frac{1}{2}\left(u^{\gamma,\text{N}} + u^{\gamma,\text{D}}\right), \\ q_{i,j}^\gamma &= u_i^\gamma\left(v_{c(i,j)} - v_j\right), \\ m_{i,j}^\gamma &= q_{i,j}^\gamma + q_{j,c(i,j)}^\gamma + q_{c(i,j),i}^\gamma, \\ \left(\nabla u^\gamma\right)_{i,j} &= N_{i,j} \times m_{i,j}^\gamma, \\ X_{i,j}^\gamma &= -\frac{\left(\nabla u^\gamma\right)_{i,j}}{\left\|\left(\nabla u^\gamma\right)_{i,j}\right\|_2}, \\ p_{i,j} &= \cot\left(\theta_{i,j}\right)\left(v_j - v_i\right), \end{aligned}$$

$$\left(\nabla \cdot X^\gamma\right)_i = \frac{1}{2} \sum_{\substack{k \\ (v_i, v_k) \text{ is} \\ \text{a half-edge}}} \left(p_{i,k} - p_{c(i,k),i}\right) \cdot X^\gamma_{i,k},$$

$$\phi^\gamma = \left(L_C^{\mathrm{N}}\right)^+ \cdot \left(\nabla \cdot X^\gamma\right).$$

Here, $\left(L_C^{\mathrm{N}}\right)^+$ is the pseudoinverse of L_C^{N} (as it is singular).

Note that we're being careful about which pieces have a dependence on γ, as we can reuse certain computations if we want to compute distances from multiple sources. We can get the distance matrix (that is, get rid of the γ dependence) from

$$\phi_{i,j} = \left(\phi^{\{v_j\}}\right)_i.$$

2.2.2 Reverse Computation

Note that $c\left(i, c(j,i)\right) = j$. This is helpful for reindexing some sums (in particular, the one for $\nabla \cdot X$).

We then have the following partial derivatives:

$$\begin{aligned}
\frac{\partial h}{\partial \rho_\ell} &= \text{TODO} \\
\frac{\partial u^{\gamma,\mathrm{N}}}{\partial \rho_\ell} &= -\left(D - h^2 L_C^{\mathrm{N}}\right)^{-1} \left(\frac{\partial D}{\partial \rho_\ell} - 2h \frac{\partial h}{\partial \rho_\ell} L_C^{\mathrm{N}} - h^2 \frac{\partial L_C}{\partial \rho_\ell}\right) u^{\gamma,\mathrm{N}}, \\
\frac{\partial u^{\gamma,\mathrm{D}}}{\partial \rho_\ell} &= -\left(D - h^2 L_C^{\mathrm{D}}\right)^{-1} \left(\frac{\partial D}{\partial \rho_\ell} - 2h \frac{\partial h}{\partial \rho_\ell} L_C^{\mathrm{D}} - h^2 \frac{\partial L_C}{\partial \rho_\ell}\right) u^{\gamma,\mathrm{D}}, \\
\frac{\partial u^{\gamma}}{\partial \rho_\ell} &= \frac{1}{2}\left(\frac{\partial u^{\gamma,\mathrm{N}}}{\partial \rho_\ell} + \frac{\partial u^{\gamma,\mathrm{D}}}{\partial \rho_\ell}\right), \\
\frac{\partial q^\gamma_{i,j}}{\partial \rho_\ell} &= \begin{cases} \frac{\partial u^\gamma_i}{\rho_\ell}\left(v_{c(i,j)} - v_j\right) - u^\gamma_i \frac{\partial v_\ell}{\rho_\ell} & \text{if } \ell = j, \\ \frac{\partial u^\gamma_i}{\rho_\ell}\left(v_{c(i,j)} - v_j\right) + u^\gamma_i \frac{\partial v_\ell}{\partial \rho_\ell} & \text{if } \ell = c(i,j), \\ \frac{\partial u^\gamma_i}{\rho_\ell}\left(v_{c(i,j)} - v_j\right) & \text{otherwise,} \end{cases} \\
\frac{\partial m^\gamma_{i,j}}{\partial \rho_\ell} &= \frac{\partial q^\gamma_{i,j}}{\partial \rho_\ell} + \frac{\partial q^\gamma_{j,c(i,j)}}{\partial \rho_\ell} + \frac{\partial q^\gamma_{c(i,j),i}}{\partial \rho_\ell}, \\
\frac{\partial (\nabla u^\gamma)_{i,j}}{\partial \rho_\ell} &= \frac{\partial N_{i,j}}{\partial \rho_\ell} \times m^\gamma_{i,j} + N_{i,j} \times \frac{\partial m^\gamma_{i,j}}{\partial \rho_\ell}, \\
\frac{\partial X^\gamma_{i,j}}{\partial \rho_\ell} &= -\frac{1}{\left\|(\nabla u^\gamma)_{i,j}\right\|_2} \left(I - X^\gamma_{i,j}\left(X^\gamma_{i,j}\right)^\intercal\right) \frac{\partial (\nabla u^\gamma)_{i,j}}{\partial \rho_\ell}, \\
\frac{\partial p_{i,j}}{\partial \rho} &= \begin{cases} \left(\frac{\partial}{\partial \rho_\ell} \cot(\theta_{i,j})\right)(v_j - v_i) - \cot(\theta_{i,j}) \frac{\partial v_\ell}{\rho_\ell} & \text{if } \ell = i, \\ \left(\frac{\partial}{\partial \rho_\ell} \cot(\theta_{i,j})\right)(v_j - v_i) + \cot(\theta_{i,j}) \frac{\partial v_\ell}{\rho_\ell} & \text{if } \ell = j, \\ \left(\frac{\partial}{\partial \rho_\ell} \cot(\theta_{i,j})\right)(v_j - v_i) & \text{if } \ell = c(i,j), \\ 0 & \text{otherwise,} \end{cases} \\
\frac{\partial (\nabla \cdot X^\gamma)_i}{\partial \rho_\ell} &= \frac{1}{2} \sum_{\substack{k \\ (v_i, v_k) \text{ is} \\ \text{a half-edge}}} \left(\left(\frac{\partial p_{i,k}}{\partial \rho_\ell} - \frac{\partial p_{c(i,k),i}}{\partial \rho_\ell}\right) \cdot X^\gamma_{i,k} + \left(p_{i,k} - p_{c(i,k),i}\right) \cdot \frac{\partial X^\gamma_{i,k}}{\partial \rho_\ell}\right) \\
\frac{\partial \phi^\gamma}{\partial \rho_\ell} &= \left(L_C^{\mathrm{N}}\right)^+ \left(\frac{\partial (\nabla \cdot X^\gamma)}{\partial \rho_\ell} - \frac{\partial L_C^{\mathrm{N}}}{\partial \rho_\ell} \phi^\gamma\right).
\end{aligned}$$