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.

2 Objective/Loss Functions

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

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

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 an 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.

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	Cotangent operator; sparse
L_C^{Neumann}	Cotangent operator with zero-Neumann boundary condition
$L_C^{\text{Dirichlet}}$	Cotangent operator with zero-Dirichlet boundary condition

2.1.1 Forward Computation

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

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

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

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

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

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

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

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

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

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

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

$$\left(\frac{\partial L_C^{\text{Neumann}}}{\partial \rho_\ell}\right)_{i,j} = \begin{cases} \frac{1}{2}\left(\frac{\partial}{\partial \rho_\ell}\cot\big(\theta_{i,j}\big) + \frac{\partial}{\partial \rho_\ell}\cot\big(\theta_{j,i}\big)\right) & \text{if } \big(v_i, v_j\big) \text{ or } \big(v_j, v_i\big) \text{ is a half-edge,} \\ -\frac{1}{2} \sum\limits_{\substack{k \\ (v_i, v_k) \text{ or } (v_k, v_i) \\ \text{is a half-edge}}} \left(\frac{\partial}{\partial \rho_\ell}\cot\big(\theta_{i,k}\big) + \frac{\partial}{\partial \rho_\ell}\cot\big(\theta_{k,i}\big)\right) & \text{if } i = j, \\ 0 & \text{otherwise,} \end{cases}$$

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