

7.3 Solving PDEs on incomplete-distance data

Incomplete inter-point distances arise naturally in applications such as protein structure determination from NMR spectroscopy [Crippen and Havel (1988); Berger *et al.* (1999); Leung and Toh (2009); Fang and Toh (2013)] and sensor network localization [Ji and Zha (2004); Biswas *et al.* (2006)]. In this data model, the input is not a set of embedded points but an *incomplete* distance matrix on an unknown point set $\{\mathbf{p}_i\}$ sampled from a manifold \mathcal{M} . While distance geometry traditionally focuses on recovering global coordinates from these distances, there has been less exploration of solving *geometric PDEs* directly from (incomplete) distances. A local strategy was proposed in [Lai and Li (2018)] to reconstruct only *local* coordinates sufficient to approximate intrinsic differential operators pointwise, then assemble numerical solvers for PDEs without global embedding.

Two key observations motivate this approach. First, geometric differential operators (gradient, divergence, Laplace–Beltrami) are *local* and defined pointwise from the manifold metric; second, these operators are invariant under rigid motions and local coordinate choices. Thus, it suffices to perform *local* coordinate reconstruction around each point from available distances, approximate operators intrinsically in the local chart, and build PDE solvers from these local pieces.

7.3.1 Problem setup and local coordinate reconstruction

We consider the setting where the discretization of \mathcal{M} is given by an unknown point set $\{\mathbf{p}_1, \dots, \mathbf{p}_n\} \subset \mathcal{M} \subset \mathbb{R}^D$, but the coordinates of these points are not provided. Instead, only a subset of their pairwise Euclidean distances is known:

$$D_{ij} = \|\mathbf{p}_i - \mathbf{p}_j\|, \quad (i, j) \in \Omega,$$

where $\Omega \subset \{(i, j) \mid 1 \leq j < i \leq n\}$ is the index set of known measurements. If the full distance matrix were available, classical multidimensional scaling (MDS) would allow exact coordinate reconstruction. Here, however, only an *incomplete* distance matrix is given, and yet the objective is to approximate intrinsic differential operators and solve PDEs (e.g. the Laplace–Beltrami eigenproblem, Poisson, heat and Eikonal equations) directly from the incomplete distances. To this end, coordinates need to be reconstructed first. A computational framework for reconstructing local coordinates proceeds as follows.

Step 1: KNN search.

For finding KNN (K-nearest neighbors) for each point \mathbf{p}_i , choose a neighborhood $\mathcal{I}(i)$ of size K using the available distances (or side information if provided), and form the local incomplete distance matrix $D \in \mathbb{R}^{K \times K}$ over $\mathcal{I}(i)$.

Step 2: Local coordinate reconstruction (low-rank completion).

Suppose $\mathcal{I}(i)$ contains points $\mathbf{p}_1, \dots, \mathbf{p}_K$. For $\mathcal{I}(i)$, the Gram matrix $B \in \mathbb{R}^{K \times K}$ is defined as $B_{ij} = \langle \mathbf{p}_i, \mathbf{p}_j \rangle$. The Gram matrix satisfies $B = P^\top P$ where columns of P are coordinates of \mathbf{p}_i . The classical multi-dimensional scaling (MDS) relation between a Gram matrix $B \succeq 0$ and squared distances is

$$D_{ij} = B_{ii} + B_{jj} - 2B_{ij}.$$

Our goal here is to reconstruct the Gram matrix using given data, which are the entries of distance matrix in a subset $\Omega \subset \{(i, j) \mid 1 \leq j < i \leq K\}$. In other words, the given data for B can be formulated as

$$B_{ii} + B_{jj} - 2B_{ij} = D_{ij}, \quad (i, j) \in \Omega, \quad (7.17)$$

which is a linear constraint about unknown Gram matrix B .

For enforcing translation elimination, we can impose $B\mathbf{1} = \mathbf{0}$, which is equivalent to $\sum_i \mathbf{p}_i = \mathbf{0}$.

Notice that $B = PP^\top$ implies that B is real symmetric positive semi-definite and the rank of B should not exceed the dimension of ambient Euclidean space.

For reconstructing the unknown Gram matrix B , we seek a low-rank matrix B satisfying the properties above:

$$\min_{B \in \mathcal{S}} \text{Rank}(B) \quad \text{s.t.} \quad B \succeq 0, \quad \mathcal{R}_\Omega(B) = D_\Omega, \quad B\mathbf{1} = \mathbf{0}, \quad (7.18)$$

where $\mathcal{S} = \{B \in \mathbb{R}^{K \times K} \mid B = B^\top\}$ denotes symmetric matrices, and $\mathcal{R}_\Omega(B) = D_\Omega$ denotes the linear constraint (7.17), i.e., D_Ω is the restriction of D on the index set Ω , and $\mathcal{R}_\Omega(B)(i, j) = B_{ii} + B_{jj} - 2B_{ij}$ for $(i, j) \in \Omega$.

Following matrix completion principles [Candès and Recht (2008)], we relax rank to the matrix nuclear norm. See Appendix A.3 for the definition of the nuclear norm of a matrix. Let $\zeta : \mathcal{S} \rightarrow \mathbb{R}^{K(K+1)/2}$ be the canonical vectorization with inverse ι , and define

$$\mathcal{A} : \mathbb{R}^{K(K+1)/2} \rightarrow \mathbb{R}^{|\Omega|} \times \mathbb{R}^K, \quad \hat{B} \mapsto (\mathcal{R}_\Omega(\iota(\hat{B})), \iota(\hat{B})\mathbf{1}),$$

with target $\tilde{D}_\Omega = (D_\Omega, \mathbf{0})$. Since $B \succeq 0$, we have $\|B\|_* = \text{Tr}(B)$, thus it becomes a SDP (semi-definite programming)

$$\min_{\hat{B} \in \mathbb{R}^{K(K+1)/2}} \text{Tr}(\iota(\hat{B})) \quad \text{s.t.} \quad \mathcal{A}\hat{B} = \tilde{D}_\Omega, \quad \iota(\hat{B}) \succeq 0. \quad (7.19)$$

Algorithm 1 ADMM for local Gram matrix completion (7.19)

-
- 1: **Init:** $\hat{C}^0 = 0$, $H_1^0 = 0$, $H_2^0 = 0$; choose $\mu_1, \mu_2 > 0$ and tolerance ε .
 - 2: **while** $|E(k) - E(k-1)|/E(k) \geq \varepsilon$ **do**
 - 3: Solve (7.21) for \hat{B}^{k+1} .
 - 4: $\hat{C}^{k+1} = \zeta \circ \mathcal{T}_E \circ \iota(\hat{B}^{k+1} + H_2^k)$.
 - 5: $H_1^{k+1} = H_1^k + \mathcal{A}\hat{B}^{k+1} - \tilde{D}_\Omega$;
 - 6: $H_2^{k+1} = H_2^k + \hat{B}^{k+1} - \hat{C}^{k+1}$.
 - 7: $E(k+1) = \text{Tr}(\iota(\hat{B}^{k+1})) + \frac{\mu_1}{2} \|\mathcal{A}\hat{B}^{k+1} - \tilde{D}_\Omega\|_2^2 + \frac{\mu_2}{2} \|\hat{B}^{k+1} - \hat{C}^{k+1}\|_2^2$.
 - 8: **end while**
 - 9: **Return:** $B = \iota(\hat{B})$ and its MDS coordinates on $\mathcal{I}(i)$.
-

Theoretical guarantees differ from standard *restricted isometry property* settings [Recht *et al.* (2010)], but under coherence-type conditions, exact local completion can still be proved by dual-basis analysis [Tasissa and Lai (2018)]. After solving (7.19), local coordinates can be recovered by the eigendecomposition of B , since the Gram matrix satisfies $B = PP^\top$.

Interior-point SDP solvers can become expensive when constraints scale as $\mathcal{O}(K^2)$. Instead, We may consider an ADMM splitting (see Appendix C) by introducing an auxiliary $\hat{C} = \hat{B}$:

$$\min_{\hat{B}, \hat{C}} \text{Tr}(\iota(\hat{B})) \quad \text{s.t.} \quad \mathcal{A}\hat{B} = \tilde{D}_\Omega, \quad \iota(\hat{C}) \succeq 0, \quad \hat{B} = \hat{C}. \quad (7.20)$$

A practical implementation is summarized in Algorithm 1, in which the updates are

$$(\mu_1 \mathcal{A}^* \mathcal{A} + \mu_2) \hat{B}^{k+1} = \mu_1 \mathcal{A}^* (\tilde{D}_\Omega - H_1^k) + \mu_2 (\hat{C}^k - H_2^k) - \zeta(I_K), \quad (7.21)$$

$$\hat{C}^{k+1} = \zeta \circ \mathcal{T}_E \circ \iota(\hat{B}^{k+1} + H_2^k), \quad (7.22)$$

where $\mathcal{T}_E(X) = V \text{diag}(\max\{\Lambda_{ii}, 0\}) V^\top$ is eigenvalue hard-thresholding of a symmetric matrix $X = V\Lambda V^\top$.

7.3.2 Operator approximation from local reconstructions

Once local coordinates for each neighborhood $\mathcal{I}(i)$ are obtained, intrinsic differential operators at \mathbf{p}_i can be approximated by either:

- **MLS method** (Section 7.1): fit local polynomials in tangent coordinates and recover $\nabla_{\mathcal{M}}$, $\text{div}_{\mathcal{M}}$, and $\Delta_{\mathcal{M}}$ from the fitted jets;
- **Local mesh method** (Section 7.2): build a Delaunay triangulation in the local chart and use area/cotangent formulae to assemble per-ring gradient/divergence and the Laplacian.

Both are *intrinsic* (coordinate invariant) and use only local reconstructions, and no global embedding is required. Row-by-row, we assemble discrete operators and key components in PDE solvers such as FEM-like mass and stiffness matrices. By combining local low-rank distance completion with MLS or local mesh discretizations, we obtain an intrinsic, meshless framework for solving PDEs directly from incomplete distance data—circumventing global embedding while retaining accuracy, robustness, and scalability [Lai and Li (2018)].

7.3.3 Numerical examples

Consider points $\{p_1, \dots, p_n\}$ sampled uniformly distributed on a manifold $\mathcal{M} \subset \mathbb{R}^D$. Let D denote the complete pairwise Euclidean distance matrix. To simulate incomplete measurements, select a random subset $\Omega_\gamma \subset \{(i, j) \mid 1 \leq j < i \leq n\}$ containing $|\Omega_\gamma| = \lceil \gamma n(n-1)/2 \rceil$ entries, and restrict to the K nearest neighbors of each point:

$$D_{\gamma, K} = \{D(i, j) \mid j \in \mathcal{I}_K(i), (i, j) \in \Omega_\gamma, i = 1, \dots, n\}.$$

This subsampled matrix $D_{\gamma, K}$ typically retains only a fraction $\gamma K/n$ of all distances, which can be extremely sparse when $K \ll n$. Local coordinate reconstruction (Step 1) and differential operator approximation (Step 2) are then performed based solely on this incomplete distance information.

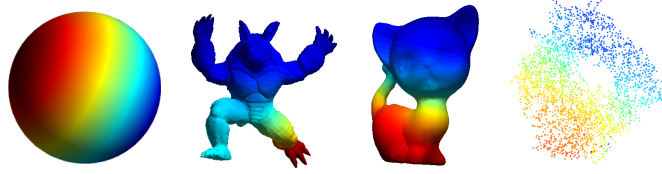


Fig. 7.7 The first nontrivial LB eigenfunctions are color-coded on the manifolds based on incomplete distance. From left to right: The unit sphere with $D_{50\%, 30}$, armadillo with $D_{50\%, 30}$, kitten with $D_{50\%, 30}$ and a protein data with $D_{3\%, \infty}$ [Lai and Li (2018)].

Laplace–Beltrami eigenvalue problem The reconstructed local coordinates provide the necessary structure to assemble the discrete Laplace–Beltrami operator and associated mass matrix. The resulting generalized eigenvalue problem,

$$\mathbb{S}v = \lambda \mathbb{M}v,$$

is used to approximate the intrinsic spectrum of the manifold directly from incomplete distance data. Numerical experiments examine the convergence

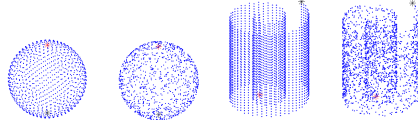
of the computed eigenvalues and eigenfunctions with respect to the sampling parameters (γ, K) , as well as the method's robustness under Gaussian noise perturbations. Additional tests extend the formulation to manifolds of higher intrinsic or ambient dimension, confirming that accurate spectral approximations can be achieved even from a small fraction of pairwise distances.

Table 7.1 Comparison of computational time (minutes) of the LB eigenvalue based on local/global reconstruction methods.

number of points				
1002	1962	4002	7842	16002
$\gamma = 100\%, K = 6$				
0.26	0.51	1.01	2.03	4.05
$\gamma = 80\%, K = 9$				
2.28	5.60	11.17	22.28	45.02
$\gamma = 50\%, K = 18$				
4.03	8.09	16.14	32.44	64.71
$\gamma = 30\%, K = 30$				
15.13	30.19	60.42	120.95	241.63
global reconstruction using 3% distance ($K = 6$ for MLS)				
2.09	9.86	40.13	154.40	597.06

The local coordinate reconstruction is performed independently for each neighborhood, leading to significant advantages in both computational efficiency and memory usage. In contrast to global reconstruction approaches that recover all coordinates before approximating differential operators, the local strategy reconstructs only the K nearest neighbors of each point. The most expensive step of the augmented Lagrangian solver is the eigenvalue hard-thresholding operation. For an incomplete distance matrix of size n , the cost of global reconstruction scales as $\mathcal{O}(n^2m)$ when computing the largest m eigenvalues. The local strategy reduces this to $\mathcal{O}(nK^2m)$, since only K -neighborhoods are processed. When n is large and $K \leq 30$, the method achieves a speedup on the order of n/K and naturally supports parallel execution across neighborhoods.

The memory requirement is also reduced from $\mathcal{O}(n^2)$ for the dense global matrix to $\mathcal{O}(nK^2)$ for local reconstruction, avoiding the prohibitive storage demands of global completion. Empirical results in Table 7.1 confirm that the local approach substantially accelerates PDE computation. For example, in computing the first 100 Laplace–Beltrami eigenvalues with $n = 16002$ and $K = 6$, the local reconstruction is approximately 150 times faster than the global method.



		1002	1962	4002	7842	16002
Uniform sampling on S^2						
DK	E_{av}	0.0348	0.0285	0.0248	0.0232	0.0224
	E_{se}	0.00862	0.00861	0.00830	0.01064	0.01150
FMD	E_{av}	0.0113	0.0095	0.0080	0.0064	0.0022
	E_{se}	0.00810	0.00589	0.00411	0.00288	0.00216
Non-Uniform sampling on S^2						
DK	E_{av}	0.0363	0.0344	0.0319	0.0305	0.0294
	E_{se}	0.01121	0.01609	0.01838	0.01639	0.01995
FMD	E_{av}	0.0200	0.0163	0.0141	0.0124	0.0088
	E_{se}	0.01202	0.00879	0.00374	0.00174	0.00277
Uniform sampling on a Swiss roll						
DK	E_{av}	0.0119	0.0156	0.0198	0.0200	0.0203
	E_{se}	0.01310	0.02124	0.02456	0.02431	0.02600
FMD	E_{av}	0.0065	0.0044	0.0052	0.0033	0.0022
	E_{se}	0.00313	0.00164	0.00113	0.00078	0.00062
Non-Uniform sampling on a Swiss roll						
DK	E_{av}	0.0400	0.0105	0.0114	0.0151	0.0180
	E_{se}	0.01661	0.01578	0.01457	0.01659	0.01865
FMD	E_{av}	0.0138	0.0053	0.0047	0.0029	0.0073
	E_{se}	0.00475	0.00519	0.00309	0.00517	0.00725

Fig. 7.8 Left to right images: Incomplete distance data $D_{60\%,20}$ from 1002 points sampled uniformly and non-uniformly on the unit sphere with fixed north pole (red star) and south pole (blue star) for calculating geodesic distance, and $D_{60\%,20}$ from 1962 points sampled uniformly and non-uniformly on the swiss roll with a fixed starting point (red star) and an ending point (blue star) for calculating geodesic distance. Table: Error comparison between Dijkstra (DK) and Fast Marching on distance data (FMD). Relative error of geodesic distances E_{av} (averaging from starting to all the points) and E_{se} (from the starting point to the ending point).

Eikonal equation and geodesic accuracy. The framework further extends to nondiffusive PDEs, most notably the Eikonal equation. By employing the local mesh method to reconstruct local coordinates from pairwise distances—referred to as fast marching on distance data (FMD) [Lai and Li (2018)]—the Eikonal equation can be solved directly on distance-sampled manifolds. We demonstrate its numerical accuracy in comparison with the classical Dijkstra algorithm. Figure 7.8 presents the relative errors of the computed geodesic distances from a fixed source to a target point and to all other points on both the unit sphere and the Swiss roll mani-

fold, using incomplete distance data $D_{60\%,20}$. The results demonstrate that FMD yields substantially smaller errors than Dijkstra's method [Dijkstra (1959)]. Moreover, the relative errors decrease as the number of sampled points increases. For uniformly sampled data, the geodesic distance from the north pole to the south pole exhibits first-order convergence with respect to sampling density under the fast marching discretization, whereas Dijkstra's method does not exhibit comparable convergence behavior.