Computation of Curvature of Cayley Graphs

Anthony Hong

Department of Mathematics and Statistics
Washington University in St. Louis

December 31, 2023

Contents

Current Section

Combinatorial Curvature

Curvature is a smooth concept, so the principle of defining discrete version is to find common feature between two settings.

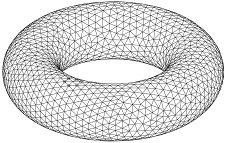

Figure 1: triangulation of a torus; taken from Wikipedia "Triangulation (topology)"

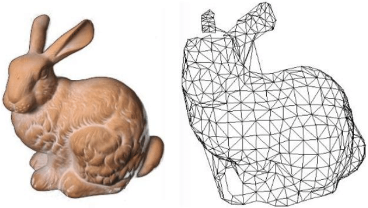

Figure 2: The Stanford Bunny. [1]

1. Common feature: Euler characteristic χ; define curvature by Gauss-Bonnet theorem.
2. Common feature: Laplace operator $\mathcal{L}$; define curvature by Bochner-Weitzenböck formula.

Ollivier-Ricci Curvature

1. Bakry and Émery on heat flow-type equations
2. Sturm [2] and Lott and Villani [3]'s generalization of Ricci curvature on metric measure spaces with W_2 distance.
3. Ollivier [4] uses W_1 distance. His definition is “very easy to implement on concrete examples."
4. Lin, Lu, and Yau obtained a modified Ollivier-Ricci curvature for graphs [5].
5. For Cayley graphs in particular, there are also variants like conjugation curvature.

Gromov's δ-Hyperbolicity

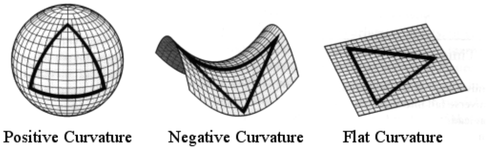

Figure 3: Triangles of three characteristic surfaces of different curvatures.[1]

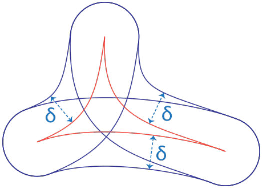

Figure 4: δ-thin triangle illustration[2]

[1] Figure taken from Harrison Hartle's blog.
[2] Figure taken from Wikimedia Commons.

Wasserstein Distance

Definition (Coupling)

A **coupling** of two probability measures μ and ν on the measurable spaces $(X, \mathcal{X})$ and $(Y, \mathcal{Y})$ respectively is any probability measure π on the product measurable space $(X \times Y, \mathcal{X} \otimes \mathcal{Y})$ whose marginals are μ and ν:

$$\pi(A \times Y) = \mu(A), \quad \pi(X \times B) = \nu(B)$$

Definition (Wasserstein Distance)

Let (X, d) be a metric space and μ, ν be two probability measures on X. The **Wasserstein distance** between μ and ν is defined as

$$W(\mu, \nu) := \inf_{\pi \in \Pi(\mu,\nu)} \int_{X \times X} d(x, y)\, d\pi(x, y).$$

where $\Pi(\mu, \nu)$ is the set of all the couplings of μ and ν. Finding such a minimizer is called the **Kantorovich problem**.

Wasserstein Distance on Graph

Let $G = (V, E)$ be a simple and connected graph and with a countable vertex set V.

1. It can be realized as a **metric space** (V, d) where d is the number of edges of the shortest path connecting two vertices.
2. Probability measure μ on $(V, \wp(V) = \mathcal{V})$ are determined by their value on each vertex $\mu(v)$, so a **vector** $(\mu(v))_{v\in V}$ represents a measure.
3. The coupling $\pi \in \Pi(\mu, \nu)$ will be called a **transport plan**. It's a map $\pi : V \times V \to [0, 1]$ satisfying marginality conditions

$$v \in V : \mu(v) = \sum_{w\in V} \pi(v, w) \quad \text{and} \quad w \in V : \nu(w) = \sum_{v\in V} \pi(v, w)$$

   and is determined by its value on each $(x, y) \in V \times V$. So it is a **matrix**.
4. The **(total) cost function** of π is

$$\text{cost}(\pi) = \sum_{v,w\in V} d(v, w)\pi(v, w)$$

   The **optimal transport** minimizes total cost and gives **Wasserstein distance** between measures μ, ν.

Markov Chain

One way to think about Markov Chain is to consider a **Markov kernel** K associating each point x of the space X a probability measure K_x on measurable space $(X, \mathcal{X})$.
A naive example:

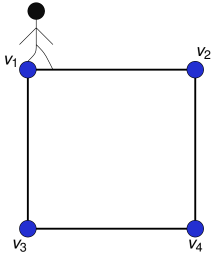

Figure 5: A random walker in a very small town

For each vertex v_i, the random walker has probabilities of moving to one of the four vertices. We thus have a kernel K. In this discrete case, it is also the same as the transition matrix. In case of Riemannian manifold realized as a geodesic metric space, things are much more intricate.

Current Section

Ollivier-Ricci Curvature

Definition (Ollivier-Ricci Curvature)

Let (X, d) be a metric space with a random walk $m = (m_x(\cdot))_{x \in X}$. Let $x, y \in X$ be two distinct points. The **coarse Ricci curvature** of (X, d, m) along (xy) is

$$\kappa(x, y) = 1 - \frac{W(m_x, m_y)}{d(x, y)}$$

It approximates to Ricci curvature of Riemannian manifold (M^n, d_g, vol) by random walk

$$m_x^\varepsilon(dy) = \begin{cases} \frac{1}{\mathrm{vol}(B_\varepsilon(x))}\mathrm{vol}(dy), & \text{if } y \in B_\varepsilon(x), \\ 0, & \text{if } y \notin B_\varepsilon(x). \end{cases}$$

or

$$m_x^\varepsilon(A) = \frac{\mathrm{vol}(B_\varepsilon(x) \cap A)}{\mathrm{vol}(B_\varepsilon(x))}, \quad A \in \mathcal{B}(M)$$

Ollivier-Ricci Curvature

Ollivier in [4] shows that for a unit tangent vector v at point p and a point y on the maximal geodesic γ_v, one has

$$\kappa(x,y) = \frac{\varepsilon^2 Rc(v,v)}{2(n+2)} + O\left(\varepsilon^2 + \varepsilon^2 d_g(x,y)\right)$$

with suffciently small $d_g(x,y)$.

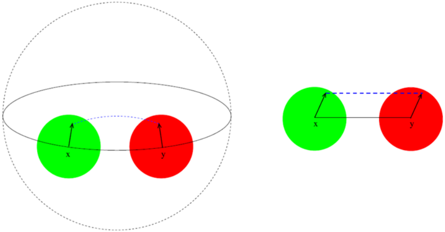

FIGURE 8. In the 2-sphere, corresponding points in small metric balls $B_\epsilon(x), B_\epsilon(y)$ in parallel directions have smaller distance than $d(x,y)$. In the Euclidean plane, they have the same distance $d(x,y)$.

Lin-Lu-Yau Curvature

In [5], Lin, Lu, and Yau similarly used a neighborhood-supported uniform jumping to defined curvature on graph $G = (V, E)$:

$$v \in V : \; m_x^\alpha(v) = \begin{cases} \alpha & \text{if } v = x, \\ \frac{1-\alpha}{\deg_x} & \text{if } v \in \Gamma(x), \\ 0 & \text{otherwise.} \end{cases}$$

where $\Gamma(x)$ is the neighborhood of x and degree deg_x is the cardinality of $\Gamma(x)$. We assume the graph is locally finite. Then we have the following notion of curvature.

Lin-Lu-Yau Curvature

Definition (Lin-Lu-Yau curvature)

For any $x, y \in V$, the α**-curvature** κ_α is given by

$$\kappa_\alpha(x, y) = 1 - \frac{W(m_x^\alpha, m_y^\alpha)}{d(x, y)}$$

where W is the Wasserstein distance in defn 1.8. Due to [5] lemma 2.1 on concavity of $\kappa_\alpha(x, y)$ with respect to α and fixed x, y , we arrive at the Lin-Lu-Yau curvature

$$\kappa(x, y) = \lim_{\alpha \to 1} \frac{\kappa_\alpha(x, y)}{1 - \alpha}.$$

We look at an example: consider a random walk on the complete graph K_5.

Lin-Lu-Yau Curvature

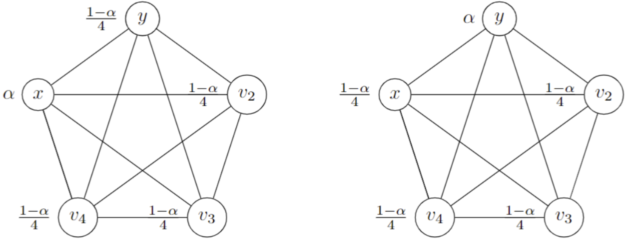

Figure 6: Random walk on a complete graph

$$\pi(x,x) = \frac{1-\alpha}{4}, \pi(x,y) = \frac{5\alpha-1}{4}, \pi(x, \text{any other vertex}) = 0$$

$$\pi(y,y) = \frac{1-\alpha}{4}, \pi(y, \text{any v}) = 0, \pi(v_i, v_i) = \frac{1-\alpha}{4}, \pi(v_i, \text{any other v}) = 0$$

Linear Optimization

In fact, Kantorovich problem on graph is a linear optimization problem. Let $\mathbf{P}_{ij} = \pi(v_i, v_j)$ and

$$\mathbf{u} = (m_x(v_i))_{v_i \in B(x)}, \mathbf{v} = (m_y(v_j))_{v_j \in B(y)}$$

Then the set of all coupllings $\Pi(\mu, \nu)$ becomes

$$\Pi(m_x, m_y) = \Pi(\mathbf{u}, \mathbf{v}) = \left\{ \mathbf{P} \in M_{k \times l}(\mathbb{R}) : \mathbf{P}\mathbb{1}_l = \mathbf{u}, \mathbf{P}^T \mathbb{1}_k = \mathbf{v} \right\}$$

Let $\mathbf{D}_{ij} = d(v_i, v_j)$. Then, Kantorovich problem becomes

$$W(\mathbf{u}, \mathbf{v}) = \min_{\mathbf{P} \in \Pi(\mathbf{u}, \mathbf{v})} \langle \mathbf{D}, \mathbf{P} \rangle = \min_{\mathbf{P} \in \Pi(\mathbf{u}, \mathbf{v})} \sum_{i,j} \mathbf{D}_{ij} \mathbf{P}_{ij} \tag{1}$$

Network Flow Problem

If we concatenate the rows of **P** and transpose it to get vector **f**, and do the same to **D** to get **c**, and define **b** by values of two measures, we can get a network flow problem (NFP).

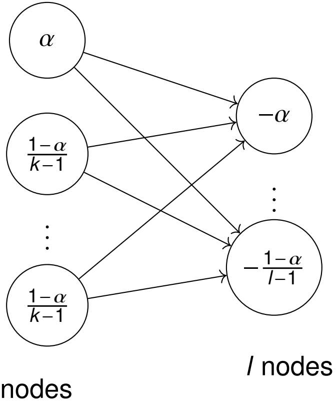

Figure 7: Bipartite-graphical representation of Kantorovich problem

Network Flow Problem

In particular,

$$b_i = \begin{cases} m_x^\alpha(v_i), & 1 \leqslant i \leqslant k \\ -m_y^\alpha(v_i), & k+1 \leqslant i \leqslant k+l \end{cases}$$

where $|B(x)| = k, |B(y)| = l$ and we reorder the vertices so that $v_1 = x, v_{k+1} = y$.

$$\mathbf{f} = [f_{1,k+1}, \cdots, f_{1,k+l}; f_{2,k+1}, \cdots, f_{2,k+l}; \cdots; f_{k,k+1}, \cdots, f_{k,k+l}]^T.$$

$$\mathbf{c} = [d(v_1, v_{k+1}), \cdots, d(v_1, v_{k+l}); d(v_2, v_{k+1}), \cdots, d(v_2, v_{k+l}); \\ \cdots; d(v_k, v_{k+1}), \cdots, d(v_k, v_{k+l})]^T$$

and consider incidence matrix

Network Flow Problem

$$
\mathbf{A} = \begin{array}{c} \\ v_1 \\ v_2 \\ \vdots \\ v_k \\ v_{k+1} \\ v_{k+2} \\ \vdots \\ v_{k+l} \end{array}
\begin{array}{c}
\begin{array}{cccccccc} (1,k+1) & (1,k+2) & \cdots & (1,k+l); & \cdots & ;(k,k+1) & \cdots & (k,k+l) \end{array} \\
\left[\begin{array}{cccccccc}
1 & 1 & \cdots & 1 & \cdots & 0 & \cdots & 0 \\
0 & 0 & \cdots & 0 & \cdots & 0 & \cdots & 0 \\
\vdots & \vdots & \ddots & \vdots & \ddots & \vdots & \ddots & \vdots \\
0 & 0 & \cdots & 0 & \cdots & 1 & \cdots & 1 \\
-1 & 0 & \cdots & 0 & \cdots & -1 & \cdots & 0 \\
0 & -1 & \cdots & 0 & \cdots & 0 & \cdots & 0 \\
\vdots & \vdots & \ddots & \vdots & \ddots & \vdots & \ddots & \vdots \\
0 & 0 & \cdots & -1 & \cdots & 0 & \cdots & -1
\end{array}\right]
\end{array}
$$

$$
= \begin{array}{c} \\ v_1 \cdots v_k \\ v_{k+1} \cdots v_{k+l} \end{array}
\begin{array}{c}
\begin{array}{cccc} (1,k+1)\cdots(1,k+l) & (2,k+1)\cdots(2,k+l) & \cdots & (k,k+1)\cdots(k,k+l) \end{array} \\
\left[\begin{array}{cccc}
\mathbf{1}_1 & \mathbf{1}_2 & \cdots & \mathbf{1}_k \\
-\mathbf{I}_l & -\mathbf{I}_l & \cdots & -\mathbf{I}_l
\end{array}\right]
\end{array}
$$

Then we get the standard form NFP and its dual

$$
\begin{aligned}
\min \quad & \mathbf{c}^T\mathbf{f} \\
\text{subject to} \quad & \mathbf{A}\mathbf{f} = \mathbf{b} \\
& \mathbf{f} \geq \mathbf{0}
\end{aligned}
$$

$$
\begin{aligned}
\max \quad & \mathbf{p}^T\mathbf{b} \\
\text{subject to} \quad & \mathbf{p}^T\mathbf{A} \leq \mathbf{c}^T
\end{aligned}
$$

Network Flow Problem

1. simplex method and dual simplex method (“linprog" of “scipy.optimize" library)
2. “ot.emd2” imported from “ot" library (see POT: Python Optimal Transport)

```
#Solve Kantorovich problem of complete graph with n=5 and alpha=1/4
c = [1, 0, 1, 1, 1, 1, 1, 0, 1, 1, 1, 1, 1, 0, 1, 1, 1, 1, 1, 0, 0, 1, 1, 1, 1]
b = [1/4, 3/16, 3/16, 3/16, 3/16, -1/4, -3/16, -3/16, -3/16, -3/16]

# primal
result_primal = linprog(c, A_eq=matrix_A(5,5), b_eq=b, method="simplex")
# output:
# optimal_value = result_primal.fun = 0.0625
# optimal_solution = [0.0625 0.1875 0. 0. 0. 0. 0. 0.1875 0. 0. 0. 0. 0. 0.1875 0.
# 0. 0. 0. 0. 0.1875 0.1875 0. 0. 0. 0.]

#dual
neg_b = [-x for x in b]
result_dual = linprog(neg_b, A_ub=matrix_A(5,5).T, b_ub=c, method="simplex")
# output:
# optimal_value = result_primal.fun = -0.0625 (note that max(f)=-min(-f))
# optimal_solution = [1. 0. 0. 0. 0. 0. 1. 0. 0. 0.]

# ot library
a = [1/4, 3/16, 3/16, 3/16, 3/16] # distribution m^0.25_x
b = [1/4, 3/16, 3/16, 3/16, 3/16] # distribution m^0.25_y
M = [
[1, 0, 1, 1, 1],
[1, 1, 0, 1, 1],
[1, 1, 1, 0, 1],
[1, 1, 1, 1, 0],
[0, 1, 1, 1, 1]
]

Wd = ot.emd2(a, b, M)
# output: 0.0625
```

Current Section

Examples of Groups and Their Cayley Graphs

definition (Cayley graph)

Let $\mathcal{G}$ be a group with a finite set of generators S with the following two properties: (1) identity is not in S (i.e. $e \notin S$); (2) (symmetric property) element s is in S if and only if the inverse s^{-1} is in S (denoted as $S = S^{-1}$). Then **Cayley graph** $\mathrm{Cay}(\mathcal{G}; S)$ with **connection set** S is a graph with V and E defined as follows:

- the vetex set V contains exactly all elements of $\mathcal{G}$;
- For $x, y \in V$, $(xy) \in E$ (or $x \sim y$) if and only if $\exists s \in S$ s.t. $y = xs$.

We may use summation $+$ to denote group operation when $\mathcal{G}$ is abelian. ♦

Example: Let the group be $\mathcal{G} = (\mathbb{Z}^n, +)$ and set connection set S as

$$S = \left\{ (x_1, x_2, \cdots, x_n) \in \mathbb{Z}^n \middle| \sum_{i=1}^{n} |x_i| = 1 \right\},$$

that is, the set of points with exactly one nonzero component ± 1. In case $n = 1$, the set S degenerates to $\{-1, 1\}$. Thus, $x \sim y$ if $x - y = 1$ or $x - y = -1$.

Examples of Groups and Their Cayley Graphs

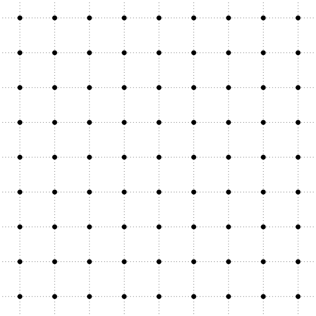

Figure 8: graph $\mathbb{Z}^2$

When $n = 2$, $S = \{(0,-1),(0,1),(-1,0),(1,0)\}$. We observe that

1. $W\left(m^{\alpha}_{(0,0)}, m^{\alpha}_{(\pm 2,\pm 2)}\right)$. Dihedral group D_4?
2. $W\left(m^{\alpha}_{(0,0)}, m^{\alpha}_{(2,3)}\right) = W\left(m^{\alpha}_{(5,5)}, m^{\alpha}_{(7,8)}\right)$.homomeneity of the Cayley graph?

Algorithmic Improvment on Finding Curvatures of Cayley Graphs

Main idea

For more complicated graph, need computational programs to assist theoretical observation. Thus, our goals are (1) compute optimal transport correctly and efficiently by programs; (2) detect patterns from results.

Let $|S| = k$. Each vertex x in the Cayley graph then has k edges connecting to other vertices. Then the measure on x becomes

$$v \in V : \; m_x^\alpha(v) = \begin{cases} \alpha & \text{if } v = x, \\ \frac{1-\alpha}{k} & \text{if } v \in \Gamma(x), \\ 0 & \text{otherwise.} \end{cases}$$

Then the Wasserstein distances between m_x^α and m_y^α are determined by the $(k+1) \times (k+1)$ cost matrix D. By revising Proposition 2.1 of [6], we see the assignment problem (Monge problem) attains the minimum of the Kantorovich problem on k-regular graph, that is, $k = l = |B(x)| = |B(y)|$ for each x and y. One of the best solvers is Kuhn-Munkres algorithm.

Algorithmic Improvment on Finding Curvatures of Cayley Graphs

Table 5: The Ricci curvature of the Cayley graph of $\mathbb{Z}/n\mathbb{Z}$ $(6 \leq n \leq 15)$ generated by $S_{1,3}$.

n	6	7	8	9	10	11	12	13	14	15
Ricci curvature (type A)	$\frac{2}{3}$	$\frac{3}{4}$	$\frac{1}{2}$	$\frac{1}{2}$	$\frac{1}{2}$	$\frac{1}{2}$	$\frac{1}{2}$	$\frac{1}{2}$	$\frac{1}{2}$	$\frac{1}{2}$
Ricci curvature (type B)	$\frac{2}{3}$	1	$\frac{1}{2}$	$\frac{3}{4}$	$\frac{1}{2}$	$\frac{3}{4}$	$\frac{1}{2}$	$\frac{1}{4}$	0	$\frac{1}{4}$

The Ricci curvatures of the Cayley graphs $\Gamma(\mathbb{Z}/n\mathbb{Z}, S_{1,4})$ of generating set $S_{1,4} = \{+1, +4, -1, -4\}$ $(6 \leq n \leq 22)$ are given in Table 7.

Table 7: The Ricci curvature of the Cayley graph of $\mathbb{Z}/n\mathbb{Z}$ generated by $S_{1,4}$

n	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
Ricci curvature (type A)	$\frac{2}{3}$	$\frac{3}{4}$	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$
Ricci curvature (type B)	$\frac{2}{3}$	1	$\frac{1}{4}$	$\frac{3}{4}$	$\frac{1}{4}$	$\frac{1}{2}$	$\frac{3}{4}$	$\frac{1}{2}$	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{2}$	$\frac{1}{4}$	$\frac{1}{4}$	0	$\frac{1}{4}$	0	$\frac{1}{4}$

Figure 9: α-curvature of cyclic group with connection sets $S_{1,3}$, $S_{1,4}$; [7]

References

[1] Christakis Kallis, Kyriakos Deliparaschos, George Moustris, Avraam Georgou, and Themistoklis Charalambous. Incremental 2d delaunay triangulation core implementation on fpga for surface reconstruction via high-level synthesis. 09 2017.

[2] K.-T Sturm. On the geometry of metric measure spaces i and ii. *Acta Math.*, 196:65–131,133–177, 2006.

[3] John Lott and Cedric Villani. Ricci curvature for metric-measure spaces via optimal transport. *Annals of Mathematics*, pages 903–991, 2009.

[4] Yann Ollivier. Ricci curvature of markov chains on metric spaces. *Journal of Functional Analysis*, 256:810–864, 2009.

[5] Yong Lin, Linyuan Lu, and Shing-Tung Yau. Ricci curvature on graphs. *Tohoku Math. J.*, 63(1):605–627, 2010.

[6] Gabriel Peyre and Marco Cuturi. Computational optimal transport. *arXiv preprint arXiv:1803.00567v4*, 2020.

[7] Iwao Mizukai and Akifumi Sako. Ollivier ricci curvature of cayley graphs for dihedral groups, generalized quaternion groups, and cyclic groups. *arXiv preprint*, 2022. arXiv:2210.00860.

References

[8] Cedric Villani. *Topics in Optimal Transportation*. American Mathematical Society, 2003.

[9] Luigi Ambrosio, Nicola Gigli, and Giuseppe Savare. *Gradient Flows: In Metric Spaces and in the Space of Probability Measures*. Springer Science+Business Media, 2005.

Thank You