

Discrete Ricci Flow applied to Complex Networks

Ollivier-Ricci Curvature and Its Relevance to Community Detection

**Lorenzo Fabbri
Giancarlo Oancia**

This essay is submitted for the exam of
Complex Networks



Department of Physics
Theoretical Physics Curriculum
University of Bologna
01/03/2025

Draft - v1.0

February 27, 2025 – 08:51

ABSTRACT

This project uses Ricci Flow as a geometric approach to detect community structures in networks. It applies Ollivier-Ricci curvature to adjust the weights of edges in a graph, iterating the process to shrink intra-community edges and stretch inter-community edges. Having tested the Ricci Flow method, our goal was identifying the pre-labelled communities of a real dataset: Zachary's Karate Club graph [15].

Draft - v1.0

February 27, 2025 – 08:51

CONTENTS

1	Overview of the Project	7
2	Introduction and Motivation	9
2.1	Poincaré Conjecture	9
2.2	Basics of Differential Geometry	12
2.2.1	Differentiable Manifolds	13
2.2.2	Differentiable Maps	14
2.2.3	Vectors	16
2.2.4	One-Forms	18
2.2.5	Tensors and Tensor Fields	19
2.2.6	Flow generated by a vector field	20
2.2.7	Riemannian Manifolds	21
2.2.8	Curvature and Ricci tensor	23
2.2.9	Covariant Derivative	23
2.3	Non-Linear Sigma Models and String Theory	23
2.4	Ricci Flow to Tackle Topological Problems	25
3	Ricci Flow and Oliver-Ricci Curvature	27
3.1	Riemannian Geometry and Curvature Notions	27
3.1.1	Hamilton's Ricci Flow Equation	28
3.1.2	From Smooth Settings to Discrete Geometry	28
3.2	Optimal Transport and Ollivier's Ricci Curvature	29
4	Application to Complex Networks	31
4.1	General Application of Ricci Flow to Complex Networks: ARI, Modularity, and Performance	32

6 | Contents

4.1.1	Adjusted Rand Index (ARI)	33
4.1.2	Modularity	34
4.1.3	Interpreting Network “Surgery” in This Context	36
4.1.4	Algorithmic Complexity and Practical Considerations	36
4.1.5	Empirical Performance and Use Cases	37
4.1.6	Choosing Parameters: Number of Iterations, Threshold, and Measures	37
4.1.7	Broader Theoretical Context and Future Directions	38
4.1.8	Concluding Remarks on Sections ?? and 4.1	38
4.2	Discrete Ricci Flow	39
5	The Developed Code	41
5.1	The setup	42
5.2	Tests on Synthetic Graphs	42
5.2.1	Stochastic Block Model Test Graph	42
5.2.2	Lancichinetti-Fortunato-Radicchi Test Graph	44
5.2.3	Further Tests on Lancichinetti-Fortunato-Radicchi Graphs	44
5.3	Application to Zachary’s Karate Club Graph	44
6	Conclusions and Future Directions	51
Appendix A	Metric of a 2-sphere	53
References		55

CHAPTER 1

OVERVIEW OF THE PROJECT

The study of networks has gained considerable attention in various fields, ranging from sociology to biology, and beyond. One of the central problems in network science is community detection, where the objective is to identify groups of nodes (communities) that are more densely connected internally than with the rest of the network. Traditional methods for community detection often rely on statistical or combinatorial approaches. However, recent developments in geometric methods have introduced new ways to approach this problem by leveraging concepts from differential geometry [7].

A powerful geometric tool, the Ricci Flow, originally developed in the context of smooth Riemannian manifolds, can be adapted to discrete network structures. In its original formulation, the Ricci Flow evolves the metric of a manifold according to the curvature (represented by Ricci tensor), leading to a smoothing process over time. Ollivier-Ricci curvature, a discretization of Ricci curvature for graphs, provides a framework to extend this idea to networks, where the "curvature" of edges encodes structural information about node connectivity. Specifically, positive curvature tends to shrink intra-community edges, while negative curvature expands inter-community edges [7].

In this project, we apply Ollivier-Ricci curvature and Ricci Flow to detect the two known communities in Zachary's Karate Club graph. The approach follows the work of Ni et al. [7], where Ricci Flow is used to reshape edge weights iteratively, enhancing the separation between different communities. After applying Ricci Flow, we perform edge surgery to remove weakly connected edges and extract communities as the connected components of the resulting graph.

8 | Overview of the Project

1 The results will be compared to the predefined community labels, allowing for
2 a direct comparison between the detected communities and the actual community
3 structures.

4 The developed code can be accessed in the corresponding GitHub repository:
5 [RicciFlowNetwork](#). Code documentation is accessible at [CodeDocumentation](#).

6 Diciamo che la prima parte del report è introduttiva e riguarda le stringhe ecc.?

CHAPTER 2

INTRODUCTION AND MOTIVATION

2.1 Poincaré Conjecture

To understand the origins of Ricci Flow and its relevance in the mathematical context, one has to get an insight on its application on geometrical and topological problems. Therefore, to begin with the discussion, we'll introduce the Poincaré conjecture and Perelman's proof.

In the 1985 paper *Analysis Situs* [9], Poincaré laid the foundations for what we call today as *topology*. Its purpose is clearly underlined in the articles, through:

...geometry is the art of reasoning well from badly drawn figures; however, these figures, if they are not to deceive us, must satisfy certain conditions; the proportions may be grossly altered, but the relative positions of the different parts must not be upset.

Since there's no reason to follow the historical development, we'll furnish the modern definitions and intuition about topology.

Definition 1 (Topological space). Let X be a non-empty set. A *topology* on X is a family \mathcal{A} of subsets $\mathcal{A} \ni A \subseteq X$, called *open sets*, such that

- The empty set and X are open sets,

$$\emptyset \in \mathcal{A}, \quad X \in \mathcal{A}. \quad (2.1)$$

- The union of any collection of open sets is again an open set,

$$A_i \in \mathcal{A}, i \in J \implies \bigcup_{i \in J} A_i \in \mathcal{A}, \quad (2.2)$$

with J a set of indices.

10 | Introduction and Motivation

- 1 • The intersection of any finite number of open sets is again an open set,

2
$$A_i \in \mathcal{A}, i = 1, 2, \dots, N \implies \bigcap_{i=1}^N A_i \in \mathcal{A}. \quad (2.3)$$

3 The set X , with the topology \mathcal{A} , is called *topological space* (X, \mathcal{A}) .

4 We can then define straightforwardly what a closed set is. Notice that we could
5 have defined the topology with closed sets as well.

6 **Definition 2** (Closed sets). Given $A \in \mathcal{A}$, a subset $C \subseteq X$ is *closed* if is of the form

7
$$C \text{ closed } \iff X \setminus C \text{ open.} \quad (2.4)$$

8 There are some crucial properties of topological spaces which will be extensively
9 used throughout this works. The first, concerns compactness, which is a subtle concept
10 to identify finiteness in this context.

11 **Definition 3** (Compact). A topological space (X, \mathcal{A}) is called *compact* if every open
12 cover,

13
$$X \in \bigcup_{\alpha \in C} U_\alpha, \quad (2.5)$$

14 contains a finite subcover,

15
$$X \in \bigcup_{i=1}^N U_i. \quad (2.6)$$

16 Further, another characterizing property of some topological spaces is the con-
17 nectedness. The intuition is that a connect space can't be cut up into parts that have
18 nothing to do with each other. To make this sentence more precise, we need first to
19 define a path on a topological space.

20 **Definition 4** (Path). Given a topological space (X, \mathcal{A}) and two points $x, y \in X$, a *path* is
21 a continuous function $f : [0, 1] \rightarrow X$ such that $f(0) = x$ and $f(1) = y$. A *path-component*
22 of X is an equivalence class of X under the equivalence relations which makes x
23 equivalent to y if and only if there is a path from x to y

24 **Definition 5** (Connected). A topological space (X, \mathcal{A}) is said to be *connected* if it can't
25 be represented as the union of two or more disjoint non-empty open subsets.

26 **Definition 6** (Path connected). A topological space (X, \mathcal{A}) is said to be *path connected*
27 if there is exactly one path-component. For non-empty spaces, this is equivalent to the
28 statement that there is a path joining any two points in X .

Definition 7 (Simply connected). A topological space (X, \mathcal{A}) is said to be *simply connected* if it is path connected and every path between two points can be continuously transformed into any other such path while preserving the two endpoints in question.

Other than these properties, we need maps from a topological space to another, both for relating these abstract concepts to the Euclidean space, for which we have intuition, and for studying the algebraic properties of the structures. This gives rise to the necessity of defining a homeomorphism.

Definition 8 (Homeomorphism). A *homeomorphism*, also known as *topological isomorphism*, is a bijective and continuous between topological spaces that has a continuous inverse function.

We should have defined what a continuous map is, but the definition is not too different from the one used in multivariate calculus.

In particular, we're interested in applying those properties to some particular types of topological spaces, that is, topological manifolds. To apply the Ricci Flow to a topological context, the manifold must be further equipped with a wider structure, i.e., must be a Riemannian Manifold. While this topic will be covered in the next section, for now let's just define a Manifold.

Definition 9 (Manifold). A *topological manifold* is a topological space which locally resembles the real n -dimensional Euclidean space.

Without delving into the details of the Manifolds with boundaries, we make use of our intuition to understand this concept, and directly define a closed manifold.

Definition 10 (Closed manifold). A *closed manifold* is a manifold without boundary that is also compact.

A simple example is provided by the circle, for one-dimensional manifolds, and the sphere for two-dimensional ones. We're interested in the 3-sphere, which is a straightforward generalization of the previous examples.

Definition 11 (3-sphere). In coordinates¹, a 3-sphere with center (C_0, C_1, C_2, C_3) and radius r is the set of all points (x_0, x_1, x_2, x_3) in \mathbb{R}^4 such that

$$\sum_{i=0}^3 (x_i - C_i)^2 = (x_0 - C_0)^2 + (x_1 - C_1)^2 + (x_2 - C_2)^2 + (x_3 - C_3)^2 = r^2. \quad (2.7)$$

¹We'll define a chart on a manifold later on, for now let's use our intuitive understanding of a set of coordinates in \mathbb{R}^n .

12 | Introduction and Motivation

¹ In particular, the 3-sphere centred at the origin with radius 1 is called the *unit*
² 3-sphere and is usually denoted ad S^3

$$\text{³} S^3 = \{(x_0, x_1, x_2, x_3) \in \mathbb{R}^4 : x_0^2 + x_1^2 + x_2^2 + x_3^2 = 1\}. \quad (2.8)$$

⁴ As a Manifold, the 3-sphere is a compact, connected, 3-dimensional Manifold
⁵ without boundary. We're now able to understand the formulation of the Poincaré
⁶ conjecture in its work:

⁷ *A simply-connected closed manifold is homeomorphic to a sphere.*

⁸ To be fair, at the beginning it didn't think of it as a conjecture, since it considered
⁹ it a trivial statement. It turned out to be one of the biggest unresolved problems in
¹⁰ mathematics. After some refinements during the following years [cite](#), he came up to
¹¹ the final form of its conjecture [10]:

¹² *Every three-dimensional topological manifold which is closed, connected, and has
¹³ trivial fundamental group is homeomorphic to the three-dimensional sphere.*

¹⁴ To overcome the complexity of fundamental groups and homologies, we'll just
¹⁵ quote the following theorem, which allows us to understand the necessity of the
¹⁶ presence of the trivial fundamental group in the latter formulation of the conjecture.

¹⁷ **Theorem T.1.** *A path-connected topological space is simply connected if and only if its
¹⁸ fundamental group is trivial.*

¹⁹ To understand how to tackle the problem of Poincaré conjecture proof, we first
²⁰ need to investigate the manifold structure of some particular topological spaces.

²¹ 2.2 Basics of Differential Geometry

²² In order to understand the Ricci Flow applications, it's necessary to study the
²³ properties of Riemannian Manifolds, in particular how to define a flow on a Manifold
²⁴ and how this is related to some differential equations. We'll see that the Ricci Flow is
²⁵ nothing but a differential equation for some class of different metrics on a Riemannian
²⁶ Manifold.

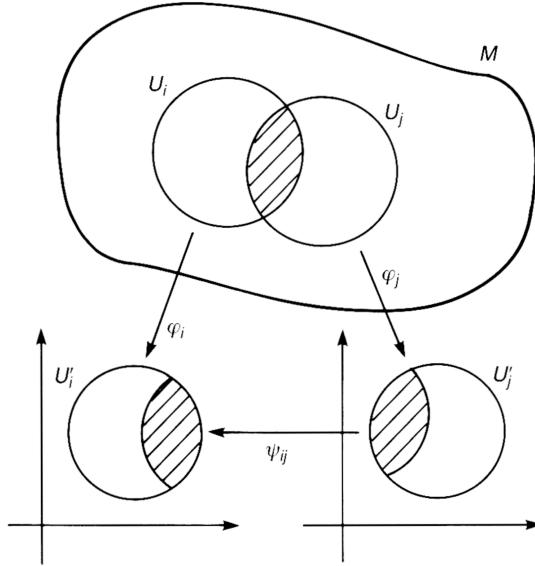


Figure 2.1 φ_i is a homeomorphism from U_i onto an open subset U'_i of \mathbb{R}^n .

2.2.1 Differentiable Manifolds

Let's first refine the definition of a differentiable Manifold 9.

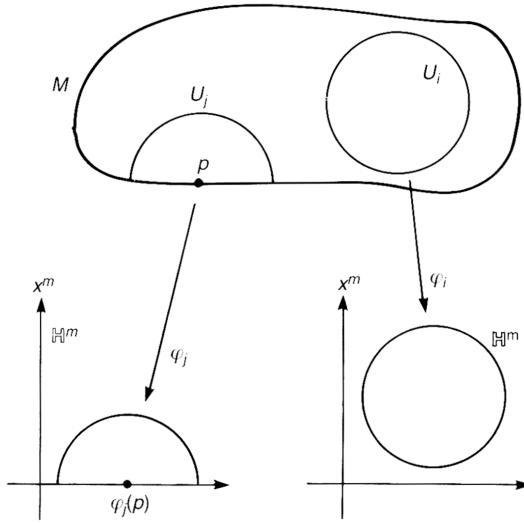
Definition 12 (Differentiable manifold). \mathcal{M} is an n -dimensional differentiable manifold if

- \mathcal{M} is a topological space.
- \mathcal{M} is provided with a family of pairs $\{(U_i, \varphi_i)\}$.
- $\{U_i\}$ is a family of open sets which covers \mathcal{M} , that is, $\cup_i U_i = \mathcal{M}$. φ_i is a homeomorphism from U_i onto open subsets U'_i of \mathbb{R}^n . See fig. 2.1
- Given U_i and U_j such that $U_i \cap U_j \neq \emptyset$, the map $\psi_{ij} = \varphi_i \circ \varphi_j^{-1}$ from $\varphi_j(U_i \cap U_j)$ to $\varphi_i(U_i \cap U_j)$ is infinitely differentiable.

The pair (U_i, φ_i) is called a *chart*, while the whole family $\{(U_i, \varphi_i)\}$ is an *atlas*. The subset U_i is called the *coordinate neighbourhood*, while φ_i is the *coordinate function*. This homeomorphism is represented by n functions $\{x_1(p), \dots, x_n(p)\}$, which are called *coordinates*. However, notice that a point $p \in \mathcal{M}$ is independent of its coordinates. In each coordinate neighbourhood U_i , \mathcal{M} looks like an open subset of \mathbb{R}_n whose element is $\{x^1, \dots, x^n\}$, and this explains our previous definition 9.

Recall from the previous section that we're interested in Manifolds with boundaries. We're now able to define them.

14 | Introduction and Motivation

Figure 2.2 A Manifold \mathcal{M} with boundary. Here, $p \in \partial\mathcal{M}$.

¹ **Definition 13** (Manifold with boundary). A *Manifold with boundary* (see fig. 2.2) is a
² topological space \mathcal{M} which is covered by a family of open sets $\{U_i\}$, each of which is
³ homeomorphic to an open set of H^n , where

⁴
$$H^n := \{(x^1, \dots, x^n) \in R^n | x^n \geq 0\}. \quad (2.9)$$

⁵ The set of points which are mapped to points with $x_n = 0$ is called the *boundary*
⁶ of \mathcal{M} , denoted by $\partial\mathcal{M}$. The coordinates of $\partial\mathcal{M}$ may be given by $n - 1$ numbers
⁷ $(x^1, \dots, x^{n-1}, 0)$. One must then be careful to define smoothness. Indeed, the map
⁸ $\psi_{ij}: \varphi_j(U_i \cap U_j) \rightarrow \varphi_i(U_i \cap U_j)$ is defined on an open set of H^n in general, and φ_{ij} is
⁹ said to be smooth if it's C^∞ in an open set of R^n which contains $\varphi_j(U_i \cap U_j)$.

¹⁰ **2.2.2 Differentiable Maps**

¹¹ Due to the presence of the differentiable structure on a Manifold, we're able to use
¹² the calculus techniques developed for R^n . We can define a map between manifolds.

¹³ Let \mathcal{M} and \mathcal{N} be an m -dimensional and an n -dimensional Manifold, respectively.
¹⁴ Let's then consider a map between them, i.e., $f: \mathcal{M} \rightarrow \mathcal{N}$, where $\mathcal{M} \ni p \mapsto f(p) \in \mathcal{N}$,
¹⁵ see fig. 2.3 Taking the charts (U, φ) on \mathcal{M} and (V, ψ) on \mathcal{N} , the coordinate representation
¹⁶ of f will be

¹⁷
$$\varphi \circ f \circ \varphi^{-1}: R^m \rightarrow R^n. \quad (2.10)$$

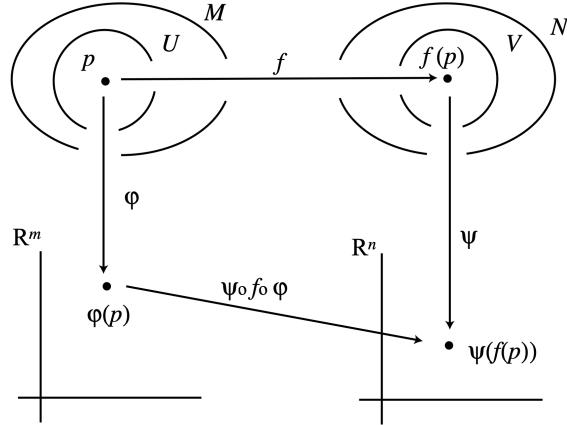


Figure 2.3 A map $f: \mathcal{M} \rightarrow \mathcal{N}$ has coordinates representation $\psi \circ f \circ \varphi^{-1}: \mathbb{R}^m \rightarrow \mathbb{R}^n$.

We can write $\varphi(p) = \{x^\mu\}$ and $\psi(f(p)) = \{y^\alpha\}$, which makes explicit that $y \equiv \psi \circ f \circ \varphi^{-1}(x)$ is an m -variables, vector-valued, usual function. Rendering the coordinate charts implicit, we may write $y^\alpha = f^\alpha(x^\mu)$, which makes clear why f is said to be *differentiable* at p if y is C^∞ . Notice, however, that the differentiability of f is independent of the chosen coordinates.

A very important type of maps between manifold is given by diffeomorphisms.

Definition 14 (Diffeomorphism). Let $f: \mathcal{M} \rightarrow \mathcal{N}$ be a homeomorphism and ψ and φ the same coordinate functions as before. Then, if $\psi \circ f \circ \varphi^{-1}$ is invertible and both $y \equiv \psi \circ f \circ \varphi^{-1}(x)$ and $x \equiv \varphi \circ f^{-1} \circ \psi^{-1}(y)$ are C^∞ , f is called a *diffeomorphism* and \mathcal{M} is said to be *diffeomorphic* to \mathcal{N} , $\mathcal{M} \equiv \mathcal{N}$.

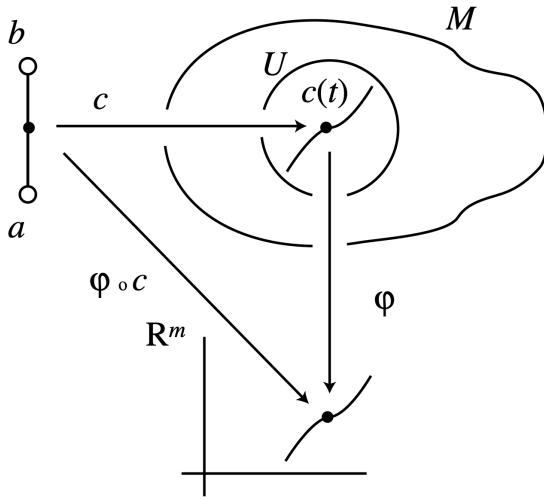
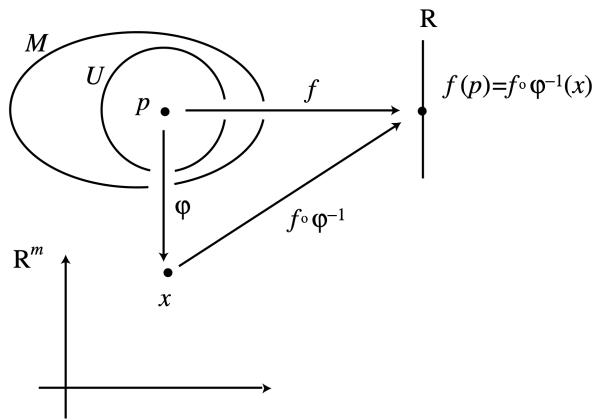
Taking a diffeomorphism from a Manifold into itself, we can implement the concept of change of coordinates, in two different interpretation. First, the set of diffeomorphisms $f: \mathcal{M} \rightarrow \mathcal{M}$ form a group denoted by $\text{Diff}(\mathcal{M})$. Considering a particular $f \in \text{Diff}(\mathcal{M})$, and a chart (U, φ) , such that, for $p \in U$ and $f(p) \in U$, we get $\varphi(p) = x^\mu(p)$ and $\varphi(f(p)) = y^\mu(f(p))$, then y is a differentiable function of x and the above diffeomorphism can be thought as an *active transformation* for a change of coordinates.

However, if (U, φ) and (V, ψ) are overlapping charts, for a point $p \in U \cap V$, there are two coordinates values, i.e., $x^\mu = \varphi(p)$ and $y^\mu = \psi(p)$. Then, the map $x \mapsto y$ is differentiable, and it represents a *passive transformation* for a change of coordinates.

Two important kinds of maps are curves and functions.

Definition 15 (Curve). An *open curve* in an n -dimensional Manifold \mathcal{M} is a map $c: (a, b) \rightarrow \mathcal{M}$, where (a, b) is an open interval such that $a < 0 < b$. A *closed curve* is

16 | Introduction and Motivation

Figure 2.4 A curve c in \mathcal{M} and its coordinate representation $\psi \circ c$.Figure 2.5 A function $f: \mathcal{M} \rightarrow \mathbb{R}$ and its coordinate representation $f \circ \varphi^{-1}$.

¹ a map $c: S^1 \rightarrow \mathcal{M}$. On a chart (U, φ) , a curve $c(t)$ has the coordinate representation
² $x = \varphi \circ c: \mathbb{R} \rightarrow \mathbb{R}^n$. See fig. 2.4

³ **Definition 16 (Function).** A *function* f on \mathcal{M} is a smooth map from \mathcal{M} to \mathbb{R} . On a
⁴ chart (U, φ) , the coordinate representation of f is given by $f \circ \varphi^{-1}: \mathbb{R}^n \rightarrow \mathbb{R}$, which is
⁵ a real-valued function of n variables. The set of functions is denoted by $\mathfrak{F}(\mathcal{M})$. See
⁶ fig. 2.5

⁷ **2.2.3 Vectors**

⁸ The curves allow us to define a vector as a differential operator. So understand this
⁹ more deeply, let's take a Manifold \mathcal{M} , a curve $c: (a, b) \rightarrow \mathcal{M}$ and a function $f: \mathcal{M} \rightarrow \mathbb{R}$,

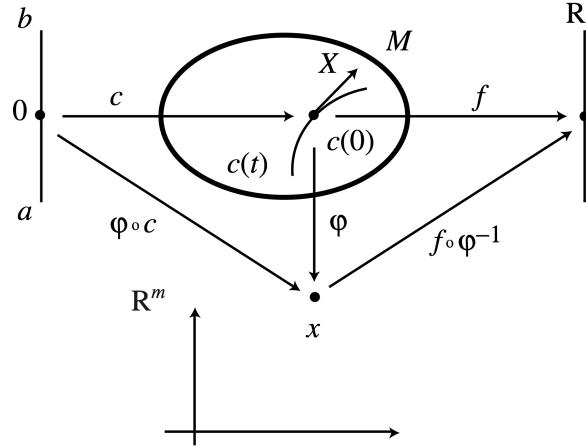


Figure 2.6 A curve c and a function f define a tangent vector along the curve in terms of directional derivatives.

where (a, b) is an open interval containing $t = 0$, as showed in fig. 2.6. The tangent vector at $c(0)$ is defined to be the directional derivative of a function $f(c(t))$ along the curve $c(t)$ at $t = 0$. In particular,

$$X[f] := \frac{df(c(t))}{dt} \Big|_{t=0} = \frac{\partial f}{\partial x^\mu} \frac{dx^\mu(c(t))}{dt} \Big|_{t=0} = X^\mu \left(\frac{\partial f}{\partial x^\mu} \right), \quad (2.11)$$

where we defined

$$X = X^\mu \left(\frac{\partial}{\partial x^\mu} \right), \quad X^\mu = \frac{dx^\mu(c(t))}{dt} \Big|_{t=0} \quad (2.12)$$

Notice that we used abuse of notation, that is $\partial f / \partial x^\mu$ really means $\partial(f \circ \varphi^{-1}(x)) / \partial x^\mu$.

Trying to be more precise, as known from multivariate analysis, whenever we have a curve, we can define an equivalence class. This case is no different. Indeed, given two curves $c_1(t)$ and $c_2(t)$ on \mathcal{M} , if they satisfy

$$c_1(0) = c_2(0) = p, \quad (2.13a)$$

$$\frac{dx^\mu(c_1(t))}{dt} \Big|_{t=0} = \frac{dx^\mu(c_2(t))}{dt} \Big|_{t=0}, \quad (2.13b)$$

then they yield the same differential operator X at p . Then, the above conditions define an equivalence relation \sim such that $c_1(t) \sim c_2(t)$. This allows us to define

Definition 17 (Vector). A *tangent vector* X is the *equivalence class of curves*

$$[c(t)] = \left\{ \tilde{c}(t) \mid \tilde{c}(0) = c(0) \wedge \frac{dx^\mu(\tilde{c}(t))}{dt} \Big|_{t=0} = \frac{dx^\mu(c(t))}{dt} \Big|_{t=0} \right\}. \quad (2.14)$$

18 | Introduction and Motivation

1 All the tangent vectors at a particular point $p \in \mathcal{M}$ form a vector space called *tangent*
 2 *space* of \mathcal{M} at p , denoted by $T_p\mathcal{M}$. By means of equation (2.12), it's straightforward to
 3 see that the basis vectors of this vector space are

$$4 \quad \left\{ e_\mu = \frac{\partial}{\partial x^\mu} \right\}, \quad \mu = 1, \dots, n. \quad (2.15)$$

5 Clearly, $\dim T_p\mathcal{M} = \dim \mathcal{M}$. Further, for each $V \in T_p\mathcal{M}$, we can expand it as $V =$
 6 $V^\mu e_\mu = V^\mu \partial_\mu$, and we call V^μ the components of V with respect to the basis.

7 Obviously, $\{e_\mu\}$ are not the only possible basis for the vector space $T_p\mathcal{M}$. Indeed,
 8 as known from linear algebra, an arbitrary linear combination $\hat{e}_i := A_i^\mu e_\mu$, with $A =$
 9 $(A_i^\mu) \in GL(n, \mathbb{R})$, is a basis as well. In this case, $\{\hat{e}_i\}$ is called a *non-coordinate basis*.

10 Further, given that a vector exists independently of its coordinates, we can take two
 11 coordinate charts, with a point in the intersection of their domains, i.e., $p \in U_i \cap U_j$, with
 12 $x = \varphi_i(p)$ and $y = \varphi_j(p)$. A generic vector $X \in T_p\mathcal{M}$ can be equivalently expanded as

$$13 \quad X = X^\mu \frac{\partial}{\partial x^\mu} = \tilde{X}^\mu \frac{\partial}{\partial y^\mu}, \quad (2.16)$$

14 which shows the relation of the components of a vector in two different charts, i.e,

$$15 \quad \tilde{X}^\mu = X^\nu \frac{\partial y^\mu}{\partial x^\nu}. \quad (2.17)$$

16 2.2.4 One-Forms

17 Given that $T_p\mathcal{M}$ is a vector space, we can define its dual vector space, whose
 18 elements are linear functional from $T_p\mathcal{M}$ to \mathbb{R} . This space is called *cotangent space*, is
 19 denoted by $T_p^*\mathcal{M}$, and their elements $\omega: T_p\mathcal{M} \rightarrow \mathbb{R} \in T_p^*\mathcal{M}$ are called *one-forms*.

20 The simplest one-form is the *differential* of a function $f \in \mathfrak{F}(\mathcal{M})$. Indeed, for a vector
 21 $V \in T_p\mathcal{M}$, its action on f is

$$22 \quad V[f] = V^\mu \frac{\partial f}{\partial x^\mu} \in \mathbb{R}. \quad (2.18)$$

23 So, we define the action of $df \in T_p^*\mathcal{M}$ on a vector $V \in T_p\mathcal{M}$ by

$$24 \quad \langle df, V \rangle \equiv V[f] = V^\mu \frac{\partial f}{\partial x^\mu} \in \mathbb{R}. \quad (2.19)$$

25 In coordinates $x = \varphi(p)$, the differential can be expanded as

$$26 \quad df = \frac{\partial f}{\partial x^\mu} dx^\mu, \quad (2.20)$$

which clearly shows that $\{\mathrm{d}x^\mu\}$ is a basis of $T_p^*\mathcal{M}$. In particular, it's the dual basis of $\{\partial_\mu\}$, since

$$\langle \mathrm{d}x^\mu, \partial_\mu \rangle = \frac{\partial x^\nu}{\partial x^\mu} = \delta_\mu^\nu. \quad (2.21)$$

It's now easy to consider generic one-forms $\omega \in T_p^*\mathcal{M}$ and generic vectors $V \in T_p\mathcal{M}$. Picking a coordinate basis for the tangent space and its dual basis for the cotangent one, we can expand $\omega = \omega_\mu \mathrm{d}x^\mu$ and $V = V^\mu \partial_\mu$. Then, as usual, we can define an *inner product* $\langle \cdot, \cdot \rangle : T_p^*\mathcal{M} \times T_p\mathcal{M} \rightarrow \mathbb{R}$ by

$$\langle \omega, V \rangle = \omega_\mu V^\mu \langle \mathrm{d}x^\mu, \partial_\nu \rangle = \omega_\mu V^\nu \delta_\nu^\mu = \omega_\mu V^\mu. \quad (2.22)$$

Finally, analogously than before, we can pick two charts and a point in their domain's intersection, $p \in U_i \cap U_j$, so that

$$\omega = \omega_\mu \mathrm{d}x^\mu = \tilde{\omega}_\nu \mathrm{d}y^\nu, \quad (2.23)$$

with $x = \varphi_i(p)$ and $y = \varphi_j(p)$. Then, we can easily infer the transformation rule of a one-form under a change of coordinates, indeed,

$$\mathrm{d}y^\nu = \frac{\partial y^\nu}{\partial x^\mu} \mathrm{d}x^\mu \implies \tilde{\omega}_\nu = \omega_\mu \frac{\partial x^\mu}{\partial y^\nu}. \quad (2.24)$$

2.2.5 Tensors and Tensor Fields

From multilinear algebra, it should be familiar that a *tensor* of type (p, q) is a multilinear map which maps q elements of $T_p^*\mathcal{M}$ and r elements of $T_p\mathcal{M}$ to a real number:

$$T^{(q,r)} : \underbrace{T_p^*\mathcal{M} \otimes \cdots \otimes T_p^*\mathcal{M}}_{q \text{ times}} \otimes \underbrace{T_p\mathcal{M} \otimes \cdots \otimes T_p\mathcal{M}}_{r \text{ times}} \longrightarrow \mathbb{R}. \quad (2.25)$$

Picking a coordinate basis and its dual basis,

$$T^{(q,r)} = T^{\mu_1 \dots \mu_q}_{\nu_1 \dots \nu_r} \frac{\partial}{\partial x^{\mu_1}} \dots \frac{\partial}{\partial x^{\mu_q}} \mathrm{d}x^{\nu_1} \dots \mathrm{d}x^{\nu_r}. \quad (2.26)$$

The set of type (q, r) tensors at $p \in \mathcal{M}$ is denoted by $\mathcal{T}_{r;p}^q(\mathcal{M})$. Further, taking r -vectors $V_a = V_a^\mu \partial_\mu$ and q one-forms $\omega_a = \omega_{a\mu} \mathrm{d}x^\mu$, the action of $T^{(q,r)}$ on them is given by

$$T(\omega_1, \dots, \omega_q; V_1, \dots, V_r) = T^{\mu_1 \dots \mu_q}_{\nu_1 \dots \nu_r} \omega_{1\mu_1} \dots \omega_{q\mu_q} V_1^{\nu_1} \dots V_r^{\nu_r}. \quad (2.27)$$

20 | Introduction and Motivation

1 In order to probe a Manifold structure, it's necessary to have fields defined in it,
 2 i.e., structures which are smoothly defined for each point of \mathcal{M} . Knowing the tensor
 3 structure at a point, it's easy to generalize it to the entire Manifold. To begin with, let's
 4 consider the simplest tensor, i.e., a vector, which is a $(1, 0)$ tensor.

5 A *vector field* is a vector assigned smoothly to each point of \mathcal{M} . In other words, V is
 6 a vector field if $V[f] \in \mathfrak{F}(\mathcal{M})$, for any $f \in \mathfrak{F}(\mathcal{M})$. Then, each component of a vector
 7 field is itself a smooth function from \mathcal{M} to \mathbb{R} . We denote the set of vector fields with
 8 $\mathfrak{X}(\mathcal{M})$. Then, a vector $X \in \mathfrak{X}(\mathcal{M})$ computed at a point $p \in \mathcal{M}$ is a vector at $T_p\mathcal{M}$, i.e.,
 9 $X|_p \in T_p\mathcal{M}$.

10 Similarly, a *tensor field* of type (q, r) is a smooth assignment of an element of $\mathcal{T}_{r;p}^q(\mathcal{M})$
 11 at each point $p \in \mathcal{M}$. The set of tensor fields of type (q, r) on \mathcal{M} is denoted by $\mathcal{T}_r^q(\mathcal{M})$.

12 An important tensor field we're interested in is the metric tensor, which is tightly
 13 related to the definition of a Ricci Flow, in the context of Riemannian Manifolds.
 14 However, let's first study an important property of vector fields: they can generate
 15 a flow on a Manifold. Studying further the relation among flows and differential
 16 equation, allows us to better grasp the definition of a Ricci Flow.

17 2.2.6 Flow generated by a vector field

18 Let X be a vector field in \mathcal{M} , $X \in \mathfrak{X}(\mathcal{M})$. An *integral curve* $x(t)$ of X is a curve in
 19 \mathcal{M} , whose tangent vector at $x(t)$ is $X|_{x(t)}$. In a chart (U, φ) , this is equivalent to

$$20 \quad \frac{dx^\mu}{dt} = X^\mu(x(t)), \tag{2.28}$$

21 where $x^\mu(t)$ is the μ -th component of $\varphi(x(t))$ and $X = X^\mu \partial_\mu$. Pay attention to the abuse
 22 of notation. Indeed, we used x to denote both a point on \mathcal{M} and its coordinates.

23 Then, to finding the integral curve of a vector field X is equivalent to solving the
 24 system of ODEs (2.28), with initial condition $x_0^\mu = x^\mu(0)$, which are the coordinates of
 25 the integral curve at $t = 0$.

26 The existence and uniqueness theorem of ODEs guarantees that there is a unique
 27 solution to (2.28), at least locally, with the initial condition x_0^μ .

28 Let, then, $\sigma(t, x_0)$ be an integral curve of X passing through a point x_0 at $t = 0$ and
 29 let's denote its coordinates by $\sigma^\mu(t, x_0)$. Then, eq. (2.28) becomes

$$30 \quad \frac{d}{dt} \sigma^\mu(t, x_0) = X^\mu(\sigma(t, x_0)), \tag{2.29}$$

with the initial condition

$$\sigma^\mu(0, x_0) = x_0^\mu. \quad (2.30)$$

The map $\sigma: \mathbb{R} \times \mathcal{M} \rightarrow \mathcal{M}$ is called *flow* generated by $X \in \mathfrak{X}(\mathcal{M})$. We can prove it satisfies the condition

$$\sigma(t, \sigma^\mu(s, x_0)) = \sigma(t + s, x_0), \quad (2.31)$$

for $s, t \in \mathbb{R}$.

Proof. Since

$$\frac{d}{dt}\sigma^\mu(t, \sigma^\mu(s, x_0)) = X^\mu(\sigma(t, \sigma^\mu(s, x_0))),$$

$$\sigma(0, \sigma(s, x_0)) = \sigma(s, x_0),$$

and

$$\frac{d}{dt}\sigma^\mu(t + s, x_0) = \frac{d}{d(t+s)}\sigma^\mu(t + s, x_0) = X^\mu(\sigma(t + s, x_0)),$$

$$\sigma(0 + s, x_0) = \sigma(s, x_0),$$

both sides of eq. (2.31) satisfy the same ODE and the same initial condition. Then, from the uniqueness theorem for the solution of ODEs, they must be the same. ■

This leads to the following theorem.

Theorem T.2. *For any point $x \in \mathcal{M}$, there exists a differentiable map $\sigma: \mathbb{R} \times \mathcal{M} \rightarrow \mathcal{M}$ such that*

- $\sigma(0, x) = x;$

- $t \mapsto \sigma(t, x)$ is a solution of (2.29) and (2.30);

- $\sigma(t, \sigma^\mu(s, x)) = \sigma(t + s, x).$

We're now ready to introduce Riemannian Manifold and their most relevant properties.

2.2.7 Riemannian Manifolds

A Riemannian Manifold is defined as a smooth Manifold which admits a Riemannian metric.

22 | Introduction and Motivation

¹ **Definition 18.** Let \mathcal{M} be a differentiable Manifold. A *Riemannian metric* g on \mathcal{M} is a
² type $(0, 2)$ tensor field on \mathcal{M} such that, at each point $p \in \mathcal{M}$:

- ³ • $g_p(U, V) = g_p(V, U)$,
- ⁴ • $g_p(U, V) \geq 0$, where the equality holds only when $U = 0$.

⁵ Here, $U, V \in T_p\mathcal{M}$ and $g_p = g|_p$. Basically, g_p is a symmetric positive-definite bilinear
⁶ form.

⁷ Recall the previous definition of inner product (2.22) between vectors and dual
⁸ forms. For $V \in T_p\mathcal{M}$ and $\omega \in T_p^*\mathcal{M}$, then $\langle ., . \rangle: T_p^*\mathcal{M} \times T_p\mathcal{M} \rightarrow \mathbb{R}$. If there exists a
⁹ metric tensor g , then, we can use it to define the inner product between two vectors
¹⁰ $U, V \in T_p\mathcal{M}$, specifically by $g_p(U, V)$. Since $g_p: T_p\mathcal{M} \otimes T_p\mathcal{M} \rightarrow \mathbb{R}$, we may define
¹¹ a linear map $g_p(U, .): T_p\mathcal{M} \rightarrow \mathbb{R}$ by $V \mapsto g_p(U, V)$. Then, it's straightforward that
¹² $g_p(U, .) \in T_p^*\mathcal{M}$ is a one-form. Thus, the metric g_p gives rise to an isomorphism
¹³ between $T_p\mathcal{M}$ and $T_p^*\mathcal{M}$.

¹⁴ Picking a chart (φ, U) , with coordinates $\{x^\mu\}$, we can expand the metric tensor in
¹⁵ coordinates as

$$\text{¹⁶} \quad g_p = g_{\mu\nu}(p) dx^\mu \otimes dx^\nu, \quad g_{\mu\nu}(p) = g_p(\partial_\mu, \partial_\nu) = g_{\nu\mu}(p), \quad p \in \mathcal{M}. \quad (2.32)$$

¹⁷ It's usual convention to forget about the point p , denote the inverse metric as $g^{\mu\nu}$,
¹⁸ and the determinant as $\det(g_{\mu\nu}) := g$ and $\det(g^{\mu\nu}) := g^{-1}$. Then, the isomorphism
¹⁹ between $T_p\mathcal{M}$ and $T_p^*\mathcal{M}$ can be expressed as

$$\text{²⁰} \quad \omega_\mu = g_{\mu\nu} U^\nu, \quad U^\mu = g^{\mu\nu} \omega_\nu. \quad (2.33)$$

²¹ The purpose of the next chapter will be to rigorously define the Ricci Flow. How-
²² ever, to conclude this motivation part, let's take a brief detour onto string theory, in
²³ which, with a simple example in the context of non-linear sigma models, it's possible to
²⁴ grasp the meaning of the Ricci Flow application to understand topological properties
²⁵ of a Manifold through renormalization group flows in a Riemannian Manifold.

2.2.8 Curvature and Ricci tensor

1

2.2.9 Covariant Derivative

2

2.3 Non-Linear Sigma Models and String Theory

3

The standard starting point of string theory is *Polyakov action*, which describes a bosonic classical, one-dimensional, string, which describes a two-dimensional worldsheet Σ on a 26-dimensional spacetime described by the spacetime coordinates $X^\mu(\xi)$, $\mu = 0, \dots, 25$, where $\xi^a = (\tau, \sigma)$, $a = 1, 2$, are the intrinsic coordinates on Σ . The metric on spacetime is denoted by $g_{\mu\nu}$, while the metric on the worldsheet is γ_{ab} . For a flat spacetime, with Minkowski metric $g_{\mu\nu} \equiv \eta_{\mu\nu} = \text{diag}(-1, +1, +1, +1)$, the action reads

$$S_P[X^\mu(\xi), \gamma_{ab}(\xi)] = -\frac{T}{2} \int_\Sigma d\tau d\sigma \sqrt{-\det(\gamma)} \gamma^{ab} \partial_a X_\mu(\xi) \partial_b X^\mu(\xi), \quad (2.34)$$

where T is a characteristic parameter of the string, related to the string length l .

11

The symmetries of this action allow us to consider a flat worldsheet metric, $\gamma_{ab} = \eta_{ab}$, considering the so-called *unit gauge*. Then, reintroducing explicitly the metric $g_{\mu\nu}$, even if it's flat in this case, we obtain

$$S_P = -T \int_\Sigma d^2\xi g_{\mu\nu}(X) \partial_a X^\mu \partial^a X^\nu. \quad (2.35)$$

Basically, it represents a 2-dimensional field theory on the worldsheet, where the coordinates X^μ are 26 dynamical fields in there. This allows us to quantize the theory with the usual quantization prescription, based on the substitution of the classical Poisson brackets defined on a symplectic manifold with the commutators of operators acting on a Hilbert space.

16

17

18

19

20

After quantization, one notice that, for a closed string, defined by the periodicity condition $X^\mu(\tau, \sigma) = X^\mu(\tau, \sigma + l)$, the particle spectrum contains a *graviton* $\gamma_{\mu\nu}$, which resembles a gravitational wave at low energies, a scalar field φ called *dilaton* and an antisymmetric two-tensor $b_{\mu\nu}$ called *Kalb-Ramond tensor*.

21

22

23

24

Therefore, due to the presence of the graviton, one could wonder what happens for a non-flat spacetime. Then, after redefining the coordinates as a constant X_0^μ plus some other arbitrary fields Y^μ , i.e.,

$$X^\mu(\xi) = X_0^\mu(\xi) + \sqrt{\alpha'} Y^\mu(\xi), \quad (2.36)$$

25

26

27

28

24 | Introduction and Motivation

1 we can expand the term in the Lagrangian as

$$2 g_{\mu\nu}(X) \partial_a X^\mu \partial^a X^\nu \\ = \alpha' \left[g_{\mu\nu}(X_0) + \sqrt{\alpha'} g_{\mu\nu,\rho}(X_0) Y^\rho(\xi) + \frac{\alpha'}{2} g_{\mu\nu,\rho\sigma}(X_0) Y^\rho(\xi) Y^\sigma(\xi) + \dots \right] \partial_a Y^\mu \partial^a Y^\nu, \quad (2.37)$$

3 where $g_{\mu\nu,\rho} \equiv \partial_\rho g_{\mu\nu}$.

4 We obtained an expansion in α' , where each term is an interaction term for the fields
5 Y^μ , with couplings given by the derivatives of the metric. However, a crucial symmetry
6 of the Polyakov action (2.34) is the invariance under *conformal transformations*. Those
7 are diffeomorphisms on a Riemannian Manifold which preserve the metric up to
8 rescaling, i.e.,

$$9 g(x) \rightarrow \tilde{g}(\tilde{x}) = e^{2\omega(\tilde{x})} g(\tilde{x}). \quad (2.38)$$

10 Without going into the details, the presence of this symmetry is considered as a
11 consistency condition for the theory, as it allows for a perturbative interpretation of the
12 interactions. In addition, the above interacting quantum field theory must undergo
13 renormalization, in order to cure the divergences.

14 However, after renormalization, the conformal symmetry may be anomalous².
15 Indeed, a particular example of conformal transformation (2.38) is provided by scale-
16 invariance. That is to say, the theory can't depend on a scale to be conformal invariant.
17 But, as the methods of *renormalization group* teach us, after renormalization the cou-
18 plings run with the energy scale M of the system, dependence which is encapsulated
19 into the β -function

$$20 \beta(g_{\mu\nu}) = M \frac{\partial}{\partial M} g_{\mu\nu}. \quad (2.39)$$

21 As a consequence, for a quantum string theory to make sense, it must be conformal
22 invariance, so the β -function associated to the metric, in the action (2.35) with the
23 expansion (2.37), must vanish

$$24 \beta(g_{\mu\nu}) \stackrel{!}{=} 0. \quad (2.40)$$

25 A similar argument can be pursued for the other two particles in the closed string
26 spectrum, i.e., the dilaton φ and the Kalb-Ramond form $b_{\mu\nu}$. The corresponding action

²An anomaly is a classical symmetry which is not preserved at the quantum level. In particular, it is due to a non-invariance of the measure of the path integral.

will be

$$S_\sigma = -\frac{T}{2} \int_{\Sigma} d^2\xi \sqrt{-\det(\gamma)} [(\gamma^{ab} g_{\mu\nu}(X) + i\varepsilon^{ab} b_{\mu\nu}(X)) \partial_a X^\mu \partial_b^\nu + \alpha' \mathcal{R}\varphi(X)], \quad (2.41)$$

where ε^{ab} is the 2d Levi-Civita symbol, while $\mathcal{R} = \mathcal{R}(\gamma)$ is the Ricci scalar on the worldsheet.

Defining the Field Strength $H_{\mu\nu\rho} = \partial_\mu b_{\nu\rho} + \partial_\nu b_{\rho\mu} + \partial_\rho b_{\mu\nu}$, the vanishing of the beta functions reads

$$\beta(g_{\mu\nu}) = \alpha' \left(R_{\mu\nu} - \frac{1}{4} H_{\mu\lambda\rho} H^{\mu\lambda\rho} + 2\nabla_\mu \nabla_\nu \varphi \right) + O(\alpha'^2) \stackrel{!}{=} 0, \quad (2.42a)$$

$$\beta(b_{\mu\nu}) = \alpha' \left(\frac{1}{2} \nabla^\rho H_{\rho\mu\nu} + \nabla^\rho \varphi H_{\rho\mu\nu} \right) + O(\alpha'^2) \stackrel{!}{=} 0, \quad (2.42b)$$

$$\beta(\varphi) = \alpha' \left(\frac{1}{2} \nabla_\mu \varphi \nabla^\mu \varphi - \frac{1}{2} \nabla^2 \varphi - \frac{1}{24} H_{\mu\nu\rho} H^{\mu\nu\rho} \right) + O(\alpha'^2) \stackrel{!}{=} 0. \quad (2.42c)$$

The above equations are constraints for the spacetime fields (g, b, φ) , imposed to preserve conformal invariance of the quantum string. However, since those fields should be dynamical on spacetime, those must also be their equations of motion. This leads to the following *low-energy effective action*, which has (2.42) as equations of motion

$$S_{26} = \frac{1}{k_0^2} \int d^{26}x \sqrt{\det(g)} e^{-2\varphi} \left(\mathcal{R}(g) - \frac{1}{12} H_{\mu\nu\rho} H^{\mu\nu\rho} + 4\nabla_\mu \varphi \nabla^\mu \varphi \right). \quad (2.43)$$

To set this problem to a more general ground, and understand how this model is related to Ricci Flow, let's define more accurately what a σ -model is in field theory. A σ -model is a field theory for a field $\Phi: \Sigma \rightarrow \mathcal{M}$ that takes values in a manifold \mathcal{M} . Traditionally, Σ is the spacetime on which the field theory lives, and \mathcal{M} is called the target space. If the target space carries some linear structure, like a vector space, then the whole physical system is called a linear σ -model. For general manifolds such as generic Riemannian ones, it is then called a non-linear σ -model. What we did above is to study the σ -model spacetime renormalization effects on the target space \mathcal{M} , and this is indeed an instance of Ricci Flow.

2.4 Ricci Flow to Tackle Topological Problems

Draft - v1.0

February 27, 2025 – 08:51

CHAPTER 3

RICCI FLOW AND OLIVER-RICCI CURVATURE

Ricci Flow arose historically as a powerful method in differential geometry, initially introduced by Richard S. Hamilton in the early 1980s. At its heart, Ricci Flow seeks to “smooth out” geometric irregularities of manifolds by evolving the underlying Riemannian metric through a partial differential equation (PDE) reminiscent of the classical heat equation. In the context of differential geometry, a manifold is a topological space that locally resembles Euclidean space, and a *Riemannian manifold* is such a space equipped with an inner product on each tangent space, making it possible to measure angles, distances, and curvature.

Curvature, in particular, is fundamental to geometry: it describes how space bends or deviates from flatness. On a two-dimensional surface embedded in three-dimensional space, for example, curvature can be visualized by examining the deviation of geodesics (the generalization of “straight lines” in curved spaces) from parallelism, or by looking at how areas or angles are distorted compared to those in flat Euclidean geometry. The extension to higher dimensions and more abstract manifolds involves careful definitions but retains this key notion of “spatial bending.”

3.1 Riemannian Geometry and Curvature Notions

In classical Riemannian geometry, curvature can be examined from multiple perspectives. One can study the *sectional curvature*, which measures how a two-dimensional plane (spanned by two tangent directions) curves. One can also investigate the *Ricci curvature*, which is obtained by summing or averaging the sectional curvatures over all planes containing a given direction. Ricci curvature has special importance: for instance, it appears in Einstein’s equations of General Relativity, linking geometry to matter and energy distributions in spacetime.

28 | Ricci Flow and Oliver-Ricci Curvature

1 Concretely, let (M, g) be a Riemannian manifold, where M is a smooth manifold
 2 and g is the metric tensor. The Ricci curvature Ric is derived as a contraction of the
 3 Riemann curvature tensor, itself an operator capturing how much nearby geodesics
 4 converge or diverge. Positive Ricci curvature typically implies that geodesics tend to
 5 converge, reflecting a “crowded” or positively curved geometry akin to the sphere.
 6 Negative Ricci curvature implies geodesics tend to diverge, mirroring a hyperbolic or
 7 “saddle-like” structure. Zero Ricci curvature is the hallmark of Ricci-flat manifolds,
 8 with many implications for geometry and topology.

9 **3.1.1 Hamilton’s Ricci Flow Equation**

10 Hamilton introduced the Ricci Flow as the PDE:

$$11 \quad \frac{\partial g_{ij}}{\partial t} = -2 R_{ij},$$

12 where g_{ij} are the components of the metric tensor g in local coordinates and R_{ij} are the
 13 components of the Ricci curvature tensor. Informally, each infinitesimal piece of the
 14 manifold changes in time, guided by curvature. Regions of *high positive* Ricci curvature
 15 shrink faster, while regions of *negative* Ricci curvature expand. This leads to a *flow* that
 16 tends to smooth out the geometric and topological features of M .

17 One of the most famous applications of Ricci Flow on manifolds is Grigori Perel-
 18 man’s resolution of the Poincaré Conjecture and the more general Geometrization
 19 Conjecture for three-dimensional manifolds. Perelman’s work introduced the notion of
 20 *Ricci Flow with surgery*, a procedure to remove singular regions (places where curvature
 21 blows up to infinity) and continue the flow on the remaining parts. In 3D manifolds,
 22 these singularities can be visualized as “neck pinches” that effectively separate the
 23 manifold into topologically simpler pieces. Perelman showed that by performing a
 24 series of well-defined surgeries, one could decompose a three-dimensional manifold
 25 into model geometric pieces, completing Hamilton’s program toward a proof of the
 26 Geometrization Conjecture.

27 **3.1.2 From Smooth Settings to Discrete Geometry**

28 While Ricci Flow is classically defined on smooth manifolds, there has been con-
 29 siderable interest in transferring these ideas to *discrete* or combinatorial settings such
 30 as polyhedral surfaces, graphs, and complex networks. The general question is how
 31 to define concepts like “curvature” when one does not have a smooth manifold or a

Riemannian metric in the usual sense. Instead, discrete analogs focus on adjacency, distances along edges, and combinatorial properties that mimic or reflect continuum notions.

For surfaces composed of polygons (triangulations), one can define the curvature at a vertex via angle deficits, a concept dating back to classical differential geometry of polyhedral surfaces. However, for higher-dimensional graphs and networks that are not neatly embedded in any Euclidean space, a more general curvature definition is needed—one that depends mostly on the underlying distances and probability measures rather than an explicit embedding.

3.2 Optimal Transport and Ollivier’s Ricci Curvature

A major breakthrough in defining a curvature notion for general metric spaces (including discrete networks) came via *optimal transport*. Historically, the optimal transport problem, originating in the work of Gaspard Monge in the 18th century, asks how to map one mass distribution into another with minimal transportation cost. Subsequent reformulations by Leonid Kantorovich turned this into a linear optimization problem known as the *Kantorovich relaxation*.

If (X, d) is a metric space, and we have two probability measures μ and ν on X , the *Wasserstein distance* (also called the earth mover’s distance) measures how much “effort” is needed to move mass from μ to ν , given the metric d . Specifically, one solves an optimization problem that tries to minimize the total cost of moving infinitesimal amounts of mass from one location to another.

Yann Ollivier harnessed this framework to define a notion of *coarse Ricci curvature* on general metric measure spaces. Ollivier’s definition, now widely called *Ollivier Ricci curvature*, is based on considering small probability balls of radius ε around points (or sometimes discrete probability measures concentrated on nearest neighbors in a graph) and calculating how much these small balls cost to move one onto the other under optimal transport. If moving these balls requires comparatively more effort than just their pairwise distance would suggest, the edge or connection between them is deemed negatively curved; if it requires less effort, it is positively curved. This emerges from an analog of the well-known statement in Riemannian geometry that Ricci curvature controls how geodesic balls deviate from each other, which can be recast in terms of mass transport.

Draft - v1.0

February 27, 2025 – 08:51

CHAPTER 4

APPLICATION TO COMPLEX NETWORKS

To adapt Ollivier's construction to a graph $G = (V, E)$ where V is the set of nodes and E is the set of edges (possibly with weights), one often assigns to each node x a probability measure m_x . A typical choice is to concentrate the mass uniformly on x 's neighbors, possibly with some parameter α to keep a fraction of the mass at x itself. Let $W(m_x, m_y)$ denote the Wasserstein distance between m_x and m_y . The Ollivier Ricci curvature $\kappa(x, y)$ along the edge (x, y) is defined as

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

where $d(x, y)$ is the usual shortest-path distance (or a weight-based distance) between x and y . Intuitively:

- If many of the neighbors of x align well with the neighbors of y , then $W(m_x, m_y)$ is relatively small compared to $d(x, y)$, giving a larger curvature.
- If the neighbors do not overlap much, $W(m_x, m_y)$ will be comparatively large, implying smaller (or possibly negative) curvature.

This lines up with the idea that a highly “clustered” or cohesive set of nodes—often indicating an underlying community—acts more like a positively curved region in the manifold analogy, whereas edges bridging distant clusters reflect negative curvature. These insights lead to a method for analyzing and partitioning networks by focusing on edges with specific curvature characteristics.

1 **4.1 General Application of Ricci Flow to Complex Net-**
 2 **works: ARI, Modularity, and Performance**

3 In the study of complex networks, a fundamental task is to identify densely con-
 4 nected groups of nodes, commonly referred to as *communities*. These communities
 5 often correspond to meaningful substructures such as friend groups in social networks,
 6 functionally related proteins in biological networks, or topics in citation networks.
 7 There are numerous algorithms that aim to extract communities—ranging from graph
 8 partitioning heuristics and centrality-based edge removal to statistical and probabilistic
 9 methods.

10 The Ricci Flow-based technique for network community detection focuses on the
 11 geometric viewpoint: an edge with significantly negative Ricci curvature may signify
 12 a *bridge* between distinct communities, whereas edges with positive curvature are
 13 typically nestled within cohesive communities. Iteratively adjusting edge weights
 14 according to Ollivier Ricci curvature has the effect of “magnifying” bridging edges
 15 and “contracting” internal edges, ultimately making a subsequent threshold-based cut
 16 reveal the inherent clusters.

17 Here are the major steps (in broad terms) for using Ricci Flow to detect communi-
 18 ties:

- 19 1. **Initialization:** Assign an initial weight to each edge (x, y) . Often, one starts
 20 with uniform weights or with weights based on an existing property such as
 21 adjacency or similarity.
- 22 2. **Probability Measures:** Choose how to define the measure m_x at each node
 23 x . A popular simple choice is to put uniform weight on all neighbors of x ,
 24 ensuring $\sum_{v \in \text{neighbors}(x)} m_x(v) = 1$. Other weighting schemes (e.g., discounting
 25 more distant neighbors) can also be used.
- 26 3. **Curvature Computation:** For each edge (x, y) , compute the Ollivier Ricci curva-
 27 ture $\kappa(x, y)$ by solving the discrete optimal transport problem and using

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

- 28 4. **Discrete Ricci Flow Update:** Adjust the edge weight according to

$$w_{xy}^{(i+1)} = w_{xy}^{(i)} - \eta \kappa_{xy}^{(i)} d_{xy}^{(i)}.$$

 4.1 General Application of Ricci Flow to Complex Networks: ARI, Modularity, and
 Performance | 33

Often, one sets $\eta = 1$ and updates all edges simultaneously, then recomputes shortest path distances $d(\cdot, \cdot)$ for the next iteration. The total number of iterations can be chosen based on convergence criteria or practical heuristics.

5. **Network Surgery:** After a certain number of iterations, examine the distribution of edge weights. Typically, bridging edges (those connecting separate communities) will have grown in length. Choose a threshold T such that edges with $w_{xy} > T$ are considered “cuts,” removing them from the graph. The connected components that remain are taken as the identified communities.
6. **Post-Processing:** If a graph has hierarchical communities, additional steps (e.g., repeating the process within subcomponents) might be performed to further subdivide the clusters.

This pipeline is quite flexible in terms of parameter choices (e.g., the measure definition, the number of flow iterations, the threshold for edge cuts) and can adapt to various network topologies. The next subsections explain how to measure the resulting partition quality, focusing on two widely employed tools: the *Adjusted Rand Index (ARI)* and *Modularity*.

4.1.1 Adjusted Rand Index (ARI)

General Definition

The *Adjusted Rand Index (ARI)* is a popular external validation measure to compare a discovered clustering with a known ground-truth partition. Suppose you have a set of n items (in our case, the n nodes of a network). Let $C = \{C_1, \dots, C_r\}$ be a partition of these items into r clusters found by some method, and let $G = \{G_1, \dots, G_s\}$ be the ground-truth (or reference) partition into s clusters. The Rand Index (RI) measures the fraction of item pairs that are *consistently* assigned in both partitions (i.e., either assigned together in both or assigned to different clusters in both).

Formally, the Rand Index is given by:

$$\text{RI}(C, G) = \frac{a + d}{a + b + c + d},$$

where

- a is the number of pairs of items that are in the same cluster in C *and* in the same cluster in G ,

34 | Application to Complex Networks

- 1 • b is the number of pairs of items that are in the same cluster in C but in different clusters in G ,
- 3 • c is the number of pairs that are in different clusters in C but in the same cluster in G ,
- 5 • d is the number of pairs that are in different clusters in C and in different clusters in G .

7 Because $a + d$ counts all the agreements (put together or kept apart) and $b + c$ counts
8 the disagreements, $\text{RI}(C, G)$ is between 0 and 1, with 1 meaning a perfect match of the
9 partitions.

10 However, the Rand Index does not correct for chance agreement. The *Adjusted Rand Index*
11 refines this by subtracting the expected RI of random partitions and rescaling.
12 One can define:

$$13 \quad \text{ARI}(C, G) = \frac{\text{RI}(C, G) - \text{Expected[RI]}}{\max(\text{RI}) - \text{Expected[RI]}},$$

14 which yields a value that ranges from 0 (or negative, depending on definition) up to
15 1. Here, 1 indicates the clustering C exactly matches the ground-truth G , whereas an
16 ARI near 0 suggests random agreement.

17 Applying ARI to Ricci Flow-based Community Detection

18 When applying Ricci Flow on a network, one typically obtains a final partition of
19 the graph by removing edges beyond a certain length. If a ground-truth labeling exists,
20 we compute the ARI to assess how well these communities align with the reference. If
21 ARI is high, that means the geometric approach successfully captured the intended
22 grouping. By examining the ARI as a function of the threshold, one can also decide the
23 best cutoff for the “network surgery.” In practice, one might do a range of threshold
24 values, measure the ARI for each, and pick the threshold that maximizes it (assuming
25 the ground-truth is known). In contexts where the ground-truth partition is not known,
26 we might rely on internal validation measures, such as modularity, to guess a suitable
27 threshold.

28 4.1.2 Modularity

29 Definition of Modularity

30 Modularity is one of the most commonly used internal metrics for community
31 detection in networks. Proposed initially by Newman and Girvan, it quantifies how

 4.1 General Application of Ricci Flow to Complex Networks: ARI, Modularity, and
 Performance | 35

well a particular partition of the network divides the nodes into communities that are dense internally and sparse between each other.

Let us consider a network with n nodes and m edges (or total edge weight if it is a weighted graph). For a given partition of the network into k communities, the modularity Q is computed as:

$$Q = \frac{1}{2m} \sum_{i,j} \left(A_{ij} - \frac{d_i d_j}{2m} \right) \delta(c_i, c_j),$$

where A_{ij} is the adjacency matrix (or the weight matrix), d_i is the degree (or sum of weights) of node i , c_i is the community label of node i , and $\delta(c_i, c_j)$ is 1 if i and j are in the same community and 0 otherwise. The term $\frac{d_i d_j}{2m}$ approximates the expected number of edges (or expected weight) between i and j if edges are distributed randomly but respect node degrees. High modularity indicates that the actual number of intra-community edges is significantly above random expectation.

Modularity as a Stopping or Surgery Criterion

When using Ricci Flow for community detection, one can track how modularity evolves as edges get re-weighted and as one tries different thresholds for cutting. Typically, there is an intuitive sweet spot where further cutting does not substantially improve the modularity and might begin to over-segment the network.

A practical strategy is as follows:

1. Perform a fixed number of Ricci Flow iterations.
2. For a range of potential cut thresholds T_1, T_2, \dots, T_r , remove edges with weight above T_j .
3. Compute modularity Q_j for each T_j .
4. Select the threshold T_j that yields the maximum Q_j .

This threshold selection process is akin to the notion of “neck pinches” or “surgeries” in the manifold setting: a large weight often signifies a bridging structure (negative curvature region grown large) that is “pinching off” from the main components. If an external ground-truth is known, one may prefer the threshold that simultaneously optimizes ARI and modularity. In the absence of external labels, maximizing modularity is a common choice to define the “best” partition.

¹ 4.1.3 Interpreting Network “Surgery” in This Context

² A hallmark of the classical Ricci Flow with surgery on manifolds is that when
³ the flow develops singularities (often visualized as “neck pinches”), the manifold is
⁴ physically separated into topologically distinct pieces. Drawing an analogy, in discrete
⁵ Ricci Flow on graphs, the edges that grow large (due to negative curvature) can be
⁶ viewed as “singularities” or bridging regions, reminiscent of the neck that pinches
⁷ in a continuous manifold. The act of removing these edges at some iteration is the
⁸ direct analog of performing surgery on the manifold. After removing these “necks,”
⁹ the graph breaks into connected components, each presumably representing a dense
¹⁰ or well-curved subregion, i.e., a community.

¹¹ In practice, we carry out this network surgery step once or multiple times, bal-
¹² ancing the preservation of meaningful connectivity with the desire to isolate truly
¹³ separated clusters. Because many real networks can exhibit hierarchical or multi-level
¹⁴ community structures, it is possible that each sub-component can be further refined if
¹⁵ we continue the process within it. This multi-scale approach can be repeated if one
¹⁶ suspects nested communities.

¹⁷ 4.1.4 Algorithmic Complexity and Practical Considerations

¹⁸ While the continuous Ricci Flow PDE can be computationally demanding in the
¹⁹ smooth case, the discrete version has its own set of computational challenges. The
²⁰ major cost typically arises in:

- ²¹ 1. **Wasserstein Distances:** Computing $W(m_x, m_y)$ for each edge (x, y) . In a graph
²² with n nodes and e edges, if we attempt an exact solution of the optimal transport
²³ problem, it can become computationally expensive. However, in many practical
²⁴ discrete settings, especially with local probability measures m_x that concentrate
²⁵ mass on immediate neighbors, $W(m_x, m_y)$ can often be computed with simpler
²⁶ combinatorial formulas or approximations.
- ²⁷ 2. **Repeated Distance Computation:** After each iteration, we update edge weights
²⁸ and potentially must recompute shortest-path distances $d(x, y)$ for all node pairs
²⁹ or for edges. If e is large, repeated all-pairs computations might be costly, unless
³⁰ efficient incremental shortest-path algorithms or approximate methods are used.

³¹ Despite these costs, modern computational resources and heuristics usually make
³² discrete Ricci Flow feasible for networks of moderate size. For very large networks

4.1 General Application of Ricci Flow to Complex Networks: ARI, Modularity, and Performance | 37

(millions of edges), one might rely on approximations, sampling techniques, or simplified versions of curvature (e.g., proxies to Ollivier Ricci curvature).
1
2

4.1.5 Empirical Performance and Use Cases

In various studies—including the main article on *Ricci Flow in Networks* [main-article]—this curvature-based approach has been tested on real-world networks (e.g., social graphs, protein-protein interaction networks, and technological networks) as well as classical synthetic benchmarks. Two classes of synthetic networks are often used:
5
6
7

- **Stochastic Block Models (SBM):** Here, the nodes are divided into communities, with an internal vs. external edge probability controlling how well-separated they are. Ricci Flow methods can detect these communities, typically succeeding until the external link probability becomes too high (i.e. the communities are no longer distinct).
8
9
10
11
12
- **LFR (Lancichinetti-Fortunato-Radicchi) Benchmarks:** LFR graphs incorporate power-law degree distributions and community sizes, matching many real networks. Curvature-based detection has shown strong performance across a range of LFR parameters, typically measured via ARI or modularity.
13
14
15
16

Real-world networks might offer additional challenges, such as overlapping or hierarchical community structures and heterogeneous degree distributions. Nonetheless, the geometric viewpoint remains advantageous, offering both interpretability (in the sense of manifold analogies) and robust detection of bridging edges.
17
18
19
20

4.1.6 Choosing Parameters: Number of Iterations, Threshold, and Measures

In practice, one needs to choose:
23

1. **Number of Ricci Flow Iterations:** Stopping too soon might not emphasize bridging edges enough to isolate communities; iterating too long might lead to degeneracies (e.g., certain edges become extremely large or extremely small). An empirical strategy is to run a moderate number (e.g., 10–20 iterations) and check if measures like ARI (if a ground truth is available) or modularity saturate.
24
25
26
27
28
2. **Cut Threshold for Network Surgery:** Typically chosen by scanning multiple thresholds and evaluating a measure of clustering quality (e.g., modularity) or ARI.
29
30
31

38 | Application to Complex Networks

1 3. **Mass Distributions:** The simplest is uniform distribution on neighbors (some-
2 times with or without a fraction of mass at the node itself). More sophisti-
3 cated distributions might downweight distant neighbors, especially in large or
4 weighted networks.

5 **4.1.7 Broader Theoretical Context and Future Directions**

6 From a more theoretical vantage point, *why* does curvature—in the sense of
7 Ollivier—detect communities so well? Intuitively, the geometry of a manifold with
8 positive curvature is reminiscent of a cohesive, “ball-like” region, while negatively
9 curved regions show hyperbolic expansions akin to branching. In network terms,
10 cohesive subgraphs correspond to a “positive curvature signature” because local ran-
11 dom walks or local mass distributions align more easily, whereas bridging edges or
12 tree-like expansions create a negative curvature effect.

13 Future directions could include:

- 14 • Exploring *Forman-Ricci* curvature or other discrete curvature notions to see how
15 they compare in community detection or anomaly detection tasks.
- 16 • Combining *higher-order* or *simplicial* network topologies, in which edges, triangles,
17 and higher-dimensional simplices each carry a curvature notion, potentially
18 refining the detection of substructures.
- 19 • Investigating multi-scale or hierarchical surgery, systematically refining each
20 sub-community using iterative curvature-based updates.

21 All these highlight that curvature-based community detection offers a distinctive
22 geometric lens on networks, bridging rich ideas from differential geometry to the
23 realm of data science and large-scale graph analysis.

24 **4.1.8 Concluding Remarks on Sections ?? and 4.1**

25 We have seen how the Ricci Flow—born in smooth differential geometry—finds a
26 natural discrete counterpart in network analysis. The conceptual framework of *positive*
27 *curvature* mapping to regions of local coherence and *negative curvature* mapping to
28 bridging edges provides a strong theoretical impetus for using curvature to detect
29 clusters. Alongside this geometric perspective, performance metrics such as *Adjusted*
30 *Rand Index* and *modularity* are critical both for evaluating detection quality and guiding
31 the user to select an appropriate threshold for “network surgery.”

Hence, these two sections form the backbone of a Ricci Flow approach to network community detection, elaborating the geometry behind Ollivier Ricci curvature and the practical aspects of applying a flow-based method. In the subsequent parts of a full report, one might delve into *specific* examples, either from synthetic benchmarks or real-world data sets, to illustrate how the method works in practice. Additionally, code implementations can be crafted (in Python, C++, MATLAB, etc.) to follow the discrete flow steps.

By situating discrete Ricci Flow in this overarching context, we emphasize that the approach is not simply a novel graph partitioning algorithm; rather, it stems from deep geometric principles with extensive theoretical underpinnings. This solid foundation helps ensure that the method is not merely a heuristic but is intimately connected to well-studied ideas in mathematical analysis, topology, and geometry.

4.2 Discrete Ricci Flow

Following Ollivier’s curvature definition, we can define a discrete analog of Ricci Flow by iteratively adjusting the weights of edges in G according to the local curvature. In classical geometry, Ricci Flow modifies the metric in a continuous manner according to -2 Ric . In the discrete setting, one can proceed iteratively:

$$w_{xy}^{(i+1)} = w_{xy}^{(i)} - \eta \kappa_{xy}^{(i)} d_{xy}^{(i)},$$

where $w_{xy}^{(i)}$ is the weight of the edge (x, y) at iteration i , $\kappa_{xy}^{(i)}$ is the Ollivier Ricci curvature at iteration i , and $d_{xy}^{(i)}$ is the distance between x and y at iteration i (itself inferred from the current set of edge weights). A small parameter η may be included to control the step size. Some versions set $\eta = 1$ (or a small fraction of 1) to maintain stability. Edges that exhibit large positive curvature shrink in length, whereas edges that exhibit large negative curvature expand. Over several iterations, clusters of nodes with strong internal connections typically collapse, whereas the bridging edges between separate clusters get longer.

Mathematically, one can think of this process as a discrete approximation of smoothing out highly curved regions, akin to the continuous case in which narrower “necks” get pinched, dividing the manifold into multiple simpler components. In networks, these necks correspond to edges that connect otherwise densely connected subgraphs. Eventually, to isolate these communities rigorously, one can delete or “cut” edges that have become elongated beyond some threshold.

40 | Application to Complex Networks

1 While the process is not exactly identical to Hamilton’s PDE-based Ricci Flow,
2 it follows the same fundamental principle: curvature is a driver that changes local
3 distances over time. Practical heuristics include stopping after a fixed number of
4 iterations or until some measure of curvature or partitioning becomes stable.

5 In summary, the Ricci Flow approach in differential geometry has deep and pro-
6 found implications for how we dissect and understand global properties of curved
7 spaces. By translating it into the language of optimal transport and Ollivier Ricci
8 curvature, we gain a discrete tool that can be applied to purely combinatorial graphs
9 or other abstract metric spaces. This theoretical backdrop provides the motivation for
10 analyzing complex networks—be they social, biological, informational, or otherwise—
11 in a geometric framework, where communities are interpreted as regions of high
12 effective curvature.

13 The next section examines how this perspective concretely applies to large-scale
14 networks, and how the *Adjusted Rand Index (ARI)* and *modularity* come into play as
15 metrics to gauge the performance of Ricci Flow-based community detection. We also
16 clarify how the notion of a “cutting point” or “surgery” in the discrete flow context
17 helps isolate the fundamental structural blocks of a network.

CHAPTER 5

THE DEVELOPED CODE

The objectives of the developed code were

- **Implementation of a Ricci Flow method** able to evaluate the Ollivier-Ricci curvature of a given graph and update its edges' weights accordingly. Then we wanted the method to perform surgery on weakly connected edges to allow for community detection.

For these purposes we relied on Ollivier-Ricci library developed by Ni et al. [8], see [section 5.1](#).

- **Plotting graphs and communities** to appreciate the behavior of the method and obtain graphical results.
- **Testing the method** on synthetic graphs, trying to benchmark with an analogue test made by Ni et al. [7]. Results and further details are given in [section 5.2](#).
- **Evaluation of performances** by comparing the method with other commonly used community detection methods.
- **Application on real world data.** To do this we chose Zachary's Karate Club graph, which is directly accessible from Networkx library. Results and further details are given in [section 5.3](#).

In the this chapter we present the main ideas and tools related to the code. To get more insights on how the code as been constructed and subdivided (i.e. various classes and functions) we recommend consulting [CodeDocumentation](#) or the GitHub of the whole project: [RicciFlowNetwork](#).

42 | The Developed Code
1 5.1 The setup

2 To facilitate the computation of Ricci curvature within networks, we made use of
 3 the GraphRicciCurvature library as our starting point. This Python library is part of
 4 a more comprehensive library on Ricci curvature for networks. The latter provides
 5 tools to compute two discrete Ricci curvatures: Ollivier-Ricci and Forman-Ricci; it
 6 supports the analysis of both weighted and unweighted graphs. In addition, it offers
 7 basic methods for graph surgery and evaluation of the *adjusted rand index* (ARI).

8 Each graph we employed was generated with Networkx library. We also used its
 9 methods as a starting point for graphical representations. For plotting we implemented
 10 GraphDrawer class which, among other functionalities, allows us to see the detected
 11 communities separated in subgraphs. Nodes are colored with their corresponding
 12 community color, giving a visual indication of method's performance.

13 We used this setup to implement the code workflow depicted in fig. 5.1.

14 5.2 Tests on Synthetic Graphs
15 5.2.1 Stochastic Block Model Test Graph

16 To start testing our code we choose a Stochastic Block Model (SBM) graph. A SBM
 17 graph is a type of random graph model used to generate networks with a predefined
 18 community structure. It is an extension of the Erdos–Renyi, where nodes are divided
 19 into communities, and the probability of an edge between two nodes depends on
 20 their community membership. It is characterized by N nodes, k communities and a
 21 connectivity matrix P_{ij} which represents the probability of an edge between a node in
 22 community i and a node in community j . Since communities are predefined when
 23 generating the graph, we can directly compare the detected communities with the true
 24 ones.

25 For our test we set

$$26 \quad N = 500, \quad k = 2, \quad P = \begin{bmatrix} 0.20 & 0.03 \\ 0.03 & 0.20 \end{bmatrix}$$

27 with every initial weight equal to one.

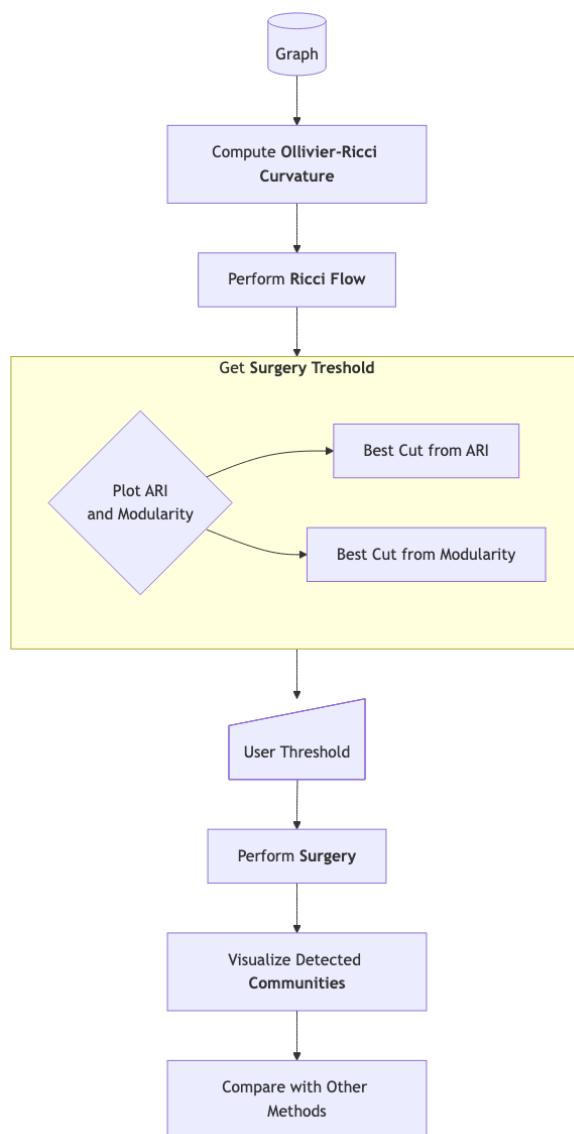


Figure 5.1 Code workflow.

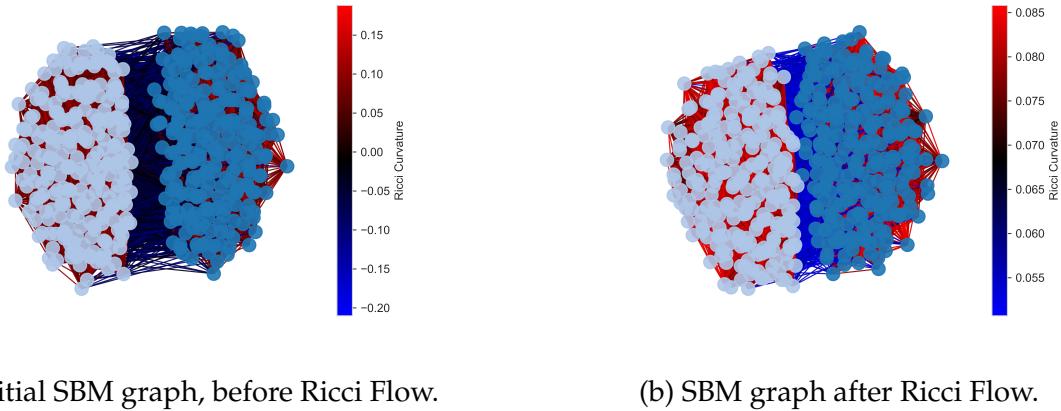


Figure 5.2 Comparison of SBM graph before and after having applied Ricci Flow on edges.

¹ **5.2.2 Lancichinetti-Fortunato-Radicchi Test Graph**

² Lancichinetti-Fortunato-Radicchi (LFR)

³ **5.2.3 Further Tests on Lancichinetti-Fortunato-Radicchi Graphs**

⁴ **5.3 Application to Zachary's Karate Club Graph**

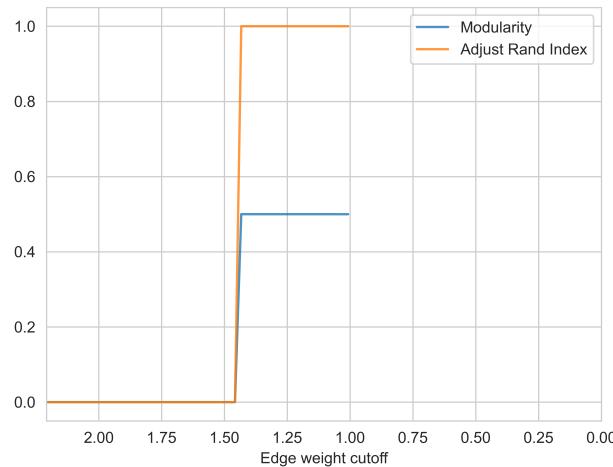
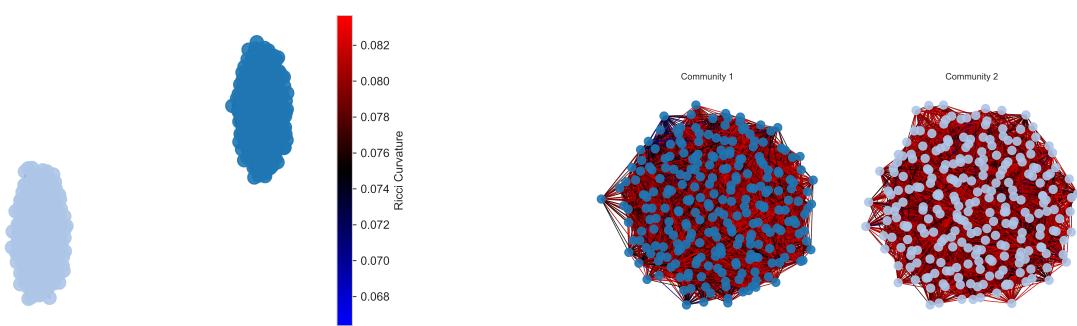


Figure 5.3 SBM acc

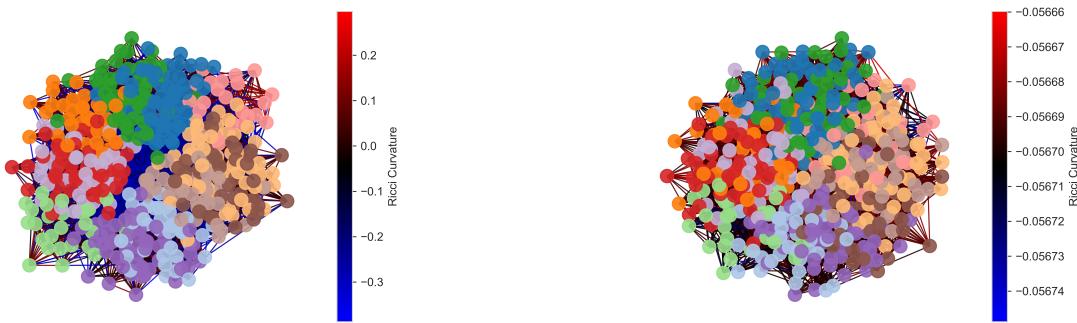


(a) Final SBM graph, after surgery process.

(b) Detected communities after surgery on SBM graph.

Figure 5.4 Comparison of SBM graph after surgery and corresponding connected components (i.e. the detected communities).

46 | The Developed Code



(a) Initial LFR graph, before Ricci Flow.

(b) LFR graph after Ricci Flow.

Figure 5.5 Comparison of LFR graph before and after having applied Ricci Flow on edges.

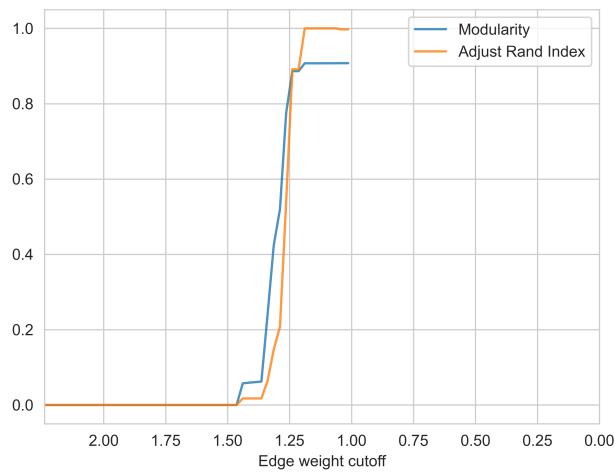


Figure 5.6 LFR acc

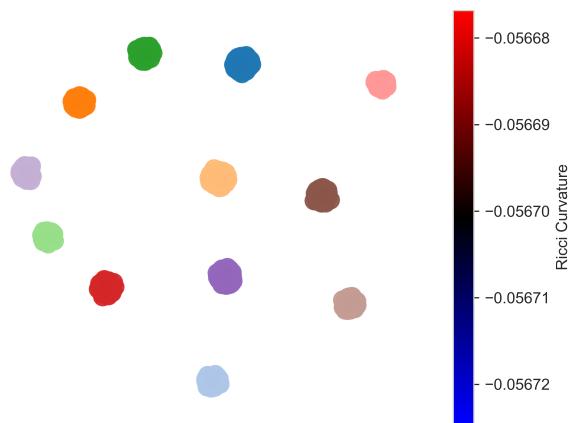


Figure 5.7 Final LFR graph, after surgery process.

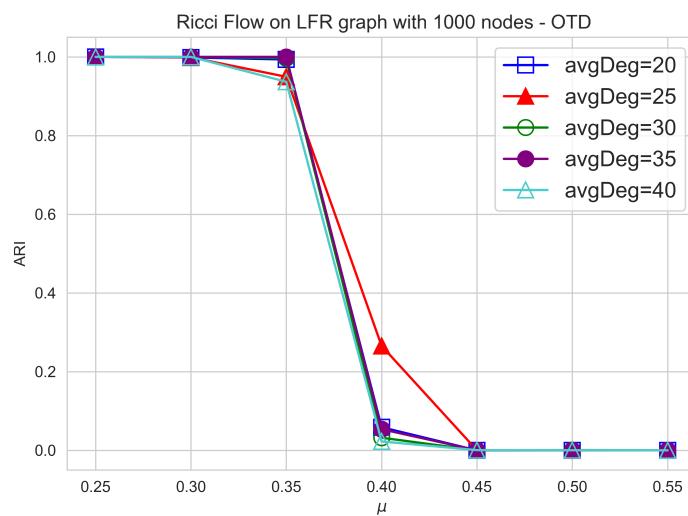


Figure 5.8 LFR OTD

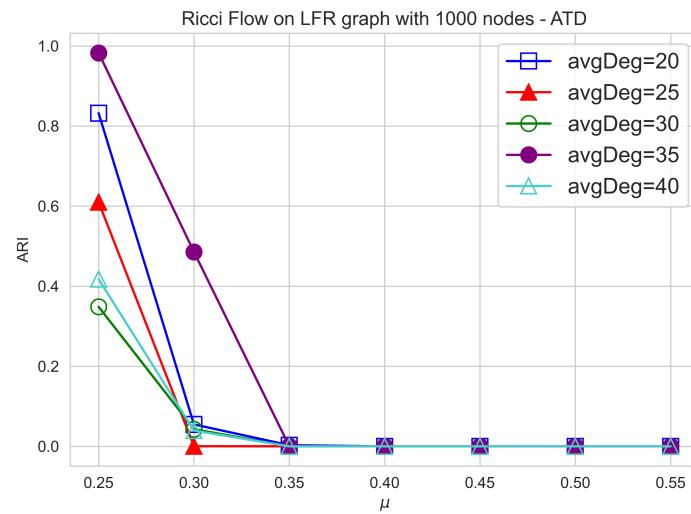
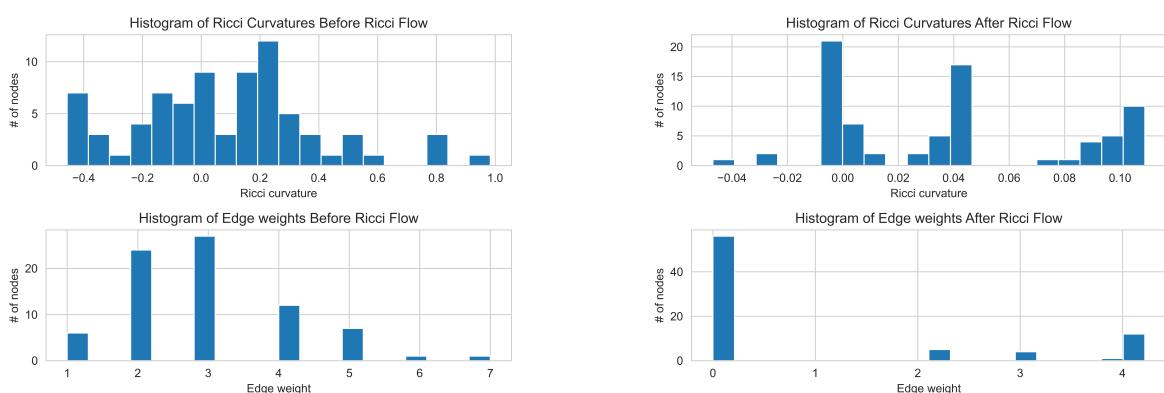


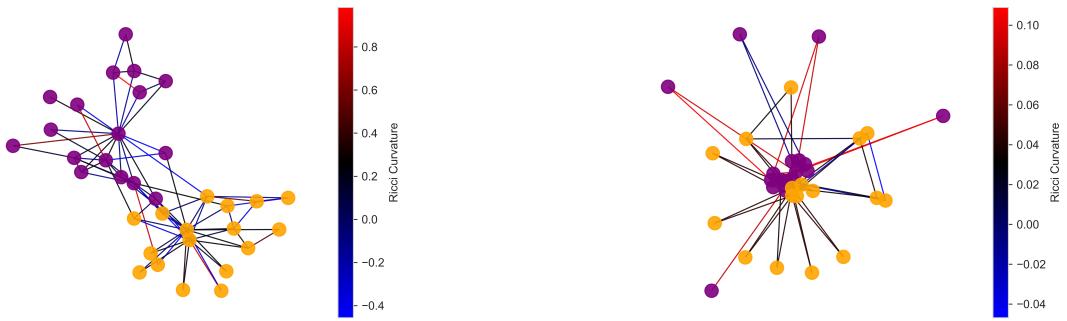
Figure 5.9 LFR ATD



(a) Initial Karate graph, before Ricci Flow.

(b) Karate graph after Ricci Flow.

Figure 5.10 Comparison of curvature values and weights before and after having applied Ricci Flow on edges.



(a) Initial Karate graph, before Ricci Flow.

(b) Karate graph after Ricci Flow.

Figure 5.11 Comparison of Karate graph before and after having applied Ricci Flow on edges.

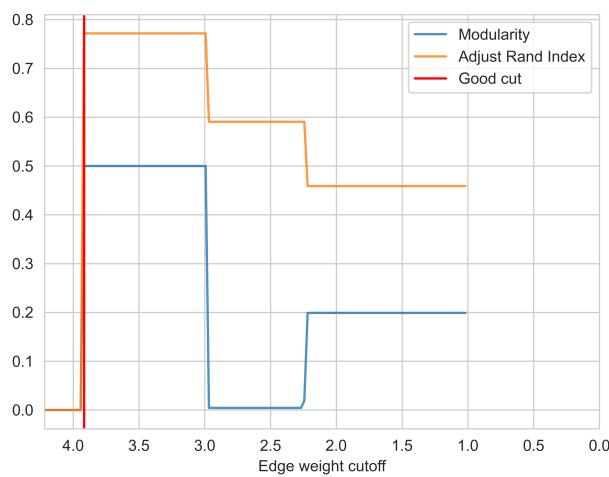
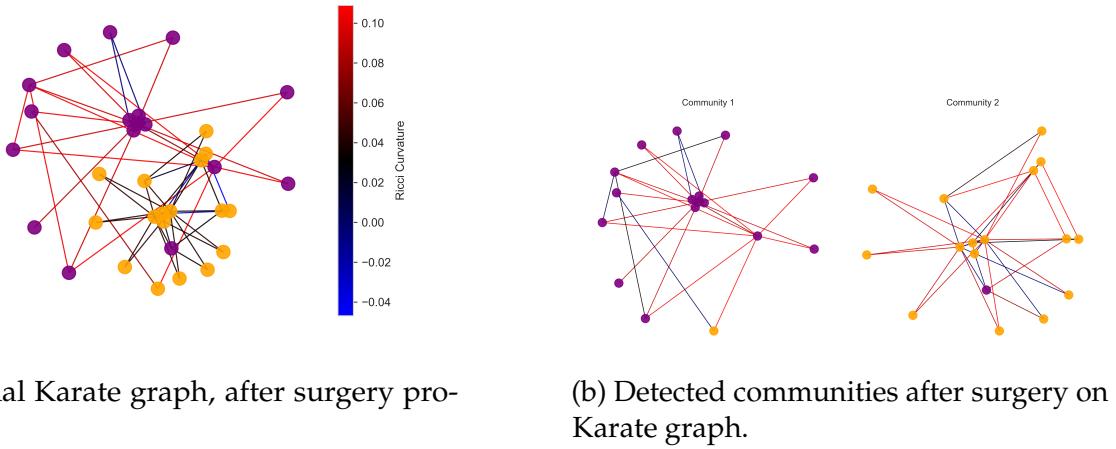


Figure 5.12 Karate acc



(a) Final Karate graph, after surgery process.

(b) Detected communities after surgery on Karate graph.

Figure 5.13 Comparison of Karate graph after surgery and corrensponding connected components (i.e. the detected communities).

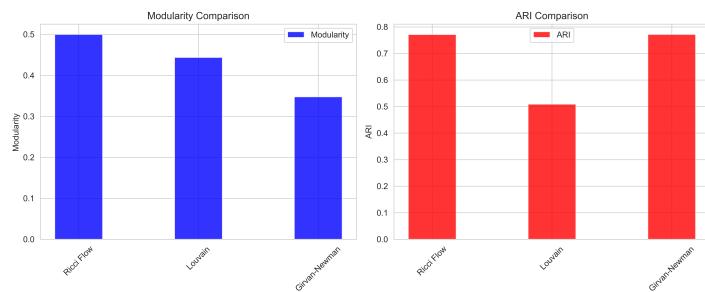


Figure 5.14 Karate acc

CHAPTER 6

CONCLUSIONS AND FUTURE DIRECTIONS

Draft - v1.0

February 27, 2025 – 08:51

APPENDIX A

METRIC OF A 2-SPHERE

Draft - v1.0

February 27, 2025 – 08:51

REFERENCES

-
- [1] Richard H Bamler. “Recent developments in Ricci flows”. In: (2021). arXiv: 2102.12615 [math.DG]. URL: <https://arxiv.org/abs/2102.12615>. 1
- [2] Mauro Carfora et al. “Ricci Flow from the Renormalization of Nonlinear Sigma Models in the Framework of Euclidean Algebraic Quantum Field Theory”. In: (2019). arXiv: 1809.07652 [math-ph]. URL: <https://arxiv.org/abs/1809.07652v2>. 1
- [3] B. Chow and D. Knopf. *The Ricci Flow: An Introduction*. American Mathematical Society, 2004. ISBN: 0-8218-3515-7. 1
- [4] Martin D. Crossley. *Essential Topology*. Springer, 2005. DOI: [10.1007/1-84628-194-6](https://doi.org/10.1007/1-84628-194-6). 1
- [5] Lee J. M. *Introduction to Riemannian Manifolds*. Second Edition. Springer, 2018. DOI: [10.1007/978-3-319-91755-9](https://doi.org/10.1007/978-3-319-91755-9). 1
- [6] Lee J. M. *Introduction to Smooth Manifolds*. Second Edition. Springer, 2013. DOI: [10.1007/978-981-99-0565-2](https://doi.org/10.1007/978-981-99-0565-2). 1
- [7] Chien-Chun Ni et al. “Community Detection on Networks with Ricci Flow”. In: (2019). arXiv: 1907.03993 [cs.SI]. URL: <https://arxiv.org/abs/1907.03993>. 1
- [8] Barrett S. Ni C.C. Cruceru C. *GraphRicciCurvature*. <https://github.com/saibalmars/GraphRicciCurvature.git>. 2019. 1
- [9] Henri Poincaré. “Analysis Situs”. Trans. by John Stillwell. In: *Journal de l’École Polytechnique* 2 (1895). 1
- [10] Henri Poincaré. “Cinquième complément à l’analysis situs”. Fifth Complement to Analysis Situs. Trans. by John Stillwell. In: *Rendiconti del Circolo Matematico di Palermo* 18 (1904 2008), pp. 45–110. 1
- [11] J. Polchinski. *String theory. Vol. 1: An introduction to the bosonic string*. Cambridge Monographs on Mathematical Physics. Cambridge University Press, Dec. 2007. DOI: [10.1017/CBO9780511816079](https://doi.org/10.1017/CBO9780511816079). 1
- [12] Ryan A. Rossi and Nesreen K. Ahmed. “The Network Data Repository with Interactive Graph Analytics and Visualization”. In: AAAI. 2015. URL: <https://networkrepository.com>. 1
- [13] Loring W. Tu. *An Introduction to Manifolds*. Second Edition. Springer, 2011. DOI: [10.1007/978-1-4419-7400-6](https://doi.org/10.1007/978-1-4419-7400-6). 1
- [14] Timo Weigand. “Introduction to String Theory”. Lecture Notes. 2012. 1

56 | References

- ¹ [15] Wayne W. Zachary. “An Information Flow Model for Conflict and Fission in
² Small Groups”. In: *The University of Chicago Press Journals* (1997). URL: <https://doi.org/10.1086/jar.33.4.3629752>.
³