A Riemannian Perspective on Optimization Problems in Markov Chains

Fabio Durastante

CORMSIS Seminar, University of Southampton, 19 February 2026

Dipartimento di Matematica
Università di Pisa

With a little help from my friends

1 Collaborators and Fundings

Miryam Gnazzo
ISTI-CNR, Italy

Beatrice Meini
DM-PISA, Italy

This research was partially granted by the MUR through the PRIN 2022

- "MOLE: Manifold constrained Optimization and LEarning", code: 2022ZK5ME7 (CUP B53C24006410006).
- "Low-rank Structures and Numerical Methods in Matrix and Tensor Computations and their Application", code: 20227PCCKZ (CUP I53D23002280006).

Table of Contents

Markov Chains: definition

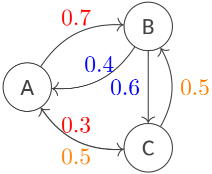
A Markov chain is a stochastic process where the next state depends only on the current state, *not* on the *past history*.

Markov Property:

$$P(X_{n+1} = j \mid X_n = i, X_{n-1}, \dots, X_0) = P(X_{n+1} = j \mid X_n = i)$$

Transition Matrix: $P \in \mathbb{R}^{m \times m}$ where

- $P_{ij} \geqslant 0$ for all i, j
- $\sum_{j=1}^{m} P_{ij} = 1$ (row stochastic)



Evolution: $\pi_{n+1}^\top = \pi_n^\top P$ where π_n is the **probability distribution** at step n.

Markov Chains: definition

A Markov chain is a stochastic process where the next state depends only on the current state, *not* on the *past history*.

Markov Property:

$$P(X_{n+1} = j \mid X_n = i, X_{n-1}, \ldots, X_0) = P(X_{n+1} = j \mid X_n = i)$$

Transition Matrix: $P \in \mathbb{R}^{m \times m}$ where

- $P_{ij} \geqslant 0$ for all i, j
- $\sum_{j=1}^{m} P_{ij} = 1$ (row stochastic)

$$P = \begin{bmatrix} 0 & 0.7 & 0.3 \\ 0.4 & 0 & 0.6 \\ 0.5 & 0.5 & 0 \end{bmatrix}$$

Evolution: $\pi_{n+1}^\top = \pi_n^\top P$ where π_n is the **probability distribution** at step n.

Applications and Properties

Common Applications:

- PageRank algorithm
- Queueing systems
- Population dynamics
- Molecular dynamics
- Financial modeling

Properties:

Irreducibility: All states communicate, i.e., $\forall\, i, j, \exists\, n : (P^n)_{ij} > 0$

Aperiodicity: No cyclic patterns

Stationary Distribution: $\boldsymbol{\pi}^* = \boldsymbol{\pi}^* P$ (if it exists)

Reversibility: $\pi_i P_{ij} = \pi_j P_{ji}$ for all i, j (detailed balance)

Under *mild conditions*, $\lim_{n\to\infty} \boldsymbol{\pi}_n = \boldsymbol{\pi}^* \geqslant 0$ regardless of initial distribution.

Applications and Properties

Common Applications:

- PageRank algorithm
- Queueing systems
- Population dynamics
- Molecular dynamics
- Financial modeling

? But how do we obtain them ?

Transition matrices are often derived from data (e.g., observed transitions), hence we may know that they should satisfy certain properties but **cannot guarantee them**.

Properties:

Irreducibility: All states communicate, i.e., $\forall\, i, j, \exists\, n : (P^n)_{ij} > 0$

Aperiodicity: No cyclic patterns

Stationary Distribution: $\pi^* = \pi^* P$ (if it exists)

Reversibility: $\pi_i P_{ij} = \pi_j P_{ji}$ for all i, j (detailed balance)

Under *mild conditions*, $\lim_{n\to\infty} \pi_n = \pi^* \geqslant 0$ regardless of initial distribution.

An **example** consider the *crediting rating* of a Country:

> *“Credit ratings are the evaluation of the credit risk of a prospective debtor (a government), predicting their ability to pay back the debt, and an implicit forecast of the likelihood of the debtor defaulting.”*

Credit Rating and Embeddability

An **example** consider the *crediting rating* of a Country:

	AAA	AA	A	BBB	BB	B	CCC	D
AAA	0,891	0,0963	0,0078	0,0019	0,003	0	0	0
AA	0,0086	0,901	0,0747	0,0099	0,0029	0,0029	0	0
A	0,0009	0,0291	0,8894	0,0649	0,0101	0,0045	0	0,0009
BBB	0,0006	0,0043	0,0656	0,8427	0,0644	0,016	0,0018	0,0045
BB	0,0004	0,0022	0,0079	0,0719	0,7764	0,1043	0,0127	0,0241
B	0	0,0019	0,0031	0,0066	0,0517	0,8246	0,0435	0,0685
CCC	0	0	0,0116	0,0116	0,0203	0,0754	0,6493	0,2319
D	0	0	0	0	0	0	0	1

The **transition matrix** is estimated from data sampled at yearly/semi-yearly intervals, but I would like to **know before the next update if I need to sell** (or maybe **buy**).

The **idea** is that:

*If one step represents a **year**, then half-a-step represents six months, and **quarter of a step** represents three months...*

$$\pi_{n/2+1/2}^{\top} = \pi_{n/2}^{\top} P^{1/2} \quad \text{or} \quad \pi_{n/4+1/4}^{\top} = \pi_{n/4}^{\top} P^{1/4}$$

However $\sqrt[q]{P}$ is **not necessarily a transition matrix in general!**

All the possible things that can go wrong go wrong as shown in:

N. J. Higham and L. Lin, On pth roots of stochastic matrices, Linear Algebra Appl. **435** (2011), no. 3, 448–463.

The **transition matrix** is estimated from data sampled at yearly/semi-yearly intervals, but I would like to **know before the next update if I need to sell** (or maybe **buy**).

The **idea** is that:

*If one step represents a **year**, then half-a-step represents six months, and **quarter of a step** represents three months...*

$$\pi^\top_{n/2+1/2} = \pi^\top_{n/2} P^{1/2} \quad \text{or} \quad \pi^\top_{n/4+1/4} = \pi^\top_{n/4} P^{1/4}$$

However $\sqrt[q]{P}$ is **not necessarily a transition matrix in general!**

All the possible things that can go wrong go wrong as shown in:

N. J. Higham and L. Lin, On pth roots of stochastic matrices, Linear Algebra Appl. **435** (2011), no. 3, 448–463.

We **look for an approximation** X s.t. $X^q \approx P$ and X is a transition matrix, i.e., we solve:

$$\min_{X \in \mathcal{S}} \|X^q - P\|_F^2, \qquad \mathcal{S} = \{X \in \mathbb{R}^{m\times m} : X_{ij} \geqslant 0, X\mathbf{1} = \mathbf{1}, \boldsymbol{\pi}^\top X = \boldsymbol{\pi}^\top\}$$

Nearest Reversible Markov Chain

Consider a physical system modeled by the Langevin stochastic differential equation:

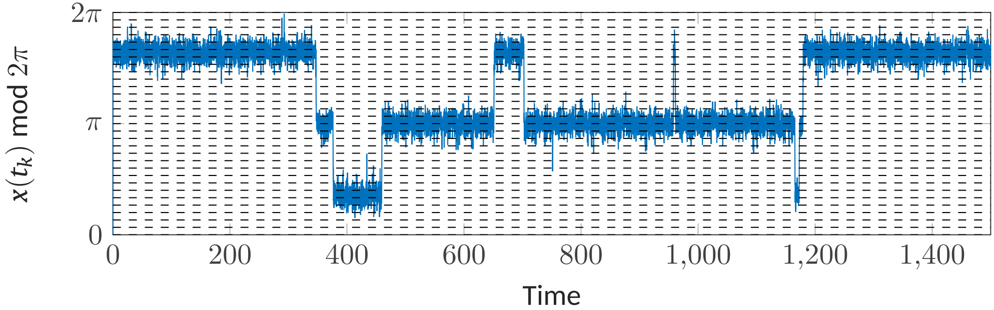
$$\dot{x} = -\frac{\partial U(x)}{\partial x} + \xi(t), \qquad x \in \mathbb{R},$$

where $\xi(t)$ is Gaussian white noise with $\langle \xi(t)\, \xi(s) \rangle = \sigma^2\, \delta(t-s)$.

Discretization: Using the Euler–Maruyama scheme with time step Δt:

$$x(t+\Delta t) = x(t) - \frac{\partial U(x(t))}{\partial x}\, \Delta t + \sigma \sqrt{\Delta t}\, \eta(t),$$

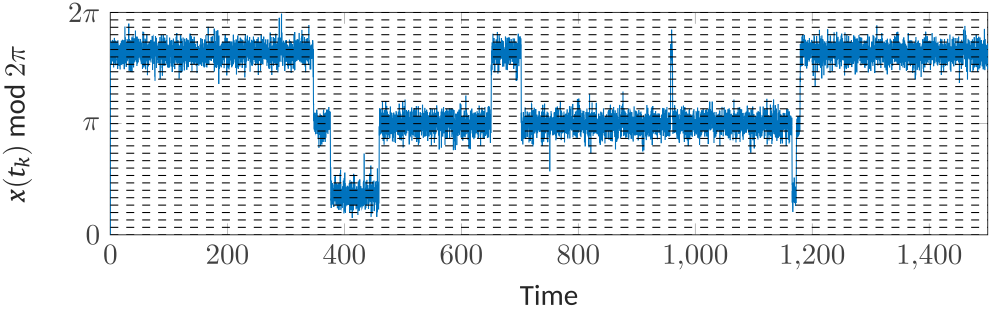
Nearest Reversible Markov Chain



State Space Discretization:

- Partition into M disjoint regions $\{\Omega_i\}_{i=1}^{M}$
- $C_{ij} = \sharp\{$ transitions $\Omega_i \to \Omega_j\}$
- Estimate: $P_{ij} = \frac{C_{ij}}{\sum_{k=1}^{M} C_{ik}}$

Nearest Reversible Markov Chain



State Space Discretization:

- Partition into M disjoint regions $\{\Omega_i\}_{i=1}^{M}$
- C_{ij} = $\sharp\{$ transitions $\Omega_i \to \Omega_j\}$
- Estimate: $P_{ij} = \frac{C_{ij}}{\sum_{k=1}^{M} C_{ik}}$

The Problem:

- Finite sampling breaks detailed balance
- Need reversibilization: find X s.t.

$$\min_{X \in \mathcal{S}_{\text{rev}}} \|X - P\|_F^2$$

Minimizing Kemeny's Constant

Kemeny's Constant: A fundamental quantity in Markov chain theory measuring the **expected number of steps** to reach a random state from a given state.

Definition:

$$\kappa(P) = \sum_{j=1}^{m} \pi_j (I - P + \mathbf{1}\boldsymbol{\pi}^\top)^{-1}_{ij}$$

for any i (independent of initial state).

Properties:

- Depends only on P and $\boldsymbol{\pi}^*$
- Measures mixing speed
- Lower κ = faster convergence

The Problem:

- Given a *transition matrix* P
- Find optimal Δ to minimize

$$\min_{\Delta \in \mathcal{A}} \kappa(P + \Delta) + \lambda \|\Delta\|_F^2$$

where $\mathcal{A}$ preserves stochasticity

Goal: Speed up convergence to equilibrium via controlled modifications.

Constrained Optimization?

All the problems we have seen so far can be cast as **constrained optimization problems**:

$$\min_{X \in \mathcal{S}} f(X)$$

where $\mathcal{S}$ is a **constraint set** built of stochastic matrices with certain additional properties:

- Sharing the same stationary distribution π,
- Being reversible w.r.t. π,

or of **admissible perturbations** Δ of a given transition matrix P.

While f is a **smooth** function:

- $f(X) = \|X^q - P\|_F^2$, (qth root) $f(X) = \|X - P\|_F^2$ (reversible approximation),
- $f(\Delta) = \operatorname{tr}((I - (P + \Delta) + \mathbf{1}\boldsymbol{\pi}^\top)^{-1}) + \|\Delta\|_F^2$ (Kemeny's constant).

Constrained Optimization?

All the problems we have seen so far can be cast as **constrained optimization problems**:

$$\min_{X \in \mathcal{S}} f(X)$$

where $\mathcal{S}$ is a **constraint set** built of stochastic matrices with certain additional properties:

- Sharing the same stationary distribution π,
- Being reversible w.r.t. π,

or of **admissible perturbations** Δ of a given transition matrix P.

While f is a **smooth** function:

- $f(X) = \|X^q - P\|_F^2$, (qth root) $f(X) = \|X - P\|_F^2$ (reversible approximation),
- $f(\Delta) = \operatorname{tr}((I - (P + \Delta) + \mathbf{1}\boldsymbol{\pi}^\top)^{-1}) + \|\Delta\|_F^2$ (Kemeny's constant).

We **exploit the structure** of $\mathcal{S}$ to design efficient algorithms.

Table of Contents

3 Riemannian Optimization

Riemannian Optimization: Main Idea

Classical Constrained Optimization:

- Constraints are treated as *penalties* or *barriers*
- Algorithms must handle infeasibility carefully
- Convergence analysis becomes intricate

Riemannian Optimization: Main Idea

Riemannian Perspective:

Treat the constraint set $\mathcal{S}$ as a ***smooth manifold*** *$\mathcal{M}$ equipped with a Riemannian metric.*

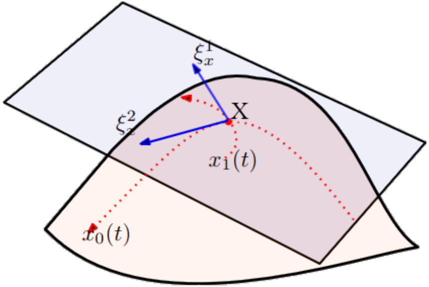
Key Advantages:

- Optimize *on* the manifold, not *around* it
- Leverage geometric structure for better algorithms
- Natural handling of constraints via the metric
- Cleaner convergence theory

Euclidean View:
Constraint set as obstacle

Riemannian View:
Constraint set as the natural domain

Ingredients: Tangent Spaces and Retractions



Tangent Space $\mathcal{T}_X\mathcal{M}$:

- Directions along which we can move while staying on $\mathcal{M}$
- Equipped with an inner product $\langle \cdot, \cdot \rangle_X$
- Enables computing gradients *intrinsic* to the manifold

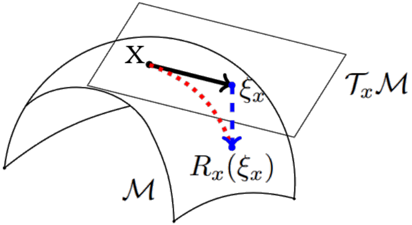
Retraction $R_X(\eta)$:

- Maps a tangent vector $\eta \in \mathcal{T}_X\mathcal{M}$ back to the manifold
- Plays the role of an exponential map (locally)
- Ensures iterates stay feasible: $X_{k+1} = R_X(\eta)$

Ingredients: Tangent Spaces and Retractions

Tangent Space $\mathcal{T}_X\mathcal{M}$:

- Directions along which we can move while staying on $\mathcal{M}$
- Equipped with an inner product $\langle \cdot, \cdot \rangle_X$
- Enables computing gradients *intrinsic* to the manifold



Retraction $R_X(\eta)$:

- Maps a tangent vector $\eta \in \mathcal{T}_X\mathcal{M}$ back to the manifold
- Plays the role of an exponential map (locally)
- Ensures iterates stay feasible: $X_{k+1} = R_X(\eta)$

Levi-Civita Connection

To build a second-order geometry, and hence **second-order optimization** algorithms, we need to define the notion of **connection** on the manifold $\mathcal{M}$.

Levi-Civita Connection $\nabla : \mathcal{T}_X\mathcal{M} \times \mathcal{T}_X\mathcal{M} \to \mathcal{T}_X\mathcal{M}$:

- A way to differentiate vector fields along curves on the manifold
- Preserves the metric (metric compatibility)
- Has no torsion (symmetry)

Riemannian Gradient: The gradient of f at X is the unique vector $\operatorname{grad} f(X) \in \mathcal{T}_X\mathcal{M}$ such that

$$\langle \operatorname{grad} f(X), \eta \rangle_X = \mathrm{D}f(X)[\eta] \quad \forall \eta \in \mathcal{T}_X\mathcal{M}$$

where $\mathrm{D}f(X)[\eta]$ is the directional derivative of f at X in the direction η.

Levi-Civita Connection

To build a second-order geometry, and hence **second-order optimization** algorithms, we need to define the notion of **connection** on the manifold $\mathcal{M}$.

Levi-Civita Connection $\nabla : \mathcal{T}_X\mathcal{M} \times \mathcal{T}_X\mathcal{M} \to \mathcal{T}_X\mathcal{M}$**:**

- A way to differentiate vector fields along curves on the manifold
- Preserves the metric (metric compatibility)
- Has no torsion (symmetry)

Riemannian Hessian: The Hessian of f at X is the linear operator $\mathrm{hess}\, f(X) : \mathcal{T}_X\mathcal{M} \to \mathcal{T}_X\mathcal{M}$ defined by

$$\mathrm{hess}\, f(X)[\eta] = \nabla_\eta \,\mathrm{grad}\, f(X)$$

where ∇_η denotes the *application of the Levi-Civita connection* to the vector field $\mathrm{grad}\, f$ in the direction η.

Algorithm: Riemannian Newton Method

Require: $X_0 \in \mathcal{M}$, tolerance $\epsilon > 0$

$k \leftarrow 0$

while $\|\operatorname{grad} f(X_k)\| > \epsilon$ **do**

- Compute $\operatorname{grad} f(X_k)$
- Compute $\operatorname{hess} f(X_k)$
- Solve $\operatorname{hess} f(X_k)[\eta_k] = -\operatorname{grad} f(X_k)$
- $X_{k+1} \leftarrow R_{X_k}(\eta_k)$
- $k \leftarrow k + 1$

Ensure: X_k

Convergence: Superlinear (or quadratic under regularity conditions)

Required Subroutines:

Metric	Inner product $\langle \cdot, \cdot \rangle_X$ on $\mathcal{T}_X\mathcal{M}$
Tangent Space	Characterization of $\mathcal{T}_X\mathcal{M}$
Retraction	Map $R_X(\eta)$ back to $\mathcal{M}$
Gradient	Compute $\operatorname{grad} f(X)$
Hessian	Compute $\operatorname{hess} f(X)[\eta]$
Connection	Levi-Civita connection $\nabla_\eta v$ for $v \in \mathcal{T}_X\mathcal{M}$
Linear Solve	Solve $\operatorname{hess} f(X_k)[\eta] = -\operatorname{grad} f(X_k)$

Algorithm: Riemannian Newton Method

Require: $X_0 \in \mathcal{M}$, tolerance $\epsilon > 0$

$k \leftarrow 0$

while $\|\operatorname{grad} f(X_k)\| > \epsilon$ **do**

- Compute $\operatorname{grad} f(X_k)$
- Compute $\operatorname{hess} f(X_k)$
- Solve $\operatorname{hess} f(X_k)[\eta_k] = -\operatorname{grad} f(X_k)$
- $X_{k+1} \leftarrow R_{X_k}(\eta_k)$
- $k \leftarrow k + 1$

Ensure: X_k

Convergence: Superlinear (or quadratic under regularity conditions)

Required Subroutines:

Metric	Inner product $\langle \cdot, \cdot \rangle_X$ on $\mathcal{T}_X\mathcal{M}$
Tangent Space	Characterization of $\mathcal{T}_X\mathcal{M}$
Retraction	Map $R_X(\eta)$ back to $\mathcal{M}$
Gradient	Compute $\operatorname{grad} f(X)$
Hessian	Compute $\operatorname{hess} f(X)[\eta]$
Connection	Levi-Civita connection $\nabla_\eta v$ for $v \in \mathcal{T}_X\mathcal{M}$
Linear Solve	Solve $\operatorname{hess} f(X_k)[\eta] = -\operatorname{grad} f(X_k)$

For any manifold, we need to **derive these components** to implement a second-order Riemannian method.

The multinomial manifold

The **standard construction** for the manifold of stochastic matrices has been introduced in

- A. Douik and B. Hassibi, Manifold optimization over the set of doubly stochastic matrices: a second-order geometry, IEEE Trans. Signal Process. **67** (2019), no. 22, 5761–5774.

It uses the **Fisher information metric**:

$$\langle \eta, \xi \rangle_P = \sum_{i,j} \frac{\eta_{ij}\xi_{ij}}{P_{ij}} = \mathrm{tr}((\eta \oslash P)\xi^\top)$$

Natural for **probability distributions**, but requires $P > 0$ (strictly positive)

The multinomial manifold

The **standard construction** for the manifold of stochastic matrices has been introduced in

- A. Douik and B. Hassibi, Manifold optimization over the set of doubly stochastic matrices: a second-order geometry, IEEE Trans. Signal Process. **67** (2019), no. 22, 5761–5774.

Multinomial manifold (strict simplex elements)

$$\{X \in \mathbb{R}^{n\times m} : X_{ij} > 0 \ \forall i,j \text{ and } X^\top \mathbf{1}_m = \mathbf{1}_n\}$$

Multinomial doubly stochastic manifold

$$\{X \in \mathbb{R}^{n\times n} : X_{ij} > 0 \ \forall i,j \text{ and } X\mathbf{1}_n = \mathbf{1}_n, X^\top \mathbf{1}_n = \mathbf{1}_n\}$$

Multinomial symmetric and stochastic manifold

$$\{X \in \mathbb{R}^{n\times n} : X_{ij} > 0 \ \forall i,j \text{ and } X\mathbf{1}_n = \mathbf{1}_n, X = X^\top\}$$

The multinomial manifold

The **standard construction** for the manifold of stochastic matrices has been introduced in

A. Douik and B. Hassibi, Manifold optimization over the set of doubly stochastic matrices: a second-order geometry, IEEE Trans. Signal Process. **67** (2019), no. 22, 5761–5774.

Multinomial manifold (strict simplex elements)

$$\{X \in \mathbb{R}^{n\times m} : X_{ij} > 0 \ \forall i,j \text{ and } X^\top \mathbf{1}_m = \mathbf{1}_n\}$$

Multinomial doubly stochastic manifold

$$\{X \in \mathbb{R}^{n\times n} : X_{ij} > 0 \ \forall i,j \text{ and } X\mathbf{1}_n = \mathbf{1}_n, X^\top \mathbf{1}_n = \mathbf{1}_n\}$$

Multinomial symmetric and stochastic manifold

$$\{X \in \mathbb{R}^{n\times n} : X_{ij} > 0 \ \forall i,j \text{ and } X\mathbf{1}_n = \mathbf{1}_n, X = X^\top\}$$

We extend the construction to the manifold we need for our three problems.

A Riemannian Manifold with prescribed stationary vector

A new Riemannian Manifold

Let $\boldsymbol{\pi} \in \mathbb{R}^n$ be a positive vector such that $\boldsymbol{\pi}^T\mathbf{1} = 1$, and define the set

$$\mathbb{S}_n^{\boldsymbol{\pi}} = \{S \in \mathbb{R}^{n\times n} \; : \; S\mathbf{1} = \mathbf{1},\; \boldsymbol{\pi}^T S = \boldsymbol{\pi}^T,\; S > 0\}.$$

- $\mathbb{S}_n^{\boldsymbol{\pi}}$ is an **embedded manifold** of $\mathbb{R}^{n\times n}$ of dimension $(n-1)^2$, since it is indeed generated by $2n-1$ linearly independent equations.

A Riemannian Manifold with prescribed stationary vector

A new Riemannian Manifold

Let $\pi \in \mathbb{R}^n$ be a positive vector such that $\pi^T \mathbf{1} = 1$, and define the set

$$\mathbb{S}_n^{\pi} = \{S \in \mathbb{R}^{n\times n} \ : \ S\mathbf{1} = \mathbf{1},\ \pi^T S = \pi^T,\ S > 0\}.$$

- $\mathbb{S}_n^{\pi}$ is an **embedded manifold** of $\mathbb{R}^{n\times n}$ of dimension $(n-1)^2$, since it is indeed generated by $2n-1$ linearly independent equations.
- We **implemented all the operations** required to run Riemannian optimization algorithms in the MANOPT toolbox:
  - Boumal, N., Mishra, B., Absil, P. A., & Sepulchre, R. (2014). Manopt, a Matlab toolbox for optimization on manifolds. The Journal of Machine Learning Research, 15(1), 1455-1459.

Tangent space, metric and projections

Lemma (D., Meini, 2024)

The **tangent space** to $\mathbb{S}_n^{\pi}$ at $S \in \mathbb{S}_n^{\pi}$ is $\mathcal{T}_S \mathbb{S}_n^{\pi} = \{\xi_S \in \mathbb{R}^{n \times n} \ : \ \xi_S \mathbf{1} = \mathbf{0}, \ \boldsymbol{\pi}^T \xi_S = \mathbf{0}\}$.

Tangent space, metric and projections

Lemma (D., Meini, 2024)

The **tangent space** to $\mathbb{S}_n^\pi$ at $S \in \mathbb{S}_n^\pi$ is $\mathcal{T}_S\mathbb{S}_n^\pi = \{\xi_S \in \mathbb{R}^{n\times n} \;:\; \xi_S\mathbf{1} = \mathbf{0},\; \boldsymbol{\pi}^T\xi_S = \mathbf{0}\}$.

- $\mathbb{S}_n^\pi$, endowed with the **Fisher metric** is a **Riemannian manifold**.

Tangent space, metric and projections

Lemma (D., Meini, 2024)

The **tangent space** to $\mathbb{S}_n^{\pi}$ at $S \in \mathbb{S}_n^{\pi}$ is $\mathcal{T}_S\mathbb{S}_n^{\pi} = \{\xi_S \in \mathbb{R}^{n\times n} \;:\; \xi_S\mathbf{1} = \mathbf{0},\; \boldsymbol{\pi}^T\xi_S = \mathbf{0}\}$.

$\mathbb{S}_n^{\pi}$, endowed with the **Fisher metric** is a **Riemannian manifold**.

Proposition (D., Meini, 2024)

The **orthogonal projection** $\Pi_S \;:\; \mathbb{R}^{n\times n} \to \mathcal{T}_S\mathbb{S}_n^{\pi}$ of a matrix Z w.r.t. the scalar product induced by Fisher's metric has the following expression:

$$\Pi_S(Z) = Z - (\boldsymbol{\alpha}\mathbf{1}^T + \boldsymbol{\pi}\boldsymbol{\beta}^T) \odot S,$$

where the vectors $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ are a solution to the following consistent linear system

$$\begin{bmatrix} Z\mathbf{1} \\ Z^T\boldsymbol{\pi} \end{bmatrix} = \begin{bmatrix} I & D_\pi S \\ S^T D_\pi & \mathrm{diag}(S^T D_\pi \boldsymbol{\pi}) \end{bmatrix} \begin{bmatrix} \boldsymbol{\alpha} \\ \boldsymbol{\beta} \end{bmatrix}, \quad D_\pi = \mathrm{diag}(\boldsymbol{\pi}).$$

Riemannian Gradient and Levi–Civita Connection

As promised we express the Riemannian gradient in terms of the Euclidean one.

Proposition (D., Meini 2024)

The **Riemannian gradient** $\operatorname{grad} f(S)$ is expressed in terms of the Euclidean gradient $\operatorname{Grad} f(S)$ as:

$$\operatorname{grad} f(S) = \Pi_S(\operatorname{Grad} f(S) \odot S).$$

Riemannian Gradient and Levi–Civita Connection

As promised we express the Riemannian gradient in terms of the Euclidean one.

Proposition (D., Meini 2024)

The **Riemannian gradient** $\operatorname{grad} f(S)$ is expressed in terms of the Euclidean gradient $\operatorname{Grad} f(S)$ as:

$$\operatorname{grad} f(S) = \Pi_S(\operatorname{Grad} f(S) \odot S).$$

To write the Riemannian Hessian we need an expression for the *Levi-Civita connection*.

Theorem (Koszul formula)

The Levi-Civita connection on the Euclidean space $\mathbb{R}^{n\times n}$ endowed with the Fisher information metric is given by

$$\nabla_{\eta_S}\xi_S = \mathrm{D}(\xi_S)[\eta_S] - \frac{1}{2}(\eta_S \odot \xi_S) \oslash S$$

Riemannian Hessian

Theorem (D., Meini, 2024)

The **Riemannian Hessian** $\mathrm{hess} f(S)[\xi_S]$ can be obtained from the Euclidean gradient $\mathrm{Grad}\, f(S)$ and the Euclidean Hessian $\mathrm{Hess}\, f(S)$ by using the identity

$$\mathrm{hess} f(S)[\xi_S] = \Pi_S(\mathrm{D}(\mathrm{grad} f(S))[\xi_S]) - \frac{1}{2}\Pi_S((\Pi_S(\mathrm{Grad} f(S)) \odot S) \odot \xi_S) \oslash S),$$

where $D(\mathrm{grad} f(S))[\xi_S] = \dot{\gamma}[\xi_S] - (\dot{\boldsymbol{\alpha}}[\xi_S]\mathbf{1}^T + \boldsymbol{\pi}\dot{\boldsymbol{\beta}}^T[\xi_S]) \odot S - (\boldsymbol{\alpha}\mathbf{1}^T + \boldsymbol{\pi}\boldsymbol{\beta}^T) \odot \xi_S$, and

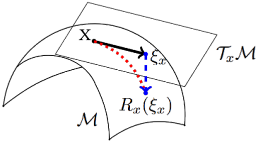
$$\gamma = \mathrm{Grad} f(S) \odot S, \quad \dot{\gamma}[\xi_S] = \mathrm{Hess}\, f(S)[\xi_S] \odot S + \mathrm{Grad}\, f(S) \odot \xi_S,$$

$$\mathcal{A} = \begin{bmatrix} I & D_\pi S \\ S^T D_\pi & \mathrm{diag}(S^T D_\pi \boldsymbol{\pi}) \end{bmatrix}, \quad \boldsymbol{\alpha}, \boldsymbol{\beta} \text{ s.t. } \mathcal{A}\begin{bmatrix} \boldsymbol{\alpha} \\ \boldsymbol{\beta} \end{bmatrix} = \begin{bmatrix} \gamma\mathbf{1} \\ \gamma^T \boldsymbol{\pi} \end{bmatrix},$$

$$\dot{\boldsymbol{\alpha}}[\xi_S], \dot{\boldsymbol{\beta}}[\xi_S] \text{ s.t. } \mathcal{A}\begin{bmatrix} \dot{\boldsymbol{\alpha}}[\xi_S] \\ \dot{\boldsymbol{\beta}}[\xi_S] \end{bmatrix} = \begin{bmatrix} \dot{\gamma}[\xi_S]\mathbf{1} \\ \dot{\gamma}^T[\xi_S]\boldsymbol{\pi} \end{bmatrix} - \begin{bmatrix} 0 & D_\pi \xi_S \\ \xi_S^T D_\pi & \mathrm{diag}(\xi_S^T D_\pi \boldsymbol{\pi}) \end{bmatrix}\begin{bmatrix} \boldsymbol{\alpha} \\ \boldsymbol{\beta} \end{bmatrix}.$$

Retraction

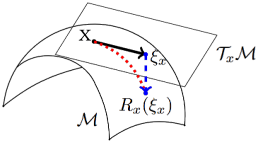
The last ingredient we need is a way to write the **retraction** for a point on the manifold.



Retraction

3 Riemannian Optimization

The last ingredient we need is a way to write the **retraction** for a point on the manifold.



Generalized Sinkhorn (D., Meini, 2024)

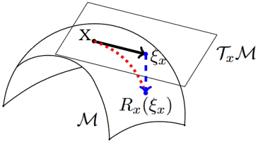
Let $A \in \mathbb{R}^{n\times n}$ be a matrix with positive entries. Then there exist diagonal matrices D_1 and D_2 such that

$$D_1 A D_2 \mathbf{1} = \mathbf{1}, \qquad \boldsymbol{\pi}^T D_1 A D_2 = \boldsymbol{\pi}^T.$$

Moreover, D_1 and D_2 are diagonal matrices such that $D_1 \widehat{A} D_2 \mathbf{1} = \boldsymbol{\pi}$ and $\mathbf{1}^T D_1 \widehat{A} D_2 = \boldsymbol{\pi}^T$, where $\widehat{A} = \mathrm{diag}(\boldsymbol{\pi}) A$.

Retraction

The last ingredient we need is a way to write the **retraction** for a point on the manifold.



Proposition (D., Meini, 2024)

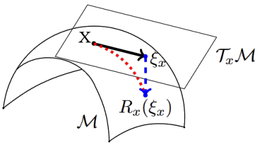
The map $R : \mathcal{T}\mathbb{S}_n^\pi \longrightarrow \mathbb{S}_n^\pi$ whose restriction R_S to $\mathcal{T}_S\mathbb{S}_n^\pi$ is given by:

$$R_S(\xi_S) = \mathcal{S}\left(S \odot \exp(\xi_S \oslash S)\right),$$

is a first-order retraction on $\mathbb{S}_n^\pi$, where $\mathcal{S}(\cdot)$ represents an application of the modified Sinkhorn-Knopp's algorithm, and $\exp(\cdot)$ the **entry-wise exponential**.

Retraction

The last ingredient we need is a way to write the **retraction** for a point on the manifold.



Proposition (D., Meini, 2024)

The map $R : \mathcal{T}\mathbb{S}_n^\pi \longrightarrow \mathbb{S}_n^\pi$ whose restriction R_S to $\mathcal{T}_S\mathbb{S}_n^\pi$ is given by:

$$R_S(\xi_S) = \mathcal{S}\left(S \odot \exp(\xi_S \oslash S)\right),$$

is a first-order retraction on $\mathbb{S}_n^\pi$, where $\mathcal{S}(\cdot)$ represents an application of the modified Sinkhorn-Knopp's algorithm, and $\exp(\cdot)$ the **entry-wise exponential**.

- The modified Sinkhorn–Knopp's algorithm is also used to **generate a random point on the manifold**.

Let's solve our credit score problem

Warning

To compute a stochastic approximation of $\sqrt{A}$, **we cannot use manifold-based optimization directly**, since the stationary distribution is $\pi = [0, \ldots, 0, 1]^T$.

Let's solve our credit score problem

Warning

To compute a stochastic approximation of $\sqrt{A}$, **we cannot use manifold-based optimization directly**, since the stationary distribution is $\pi = [0, \ldots, 0, 1]^T$.

- We use the **PageRank idea**

$$\tilde{A} = (1-\gamma)A + \gamma(\mathbf{1}\mathbf{1}^T)/n, \qquad 0 < \gamma \ll 1,$$

Let's solve our credit score problem

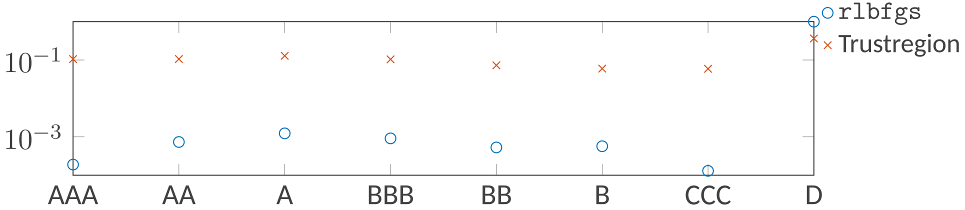
⚠ Warning ⚠

To compute a stochastic approximation of $\sqrt{A}$, **we cannot use manifold-based optimization directly**, since the stationary distribution is $\pi = [0, \ldots, 0, 1]^T$.

- We use the **PageRank idea**

$$\tilde{A} = (1-\gamma)A + \gamma(\mathbf{1}\mathbf{1}^T)/n, \qquad 0 < \gamma \ll 1,$$

- The **stationary vector** with $\gamma = 10^{-4}$ behaves as



Let’s solve our credit score problem

Warning

To compute a stochastic approximation of $\sqrt{A}$, **we cannot use manifold-based optimization directly**, since the stationary distribution is $\pi = [0, \ldots, 0, 1]^T$.

Riemannian L-BFGS

	1	2	3	4	5	6	7	8
1	0.9381	0.04559	2.47e-11	2.982e-12	0.004512	0.003408	0.003327	0.005103
2	0.006198	0.9485	0.03621	0.001512	0.003126	0.002732	0.0009793	0.0007403
3	0.0005382	0.01475	0.9496	0.02788	0.004653	0.002163	0.0004162	1.703e-06
4	0.0007266	0.001479	0.03398	0.9177	0.03377	0.008544	0.001459	0.002295
5	0.0004222	0.0005102	0.002895	0.03754	0.8814	0.05646	0.005312	0.01549
6	0.0002515	0.0002272	0.001226	0.001394	0.02827	0.909	0.02619	0.03349
7	0.0001144	0.0002228	0.004256	0.005232	0.009673	0.04372	0.816	0.1208
8	5.509e-06	9.268e-06	1.724e-06	1.826e-05	5.832e-06	3.127e-06	3.033e-06	1

Constrained Optimization (Trustregion)

	1	2	3	4	5	6	7	8
1	0.01805	0.9268	0.01508	0.008454	0.008151	0.008137	0.007701	0.007634
2	0.9231	0.01991	0.01593	0.00876	0.008369	0.008155	0.008022	0.007802
3	0.009915	0.01018	0.922	0.02399	0.009388	0.008503	0.008109	0.007893
4	0.008976	0.009147	0.02677	0.899	0.02654	0.01067	0.009724	0.00913
5	0.01005	0.01021	0.01134	0.03322	0.8652	0.03278	0.02315	0.01408
6	0.01063	0.01062	0.01194	0.01256	0.02281	0.03813	0.8465	0.04679
7	0.008773	0.009007	0.009906	0.01016	0.01492	0.8329	0.02088	0.09344
8	0.004959	0.00499	0.005011	0.005023	0.005075	0.005443	0.005045	0.9645

We want to find the **nearest reversible matrix** to a given transition matrix P with stationary distribution $\boldsymbol{\pi}$, i.e., the *feasible* set would be:

$$\mathcal{R} = \left\{ S \in \mathbb{R}^{n \times n} : S\mathbf{1} = \mathbf{1},\ \boldsymbol{\pi}^\top S = \boldsymbol{\pi}^\top,\ D_{\boldsymbol{\pi}} S = S^\top D_{\boldsymbol{\pi}},\ S \geqslant 0 \right\}$$

We want to find the **nearest reversible matrix** to a given transition matrix P with stationary distribution π, i.e., the *feasible* set would be:

$$\mathcal{R} = \left\{ S \in \mathbb{R}^{n \times n} : S\mathbf{1} = \mathbf{1},\ D_{\boldsymbol{\pi}} S = S^\top D_{\boldsymbol{\pi}},\ S \geqslant 0 \right\},$$

The manifold of reversible stochastic matrices

We want to find the **nearest reversible matrix** to a given transition matrix P with stationary distribution π, i.e., the *feasible* set would be:

$$\mathcal{R}_{>} = \left\{ S \in \mathbb{R}^{n\times n} : S\mathbf{1} = \mathbf{1},\ D_{\pi}S = S^{\top}D_{\pi},\ S \geqslant 0 \right\} \cap \{S \in \mathbb{R}^{n\times n} : S > 0\},$$

We want to use again the **Fisher information metric**.

The manifold of reversible stochastic matrices

We want to find the **nearest reversible matrix** to a given transition matrix P with stationary distribution π, i.e., the *feasible* set would be:

$$\mathcal{R}_{>} = \left\{ S \in \mathbb{R}^{n\times n} : S\mathbf{1} = \mathbf{1},\ D_\pi S = S^\top D_\pi,\ S \geqslant 0 \right\} \cap \{ S \in \mathbb{R}^{n\times n} : S > 0 \},$$

- We want to use again the **Fisher information metric**.
- We introduce the symmetrizing change of variables $\hat{S} = D_{\hat{\pi}} S D_{\hat{\pi}}^{-1}$, with $\hat{\pi} = \pi^{1/2}$:

$$\begin{aligned} D_\pi S D_\pi^{-1} &= S^\top, \\ D_{\hat{\pi}}\, D_{\hat{\pi}} S D_{\hat{\pi}}^{-1}\, D_{\hat{\pi}}^{-1} &= S^\top, \\ \underbrace{D_{\hat{\pi}} S D_{\hat{\pi}}^{-1}}_{\hat{S}} &= \underbrace{D_{\hat{\pi}}^{-1} S^\top D_{\hat{\pi}}}_{\hat{S}^\top}. \end{aligned}$$

The manifold of reversible stochastic matrices

We want to find the **nearest reversible matrix** to a given transition matrix P with stationary distribution π, i.e., the *feasible* set would be:

$$\mathcal{R}_{>} = \left\{ S \in \mathbb{R}^{n\times n} : S\mathbf{1} = \mathbf{1},\ D_{\pi} S = S^{\top} D_{\pi},\ S \geqslant 0 \right\} \cap \{S \in \mathbb{R}^{n\times n} : S > 0\},$$

- We want to use again the **Fisher information metric**.
- We introduce the symmetrizing change of variables $\hat{S} = D_{\hat{\pi}} S D_{\hat{\pi}}^{-1}$, with $\hat{\pi} = \pi^{1/2}$:

$$\hat{S}\hat{\pi} = D_{\hat{\pi}} S D_{\hat{\pi}}^{-1} \hat{\pi} = D_{\hat{\pi}} S\mathbf{1} = D_{\hat{\pi}}\mathbf{1} = \hat{\pi},$$

The manifold of reversible stochastic matrices

We want to find the **nearest reversible matrix** to a given transition matrix P with stationary distribution π, i.e., the *feasible* set would be:

$$\mathcal{M}_{\pi} = \left\{ S \in \mathbb{R}^{n\times n} : S > 0,\ S = S^{\top},\ S\hat{\pi} = \hat{\pi} \right\}, \qquad \hat{\pi} = \pi^{1/2}.$$

- We want to use again the **Fisher information metric**.
- We introduce the symmetrizing change of variables $\hat{S} = D_{\hat{\pi}} S D_{\hat{\pi}}^{-1}$, with $\hat{\pi} = \pi^{1/2}$:
- The problem is now equivalent to:

$$P^{\star} = \arg\min_{\hat{X}\in\mathcal{M}_{\pi}} \frac{1}{2}\|D_{\hat{\pi}}^{-1}\hat{X}D_{\hat{\pi}} - A\|_F^2,$$

where $\mathcal{M}_{\pi}$ is a Riemannian manifold and the **nearest reversible chain** is

$$P = D_{\hat{\pi}}^{-1} P^{\star} D_{\hat{\pi}}.$$

- $\mathcal{M}_{\pi}$ is an **embedded manifold** of $\mathcal{S}_n = \{X \in \mathbb{R}^{n\times n} : X = X^{\top}\}$.

The computational tools for $\mathcal{M}_{\pi}$

As for the manifold $\mathbb{S}_n^{\pi}$, we need to derive the **tangent space**, the **projection**, the **retraction** and the **Levi-Civita connection** for $\mathcal{M}_{\pi}$.

The computational tools for $\mathcal{M}_\pi$

As for the manifold $\mathbb{S}_n^\pi$, we need to derive the **tangent space**, the **projection**, the **retraction** and the **Levi-Civita connection** for $\mathcal{M}_\pi$.

Lemma (D., Gnazzo, Meini 2025a)

The **tangent space** to $\mathcal{M}_\pi$ at a point $S \in \mathcal{M}_\pi$ is

$$\mathcal{T}_S\mathcal{M}_\pi = \{\xi_S \in \mathcal{S}_n : \ \xi_S \hat{\pi} = 0\}.$$

The computational tools for $\mathcal{M}_\pi$

As for the manifold $\mathbb{S}_n^\pi$, we need to derive the **tangent space**, the **projection**, the **retraction** and the **Levi-Civita connection** for $\mathcal{M}_\pi$.

Lemma (D., Gnazzo, Meini 2025a)

The **tangent space** to $\mathcal{M}_\pi$ at a point $S \in \mathcal{M}_\pi$ is

$$\mathcal{T}_S\mathcal{M}_\pi = \{\xi_S \in \mathcal{S}_n : \ \xi_S \hat{\boldsymbol{\pi}} = 0\} .$$

Lemma (D., Gnazzo, Meini 2025a)

The **orthogonal complement** of the tangent space $\mathcal{T}_S\mathcal{M}_\pi$, with respect to the Fisher metric, has the expression

$$\mathcal{T}_S^\perp\mathcal{M}_\pi = \{\xi_S^\perp \in \mathcal{S}_n \ : \ \xi_S^\perp = (\boldsymbol{\alpha}\hat{\boldsymbol{\pi}}^\top + \hat{\boldsymbol{\pi}}\boldsymbol{\alpha}^\top) \odot S\}, \qquad \boldsymbol{\alpha} \in \mathbb{R}^n.$$

The **Riemannian gradient** $\operatorname{grad} f(S)$ can be expressed in terms of the Euclidean gradient $\operatorname{Grad} f(S)$ via *projection* onto the tangent space $\mathcal{T}_S\mathcal{M}_\pi$:

$$\begin{aligned}\Pi_S : \mathcal{S}_n &\mapsto \mathcal{T}_S\mathcal{M}_\pi \\ Z &\mapsto Z - (\alpha\hat{\pi}^\top + \hat{\pi}\alpha^\top) \odot S,\end{aligned}$$

where α is the solution to the linear system

$$\mathcal{A}\alpha \equiv \left(\operatorname{diag}(SD_{\hat{\pi}}\hat{\pi}) + D_{\hat{\pi}}SD_{\hat{\pi}}\right)\alpha = Z\hat{\pi}.$$

The **Riemannian gradient** $\operatorname{grad} f(S)$ can be expressed in terms of the Euclidean gradient $\operatorname{Grad} f(S)$ via *projection* onto the tangent space $\mathcal{T}_S\mathcal{M}_\pi$:

Lemma (D., Gnazzo, Meini 2025a)

Consider a smooth function $f : \mathcal{M}_\pi \mapsto \mathbb{R}$, the **Riemannian gradient** $\operatorname{grad} f(S)$ has the following expression:

$$\operatorname{grad} f(S) = \widetilde{\Pi}_S(\operatorname{Grad} f(S) \odot S),$$

where the projection $\widetilde{\Pi}_S : \mathbb{R}^{n\times n} \mapsto \mathcal{T}_S\mathcal{M}_\pi$ is defined as $\widetilde{\Pi}_S(Z) := \Pi_S\left({}^{Z+Z^\top}/_2\right)$, and $\operatorname{Grad} f(S)$ denotes the euclidean gradient.

Lemma (D., Gnazzo, Meini 2025a)

The Riemannian Hessian $\operatorname{hess} f(S)[\xi_S]$ can be obtained from the Euclidean gradient $\operatorname{Grad} f(S)$ and Hessian $\operatorname{Hess} f(S)$ as

$$\operatorname{hess} f(S)[\xi_S] = \widetilde{\Pi}_S(\mathrm{D}(\operatorname{grad} f(S))[\xi_S]) - \frac{1}{2}\widetilde{\Pi}_S(((\widetilde{\Pi}_S(\operatorname{Grad} f(S)) \odot S) \odot \xi_S) \oslash S),$$

where

$$\mathrm{D}(\operatorname{grad} f(S))[\xi_S] = \dot{\gamma}[\xi_S] - (\dot{\boldsymbol{\alpha}}[\xi_S]\hat{\boldsymbol{\pi}}^\top + \hat{\boldsymbol{\pi}}\dot{\boldsymbol{\alpha}}[\xi_S]^\top) \odot S - (\boldsymbol{\alpha}\hat{\boldsymbol{\pi}}^\top + \hat{\boldsymbol{\pi}}\boldsymbol{\alpha}^\top) \odot \xi_S.$$

$$\mathcal{A}\boldsymbol{\alpha} = \frac{(\gamma + \gamma^\top)}{2}\hat{\boldsymbol{\pi}},$$

$$\mathcal{A}\dot{\boldsymbol{\alpha}}[\xi_S] = \mathbf{b} \equiv \frac{(\dot{\gamma}[\xi_S] + \dot{\gamma}[\xi_S]^\top)}{2}\hat{\boldsymbol{\pi}} - (\operatorname{diag}(\xi_S D_{\hat{\boldsymbol{\pi}}}\hat{\boldsymbol{\pi}}) + D_{\hat{\boldsymbol{\pi}}}\xi_S D_{\hat{\boldsymbol{\pi}}})\boldsymbol{\alpha},$$

$$\dot{\gamma}[\xi_S] = \operatorname{Hess} f(S)[\xi_S] \odot S + \operatorname{Grad} f(S) \odot \xi_S.$$

Retraction

For the **retraction** we need a further extension of the *Sinkhorn–Knopp's algorithm*,

Lemma (D., Gnazzo, Meini 2025a)

Given a symmetric $A \in \mathbb{R}^{n \times n}$, with positive entries, there exists a diagonal matrix D s.t.

$$DAD\hat{\pi} = \hat{\pi}.$$

Retraction

For the **retraction** we need a further extension of the *Sinkhorn–Knopp's algorithm*,

Proposition (D., Gnazzo, Meini 2025a)

The map $R_S : \mathcal{T}_S\mathcal{M}_\pi \to \mathcal{M}_\pi$ given by

$$R_S(\xi_S) = \mathcal{P}\left(S \odot \exp(\xi_S \oslash S)\right),$$

is a **first-order retraction** on $\mathcal{M}_\pi$, where $\exp(\cdot)$ is the entry-wise exponential and $\mathcal{P}$ applies the modified Sinkhorn.

- For numerical stability, we **symmetrize** the output:

$$R_S(\xi_S) \leftarrow \frac{1}{2}\left(D_1\mathcal{P}(S \odot \exp(\xi_S \oslash S))D_2 + D_2\mathcal{P}(S \odot \exp(\xi_S \oslash S))^\top D_1\right),$$

  where D_1, D_2 are the diagonal matrices from the Sinkhorn variant.
- This ensures exact symmetry and preservation of marginals up to machine precision.

Synthetic Test Problems

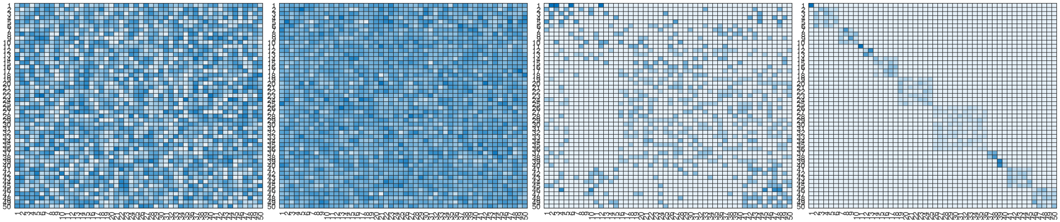
Four classes of synthetic Markov chains:

Uniformly random Matrix G with entries $\sim \mathcal{U}(0,1)$, then $A = \mathrm{diag}(G\mathbf{1})^{-1}G$

Normal random Matrix G with entries $\sim \mathcal{N}(1,1)$, then $A = \mathrm{diag}(G\mathbf{1})^{-1}G$

Stochastic block model Adjacency matrix from random walk on SBM graph (via NetworkX)

Multiple ergodic classes Concatenation of independent uniform Markov chains



Experimental Setup

Comparison of three algorithms:

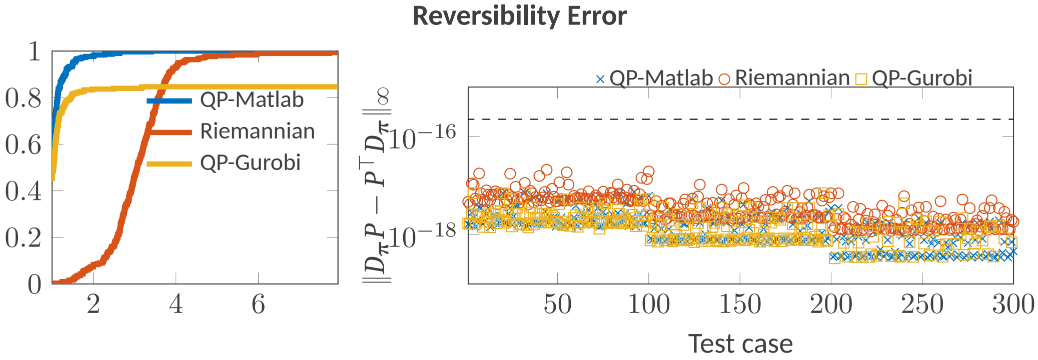
- **Riemannian:** Newton method on $\mathcal{M}_\pi$
- **QP-Matlab:** Quadratic programming with quadprog
- **QP-Gurobi:** Quadratic programming with barrier method
- The **QP methods** solve the original problem with linear constraints, and are the one proposed in: A. Nielsen and M. Weber, Computing the nearest reversible Markov chain, Numer. Linear Algebra Appl. **22** (2015), no. 3, 483–499.

Test suite: $n_p = 300$ problems

- Three matrix sizes: $n \in \{50, 100, 200\}$
- 25 problems per class per size

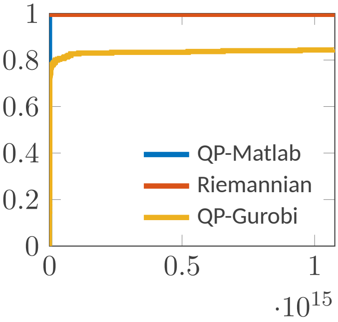
Performance metrics:

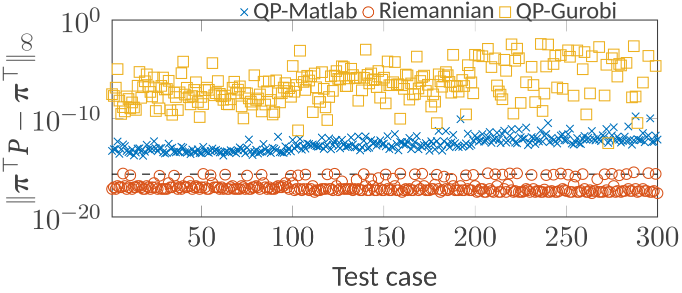
1. Computation time
2. Reversibility error: $\|D_\pi P - P^\top D_\pi\|_\infty$
3. Stationarity error: $\|\pi^\top P - \pi^\top\|_\infty$
4. Relative distance: $\|A - P\|_F / \|A\|_F$



All methods achieve machine precision $\approx 2^{-52}$, with Riemannian showing smallest variability.

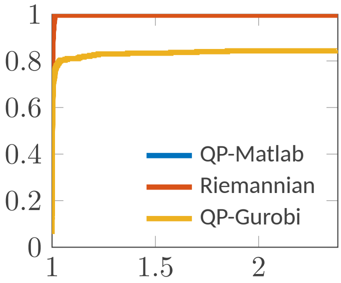
Stationarity Error

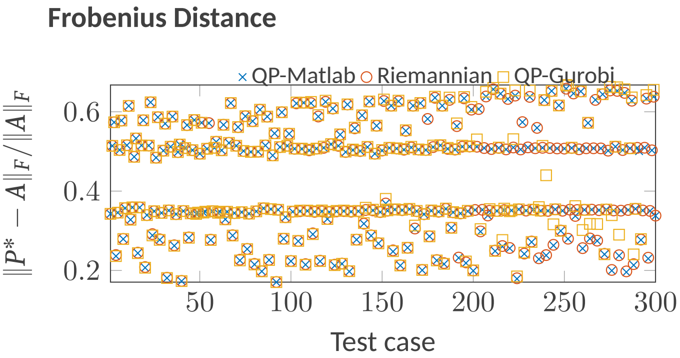




All methods achieve machine precision $\approx 2^{-52}$, with Riemannian showing smallest variability.

Results: Reversibility, Stationarity, Quality and Speed

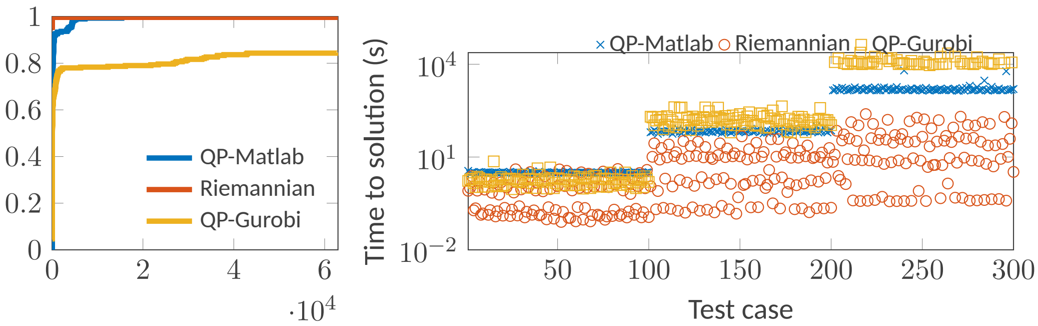




Riemannian and QP-Matlab achieve identical minimal distances.

Results: Reversibility, Stationarity, Quality and Speed

Computation Time



Riemannian is fastest, solving $\approx 70\%$ of problems in < 1 second, while QP-Gurobi scales poorly.

Summary of Results

Accuracy:

- All methods: reversibility/stationarity at machine precision
- Riemannian: least variable results
- Riemannian + QP-Matlab: minimal Frobenius distance

Efficiency:

- Riemannian: fastest on majority of instances
- Most instances solved in < 1 second
- QP methods: poor scaling with size
- Riemannian: best for moderate-sized matrices

Conclusion

The **Riemannian algorithm** offers the best balance of **accuracy**, **reliability**, and **computational efficiency** for computing nearest reversible Markov chains.

$$\min_{\Delta \in \mathbb{R}^{n\times n}} \quad \operatorname{tr}\left(\left(I - (P+\Delta) + \mathbf{1}\mathbf{h}^\top\right)^{-1}\right) + \frac{1}{2}\|\Delta\|_F^2$$
$$\text{s.t.} \quad \Delta\mathbf{1} = \mathbf{0}, \quad P + \Delta \geqslant 0$$

Objective:

- Minimize Kemeny's constant of perturbed chain
- Regularize via $\frac{1}{2}\|\Delta\|_F^2$
- Penalize large perturbations

Constraints:

- $\Delta\mathbf{1} = \mathbf{0}$: row sum zero
- $P + \Delta \geqslant 0$: non-negativity
- Ensures $P + \Delta$ remains stochastic

Challenge

The mapping $P \mapsto \kappa(P)$ is **non-convex**, so global optimality is generally not guaranteed.

The Challenge:

- General problem: $P \mapsto \kappa(P)$ is **non-convex**
- No guarantee of global optimality
- Solution: **restrict to reversible chains**

Making Kemeny's Constant Convex

Key Assumption:

- P is **reversible**: $D_\pi P = P^\top D_\pi$
- π is the **stationary distribution**
- Perturbations Δ preserve these properties

Constraints on Δ:

$$\Delta \mathbf{1} = \mathbf{0}, \quad P + \Delta \geqslant 0, \quad D_\pi \Delta = \Delta^\top D_\pi$$

Symmetrizing Transformation:
Define $\hat{\pi} = \pi^{1/2}$ (component-wise), then multiply the matrix inverse by $D_{\hat{\pi}}$ on the left and $D_{\hat{\pi}}^{-1}$ on the right:

$$\operatorname{tr}\left(\left(I - D_{\hat{\pi}}(P+\Delta)D_{\hat{\pi}}^{-1} + \hat{\pi}\hat{\pi}^\top\right)^{-1}\right) + \frac{1}{2}\|\Delta\|_F^2$$

This **symmetrizes** the problem while preserving the objective value.

Change of Variables:

Let $X = P + \Delta$ (the perturbed Markov chain), then:

$$\min_{X \in \mathbb{R}^{n\times n}(\mathcal{S})} \operatorname{tr}\left(\left(I - D_{\hat{\pi}} X D_{\hat{\pi}}^{-1} + \hat{\pi}\hat{\pi}^\top\right)^{-1}\right) + \frac{1}{2}\|X - P\|_F^2,$$
$$\text{s.t. } X \geqslant 0,\ D_{\pi} X = X^\top D_{\pi},\ X\mathbf{1} = \mathbf{1}.$$

where $\mathbb{R}^{n\times n}(\mathcal{S})$ is the set of **reversible stochastic matrices** respecting the pattern $\mathcal{S}$.

Sparsity Pattern: For an integer $n \geqslant 1$, a *pattern* $\mathcal{S}$ is a set of unordered pairs where

$$\{i, i\} \in \mathcal{S}\ \forall i = 1, \dots, n \quad \text{and} \quad \mathcal{S} \subseteq \{\{i, j\} \mid 1 \leqslant i, j \leqslant n\}.$$

A matrix X *respects the pattern* $\mathcal{S}$, i.e., $X \in \mathbb{R}^{n\times n}(\mathcal{S})$ if

$$\{i, j\} \notin \mathcal{S} \ \Rightarrow\ X_{ij} = X_{ji} = 0.$$

Making Kemeny's Constant Convex

Why This Becomes Convex:

- **Reversibility constraint:** $D_\pi X = X^\top D_\pi$ defines a convex set
- **Stochasticity constraints:** $X\mathbf{1} = \mathbf{1}$ and $X \geqslant 0$ are linear
- **Objective:** On the restricted domain, the Kemeny constant becomes **convex**
- **Riemannian approach:** Exploit manifold structure of reversible chains for efficient optimization

Result

The **convex problem** can be solved efficiently using **Riemannian optimization** on the manifold of reversible stochastic matrices with given pattern.

Why This Becomes Convex:

- **Reversibility constraint:** $D_\pi X = X^\top D_\pi$ defines a convex set
- **Stochasticity constraints:** $X\mathbf{1} = \mathbf{1}$ and $X \geqslant 0$ are linear
- **Objective:** On the restricted domain, the Kemeny constant becomes **convex**
- **Riemannian approach:** Exploit manifold structure of reversible chains for efficient optimization

Proposition (D., Gnazzo, Meini 2025b)

Given a stationary distribution $\pi > 0$ and an irreducible pattern $\mathcal{S}$, the set of reversible stochastic matrices $X \in \mathbb{R}^{n \times n}(\mathcal{S})$, with pattern $\mathcal{S}$ is not empty.

Applying the Riemannian Recipe

At this point we can apply the **Riemannian recipe** to solve the problem on the manifold:

$$\mathcal{M}_{\pi} = \left\{ X \in \mathbb{R}^{n\times n}_{\text{exact}}(\mathcal{S}) : X = X^{\top}, X\hat{\pi} = \hat{\pi}, X_{ij} > 0 \text{ for } \{i,j\} \in \mathcal{S} \right\},$$

Exact pattern: A matrix $\Delta \in \mathbb{R}^{n\times n}_{\text{exact}}(\mathcal{S})$ if $\Delta_{ij} \neq 0$ and $\Delta_{ji} \neq 0 \quad \Leftrightarrow \quad \{i,j\} \in \mathcal{S}$.

Symmetrization: We use the symmetrizing change of variables $\hat{X} = D_{\hat{\pi}} X D_{\hat{\pi}}^{-1}$ to transform the problem into one on **symmetric matrices**.

Modified Fisher metric: We equip $\mathcal{M}_{\pi}$ with a modified **Fisher information metric**:

$$\langle \xi_X, \eta_X \rangle_X = \sum_{X_{ij}\neq 0} \frac{(\xi_X)_{ij}(\eta_X)_{ij}}{X_{ij}} = \operatorname{tr}\left((\xi_X \oslash X)\eta_X^{\top} \right),$$

where $\oslash$ denotes the entry-wise division: $(A \oslash B)_{ij} = \begin{cases} A_{ij}/B_{ij}, & B_{ij} \neq 0, \\ 0, & B_{ij} = 0. \end{cases}$

Computational Tools for $\mathcal{M}_\pi$

The manifold $\mathcal{M}_\pi$ is an **embedded submanifold** of the symmetric matrices with the same pattern, so we can derive the **tangent space**, the **orthogonal space**, and the **projection** by extending the tools from the manifold of reversible chains without pattern constraints.

Tangent: $\mathcal{T}_X\mathcal{M}_{\pi} = \left\{\xi_X \in \mathbb{R}^{n\times n}(\mathcal{S}) : \xi_X = \xi_X^\top, \xi_X\hat{\pi} = 0\right\}.$

Computational Tools for $\mathcal{M}_\pi$

The manifold $\mathcal{M}_\pi$ is an **embedded submanifold** of the symmetric matrices with the same pattern, so we can derive the **tangent space**, the **orthogonal space**, and the **projection** by extending the tools from the manifold of reversible chains without pattern constraints.

Tangent: $\mathcal{T}_X\mathcal{M}_\pi = \left\{\xi_X \in \mathbb{R}^{n\times n}(\mathcal{S}) : \xi_X = \xi_X^\top, \xi_X\hat{\boldsymbol{\pi}} = 0\right\}$.

Orthogonal: Given $\boldsymbol{\alpha} \in \mathbb{R}^n$, and $S \in \mathbb{R}^{n\times n}$ is defined as $S_{ij} = \begin{cases} 1, & \text{for } \{i,j\} \in \mathcal{S}, \\ 0, & \text{for } \{i,j\} \notin \mathcal{S}. \end{cases}$

$$\mathcal{T}_X^\perp\mathcal{M}_\pi = \left\{\xi_X^\perp \in \mathbb{R}^{n\times n} : \xi_X^\perp = \left(\boldsymbol{\alpha}\hat{\boldsymbol{\pi}}^\top + \hat{\boldsymbol{\pi}}\boldsymbol{\alpha}^\top\right) \odot (X \odot S)\right\},$$

Computational Tools for $\mathcal{M}_\pi$

The manifold $\mathcal{M}_\pi$ is an **embedded submanifold** of the symmetric matrices with the same pattern, so we can derive the **tangent space**, the **orthogonal space**, and the **projection** by extending the tools from the manifold of reversible chains without pattern constraints.

Tangent: $\mathcal{T}_X\mathcal{M}_\pi = \left\{\xi_X \in \mathbb{R}^{n\times n}(\mathcal{S}) \,:\, \xi_X = \xi_X^\top, \xi_X\hat{\boldsymbol{\pi}} = 0\right\}.$

Orthogonal: Given $\boldsymbol{\alpha} \in \mathbb{R}^n$, and $S \in \mathbb{R}^{n\times n}$ is defined as $S_{ij} = \begin{cases} 1, & \text{for } \{i,j\} \in \mathcal{S}, \\ 0, & \text{for } \{i,j\} \notin \mathcal{S}. \end{cases}$

$$\mathcal{T}_X^\perp\mathcal{M}_\pi = \left\{\xi_X^\perp \in \mathbb{R}^{n\times n} : \xi_X^\perp = \left(\boldsymbol{\alpha}\hat{\boldsymbol{\pi}}^\top + \hat{\boldsymbol{\pi}}\boldsymbol{\alpha}^\top\right) \odot (X \odot S)\right\},$$

Projection: $\Pi_X : \mathbb{R}^{n\times n} \mapsto \mathcal{T}_X\mathcal{M}_\pi$ with $Z \mapsto \frac{(Z+Z^\top)\odot S}{2} - (\boldsymbol{\alpha}\hat{\boldsymbol{\pi}}^\top + \hat{\boldsymbol{\pi}}\boldsymbol{\alpha}^\top) \odot (X \odot S)$, and $\boldsymbol{\alpha}$ is the solution to the linear system

$$\frac{(Z+Z^\top)\odot S}{2}\hat{\boldsymbol{\pi}} = \left(\operatorname{diag}((X\odot S)\boldsymbol{\pi}) + D_{\hat{\boldsymbol{\pi}}}(X\odot S)D_{\hat{\boldsymbol{\pi}}}\right)\boldsymbol{\alpha}.$$

We extend the *Sinkhorn–Knopp's algorithm* to ensure that the retraction preserves the pattern $\mathcal{S}$:

Lemma (D., Gnazzo, Meini 2025b)

Let $P \in \mathbb{R}^{n\times n}$ be a reversible stochastic matrix with stationary distribution $\boldsymbol{\pi}$, and $\mathcal{S}$ a pattern for which

$$P^0_{ij} = \begin{cases} P_{ij}, & \text{for } \{i,j\} \notin \mathcal{S}, \\ 0, & \text{for } \{i,j\} \in \mathcal{S}. \end{cases}$$

is such that $\|P^0\mathbf{1}\|_\infty < 1$. Then, for any nonnegative symmetric matrix $A \in \mathbb{R}^{n\times n}(\mathcal{S})$, $A \neq 0$ with total support—i.e., such that all its nonzero elements lie on a positive diagonal—there exists a diagonal matrix D with positive diagonal entries such that $DAD\hat{\boldsymbol{\pi}} = \hat{\boldsymbol{\pi}} - \boldsymbol{\beta}$, with $\boldsymbol{\beta} = P^0\hat{\boldsymbol{\pi}}$.

We extend the *Sinkhorn–Knopp's algorithm* to ensure that the retraction preserves the pattern $\mathcal{S}$:

Lemma (D., Gnazzo, Meini 2025b)

Let $P \in \mathbb{R}^{n\times n}$ be a reversible stochastic matrix with stationary distribution π, and $\mathcal{S}$ a pattern for which P^0 is such that $\|P^0\mathbf{1}\|_\infty < 1$. Then, for any nonnegative symmetric matrix $A \in \mathbb{R}^{n\times n}(\mathcal{S})$, $A \neq 0$ with total support—i.e., such that all its nonzero elements lie on a positive diagonal—there exists a diagonal matrix D with positive diagonal entries such that $DAD\hat{\pi} = \hat{\pi} - \beta$, with $\beta = P^0\hat{\pi}$.

⚠ Due to the complexity of the objective function **we do not go for second order optimization methods**, and avoid the explicit computation of the Riemannian Hessian.

Changing the pattern $\mathcal{S}$ from the pattern of P

- We have also considered the problem of changing the pattern $\mathcal{S}$ from the pattern of P to a different one $\mathcal{S}'$, with $\mathcal{S} \subseteq \mathcal{S}'$.
- This is equivalent to allowing some entries of X to be zero, and others to be positive, without changing the pattern of P.
- The manifold is now defined as

$$\mathcal{M}_{P,\pi} = \left\{ X \in \mathbb{R}^{n\times n}_{\text{exact}}(\mathcal{P} \cup \mathcal{S}) : X = X^\top, X\hat{\pi} = \hat{\pi}, X_{ij} > 0 \text{ if } \{i,j\} \in \mathcal{S}, \quad X_{ij} = \frac{\hat{\pi}_i}{\hat{\pi}_j} P_{ij} \text{ if } \{i,j\} \notin \mathcal{S} \right\}.$$

- Computational tools are similar to the case $\mathcal{S} = \mathcal{P}$, with some technical adjustments.
- This allows us to **select the pattern adaptively**, i.e., **removing entries which are going to zero**.

Application: Power Grid Networks

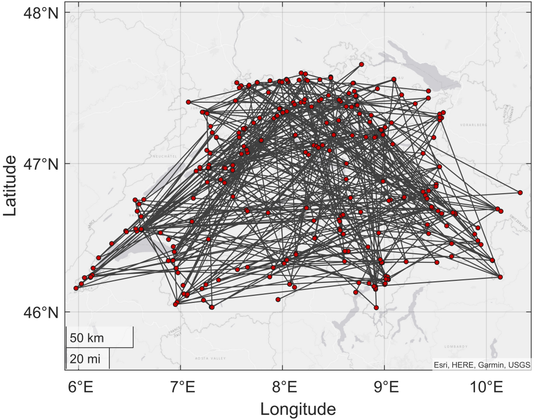
Dataset: Power grids from `Power grid` repository (5 countries, largest connected components)

Setup:

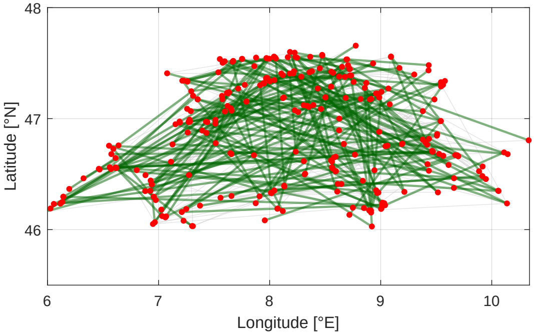
- Adjacency matrix $A \to$ Random walk: $P = D_{\mathbf{d}}^{-1} A$ with $\mathbf{d} = A\mathbf{1}$
- P is irreducible and reversible w.r.t. $\pi = \mathbf{d}/\|\mathbf{d}\|_1$
- Apply Riemannian optimizer to minimize Kemeny's constant

Network	Size	nnz	Density	$\mathcal{K}(P)$	$\mathcal{K}(X)$	Time (s)
Austria	147	336	1.55%	1.19×10^3	1.15×10^3	17.91
Belgium	90	218	2.69%	5.21×10^2	4.95×10^2	6.12
Denmark	63	136	3.42%	6.68×10^2	6.46×10^2	4.59
Netherlands	84	190	2.69%	5.23×10^2	5.05×10^2	5.50
Switzerland	310	736	0.76%	1.94×10^4	2.62×10^3	102.98

Power Network: Switzerland

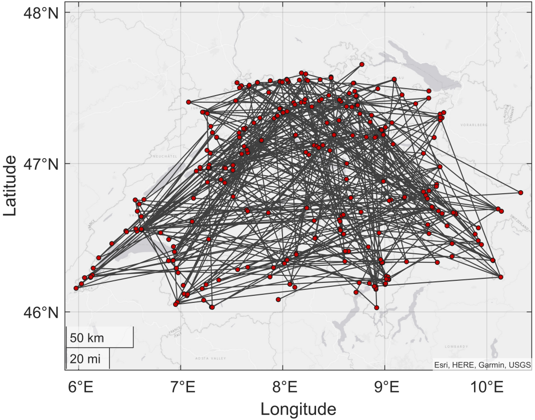


Original power network

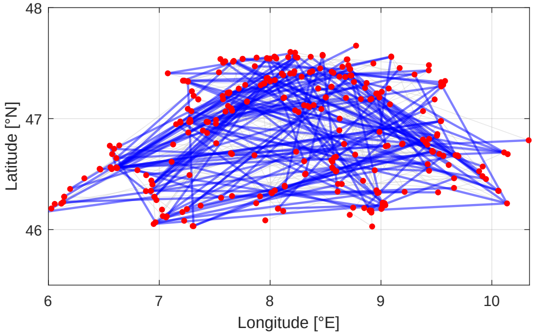


Edges with decreased weights

Power Network: Switzerland



Original power network



Edges with increased weights

Conclusion and Future Work

Conclusion:

- The Riemannian algorithm provides an **effective** method for problems in Markov chains.
- It balances *accuracy*, *reliability*, and *computational efficiency*.

Future Work:

- Can we say something about the **geodesics** on these manifolds? Can we use them to design better optimization algorithms?
- Are there other applications of this Riemannian framework to problems in Markov chains, e.g., **model reduction**, **clustering**, or **spectral analysis**?
- Making *pull requests* to MANOPT to include these manifolds as built-in options.

Bibliography and </> Code

The **main references** for this seminar are:

[3] F. Durastante, M. Gnazzo, and B. Meini. *A Riemannian Optimization Approach for Finding the Nearest Reversible Markov Chain*. 2025. arXiv: 2505.16762 [math.NA]. url: https://arxiv.org/abs/2505.16762.

[4] F. Durastante, M. Gnazzo, and B. Meini. *Kemeny's constant minimization for reversible Markov chains via structure-preserving perturbations*. 2025. arXiv: 2510.24679 [math.NA]. url: https://arxiv.org/abs/2510.24679.

[5] F. Durastante and B. Meini. "Stochastic pth root approximation of a stochastic matrix: a Riemannian optimization approach". In: *SIAM J. Matrix Anal. Appl.* 45.2 (2024), pp. 875–904. issn: 0895-4798,1095-7162. doi: 10.1137/23M1589463. url: https://doi.org/10.1137/23M1589463.

Bibliography and </> Code

The **references for algorithmic and theoretical comparison** are:

[1] N. Boumal et al. "Manopt, a Matlab Toolbox for Optimization on Manifolds". In: *Journal of Machine Learning Research* 15.42 (2014), pp. 1455–1459. url: http://jmlr.org/papers/v15/boumal14a.html.

[2] A. Douik and B. Hassibi. "Manifold optimization over the set of doubly stochastic matrices: a second-order geometry". In: *IEEE Trans. Signal Process.* 67.22 (2019), pp. 5761–5774. issn: 1053-587X,1941-0476. doi: 10.1109/TSP.2019.2946024. url: https://doi.org/10.1109/TSP.2019.2946024.

Bibliography and </> Code

[6] N. J. Higham and L. Lin. "On pth roots of stochastic matrices". In: *Linear Algebra Appl.* 435.3 (2011), pp. 448–463. issn: 0024-3795,1873-1856. doi: 10.1016/j.laa.2010.04.007. url: https://doi.org/10.1016/j.laa.2010.04.007.

[7] A. J. N. Nielsen and M. Weber. "Computing the nearest reversible Markov chain". In: *Numer. Linear Algebra Appl.* 22.3 (2015), pp. 483–499. issn: 1070-5325,1099-1506. doi: 10.1002/nla.1967. url: https://doi.org/10.1002/nla.1967.

The **code** is available on GitHub:

- https://github.com/Cirdans-Home/pth-root-stochastic
- https://github.com/miryamgnazzo/nearest-reversible
- https://github.com/Cirdans-Home/optimize-kemeny

A Riemannian Perspective on Optimization Problems in Markov Chains

Thank you for listening!
Any questions?