Adaptive Riemannian Metrics for Improved Geodesic Tracking of White Matter

Xiang Hao, Ross T. Whitaker, and P. Thomas Fletcher

Scientific Computing and Imaging Institute
University of Utah, Salt Lake City, UT

July 4, 2011

Geodesic Tracking on DTI

Given

- A starting point p
- A local cost function $f(x, v)$

$$\min \int_0^1 f(x, v)\,dt$$

Geodesic Tracking on DTI

Given

- A starting point p
- A local cost function $f(x, v)$

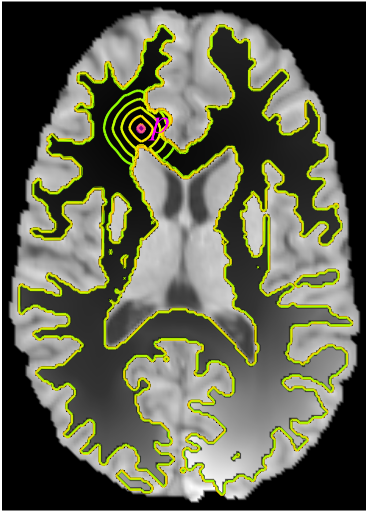
$$\min \int_0^1 f(x, v)dt$$

Algorithm:

1. Compute the time-of-arrival function u by propagating from p

Output:

- Time-of-arrival function u



Geodesic Tracking on DTI

Given

- A starting point p
- A local cost function $f(x, v)$

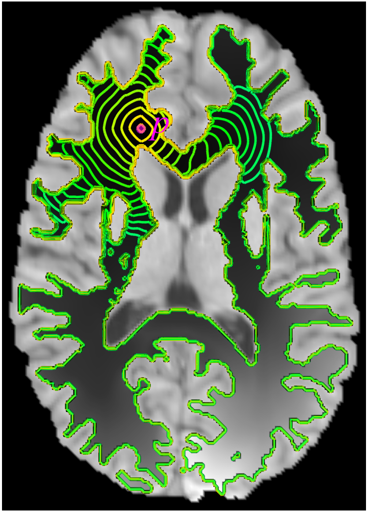
$$\min \int_0^1 f(x,v)dt$$

Algorithm:

1. Compute the time-of-arrival function u by propagating from p

Output:

- Time-of-arrival function u



Geodesic Tracking on DTI

Given

- A starting point p
- A local cost function $f(x, v)$

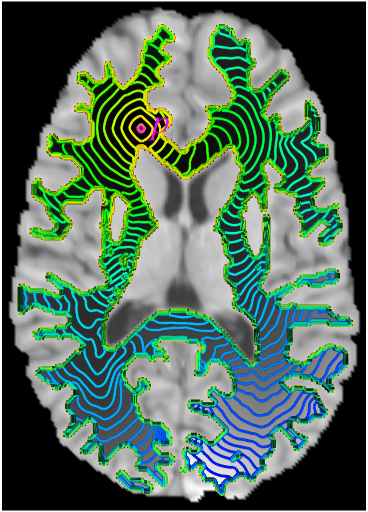
$$\min \int_0^1 f(x, v) dt$$

Algorithm:

1. Compute the time-of-arrival function u by propagating from p

Output:

- Time-of-arrival function u



Geodesic Tracking on DTI

Given

- A starting point p
- A local cost function $f(x, v)$

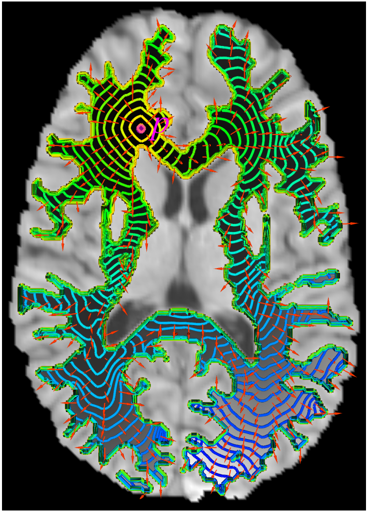
$$\min \int_0^1 f(x, v)dt$$

Algorithm:

1. Compute the time-of-arrival function u by propagating from p
2. Compute '∇u', from any point q integrate '∇u' backward to p

Output:

- Time-of-arrival function u



Geodesic Tracking on DTI

Given

- A starting point p
- A local cost function $f(x, v)$

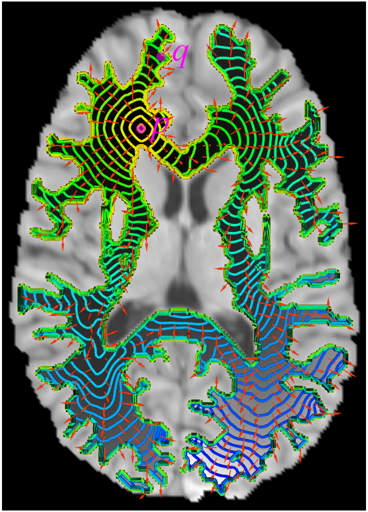
$$\min \int_0^1 f(x, v)dt$$

Algorithm:

1. Compute the time-of-arrival function u by propagating from p
2. Compute '∇u', from any point q integrate '∇u' backward to p

Output:

- Time-of-arrival function u



Geodesic Tracking on DTI

Given

- A starting point p
- A local cost function $f(x, v)$

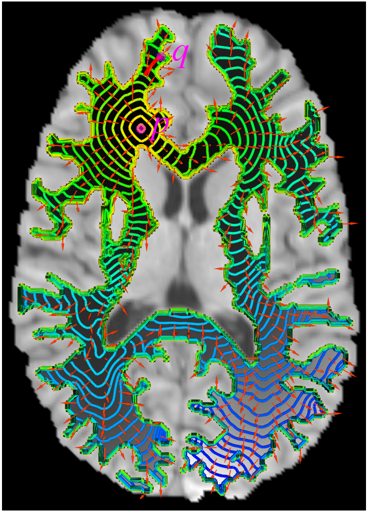
$$\min \int_0^1 f(x, v)dt$$

Algorithm:

1. Compute the time-of-arrival function u by propagating from p
2. Compute '∇u', from any point q integrate '∇u' backward to p

Output:

- Time-of-arrival function u



Geodesic Tracking on DTI

Given

- A starting point p
- A local cost function $f(x, v)$

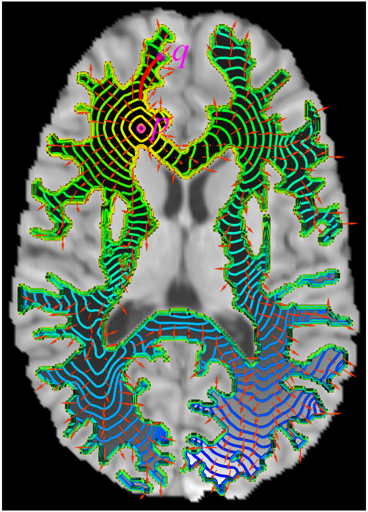
$$\min \int_0^1 f(x, v)dt$$

Algorithm:

1. Compute the time-of-arrival function u by propagating from p
2. Compute '∇u', from any point q integrate '∇u' backward to p

Output:

- Time-of-arrival function u



Geodesic Tracking on DTI

Given

- A starting point p
- A local cost function $f(x, v)$

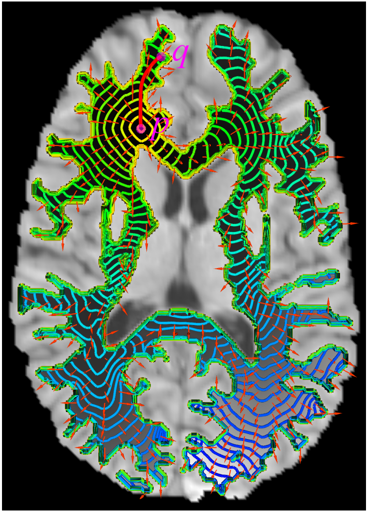
$$\min \int_0^1 f(x,v)dt$$

Algorithm:

1. Compute the time-of-arrival function u by propagating from p
2. Compute '∇u', from any point q integrate '∇u' backward to p

Output:

- Time-of-arrival function u



Geodesic Tracking on DTI

Given

- A starting point p
- A local cost function $f(x, v)$

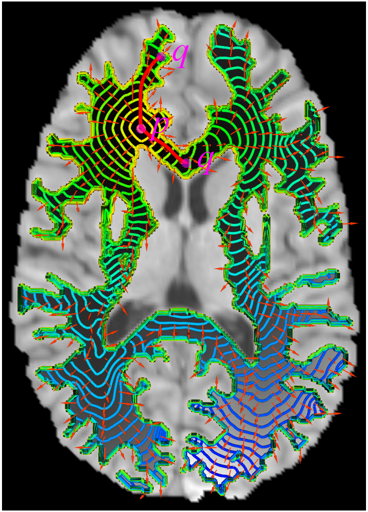
$$\min \int_0^1 f(x, v)dt$$

Algorithm:

1. Compute the time-of-arrival function u by propagating from p
2. Compute '∇u', from any point q integrate '∇u' backward to p

Output:

- Time-of-arrival function u



Geodesic Tracking on DTI

Given

- A starting point p
- A local cost function $f(x, v)$

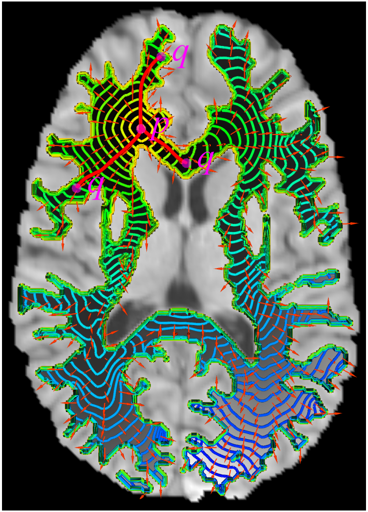
$$\min \int_0^1 f(x, v)dt$$

Algorithm:

1. Compute the time-of-arrival function u by propagating from p
2. Compute '∇u', from any point q integrate '∇u' backward to p

Output:

- Time-of-arrival function u
- The geodesics between p and any other point q



Related Work

- Parker et al. TMI 2002, proposed two cost different functions that related to the propagation normal and the principal eigenvector.
- O'Donnell et al. MICCAI 2002, use the inverse diffusion tensor as a Riemannian metric for the first time.

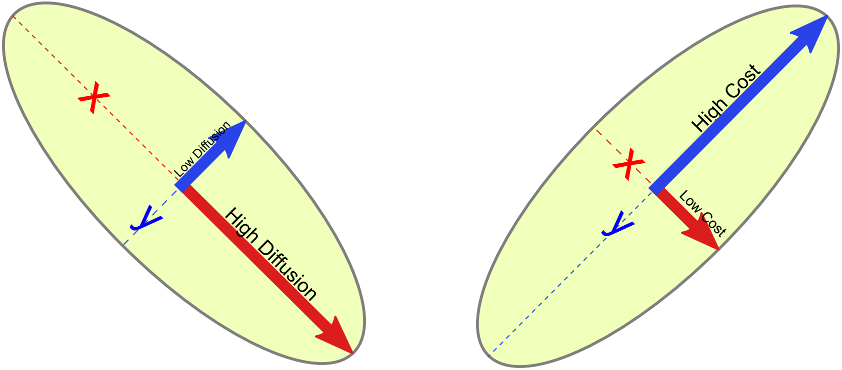
Related Work

- Parker et al. TMI 2002, proposed two cost different functions that related to the propagation normal and the principal eigenvector.
- O'Donnell et al. MICCAI 2002, use the inverse diffusion tensor as a Riemannian metric for the first time.
- Jackowski et al. MICCAI 2004
- Pichon et al. MICCAI 2005
- Fletcher et al. IPMI 2007
- Jbabdi et al. IJBI 2008
- Sepasian et al. MICCAI Workshop on Computational Diffusion MRI (CDMRI) 2009

Inverse Diffusion Tensor as Riemannian Metric

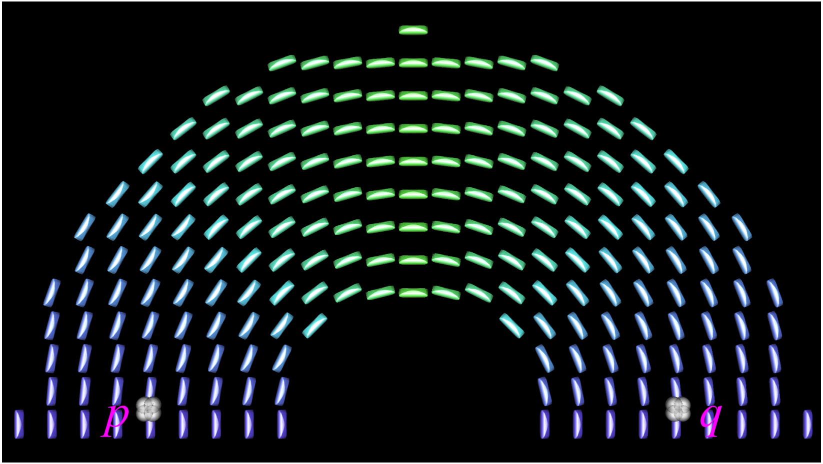
Diffusion Tensor:
$D = R\Sigma R^{-1}$

Riemannian Metric:
$g = D^{-1} = R\Sigma^{-1}R^{-1}$

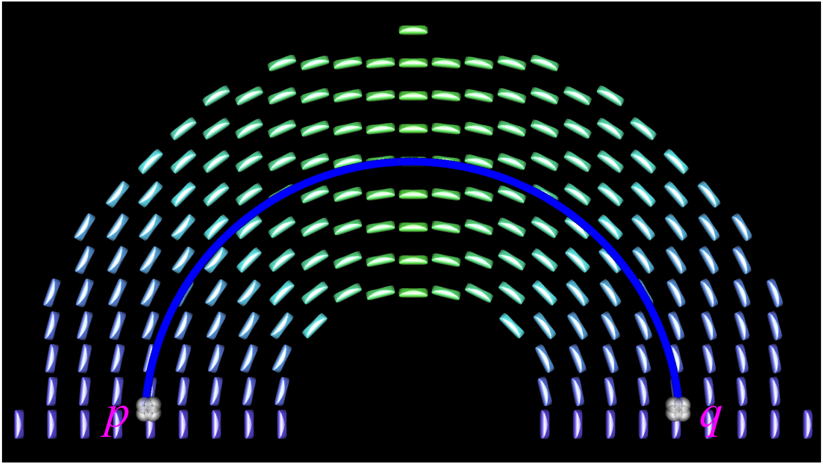


The corresponding cost function $f = \langle T(t), T(t) \rangle_g = T(t)^T D^{-1} T(t)$.

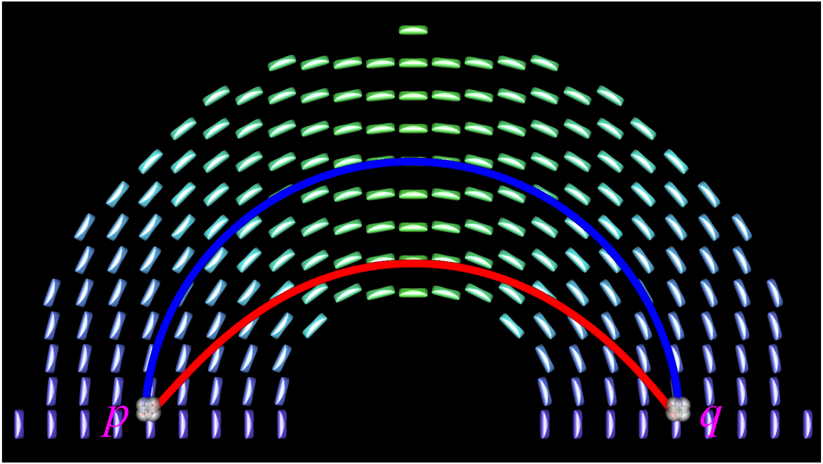
Motivation: Problem With Inverse Tensor Metric



Motivation: Problem With Inverse Tensor Metric



Motivation: Problem With Inverse Tensor Metric



Our Contribution: Improved Geodesic Tracking

Goal

We want the geodesics closely conform to the principal eigenvector field.

How?

Our Contribution: Improved Geodesic Tracking

Goal

We want the geodesics closely conform to the principal eigenvector field.

How?

We compute a spatially-varying scalar function,α, that modulates the inverse tensor field at each point.

Our Contribution: Improved Geodesic Tracking

Goal

We want the geodesics closely conform to the principal eigenvector field.

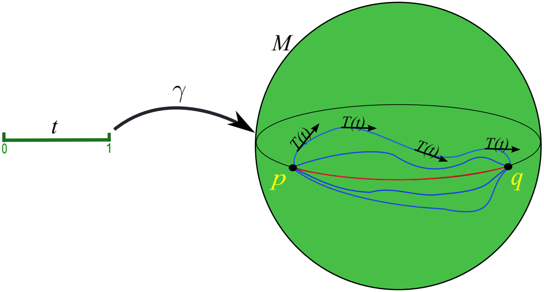
How?

We compute a spatially-varying scalar function,α, that modulates the inverse tensor field at each point.

$$g = e^{\alpha} D^{-1}$$

We call the function e^{α} the *metric modulating function*.

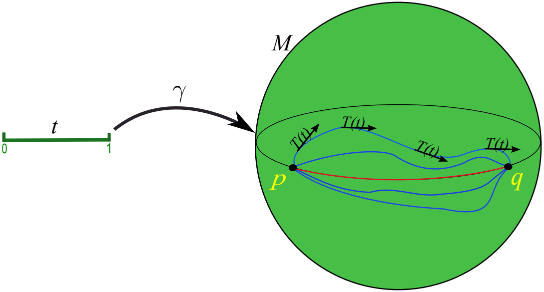
Background: Riemannian Manifold and Geodesic



On a Riemannian manifold, M, the geodesic between two points $p, q \in M$ is defined by the minimization of the energy functional

$$E(\gamma) = \int_0^1 \langle T(t), T(t) \rangle_g \, dt.$$

Background: Riemannian Manifold and Geodesic



On a Riemannian manifold, M, the geodesic between two points $p, q \in M$ is defined by the minimization of the energy functional

$$E(\gamma) = \int_0^1 \langle T(t), T(t) \rangle_g \, dt.$$

Euler-Lagrange Equation is

$$\nabla_T T = 0.$$

The symbol ∇ here is affine connection.

Background: Affine Connection

Affine connection, $\nabla_T Y$, is the derivative of a vector field Y in the direction of a vector field T.

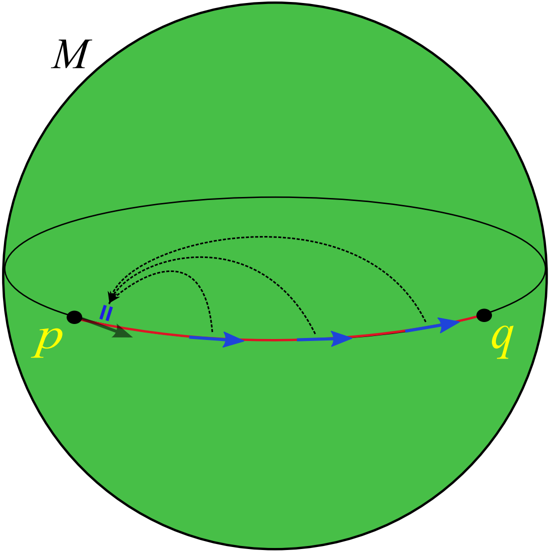
Background: Affine Connection

Affine connection, $\nabla_T Y$, is the derivative of a vector field Y in the direction of a vector field T.

In the special case of $Y = T$,

- $\nabla_T T$ measures how the vector field T bends along its integral curves.
- $\nabla_T T = 0$ means that tangent vectors, T, remain parallel if they are transported along the geodesic.

Background: Affine Connection $\nabla_T T = 0$



Compute the Geodesic by Solving the Eikonal Equation

A geodesic which minimize $E(\gamma(t))$ also satisfies the Eikonal equation.

$$\nabla u^T g^{-1} \nabla u = 1 \quad (1)$$

$$T(t) \propto g^{-1} \nabla u \quad (2)$$

u is the time-of-arrival function from a starting location p.

Compute the Geodesic by Solving the Eikonal Equation

A geodesic which minimize $E(\gamma(t))$ also satisfies the Eikonal equation.

$$\nabla u^T g^{-1} \nabla u = 1 \tag{1}$$

$$T(t) \propto g^{-1} \nabla u \tag{2}$$

u is the time-of-arrival function from a starting location p.

To compute the all geodesics from a starting point p, we need 2 steps:

1. Choose a starting point p, solve the Eikonal equation(1) for u with the initial condition $u(p) = 0$.
2. Compute $T(t)$ from (2) and then integrate $T(t)$ backward to p from any other points.

Geodesic Energy Functional

The traditional geodesic energy functional is

$$E(\gamma) = \int_0^1 \langle T(t), T(t) \rangle \, dt.$$

Geodesic Energy Functional

The traditional geodesic energy functional is

$$E(\gamma) = \int_0^1 \langle T(t), T(t) \rangle \, dt.$$

Now, since we scale the Riemannian metric by a positive function $e^{\alpha(x)}$, which results in the new geodesic energy functional

$$E_\alpha(\gamma) = \int_0^1 e^{\alpha(\gamma(t))} \langle T(t), T(t) \rangle \, dt.$$

Compute the New Geodesic Equation

We take the variational of the energy $E_\alpha(\gamma)$, results in the new geodesic equation

$$\operatorname{grad}\alpha \cdot \|T\|^2 = 2\nabla_T T + 2d\alpha(T) \cdot T,$$

where $\operatorname{grad}\alpha$ is the Riemannian gradient defined as $g^{-1}\left(\frac{\partial\alpha}{\partial x^1}, \frac{\partial\alpha}{\partial x^2}, \ldots, \frac{\partial\alpha}{\partial x^n}\right)$.

Simplify the New Geodesic Equation

The new geodesic equation is

$$\operatorname{grad}\alpha \cdot \|T\|^2 = 2\nabla_T T + 2d\alpha(T) \cdot T \tag{3}$$

Simplify the New Geodesic Equation

The new geodesic equation is

$$\operatorname{grad}\alpha \cdot \|T\|^2 = 2\nabla_T T + 2d\alpha(T) \cdot T \tag{3}$$

Without loss of generality, if we assume $\|T\| = 1$, then $\langle \nabla_T T, T \rangle = 0$.

Simplify the New Geodesic Equation

The new geodesic equation is

$$\operatorname{grad}\alpha \cdot \|T\|^2 = 2\nabla_T T + 2d\alpha(T)\cdot T \tag{3}$$

Without loss of generality, if we assume $\|T\| = 1$, then $\langle \nabla_T T, T\rangle = 0$.

Now taking the inner product with T on both sides of (3), we obtain

$$\begin{aligned} \langle \operatorname{grad}\alpha, T\rangle &= 2d\alpha(T) = 2\langle \operatorname{grad}\alpha, T\rangle \\ \Rightarrow \langle \operatorname{grad}\alpha, T\rangle &= 0 \qquad\qquad \Rightarrow d\alpha(T) = 0 \end{aligned}$$

Simplify the New Geodesic Equation

The new geodesic equation is

$$\operatorname{grad}\alpha \cdot \|T\|^2 = 2\nabla_T T + 2d\alpha(T) \cdot T \tag{3}$$

Without loss of generality, if we assume $\|T\| = 1$, then $\langle \nabla_T T, T \rangle = 0$.

Now taking the inner product with T on both sides of (3), we obtain

$$\begin{aligned} \langle \operatorname{grad}\alpha, T \rangle &= 2d\alpha(T) = 2\langle \operatorname{grad}\alpha, T \rangle \\ \Rightarrow \langle \operatorname{grad}\alpha, T \rangle &= 0 \qquad\qquad \Rightarrow d\alpha(T) = 0 \end{aligned}$$

Therefore, assuming $\|T\| = 1$, $d\alpha(T)$ must vanish, and we get

$$\operatorname{grad}\alpha = 2\nabla_T T.$$

Compute the Metric Modulating Function e^{α}

The simplified geodesic equation is

$$\operatorname{grad}\alpha = 2\nabla_T T.$$

Our goal is to **enforce** T **follow** the principal eigenvector field V.

Compute the Metric Modulating Function e^{α}

The simplified geodesic equation is

$$\operatorname{grad} \alpha = 2\nabla_T T.$$

Our goal is to **enforce** T **follow** the principal eigenvector field V.

Satisfying this property directly would result in the equation

$$\operatorname{grad} \alpha = 2\nabla_V V,$$

which we would need to solve for α.

Compute the Metric Modulating Function e^{α}

However, given an arbitrary V, there may not exist such a function, α, with the desired gradient field. Instead we minimize the functional

$$F(\alpha) = \int_M \|\operatorname{grad}\alpha - 2\nabla_V V\|^2 \, dx.$$

Compute the Metric Modulating Function e^{α}

However, given an arbitrary V, there may not exist such a function,α, with the desired gradient field. Instead we minimize the functional

$$F(\alpha) = \int_M \|\mathrm{grad}\,\alpha - 2\nabla_V V\|^2\,dx.$$

Euler-Lagrange Equation is

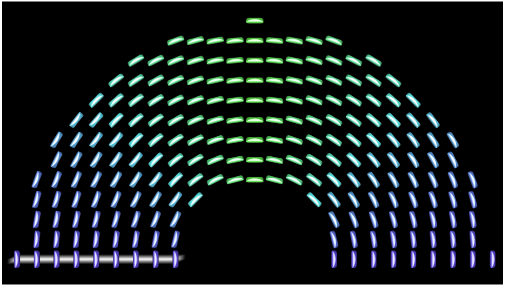
$$\Delta\alpha = 2\,\mathrm{div}\,(\nabla_V V)\,,$$

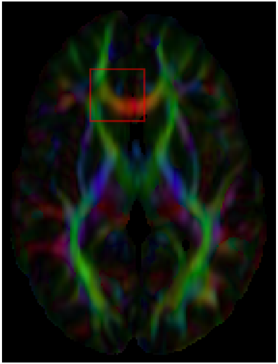
which is a Poisson equation on the Riemannian manifold. $\Delta\alpha$ is the Laplace-Beltrami operator on M.

The appropriate boundary conditions for this problem are the Neumann conditions,

$$\frac{\partial\alpha}{\partial\overrightarrow{n}} = \langle\mathrm{grad}\,\alpha, \overrightarrow{n}\rangle = \langle 2\nabla_V V, \overrightarrow{n}\rangle.$$

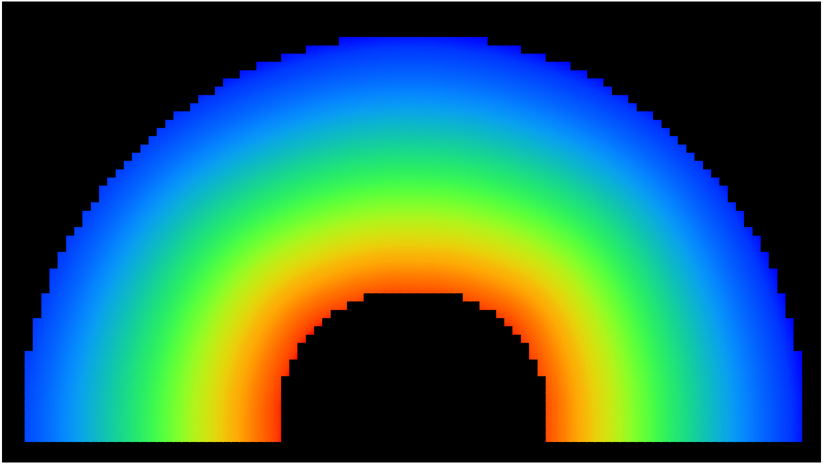
Data





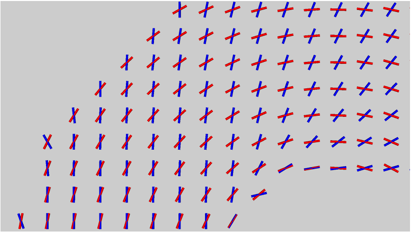
Solution α for the Annulus

$$\alpha(x) = -2\ln r(x) + C \qquad \operatorname{grad}\alpha = 2\nabla_V V$$

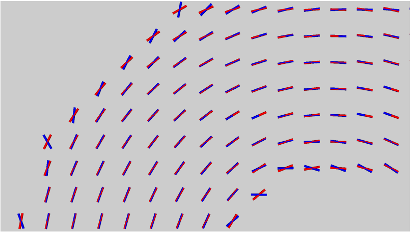


Comparing T with V

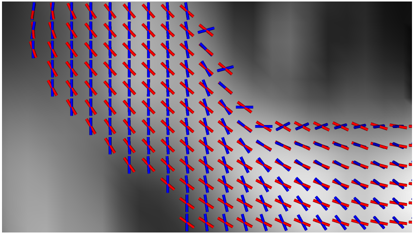
Geodesic Tangent Vectors — Principal Eigenvectors



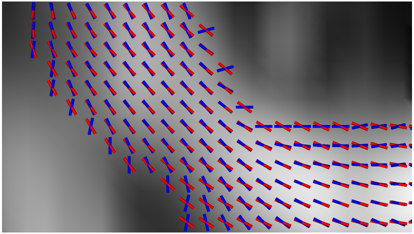
Inverse Tensor Metric



Our Modulated Metric

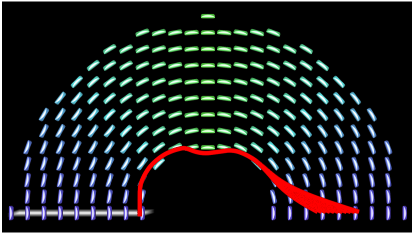


Inverse Tensor Metric

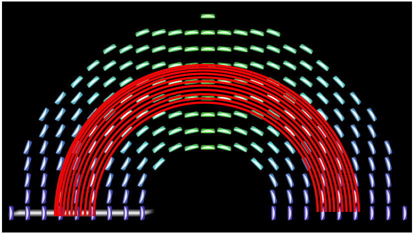


Our Modulated Metric

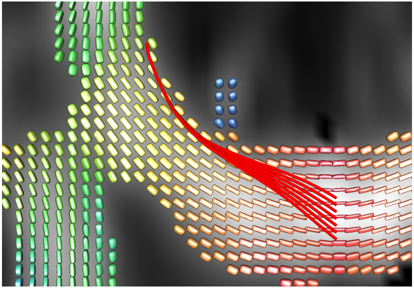
Geodesics



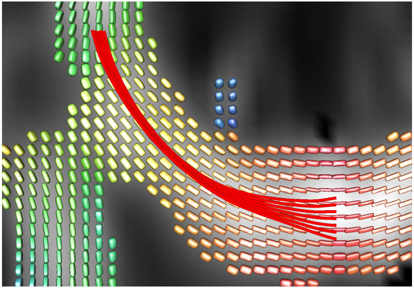
Inverse Tensor Metric



Our Modulated Metric



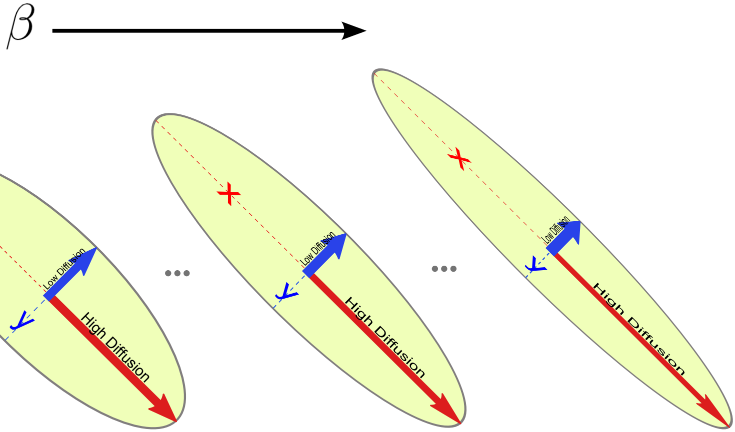
Inverse Tensor Metric



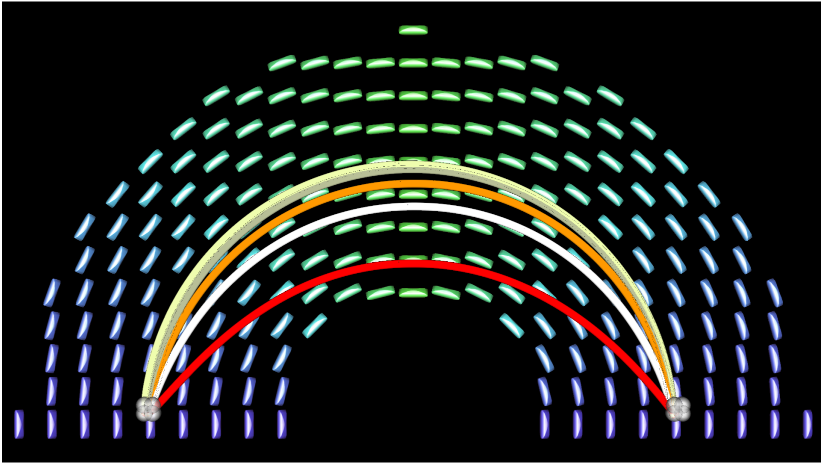
Our Modulated Metric

Sharpened Diffusion Tensor as Riemannian Metric

$$D' = |D|^{\frac{1}{3}} \left(\frac{D}{|D|^{\frac{1}{3}}} \right)^{\beta}, \qquad g = D'^{-1}$$

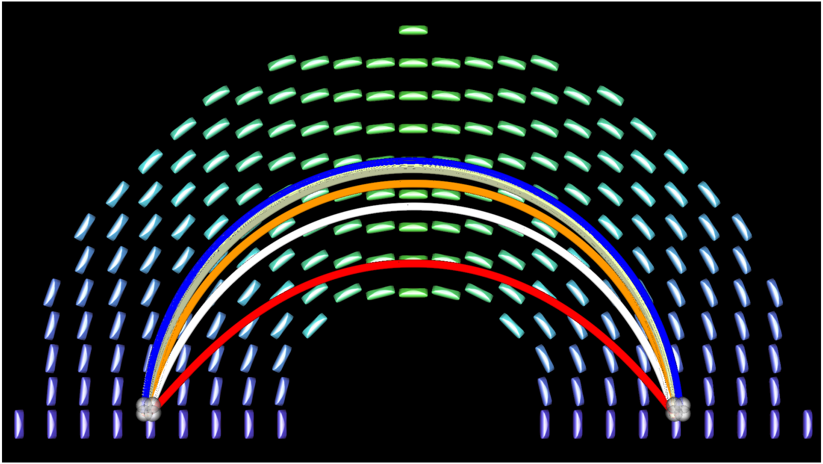


Sharpened Tensor Metric



Geodesics using sharpened tensor metric with different β.

Sharpened Tensor Metric



The blue curve is the ideal geodesic.

Summary and Future Work

To summarize, we compute a spatially-varying scalar function,α, that modulates the inverse tensor metric at each point. The geodesics computed using our modulated metric can faithfully follow the principal eigenvector field.

Summary and Future Work

To summarize, we compute a spatially-varying scalar function, α, that modulates the inverse tensor metric at each point. The geodesics computed using our modulated metric can faithfully follow the principal eigenvector field.

Future Work:

- Segment white matter tracts from the computed geodesics.
- Speed up the numerical α solver by parallel computing.
- Solve the edge-crossing issue.
- Extend the proposed method to HARDI.

Thank you!

Xiang Hao, Ross T. Whitaker, and P. Thomas Fletcher

Scientific Computing and Imaging Institute
University of Utah, Salt Lake City, UT