Prigarin, Hahn:

Stochastic Algorithms for White Matter Fiber Tracking and the Inference of Brain Connectivity from MR Diffusion Tensor Data

Sonderforschungsbereich 386, Paper 408 (2005)

Stochastic Algorithms for White Matter Fiber Tracking and the Inference of Brain Connectivity from MR Diffusion Tensor Data

Sergei M. Prigarin *
Institute of Computational Mathematics
and Mathematical Geophysics †

Klaus Hahn ‡
Institute of Biomathematics and Biometry
GSF-National Research Centre for Environment and Health §

December 21, 2004

Abstract

We consider several stochastic algorithms for fiber tracking and compute the connectivity matrix from data obtained by magnetic resonance diffusion tensor imaging of the living human brain.

Key words: *diffusion tensor imaging (DTI), stochastic fiber tracking, brain connectivity.*

*e-mail: smp@osmf.sscc.ru
†pr. Lavrentieva 6, Novosibirsk, 630090, Russia
‡e-mail: hahn@gsf.de
§Ingolstädter Landstrasse 1, 85764 Neuherberg/München, Germany

1 Statement of the problem

Let V be a bounded set in the 3-dimensional space, $V \subset R^3$, and $T(x)$, $x \in V$, be a given tensor field in V. All tensors $T(x)$ (i.e. 3*3-matrices) are assumed to be symmetric and positive definite. A problem is to generate a "connectivity map" in volume V "coordinated" with the tensor field and to find tracks that connect different parts of the volume V. A track is considered as a "compatible" one with the tensor field if principal eigenvectors of tensors are tangents or "nearly tangents" to the track. Below we consider two types of algorithms:

1. "Single tracking": to construct a curve "compatible" with the tensor field T starting from a fixed point x_0 in volume V.

2. "Connectivity matrix": In this case, points $x_1, \ldots, x_M$ in volume V are given and they are supposed to be extremities of tracks. The aim is to calculate the connectivity matrix of weights $W = (W_{ij})$, $i, j = 1, \ldots, M$, that can be interpreted as "probabilities" of existence of a track connecting points x_i, x_j.

Such problems arise in Diffusion Tensor Magnetic Resonance Imaging (DT-MRI), where, among others, the course and the connections of the nerve fibers in the brain are studied. The tensor field indicates a local orientation of nerve bundles. The goal is to decide which parts of grey matter are connected with each other by white matter nerve fibers and to estimate the location of the fibers. The described problems require additional mathematical specifications (for example, it is desirable to have a detailed description of what is the "compatibility" of the tensor field and the track). The partial volume effects fiber crossing and bifurcation are not-well recognized because of the poor resolution of the tensor field, in addition, noise of different nature introduces more uncertainty to the problem. The uncertainty makes it reasonable to use stochastic methods in addition to deterministic ones to find probable tracks. Using Monte Carlo methods to simulate a set of stochastic tracks we expect to get an appropriate connectivity map. The stochastic approach in studying the brain connectivity using DT-MRI is a significant direction of the present-day research [1, 2]. In this paper we consider several stochastic algorithms (including some new methods) for fiber tracking and the computation of the connectivity matrix.

2 Several Algorithms of Stochastic Fiber Tracking

Below we assume that the tensor field $T(x)$ is defined on a grid of integer numbers and to interpolate the field we simply take the value in the nearest grid point. A fixed point x_0 is a starting point for tracks. By $e(T)$ we denote the principal eigenvector of tensor T and by $\{b\}$ we denote the normalized vector $\frac{b}{\|b\|}$. Tracks will be described by a sequence of points $x_n \in V$.

2.1 Tracking on the basis of the field of the principal eigenvectors

Following the main direction field by a deterministic differential equation was one of the first tracking methods in DTI, we extend this approach by additional random perturbations.

Algorithm E.

$$v_0 = \{e(T(x_0))\},$$

$$x_n = x_{n-1} + v_{n-1}\Delta t + \sqrt{\Delta t}\sigma\varepsilon_n,$$

$$v_n = \{e(T(x_n))\}, \quad (v_{n-1}, v_n) > 0.$$

Here ε_n are independent standard normal random vectors and Δt, σ are parameters of the algorithm. The step parameter Δt is assumed to be less than 1 (remind that 1 is the step of the grid were the tensor field is defined). For this algorithm and the algorithms presented below the parameter σ defines the intensity of artificial noise that is added to generate stochastic tracks. If $\sigma = 0$, then the algorithms become deterministic and the tracks are defined in an unique non-random way.

The directions of principal eigenvectors can be unstable under noisy fluctuations of the tensors' elements (especially, when some of eigenvalues are close to each other) or they can be simply undefined in case of multiple eigenvalues. This evident consideration encouraged us to develop sophisticated methods, which take into account more characteristics of the tensor field than only the directions of principal eigenvectors.

2.2 A method of tracking using the tensor field

Algorithm T1:

$$v_0 = \{e(T(x_0))\},$$

$$x_n = x_{n-1} + v_{n-1}\Delta t + \sqrt{\Delta t}\sigma\varepsilon_n,$$

$$v_n = \{T^k(x_n)v_{n-1}\}.$$

Here Δt, σ and k are parameters of the algorithm ($\Delta t < 1$, $k > 0$). In Monte Carlo experiments we set $k = 1$. This algorithm generates more "smooth and stable" tracks in comparison with the previous method (see. Fig.1).

Figure 1: Examples of tracks generated by Algorithm E (narrow line) and Algorithm T1 (wide line) for $\Delta t = 0.1$ with the same starting point and without random noise ($\sigma = 0$).

2.3 Tracking on the basis of a 2nd order SDE

Algorithm T2:

$$v_0 = c_0\{e(T(x_0))\}, \quad a_0 = 0,$$

$$a_{n+1} = \{T^k(x_n)v_n\} - c_1 v_n,$$

$$v_{n+1} = v_n + \Delta t a_{n+1} + \sqrt{\Delta t}\sigma\varepsilon_{n+1},$$

$$x_{n+1} = x_n + v_{n+1}\Delta t.$$

Here Δt, σ, c_0, c_1 and k are parameters of the algorithm ($\Delta t < 1$, $k > 0$). In Monte Carlo experiments we set $k = 1$. This method involves not only the principal eigenvectors but the whole tensors.

If $c_0 = 0$ and $\sigma \neq 0$, then the track "slowly starts" from x_0 in a random direction, but then it "turns according to the main diffusion stream".

The tracks strongly depend on the "braking" parameter c_1 (see Fig.2).

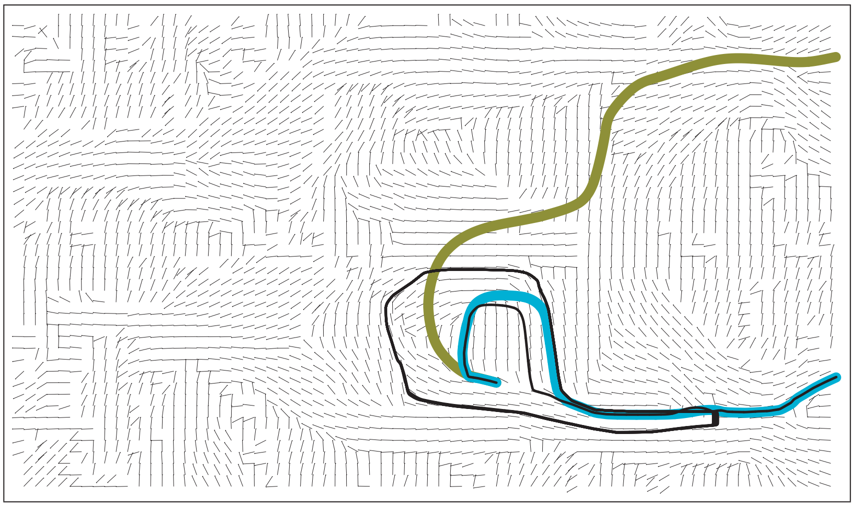

Figure 2: Examples of tracks generated by Algorithm T2 with different values of "braking" parameter c_1: $c_1 = 1$ (dark wide line), $c_1 = 3$ (light wide line), $c_1 = 5$ (narrow black line). The values of other parameters: $\Delta t = 0.1$, $c_0 = 1$, $\sigma = 0$. The vector field and the starting point for tracking are the same as for Figure 1.

This algorithm can be interpreted as a numerical method to solve the Cauchy problem for the following stochastic differential equaion

$$\xi'(t) = A(\xi)\xi(t) + \Sigma\varepsilon(t),$$

where

$$\xi(t) = \begin{bmatrix} x(t) \\ x'(t) \end{bmatrix},$$

$$A(\xi) = \begin{bmatrix} 0 & I \\ 0 & aT^k(x) - c_1 I \end{bmatrix},$$

$$\Sigma = \begin{bmatrix} 0 & 0 \\ 0 & \sigma I \end{bmatrix},$$

$\xi(t)$ is a random 6-dimensional vector process, A, Σ are 6*6-matrices, $\varepsilon(t)$ is a 6-dimensional vector of the Gaussian white noise, and I is the three-dimensional identity matrix.

One of the reasons to construct this algorithm on the basis of the stochastic differential equation was to check in future if the up-to-date results in the theory of boundary value problems for stochastic differential equations [3] can be used to build tracks with fixed starting and ending points.

2.4 A method of stochastic tracking from Literature

Algorithm H [3]:

$$x_{n+1} = x_n + \mu\Omega_n,$$

$$\Omega_n = \{\lambda d_n + \Omega_{n-1}\}, \quad (\Omega_{n-1}, \Omega_n) \geq 0,$$

$$d_n = T^\alpha(x_n))r_n.$$

Here r_n are independent random vectors uniformly distributed over a unit sphere and μ, α, λ are parameters of the algorithm ($0 < \lambda$). The authors recommend $\mu = 0.75$, $\alpha = 2$.

In our opinion, one of the main disadvantages of the method is that there is no deterministic limit of the algorithm: when the stochastic part of the algorithm vanishes ($\lambda = 0$), a track turns into a straight line.

If track points x_n are assumed to be coordinates of a solid in space, then the values of v_n for algorithms E, T1, T2 can be interpreted as velocity of the solid, and the values of a_n can be interpreted as acceleration of the solid for algorithms T2.

Bundles of stochastic tracks for different algorithms are shown on Figs.3-6.

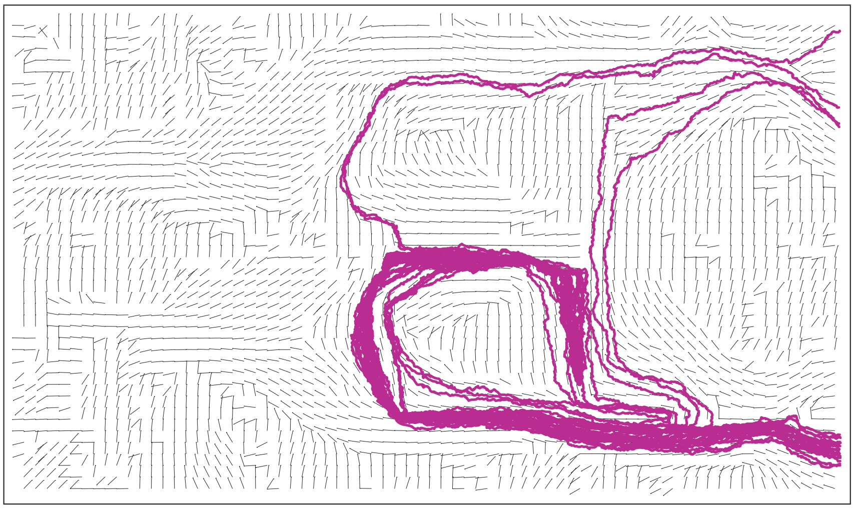

Figure 3: A bundle of 25 random tracks generated by Algorithm E with parameters $\Delta t = 0.1$, $\sigma = 0.2$. The vector field and the starting point for tracking are the same in Figures 3-6.

3 Computation of the connectivity matrices

Assume now, that the coordinates of M terminal points $x_1, \ldots, x_M$ in volume V are fixed, the problem is then to calculate the "connectivity matrix" of weights W_{ij}, $i, j = 1, \ldots, M$, which "correspond to probabilities" of existence of a fiber connecting points x_i, x_j (see Section 1).

To calculate the connectivity matrix $W = (W_{ij})$ we used the following algorithm. A large number of random tracks is started every terminal point. If a track passes close enough to another terminal point, then the track is stopped and it is announced that the terminal points are connected by the track. The weights W_{ij} are computed by the formula $W_{ij} = N_{ij}/N$. Here

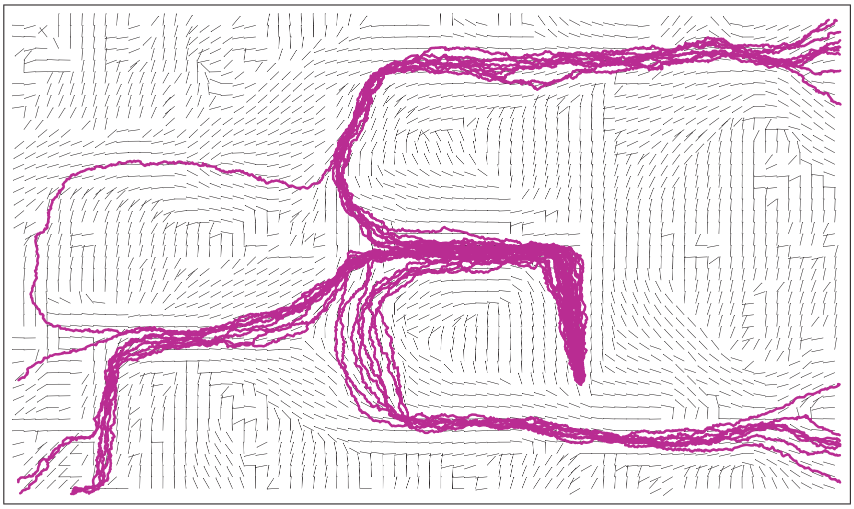

Figure 4: A bundle of 25 random tracks generated by Algorithm T1 with parameters $\Delta t = 0.1$, $\sigma = 0.2$. The vector field and the starting point for tracking are the same in Figures 3-6.

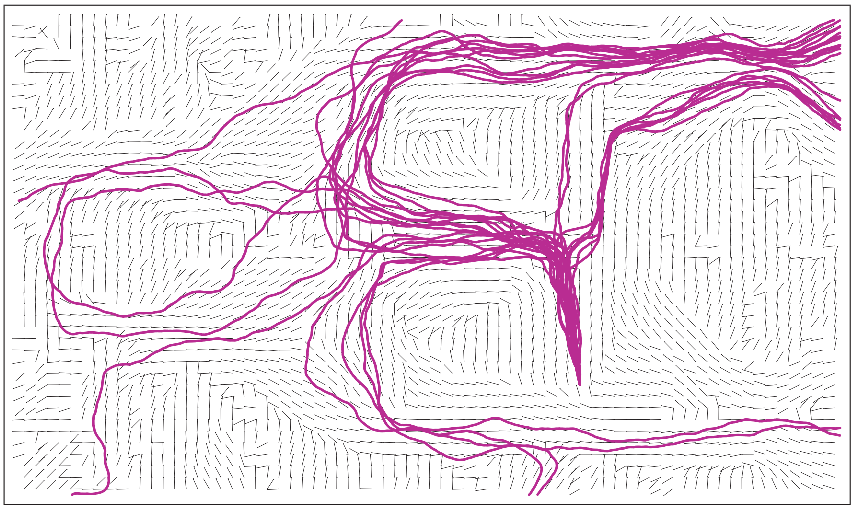

Figure 5: A bundle of 25 random tracks generated by Algorithm T2 with parameters $c_0 = 1$, $c_1 = 2.0$, $\Delta t = 0.1$, $\sigma = 0.1$. The vector field and the starting point for tracking are the same in Figures 3-6.

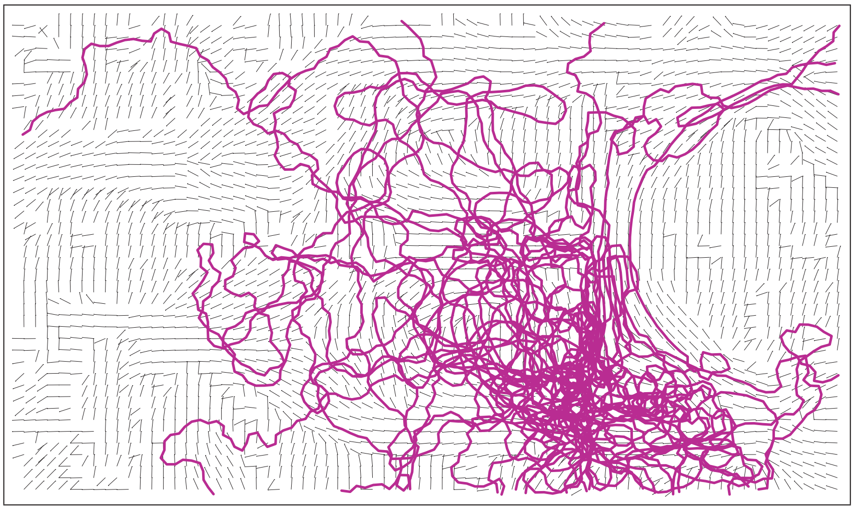

Figure 6: A bundle of 25 random tracks generated by Algorithm H with parameters $\lambda = 1000000.0$, $\alpha = 2$, $\mu = 0.75$. The vector field and the starting point for tracking are the same in Figures 3-6.

N_{ij} is the number of random tracks that connect points x_i and x_j, and N is the total number of tracks connecting terminal points (some of the simulated random tracks do not connect any pair of terminal points).

The connectivity matrix W was computed for 22 terminal points in the volume with a tensor field $T(x)$. The results for different algorithms of random tracking are presented in Figures 7–13. For every numerical experiment $22 * 10000$ random tracks were simulated. The Figures demonstrate the influence of different noise levels on the connectivity matrix. The different methods gave reasonable and similar results (may be except Hagmann's algorithm).

To make the analysis of connectivity more detailed, we estimated the probabilities P_{ij} to hit terminal point x_j for a track starting from point x_i.

In addition, to get approximate information about the location of a "real fiber" between two voxels x_1 and x_2 (which are assumed to be connected), we propose to compute the "fiber density function" $D(x)$ in volume V by the following algorithm. First, the values $D(x)$ are equal to zero for every grid point x in volume V. Then, a large number of random tracks starting from terminal points x_1 and x_2 are simulated, and for every track that crosses voxel x (once or several times) the value $D(x)$ increases by 1. So, $D(x)$ is simply

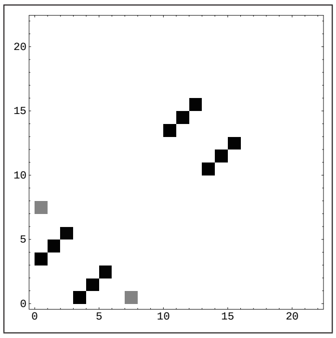

Figure 7: Example of connectivity matrix: algorithm E with parameters $\Delta t = 0.1$, $\sigma = 0$. The vector field is the same for Figures 7–13.

a number of tracks crossing the voxel x. After normalization, $D(x)$ can be interpreted as a track density distribution. The ridge of $D(x)$ connecting x_1 and x_2 defines the connection with highest probability. Methods to model this maximizing connecting track are in progress.

References

[1] Hagmann, P., Thiran, J.-P., Jonasson, L., Vandergheynst, P., Clarke, S. Maeder, P., and Meuli, R. DTI mapping of human brain connectivity: statistical fibre tracking and virtual dissection// NeuroImage - 2003. - V.19, N.3. - P.545-554

[2] Tench, C.-R., Morgan, P.-S., Blumhardt, L.-D., Constantinescu, C., Improved White Matter Fiber Tracking using Stochastic Labeling// Magn Res Med - 2002. - P.677-683

[3] Prigarin, S.M., Fedchenko, N.V. Solving of boundary value problems for linear systems of stochastic differential equations// Siberian journal of numerical mathematics - 2004. - V.7, N.4. - P.301-318

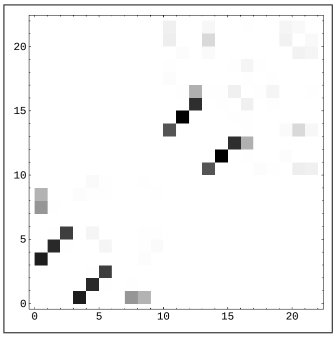

Figure 8: Example of connectivity matrix: algorithm E with parameters $\Delta t = 0.1$, $\sigma = 0.25$.

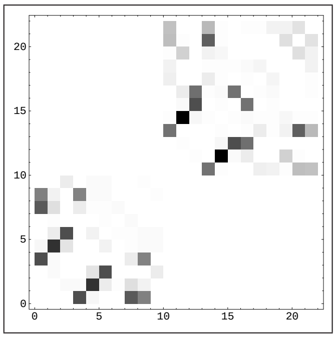

Figure 9: Example of connectivity matrix: algorithm E with parameters $\Delta t = 0.1$, $\sigma = 0.5$.

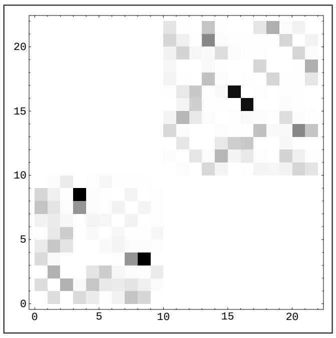

Figure 10: Example of connectivity matrix: algorithm E with parameters $\Delta t = 0.1$, $\sigma = 1.0$.

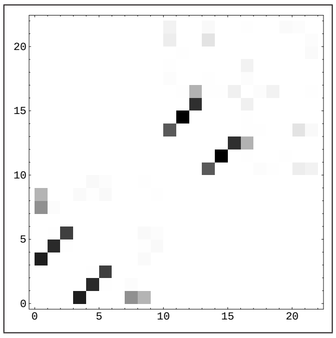

Figure 11: Example of connectivity matrix: algorithm T1 with parameters $\Delta t = 0.1$, $\sigma = 0.25$.

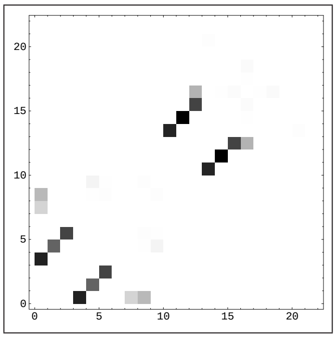

Figure 12: Example of connectivity matrix: algorithm T2 with parameters $\Delta t = 0.1$, $\sigma = 0.1$, $c_0 = 1.0$, $c_1 = 2.0$.

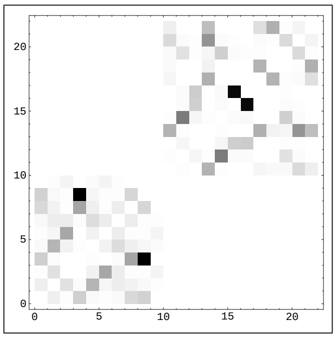

Figure 13: Example of connectivity matrix: algorithm H with parameters $\mu = 0.75$, $\alpha = 2$, $\lambda = 1000000$. The vector field is the same for Fig. 7–13.