RECONSTRUCTION OF CONVEX POLYNOMIAL DIFFUSION MRI MODELS USING SEMI-DEFINITE PROGRAMMING
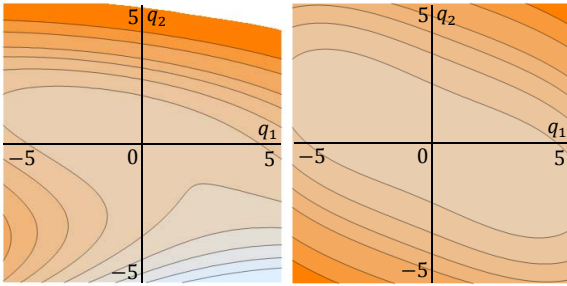
Tom Dela Haije[1], Andrea Fuster[1], and Luc Florack[1]

[1]*Mathematics and Computer Science, Eindhoven University of Technology, Eindhoven, Noord-Brabant, Netherlands*

Purpose: Polynomial basis functions form the backbone of many current signal models in diffusion MRI. These models typically impose constraints like positive semi-definiteness and symmetry, combining heuristics, physically motivated assumptions, and computational complexity arguments. In order to incorporate these constraints the reconstruction algorithms used to determine the polynomial coefficients have to be adapted accordingly. Here we consider convexity of a general polynomial model as an additional constraint, which among other things is one of the requirements when performing (Finsler) geodesic tractography.[1] Since verifying the convexity of a polynomial is generally NP-hard, we consider a subset of all convex polynomial functions called sum-of-squares polynomials, following the approach outlined by Magnani et al.[2] In this work we explain their method and show that the resulting optimization procedure is feasible for use in typical diffusion MRI reconstruction tasks.

Theory: We will consider a degree $2m$ polynomial diffusion MRI model $P_y(q)$ that is linear in its coefficients y and where $q \in \mathbb{R}^3$ is a gradient vector that parameterizes the measurement space in diffusion MRI. $P_y(q)$ is convex if and only if its Hessian $\nabla^2 P_y$ is positive semi-definite. Our goal is to find y such that the measurements $\{\hat{P}(q_i)\}_{i=1}^N$ are best represented by a convex $P_y(q)$ in the least squares sense, and so we want to minimize the residual sum of squares $\|r_y\|^2 := \sum_{i=1}^N [P_y(q_i) - \hat{P}(q_i)]^2$ over y subject to $\nabla^2 P_y(q) \succcurlyeq 0$, where $\succcurlyeq 0$ indicates positive semi-definiteness. There are no known techniques to tractably solve this problem, and so following Magnani et al.[2] we impose the stronger restriction that $\nabla^2 P_y(q)$ be the sum of a number of squared polynomials, which directly guarantees convexity of $P_y(q)$. The sum-of-squares constraint is formulated as a linear matrix inequality $V(y) \succcurlyeq 0$, where the associated Gram matrix $V(y)$ is any matrix such that for a given vector $e(s,q)$ of all monomials up to degree m in $s \in \mathbb{R}^3$ (order 1) and q (up to order $m-1$) we have $s^T \cdot \nabla^2 P_y(q) \cdot s = e(s,q)^T \cdot V(y) \cdot e(s,q)$.

Methods: We introduce a bound τ on the objective function, i.e., $\|r_y\|^2 \leq \tau$, and define an analogous optimization problem with a linear objective function:

Fig 1. Left: The ground truth data $\hat{P}(q) = (q_1)^2 + 4(q_2)^2 + 3q_1 q_2 + (q_2)^3$, sampled equidistantly on a Cartesian grid from -5 to 5 in steps of 0.5. Right: Results of the reconstruction algorithm fitting a fourth order convex-constrained polynomial $P_y(q)$. Displayed are the level sets $\{-200, -175, \ldots, 200\}$, with colors indicating the approximate function value (light blue – low; orange – high).

$$\min_{\tau, y} \tau \quad \text{subject to} \quad \begin{pmatrix} V(y) & 0 & 0 \\ 0 & I & r_y \\ 0 & (r_y)^T & \tau \end{pmatrix} \succcurlyeq 0. \qquad (1)$$

Since both the constraint and the objective function are linear in the optimization parameters, this is a convex optimization problem called a semi-definite programming problem.[3] For a given model and data we write this problem to an SDPA file, which is a standard format supported by the majority of solvers. For all experiments shown we use the SDPA solver[4] with default settings for the optimization. An illustration of an optimized fourth order convex-constrained polynomial (with even order monomials) is shown in Fig. 1, where we took $q \in \mathbb{R}^2$ for simplicity.

As proof of concept we further consider the diffusion model $\hat{P}(q_i) = -\log E(q_i)$, $\|q\|^2 \sim b$, used in the context of Finsler tractography by Melonakos et al.[1], where E is the normalized diffusion-weighted signal in a given voxel. $P_y(q)$ is given by an even order polynomial with even order monomials. The reconstruction is tested on 10 random voxels from a data set of the Human Connectome Project[5] (subject ID 100307) consisting of 270 q_i evenly distributed on shells with $b \approx 1000, 2000, 3000 \, s/mm^2$. The baseline image is estimated by averaging the 18 available $b \approx 0 \, s/mm^2$ images.

Results: Optimization for a fourth order model required $2.58 \pm 0.03 \, s/voxel$ (mean $\pm$ std) on an Intel i7-2600 CPU clocking at 3.40GHz (8 cores). Reducing the number of used measurements to 60 per voxel brought the running time for this model down to $0.46 \pm 0.03 \, s/voxel$. A second order convex-constrained model took $2.18 \pm 0.04 \, s/voxel$ on the full data, while a sixth order constrained model could be reconstructed in $5.71 \pm 0.18 \, s/voxel$. When dropping the convexity constraint the algorithm converged in similar time spans (due to the relatively small size of $V(y)$ in the constraint). The per-voxel overhead is around $0.25 \, s$ in the current implementation. The impact of the convexity constraint on the residual sum of squares (RSS) was typically small, suggesting that for this data convexity may be a reasonable assumption. Fig. 2 shows $\hat{P}(q_i)$ for the $b \approx 3000 \, s/mm^2$ shell, together with the estimated profiles produced by the reconstruction algorithm.
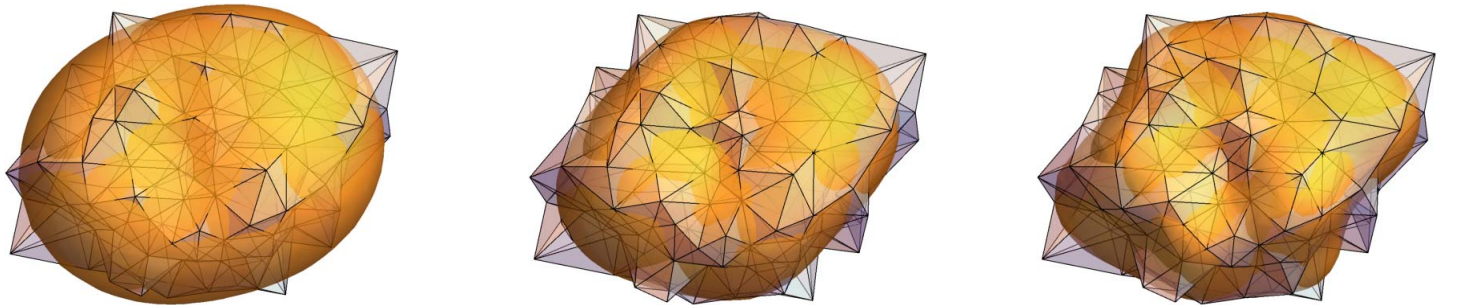

Fig 2. Left: Second order convex-constrained model, with a residual sum of squares RSS = 10.33. Middle: Fourth order convex-constrained model, RSS = 4.82. Right: Sixth order convex-constrained model, RSS = 4.40. The plots show $\hat{P}(q_i)$ for the $b \approx 3000 \, s/mm^2$ shell (purple triangulated mesh, with vertices $\hat{P}(q_i) \, q_i/\|q_i\|$), together with the estimated profiles (smooth orange surface, defined as the set $\{P_y(q) \frac{q}{\|q\|} \mid b = 3000 \, s/m^2\}$, comparable to an ADC profile). The considered voxel is located in the occipital lobe.

Discussion: Generally the algorithm scales somewhat poorly in both the number of measurements per voxel and in model complexity, as both significantly increase the problem size of the semi-definite program. The timings presented here show that for a fairly large data set (270 measurements) and a moderately complex model (21 degrees of freedom for the fourth order polynomial model), reconstruction of a full data set ($\sim 750,000$ relevant voxels) would take around three weeks. The same model reconstructed on a typical clinical data set, consisting of 60 measurements per voxel and 300,000 voxels, would take roughly a day. The overhead per voxel can be essentially eliminated by fixing the model choice, which would reduce the computation time for a clinical scale data set to half a day. Further improvements can be gained by exploiting parallelization of the reconstruction and by fine-tuning the solver settings, both of which are expected to result in significant speed-ups.

Conclusion: In this work we apply the fitting techniques described by Magnani et al.[2] to reconstruct a polynomial diffusion MRI model subject to convexity constraints. Convexity is enforced by requiring that the Hessian function of the polynomial model is sum-of-squares. We show that the resulting model fitting procedures are feasible for diffusion MRI reconstruction tasks. Future work will consist of improving the implementation and applying convex-constrained models in existing geodesic tractography.

References: [1] Melonakos et al. *IEEE Trans. Pattern Anal. Mach. Intell.* 2008; 30(3):412–23. [2] Magnani et al. *IEEE 44th CDC-ECC* 2005; 1672–77. [3] Vandenberghe et al. *SIAM Rev.* 1996; 38(1):49–95. [4] Yamashita et al. *Tokyo Tech Report* 2010; B-460. [5] Van Essen et al. *NeuroImage* 2013;80:62. [6] B. Borchers. *Optimization Meth. & Soft.* 1999:11(1-4):613–23.