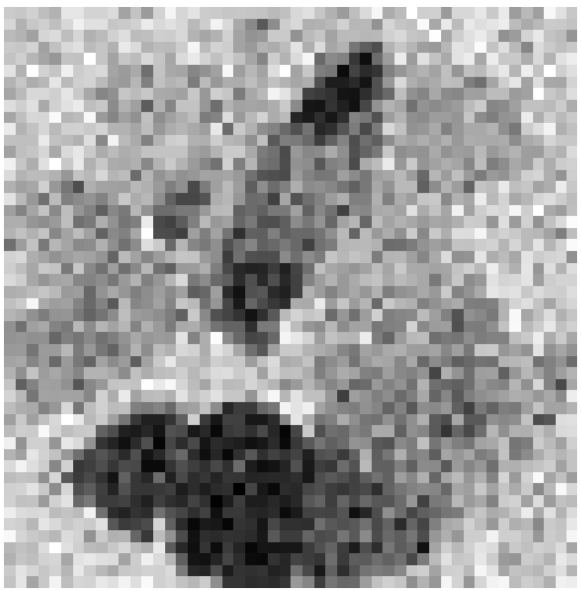
Denoising

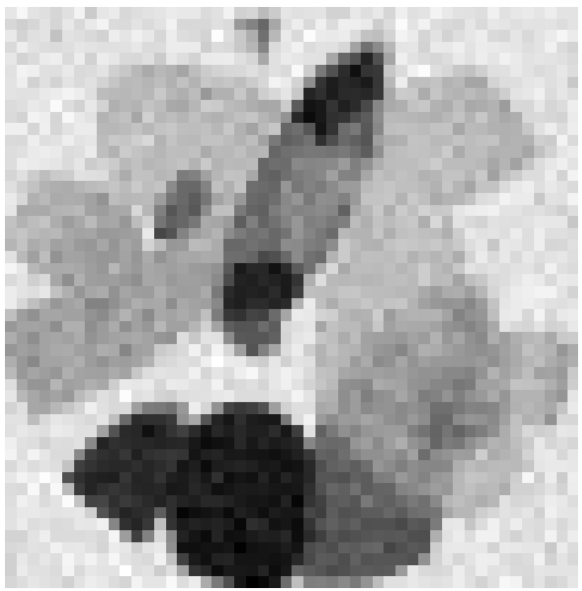
Adaptive non-local means denoising method

SNR = 10



SNR = 10 Denoised

SNR = 30



[Manjón, Coupé et al., 2010]

Multi-Tensor Model

The Multi-Tensor (MT) generative model:

$$S(b, g) = \sum_{i=0}^{N-1} f_i e^{-b g^t D_i g}$$

where
b, g are the gradient b-value and unit orientation,
f_i is the volume fraction of the i^{th} compartement,
D_i is the symmetric rank-2 tensor of the i^{th} compartement,
N is the number of compartement,

$S(b, g)$ is the signal generated by the MT model with parameters (f, D, N) at (b, g).

MT Fitting

- Constrained MT model:

$$S^C_{b,g} = f_0 e^{-bD_0} + \sum_{i=1}^{3} f_i e^{-bg^t D_i g}$$

constraints: 4 compartement (1 isotropic, 3 prolate).

MT Fitting

- Constrained MT model:

$$S^C_{b,g} = f_0 e^{-bD_0} + \sum_{i=1}^{3} f_i e^{-bg^t D_i g}$$

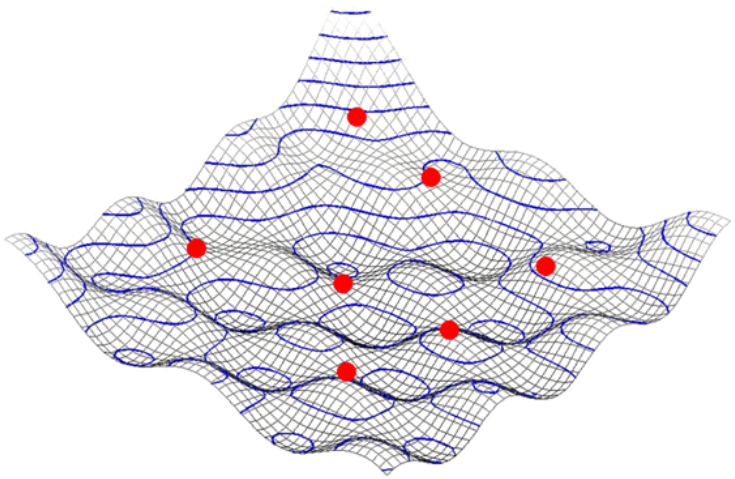
  constraints: 4 compartement (1 isotropic, 3 prolate).

- Fit by minimizing $\sum_k (S^c_{b_k,g_k} - Y_{b_k,g_k})^2$

  where Y is the measured diffusion signal.

Particle Swarm Optimization (PSO)

PSO is a stochastic global optimization method,

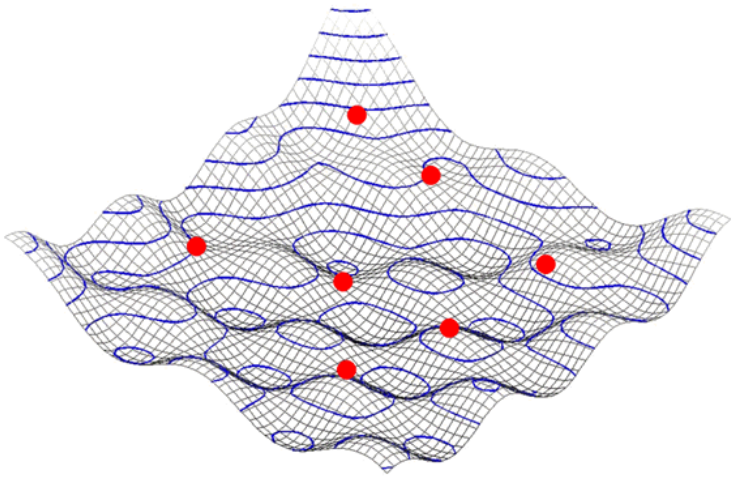


[http://www.itm.uni-stuttgart.de/research/pso_opt/bilder/pso.gif]

Particle Swarm Optimization (PSO)

PSO is a stochastic global optimization method,

- Np random position in the parameters space (called particles),

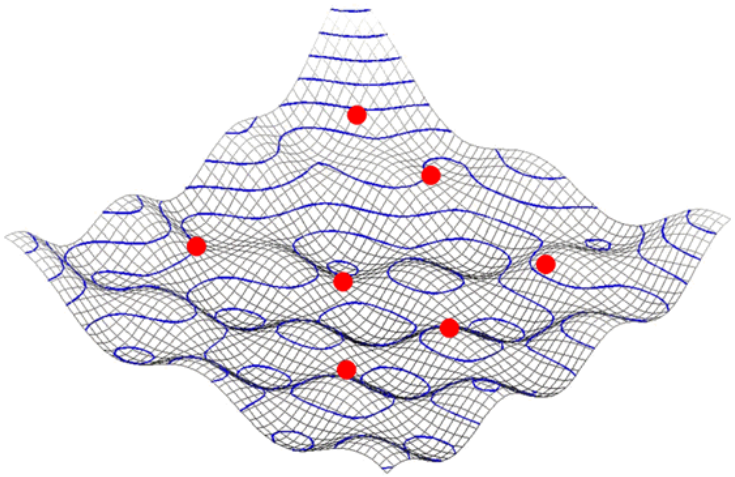


[http://www.itm.uni-stuttgart.de/research/pso_opt/bilder/pso.gif]

Particle Swarm Optimization (PSO)

PSO is a stochastic global optimization method,

- *Np* random position in the parameters space (called particles),
- Updated at every timestep by a velocity,

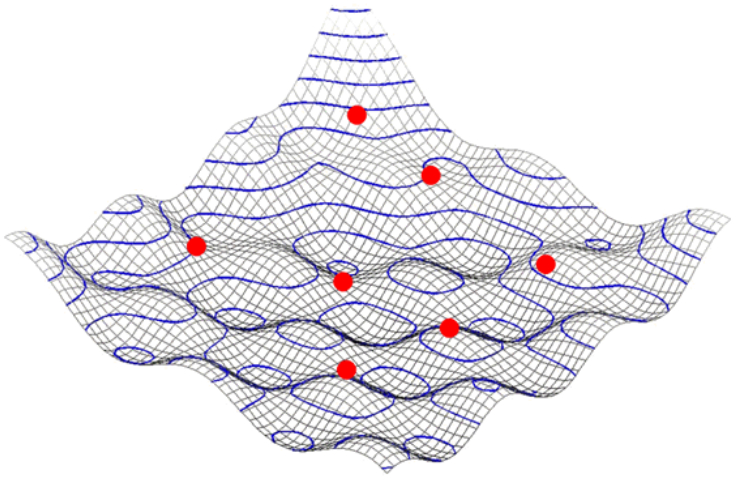


[http://www.itm.uni-stuttgart.de/research/pso_opt/bilder/pso.gif]

Particle Swarm Optimization (PSO)

PSO is a stochastic global optimization method,

- *Np* random position in the parameters space (called particles),
- Updated at every timestep by a velocity,
- Computed with information from the whole swarm of particles.



[http://www.itm.uni-stuttgart.de/research/pso_opt/bilder/pso.gif]

PSO

- Swarm: $\Omega_j^{t+1} = \Omega_j^t + v_j^{t+1}$

[Kennedy and Eberhart, 1995]

PSO

- Swarm: $\Omega_j^{t+1} = \Omega_j^t + v_j^{t+1}$
- Velocity: $v_j^{t+1} = wv_j^t + \phi_p r_p(p_j^t - \Omega_j^t) + \phi_g r_g(g^t - \Omega_j^t)$

[Kennedy and Eberhart, 1995]

PSO

- Swarm: $\Omega_j^{t+1} = \Omega_j^t + v_j^{t+1}$
- Velocity: $v_j^{t+1} = wv_j^t + \phi_p r_p(p_j^t - \Omega_j^t) + \phi_g r_g(g^t - \Omega_j^t)$

- where
  w, ϕ_p and ϕ_g are user tuned parameters,
  p_j^t is the j^{th} particle's best known position at iteration t,
  g^t is the swarm's best known position at iteration t,
  r_p, r_g are uniform random variable.

[Kennedy and Eberhart, 1995]

Model Complexity

3 Tensor Fit | Angular Threshold | Neighborhood Threshold