Modeling protein diffusion-like motion over cell membranes

M.Malyutov, T. Korobeinikova,
O.Bayborodin, R. Protassov

Northeastern University
Mathematics Department

Main Ideas

- We propose and investigate statistical models for individual protein diffusion in cell membranes to describe deviations from pure diffusion due to:
  - protein interactions with particles of various sizes
  - binding interactions
- Three statistical models are developed:
- Diffusion in Noise
- Diffusion-with-Obstacles (DwO)
- Diffusion-with-Obstacles-and-binding sites

Main Ideas (cont'd)

- To estimate the parameters of the Diffusion-with-Obstacles model, we developed and implemented two algorithms:
  - Algorithm based on method of moments
  - Gibbs sampler - Markov chain simulation
- We verified validity of the algorithms using Monte Carlo simulations
- Finally, we applied the algorithms to experimental protein tracks

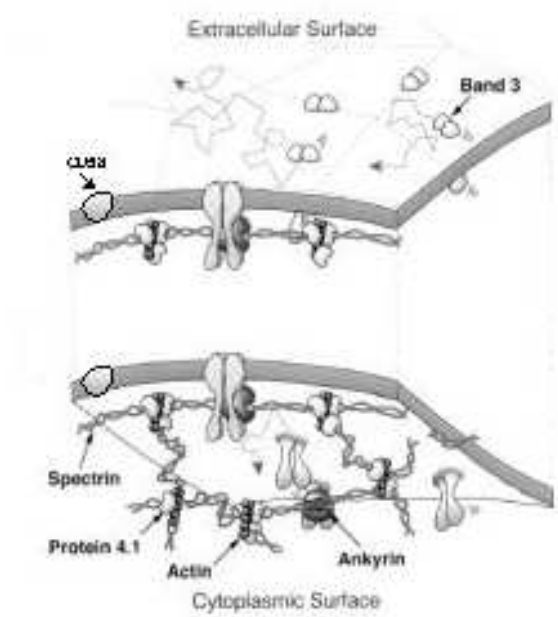
Single-particle Tracking

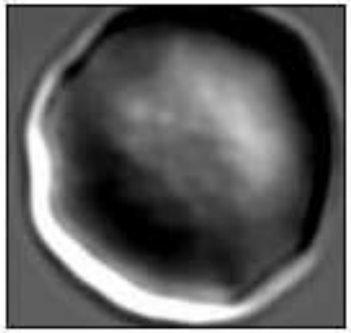
- Single-particle tracking (SPT) technique is used to study the dynamics of the individual proteins in the membranes of intact red blood cells
  - SPT is a powerful approach that allows detailed description of membrane protein motion on nm distance scale and sub-microsecond time scale.
  - High spatial and time resolution of SPT allows to differentiate between types of molecule motion that could be different from pure diffusion
  - Experimental Data: Prof. D. Golan and Dr. R. Mirchev
  - (Harvard Medical School)

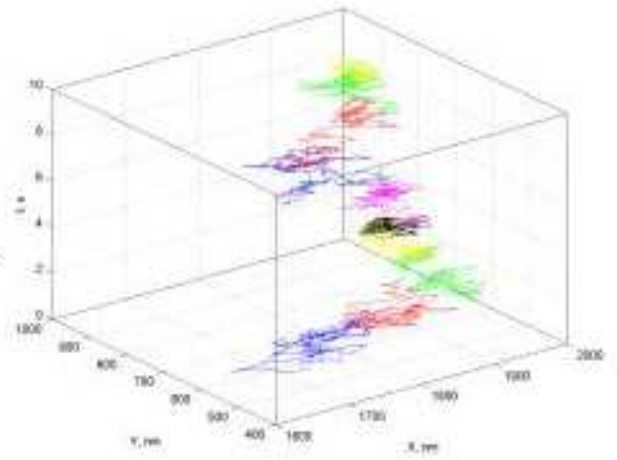
Single-particle Tracking (cont'd)

- A gold bead (5 – 40 nm) is attached to a molecule
- Computer enhanced video microscopy is used to record plane images (128x80 pixels) of the molecule's trajectory
- Molecule's *x* and *y* coordinates in every image are determined using a tracking algorithm
- We use SPT to describe the motion of several types of proteins: band 3 (transmembrane, laterally mobile), and CD58 (non-transmembrane, laterally mobile)

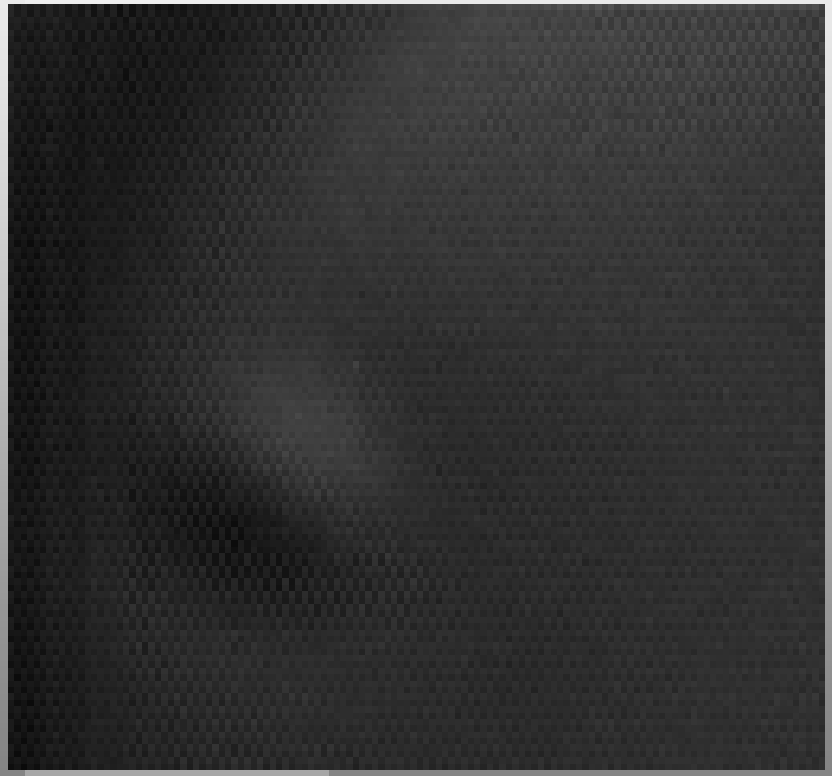
Single-particle Tracking (cont'd)

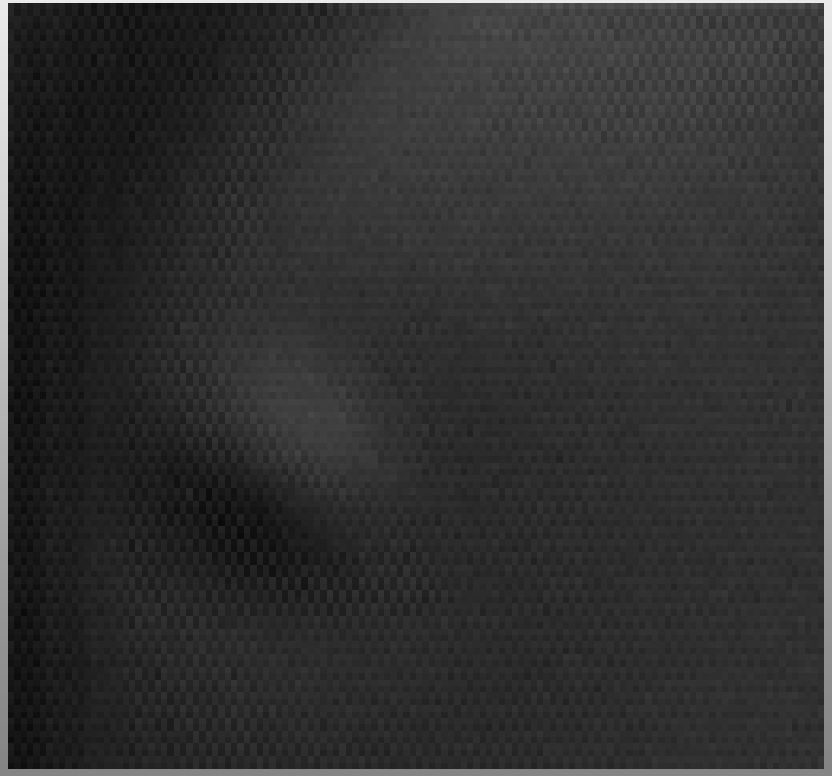






Two successive images





M.-Bayborodin (2000) model

- **Y =(*Y (t)) on lattice t =0,…, kT=N,***
- ***Y (t) = aw(t) + cF (t) + be(t);***
- **where *a > 0; b > 0 and c are real numbers; w(t)* is a standard Wiener process, *e(t) are independent* (mutually and of *w(.)) standard normal errors, F(.) is deterministic.***

Diffusion-with-Obstacles (DwO)

$$Y(t_n) = aW(t_n) + b\sum_{j=1}^{\pi(t_n)} \eta_j + ce(t_n), \quad n = 1,...N, \quad t_n = Tn/N$$

$Y(t_n)$, molecule's position at time t_n

$W(t_n)$, standard Brownian motion

$\pi(t_n)$, Poisson process with intensity μ at time t_n

η_j *i.i.d.* standard normal Random variables

$e(t_n)$, white Gaussian noise

a, b, c, μ parameters to be estimated

All three sources of randomness are independent

Mercer Transform in M-B model

- **E(Y) = F*; F = (F(t)); t = 0; :::;N;***
- ***Cov(Y) = AC + BI;***
- ***I is an identity matrix and***
- ***C = [min(m, n)]***

- **We derive a Mercer representation**
- ***Y (t)=sum Z(k)f(k,t)***

Mercer Decomposition cont'd

- **Z(k) are orthogonal(thus independent due to normality)**
- **f(k,t)= c(N)sin*(a(k,t))***
- ***a(k,t) are equidistant.***
- ***Eigenvalues are explicitly found***

Parameter estimation

- **Measurements *y(t); i = 1; : : : ;N; follow a mixed one-way gaus*sian model. Uniform LAN property of the distributions family proved in Malyutov and Protassov(2000) for One-Way mixed gaussian model under mild conditions on *F(.)***
- ***Fisher Inform. matrix asymptotics evaluated***

Method of Moments: DwO model

$$Y(t_n) = aW(t_n) + b\sum_{j=1}^{\pi(t_n)} \eta_j + ce(t_n)$$

Denote:

$$X(t, \Delta t) = Y(t + \Delta t) - Y(t)$$

Calculate the moments:

$$EX^2(t, \Delta t) = (a^2 + b^2\mu)\Delta t + 2c^2$$

$$EX^4(t, \Delta t) = 3\{(a^2 + b^2\mu)^2 \Delta t^2 + (4c^2(a^2 + b^2\mu) + b^4\mu)\Delta t + 4c^4\}$$

$$EX^6(t, \Delta t) = 15\{(a^2 + b^2\mu)^3 \Delta t^3 + (6c^2(a^2 + b^2\mu)^2 + 3(a^2 + b^2\mu)b^4\mu)\Delta t^2$$
$$+ (12(a^2 + b^2\mu)c^4 + 6b^4c^2\mu + b^6\mu)\Delta t + 8c^6\}$$

Estimation algorithm steps:

1. Calculate the 2nd, 4th, and 6th empirical moments of the process increments $X(t,\Delta t)$ for different values of Δt: $\Delta t = nT/N$. (Usually, $n=1,2,3,4$ is sufficient)

$$m_j(n) = \frac{1}{N-n}\sum_{i=1}^{N-n} X^j\left(t_i, \frac{nT}{N}\right), \qquad j = 2,4,6$$

2. Use linear regression to fit a straight line to the empirical moments

$$m_2(\Delta t), \quad \Delta t = nT/N, \quad n = 1,\ldots,4$$

The slope and the intercept of the regression line are the estimates for $(a^2 + b^2\mu)$ and $2c^2$ respectively.

3. Use the estimates for $(a^2 + b^2\mu)$ and c^2 to calculate

$$M_4(\Delta t) = \frac{m_4(\Delta t)}{3} - (a^2 + b^2\mu)^2 \Delta t^2 - 4(a^2 + b^2\mu)c^2 - 4c^4$$

Fit a straight line to

$$M_4(\Delta t) \approx b^4 \mu \Delta t, \qquad \Delta t = nT / N, \qquad n = 1,...,4$$

The slope is the estimate for $b^4\mu$.

4. Use the estimates for $(a^2 + b^2\mu)$, $b^4\mu$, c^2 to calculate

$$M_6(\Delta t) = \frac{m_6(\Delta t)}{15} - (a^2 + b^2\mu)^3 \Delta t^3 - \{6(a^2 + b^2\mu)^2 c^2 + 3(a^2 + b^2\mu)b^4\mu\}\Delta t^2$$
$$- \{12(a^2 + b^2\mu)c^4 + 6b^4\mu c^2\}\Delta t - 8c^6$$

Fit a straight line to

$$M_6(\Delta t) \approx b^6 \mu \Delta t, \qquad \Delta t = nT / N, \qquad n = 1,...,4$$

The slope is the estimate for $b^6\mu$.

5. We have the estimates for:

$$a^2 + b^2\mu, \quad b^4\mu, \quad b^6\mu, \quad c^2$$

We can now find the estimates for all the model parameters:

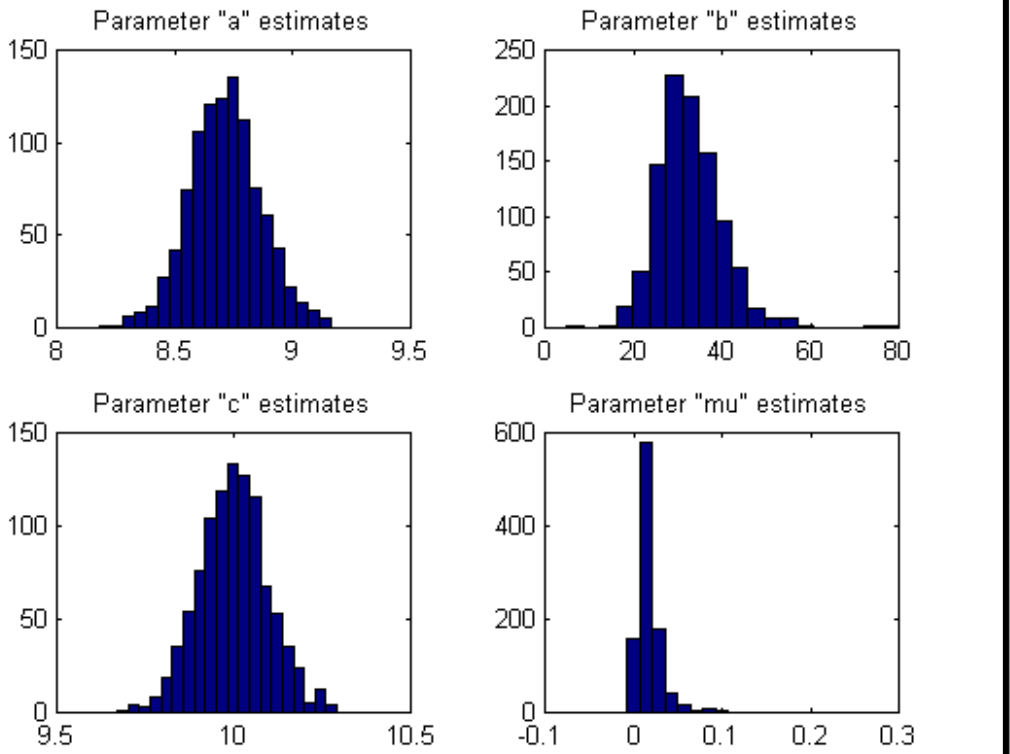
$$\hat{c} = \sqrt{c^2}, \quad \hat{b} = \sqrt{\frac{b^6\mu}{b^4\mu}}, \quad \hat{\mu} = \frac{b^4\mu}{b^4}, \quad \hat{a} = \sqrt{(a^2 + b^2\mu) - b^2\mu}$$

Using the method of moments we can obtain precise parameter estimates

- This method only gives point estimates for the parameters;
- Large data samples or repetition of experiments is necessary to find confidence intervals for the estimates
- When working with SPT data, usually only one trajectory of each molecule is available; the longest feasible trajectory contains around 10,000 points

Estimation Results:

Method applied to 1,000 simulated trajectories



True value	90% CI	95% CI
a=8	[8.5, 9.0]	[8.41; 9.03]
b=35	[22.2, 45.2]	[20.1, 48.5]
c=10	[9.8, 10.2]	[9.8, 10.2]
μ=0.01	[0.005, 0.04]	[0.003, 0.06]

Bayesian Approach

- To get the interval parameter estimates, we use Bayesian approach: we construct a Markov chain that converges to complicated posterior distribution of interest.
- We use a Markov chain Monte Carlo technique
- Specifically, we will use
  - Gibbs sampling algorithm – alternating conditional sampling
  - Data augmentation technique – introduce missing data, we will work with “augmented data” observed data together with missing data

Markov Chains

- Markov chain is a sequence of random variables in which the distribution of each element depends on the value of the previous one.
- Provided that certain regularity conditions are met, the distribution of the elements stabilize to a common distribution known as the stationary distribution.
- In Markov chain Monte Carlo, one constructs a Markov chain whose stationary distribution is a distribution of interest.
- By repeatedly simulating steps of the chain, one is eventually able to simulate from the distribution of interest.

Markov Chain Monte Carlo

- Goal: generate values of random variable Z (parameter values in our case)
- Suppose the density of *Z* is *P(Z)=f(Z)*
- Rather than attempting to draw directly from *f*, we generate a sequence $\{Z^{(1)},\ldots, Z^{(t)},\ldots\}$ where each variate in the sequence depends on the preceding ones, and where the stationary distribution of $Z^{(t)}$ is the target *f.*
- Markov Chain Monte Carlo methods are useful when it is difficult to draw from *f* directly, but drawing each variate in the sequence is straightforward.

Gibbs Sampling

- Suppose r.v. partitioned into subvectors, $Z=(Z_1,..,Z_J)$
- Denote $P(Z)$ the joint distribution of Z , which is also a target distribution to be simulated
- We iteratively draw from conditional distribution of each subvector given all the others. Given the value Z at step t, the value at step $(t+1)$ is obtained from draws:

$$Z_1^{(t+1)} \sim P(Z_1 | Z_2^{(t)}, Z_3^{(t)}, \ldots, Z_J^{(t)})$$

$$Z_2^{(t+1)} \sim P(Z_2 | Z_1^{(t+1)}, Z_3^{(t)}, \ldots, Z_J^{(t)})$$

$$\ldots$$

$$Z_J^{(t+1)} \sim P(Z_J | Z_1^{(t+1)}, Z_2^{(t+1)}, \ldots, Z_{J-1}^{(t+1)})$$

Convergence of Gibbs sampler

- Informally:
  - Target distribution P(Z) must be a genuine probability distribution and the sequence of conditional distributions above must be actual conditional distributions corresponding to this target
  - Sample space of Z must be ‘connected’ in the sense that it must be possible to reach any point in the sample space from any other point by repeated sampling from the conditionals
- Note:
  - Gibbs samplers need not to be drawn from any particular order in each iteration, nor they need to be drawn equally often. As long as each conditional distribution is visited infinitely often, the stationary distribution will be P(Z)

Gibbs sampler estimation algorithm

We will be working with the increment process:

$$X(t_n) = a\Delta W(t_n) + b \sum_{j=\pi(t_{n-1})+1}^{\pi(t_n)} \eta_j + c\Delta e(t_n)$$

$$\sim a\Delta W(t_n) + b \sum_{j=1}^{\Delta\pi(t_n)} \eta_j + c\Delta e(t_n)$$

To simplify the notations we will assume that T=N, and time intervals between the observations is equal to 1 (t_n=n)

We will use the notations:

$$X = (X_1,\ldots, X_N) = (X(t_1),\ldots, X(t_N))$$

...

- In Bayesian data analysis the knowledge of the prior distribution of the model parameters is assumed:

$$a^{-2} \sim Gamma(\alpha_a, \beta_a)$$

$$b^{-2} \sim Gamma(\alpha_b, \beta_b)$$

$$c \sim N(m_c, \sigma_c^{\ 2})$$

$$\mu \sim Gamma(\alpha_\mu, \beta_\mu)$$

- Data augmentation: components of the process X(t): $\Delta\pi$ and $\Delta\varepsilon$ are considered as missing values
- Gibbs sampling:
  - Set some starting values for the unobserved variables
  - Repeatedly visit each unobserved variable each time randomly selecting a new value for the variable from its conditional distribution given the current values of the other variables

1. Pick a starting value for the parameters θ^0, for the vector with the number of jumps $\Delta\pi^0$
   (we denote $\theta=(a,b,c,\mu)$)
2. Simulate $\Delta\varepsilon|\ \theta^0, X, \Delta\pi^0$ to obtain $\Delta\varepsilon^1$
3. Simulate $\Delta\pi|\ \theta^0, X, \Delta\varepsilon^1$ to obtain $\Delta\pi^1$
4. Simulate $\theta\ |X, \Delta\varepsilon^1, \Delta\pi^1$ to obtain $\theta^{\ 1}$
   1. Simulate $\mu\ |X, \Delta\varepsilon^1, \Delta\pi^1, a^0, b^0, c^0$ to obtain μ^1
   2. Simulate a $|X, \Delta\varepsilon^1, \Delta\pi^1, b^0, c^0, \mu^1$ to obtain a^1
   3. Simulate b$|$ X, $\Delta\varepsilon^1, \Delta\pi^1, a^1, c^0, \mu^1$ to obtain b^1
   4. Simulate c$|$ X, $\Delta\varepsilon^1, \Delta\pi^1, a^1, b^1, \mu^1$ to obtain c^1
5. Set $\theta^{\ 1}, \Delta\pi^1$ to be the new starting values and repeat steps 2—4

1. Set the starting value for the parameters *a,b,c*,μ -- use method of moments estimates

   Initialize the vector $\Delta\pi$ by taking i.i.d draws from

   $\Delta\pi \sim Poiss(\mu^0)$

   Starting values for the Gibbs sampler are:

   $$(\theta^0, \Delta\pi^0)=(a^0, b^0, c^0, \mu^0, \Delta\pi^0)$$

2. Simulate $\Delta\varepsilon | \theta^0, X, \Delta\pi^0$ to obtain $\Delta\varepsilon^1$ using a sequence of Forward-Backward calculations:

<u>Forward calculations</u>

$$Find \quad P(\Delta\varepsilon_1 \mid X_1, \Delta\pi, \theta) \rightarrow P(\Delta\varepsilon_2 \mid X_1, X_2, \Delta\pi, \theta) \rightarrow$$

$$\ldots \rightarrow$$

$$P(\Delta\varepsilon_N \mid X_1, \ldots, X_N, \Delta\pi, \theta)$$

$$P(\Delta\varepsilon_{j+1} \mid X, \Delta\pi) = N(\Delta\varepsilon_{j+1} \mid m_{j+1,f}, \sigma^2_{j+1,f})$$

where

$$m_{j+1,f} = \left[\frac{cX_{j+1}}{a^2 + \Delta\pi_{j+1} b^2} - \frac{2m_{j,f}}{\sigma^2_{j,f} + 6} \right] \sigma^2_{j+1,f}$$

$$\sigma^2_{j+1,f} = \left[\frac{c^2}{a^2 + \Delta\pi_{j+1} b^2} + \frac{4}{\sigma^2_{j,f} + 6} \right]^{-1}$$

By forward calculations we obtain density $P(\Delta\varepsilon_N \mid X, \Delta\pi)$; we use it to draw $\Delta\varepsilon_N \mid X, \Delta\pi$

Backward calculations

Draw $\Delta\varepsilon_N \mid X, \Delta\pi, \theta \rightarrow$ *Find* $P(\Delta\varepsilon_{N-1} \mid \Delta\varepsilon_N, X, \Delta\pi, \theta) \rightarrow$

Draw $\Delta\varepsilon_{N-1} \mid \Delta\varepsilon_N, X, \Delta\pi, \theta \rightarrow$

$\ldots \rightarrow$

Find $P(\Delta\varepsilon_1 \mid \Delta\varepsilon_N, \ldots, \Delta\varepsilon_2, X_1, \ldots, X_N, \Delta\pi, \theta) \rightarrow$

Draw $\Delta\varepsilon_1 \mid \Delta\varepsilon_N, \ldots, \Delta\varepsilon_2, X_1, \ldots, X_N, \Delta\pi, \theta$

$$P(\Delta\varepsilon_{j-1} \mid \Delta\varepsilon_N, \ldots, \Delta\varepsilon_j, X, \Delta\pi) = N(\Delta\varepsilon_{j-1} \mid m_{j-1,b}, \sigma^2_{j-1,b})$$

where

$$m_{j-1,b} = \left[\frac{m_{j-1,f}}{\sigma^2_{j-1,f}} - \frac{\Delta\varepsilon_j}{3} \right] \sigma^2_{j-1,b}$$

$$\sigma^2_{j-1,b} = \left[\frac{1}{6} + \frac{1}{\sigma^2_{j-1,f}} \right]^{-1}$$

3. Once all $\Delta\varepsilon$'s are drawn, we can proceed to drawing $\Delta\pi$'s:

$\Delta\pi_i|X, \Delta\varepsilon, \theta$ are independent (for all *i*) and

$$P(\Delta\pi_i = j \mid X_i, \Delta\varepsilon_i, \theta) = \frac{\frac{\mu^j}{j!} e^{-\mu} N(X_i \mid c\Delta\varepsilon_j, a^2 + jb^2)}{\sum_{l=0}^{+\infty} \frac{\mu^l}{l!} e^{-\mu} N(X_i \mid c\Delta\varepsilon_j, a^2 + lb^2)}$$

4. Now we can draw the parameters

<u>Draw μ:</u>

$$\mu \mid \Delta\pi, \Delta\varepsilon, X, a, b, c \sim Gamma(\alpha_\mu + \sum_{i=1}^{N} \Delta\pi_i, \beta_\mu + N)$$

- To draw *a* and *b* we will use Metropolis-Hastings algorithm

Metropolis-Hastings Algorithm

- Given a target distribution $P(\theta|X)$ that can be computed up to a normalizing constant, the M-H algorithm creates a sequence of random points $\theta^1, \theta^2, \ldots$ whose distribution converges to the target distribution

1. Draw a starting point θ^0, for which target distribution $P(\theta^0|X)>0$ from some starting distribution
2. For t=1,2,…

a) Sample a candidate point θ^* from a jumping distribution at time t, $J_t(\theta^*|\ \theta^{t-1})$ (we use the jumping distribution to approximate the target distribution)

b) Calculate the ratio of importance ratios:

$$r = \frac{P(\theta^* \mid X) J_t(\theta^* \mid \theta^{t-1})}{P(\theta^{t-1} \mid X) J_t(\theta^{t-1} \mid \theta^*)}$$

c) Set $\theta^t = \theta^*$ with probability min(r,1), and $\theta^t = \theta^{t-1}$ with probability (1-min(r,1))

- To draw *a* and *b* we use the M-H algorithm with following target distribution:

Draw a:

$$P_{target}(a^{-2} \mid \Delta\varepsilon, \Delta\pi, X) \propto (a^{-2})^{\alpha_a - 1} \exp(-a^{-2}\beta_a)$$

$$P_{jumping}(a^{-2} \mid \Delta\varepsilon, \Delta\pi, X) = Gamma(\alpha_a + \frac{N}{2}, \beta_a + \frac{1}{2}\sum_{i=1}^{N}(X_i - c\Delta\varepsilon_i)^2)$$

Draw b:

$$P_{target}(b^{-2} \mid \Delta\varepsilon, \Delta\pi, X) \propto (b^{-2})^{\alpha_b - 1} \exp(-b^{-2}\beta_b)$$

$$P_{jumping}(b^{-2} \mid \Delta\varepsilon, \Delta\pi, X) = Gamma\left(\alpha_b + \sum_{i=1, \Delta\pi_i > 0}^{N} \frac{1}{2}, \beta_a + \frac{1}{2}\sum_{i=1, \Delta\pi_i > 0}^{N} \frac{(X_i - c\Delta\varepsilon_i)^2}{\Delta\pi_i}\right)$$

Draw c:

$$c \mid X, \Delta\varepsilon, \Delta\pi, a, b, \mu \sim N(m_{c_{new}}, \sigma^2_{c_{new}})$$

where

$$m_{c_{new}} = \left[\frac{m_c}{\sigma_c^2} + \sum_{i=1}^{N} \frac{\Delta\varepsilon X_i}{a^2 + \Delta\pi_i b^2} \right] \sigma^2_{c_{new}}$$

$$\sigma^2_{c_{new}} = \left[\frac{1}{\sigma_c^2} + \frac{\Delta\varepsilon_i^2}{a^2 + \Delta\pi_i b^2} \right]^{-1}$$

- The random draws described above are repeated at every iteration of the algorithm
- We constructed a Markov chain with the stationary distribution that is the posterior of interest

- We constructed an irreducible aperiodic Markov chain with all states non-null persistent, therefore it has a unique stationary distribution that coincides with the limiting distribution. The limiting distribution of the Markov chain is the posterior of interest.

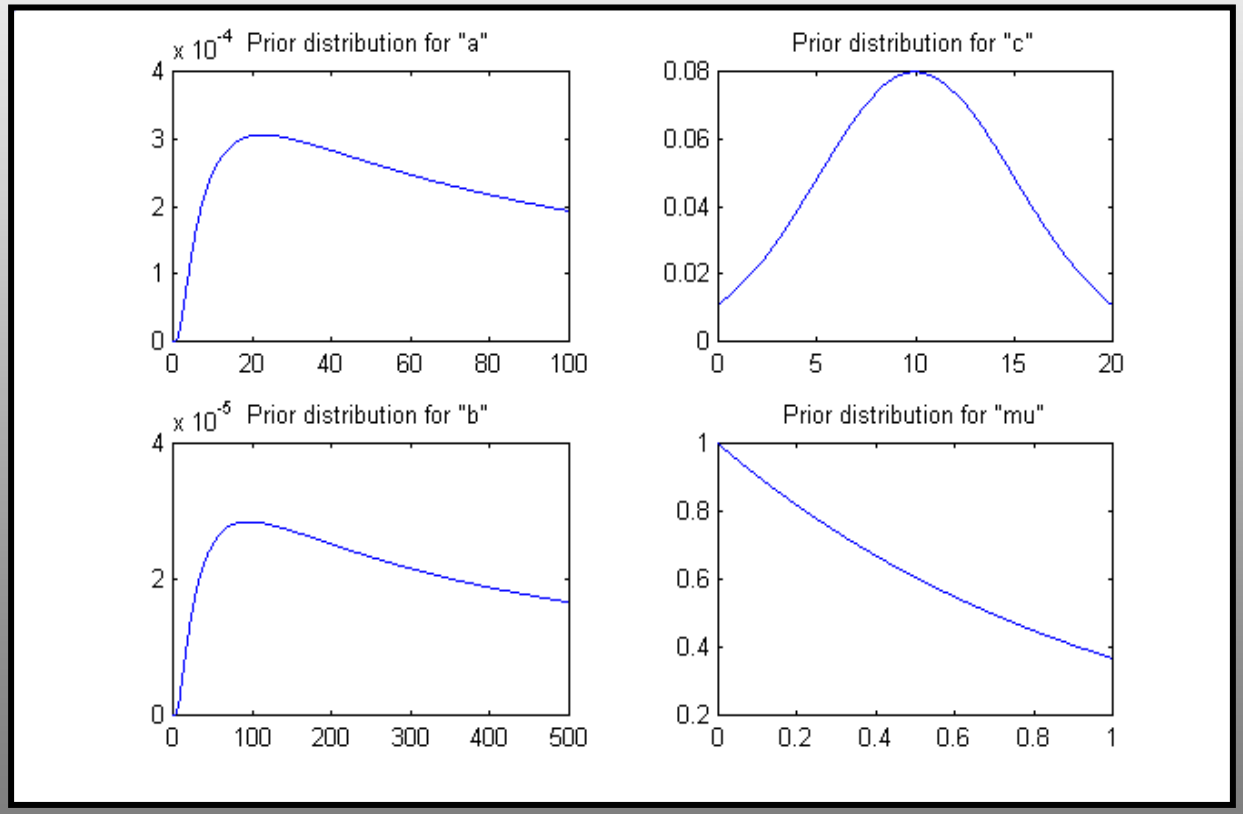
Parameter Estimation: choice of priors

$a^{-2} \sim Gamma(0.1,10)$

$b^{-2} \sim Gamma(0.04,20)$

$c \sim N(10,5) \qquad c > 0$

$\mu \sim Gamma(1,1)$



Estimation Results:

method applied to simulated trajectories containing 10,000 points

	a	b	c	μ
True values	8	35	10	0.01
Meth.of Mom Est.	8.2	37.3	9.9	0.009
Gibbs S. 95%CI	[8.6, 9.9]	[33.1, 85.0]	[9.4, 10.1]	[0.001, 0.008]
True values	8	37	10	0.15
Meth.of Mom Est.	8.6	40.1	10.5	0.126
Gibbs S. 95%CI	[6.9, 9.0]	[41.0, 48.0]	[10.0, 11.2]	[0.096, 0.136]
True values	12	50	10	0.01
Meth.of Mom Est.	12.3	50.9	9.9	0.009
Gibbs S. 95%CI	[11.5, 12.7]	[46.8, 85.0]	[9.9, 10.8]	[0.003, 0.011]
True values	10	100	10	0.01
Meth.of Mom Est.	10.4	104.5	9.7	0.008
Gibbs S. 95%CI	[8.4, 9.9]	[80.0, 115.4]	[10.4, 11.4]	[0.006, 0.012]

Estimation Results:

method applied to experimental CD58 trajectory containing 10,000 points (2-dimensional)

X-coordinate

	a	b	c	μ
Meth.of Mom Est.	7.7	22.7	9.67	0.018
Gibbs S. 95%CI	[6.3, 8.3]	[22.3, 51.1]	[9.6, 10.8]	[0.001, 0.022]

Y-coordinate

	a	b	c	μ
Meth.of Mom Est.	7.7	37.0	10.3	0.014
Gibbs S. 95%CI	[6.5, 8.3]	[25.0, 36.5]	[10.2, 11.0]	[0.013, 0.044]

Goodness-of-Fit

- We generated 100,000 i.i.d. random variables

$$X_n \sim a\Delta W_n + b\sum_{j=1}^{\pi_n}\eta_j + c\Delta e_n$$

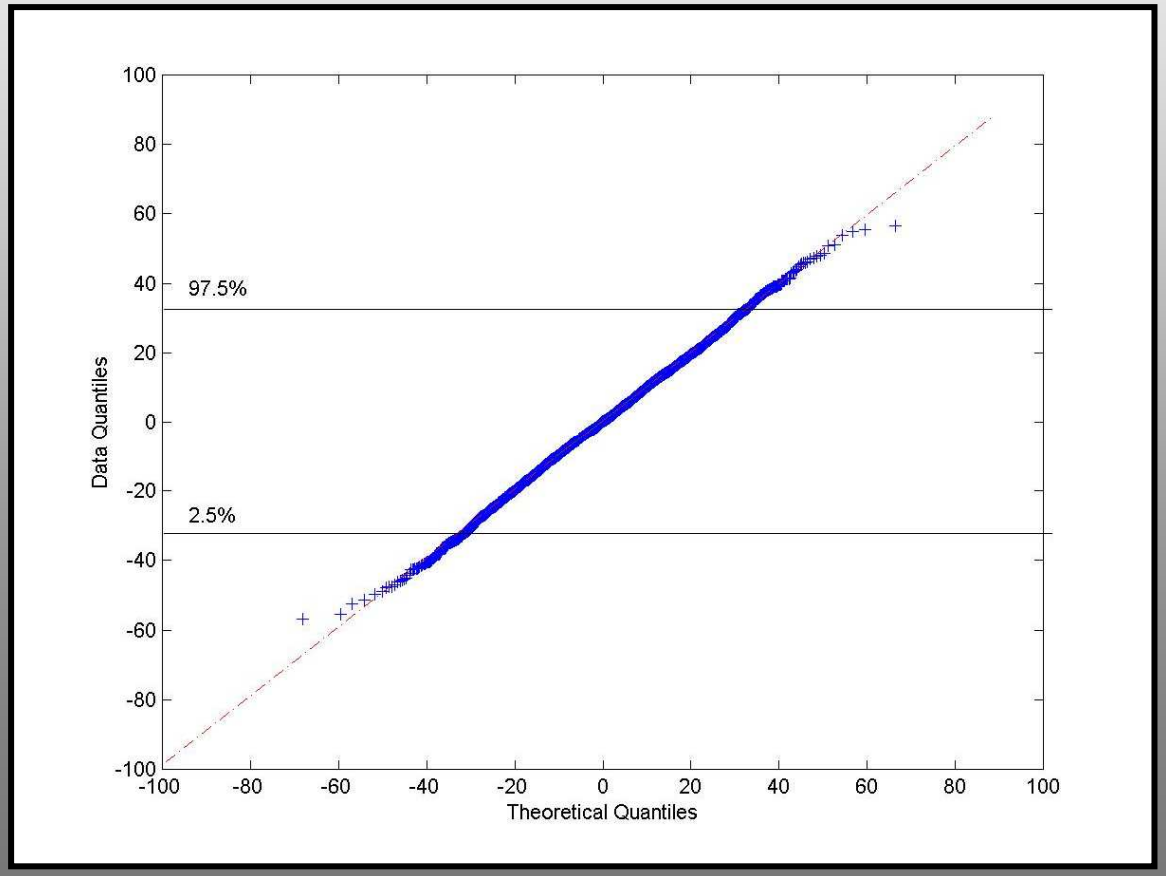
where

$$\Delta W_n \sim N(0,1), \quad \eta_j \sim N(0,1), \quad \Delta\pi_n \sim Poiss(\mu), \quad \Delta\varepsilon_n \sim N(0,2)$$

(the constants a,b,c,μ are method of moments parameter estimates)

- We used the resulting data sample to calculate the empirical distribution function.
- To obtain an i.i.d. sample from the experimental data we used every other increment of the experimental process

- We conducted Chi-square goodness-of-fit test, which confirmed the validity of the model application to the experimental data sample
- The quantile-quantile plot of the theoretical vs. experimental data presented below gives another confirmation of the model fit



Diffusion-with-Obstacles-and-Binding Sites (final) Model

- To describe binding interactions we introduce time transformation $\tau(t)$:
  - $\tau(t_0)=t_0$
  - $\tau(t_{n+1})= \tau(t_n)$, if molecule is trapped in a binding site at time t_n
  - $\tau(t_{n+1})= \tau(t_n)+T/N$, if molecule diffuses freely at time t_n
- Probability to escape a binding site is α;
- Probability to get into a binding site is β.

Diffusion-with-Obstacles-and-Binding-Sites

$$Y(t_n) = aW(\tau(t_n)) + b\sum_{j=1}^{\pi(\tau(t_n))} \eta_j + ce(t_n)$$

- Using Markov chain theory we derive the distribution of the time transformation function:

$$\tau(t) \xrightarrow{D} X \sim N\left(\frac{n\alpha}{\alpha+\beta}\frac{T}{N}, \sqrt{\frac{n\alpha\beta(2-\alpha-\beta)}{(\alpha+\beta)^3}}\frac{T}{N}\right), \quad as \quad n \to \infty.$$

- With the above approximation, we can find asymptotic characteristic function of $Y(t_n)$:

- Asymptotic Characteristic function:

$$E\{\exp(iuY(t_n))\} \to \exp\left(\frac{n\alpha}{\alpha+\beta}\frac{T}{N}\left\{ -\frac{a^2u^2}{2} + \mu\left(e^{-b^2u^2} - 1 \right) \right\} - \right.$$

$$\left. -\frac{1}{2}\frac{n\alpha\beta(2-\alpha-\beta)}{(\alpha+\beta)^3}\frac{T^2}{N^2}\left\{ \frac{a^4u^4}{4} + \mu^2\left(e^{-b^2u^2} - 1 \right)^2 \right\} - \frac{c^2u^2}{2} \right)$$

- Possible parameter estimation techniques:
  - Fit theoretical characteristic function by an empirical one using non-linear regression
  - Gibbs sampler
- The applications of the parameter estimation algorithms for this model are subject of our future research

Conclusions

- Two novel mathematical models for protein diffusion were developed
- We used two probabilistic approaches to estimate the model parameters:
  - Traditional frequentist approach
  - Bayesian approach
- Two robust algorithms for model parameter estimation were developed and implemented:
  - Algorithm based on Method of Moments
  - Gibbs sampler