1 Theory part

1.1 Introduction

In this it will be considered such the translocation of polymer. This topic is an important issue in various biological and physical processes.
The chain transfer is described by the Fokker-Plank (FP) formalism of random walk across the free energy landscape of the translocation pass represented as a sum of the free energies of *cis* and *trans* parts of the chain tethered to the pore opening.
The landscape of free energy may have different forms and the one with the minima corresponding to stable and metastable states and the maxima corresponding to the energy barriers that should be overcome.

1.2 Modeling the motion

As it was already said, the translocation may be described using FP equations. A polymer chain composed of N Kuhn segments and is modeled as a random walk trajectory of N steps. In such situation the number of Kuhn segments s ($s = 1, ..., N$) represents a discrete coordinates along the trajectory. When N is large enough, the chain may be approximated by a continuous trajectory with the coordinate s varying continuously from 0 to N.

1.3 Free energy for different chains

On the picture below we can observe the Free energy landscape for ideal Gaussian chain: In real chain, however, the excluded volume leads to higher free energies compared to Gaussian chains. This difference progresses with the increase of the chain length.
In the next picture the free energy of a tethered chain confined to a spherical pore is presented.

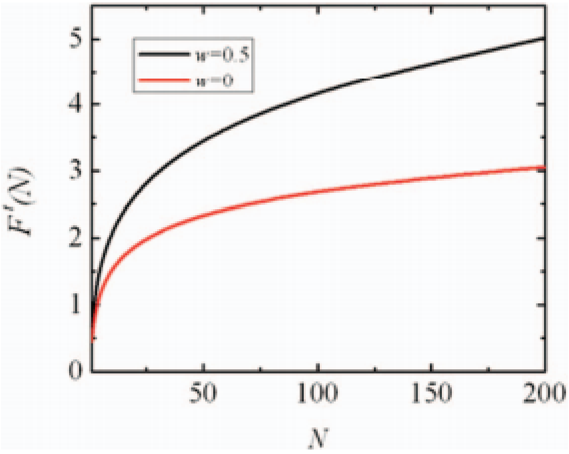


Рис. 1: Chain length dependence of the free energy of Gaussian and real chains tethered at the plain surface without adsorption.

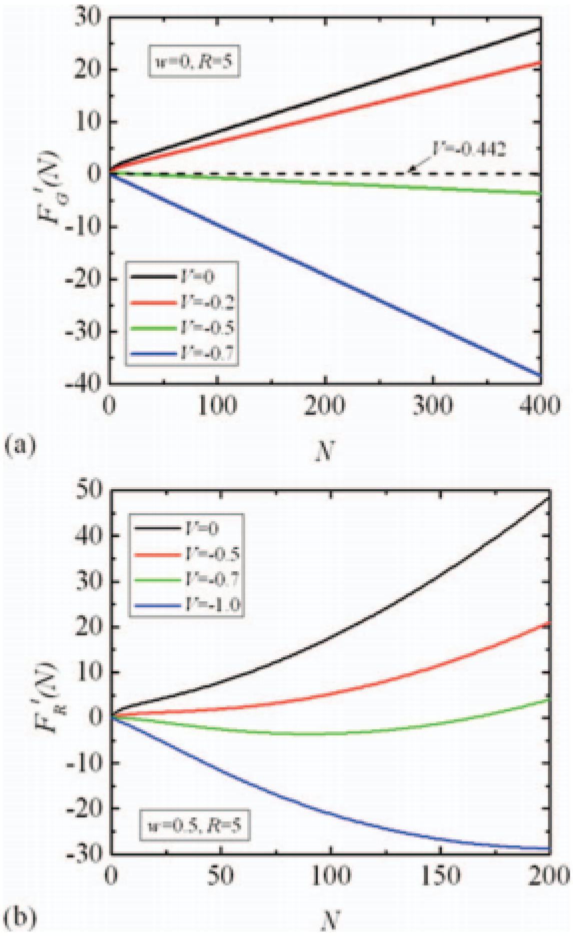


Рис. 2: a) FEL for ideal chain b) for real chain for different adsorption potential

On the picture above different regimes depending on adsorption potential can be observed. Let's discuss them in details.

1.3.1 Different characteristic regimes

There are 3 different modes can be described:

1 Strong adsorption conditions:
Resulting enthalpy due to adsorption interaction exceeds the loss of entropy due to the constraints imposed by the limiting pore geometry on the chain conformations. As the result the chains are preferentially adsorbed and the free energy of adsorbed chains decreases with the chain length.

2 The entropy penalty is prohibitive and chains are effectively repelled from pores. The free energy of adsorbed chains increases with the chain length.

3 So as weak and strong regimes exists, the critical regime should also exists. And experiments give us such results. Skvortsov and Gorbunov demonstrated this condition during adsorption in pores using the ideal chain model without the effect of excluded volume. But following modeling (using Monte Carlo simulations) studies showed that the existence of critical for chains adsorbed in pores is limited to the ideal chains and FE always depends on the chain length regardless of the adsorption potential.

The mechanism that can explain the experimental results performing the critical conditions for chain adsorption on porous substrates is the presence of partially confined chain conformations.

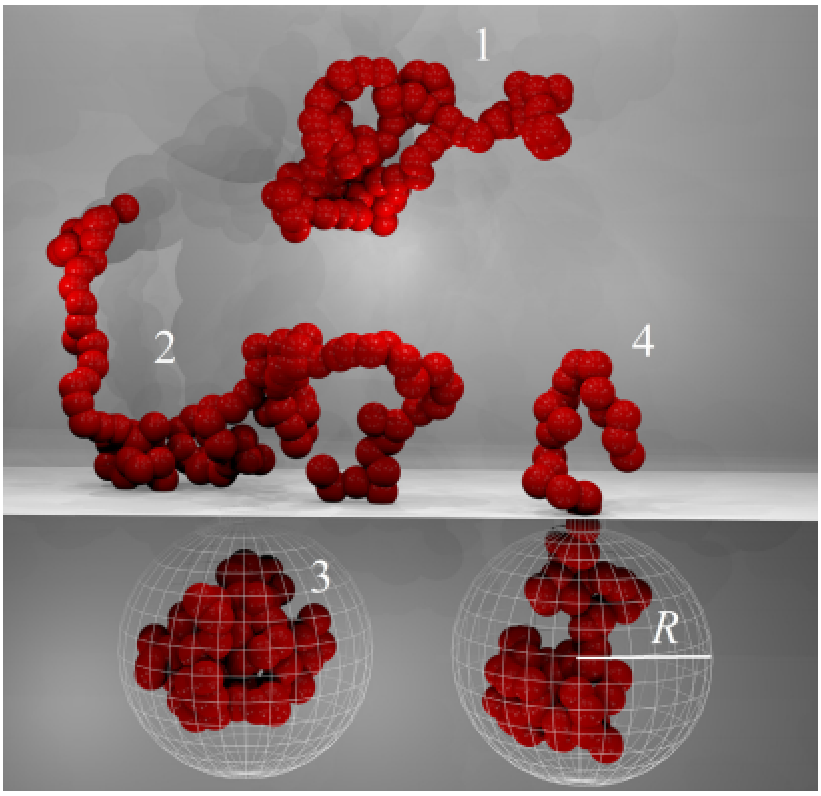


Рис. 3: 1 –unconfined chain, 2 – chain adsorbed at the external surface, 3 – chain completely confined within the pore, 4 – partially confined flower conformation.

So as we have on figure [2] - for ideal chain we have the critical value for adsorbtion potential about -0.412. But for real chain model - not.

1.4 Free energy landscape of translocating chain

And now let's move to the free energy landscape of represents the dependence of translocating chain. This free energy depend on the degree of translocation (translocation coordinate). The free energy of the translocating chain is characterised as a sum of the free energies of *cis* and *trans* subchains:

$$F_{total}(N,n) = F_{trans}(n) + F_{cis}(N-n)$$

The landscapes for real translocating chains are generally convex. The convexity is explained by the increase of the excluded volume interactions in trans compartment as the translocation progresses.

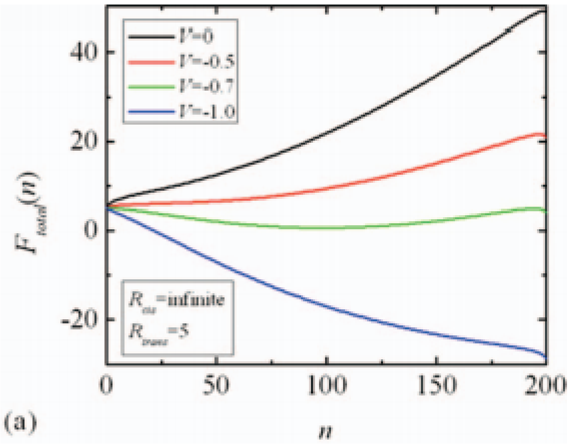


Рис. 4: Free energy landscape for real translocating chain

Also we can see different regimes here:

1 Weak adsorption ($-0.5 < V < 0$) - free energy of cis subchain is much smaller than of trans subchain. Free energy monotonically increases so a high-energy barrier is setting up. Such configuration is unfavorable because translocation implies an increase of the chain free energy.

2 Strong adsorption ($V < -0.7$) - the free energy gradually decreases and translocation is effectively lightened.

3 Intermediate regime ($-0.7 < V < -0.5$) - the free energy has a minimum at a certain degree of translocation. Such a minimum may cause very long translocation time as we have a long-living conformation of the chain composed both of cis and trans subchains.

1.5 Translocation dynamycs

The translocation is considered with the help of FP formalism modeling random walk process over the free energy landscape.
Consider $W(n, t; n_0, 0)$ - to be the probability of translocation of $n - n_0$ segments within time t.
After some mathematics calculations:

$$\frac{\partial}{\partial \tau} W(n, \tau, n_0, 0) = \frac{\partial}{\partial n} \left[\frac{\partial F_n}{\partial n} W(n, \tau, n_0, 0) + \frac{\partial}{\partial n} W_n(n, \tau, n_0, 0) \right]$$

With the initial and boundary conditions:

$$W(n{,}t{,}n_0{,}0) = 0, n = 0, n = N$$

$$W(n{,}t{,}n_0{,}0) = \delta(n - n_0), t = 0$$

Also we need the success rate of translocations.

2 Goal of the project

The main idea of this project is to be able to reestablish the profile of free energy landscape if we have 2 profiles of probability and the success rate of translocation.
The algorithm of the reaching the goal is following:

1 Collecting dataset using fortran program and the number of initial free energy landscapes.

2 Parameterize the dataset of free energy landscapes by several parameters (in my case - with 7 order polynomial coefficients).

3 As we have the profiles of probability and the success rate of translocation we can find the mean squared difference between our really profiles and profiles in our dataset. On this data we can train the regression model.

4 Choosing the initial conditions we can predict the results and using a minimization to find the minimum of our prediction result (as we want to reach 0).

5 After using minimization we find some parameters, characterizing the free energy landscape and using fortran get new profiles of probability and the success rate of translocation.

6 Adding the not optimized initial conditions in our dataset, optimized initial conditions become new initial condition and we repeat the previous steps.

In this project 2 different models with 2 different approach was built:

1 Bayesian approach - Gaussian Process Regression.

2 Probabilistic approach - Gradient Boosting Regressor.

3 Bayesian methods

Distinctive features of Bayesian methods:
If we have Θ - some parameters and x - the data:

1 Θ - random and x - fixed

2 for any sample size

3 Use Bayesian theorem:

$$P(\Theta|x_{tr}, y_{tr}) = \frac{P(y_{tr}|x_{tr}, \Theta)P(\Theta)}{P(y_{tr}|x_{tr})}$$

Example of Bayesian approach

1. **Online learning**
If we have data that comes in with some small portions. We can use it to update your parameters and then use the new posterior as a prior to the next experiment.

$$P_k(\Theta) = P(\Theta|x_k) = \frac{P(x_k|\Theta)P_{k-1}(\Theta)}{P(x_k)}$$

where P_k - new prior, $P(\Theta|x_k)$ - posterior, $P(x_k|\Theta)$ - likelihood and P_{k-1} - prior
2. **Model in linear regression**
1) We have the weights, the data, and the target. We're actually not interested in modeling the data, so we can write down the joint probability of the weights and the target, given the data:

$$P(w,y|x) = p(y|x,w) * P(w)$$

1) Now we need to define these distributions of the probability of target given the weights of the data, and the probability of the weights. Let's assume them to be normal:

$$P(y|w,x) = N(y|w^T x, \sigma^2 I)$$

$$P(w) = N(w|0,\gamma^2 I)$$

2) Train: we want to

$$P(w|y,x) = \frac{P(y,w|x)}{P(y|x)} - > max_w$$

$$P(y,w|x) - > max$$

After some mathematical calculations we get

$$||y - w^T x||^2 + \lambda||w||^2 - > min$$

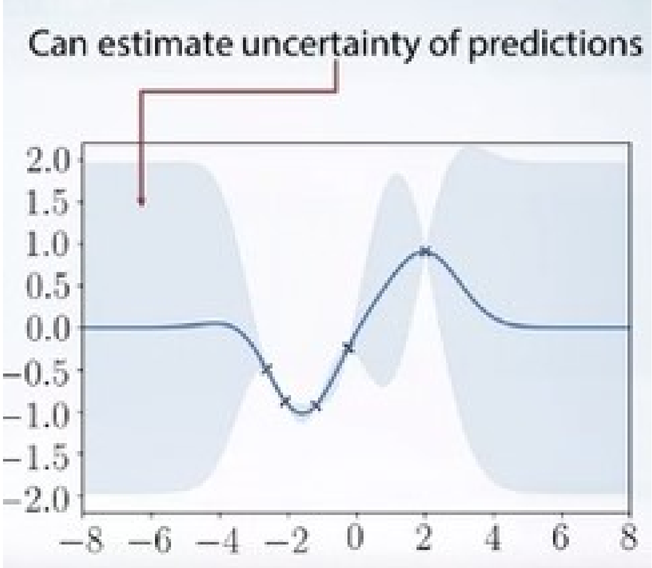
The same, that we usually get in regression tasks.

Gaussian processes and its connection to ML

Definition Gaussian Process:
$\forall n \in N \forall x_1, x_2...x_n \in R^d$:

$$(f(x_1), ...f_n(x))^T \sim N(m(x), K(x,x))$$

where $m(x) = (m(x_1), ...m(x_n))$, $m(x_i) = Ef(x_i)$ and $K(x_1, x_2) = cov(f(x_1), f(x_2))$
Gaussian processes method - non parametric method. It is able to estimate uncertainty of the predictions.



Gaussian Process in ML

Consider us to have some points x_1, x_2 ... x_n. And known values for them: $f(x_1)$, $f(x_2)$,... $f(x_n)$. For Gaussian process we will try to predict the full posterior over the f(x) given all our data points. So we would like to estimate the probability of f(x) with prediction at new point given all previous points. This will allow us to compute the mean for example and also to compute the confidence intervals at each point. This will allow us to estimate uncertainty of our predictions.

$$p(f(x)|f(x_1), f(x_2)...f(x_n)) = \frac{p(f(x), f(x_1), ...f(x_n))}{p(f(x_1),...,f(x_n))} = \frac{N(...|0, \widetilde{C})}{N(f(x_1), ...f(x_n)|0,C)} = N(f(x)|\mu, \sigma^2)$$

where $\mu = k^T c^{-1} f$ and $\sigma^2 = K(0) - k^T C^{-1} K$

$$\text{C} = \begin{pmatrix} K(0) & K(x_1 - x_2) & ... & K(x_1 - x_n) \\ ... & ... & ... & ... \\ K(x_n - x_1) & K(x_n - x_2) & ... & K(0) \end{pmatrix}$$

and

$$\widetilde{C} = \begin{pmatrix} K(0) & K^T \\ K & C \end{pmatrix}$$

Gaussian process regression (GPR) is nonparametric, GPR calculates the probability distribution over all admissible functions that fit the data. However, similar to the above, we specify a prior (on the function space), calculate the posterior using the training data, and compute the predictive posterior distribution on our points of interest.

1) In GPR, we first assume a Gaussian process prior, which can be specified using a mean function, m(x), and covariance function, k(x, x'): 2) The form of the mean function and covariance kernel

$$f(x) \sim GP(m(x), k(x, x'))$$

function in the GP prior is chosen and tuned during model selection.

Examples of the kernels:

- radial basis function
- White Kernel

And so on.

Hyperparameters of kernel we can modify using (for example) Bayesian Optimiser.

Example of the result

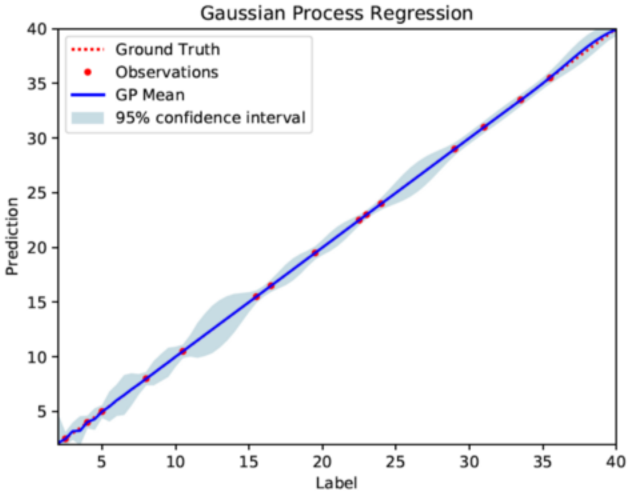


Рис. 5: Caption

Fitting Gaussian Processes in Python

Though it's entirely possible to extend the code above to introduce data and fit a Gaussian process by hand, there are a number of libraries available for specifying and fitting GP models in a more automated way. I will demonstrate and compare three packages that include classes and functions specifically tailored for GP modeling:

1 scikit-learn

2 GPflow

3 PyMC3

In particular, each of these packages includes a set of covariance functions that can be flexibly combined to adequately describe the patterns of non-linearity in the data, along with methods for fitting the parameters of the GP.

4 Useful links

1 B Approach:

1.1. https://habr.com/ru/post/276355/

1.2. Bayesian Methods for Machine Learning Coursera Национальный исследовательский университет "Высшая школа экономики"week 1 and 6

1.3. Обучение на размеченных данных Coursera week 5

2 GPR

2.1. https://towardsdatascience.com/quick-start-to-gaussian-process-regression-36d838810319

2.2. https://blog.dominodatalab.com/fitting-gaussian-process-models-python/

3 BP:

3.1. https://www.machinelearningmastery.ru/an-introductory-example-of-bayesian-optimization-in-python-with-hyperopt-aae40fff4ff0/

3.2. http://sheffieldml.github.io/GPyOpt/

4 Adsorption-driven translocation of polymer chain into nanopores J. Chem. Phys. 136, 214901 (2012); https://doi.org/10.1063/1.4720505

5 Mechanisms of chain adsorption on porous substrates and critical conditions of polymer chromatography Richard T. Cimino, Christopher J. Rasmussen, Yefim Brun, Alexander V. Neimark https://doi.org/10.1016/j.jcis.2016.07.019