

# Stat440 Final Project

## Dynamical Analysis of Molecular Interactions

Karmei Koo (kmkoo) Wanxin Li (w328li) Jayden Luo (j57luo) Yi Xiang (y25xiang)

### Abstract

Fluorescence resonance energy transfer (FRET) has been used to study biological structures and monitor the activities of molecules (Lemke, Deniz, & Groarke, 2016). However, some conformational changes are difficult to detect using ensemble FRET. In order to understand the interaction at single molecule level, single-molecule fluorescence resonance energy transfer (smFRET) is developed to probe the detailed kinetics of structure changes within a single molecule (Ha, 2001). By using this revolutionized technique, scientists unlock the opportunities to research more about protein folding-unfolding, protein conformation dynamics, ion channel dynamics, receptor-ligand interactions, nucleic acid structure and conformation, vesicle fusion, and force induced conformational changes (Sasmal, Pulido, Kasal, & Huang, 2016). All the new discoveries would reveal the mechanism of diseases at the cellular level (Ideker & Sharan, 1970). Current smFRET research employs Brownian Motion (BM) in modelling free diffusion state (Wallace & Atzberger, 2017) and Ornstein-Uhlenbeck (OU) process to model the bonding state where there is a consistent bond with stochastic perturbations between donor and acceptor (Yang, Witkoskie, & Cao, 2002). One of the remaining challenges of smFRET modelling is whether it is possible to differentiate between free diffusion and bound donor-acceptor pairs. Another challenge is whether we can detect binding and unbinding events. To do so, a simplified experiment setup as follows. The number of photons at time follows a distribution:  $Y_n \stackrel{ind}{\sim} \text{Poisson}(\exp(\beta_0 + \beta_1 X_t))$ , where  $X_t$  is the true donor-acceptor distance modelled by BM and OU process.

The challenges of this problem are the following:

- How to set up experiments so that OU parameters can be accurately estimated
- Investigate on whether and if applicable, when it is possible to identify the simulation model by combining parameter inferences and model selection criteria.

By a simulation study, our contributions include:

- Implemented parameter inferences for BM and OU models and tested likelihoods and estimates by unit tests
- Found a set of  $\beta'$ s to accurately estimate OU parameters and explained how  $\beta'$ s affect OU parameter estimation
- Found a threshold to distinguish BM simulated data from OU model, vice versa, and explained why the thresholds are reasonable mathematically.
- Compared the results generated by different optimization techniques
- Enabled random start in optimization algorithms

# Introduction

## Brownian Motion

BM is widely used to model the donor-acceptor distance during free diffusion in the molecular environment. The random flow of molecular motion is subordinated by collision of moving molecules. In our dynamical analysis,  $X_t$  is denoted as the donor-acceptor distance at time  $t$ . It is a Gaussian Markov process with the following transition density:

$$X_{t+\Delta t}|X_t = N(X_t, \sigma^2 \Delta t)$$

Every subsequent variation in distance by an increment of  $\Delta t$  has a mean-reverting nature. That is, the donor-acceptor distance in the next instance revolves around the mean of the process in various direction. In BM model, the mean is set as the current-time donor-acceptor distance. With the process involving Markov element, the sequence of possible molecular motions is only current-state dependent, and it is often referred to as memoryless. The collection of the independent processes at each instance  $\Delta t$ , are recognised as jointly Gaussian (normal).

The variation is determined by the diffusion rate  $\sigma$ , multiplied by  $\Delta t$ . The diffusion rate is set constant in our study, which implies the thermal temperature and pressure are kept unchanged and no external force is applied into the system.

However, the many-body interactions that yield the random pattern cannot be solved by BM model that accounts every involved molecule. Therefore, BM itself is not capable to model the full motion of molecules especially during the photon exchange between donor and acceptor. This is ultimately a downside of BM model in our dynamical analysis.

## Ornstein-Uhlenbeck Process

Ornstein-Uhlenbeck process is further developed from BM by L. S. Ornstein and G. E. Uhlenbeck.[1] The OU process is a stochastic Gaussian process with continuous paths. The OU process is defined as the following stochastic differential equation:

$$dX_t = \gamma(\mu - X_t)dt + \sigma dW_t \tag{1}$$

Where  $W_t$  is a standard BM on  $t \in (0, \infty)$ ,  $\gamma > 0$ ,  $\sigma > 0$   $X_t$  is a stationary Gaussian Markov process with transition density:

$$X_{t+\Delta t}|X_t \sim N(\mu + \omega_{\Delta t}(X_t - \mu), \tau^2(1 - \omega_{\Delta t}^2)) \tag{2}$$

where  $\omega_{\Delta t} = \exp(-\gamma \Delta t)$  and  $\tau^2 = \sigma^2/(2\gamma)$ . Compared to BM model, the OU process includes more factors from the environment that can impact the change of the distance. The OU process has four components to describe the molecule interactions which are  $\gamma$ ,  $\mu$ ,  $\sigma$ , and  $dW_t$ . The interpretation of  $\gamma$  is the rate of mean reversion. The OU process shares mean reversion nature with BM. This property allows that the distance will eventually revert to the long-run average. The rate indicates how strong the distance will react toward the attractor.  $\mu$  is the asymptotic mean of "bond" length.  $\sigma$  is the parameter showing the volatility of noise. (<https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4898467/>). The OU process represents how molecule move towards the attractors  $\mu$  with BM.

The OU process considers the current state and the speed of molecule interaction, which is preferred when modeling the many-body interactions. Since with different type of donor-acceptor interactions, the molecule will react differently. One of the downsides of OU process is that a very small amount of variation such as measurement error can profoundly affect the performance of the model. Also the distance is always positive. However, the donor acceptor in OU allows negative distance.

## Methodology

### NA handling

For research direction 2, whenever NA is simulated in  $Y_t$  due to large  $\lambda$  in *rpois*, we discard it together with corresponding  $X_t$ .

### Laplace implementation

In (todo:equation), we aim to have the  $\hat{\theta}$  that maximize  $l_{Lap}(\theta|Y)$ . In addition, in (todo:equation), we need the value of  $X_\theta$  for  $\hat{\theta}$ , this  $X_\theta$  and  $\theta$  have to be optimized simultaneously; we use Newton-Raphson method to in Laplace approximation to achieve this.

Also, we compare the results of our self-implemented Laplace approximation with TMB's built-in Laplace approximation. Despite of similar accuracy as demonstrated in Appendix, the built-in Laplace implementation is much faster than ours. Thus, we decide to work with the built-in implementation.

### Multi-start Optimization

We initially implemented our likelihood function under the OU model in TMB using  $(\mu, \sigma, \gamma)$  as parameters. However, from our experiments we found that using the R method `optim()` does not satisfiable inference result regardless of the choice of method e.g. BFGS and Nelder-Mead. In addition, we noticed that the passing different initial parameter values to `optim()` can caused very different inference results. Therefore, we implemented another likelihood implementation using  $(\omega, \sigma, \tau)$  as parameters so that we can perform multi-start on top of `optim()`. Below is a RMSE comparison between single start and multi-start.

```
# TODO: move to a different csv
multistart_result <- read.csv("sim1_result.csv")[1:2,]
multistart_result[1, c("beta0", "beta1", "n_obs", "n_dataset", "theta.omega",
                      "theta.mu", "theta.tau", "theta.gamma")]

##   beta0 beta1 n_obs n_dataset theta.omega theta.mu theta.tau theta.gamma
## 1    15     1   299      100   0.3678794      10       1           1

multistart_result[, c("num_multistart", "rmse.omega", "rmse.mu", "rmse.tau",
                     "rmse.gamma", "rmse.t", "rmse.sigma")]

##   num_multistart rmse.omega rmse.mu rmse.tau rmse.gamma rmse.t rmse.sigma
## 1              1      1.50  1.2000   11.000      0.88 330.00     0.190
## 2              10      0.15  0.0095    0.049      0.15  0.15     0.065
```

### Simulation for research direction 1

The goal here is to investigate on how the value of  $(\beta_0, \beta_1)$  could affect the accuracy of inference under the OU model. We setup a experiment with true parameters being  $\mu = 10$ ,  $\tau = 1$  and  $\gamma \in \{0.1, 1, 10\}$ . In order to evaluate the accuracy of inference, we simulate 100 dataset for each pair of  $(\beta_0, \beta_1)$  and calculated the root mean-squared error (RMSE) of  $t_{dec}$ ,  $\mu$  and  $\tau$  as measure of accuracy. We first find a initial pair of  $(\beta_0, \beta_1)$  such that the RMSE reasonably good. We looked into the simulated dataset when  $\gamma = 1$  (TODO: maybe a graph here). We observed that the values of  $X_n$  are around 10 and hence set a initial  $beta_1 = 1$  and  $beta_0 > 10$  such that we have  $\beta_0 - \beta_1 X_n > 0$  and  $Y_n \sim \text{Poisson}(\exp(\beta_0 - \beta_1 X_n))$  are more likely to have a large set of values. For  $\gamma = 1$ , we decided to simulate 299 observations per dataset. For  $\gamma = 0.1$ , since we were not able to get good RMSE with 299 observations per dataset, we increased to 999 per dataset. We then perform inference simulations with one of the the  $\beta(\beta_0 \text{ or } \beta_1)$  fixed and the other gradually decreasing and observe how RMSE changes.

## Simulation for research direction 2

To start, we simulate from OU process. We perform the experiment 4 times for a fixed set of  $\beta_0, \beta_1, \gamma, \mu$ . 200  $X_t$ 's and  $Y_t$ 's without NA are generated in each experiment. We observe that for small values of  $\sigma_{OU}$ , for example 0.00000001, 0.001, 0.01, 0.1, 1, AIC picks BM model after parameter inferences are performed. The threshold for AIC to pick the correct model for  $\sigma_{OU}$  lies in somewhere between 1 and  $\sqrt{2}$ .

Similarly, simulate from BM process. We perform the experiment 4 times for a fixed set of  $\beta_0, \beta_1$ . 200  $X_t$ 's and  $Y_t$ 's without NA are generated in each experiment. We observe that for small values of  $\sigma_{BM}$ , for example 0.00000001, AIC picks OU model after parameter inferences are performed. The threshold for AIC to pick the correct model for  $\sigma_{BM}$  lies in somewhere between 0.00000001 and 0.001.

The above observations match with our intuition on this research direction. As  $\sigma_{OU} \rightarrow 0, \tau \rightarrow 0$ , (todo:equation) becomes  $X_{t+\Delta t}|X_t \sim N(\mu + \omega_{\Delta t}(X_t - \mu))$ , which has the same form as (todo: equation) with a scalar  $\omega_{\Delta t}$  and a shift  $\mu - \omega_{\Delta t}\mu$  applied to  $X_t$ . Thus it is  $X_t$  generated from (todo:equation) can also be interpreted as generated from (todo:equation). Similarly, As  $\sigma_{BM} \rightarrow 0$ , (todo:equation) becomes  $X_{t+\Delta t}|X_t \sim N(X_t, 0)$ , which has the same form as (todo:equation) with  $\tau \rightarrow 0$  or  $\gamma \rightarrow 1$ . Thus it is  $X_t$  generated from (todo:equation) can also be interpreted as generated from (todo:equation).

## Results

We present and analyze our results with respect to two research directions as follows.

### Research direction 1

The result of our simulations are as follows:

```
sim1_result <- read.csv("sim1_result.csv")
sim1_result <- sim1_result[sim1_result["num_multistart"]==10,]
show_col <- c("beta0", "beta1", "n_obs", "theta.mu", "theta.t", "theta.tau",
             "rmse.mu", "rmse.t", "rmse.tau")
sim1_result <- sim1_result[with(sim1_result, order(-beta1, -beta0)),]
```

**gamma=1** Below is the simulation result with  $\gamma = 1$ . We were able to get  $RMSE_{\theta} < 1\%$  and  $RMSE_{\tau} < 10\%$  but  $RMSE_t < 20\%$  only. The result shows that when  $\beta_1 = 1$  the minimum value of  $\beta_0$  to maintain a good RMSE is between 11 and 12. The result shows that when  $\beta_0 = 15$  the minimum value of  $\beta_1$  to maintain a good RMSE is between 0.8 and 1.

```
signif(sim1_result[sim1_result[, "gamma"]==1, show_col], 3)
```

##	beta0	beta1	n_obs	theta.mu	theta.t	theta.tau	rmse.mu	rmse.t	rmse.tau
## 2	15	1.0	299	10	1	1	0.0095	1.5e-01	0.049
## 19	14	1.0	299	10	1	1	0.0088	1.7e-01	0.051
## 20	13	1.0	299	10	1	1	0.0081	1.5e-01	0.049
## 21	12	1.0	299	10	1	1	0.0083	1.7e-01	0.057
## 3	11	1.0	299	10	1	1	0.3300	1.8e+02	3.200
## 4	9	1.0	299	10	1	1	0.8800	1.8e+11	3.500
## 5	15	0.8	299	10	1	1	0.0230	4.2e+00	0.570
## 6	15	0.6	299	10	1	1	0.0680	1.4e+01	1.600

**gamma=0.1** The simulation result with  $\gamma = 1$  is shown below. In this setup, we were able to get  $RMSE_{\theta} < 5\%$  and  $RMSE_{\tau} < 10\%$  but  $RMSE_t < 20\%$  only. The result shows that our model were not able to achieve good accuracy when there are only 299 observations in the dataset. However, with the number of observations being 999, the model were able to get much better *RMSE*. In this setup, the minimum value of  $\beta_0$  for good accuracy is between 9 and 11 when  $\beta_1 = 1$ , and the minimum value of  $\beta_1$  for good accuracy is between 0.9 and 1.

```
signif(sim1_result[sim1_result[, "gamma"]==0.1, show_col], 3)
```

```
##      beta0 beta1 n_obs theta.mu theta.t theta.tau rmse.mu rmse.t rmse.tau
## 7      15   1.0  299      10      10          1  0.048  3.00  0.700
## 8      15   1.0  999      10      10          1  0.016  0.15  0.070
## 22     14   1.0  999      10      10          1  0.014  0.14  0.072
## 23     13   1.0  999      10      10          1  0.014  0.14  0.065
## 24     12   1.0  999      10      10          1  0.013  0.16  0.074
## 9      11   1.0  999      10      10          1  0.015  0.18  0.084
## 10     9    1.0  999      10      10          1  0.094 19.00  1.100
## 25     15   0.9  999      10      10          1  0.015  0.29  0.110
## 11     15   0.8  999      10      10          1  0.023  1.20  0.370
## 12     15   0.6  999      10      10          1  0.022  1.70  0.580
```

**gamma=10** For  $\gamma = 10$ , as shown below, we were able to get  $\text{RMSE}_\theta < 1\%$  and  $\text{RMSE}_\tau < 10\%$  but not able to estimate parameter  $t$  nicely even with 999 observations per dataset. The lower bound for  $\beta_0$  for a good  $\text{RMSE}_\tau$  is between 11 and 12 and the lower bound for  $\beta_1$  for good  $\text{RMSE}_\tau$  is between 0.6 and 0.8.

```
signif(sim1_result[sim1_result[, "gamma"]==10, show_col], 3)
```

```
##      beta0 beta1 n_obs theta.mu theta.t theta.tau rmse.mu rmse.t rmse.tau
## 13      15   1.0  299      10      0.1          1  0.0049 1.6e+00  0.042
## 14      15   1.0  999      10      0.1          1  0.0032 1.4e+00  0.024
## 26      14   1.0  999      10      0.1          1  0.0036 1.3e+00  0.024
## 27      13   1.0  999      10      0.1          1  0.0032 1.0e+00  0.027
## 28      12   1.0  999      10      0.1          1  0.0036 1.1e+00  0.030
## 15      11   1.0  999      10      0.1          1  1.0000 5.8e+03  9.400
## 16       9    1.0  999      10      0.1          1  0.7000 1.9e+12  4.300
## 29      15   0.9  999      10      0.1          1  0.0034 1.1e+00  0.023
## 17      15   0.8  999      10      0.1          1  0.0032 1.2e+00  0.022
## 18      15   0.6  999      10      0.1          1  0.0330 4.5e+01  0.930
```

## Research direction 2

```
sim2_OU_result <- read.csv("sim2_table1.csv", check.names=FALSE)
```

```
## Warning in read.table(file = file, header = header, sep = sep, quote = quote, :
## incomplete final line found by readTableHeader on 'sim2_table1.csv'
```

```
print(sim2_OU_result, row.names = FALSE)
```

```
##      gamma 0.01 0.05 1 2 3 4
## setup (1)  0.40 0.80 1 1 1 1
## setup (2)  0.45 0.65 1 1 1 1
## setup (3)  0.45 0.55 1 1 1 1
## setup (4)  0.30 0.65 1 1 1 1
```

```
sim2_BM_result <- read.csv("sim2_table2.csv", check.names=FALSE)
```

```
## Warning in read.table(file = file, header = header, sep = sep, quote = quote, :
## incomplete final line found by readTableHeader on 'sim2_table2.csv'
```

```
print(sim2_BM_result, row.names = FALSE)
```

```
##      sigma 1.00E-06 1.00E-05 1.00E-04 1.00E-03 1.00E-02 1.00E-01 1.00E+00
## setup (1)      0.00      0.05      0.00      0.90      0.80      0.9      0.95
```

##	setup (2)	0.05	0.00	0.00	0.05	0.55	0.7	0.10
##	setup (3)	0.05	0.05	0.05	0.20	0.75	0.7	0.70
##	2.00E+00	3.00E+00						
##	1.00	0.9						
##	0.85	0.6						
##	0.90	0.6						

## Discussion

### Other Potential Models for Molecular Dynamics Study

#### Morse interaction model

Morse Interaction model is described as a combination of BM and improved OU process. Under this model,  $X_t$  is a Markov process satisfying the stochastic differential equation (SDE):

$$dX_t = -U'(X_t)dt + \sigma dB_t$$

And  $U'(x)$  is the derivative of the Morse potential energy function:

$$U'(x) = \gamma \cdot (1 - e^{-\alpha(x-u)})$$

The distance  $X_t$  is set to be strictly positive,  $X_t > 0$ . When  $X_t$  is too large, repulsive forces allow bond breaking to occur - the molecules again resemble BM. When the donor and acceptor again get close to each other, the bond is reformed and exchange of photons takes place. The dynamics then resembles OU process.

Morse interaction model is a comprehensive approach towards dynamics study of molecular interaction.

S.Schelstraete, H. Verschelde (1999). Molecular Dynamics. Retrieved from <https://www.sciencedirect.com/topics/chemistry/morse-potential>

#### Lennard-Jones Potential

The interaction between two non-bonded and un-charged atoms, known as Van der Waals interaction, has been expressed in terms of potential energy. Lennard-Jones potential is probably the most famous pair potential describing interatomic Van der Waals forces. It consists of two parts:

- 1) A steep repulsive term; and
- 2) A smooth attractive term

$$V(r) = 4\epsilon \left[ \left( \frac{\sigma}{r} \right)^{12} - \left( \frac{\sigma}{r} \right)^6 \right]$$

In particular,  $V(x)$  is the intermolecular potential between the two molecules, and  $r$  is the distance of separation between both molecules.

Apart from being a widely used model itself, Lennard-Jones potential also sometimes forms one of 'building blocks' of many force fields, due to its computational expediency.

However, Lennard-Jones potential is not designated to model donor-acceptor interactions.

Libretexts. (2019, June 5). Lennard-Jones Potential. Retrieved from [https://chem.libretexts.org/Bookshelves/Physical\\_and\\_Theoretical\\_Chemistry\\_Textbook\\_Maps/Supplemental\\_Modules\\_\(Physical\\_and\\_Theoretical\\_Chemistry\)/Physical\\_Properties\\_of\\_Matter/Atomic\\_and\\_Molecular\\_Properties/Intermolecular\\_Forces/Specific\\_Interactions/Lennard-Jones\\_Potential](https://chem.libretexts.org/Bookshelves/Physical_and_Theoretical_Chemistry_Textbook_Maps/Supplemental_Modules_(Physical_and_Theoretical_Chemistry)/Physical_Properties_of_Matter/Atomic_and_Molecular_Properties/Intermolecular_Forces/Specific_Interactions/Lennard-Jones_Potential)

## Research Direction 1

## Research Direction 2

For experiments simulated from OU, we can see a constant improvement in picking the correct model when  $\gamma$  goes up. As  $\gamma \rightarrow 0$ ,  $\omega_{\Delta t} \rightarrow 1$  and  $\tau$  is still a constant, (todo: equation) becomes  $X_{t+\Delta t}|X_t \sim N(X_t, 0)$ , which is in the same form as (todo: equation). Thus, it is difficult to distinguish the OU simulation from the BM model. As  $\gamma$  goes up,  $\omega_{\Delta t} \rightarrow 0$ , two model equations differ in mean and variance, and hence becomes easier to distinguish.

For experiments simulated from BM, we can see a general improvement in picking the correct model when  $\sigma_{BM}$  goes up, however, the improvement stops at  $\sigma_{BM} = 3$  for most of the experiments. We analyze in the following two aspects. First of all, when  $\sigma_{BM} \rightarrow 0$ , (todo: equation) becomes  $X_{t+\Delta t}|X_t \sim N(X_t, 0)$ . From the discussion in the previous paragraph, when  $\gamma \rightarrow 0$ , (todo: equation) also becomes  $X_{t+\Delta t}|X_t \sim N(X_t, 0)$ . Thus, if we do parameter inferences relatively well, it will be difficult to tell apart two models. Furthermore, when  $\sigma$  becomes too large with respect to  $\beta_0$  and  $\beta_1$ , for example, when  $\sigma = 3$  with respect to  $\beta_0 = 5$  and  $\beta_1 = 1$ , the probability of  $X_t$  being large becomes higher. In this case,  $\exp(\beta_0 - \beta_1 X_t)$  becomes close to 0, and the generated  $Y_t$  from Poission (todo: equation) is more likely to be 0. Consequently,  $Y_t$  does not provide enough information to the underlying generated process, and hence difficult to tell apart BM model from OU model.

Another observation to note is that as a general rule,  $|\beta_0 - \beta_1|$  cannot be too large, otherwise, many NAs from (todo: equation) will be generated because R cannot handle values  $> e^{\{22\}}$  (todo: check this value). Repetively generating values until no NAs results in the enter process being slow.

## Potential improvements in implementation

### Other model selection criterion

We can consider other model selection criteria for research direction 2. For example, AIC and Mallow CP. Using a weighted of different criteria for model selection makes the result less biased.

### Random start for BM model

As discussed in the methodology section, random. We figured out how to implement random start for OU model by reparameterization. Consequently, the accuracy of OU inferences has improved significantly.

### R Drawback

One of the challenges encountered in the project is the efficiency of running simulation results. Each simulation with different parameters took over 1 hour to obtain the final results. This exposes one of the disadvantages of R, which is that R does not support running test cases in parallel.

## Appendix



## References