Supporting Information: Time Series Modeling of Solute Transport in an H_{II} Phase Lyotropic Liquid Crystal Membrane

Benjamin J. Coscia Michael R. Shirts

September 18, 2019

S1 Setup and analysis scripts

All python and bash scripts used to set up systems and conduct post-simulation trajectory analysis are available online at `https://github.com/shirtsgroup/LLC_Membranes`. Documentation for the `LLC_Membranes` repository is available at `https://llc-membranes.readthedocs.io/en/latest/`. Table S1 provides more detail about specific scripts used for each type of analysis performed in the main text.

Script Name	Section	Description
`/setup/param.sh`	2.1	Parameterize liquid crystal monomers and solutes with GAFF

Table S1: The first column provides the names of the python scripts available in the `LLC_Membranes` GitHub repository that were used for system setup and post-simulation trajectory analysis. Paths preceding script names are relative to the `LLC_Membranes/LLC_Membranes` directory. The second columns lists the section in the main text where the output or usage of the script is first described. The third column gives a brief description of the purpose of each script.

S2 Choosing a transport model

We used the toolbox created by Meroz and Sokolov in order to justify our choice of transport model.[1] The solutes in our systems exhibit anomalous transport properties characteristic of a Continuous Time Random Walk (CTRW).

Mean Squared Displacement

The general form of a mean squared displacement (MSD) curve is:

$$\langle x^2(t) \rangle \sim t^\alpha \tag{1}$$

For brownian motion, $\alpha = 1$ and the MSD is linear. When $\alpha \neq 1$, the particle of interest exhibits anomalous diffusion. Values of α greater than 1 give rise to superdiffusion, while values of α less than 1 give rise to subdiffusion.

We can calculate the ensemble-averaged MSD curve by averaging the MSDs of each particle trajectory, where each MSD is calculated using:

$$\delta^2(t) = \|\mathbf{r}(t) - \mathbf{r}(0)\|^2 \tag{2}$$

where $\|\cdot\|$ represents the Euclidean norm.

The mean squared displacement of solutes in our model is a non-linear function of time, with $\alpha < 1$ which is indicative of anomalous subdiffusion. Figure S1a plots the ensemble-averaged MSD curve for 24 ethanol molecules diffusing in a 10 wt% water H_{II} LLC membrane system. We fit a power law of the form Ae^α to the MSD curve. We performed 2000 bootstrap trials by randomly sampling 24 MSD curves with replacement from the 24 total ethanol MSD curves. The bootstrapped average value of α is 0.75 for this system.

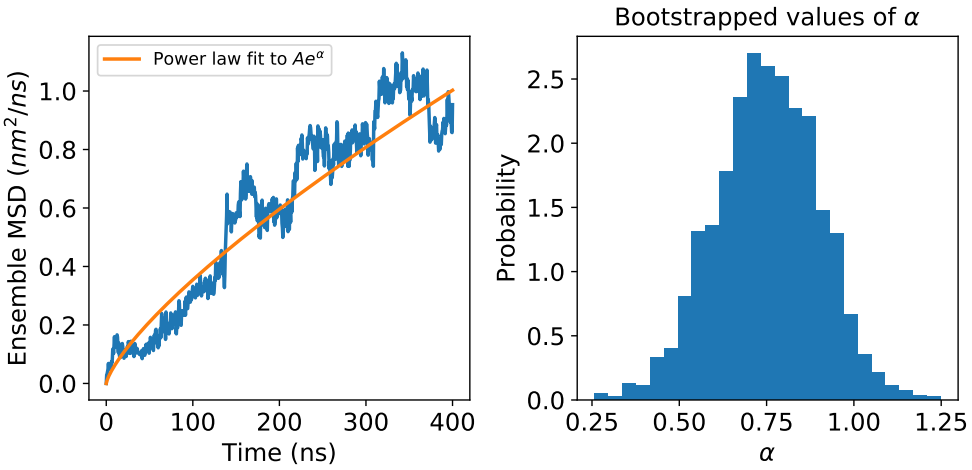


Figure S1: (a) We fit a curve with the form of Equation 1 to the ensemble-averaged MSD curve. (b) The average value of α, obtained using fits to MSDs calculated from bootstrapped ensembles, is less than 1 suggesting that ethanol molecules in our model exhibit subdiffusive behavior.

Ergodicity

The ergodicity of a system can help us narrow down the possible anomalous diffusion mechanisms. In an ergodic system, the time-averaged behavior of an observable should yield the same result as the ensemble average of the same observable. Examples of anomalous diffusion processes that are ergodic include random walks on fractals (RWF) and fractional brownian motion (FBM). Non-ergodic systems generally give rise to CTRWs with the possibility of combination with a RWF and/or FBM.[1]

We tested the ergodicity of our system by comparing the ensemble-averaged and time-averaged MSD curves. We calculated the MSD of each ethanol trajectory using Equation 2 and a time-averaged algorithm:

$$\delta^2(t) = \frac{1}{N-t} \sum_{i=0}^{N-t-1} \|\mathbf{r}(i+t) - \mathbf{r}(i)\|^2 \tag{3}$$

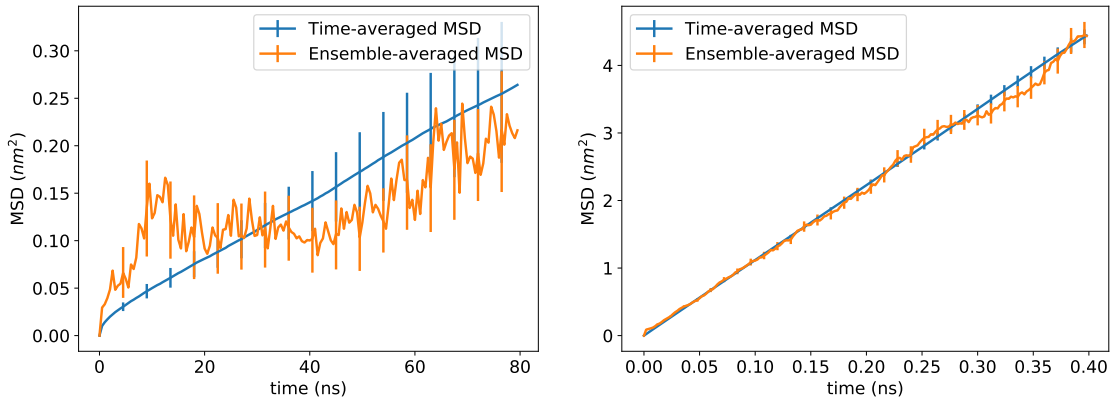
where N is the total number of simulation frames, and t represents the length of subinterval or number of frames per subinterval. We averaged the MSD curves from each trajectory in order to create final MSD plots.

The ethanol molecules exhibit non-ergodic behavior because their time-averaged and ensemble-averaged MSDs do not agree with each other (Figure S2a). We validated our analysis using a 1 ns simulation of a box of tip3p water molecules. As expected, since the particles exhibit Brownian motion, the time-averaged and ensemble-averaged MSDs agree with each within error (Figure S2b).

Autocorrelation of steps

Based on the previous two sections, our model can likey be studied as a CTRW. However, it is still possible that our CTRW model might also be convoluted with an FBM or a RWF process. In a pure CTRW, the steps are uncorrelated. Both FBM and RWF exhibit anti-correlated steps.

The steps in our system are not correlated. We showed this by calculating the autocorrelation function (ACF) of the step lengths in the z-direction. The ACF of a representative trajectory is shown in Figure S3.



(a) (b)

Figure S2: (a) The time-averaged and the ensemble-averaged MSDs for ethanol in an H_{II} nanopore are not in agreement, implying non-ergodicity. (b) A box of tip3p water molecules is expected to be ergodic and it is shown to be true here because both MSDs are in agreement.

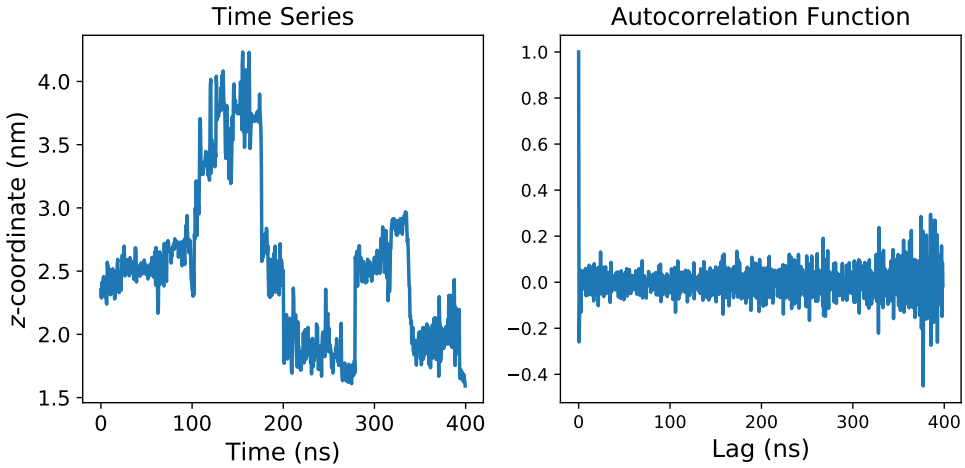


Figure S3: The autocorrelation function (right) of a representative ethanol center of mass z-coordinate trajectory (left) almost immediately decays to zero, indicating a complete loss of memory of it's previous position. Noise increases at large time lags due to decreased sampling.

S3 Estimating the Hurst Parameter

We chose to estimate the Hurst parameter, H by a least squares fit to the analytical autocorrelation function for fractional Brownian motion (the variance-normalized version of Equation 7 in the main text):

$$\gamma(k) = \frac{1}{2}\left[|k-1|^{2H} - 2|k|^{2H} + |k+1|^{2H}\right] \tag{4}$$

In Figure S4, we plotted Equation 4 for different values of H. When $H > 0.5$, Equation 4 decays slowly to zero meaning one needs to study large time lags with high frequency in order to obtain accurate estimates. Fortunately, all of our solutes show anti-correlated motion, so most of the information in Equation 4 is contained within the first few lags.

The autocovariance function of Fractional Lévy motion is different from fractional Brownian motion (see Equation 7 and 9 of the main text), but their autocorrelation structures are the same. The autocovariance

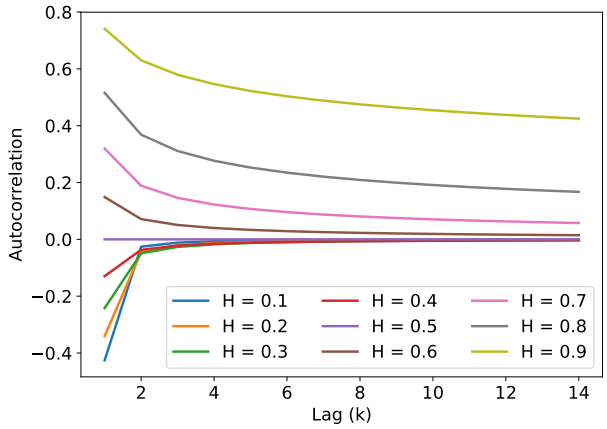


Figure S4: The analytical autocorrelation function of FBM decays to zero faster when H < 0.5 compared to when H > 0.5.

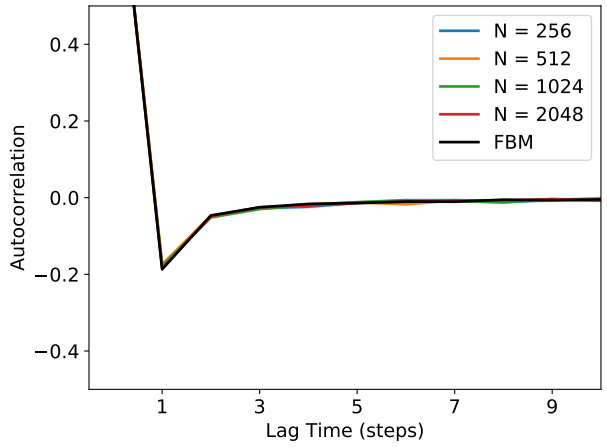


Figure S5: The autocorrelation function of an FLM process does not change with increasing sequence length (N). It shares the same autocorrelation function as fractional Brownian motion (FBM). All sequenced used to make this plot were generated using H=0.35 and α=1.4 (for FLM).

function of FLM is dependent on the expected value of squared draws from the underlying Lévy distribution, $E\left[L(1)^2\right]$. This is effectively the distribution's variance, which is undefined for most Lévy stable distributions due to their heavy tails. As a consequence, one should expect $E\left[L(1)^2\right]$ to grow as more samples are drawn from the distribution. However, we are only interested in the autocorrelation function. In order to predict the Hurst parameter from the autocorrelation function, we must show that it has well-defined structure and is independent of the coefficient in Equation 9 of the main text. In Figure **??**, we plot the average autocorrelation function from an FLM process with an increasing number of observations per generated sequence. For all simulations we set H=0.35 and α=1.4. The variance-normalized autocovariance function, i.e. the autocorrelation function does not change with increasing sequence length. Additionally, the autocorrelation function of FBM, with the same H is the same.

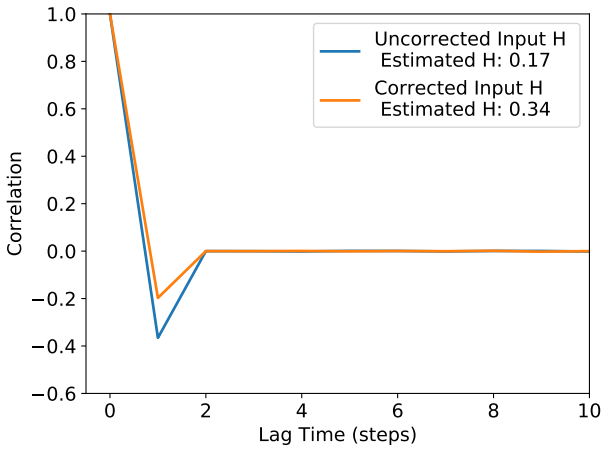


Figure S6: Correcting the Hurst parameter input to the algorithm of Stoev and Taqqu results in an FLM process with a more accurate correlation structure. We generated sequences with an input H of 0.35. We estimated H by fitting the autocorrelation function. Without the correction, H is underestimated, meaning realizations are more negatively correlated than they should be.

S4 Simulating Fractional Lévy Motion

Achieving the right correlation structure

We simulated FLM using the algorithm of Stoev and Taqqu [2]. There are no known exact methods for simulating FLM. As a consequence, passing a value of H and α to the algorithm does not necessarily result in the correct correlation structure, although the marginal Lévy stable distribution is correct. We applied a database-based empirical correction in order to use the algorithm to achieve the correct marginal distribution and correlation structure.

Stoev and Taqqu note that the transition between negatively and positively correlated draws occurs when $H = 1/\alpha$. When $\alpha = 2$, the marginal distribution is Gaussian and $H = 0.5$ as expected from FBM. We corrected the input H so that the value of H measured based on the output sequence equaled the desired H. We first adjusted the value of H by adding $(1/\alpha - 0.5)$, effectively recentering the correlation sign transition for any value of $1 \leq \alpha \leq 2$. This correction alone does a good job for input H values near 0.5, but is insufficient if one desires a low values of H. The exact correction to H is not obvious so we created a database of output H values tabulated as a function of input H and α values. Figure S6 demonstrates the results of applying our correction. Without the correction, FLM realizations are more negatively correlated. This would result in under-predicted mean squared displacements when applying the model.

FLM realizations from truncated Lévy stable distributions

To generate realizations from an uncorrelated truncated Lévy process, one would randomly sample from the base distribution and replace values that are too large with new random samples from the base distribution, repeating the process until all samples are under the desired cut-off.

This procedure is complicated by the correlation structure of FLM. At a high level, Stoev and Taqqu use Riemann-sum approximations of the stochastic integrals defining FLM in order to generate realizations. They do this efficiently with the help of Fast Fourier Transforms. In practice, this requires one to Fourier transform a zero-padded vector of random samples drawn from the appropriate Lévy stable distribution, multiply the vector in Fourier space by a kernel function and invert back to real space. The end result is a correlated vector of fractional Lévy noise.

If one is to truncate an FLM process, one can apply the simple procedure above for drawing uncorrelated values from the marginal Lévy stable distribution, but after adding correlation, the maximum drawn value is typically lower than the limit set by the user. Additionally, the shape of the distribution itself changes.

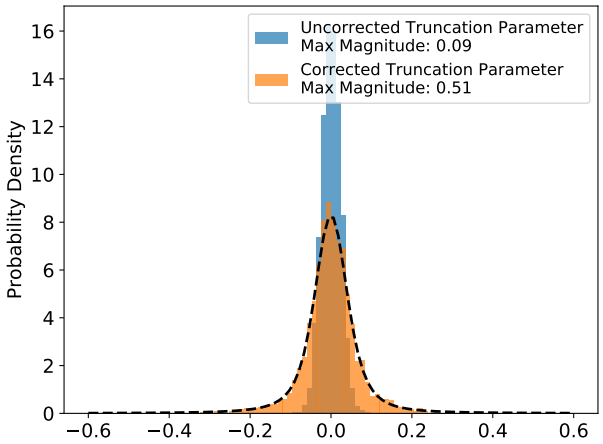


Figure S7: We can accurately truncate the marginal distribution of FLM innovations by applying a correction to the input truncation parameter. We generated FLM sequences and truncated the initial Lévy stable distribution (before Fourier transforming) at a value of 0.5. After correlation structure is added, the width of the distribution of fractional Lévy noise decreases significantly. We corrected the input truncation parameter with our database resulting in a much more accurate distribution with a maximum value close to 0.5.

Analogous to the database used to correct the Hurst parameter, we created a database to correct the input truncation parameter (the maximum desired draw). The database returns the value of the truncation parameter that will properly truncate the output marginal distribution based on H, α and σ (the width parameter). Figure S7 shows the result of applying our correction.

References

[1] Y. Meroz and I. M. Sokolov, "A Toolbox for Determining Subdiffusive Mechanisms," *Phys. Rep.*, vol. 573, pp. 1–29, Apr. 2015.

[2] S. Stoev and M. S. Taqqu, "Simulation methods for linear fractional stable motion and farima using the fast fourier transform," *Fractals*, vol. 12, pp. 95–121, Mar. 2004.